Journal of Artificial Intelligence Research 65 (2019) 31–85 Submitted 09/2018; published 05/2019

A Coupled Operational Semantics for Goals and Commitments

Pankaj R. Telang PTELANG@GMAIL.COM
SAS Institute Inc.
100 SAS Campus Dr., Cary, NC 27513, USA

Munindar P. Singh SINGH@NCSU.EDU
North Carolina State University
Raleigh, NC 27695, USA

Neil Yorke-Smith N.YORKE-SMITH@TUDELFT.NL
Delft University of Technology
& American University of Beirut
2600GA Delft, The Netherlands

Abstract

Commitments capture how an agent relates to another agent, whereas goals describe states of the world that an agent is motivated to bring about. Commitments are elements of the social state of a set of agents whereas goals are elements of the private states of individual agents. It makes intuitive sense that goals and commitments are understood as being complementary to each other. More importantly, an agent's goals and commitments ought to be coherent, in the sense that an agent's goals would lead it to adopt or modify relevant commitments and an agent's commitments would lead it to adopt or modify relevant goals. However, despite the intuitive naturalness of the above connections, they have not been adequately studied in a formal framework. This article provides a combined operational semantics for goals and commitments by relating their respective life cycles as a basis for how these concepts (1) cohere for an individual agent and (2) engender cooperation among agents. Our semantics yields important desirable properties of convergence of the configurations of cooperating agents, thereby delineating some theoretically well-founded yet practical modes of cooperation in a multiagent system.

1. Introduction and Motivation

Whereas the study of goals is a long-standing theme in autonomous agents, the last two decades have seen the motivation and elaboration of a theory of (social) commitments. The concepts of goals and commitments are intuitively complementary: a commitment describes how an agent relates with another agent, whereas a goal describes a state of the world that an agent is motivated to bring about. A commitment from one agent (its debtor) to another (its creditor) states that the debtor promises to achieve the consequent if the creditor or another agent (first) achieves the antecedent. A commitment carries normative or deontic force in terms of what an agent would bring about for another agent, whereas a goal describes an agent's proattitude towards some condition.

Commitments have been extensively applied in the formulation of agent communication, especially from the standpoint of multiagent systems with a view to promoting openness by capturing elements of the social state (Fornara & Colombetti, 2002; Maudet & Chaib-

Draa, 2002; Singh, 1998; Verdicchio & Colombetti, 2003). Commitments have been used as a foundation for the study of agent organizations and institutions, most notably in the work of Fornara, Colombetti, and their colleagues (Fornara & Colombetti, 2009; Fornara, Viganò, Verdicchio, & Colombetti, 2008), but also in related forms in the work of V. Dignum and F. Dignum and their colleagues (Aldewereld & Dignum, 2010; V. Dignum, Dignum, & Meyer, 2004). At the same time, goals provide an effective high-level way to characterize the states and behaviours of individual agents (van Riemsdijk, Dastani, & Winikoff, 2008). Goals are conceptually simpler than both desires (in imposing consistency) and intentions (in avoiding considerations of know-how or strategies) (Singh, 1994).

Developing a unified theory of commitments and goals would be significant for the following two reasons. First, it would close the theoretical gap in present understanding between the organizational and individual perspectives, which are both essential in a comprehensive account of rational agency in a social world. Second, it would provide a basis for a comprehensive account of the software engineering of multiagent systems going from interaction-orientation (with commitments) to agent-orientation (with goals).

Whereas a goal is specific to an agent, a commitment involves a pair of agents. On the one hand, an agent may create commitments toward other agents in order to achieve its goals. On the other hand, an agent may consider goals in order to fulfil its commitments to other agents. Without appropriate reasoning rules, a related goal and commitment may not cohere. As an example, if the commitment created by an agent for achieving a goal expires, then the agent's goal may fail. Assuming that the agent does not want the goal to fail, then the agent should reason and, as a result, possibly create a new commitment. We develop a set of *practical rules* of reasoning that, under certain conditions, guarantee convergence of the agent's commitments and goals, in a sense that we will make precise.

Given the importance of the concepts, it is no surprise that researchers, e.g., Chopra, Dalpiaz, Giorgini, and Mylopoulos (2010a); Günay, Liu, and Zhang (2016); Meneguzzi, Telang, and Singh (2013), have begun tying the concepts of goals and commitments together. We go beyond existing work by developing a foundational, formal approach.

1.1 Contributions

In brief, this article makes the following contributions. First, it provides a formalization of the combined life cycles of commitments and goals. Second, this article provides a way to formalize a variety of practical rules of reasoning by which agents may reason about their commitments and goals in tandem. Such rules characterize the cooperative interactions of the agents, and can be treated as patterns of reasoning. An interesting outcome of this formalization is that it shows some of the limits of what we can conclude given that the agents are autonomous. In general, because agents may terminate their commitments or goals, convergence to certain 'good' states cannot be assured. However, we are able to show positive results when we introduce assumptions that suitably constrain the autonomy of the agents. Third, this article provides a methodology for reasoning about sets of practical rules so as to verify that those sets satisfy important properties, chiefly the property of convergence. In this manner, this methodology supports the development of sets of practical rules that are suited to specific domains and cooperative environments.

This article builds on a preliminary workshop paper (Telang, Singh, & Yorke-Smith, 2012) that introduced the problem of a coupled semantics for goals and commitments. This article develops a complete operational semantics built on a formal framework for agent operations and states that we introduce here, a more extensive treatment of correctness properties, analytical proofs of the properties, and an application on a well-known case study. We note that the article also identifies gaps and some errors in the rules of Telang et al. (2012).

1.2 Outline

We begin in Section 2 by introducing the concepts of commitments and goals, and for each presenting their life cycle as a state transition diagram. Section 3 presents our combined operational semantics, which is based on guarded rules. We term these *practical* rules because they capture patterns of practical reasoning that an agent may adopt. That is, an agent may choose to follow or not to follow any of these rules in order to achieve certain desirable properties. These rules apply on top of the life cycle of goals and commitments. In Section 4, we state and prove convergence properties for agents that adopt our practical rules. Section 5 places our work in context. We conclude in Section 6 with a discussion of some promising research directions.

2. Background

This section consolidates the salient background needed for understanding our approach. We begin by describing social commitments, and then describe achievement goals.

For both commitments and goals, the life cycle transitions—introduced below—occur when the specified conditions come about. To focus on our main contributions and avoid the complexity brought on by temporal operators, we refine the idea proposed by Singh (2008, p. 180) to model the atomic propositions as *stable* atomic propositions, which are such that once true, they remain true forever. Note however that in general, a proposition that is false is not forever false. For concreteness, assume the atomic propositions can be temporally qualified with deadlines. For example, we may represent the following atomic proposition: "The package has arrived by 11am." The proposition is initially false but has the possibility of becoming true. It can become true as soon as the package arrives, provided the package arrives by 11am, and once true it remains true. However, if 11am passes and the package has not arrived, there is no longer a possibility of this atomic proposition becoming true. That is, the proposition goes from being merely false to becoming forever false. In general, we can express each proposition such that any negation applies only to atomic propositions (Singh, 2008). In such a form, any proposition that includes conjunction or disjunction (but not negation) is stable whereas a proposition that involves negation is generally not stable. Chopra and Singh (2015a) have worked out the computational aspects of deadlines.

2.1 Commitments

A *commitment* expresses a social or organizational relationship between two agents. This sense of commitments was defined by Singh (1991) and adopted by Castelfranchi (1995): its key feature is that it relates one agent to another and thus contrasts with an agent's

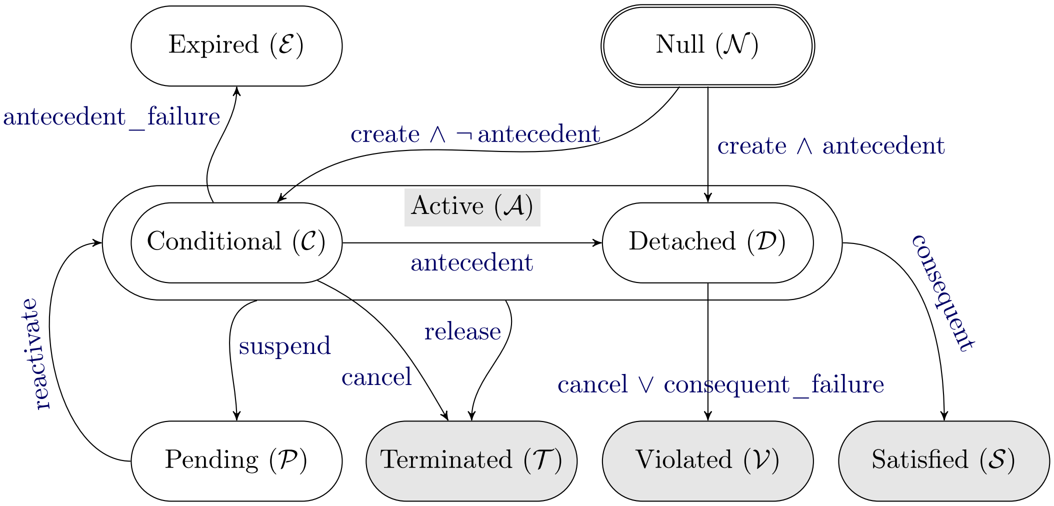


Figure 1: Commitment life cycle as a state transition diagram.

commitment to its intention (Bratman, 1987). Singh (2012) provides additional motivation and historical background on our view of commitments. Specifically, a commitment $C =$ C(DEBTOR, CREDITOR, antecedent, consequent) denotes that the DEBTOR commits to the CREDITOR for bringing about the consequent if the antecedent becomes true (Singh, 1999). We write $ant(C)$ to denote the antecedent of commitment C and $cons(C)$ to denote its consequent.

Figure 1 shows the life cycle of a commitment, simplified from Telang and Singh (2012). For simplicity, we disregard commitment delegation or assignment, as well as the notion of an organizational context, since they are not essential to our present contribution. In Section 3.2, we discuss adding these aspects to the life cycle. A labelled rounded rectangle represents a commitment state, and a directed edge represents a transition; transitions are labelled with the corresponding action or event. A double-rounded rectangle indicates an initial state. The terminal states are highlighted in grey.

A commitment can be in one of the following states: `Null` (before it is created), `Conditional` (when it is initially created), `Expired` (when its antecedent becomes forever false, while the commitment was `Conditional`), `Satisfied` (when its consequent is brought about while the commitment was `Active`, regardless of its antecedent), `Violated` (when its antecedent has been true but its consequent will forever be false, or if the commitment is cancelled when `Detached`), `Terminated` (when cancelled while `Conditional` or released while `Active`), or `Pending` (when suspended while `Active`). `Active` has two substates: `Conditional` (when its antecedent is false) and `Detached` (when its antecedent has become true). A debtor may create, cancel, suspend, or reactivate a commitment; a creditor may release a debtor from a commitment.

We consider commitments whose antecedents and consequents are propositions. Note that we specify the *truth of a condition*—and our approach is neutral as to who brings about a condition—as opposed to the performance of an action by an agent. In general, focusing on conditions facilitates greater flexibility during enactment. Further, for some commitments, the creditor may adopt a goal to bring about the antecedent. However, in other commitments, the creditor may not. For example, an insurance company may commit to paying a car driver's medical bills if the driver is injured in an accident: the driver would not ordinarily have a goal to get injured! For these reasons, we keep the life cycle of commitments general and enable agents in different settings to exercise that life cycle in ways that best make sense to them in the appropriate context.

Second, note that, in general, commitments need not be symmetric, i.e., reciprocal. That is, in general, an agent may have a commitment to another agent without the latter having a converse commitment to the former agent. For example, when a merchant commits to a customer to providing a coffee for \$9 that does not mean the customer commits to paying \$9 for a coffee. Singh (2012) discusses such properties of commitments at length.

Third, note that we do not include penalties within a commitment but would capture them separately. In the literature on commitments, a penalty is customarily handled from the organizational context (Bulling & Dastani, 2016; Singh, Chopra, & Desai, 2009). The organizational context refers to the organization within whose scope the given commitment arises (Singh, 1999). A classical example is an online marketplace within whose scope and regulations a buyer and a seller enter into commitments (namely, the seller to ship the goods in question and the buyer to pay the seller). If the buyer does not pay for an auction she won, then the marketplace can penalize her in various ways, including closing her account. Our chosen setting of autonomous agents contrasts with a conceptually centralized, *regimented* system wherein the 'system' can prevent the violation of the applicable commitments (see the discussion in, e.g., Boella, Broersen, and van der Torre, 2008). A pertinent example of a regimented system is the use of 'proxy' (more properly, controller) agents in AMELI/ISLANDER (Sabater-Mir, Pinyol, Villatoro, & Cuni, 2007) that prevent agents from performing forbidden actions. In effect, the forbidden action never occurs and therefore no compensation or penalty is needed to undo or mitigate the effects of the forbidden action.

Lastly, as Section 3.2 will discuss, we assume that agents communicate synchronously. Synchrony prevents race conditions between agents affecting the same commitment. Inter-agent communication arises for commitments due to their public nature; this is one of the key differences between commitments and goals.

2.2 Goals

A *goal* expresses a state of the world that an agent wishes to bring about. Our conception of goals follows Harland, Morley, Thangarajah, and Yorke-Smith (2014) and Harland, Morley, Thangarajah, and Yorke-Smith (2017).[1] Goals differ from both desires and intentions. An agent's desires represent the agent's proattitudes[2] (Rao & Georgeff, 1992); an agent may concurrently hold mutually inconsistent desires. By contrast, it is customary to require that

[1] Their formulation of goals includes also a *precondition* (or context) that must be true before a goal G can become `Active` and some intention can be adopted to achieve it, and a *post-condition* (or effect) that becomes true if G is successfully achieved. Pre- and post-conditions of goals do not have a direct bearing on our semantics and we need not treat them; see also Günay, Winikoff, and Yolum (2012). We also do not follow Thangarajah, Harland, Morley, and Yorke-Smith (2011)'s inclusion of an *in-condition* that is true once a goal is `Active` until its achievement.

[2] That is, an agent's mental attitude directed towards an action (Davidson, 2001).

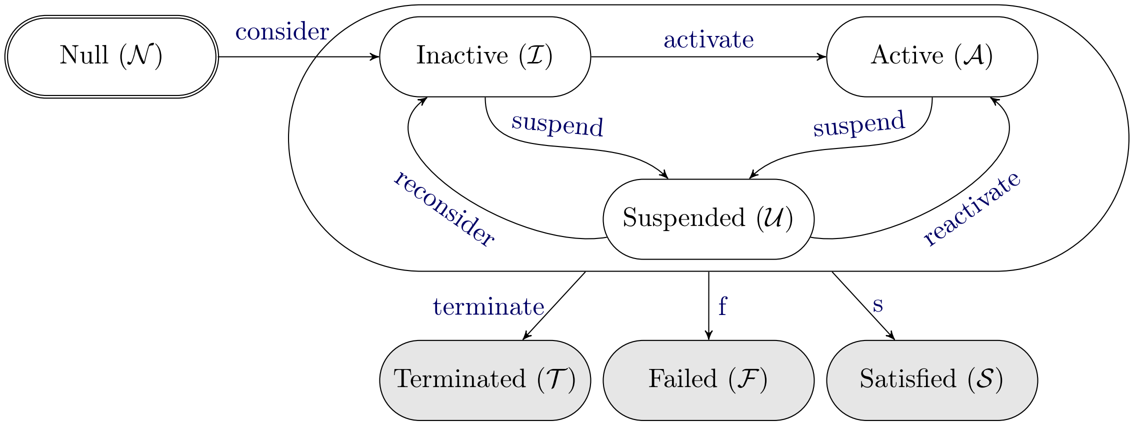


Figure 2: Simplified life cycle of an achievement goal as a state transition diagram.

a rational agent believe its goals are mutually consistent (Winikoff, Padgham, Harland, & Thangarajah, 2002). An agent's intentions are its adopted or activated goals.

Specifically, a goal $G = \mathsf{G}(x, s, f)$ of an agent x has a *success condition* s that defines the success of G, and a *failure condition* f that defines its failure. A goal is successful if and only if s becomes true prior to f: that is, the truth of s entails the satisfaction of the goal only if f does not intervene. Note that s and f should be mutually exclusive. We write $succ(G)$ to denote the success condition of a goal G, i.e., $succ(G) = s$.

As for commitments, the success or failure of a goal depends only on the truth or falsity of the various conditions, not on which agent brings them about.

Figure 2 simplifies Harland et al.'s (2014) life cycle of an achievement goal (we do not consider maintenance goals in this article). A goal can be in one of the following states: `Null`, `Inactive`,[3] `Active`, `Suspended`, `Satisfied`, `Terminated`, or `Failed`. The last three collectively are *terminal states*: once a goal enters any of these states, it stays there forever. Note how both commitment and goal life cycles have `Satisfied` and `Failed` states, and also have `Null` and `Active` states. The semantic rules in the text below link the definition of a goal and its states.

Before its creation, a candidate goal is in state `Null`. Once considered by an agent (its '*goal holder*'), a goal commences as `Inactive`. Upon activation, the goal becomes `Active`; the agent may pursue its satisfaction by attempting to achieve s. If s is achieved, the goal transitions to `Satisfied`. At any point, if the failure condition of the goal becomes true, the goal transitions to `Failed`. At any point, the goal may become `Suspended`, from which it may eventually return to an `Inactive` or `Active` state appropriately. Lastly, the agent may terminate the goal at any point,[4] thereby moving it to the `Terminated` state.

It is worth remarking here on a subtle but important point. Although we represent commitment and goal life cycles using the same notation, they are conceptually quite different

[3] Renamed from `Pending` to avoid conflict with the commitments nomenclature. Although goal and commitment state names could be reconciled further, we have made the minimal change: we retain all other names unchanged to facilitate comparison with the literature.

[4] We combine the drop or abort transitions of Harland et al. (2014) since we do not need to distinguish them in our semantics.

in that a goal represents an element of a private state whereas a commitment represents an element of a social state. Thus a commitment comes into being in state `Active` the moment it is created (typically due to communications and based on the social norms in play) whereas for a goal, an agent may mull it over. In other words, a commitment is created via an atomic public event whereas a goal, being private, can be created `Inactive` and subsequently transition to `Active`. We return to this point in Section 3.3.

We also remark that the careful analysis of states and transitions of a goal life cycle, demanded by our coupled commitment–goal operational semantics, has the side benefit of clarifying minor points in the operational semantics for goals themselves. For example, in some cases, the literature is ambiguous about the possibility of the simultaneous truth of the success and failure conditions of a goal, and its semantic implication (if permitted). Our semantics is explicit in disallowing states in which both the success and failure conditions of a goal simultaneously become true.[5]

As discussed earlier, a conceptual relationship is established between a goal and a commitment when they reference each other's objective conditions. Even when related in such a manner, however, a goal and a commitment independently progress in accordance with their respective life cycles. For example, when agent y brings about condition s—an objective condition—a commitment $C = \mathsf{C}(x, y, s, u)$ detaches, and a goal $G = \mathsf{G}(x, s, f)$ is satisfied. That is, C has s as a condition (its antecedent) and G has s as a condition (its success condition), but C and G progress independently through their life cycle when s comes about.

Further, notice that goals are private to each agent: no agent may inspect the goals of another. However, a commitment, being an element of the social state, is represented in both its creditor and its debtor. The rules we introduce in our operational semantics apply to each agent's internal representation separately. Further, we emphasize that the agents *do not* agree upon any actions. Each agent affects its goals and commitments (i.e., of which it is the debtor) unilaterally; no agent can commit another agent (Singh, 2012).

2.3 Contributions Summarized

Our formal operational semantics, presented in the next sections, adds value to the understanding of intra-agent deliberation and inter-agent collaboration, and to the specification and implementation of agent systems. Our formalization captures the combined life cycles of commitments and goals, and provides a set of practical rules by which agents may reason about their commitments and goals in tandem. Further, under suitable constraints about the autonomy of the agents, we analytically prove results about the coherence of commitments and goals in the multiagent system. The practical benefit of our contribution to the agent designer is seen in automatic protocol generation (Meneguzzi, Magnaguagno, Singh, Telang, & Yorke-Smith, 2018) and high-level agent programming of Belief-Desire-Intention (BDI) style agents with social state (Baldoni, Baroglio, Capuzzimati, & Micalizio, 2015).

Commitments find ready application in domains that emphasize agent autonomy and heterogeneity, such as the aerospace aftermarket (Desai, Chopra, & Singh, 2009) and healthcare (Meneguzzi et al., 2018). In these domains, conflicts between commitments and goals can arise. Whereas goal conflicts are not our direct interest here (see, e.g., Thangarajah and

[5] We note that it is possible to have goals such as $\mathsf{G}(x, s \wedge \neg f, f \wedge \neg s)$ and similar variations if desired.

Padgham, 2011), the results of this article ensure, under specified conditions, that an agent maintains commitment–goal consistency. Further, our semantics helps to exclude certain types of goal conflicts. Suppose two agents, say x and y, have mutually conflicting goals. For example, suppose we have $\mathsf{G}(x, \mathsf{eat\text{-}cake})$ and $\mathsf{G}(y, \mathsf{eat\text{-}cake})$ and there is a single slice of cake. Now if x or y has the capability to achieve its goal, it can simply do so. If the agents lack the capability, or choose not to achieve the goals themselves, then they might make a commitment to another agent (following the reasoning enabled by our practical rules)—say agent z who has the cake. Hence the system would then have two commitments, say $\mathsf{C}(x, z, \mathsf{give\text{-}cake\text{-}to\text{-}x}, \mathsf{pay\text{-}dollar})$ and $\mathsf{C}(y, z, \mathsf{give\text{-}cake\text{-}to\text{-}y}, \mathsf{pay\text{-}dollar})$. Now supposing z will not adopt mutually conflicting goals, then z cannot detach both commitments. How z deliberates over which goal to adopt of $\mathsf{G}(z, \mathsf{give\text{-}cake\text{-}to\text{-}x})$ and $\mathsf{G}(z, \mathsf{give\text{-}cake\text{-}to\text{-}y})$—if either—is part of agent reasoning which we do not aim to treat in this article. The point is that at most one of the conflicting commitments will be detached along any execution path. A formal treatment of conflicts like the foregoing is part of future work discussed in Section 6.

In addition to the formal semantics, we propose a simple methodology for dealing with bespoke formulations of commitments and goals. The methodology generalizes what we demonstrate above, which does not depend on the particular commitment and goal life cycles (Figures 1 and 2).

In simple terms, the modeller would proceed as follows to accommodate the needs of a particular domain.

- Specify a model for commitments along with its life cycle characterized via a life cycle rule definition that specifies under what actions what parts of a commitment change, if any. That is, provide an alternative to Figure 1.
- Specify a model for goals along with its life cycle, characterized via a rule, as for commitments. That is, provide an alternative to Figure 2.
- Specify a set of practical rules capturing appropriate social reasoning patterns for the chosen domain. That is, provide a set of practical rule templates as an alternative to Section 3.4.1.

Our methodology then prescribes that we:

- Identify an agent's configuration in terms of its beliefs, goals, and commitments, as represented by state functions for each of these three elements, as in Section 3.3.6.
- Define the configuration of the multiagent system in terms of agents' beliefs, goals and commitments, as in Section 3.3.7.
- Ensure how the life cycle rule maintains consistency of the configuration over their transitions, i.e., no conflicts occur between an agents' beliefs and commitments, and between its beliefs and goals, as in Lemmas 1 and 2.
- Study the properties that follow from the practical rules, as in Section 4.

Then, depending on the set of practical rules and the other components above, properties about the execution traces of the multiagent system can be established. In particular, with the models of commitments and goals from the literature that we adopt in this article, and the—in our experience—reasonable set of practical rules, our approach ensures coherence between agents' commitments and goals.

Although we select models of commitments and goals and social reasoning patterns in this paper, our methodology is more general. For instance, an alternative formulation of the commitment life cycle that included explicit acceptance by the creditor of proposed commitments, would result in a different set of commitment actions than in this article, but the methodology to develop the operational semantics would hold the same.

3. Operational Semantics

As we have observed, whereas a goal is specific to an agent (but see Section 5), a commitment involves a pair of agents. On the one hand, an agent may create commitments towards other agents in order to achieve its goals. On the other hand, an agent may consider goals in order to fulfil its commitments to other agents. In general, the antecedent and consequent of a commitment can, in theory, refer to goals and commitments explicitly. However, we hold that it would be conceptually unclear to posit a commitment whose antecedent or consequent is a goal, since a commitment captures a *public relationship* between two agents whereas a goal captures a *private state* of one of the agents. Therefore, we consider commitments whose antecedent and consequent are objective conditions, which might also be the success conditions of one or more goals. Such a view agrees with Chopra, Dalpiaz, Giorgini, and Mylopoulos's (2010b) representation.

In this section our aim is to establish all the possible actions that a rational agent would be able to do, with respect to its commitments and goals, rather than specifying what it actually *chooses* to do. That is, our purpose in the operational semantics is to formalize what the agent may choose to do, as distinct from what it does choose to do.

3.1 Agent Architecture

We consider agents who follow a simple architecture, as depicted in Figure 3. We discuss the architecture informally here and formalize it below. The intent of the figure is to show a simple agent architecture for expository purposes, not to present a new variant of BDI architecture.

Each agent maintains a set of beliefs, goals, and commitments, denoted by SMALL CAPS labels. The agent executes iteratively in a control loop. Based on its perception of the environment, the agent updates its beliefs and updates the goal and commitment states according to their life cycles. The agent then executes the *practical rules* of reasoning that apply. These practical rules, described in Section 3.4, capture patterns of pragmatic reasoning that agents may or may not adopt under different circumstances. In that sense, practical rules are the rules of an agent program. They are specified by the designers of the agents.

The practical rules apply according to the state of a commitment, a goal, or a commitment–goal pair. For each commitment and each goal, the agent selects at most one of the applicable practical rules to execute. Each selected practical rule can yield one or more possible actions; from the set of actions, the agent selects at most one action for each commitment and each goal. For example, the agent is not allowed to simultaneously select two practical rules on

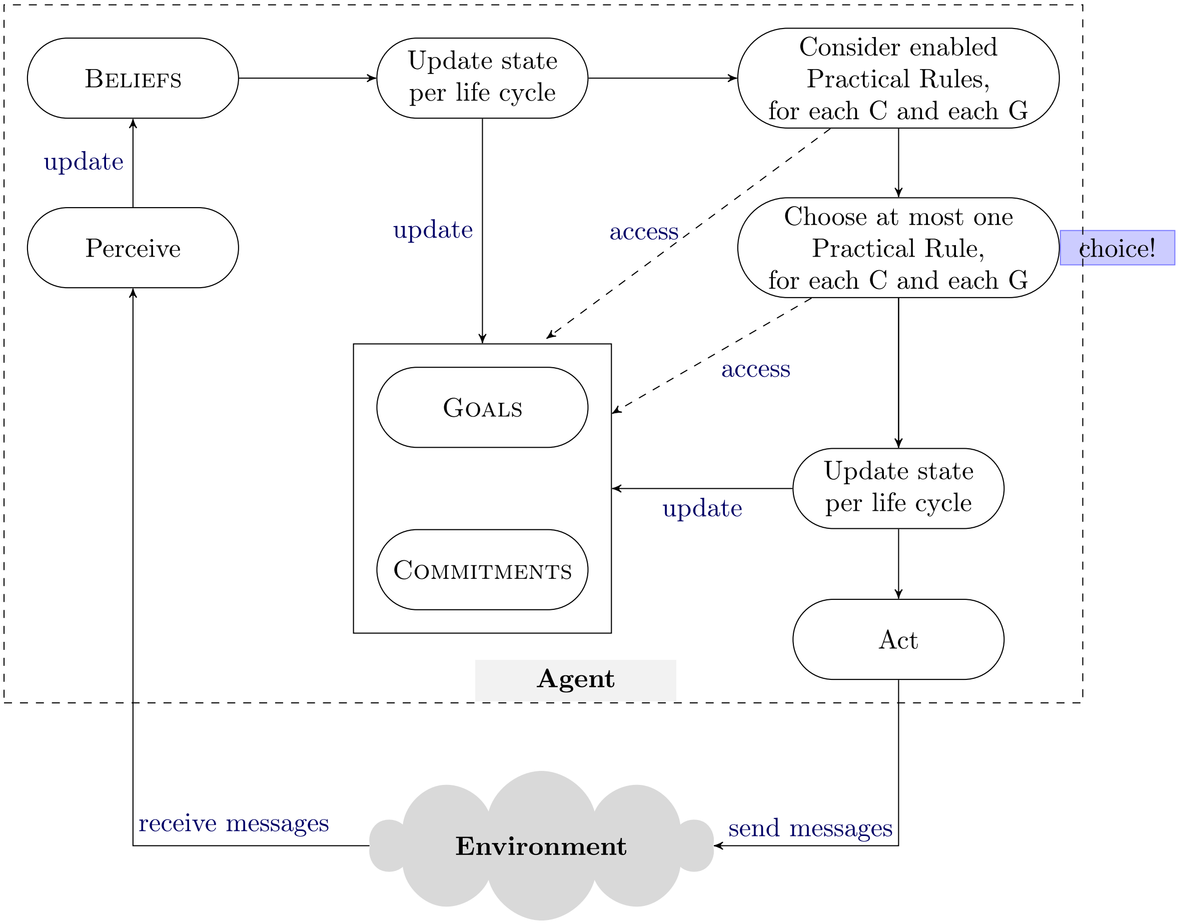

Beliefs
Update state per life cycle
Consider enabled Practical Rules, for each C and each G
update
Perceive
update
access
Choose at most one Practical Rule, for each C and each G
choice!
Goals
access
Update state per life cycle
update
Commitments
Act
Agent
receive messages
Environment
send messages

Figure 3: Simple agent architecture.

a commitment, one to cancel and the other to suspend it. Each selected action corresponds to at most one transition in the life cycle of each commitment and each goal and results in updating the state of any affected commitment or goal, i.e., the agent's beliefs about them. All such transitions are executed in parallel.

Next, after the practical rule selection and subsequent goal and commitment state updates, the agent proceeds with 'standard' BDI deliberation about goals and intentions (or plans), such as plan selection (Myers & Yorke-Smith, 2005; Rao & Georgeff, 1992; Visser, Thangarajah, Harland, & Dignum, 2016). This can result in goal (and plan) state updates, i.e., changes to the agent's beliefs about them, and is not shown separately in the figure.

Finally, in addition to modifying its own state, the agent acts to modify the environment by sending messages. These agent environmental 'actions', whether from practical rules, plan actions, or otherwise, are shown occurring together in the 'Act' box in the figure. In a message, the agent communicates its action on a commitment, or its (attempt of) bringing about an objective condition in the environment. The control loop repeats with the next perception. The success of environmental actions (and corresponding updates to agents' beliefs) is perceived at the start of the next execution cycle.

3.2 Assumptions

Here we collate the assumptions we make in this article:

1. The propositions incorporate the necessary quantitative or qualitative time specification that ensures stability. As discussed in Section 2, whereas a traditional time-dependent proposition is “the door is open” (may change from true to false), a stable proposition is “the door was opened after the courier rang the bell” (once true it doesn’t become false). Another stable proposition would involve deadlines. An unconditional commitment to achieve p would be satisfied when p holds and would be violated when it is no longer possible for p to be true. For example, as before, let p be “the package has arrived by 11am.” Then, the commitment to achieve p is satisfied if the package arrives by 11am and is violated if the package has not arrived by 11am.

2. We consider commitments whose antecedents and consequents are propositions. This assumption simplifies the presentation and enables us to focus on the essentials of the semantics.

3. We disregard three aspects of commitments studied in the literature: timeouts, commitment delegation or assignment, and the notion of an organizational context, since they are not essential to our present contribution. Timeouts, and indeed temporal commitments and goals, add both realism and complexity (see, for instance, Marengo et al., 2011). We defer a treatment of delegation and assignment to future work. The organizational context is an important consideration and considering commitments and goals in such a larger context is a significant area for future investigation.

4. We consider only achievement goals, deferring maintenance goals to future work. Achievement goals are the most common form of goals in the literature.

5. We take goals as the sources of commitments. Although we recognize that there are alternative social psychological and philosophical positions, the rationalist assumption of goal-driven agents is appropriate for our purposes. An alternative position would be that agents are primarily social creatures, whose goals are based on the commitments they find themselves as having undertaken when they join a society. A modification of our approach would tackle such a setting. Yet another setting is that the goals are not autonomously acquired but result from some phenomenon such as imitation. For example, a teenager may want the smartphone model all his friends have. Our approach is equally applicable to this setting.

6. As in previous work on commitments (with notable exceptions such as Chopra and Singh; Chopra and Singh, 2009; 2015b), we assume for simplicity that the agents communicate synchronously. Synchronous communication simplifies *alignment*, meaning that when a creditor represents a commitment from a debtor, the debtor represents that commitment (to that creditor). Under the assumption of synchrony, each commitment is represented in the same state by both its debtor and creditor, and alignment is trivial. This assumption is standard in the multiagent systems literature and corresponds to interposing a common entity such as a commitment store or blackboard through which they interact. Lifting the assumption, however, is not trivial and is a relevant topic for future work on the interplay of goals and commitments.

3.3 Formalization

Recall that we wish to characterize the interplay between an agent's goals and commitments, and how the ensuing interactions of the agents belonging to a multiagent system serve to characterize the multiagent system as a whole. A multiagent system is not a separate executing entity from its constituent agents. To this end, the underlying intuition expressed in our operational semantics is to describe how a multiagent system moves from one configuration to the next in terms of the movements of its constituent agents across their respective configurations.

The configuration of an agent is defined precisely in terms of three elements, namely, its beliefs, goals, and commitments. We ensure that each configuration corresponds to a meaningful information model by imposing appropriate properties on these elements. Specifically, first, beliefs, goals, and commitments respect certain closure properties. For example, an agent who has a goal $p \wedge q$ must have a goal p and a goal q. Second, these elements are mutually consistent. For example, if a goal to achieve p is satisfied then a commitment whose antecedent is p is also detached (modulo technical conditions such as that it is not already discharged). Third, the transitions between the states of a goal or commitment are precisely specified. The life cycle rule of Definition 29 (on page 52) describes how the various elements of an agent configuration are updated in response to events. In essence, each part of the life cycle rule takes, when it is instantiated, a left hand side (current configuration) to a right hand side (successor configuration). The updates are modular in that whereas beliefs affect goals and commitments, an action on a goal or a commitment affects only that goal or commitment plus any goals or commitments that are affected because of the closure properties.

The life cycle rule thus characterizes how beliefs, goals, and commitments progress: specifically, how changes in an agent's beliefs affect its goals and commitments and how an agent's goal and commitment actions respectively affect its goals and commitments. The RHS of a case within the life cycle rule *does* affect the multiagent system's configuration. The various parts of the life cycle rule make sure that the above-mentioned properties are preserved and that a unique configuration results from the application of a life cycle rule (as shown in Lemma 1). These properties are formalized below.

Lastly, each agent's potential decision making is characterized through practical rules, each of which applies in possible configurations of the agent and yields an action that the agent may perform. For a practical rule, the LHS is the current configuration and the RHS is a putative set of actions on goals or commitments. Thus the RHS of a practical rule does *not* directly affect the multiagent system's configuration. Updates to beliefs are not part of the RHS because they happen in response to what the agent senses and to the choices of the agent.

We clarify that the practical rules are 'potential' because two or more rules may apply in some circumstances and yield distinct actions for the same goal or commitment. For example, two practical rules may apply on the same commitment, one advocating suspending it and the other advocating cancelling it. As we will explain, it is the choice of the agent which among such 'competing' rules to choose to apply. Further, a practical rule can be applied more than once to the same goals and commitments over time: for example, a commitment can be suspended and resumed several times over the course of its life.

Our operational semantics captures the autonomy of the agents by leaving each agent's decision-making unspecified. That is, our semantics considers any action that an agent could perform within the remit we treat in this article, regardless of the practical rules, and constrained only by the properties mentioned above. The operational semantics makes clear how the configuration of a multiagent system progresses as the agents act. However, in addition, we establish results such as regarding convergence that are specific to our proposed set of practical rules.

The rest of this subsection formulates the concepts underlying the rules of our operational semantics. An informal summary of all definitions in this subsection is provided by Table 1. The rules themselves are presented in the subsections that follow. Specifically, the aim of the current subsection is to define the system configuration and how agent actions modify it. We therefore begin with preliminaries for defining the configuration of an agent (Sections 3.3.1–3.3.5), and then provide the definition itself (Section 3.3.6). This enables us to define the configuration of a whole multiagent system. Mandatory actions of agents, in a sense we make precise, are captured by life cycle rules (Section 3.3.7). These rules constitute a labelled transition system, with the actions being the labels and the multiagent system configuration being the state. We prove that configuration consistency is maintained by life cycle rules, which allows us to conclude the subsection by defining traces of configurations. Practical rules are presented in the next Section 3.4.

3.3.1 Preliminaries

We suppose a finite set of agents, $x_1, x_2, \ldots \in \mathscr{A}$, and a finite set of propositional atoms, $a_1, a_2, \ldots \in \Omega$. We write Ψ for the set of all propositional formulae over Ω. The symbol $\top$ abbreviates $a \vee \neg a$ for any atom a, and the symbol $\bot$ abbreviates $\neg\top$. We assume classical propositional logic. Specifically, given a set of propositions $\Phi \subseteq \Psi$ and a proposition $\psi \in \Psi$, $\Phi \models \psi$ denotes that Φ entails ψ. We say that a set of propositions Φ is *consistent* iff $\Phi \not\models \bot$.

3.3.2 Beliefs

The first element of an agent's configuration is its beliefs.

Definition 1 (Belief)**.** *A* belief *is a proposition $\psi \in \Psi$.*

Note that we need not include the agent's name in the definition of a belief (i.e., "agent x believes ψ") since beliefs will be included as part of the configuration of an agent.

The next definition provides a means of obtaining the state of an agent's belief:

Definition 2 (Belief state function)**.** *A* belief state function $\mathscr{B} : \mathscr{A} \times \Psi \to \{\top, \bot\}$ *returns $\top$ if an agent believes a proposition, otherwise $\bot$. We write $\mathscr{B}_x$ for the set of all (current) beliefs of an agent $x \in \mathscr{A}$, i.e., $\{\psi \in \Psi : \mathscr{B}(x, \psi)\}$. An agent's beliefs are consistent, i.e., $\neg\mathscr{B}(x, \bot)$, and closed under entailment, i.e., if $\mathscr{B}(x, \phi)$ and $\phi \models \psi$ then $\mathscr{B}(x, \psi)$.*

The previous definition imposes that agents are rational in their beliefs, in the sense that an agent's beliefs are consistent and closed under entailment. Although we require beliefs to be mutually consistent, we do not require them to be exhaustive. That is, an agent x may have no belief about p and $\neg p$, meaning that $B(x, p) = \bot$ and $B(x, \neg p) = \bot$ can coexist. However, by consistency, $B(x, p) = \top$ and $B(x, \neg p) = \top$ cannot coexist. Further note that,

$x, y \in \mathscr{A}$	finite set of agents
$a \in \Omega$	finite set of propositional atoms
$\phi, \psi \in \Psi := a \mid \top \mid \bot \mid \phi \wedge \psi \mid \phi \vee \psi \mid \neg\phi$	propositions
$p, q, r, s, u, v \in \Psi$	antecedent and consequents
$s, t, f, h \in \Psi$	success and failure conditions
$\mathscr{B} : \mathscr{A} \times \Psi \to \{\top, \bot\}$	belief state function
$\mathsf{C}(x, y, p, q) \in \mathscr{C}_x$	commitment
$\sigma \in \chi_C := \{\mathcal{N}, \mathcal{C}, \mathcal{E}, \mathcal{D}, \mathcal{P}, \mathcal{T}, \mathcal{V}, \mathcal{S}\}$	commitment states
$\mathscr{C} : \mathsf{C}(x, y, p, q) \to \chi_C$	commitment state function
$\mathsf{G}(x, s, f) \in \mathscr{G}_x$ (with $s \wedge f \models \bot$)	goal
$\sigma \in \chi_G := \{\mathcal{N}, \mathcal{I}, \mathcal{A}, \mathcal{U}, \mathcal{T}, \mathcal{F}, \mathcal{S}\}$	goal states
$\mathscr{G} : \mathsf{G}(x, s, f) \to \chi_G$	goal state function
$BACTS := \{+\}$	belief actions
$GACTS := \{$consider, activate, suspend-G, reconsider, reactivate-G, terminate$\}$	goal actions
$CACTS := \{$create, suspend-C, cancel, release, reactivate-C$\}$	commitment actions
$\alpha \in \mathbb{A} := (BACTS \times \mathbb{B}) \mathbin{\dot\cup} (GACTS \times \mathbb{G}) \mathbin{\dot\cup} (CACTS \times \mathbb{C})$	action
$\langle \mathscr{B}, \mathscr{G}, \mathscr{C} \rangle_x \xrightarrow{\text{RULENAME}} \alpha$	practical rule instance
$E \xrightarrow{\text{RULENAME}} \alpha$	practical rule template
$ant(\mathsf{C}(x, y, p, q)) := p$	antecedent of a commitment
$succ(\mathsf{G}(x, s, f)) := s$	success condition of a goal
$maxc(\Sigma)$ (with $\Sigma \subseteq \chi_C$)	maximally strong commit. set w.r.t. Σ
$maxg(\Sigma)$ (with $\Sigma \subseteq \chi_G$)	maximally strong goal set w.r.t. Σ
CSG, CAG, CCG	commitment support sets
GSC, GAC, GCC	goal support sets
$S(x) := \langle \mathscr{B}_x, \mathscr{G}_x, \mathscr{C}_x \rangle := \langle \mathscr{B}, \mathscr{G}, \mathscr{C} \rangle_x$	agent configuration
M	multiagent system (with n agents)
$S(M) := \langle S(1), \ldots, S(n) \rangle$	system configuration
$\mathbb{B}, \mathbb{G}, \mathbb{C}$	beliefs, goals, commitments of M
$L : \mathbb{A} \times \mathbb{B} \times \mathbb{G} \times \mathbb{C} \to \mathbb{B} \times \mathbb{G} \times \mathbb{C}$	life cycle rule
$\tau = S_1, S_2, \ldots$	trace

Table 1: Summary of notation. Top: ‘external’ notation relevant for an agent designer. Bottom: ‘internal’ notation used in the semantics.

due to the stable propositions assumption, for our purposes we need only consider belief addition operations.

3.3.3 Commitments and Commitment Consistency

The second element of an agent's configuration is its commitments. We formalize commitments; introduce a notion of the relative strength of commitments, which is necessary for the closure properties; provide a means of obtaining the state of a commitment, and using it define the closure properties; and define commitment consistency.

Definition 3 (Commitment). *A* commitment *is a tuple consisting of two agents (its* debtor *and* creditor*, denoted $x \in \mathscr{A}$ and $y \in \mathscr{A}$, respectively), and two propositions (its* antecedent *and* consequent*, denoted $p \in \Psi$ and $q \in \Psi$, respectively), i.e., $\langle x, y, p, q \rangle$, where $x \neq y$, and $p \not\models q$ and $p \not\models \neg q$. We write a commitment as: $C = \mathsf{C}(x, y, p, q)$.*

Next, following Chopra and Singh (2009); Singh (2008) we define the notion of commitment strength. We employ commitment strength in defining the important commitment closure properties below.

Definition 4 (Commitment strength). *A commitment $C_1 = \mathsf{C}(x, y, r, u)$ is stronger than $C_2 = \mathsf{C}(x, y, s, v)$, written $C_1 \succeq C_2$ or $C_2 \preceq C_1$, iff $s \models r$ and $u \models v$.*

For example, commitment $C_1 = \mathsf{C}(x, y, \mathsf{pay}, \mathsf{book} \wedge \mathsf{pen})$ is stronger than commitment $C_2 = \mathsf{C}(x, y, \mathsf{pay} \wedge \mathsf{pickup}, \mathsf{book})$. Note that commitment strength is a preorder relation.

As introduced in Section 2, commitments have state. We define a set of commitment state labels in line with Figure 1.

Definition 5 (Commitment states). *The* commitment states *are a set of labels $\chi_C = \{\mathcal{N}, \mathcal{C}, \mathcal{E}, \mathcal{D}, \mathcal{P}, \mathcal{T}, \mathcal{V}, \mathcal{S}\}$.*

For example, $\mathcal{D}$ denotes the `Detached` state.

We need to know the state of a given commitment. Our practical rules use the commitment state function to obtain the states of commitments.

Definition 6 (Commitment state function). *The* commitment state function *$\mathscr{C}$ returns the state of a commitment $\mathsf{C}(x, y, p, q)$, where $\mathscr{C}(\mathsf{C}(x, y, p, q)) \in \chi_C$. The commitment state function satisfies the following closure properties:*

- *If $\mathscr{C}(C_1) = \sigma$, where $\sigma \in \{\mathcal{C}, \mathcal{S}, \mathcal{E}\}$ and $C_1 \succeq C_2$, then $\mathscr{C}(C_2) = \sigma$.*
- *If $\mathscr{C}(C_1) = \sigma$, where $\sigma \in \{\mathcal{T}, \mathcal{V}\}$ and $C_2 \succeq C_1$, then $\mathscr{C}(C_2) = \sigma$.*

To simplify the notation, we write $\mathscr{C}(\mathsf{C}(x, y, p, q))$ as $\mathscr{C}(x, y, p, q)$. We write $\mathscr{C}_x$ for the set of all non-`Null` commitments in which agent x is either debtor or creditor.

The properties observed in the last definition generalize some of the postulates motivated in Chopra and Singh (2009); Singh (2008). The closure of the commitment state function with respect to commitment strength ensures that the states assigned to commitments in any configuration respect the following property: for states `Conditional`, `Satisfied`, and `Expired`, if a commitment is one of these states, then so is any commitment weaker than it; whereas for states `Terminated`, and `Violated`, if a commitment is in one of these states, then so is any commitment stronger than it.

The underlying intuition is that were such closure properties not to hold, an agent configuration could end up 'confused' in regards to the social relationships we are modelling through commitments. For example, if we identified a commitment C_1 as `Active` simultaneously with identifying a stronger commitment C_2 as `Satisfied`, then our logic would force a conclusion that C_1 was also `Satisfied`—thereby making the state of C_1 ambiguous. We term this intuition *semantic well-formedness.*

Some of the practical rules operate over the maximally strong commitments, motivating the next definition. Intuitively, the maximally strong commitments w.r.t. a set of commitment states Σ are those commitments in any state $\sigma \in \Sigma$ for which there is no strictly stronger commitment in the same state σ.

Definition 7 (Maximally strong commitment set)**.** *Let $\Sigma \subseteq \chi_C$ be a set of commitment states. $maxc(\Sigma) = \{\mathsf{C}(x, y, s, v) \in \mathscr{C}_x \mid \exists \sigma \in \Sigma$ and $\mathscr{C}(x, y, s, v) = \sigma$, and $(\forall r, u : \mathscr{C}(x, y, r, u) = \sigma,\ \mathsf{C}(x, y, r, u) \succeq \mathsf{C}(x, y, s, v) \Rightarrow \mathsf{C}(x, y, r, u) = \mathsf{C}(x, y, s, v))\}$.*

For example, consider the set of commitments: $\{C_1 = \mathsf{C}(x, y, \mathsf{pay}, \mathsf{book} \wedge \mathsf{pen}), C_2 = \mathsf{C}(x, y, \mathsf{pay}, \mathsf{book}), C_3 = \mathsf{C}(x, y, \top, \mathsf{flight\text{-}ticket} \wedge \mathsf{hotel\text{-}room}), C_4 = \mathsf{C}(x, y, \top, \mathsf{flight\text{-}ticket})\}$; commitments C_1 and C_2 are in state `Conditional`, and C_3 and C_4 are in state `Detached`. Then, C_1 is a maximally strong commitment in state `Conditional`, and C_3 is maximally strong commitment in state `Detached`, that is, $C_1 \in maxc(\mathcal{C})$ and $C_3 \in maxc(\mathcal{D})$.

Although each commitment is in a single state at any time, the $maxc()$ function finds the maximal commitments with respect to a set of states. The concept of support sets below uses $maxc()$ over a set of two states; hence note we cannot eliminate sets from Definition 7.

Lastly, as with beliefs, we need to consider the mutual consistency of commitments. A cooperative agent will not take on logically inconsistent commitments. Informally, a set of commitments is consistent if satisfying a commitment in that set does not violate some other commitment in that set. Recall a commitment is violated when it is detached (its antecedent is true) but is never discharged. Two commitments would be inconsistent if they can be detached together (thus their antecedents are consistent) and in cases where their antecedents are satisfied, their consequents cannot both be satisfied. The following definition captures this intuition, expanding it to larger sets of commitments.

Definition 8 (Commitment consistency)**.** *Let $S \subseteq \mathscr{C}_x$ be a set of commitments of a debtor $x \in \mathscr{A}$. Writing $\mathsf{C}(x, y_i, r_i, u_i)$ for the commitments in S, the set is inconsistent iff (1) $\bigwedge r_i \not\models \bot$, and (2) $(\bigwedge r_i) \wedge (\bigwedge u_i) \models \bot$. A set of commitments is* consistent *if it is not inconsistent.*

Note that the definition does not require commitment antecedents to be consistent, only that if they are, then the antecedents and consequents must together be consistent. Further, note that the definition permits consistent commitments with conflicting antecedents and conflicting consequents. Lastly, note that it does not suffice to have only the consequents consistent in the second part of the definition.

As an example of commitment consistency, the set of commitments $\{\mathsf{C}(x, y, \mathsf{make\text{-}payment}, \mathsf{open\text{-}door}),\ \mathsf{C}(x, z, \mathsf{make\text{-}payment}, \mathsf{open\text{-}window})\}$ is consistent, whereas the set of commitments $\{\mathsf{C}(x, y, \mathsf{make\text{-}payment}, \mathsf{open\text{-}door}), \mathsf{C}(x, z, \mathsf{make\text{-}payment}, \neg\mathsf{open\text{-}door})\}$ is inconsistent.

3.3.4 Goals and Goal Consistency

Having defined beliefs and commitments, the third element of an agent's configuration is its goals. We proceed as with commitments: we first formalize goals, then introduce a notion of the relative strength of goals, provide a means of obtaining the state of a goal and define closure properties, and finally define goal consistency.

Definition 9 (Goal)**.** *A* goal *is a tuple consisting of an agent x and two propositions: its* success *and* failure *conditions (denoted $s \in \Psi$ and $f \in \Psi$, respectively), i.e., $\langle x, s, f\rangle$, where $s \wedge f \models \bot$. We write a goal as: $G = \mathsf{G}(x, s, f)$.*

We define the notion of goal strength in order to be able to express the goal closure properties below. Intuitively, a goal G_1 is stronger than another goal G_2 if success of G_1 implies success of G_2, and failure of G_1 implies failure of G_2.

Definition 10 (Goal strength)**.** *A goal $G_1 = \mathsf{G}(x, s, f)$ is stronger than goal $G_2 = \mathsf{G}(x, t, h)$, written $G_1 \succeq G_2$ or $G_2 \preceq G_1$, iff $s \models t$ and $f \models h$.*

For example, the goal $\mathsf{G}(x, \mathsf{book} \wedge \mathsf{pen}, \mathsf{insufficient\text{-}money} \wedge \mathsf{insufficient\text{-}time})$ is stronger than the goal $\mathsf{G}(x, \mathsf{book}, \mathsf{insufficient\text{-}money})$. Note that goal strength is a preorder relation.

Since goals have state, following Figure 2, we define a set of goal state labels and the goal state function to obtain the states of goals:

Definition 11 (Goal states)**.** *The* goal states *are a set of labels: $\chi_G = \{\mathcal{N}, \mathcal{I}, \mathcal{A}, \mathcal{U}, \mathcal{T}, \mathcal{F}, \mathcal{S}\}$.*

Definition 12 (Goal state function)**.** *The* goal state function *$\mathscr{G}$ returns the state of a goal $\langle x, s, f\rangle$, that is, $\mathscr{G}(x, s, f) \in \chi_G$. The goal state function satisfies the following closure properties:*

- *If $\mathscr{G}(G_1) = \sigma$, where $\sigma \in \{\mathcal{A}, \mathcal{S}\}$ and $G_1 \succeq G_2$, then $\mathscr{G}(G_2) = \sigma$.*
- *If $\mathscr{G}(G_1) = \sigma$, where $\sigma \in \{\mathcal{T}, \mathcal{F}\}$ and $G_2 \succeq G_1$, then $\mathscr{G}(G_2) = \sigma$.*

To simplify the notation, we write $\mathscr{G}(\mathsf{G}(x, s, f))$ as $\mathscr{G}(x, s, f)$. We write $\mathscr{G}_x$ for the set of all non-`Null` goals of agent x.

Some of the practical rules operate over the maximally strong goals, motivating the next definition, which is akin to Definition 7: for a set of goal states Σ, the maximally strong goals are those in some $\sigma \in \Sigma$ for which there is no strictly stronger goal in the same state σ.

Definition 13 (Maximally strong goal set)**.** *Let $\Sigma \subseteq \chi_G$ be a set of goal states. $maxg(\Sigma) = \{\mathsf{G}(x, s, f) \in \mathscr{G}_x \,|\, \exists \sigma \in \Sigma$ and $\mathscr{G}(x, s, f) = \sigma$, and $(\forall t, g : \mathscr{G}(x, t, g) = \sigma,\ \mathsf{G}(x, t, g) \succeq \mathsf{G}(x, s, f) \Rightarrow \mathsf{G}(x, t, g) = \mathsf{G}(x, s, f))\}$.*

For example, consider the set of goals: $\{G_1 = \mathsf{G}(x, \mathsf{book} \wedge \mathsf{pen}, \mathsf{insufficient\text{-}money})$, $G_2 = \mathsf{G}(x, \mathsf{book}, \mathsf{insufficient\text{-}money})$, $G_3 = \mathsf{G}(x, \mathsf{book} \wedge \mathsf{pen} \wedge \mathsf{glasses}, \mathsf{insufficient\text{-}money})\}$. Suppose goals G_1 and G_2 are `Inactive`, and goal G_3 is `Active`. Then, G_1 is a maximally strong goal in state `Inactive`, and G_3 is a maximally strong goal in state `Active`, that is, $G_1 \in maxg(\{\mathcal{I}\})$ and $G_3 \in maxg(\{\mathcal{A}\})$.

The closure of the goal state function with respect to goal strength ensures that the goals in any configuration are semantically well-formed: if a goal is in some state then stronger or weaker goals are in appropriate states as well. As for commitments, the intuition of semantic well-formedness seeks to characterize configurations that are unambiguous and respect the logic of goals.

Lastly, we need to consider the mutual consistency of goals. Informally, a set of goals is consistent if satisfying a goal from the set does not cause another goal in that set to fail.

Definition 14 (Goal consistency: single agent). *Let $S \subseteq \mathscr{G}_x$ be a set of goals of an agent $x \in \mathscr{A}$. Writing $\mathsf{G}(x_i, s_i, f_i)$ for the goals in S, the set is inconsistent iff: (1) $\bigwedge s_i \models \bot$ or (2) $\bigwedge s_i \wedge \bigwedge f_i \not\models \bot$. A set of goals is consistent if it is not inconsistent.*

For example: the set of goals $\{\mathsf{G}(x, \textsf{open-door}, f_1), \mathsf{G}(x, \neg\textsf{open-door}, f_2)\}$ is inconsistent.

3.3.5 Relating Commitments and Goals: Support Sets

In this technical subsection, we introduce definitions that relate commitments and goals. We employ these definitions in the practical reasoning rules that we present in Section 3.4.

We define six sets that express different forms of *support* of a goal by a commitment or vice versa. The six sets come in three groups: (1) commitments providing support to goals, (2) goals providing antecedent support to commitments, and (3) goals providing consequent support to commitments. Figure 4 depicts the three groups of support sets.

For the remainder of this subsection, we explain the definitions using an example with goals $G_1 = \mathsf{G}(x, p \wedge q, f)$ and $G_2 = \mathsf{G}(x, p, f)$ and commitments $C_1 = \mathsf{C}(x, y, p, v)$ and $C_2 = \mathsf{C}(x, z, q, w)$ that are `Active` or `Pending`.

The first two sets relate to commitments supporting goals: the set of goals supported by a commitment (GSC), and the set of commitments supporting a goal (CSG). These are shown in the top row of Figure 4.

Definition 15 (Set of goals supported by a commitment (GSC)). *Let $C = \mathsf{C}(x, y, r, u)$ and $G = \mathsf{G}(x, s, f)$. Then $GSC(C) = \{G = \mathsf{G}(x, s, f) \in \mathscr{G}_x \,|\, C \in maxc(\{\mathcal{C}, \mathcal{D}, \mathcal{P}\}), \mathscr{G}(G) \in \{\mathcal{I}, \mathcal{A}, \mathcal{U}\}, s = \bigwedge s_i, r \models s_i, u \not\models \neg s_i\}$.*

Note that s_i are the conjuncts of s; i ranges over the number of conjuncts. Goal G is in the set $GSC(C)$ iff G is `Inactive`, `Active`, or `Suspended` and is supported by C. We leave $GSC(C)$ undefined if C is not a maximal commitment for states `Active` or `Pending`.

Example: $G_1 \in GSC(C_1)$ and $G_2 \in GSC(C_1)$.

Definition 16 (Set of commitments supporting a goal (CSG)). *Let $G = \mathsf{G}(x, s, f)$ and $C = \mathsf{C}(x, y, r, u)$. Then $CSG(G) = \{C \in \mathscr{C}_x \,|\, C \in maxc(\{\mathcal{C}, \mathcal{D}, \mathcal{P}\}), \mathscr{G}(G) \in \{\mathcal{I}, \mathcal{A}, \mathcal{U}\}, s = \bigwedge s_i, r \models s_i, u \not\models \neg s_i\}$.*

Hence, C is in the set $CSG(G)$ iff C is maximal for `Active` or `Pending`, and G is supported by C. Example: $C_1 \in CSG(G_2)$.

The remaining four sets relate to goals supporting commitments, either to the antecedent or the consequent. These are respectively shown in the bottom two rows of Figure 4.

Based on the following definition of commitment antecedent support, we define the set of commitments with antecedent support of a goal (CAG), and the set of goals providing antecedent support to a commitment (GAC).

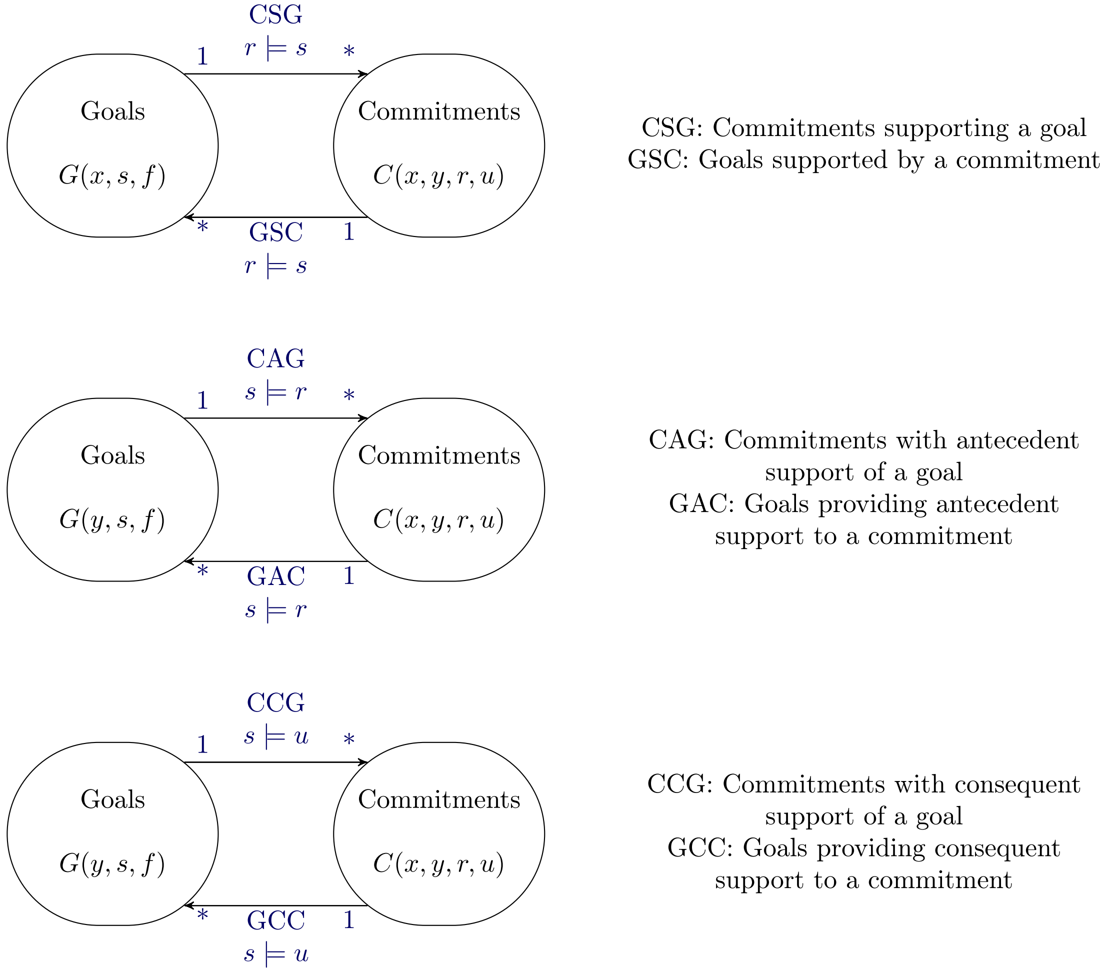
Definition 17 (Commitment antecedent support)**.** *A commitment* $\mathsf{C}(x, y, r, u)$ *has* (partial) antecedent support *of a goal* $G = \mathsf{G}(y, s, f)$ *iff* $G \in maxg(\{\mathcal{I}, \mathcal{A}, \mathcal{U}\})$*,* $\mathscr{C}(C) \in \{\mathcal{C}, \mathcal{D}, \mathcal{P}\}$*, and* $s \models r_i$ *for at least one* r_i*, where* $r = \bigwedge r_i$*, and* $s \not\models u_i$ *for any* u_i*, where* $u = \bigwedge u_i$*.*

Note that r_i and u_i are the conjuncts of r and u, respectively.

Example: Let $G'_1 = \mathsf{G}(y, p \wedge q, f)$ and $G'_2 = \mathsf{G}(y, p, f)$. Commitment C_1 has antecedent support from each of goals G'_1 and G'_2.

Definition 18 (Set of commitments with antecedent support of a goal (CAG))**.** *Let* $G = \mathsf{G}(y, s, f)$ *and* $C = \mathsf{C}(x, y, r, u)$*. Then* $CAG(G) = \{C \in \mathscr{C}_x \,|\, G \in maxg(\{\mathcal{I}, \mathcal{A}, \mathcal{U}\}), \mathscr{C}(C) \in \{\mathcal{C}, \mathcal{D}, \mathcal{P}\}, r = \bigwedge r_i, s \models r_i, u = \bigwedge u_i, s \not\models \neg u_i\}$*.*

Definition 19 (Set of goals providing antecedent support to a commitment (GAC))**.** *Let* $C = \mathsf{C}(x, y, r, u)$ *and* $G = \mathsf{G}(y, s, f)$*. Then* $GAC(C) = \{G \in \mathscr{G}_x \,|\, G \in maxg(\{\mathcal{I}, \mathcal{A}, \mathcal{U}\}), \mathscr{C}(C) \in \{\mathcal{C}, \mathcal{D}, \mathcal{P}\}, r = \bigwedge r_i, s \models r_i, u = \bigwedge u_i, s \not\models \neg u_i\}$*.*



CSG: Commitments supporting a goal
GSC: Goals supported by a commitment

CAG: Commitments with antecedent support of a goal
GAC: Goals providing antecedent support to a commitment

CCG: Commitments with consequent support of a goal
GCC: Goals providing consequent support to a commitment

Figure 4: Sets relating commitments and goals.

Similar to the commitment antecedent support, lastly we define commitment consequent support, and based on it define the set of commitments with consequent support of a goal (CCG), and the set of goals providing antecedent support to a commitment (GCC).

Definition 20 (Commitment consequent support)**.** *A commitment $C = \mathsf{C}(x, y, r, u)$ has* (partial) consequent support *of a goal $G = \mathsf{G}(x, s, f)$ iff $G \in maxg(\{\mathcal{I}, \mathcal{A}, \mathcal{S}\})$, $\mathscr{C}(C) \in \{\mathcal{C}, \mathcal{D}, \mathcal{P}\}$, and $s \models u_i$ for at least one u_i, where $u = \bigwedge u_i$, and $s \not\models \neg r_i$ for any r_i, where $r = \bigwedge r_i$.*

Example: Let $G_3 = \mathsf{G}(x, v \wedge w, f)$. Commitment C_1 has consequent support from G_3 but not from G_1 or G_2.

Definition 21 (Set of commitments with consequent support of a goal (CCG))**.** *Let $\mathsf{G}(x, s, f)$ and $C = \mathsf{C}(x, y, r, u)$. Then $CCG(G) = \{C \in \mathscr{C}_x \,|\, G \in maxg(\{\mathcal{I}, \mathcal{A}, \mathcal{U}\}), \mathscr{C}(C) \in \{\mathcal{C}, \mathcal{D}, \mathcal{P}\}, u \bigwedge u_i, s \models u_i, r = \bigwedge r_i, s \not\models \neg r_i\}$.*

Definition 22 (Set of goals providing consequent support to a commitment (GCC))**.** *Let $C = \mathsf{C}(x, y, r, u)$ and $G = \mathsf{G}(x, s, f)$. Then $GCC(C) = \{G \in \mathscr{G}_x \,|\, G \in maxg(\{\mathcal{I}, \mathcal{A}, \mathcal{U}\}), \mathscr{C}(C) \in \{\mathcal{C}, \mathcal{D}, \mathcal{P}\}, u \bigwedge u_i, s \models u_i, r = \bigwedge r_i, s \not\models \neg r_i\}$.*

3.3.6 Agent Configuration

With all the pieces in place, we are ready to define the configuration of an agent, which consists of its beliefs, goals, and commitments:

Definition 23 (Agent configuration)**.** *The* configuration *of an agent x is the tuple $S(x) = \langle \mathscr{B}_x, \mathscr{G}_x, \mathscr{C}_x \rangle$ where $\mathscr{B}_x$ and $\mathscr{G}_x$ are state functions for x's beliefs and goals, and $\mathscr{C}_x$ is a state function for commitments in which agent x is either debtor or creditor.*

In order to reduce clutter, we write the configuration of agent x as $\langle \mathscr{B}, \mathscr{G}, \mathscr{C} \rangle_x$ instead of $\langle \mathscr{B}_x, \mathscr{G}_x, \mathscr{C}_x \rangle$.

The observant reader will notice that previously we defined $\mathscr{G}_x$ (etc.) as the set of all goals (etc.) of agent x, whereas in the last definition we overload the notation to define $\mathscr{G}_x$ via the goal state function. The justification is that the set of all of x's goals comprises precisely those goals $\mathsf{G}(x, s, f)$ that have non-`Null` state, i.e., $\mathscr{G}_x = \{\mathscr{G}(x, s, f) \neq \mathcal{N}\}$. Hence, we can use the set and function views interchangeably.

Since the configuration consists of beliefs and other elements, we now define the consistency conditions between sets of beliefs and goals, and beliefs and commitments. For example, if agent x believes in the success condition of a goal, then it must be that the goal's state is either `Null` (i.e., whereupon it is not in $\mathscr{G}_x$) or `Satisfied`.

Definition 24 (Commitment–Belief consistency)**.** *A state function for commitments $\mathscr{C}$ and a state function for beliefs $\mathscr{B}$ are* consistent *with each other iff all of the following are true, where $x, y \in \mathscr{A}$:*

- *$\forall x, \forall y, \forall p, \forall u$: if state $= \mathscr{C}(x, y, p, u)$, $\mathscr{B}(x, p) = \bot$ and $\mathscr{B}(x, u) = \bot$, then state $\in \{\mathcal{N}, \mathcal{C}\}$*
- *$\forall x, \forall y, \forall p, \forall u$: if state $= \mathscr{C}(x, y, p, u)$, $\mathscr{B}(x, p) = \top$ and $u \in \Psi$, then state $\in \{\mathcal{N}, \mathcal{D}\}$*

- $\forall x, \forall y, \forall p, \forall u$: *if state* $= \mathscr{C}(x, y, p, u)$, $\mathscr{B}(x, u) = \top$ *and* $p \in \Psi$, *then state* $\in \{\mathcal{N}, \mathcal{S}\}$
- $\forall x, \forall y, \forall p, \forall u$: *if state* $= \mathscr{C}(x, y, p, u)$, $\mathscr{B}(x, \neg p) = \top$ *and* $u \in \Psi$, *then state* $\in \{\mathcal{N}, \mathcal{E}\}$
- $\forall x, \forall y, \forall p$: *if state* $= \mathscr{C}(x, y, \top, p)$ *and* $\mathscr{B}(x, \neg p) = \top$, *then state* $\in \{\mathcal{N}, \mathcal{V}\}$

Recall from Section 3.3.1 that Ω denotes the set of atoms.

Definition 25 (Goal–Belief consistency)**.** *A state function for goals* $\mathscr{G}$ *and a state function for beliefs* $\mathscr{B}$ *are* consistent *with each other iff for each belief* p *s.t.* $\mathscr{B}(x, p) = \top$ *all of the following are true:*

- $\forall f$: *if state* $= \mathscr{G}(x, p, f)$ *and* $f \in \Omega$, *then state* $\in \{\mathcal{N}, \mathcal{S}\}$
- $\forall s$: *if state* $= \mathscr{G}(x, s, p)$ *and* $s \in \Omega$, *then state* $\in \{\mathcal{N}, \mathcal{F}\}$

3.3.7 System Configuration, Life Cycle Rules, and Traces

Having defined the configuration of an agent, we next move on to define the configuration of a multiagent system and to study its consistency according to the life cycle rules of commitments and goals. We conclude the subsection by defining the trace of system configurations, by which we will prove properties of the operational semantics.

Conceptually, an agent's configuration relates to elements both of its cognitive state (i.e., beliefs and goals) and of the relevant components of the social state (i.e., commitments of which the agent is creditor or debtor). In our approach, the notional social state is not stored independently of the agents—that is, it exists only in terms of its projections in the various agents. Due to the assumption of synchronous communication, the projections of the social state on different agents remain mutually consistent. Since the goals in $\mathscr{G}_x$ are all adopted by x, we take it that these goals are mutually consistent (Winikoff et al., 2002), according to Definition 14. Recall from Section 2 that a goal is private to an agent, whereas a commitment, being an element of the social state, is represented in both its creditor and its debtor. The rules we introduce in the coming sections apply to each agent's internal representation separately. These rules constitute a labelled transition system, with the actions being the labels and the multiagent system configuration being the state, i.e., $S \xrightarrow{\alpha} S'$. The transition system is parameterized by a life cycle rule, introduced below.

Definition 26 (System configuration)**.** *Given a multiagent system* M *consisting of agents* $\mathscr{A} = x_1, \ldots, x_n$, *the* system configuration *of* M *is given by an* n*-tuple* $\langle S(1), \ldots, S(n) \rangle$, *where* $S(i)$ *is the configuration of agent* x_i.

When required, we write a multiagent system configuration with each agent's configuration expanded to its beliefs, goals, and commitments as follows: $\langle \langle \mathscr{B}, \mathscr{G}, \mathscr{C} \rangle_1, \langle \mathscr{B}, \mathscr{G}, \mathscr{C} \rangle_2, \ldots, \langle \mathscr{B}, \mathscr{G}, \mathscr{C} \rangle_n \rangle$.

We now define formally the life cycle of goals and commitments. For this, we need action sets for each of beliefs, goals, and commitments to describe the operations on goals and commitments (see Figure 3). Each of these three action sets is defined as a power set, meaning that the agent can consider zero or more actions of each type. Although the agent may consider multiple actions, recall however from Section 3.1 that in each deliberation cycle the agent can *select* at most one action for each commitment and each goal.

Denote the set of beliefs of all agents in the multiagent system as $\mathbb{B}$, the set of all goals as $\mathbb{G}$, and the set of all commitments as $\mathbb{C}$. The actions are those that agents can take on individual elements of the system configuration. For commitments and goals, the actions are exactly those in the life cycles (Figures 1 and 2). We use them in the following definition of action sets that specify what actions agents do take. Using these, we formalize the life cycle rules which tell us how the agents' actions affect the system configuration.

Definition 27 (Actions)**.** *Let the possible belief actions $BACTS$ be the set $\{+\}$. Let the possible goal actions $GACTS$ be the set $\{$consider, activate, suspend-G, reconsider, reactivate-G, terminate$\}$. Let the possible commitment actions $CACTS$ be the set $\{$create, suspend-C, cancel, release, reactivate-C$\}$.*

Definition 28 (Action set)**.** *An* action set $\mathbb{A}$ *is a disjoint union of three sets of pairs: $(BACTS \times \mathbb{B}) \uplus (GACTS \times \mathbb{G}) \uplus (CACTS \times \mathbb{C})$.*

For example, the action set $\{\langle \mathsf{activate}, G_1\rangle\}$ corresponds to the action of activating goal G_1.

Belief addition is the only belief operation that we need consider for our purposes.

We can now define a life cycle rule that captures the effect of actions on the system configuration. Specifically, it maps an action set and a system configuration into the resulting system configuration. The definition of the life cycle rule is in three parts, for beliefs, goals, and commitments, respectively. The main idea is to enumerate the possible updates to a configuration given an action. The multiple parts of the definition capture the life cycles of commitments and goals in logical terms while preserving the consistency and closure properties of the commitments and goals.

In more detail, the various items in this definition consider the strongest or weakest goals and commitments that are affected through the acquisition of a belief or through the performance of a goal or commitment action. When the strongest goal or commitment is affected, all the relevant weaker goals and commitments are affected in a manner that is consistent with respect to the closure properties. Note that although this life cycle rule includes several cases, each case is pretty simple. We need several cases to capture how beliefs, goals, and commitments progress—specifically, how changes in beliefs cause changes to goal and commitment states, and how actions on goals and commitments affect their respective states. For each item in the definition we give a brief sentence of explanation immediately following it.

Definition 29 (Life cycle rule)**.** *A* life cycle *rule is a function $L : \mathbb{A} \times \mathbb{B} \times \mathbb{G} \times \mathbb{C} \rightarrow \mathbb{B} \times \mathbb{G} \times \mathbb{C}$ such that:*

(i) $\forall \langle +, b\rangle \in \mathbb{A}$, $b = \mathscr{B}(x, p)$:
$\langle \mathscr{B}', \mathscr{G}', \mathscr{C}'\rangle = L(+p, \mathscr{B}, \mathscr{G}, \mathscr{C})$, *where:*

1. $\mathscr{B}'(x, p) = \top$ x believes the newly added proposition p
2. $\forall s, \forall f, \forall t, \forall h$: *if* $\mathsf{G}(x, s, f) \in maxg(\mathcal{I}, \mathcal{A}, \mathcal{U}), p \models s$, *then* $\mathscr{G}'(x, s, f) = \mathcal{S}$, *and if* $\mathsf{G}(x, t, h) \preceq \mathsf{G}(x, s, f)$, *then* $\mathscr{G}'(x, t, h) = \mathcal{S}$

   if $p \models s$, each maximally strong goal $\mathsf{G}(x, s, f)$ that is `Inactive`, `Active`, or `Suspended`, satisfies, and all goals $\mathsf{G}(x, t, h)$ weaker than $\mathsf{G}(x, s, f)$ also satisfy

3. *$\forall s, \forall f, \forall t, \forall h$: if $\mathscr{G}(x,s,f) \in \{\mathcal{I}, \mathcal{A}, \mathcal{U}\}, p \models f$, then $\mathscr{G}'(x,s,f) = \mathcal{F}$, and if $\mathsf{G}(x,t,h) \succeq \mathsf{G}(x,s,f)$, then $\mathscr{G}'(x,t,h) = \mathcal{F}$*

   if $p \models f$, each goal $\mathsf{G}(x,s,f)$ that is `Inactive`, `Active`, or `Suspended`, fails, and all goals $\mathsf{G}(x,t,h)$ stronger than $\mathsf{G}(x,s,f)$ also fail

4. *$\forall y, \forall r, \forall u$: if $\mathsf{C}(x,y,r,u) \in maxc(\mathcal{C}), p \models r, p \not\models u$, then $\mathscr{C}'(x,y,r,u) = \mathcal{D}$*

   if $p \models r$ and $p \not\models u$, each maximally strong commitment $\mathsf{C}(x,y,r,u)$ that is `Conditional`, detaches

5. *$\forall y, \forall r, \forall u, \forall s, \forall v$: if $\mathsf{C}(x,y,r,u) \in maxc(\mathcal{C}, \mathcal{D}), p \models u$, then $\mathscr{C}'(x,y,r,u) = \mathcal{S}$, and if $\mathsf{C}(x,y,s,v) \preceq \mathsf{C}(x,y,r,u)$, then $\mathscr{C}'(x,y,s,v) = \mathcal{S}$*

   if $p \models u$, each maximally strong commitment $\mathsf{C}(x,y,r,u)$ that is `Conditional` or `Detached`, satisfies, and all commitments $\mathsf{C}(x,y,s,v)$ weaker than $\mathsf{C}(x,y,r,u)$ also satisfy

6. *$\forall y, \forall r, \forall u, \forall s, \forall v$: if $\mathscr{C}(x,y,r,u) = \mathcal{C}, p \models \neg r, p \not\models u$, then $\mathscr{C}'(x,y,r,u) = \mathcal{E}$, and if $\mathsf{C}(x,y,s,v) \preceq \mathsf{C}(x,y,r,u)$, then $\mathscr{C}'(x,y,s,v) = \mathcal{E}$*

   if $p \models \neg r$ and $p \not\models u$, each commitment $\mathsf{C}(x,y,r,u)$ that is `Conditional`, expires, and all commitments $\mathsf{C}(x,y,s,v)$ weaker than $\mathsf{C}(x,y,r,u)$ expire

7. *$\forall y, \forall r, \forall u, \forall s, \forall v$: if $\mathscr{C}(x,y,r,u) = \mathcal{D}, p \models \neg u$, then $\mathscr{C}'(x,y,r,u) = \mathcal{V}$, and if $\mathsf{C}(x,y,s,v) \succeq \mathsf{C}(x,y,r,u)$, then $\mathscr{C}'(x,y,s,v) = \mathcal{V}$*

   if $p \models \neg u$, each commitment $\mathsf{C}(x,y,r,u)$ that is `Detached`, violates, and all commitments $\mathsf{C}(x,y,s,v)$ stronger than $\mathsf{C}(x,y,r,u)$ violate

8. *$\forall q, p \not\models q : \mathscr{B}'(x,q) = \mathscr{B}(x,q)$*

   if $p \not\models q$, all beliefs $\mathscr{B}(x,q)$ remain unaffected

9. *$\forall s, \forall f, p \not\models s, p \not\models f : \mathscr{G}'(x,s,f) = \mathscr{G}(x,s,f)$*

   if $p \not\models s, p \not\models f$, all goals $\mathsf{G}(x,s,f)$ remain unaffected

10. *$\forall r, \forall u, p \not\models r, p \not\models u : \mathscr{C}'(x,y,r,u) = \mathscr{C}(x,y,r,u)$*

    if $p \not\models r, p \not\models u$, all commitments $\mathsf{C}(x,y,r,u)$ remain unaffected

11. *$\forall s, \forall f$: if $\mathscr{G}(x,s,f) \in \{\mathcal{T}, \mathcal{F}, \mathcal{S}\}$, then $\mathscr{G}'(x,s,f) = \mathscr{G}(x,s,f)$*

    all goals $\mathsf{G}(x,s,f)$ that are `Terminated`, `Failed` or `Satisfied` remain unaffected

12. *$\forall y, \forall r, \forall u$: if $\mathscr{C}(x,y,r,u) \in \{\mathcal{T}, \mathcal{V}, \mathcal{S}\}$, then $\mathscr{C}'(x,y,r,u) = \mathscr{C}(x,y,r,u)$*

    all commitments $\mathsf{C}(x,y,r,u)$ that are `Terminated`, `Violated` or `Satisfied` remain unaffected

(ii): $\forall \langle gact, g \rangle \in \mathbb{A}, gact \in GACTS, g = \mathscr{G}(x,s,u)$:
$\langle \mathscr{B}', \mathscr{G}', \mathscr{C}' \rangle = L(\langle gact, g \rangle, \mathscr{B}, \mathscr{G}, \mathscr{C})$, where:

1. *$\mathscr{B}' = \mathscr{B}$* all beliefs are unaffected

2. *$\mathscr{C}' = \mathscr{C}$* commitments are not affected by goals

3. *if $gact = \mathsf{consider}$ and $\mathscr{G}(x,s,u) = \mathcal{N}$, then $\mathscr{G}'(x,s,u) = \mathcal{I}$*

   if agent x considers a goal $\mathsf{G}(x,s,u)$, the goal transitions from `Null` to `Inactive`

4. *$\forall t, \forall v$: if $gact = \textsf{activate}$ and $\mathscr{G}(x,s,u) = \mathcal{I}$, then $\mathscr{G}'(x,s,u) = \mathcal{A}$, and if $\mathsf{G}(x,t,v) \preceq \mathsf{G}(x,s,u)$, then $\mathscr{G}'(x,t,v) = \mathcal{A}$*

   if agent x activates an `Inactive` goal $\mathsf{G}(x,s,u)$, the goal transitions to `Active`, and all goals $\mathsf{G}(x,t,v)$ weaker than $\mathsf{G}(x,s,u)$ transition to `Active`

5. *if $gact = \textsf{suspend-G}$ and $\mathscr{G}(x,s,u) = \mathcal{I}$, then $\mathscr{G}'(x,s,u) = \mathcal{U}$*

   if agent x suspends an `Inactive` goal $\mathsf{G}(x,s,u)$, the goal transitions to `Suspended`

6. *if $gact = \textsf{reconsider}$ and $\mathscr{G}(x,s,u) = \mathcal{U}$, then $\mathscr{G}'(x,s,u) = \mathcal{I}$*

   if agent x reconsiders a `Suspended` goal $\mathsf{G}(x,s,u)$, the goal transitions to `Inactive`

7. *if $gact = \textsf{suspend-G}$ and $\mathscr{G}(x,s,u) = \mathcal{A}$, then $\mathscr{G}'(x,s,u) = \mathcal{U}$*

   if agent x suspends an `Active` goal $\mathsf{G}(x,s,u)$, the goal transitions to `Suspended`

8. *$\forall t, \forall v$: if $gact = \textsf{reactivate-G}$ and $\mathscr{G}(x,s,u) = \mathcal{U}$, then $\mathscr{G}'(x,s,u) = \mathcal{A}$, and if $\mathsf{G}(x,t,v) \preceq \mathsf{G}(x,s,u)$, then $\mathscr{G}'(x,t,v) = \mathcal{A}$*

   if agent reactivates a `Suspended` goal $\mathsf{G}(x,s,u)$, the goal transitions to `Active` and all goals $\mathsf{G}(x,t,v)$ weaker than $\mathsf{G}(x,s,u)$ transition to `Active`

9. *$\forall t, \forall v$: if $gact = \textsf{terminate}$ and $\mathscr{G}(x,s,u) \in \{\mathcal{I}, \mathcal{A}, \mathcal{U}\}$, then $\mathscr{G}'(x,s,u) = \mathcal{T}$, and if $\mathsf{G}(x,t,v) \succeq \mathsf{G}(x,s,u)$, then $\mathscr{G}'(x,t,v) = \mathcal{T}$*

   if agent x terminates a `Inactive`, `Active`, or `Suspended` goal $\mathsf{G}(x,s,u)$, the goal transitions to `Terminated`, and all goals $\mathsf{G}(x,t,v)$ stronger than $\mathsf{G}(x,s,u)$ transition to `Terminated`

10. *$\forall t, \forall v, \mathsf{G}(x,t,v) \in \mathscr{G}_x$: if $\mathsf{G}(x,t,v) \not\preceq \mathsf{G}(x,s,u)$ and $\mathsf{G}(x,t,v) \not\succeq \mathsf{G}(x,s,u)$, then $\mathscr{G}'(x,t,v) = \mathscr{G}(x,t,v)$*

    unrelated goals remain unaffected

(iii): $\forall \langle cact, c \rangle \in \mathbb{A}, cact \in CACTS, c = \langle x,y,s,u \rangle$:
$\langle \mathscr{B}', \mathscr{G}', \mathscr{C}' \rangle = L(\langle cact, c \rangle, \mathscr{B}, \mathscr{G}, \mathscr{C})$, where:

1. *$\mathscr{B}' = \mathscr{B}$* all beliefs are unaffected

2. *$\mathscr{G}' = \mathscr{G}$* goals are not affected by commitments

3. *$\forall t, \forall v$: if $cact = \textsf{create}$ and $\mathscr{C}(x,y,s,u) = \mathcal{N}$ and $\mathscr{B}(x,s) = \bot$, then $\mathscr{C}'(x,y,s,u) = \mathcal{C}$, and if $\mathsf{C}(x,y,t,v) \preceq \mathsf{C}(x,y,s,u)$, then $\mathscr{C}'(x,y,t,v) = \mathcal{C}$*

   if agent x creates a commitment $\mathsf{C}(x,y,s,u)$ and does not believe s, the commitment transitions from `Null` to `Conditional` and all commitments $\mathsf{C}(x,y,t,v)$ weaker than $\mathsf{C}(x,y,s,u)$ transition to `Conditional`

4. *if $cact = \textsf{create}$ and $\mathscr{C}(x,y,s,u) = \mathcal{N}$ and $\mathscr{B}(x,s) = \top$, then $\mathscr{C}'(x,y,s,u) = \mathcal{D}$*

   if agent x creates a commitment $\mathsf{C}(x,y,s,u)$ and believes s, the commitment transitions from `Null` to `Detached`

5. *if $cact = \textsf{suspend-C}$ and $\mathscr{C}(x,y,s,u) = \mathcal{C}$, then $\mathscr{C}'(x,y,s,u) = \mathcal{P}$*

   if agent x suspends a `Conditional` commitment $\mathsf{C}(x,y,s,u)$, the commitment transitions to `Pending`

6. *if cact* $=$ *suspend-C and* $\mathscr{C}(x,y,s,u)=\mathcal{D}$*, then* $\mathscr{C}'(x,y,s,u)=\mathcal{P}$

   if agent x suspends a `Detached` commitment $\mathsf{C}(x,y,s,u)$, the commitment transitions to `Pending`

7. $\forall t,\forall v$: *if cact* $=$ *reactivate-C,* $\mathscr{C}(x,y,s,u)=\mathcal{P}$*, and* $\mathscr{B}(x,s)=\bot$*, then* $\mathscr{C}'(x,y,s,u)=\mathcal{C}$*, and if* $\mathsf{C}(x,y,t,v)\preceq\mathsf{C}(x,y,s,u)$*, then* $\mathscr{C}'(x,y,t,v)=\mathcal{C}$

   if agent x reactivates a `Pending` commitment $\mathsf{C}(x,y,s,u)$ and does not believe s, then the commitment transitions to `Conditional`, and all commitments $\mathsf{C}(x,y,t,v)$ weaker than $\mathsf{C}(x,y,s,u)$ transition to `Conditional`

8. *if cact* $=$ *reactivate-C and* $\mathscr{C}(x,y,s,u)=\mathcal{P}$ *and* $\mathscr{B}(x,s)=\top$*, then* $\mathscr{C}'(x,y,s,u)=\mathcal{D}$

   if agent x reactivates a pending commitment $\mathsf{C}(x,y,s,u)$ and believes s, then the commitment transitions to `Detached`

9. $\forall t,\forall v$: *if cact* $=$ *cancel and* $\mathscr{C}(x,y,s,u)=\mathcal{C}$*, then* $\mathscr{C}'(x,y,s,u)=\mathcal{T}$*; if* $\mathsf{C}(x,y,t,v)\succeq\mathsf{C}(x,y,s,u)$*, then* $\mathscr{C}'(x,y,t,v)=\mathcal{T}$

   if agent x cancels a `Conditional` commitment $\mathsf{C}(x,y,s,u)$, then the commitment transitions to `Terminated`, and all commitments $\mathsf{C}(x,y,t,v)$ stronger than $\mathsf{C}(x,y,s,u)$ transition to `Terminated`

10. $\forall t,\forall v$: *if cact* $=$ *cancel and* $\mathscr{C}(x,y,s,u)=\mathcal{D}$*, then* $\mathscr{C}'(x,y,s,u)=\mathcal{V}$*; if* $\mathsf{C}(x,y,t,v)\succeq\mathsf{C}(x,y,s,u)$*, then* $\mathscr{C}'(x,y,t,v)=\mathcal{V}$

    if agent x cancels a `Detached` commitment $\mathsf{C}(x,y,s,u)$, then the commitment transitions to `Violated`, and all commitments $\mathsf{C}(x,y,t,v)$ stronger than $\mathsf{C}(x,y,s,u)$ transition to `Violated`

11. $\forall t,\forall v$: *if cact* $=$ *release and* $\mathscr{C}(x,y,s,u)=\mathcal{C}$*, then* $\mathscr{C}'(x,y,s,u)=\mathcal{T}$*; if* $\mathsf{C}(x,y,t,v)\succeq\mathsf{C}(x,y,s,u)$*, then* $\mathscr{C}'(x,y,t,v)=\mathcal{T}$

    if agent y releases a `Detached` commitment $\mathsf{C}(x,y,s,u)$, then the commitment transitions to `Terminated`, and all commitments $\mathsf{C}(x,y,t,v)$ stronger than $\mathsf{C}(x,y,s,u)$ transition to `Terminated`

12. $\forall t,\forall v$: *if cact* $=$ *release and* $\mathscr{C}(x,y,s,u)=\mathcal{D}$*, then* $\mathscr{C}'(x,y,s,u)=\mathcal{T}$*; if* $\mathsf{C}(x,y,t,v)\succeq\mathsf{C}(x,y,s,u)$*, then* $\mathscr{C}'(x,y,t,v)=\mathcal{T}$

    if agent y releases a `Conditional` commitment $\mathsf{C}(x,y,s,u)$, then the commitment transitions to `Terminated`, and all commitments $\mathsf{C}(x,y,t,v)$ stronger than $\mathsf{C}(x,y,s,u)$ transition to `Terminated`

13. $\forall t,\forall v,\mathsf{C}(x,y,t,v)\in\mathscr{C}_x$: *if* $\mathsf{C}(x,y,t,v)\not\preceq\mathsf{C}(x,y,s,u)$ *and* $\mathsf{C}(x,y,t,v)\not\succeq\mathsf{C}(x,y,s,u)$*, then* $\mathscr{C}'(x,y,t,v)=\mathscr{C}(x,y,t,v)$

    all commitments that neither stronger nor weaker than $\mathsf{C}(x,y,s,u)$ remain unaffected

Now we know how agent actions affect the system configuration, we can ask about the consistency of configurations.

Definition 30 (Consistent configuration). *A configuration* $S(x)=\langle\mathscr{B}_x,\mathscr{G}_x,\mathscr{C}_x\rangle$ *is* consistent *if the goals* $\mathscr{G}_x$ *are consistent and the commitments* $\mathscr{C}_x$ *are consistent, and there is commitment–belief consistency between* $\mathscr{C}_x$ *and* $\mathscr{B}_x$ *and goal–belief consistency between* $\mathscr{G}_x$

and $\mathscr{B}_x$. Further, a system configuration $S = \langle S(1), \ldots, S(n)\rangle$ is consistent *if every agent configuration $S(i)$ within S is consistent.*

Note that consistency of a multiagent system configuration S means that the configuration of every agent in the system is consistent; it does not require consistency between the agents. Indeed, there need not be multiagent goal consistency or multiagent commitment consistency within the system (see Section 2.3).

Lemma 1 states that a life cycle rule maps a well-defined configuration to another well-defined configuration.

Lemma 1 (Configuration update under life cycle rules)**.** *Let x be an agent with a configuration $S(x)$. Let L be a life cycle rule in which x applies action α. Let $S'(x) = \langle \mathscr{B}', \mathscr{G}', \mathscr{C}'\rangle$ be a tuple of functions for the beliefs, goals, and commitments of x, respectively, with the same signature as the respective state functions, after the application of α. Then $S'(x)$ exists and is a unique configuration.*

Proof. The three functions in $S'(x)$ have the form of belief, goal, and commitment state functions, respectively. We need to show that they are state functions, i.e., satisfy the required closure properties so that $S'(x)$ is a well-defined configuration.

First, consider $\mathscr{B}'$. We assume that agents maintain a consistent and logically-closed set of beliefs (Definition 2). Hence x's belief addition maintains these two properties, and in the life cycle rule definition, the addition operation of rule part (i).1 is the only way $\mathscr{B}'$ can differ from $\mathscr{B}$.

Note that the life cycle rules in Definition 29 are exhaustive over goal and commitment actions of the respective life cycles (Figures 1 and 2). Hence, regardless of the state of the goal or commitment in $S(x)$, the application of α always leads to a defined outcome in $S'(x)$.

Second, consider $\mathscr{G}'$. There are two properties to establish (Definition 12):

- Suppose $\mathscr{G}'(G_1) = \sigma$, where $\sigma \in \{\mathcal{A}, \mathcal{S}\}$ and $G_1 \succeq G_2$. Now $\mathscr{G}'$ is identical to $\mathscr{G}$, except for changes from part (i) or (ii) of Definition 29. By the definition of goal strength $\succeq$, if the state of G_1 is modified (from $\mathscr{G}$) by a sub-part of the rule, then the state of G_2 must be likewise modified by the same sub-part. Hence $\mathscr{G}'(G_2) = \sigma$.
- Suppose $\mathscr{G}'(G_1) = \sigma$, where $\sigma \in \{\mathcal{T}, \mathcal{F}\}$ and $G_2 \succeq G_1$. By a similar inspection of the rule (i) and (ii) sub-parts, then $\mathscr{G}'(G_2) = \sigma$.

Third, consider $\mathscr{C}'$. There are two properties to establish (Definition 6). $\mathscr{C}'$ can differ from $\mathscr{C}$ only due to parts (i) and (iii) of Definition 29. Again by the definition of commitment strength and inspection of the rule sub-parts, the two commitment closure properties must be maintained from $\mathscr{C}$.

Finally, by Definition 29, the effect of action α is deterministic; hence $S'(x)$ is unique. □

Lemma 1 means we can write $S(x) \xrightarrow{\alpha} S'(x)$ where $S'(x)$ is the (unique) configuration of x after the application of action α. Lemma 2 further states that a consistent configuration of an agent is mapped by a life cycle rule to a consistent configuration.

Lemma 2 (Configuration consistency under life cycle rules)**.** *Let x be an agent with a consistent configuration $S(x)$. Let L be a life cycle rule in which x applies action α and let $S(x) \xrightarrow{\alpha} S'(x)$. Then $S'(x)$ is a consistent configuration.*

Proof. Let $S(x)$ be a consistent configuration. Rule L takes one of three forms, from either a belief, goal, or commitment action, respectively. The three clauses of Definition 29 correspond to these three forms. Observe that in each case, goal consistency and commitment consistency is maintained, because no goal (respectively, commitment) is added or modified such that inconsistency arises. Further, for each action, the corresponding statement in Definition 29 ensures that the new state maintains commitment–belief consistency and goal–belief consistency. □

Concluding the section, we are now in the position to define the trace of system configurations.

Definition 31 (Successor configuration)**.** *Let M be a system of agents $\mathscr{A} = x_1, \ldots, x_n$ and let S and S' be two system configurations of M. Then S* progresses *to S' if and only if $\exists x \in \{1, \ldots, n\}$ and agent x applies action α, and $\forall y \in \{1, \ldots, n\}$: $S(y) \xrightarrow{\alpha} S'(y)$. We call S' a* successor system configuration *of S and say that S'* follows *from S.*

Such an action α of agent x, which moves the system configuration from S to S', may be an action corresponding to a life cycle rule (see the following discussion), or—if the agent follows our proposed practical rules of the next section—an action corresponding to a practical rule.

Intuitively, a trace is a sequence of successor configurations:

Definition 32 (Trace)**.** *A* trace *is a (possibly infinite) sequence of system configurations $S_1, S_2, \ldots$, where for $i > 0$, state S_i follows from S_{i-1} as per Definition 31. Configuration S_0 is the* initial configuration *of the system. In S_0, the sets $\mathscr{G}$ and $\mathscr{C}$ from each agent's configuration are empty.*

Definition 33 (Trace convergence)**.** *A trace* converges *to a configuration S_k if there is a k such that $S_i = S_k$, $\forall i \geq k$.*

We assume that all agents adopt the common operational semantics (life cycles) for commitments and goals. Agents must adopt the same life cycles for commitments since commitments are part of the social state; for goals, it is a reasonable technical convenience. That is, the different agents follow the same life cycle representation, which reflects the core semantics of commitments and goals, and hence the same life cycle rules. In contrast, as the next section will explain, practical rules may differ across agents, because an agent's practical rules reflect its decision-making.

3.4 Practical Rules

In contrast to life cycle rules, practical rules do not capture the necessary integrity requirements, but rather patterns of pragmatic reasoning that agents may or may not adopt under different circumstances. In that sense, rather than describing the mechanics of commitment and goal transitions, practical rules are the (potential) rules of an agent program, and they are specified by the designers of the agents. A practical rule, when executed, yields an action α. This action leads to a successor configuration of the multiagent system according to Definition 31. An example of a practical rule is: If an agent x has an active goal $\mathsf{G}(x, s, f)$ but believes it cannot achieve s by itself, then if it does not have a corresponding commitment $\mathsf{C}(x, y, s, u)$, create the commitment.

The practical rules may be neither complete nor deterministic: an agent may find itself at a loss as to how to proceed or may find itself with multiple options. Such nondeterminism corresponds naturally to a future-branching temporal model: each agent's multiplicity of options leads to many possible progressions of its configuration and of the configurations of its peers. The convergence results we show below in Section 4 indicate that our formulated set of rules is complete (i.e., sufficient) in a useful technical sense.

By virtue of adopting the practical rules as patterns of reasoning, the agent has the rules available as options. That is, the agent designer provides the agent with (some of) the practical rules at design time. At runtime, the agent can choose which, if any, of matching rules to execute in a given situation.[6] An agent may refine these rules to select from among a narrower set of the available options, for example, through other reasoning about its preferences and utilities.

In more detail, consider a situation where two practical rules apply—one of which would suspend an existing commitment C, and the second of which would terminate C and create a new commitment. The agent designer could for instance stipulate that practical rules that modify existing commitments are preferable for the agent over rules that create new commitments. Further, the agent could accommodate preferences over goals or plans (e.g., Visser et al., 2016). Our approach supports such meta-reasoning capabilities in principle, but we defer a careful investigation of it.

It is worth remarking further on interaction of agent decision making with the practical rules. Figure 3 on page 40 includes 'standard' BDI reasoning (Rao & Georgeff, 1992) regarding goal, intention, or plan selection. The figure shows the practical rule selection and subsequent goal and commitment state updates prior to the goal–plan deliberation and subsequent goal (and plan) state updates. Agent environmental actions whether from practical rules, plan actions, or otherwise, occur together in the 'Act' box. Recall that the intent of Figure 3 is to show a simple agent architecture for expository purposes. The literature has many variant BDI architectures; how practical rules as a form of commitment–goal reasoning can be incorporated with other elements of BDI reasoning is again a direction for future research. One effort in this direction is that set out in Baldoni et al. (2015).

The remainder of this section presents the practical rules we adopt here. We first introduce some terminology that practical rules employ and formally define their syntax and their operation on configurations, and then, in the following subsections, we detail the practical rules in three groups. The practical rules are expressed in a template form to obtain a compact presentation.

Definition 34 (End goal)**.** *An* end goal $G = \mathsf{G}(x, s, f)$ *of an agent* x *is a top-level goal adopted by* x *in order to achieve its desire that* s *be true.*

In other words, an end goal is adopted by an agent because it chooses to, not because it must (Harland et al., 2014).

[6] If an agent chooses to adopt a subset of the practical rules, i.e., to always ignore some rules, then the theoretical properties established later in the article hold only inasmuch as the conditions of the theorems still hold.

An agent x may attempt to achieve the success conditions of its end goals on its own, if it has the capability to do so—or if the agent lacks or prefers not to exercise the capability to bring about s on its own, agent x may create a set of commitments S_C such that $\bigwedge_j ant(C_j) \models s$, where $C_j \in S_C$. Hence the commitment derives from the end goal, as seen in practical rule OFFER. The debtor of each C_j may consider a set of goals to bring about C_j's antecedent. We refer to a goal in this set as a *means goal*.

Definition 35 (Means goal)**.** *Consider a commitment* $\mathsf{C}(x, y, s, u)$*. Agent y may consider a set of goals S_G such that $\bigwedge_i succ(G_i) \models s$, where $G_i \in S_G$. A goal $G_i \in S_G$ is a* means goal.

For example, a customer having the end goal of drinking coffee can lead (via a commitment) to the merchant having the means goal to provide a coffee.

Further, agent x may consider a set of goals to bring about the consequent of each C_j. We refer to a goal in this set as a *discharge goal*.

Definition 36 (Discharge goal)**.** *Consider a commitment* $\mathsf{C}(x, y, s, u)$*. Agent x may consider a set of goals S_G such that $\bigwedge_i succ(G_i) \models u$, where $G_i \in S_G$. A goal $G_i \in S_G$ is a* discharge goal.

The practical rules are designed to engender coherence between the sets of end goals and commitments, commitments and means goals, and commitments and discharge goals. For example, if all end goals related to a commitment are suspended, then a rule suspends the commitment. In another example, if a commitment is created, then a rule considers a set of means goals to detach the commitment.

Definition 37 (Practical rule instance)**.** *A* practical rule *of agent x in a multiagent system M is a mapping from a configuration of x to an action, i.e., $\langle \mathscr{B}, \mathscr{G}, \mathscr{C} \rangle_x \xrightarrow{\text{RULENAME}} \alpha$ where $\alpha \in \mathbb{A}$. We write $\langle \mathscr{B}, \mathscr{G}, \mathscr{C} \rangle \xrightarrow{\text{RULENAME}} \alpha$ when there is some agent $x \in \mathscr{A}$ such that $\langle \mathscr{B}, \mathscr{G}, \mathscr{C} \rangle_x \xrightarrow{\text{RULENAME}} \alpha$.* RULENAME *is optional.*

Note that the practical rules of the following subsections are constrained in that each has a 'leading' commitment or goal on the LHS, and a 'following' goal or commitment on the RHS. Intuitively, the leading commitment (respectively, goal) is the 'subject' of the practical rule, and the following goal (respectively, commitment) is part of the agent's state that will be modified if the action of the rule is executed. We call these the commitment–goal pair.

It remains to specify when and how a practical rule affects the system configuration. Consider agent $x \in \mathscr{A}$. Each practical rule of x specifies an action $\alpha \in \mathbb{A}$. This action operates on the system configuration to produce a successor configuration (Definition 31). The practical rule applies when its LHS is true in the current system configuration; if the agent selects an applicable practical rule to execute, the RHS action α produces a successor configuration.

Definition 38 (Configuration update under practical rules)**.** *Let R be a practical rule of agent x, $S(x)$ be the current configuration of x, and S the current system configuration. Let $RG \subseteq \mathscr{G}_x$ be the set of all goals and $RC \subseteq \mathscr{C}_x$ be the set of all commitments in the LHS of R. Then:*

1. *R is* applicable *if the LHS is true in $S(x)$, meaning that each member of RG and RC is in its designated state.*

2. *If R is applicable, then if x* executes *R, then the successor configuration S' is the application of the RHS of R on the system configuration according to Definition 31.*

Our agent architecture does not permit an agent to execute more than one practical rule in each execution cycle. First, if more than one practical rule is applicable for an agent's commitment–goal pair, the agent can choose at most one to execute on each execution cycle (recall Section 3.1). For example, suppose x's commitment C is `Conditional` and suppose that practical rules DETACH1 and GIVE UP (MEANS) are both applicable (these rules are defined in Section 3.4.3). Agent x can choose to apply the first or the second of these practical rules to C—or neither—but it cannot apply both practical rules during the same execution cycle.

Second, a practical rule can be executed at most once in each execution cycle for a commitment–goal pair. For example, suppose x's goal G is `Active` and suppose that practical rule Entice is applicable (as defined in Section 3.4.2). Agent x can choose to apply the practical rule, or not, but it cannot apply Entice twice for G in the same execution cycle. Note that the same practical rule can, if applicable, be applied to the same commitment–goal pair in subsequent execution cycles. For example, a commitment can be suspended and resumed several times over the course of its life.

3.4.1 Practical Rule Templates

Because practical rules are parameterized, in the following subsections we provide *practical rule templates*. Each template can generate a set of instantiations, each of which corresponds to a practical rule, as just defined. The agent can match the elements of its configuration with the terms in the practical rules, to obtain a set of actions to consider.

To map from the templates to practical rules, one can use a Prolog-like query language. The language is easily formalized but we do not go into implementation details here. The essential point is that the templates compile to practical rules as defined above. Operationally, then, if among the rules an agent finds a matching binding for a relevant proposition (e.g., u the antecedent of a commitment), it can act according to that rule, as in Prolog. In this way, the agents may treat practical rules as guidance in their decision making.

The syntax of a practical rule template has the form: $E \xrightarrow{\text{RULENAME}} \alpha$ where E is an expression, assumed to be a conjunction of the form of this goal is (or is not) in some state and that commitment is (or is not) in some state, about commitment and goal sets and their states, and $\alpha \in \mathbb{A}$ is a commitment or goal action set.

The semantics of practical rule templates is as follows. In essence, each template corresponds to every possible instantiation of it (with specific goals and commitments). In general, most such instantiated practical rules would not be applicable since their LHS conditions will not hold. However, if an agent's configuration satisfies the LHS condition E of a practical rule, then the rule applies for all matching goals or commitments, and it produces an action α on a goal (or a set of goals) or a commitment (or a set of commitments). The action updates the state of the goals or commitments.

Algorithm 1 Expansion of example practical rule template (Section 3.4.1)

```
for $G \in \mathscr{G}_x$ do
    if $G$ is Active then
        $T \leftarrow CSG(G)$
        for $C \in T$ do
            Apply action of rule to $C$
        end for
    end if
end for
```

For example, suppose the LHS of a practical rule template is: $\mathscr{G}(G) = \mathcal{A} \wedge C \in CSG(G)$, where $G = \mathsf{G}(x, s, f)$. Then the rule applies to each goal in an agent's configuration that is `Active`, and each commitment that is in the CSG set of that goal.

The practical rule templates have implicit universal quantification on all free variables: for readability, we do not write the $\forall$ symbols in front of each rule. For instance, the above template holds for all active G and for all $C \in CSG(G)$. Algorithm 1 gives a pseudocode-like representation for the procedural expansion of this rule template.

3.4.2 Practical Rules: From End Goal to Commitment

This subsection presents—in template form—an agent's practical rules of reasoning that involve the agent's end goal, and the commitments in which the agent is a debtor. As noted, we design the practical rules to engender coherence between the end goal and the commitment.

- entice: Suppose an agent x has an active goal $G = \mathsf{G}(x, s, \cdot)$. Then consider a set of commitments $C(x, y, \cdot, \cdot)$ whose detachment could lead to the success of the goal. Let $\omega = \bigwedge_i ant(C_i)$, where $C_i \in CSG(G)$ are the existing commitments, and $\Phi = \{C_j\}$ is a set of new commitments (to be created) such that $\bigwedge_j ant(C_j) \wedge \omega \models s$.

  $$\mathscr{G}(G) = \mathcal{A} \wedge \omega \not\models s \xrightarrow{\text{ENTICE}} \mathsf{create}(\Phi)$$

  Motivation: Agent x can satisfy its goal by creating the necessary commitments that together support the goal. This presumes that x lacks capabilities to bring about the goal's success condition s on its own.

- suspend offer: Suppose a goal $G = \mathsf{G}(x, \cdot, \cdot)$ is suspended. Then suspend each commitment supporting the goal that is not supporting other unsuspended goals.

  $$\mathscr{G}(G) = \mathcal{U} \wedge C \in CSG(G) \wedge G' \in GSC(C) \wedge \mathscr{G}(G') = \mathcal{U} \xrightarrow{\text{SUSPEND-O}} \mathsf{suspend\text{-}C}(C)$$

  Motivation: By suspending the commitments, x indicates to y that y may employ its resources in other tasks instead of working on the commitment.

  Note that the third and fourth conjuncts on the LHS of the rule enforce the requirement that the commitment in question does not support other unsupported goals.

- revive: Suppose a goal $G = \mathsf{G}(x, \cdot, \cdot)$ is active or satisfied and a commitment supporting that goal is pending, then reactivate the commitment.

  $$\mathscr{G}(G) \in \{\mathcal{A}, \mathcal{S}\} \wedge C \in CSG(G) \wedge \mathscr{C}(C) = \mathcal{P} \xrightarrow{\text{REVIVE}} \mathsf{reactivate\text{-}C}(C)$$

  Motivation: If the goal is active, agent x needs to reactivate the pending commitments to be able to satisfy the goal. If the goal is satisfied, agent x needs to reactivate the pending commitments to be able to satisfy the commitment.

- withdraw offer: Suppose a goal $G = \mathsf{G}(x, \cdot, \cdot)$ fails or is terminated. Then cancel each commitment supporting the goal that is not the supporting some other goal.

  $$\mathscr{G}(G) \in \{\mathcal{F}, \mathcal{T}\} \wedge C \in CSG(G) \wedge GSC(C) \setminus G = \emptyset \xrightarrow{\text{WITHDRAW-O}} \mathsf{cancel}(C)$$

  Motivation: A commitment is of no value once the goals for which it is created have failed or terminated.

- revive to withdraw: Suppose a goal $G = \mathsf{G}(x, s, f)$ fails or is terminated and a commitment supporting that goal is pending. If that commitment is not supporting some other goal, then reactivate the commitment.

  $$\mathscr{G}(G) \in \{\mathcal{F}, \mathcal{T}\} \wedge C \in CSG(G) \wedge GSC(C) \setminus G = \emptyset \wedge \mathscr{C}(C) = \mathcal{P} \xrightarrow{\text{REVIVE}} \mathsf{reactivate\text{-}C}(C)$$

  Motivation: If a goals fails or is terminated, and a commitment supporting that goal is pending, then x reactivates the commitment, and later cancels it using the withdraw offer rule. As the commitment life cycle in Figure 1 shows, an agent needs to reactivate a commitment before cancelling it.

3.4.3 Practical Rules: From Commitment to Means Goal

This subsection presents an agent's practical rules that involve the commitments in which the agent is a creditor, and the agent's means goals.

- detach1: Suppose a commitment $C = \mathsf{C}(x, y, r, \cdot)$ is conditional. Then consider a set of goals that could detach the commitment. Let $\omega = \bigwedge_i succ(G_i)$, where $G_i \in GAC(C)$, and Φ be a set of goals such that $\bigwedge_j succ(G_j) \wedge \omega \models r$ and $G_j \in \Phi$.

  $$\mathscr{C}(C) = \mathcal{C} \wedge \omega \not\models r \xrightarrow{\text{DETACH1}} \mathsf{consider}(\Phi)$$

  detach2: Suppose a commitment $\mathsf{C}(x, y, r, \cdot)$ is conditional, and $\Phi \subseteq GAC(C)$ such that $\bigwedge_i succ(G_i) \models r$, where $G_i \in \Phi$. If a goal $G \in \Phi$ is inactive, then activate the goal.

  $$\mathscr{C}(C) = \mathcal{C} \wedge G \in \Phi \wedge \mathscr{G}(G) = \mathcal{I} \xrightarrow{\text{DETACH2}} \mathsf{activate}(G)$$

  Motivation for these two rules: The creditor considers and activates goals to bring about the antecedent of a commitment, presumably to influence its debtor to discharge the commitment.

- suspend means goal: Suppose a commitment $C = \mathsf{C}(x, y, \cdot, \cdot)$ is pending. Then suspend each goal that provides antecedent support to the commitment that is not supporting other commitments.

  $$\mathscr{C}(C) = \mathcal{U} \wedge G \in GAC(C) \wedge CAG(G) \backslash C = \emptyset \wedge CCG(G) \backslash C = \emptyset \xrightarrow{\text{SUSPEND-M}} \textsf{suspend-G}(G)$$

  Motivation: By suspending the goal, the agent may employ its resources to work on other goals.

- reactivate means goal: Suppose a commitment $C = \mathsf{C}(x, y, \cdot, \cdot)$ is conditional, and a goal providing antecedent support to C is suspended. Then reactivate the goal.

  $$\mathscr{C}(C) = \mathcal{C} \wedge G \in GAC(C) \wedge \mathscr{G}(G) = \mathcal{U} \xrightarrow{\text{REACTIVATE-M}} \textsf{reactivate-G}(G)$$

  Motivation: An active means goal is necessary for the agent to detach the commitment.

- abandon means goal: Suppose a commitment $C = \mathsf{C}(x, y, \cdot, \cdot)$ is expired or terminated. Then terminate each goal providing antecedent support to C that is not providing support (antecedent or consequent) to some other commitment.

  $$\mathscr{C}(C) \in \{\mathcal{E}, \mathcal{T}\} \wedge G \in GAC(C) \wedge CAG(G) \backslash C = \emptyset \wedge CCG(G) \backslash C = \emptyset \xrightarrow{\text{ABANDON-M}} \textsf{terminate}(G)$$

  Motivation: The goal is not needed since the commitment for which it is created no longer exists.

  Note we do not need to deal with `Violated` commitments because they cannot occur here: a conditional commitment cannot become `Violated` without first becoming `Detached`. Similarly abandon-d below need not deal with `Expired` commitments.

- give up (means): Suppose a commitment $C = \mathsf{C}(x, y, r, \cdot)$ is conditional, and it lacks sufficient antecedent support. Then agent y releases x from C. Let $\omega = \bigwedge_i succ(G_i)$, where $G_i \in GSC(C)$.

  $$\mathscr{C}(C) = \mathcal{C} \wedge \omega \not\models r \xrightarrow{\text{GIVEUP-M}} \textsf{release}(C)$$

  Motivation: The agent gives up pursuing its commitment by releasing the debtor from its responsibilities in the commitment.

3.4.4 Practical Rules: From Commitment to Discharge Goal

This subsection presents an agent's practical rules that involve the commitments in which the agent is a debtor, and the agent's discharge goals.

- deliver1: Suppose a commitment $C = \mathsf{C}(x, y, \cdot, u)$ is detached. Then consider goals that will discharge the commitment. Let $\omega = \bigwedge_i succ(G_i)$, where $G_i \in GCC(C)$, and $\Phi = \{G_j\}$ is a set of goals such that $\bigwedge_j succ(G_j) \wedge \omega \models u$.

  $$\mathscr{C}(C) = \mathcal{D} \wedge \omega \not\models u \xrightarrow{\text{DELIVER1}} \textsf{consider}(\Phi)$$

deliver2: Suppose a commitment $C = \mathsf{C}(x, y, \cdot, u)$ is detached, and $\Phi \subseteq GCC(C)$ such that $\bigwedge_i succ(G_i) \models u$, where $G_i \in \Phi$. If a goal $G \in \Phi$ is inactive, then activate the goal.

$$\mathscr{C}(C) = \mathcal{D} \wedge G \in \Phi \wedge \mathscr{G}(G) = \mathcal{I} \xrightarrow{\text{DELIVER2}} \mathsf{activate}(\Phi)$$

Motivation for these two rules: The agent is honest in that it considers or activates a set of *discharge* goals that (if successful) would lead to discharging its commitment.

- suspend discharge goal: Suppose a commitment $C = \mathsf{C}(x, y, \cdot, \cdot)$ is pending. Then suspend each goal that provides consequent support to the commitment where that goal does not support other unsuspended commitments.

$$\mathscr{C}(C) = \mathcal{P} \wedge G \in GCC(C) \wedge CAG(G) \backslash C = \emptyset \wedge CCG(G) \backslash C = \emptyset \xrightarrow{\text{SUSPEND-D}} \mathsf{suspend\text{-}G}(G)$$

Motivation: By suspending the goal, the agent may employ its resources to work on other goals.

- reactive discharge goal: Suppose a commitment $C = \mathsf{C}(x, y, \cdot, \cdot)$ is detached, and a goal providing consequent support to C is suspended. Then reactivate the goal.

$$\mathscr{C}(C) = \mathcal{D} \wedge G \in GCC(C) \wedge \mathscr{G}(G) = \mathcal{U} \xrightarrow{\text{REACTIVATE-D}} \mathsf{reactivate\text{-}G}(G)$$

Motivation: An active *discharge* goal is necessary for the agent to satisfy the commitment.

- abandon discharge goal: Suppose a commitment $C = \mathsf{C}(x, y, \cdot, \cdot)$ is terminated (y releases x from the commitment) or violated (x cancels the commitment). Then terminate each goal providing consequent support to C that is not the only goal providing support (antecedent or consequent) to some other commitment.

$$\mathscr{C}(C) \in \{\mathcal{T}, \mathcal{V}\} \wedge G \in GCC(C) \wedge CAG(G) \backslash C = \emptyset \wedge CCG(G) \backslash C = \emptyset \xrightarrow{\text{ABANDON-D}} \mathsf{terminate}(G)$$

Motivation: The *discharge* goal is not needed since the commitment for which it is created no longer exists.

- give up (discharge): Suppose a commitment $C = \mathsf{C}(x, y, \cdot, \cdot)$ is detached, and no goal that provides consequent support to C exists. Then agent x cancels C. Let $\omega = \bigwedge_i succ(G_i)$, where $G_i \in GCC(C)$.

$$\mathscr{C}(C) = \mathcal{D} \wedge \omega \not\models s \xrightarrow{\text{GIVEUP-D}} \mathsf{cancel}(C)$$

Motivation: The agent quits pursuing its commitment by cancelling it and thereby violating it. The agent may be better off violating the commitment and suffering any sanctions compared to satisfying the commitment.

In the next section we prove properties of the whole set of practical rules that show their potential benefit for an agent designer. The properties are built on the notion of the trace of system configurations (Section 3.3.7).

4. Establishing Convergence Properties

There can be no guarantee in general that any agent will succeed with its goals, because the agent may lack the capabilities and resources to achieve them all, and other agents may not wish to help it; and because of exogenous effects of the environment. Thus we cannot prove that an agent's goals and commitments will all reach successful terminal states without further assumptions. For this reason, first we motivate the idea of a *coherent state* and ask whether we can guarantee that a coherent state of a multiagent system will be reached (repeatedly, if necessary), no matter how the agents in the multiagent system decide to act—provided that they act according to the life cycles and practical rules we have given. Informally, in a coherent state, corresponding goals and commitments match each other. That is, informally, for a related commitment and goal pair, if one succeeds, then the other does; if one fails, then the other does; if one is active, then the other is; if one is suspended, then the other is. Note that our theorems apply to sets of commitments and goals, and hence to all of the commitments and goals in the multiagent system.

We show that the practical rules given in Section 3.4 are sufficient for an agent to reach a coherent state, under certain mild assumptions. We then show, adopting stronger assumptions about the multiagent system and the environment, that the agents can collectively succeed in achieving all their end goals.

Coherence between the goals and commitments of an agent is crucial, as otherwise an agent may fail to achieve its goals, or may expend effort in satisfying unnecessary commitments. As an example, consider an agent x that has a goal $\mathsf{G}(x, s, f)$, but that lacks the capability to bring about s. If x fails to create a commitment $\mathsf{C}(x, y, s, u)$ towards some other agent y, then there is no clear path for x to achieve the goal. However, note that creating the commitment by itself does not guarantee that x achieves the goal, since y may fail to bring about s. In this case, then x needs to create a new commitment or to abandon the goal.

4.1 Preliminaries

This subsection formally defines our notion of coherence. Informally, an agent configuration is coherent if it satisfies a set of coherence properties, which are expressions over the beliefs, goals, and commitments of an agent. Specifically, such a configuration has coherence between: (1) end goals and commitments, (2) commitments and means goals, and (3) commitments and discharge goals.

Definition 39 (Coherent configurations)**.** *A configuration is* coherent *if and only if it satisfies all the properties below.*

End goal and commitment: *Suppose $C \in maxc(\{\mathcal{C}, \mathcal{D}\})$ is a maximal commitment, $G \in maxg(\mathcal{A})$ is a maximal goal, and Φ is a minimal subset of $CSG(G)$ such that $C_i \in \Phi$ and $\bigwedge_i ant(C_i) \models succ(G)$. A coherent configuration satisfies:*

$$\begin{aligned} GSC(C) = \emptyset &\implies \mathscr{C}(C) \in \{\mathcal{T}, \mathcal{E}, \mathcal{S}, \mathcal{V}\}, \\ \mathscr{G}(G) = \mathcal{S} &\implies \forall C \in \Phi, \mathscr{C}(C) = \mathcal{D}, \\ \mathscr{G}(G) = \mathcal{A} &\implies \forall C \in \Phi, \mathscr{C}(C) = \mathcal{C}, \\ \forall G \in GSC(C), \mathscr{G}(G) = \mathcal{U} &\implies \mathscr{C}(C) \in \{\mathcal{P}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V}\}. \end{aligned}$$

(The intuition, approximately, is that if there are no end goals supporting a commitment, the commitment can be in only a limited number of states; and if a goal is `Satisfied`, `Active`, or `Suspended`, the corresponding commitments must be in a certain state or states.)

Commitment and means goal: *Suppose $C \in maxc(\{\mathcal{C}, \mathcal{D}\})$ is a maximal commitment, and Ψ is a minimal subset of $GAC(C)$ such that $G_i \in \Psi$ and $\bigwedge_i succ(G_i) \models ant(C)$. For each goal $G_i \in GAC(C)$, the set of commitments with antecedent support of G_i is $CAG(G_i)$ and with consequent support of G_i is $CCG(G_i)$. Suppose $\Omega = \bigcup_i CAG(G_i) \cup \bigcup_i CCG(G_i)$. A coherent configuration satisfies:*

$$\begin{aligned} \forall G \in \Psi, \mathscr{G}(G) = \mathcal{S} &\implies \mathscr{C}(C) = \mathcal{D}, \\ \Omega = \emptyset &\implies GAC(C) = \emptyset, \\ \forall C \in \Omega, \mathscr{C}(C) = \mathcal{P} &\implies \forall G \in GAC(C), \mathscr{G}(G) = \mathcal{U}, \\ \mathscr{C}(C) = \mathcal{C} &\implies \forall G \in \Psi, \mathscr{G}(G) = \mathcal{A}. \end{aligned}$$

(The intuition, approximately, is that if a means goal in Ψ is `Satisfied`, the corresponding commitment must be `Detached`. If there are no commitments in Ω then there can be no means goals providing antecedent support. If a commitment is `Pending` or `Conditional` then the corresponding means goals must be, respectively, `Suspended` or `Active`.)

Commitment and discharge goal: *Suppose $C \in maxc(\{\mathcal{C}, \mathcal{D}\})$ is a maximal commitment, and Ψ is a minimal subset of $GCC(C)$ such that $G_i \in \Psi$ and $\bigwedge_i succ(G_i) \models con(C)$. For each goal $G_i \in GCC(C)$, the set of commitments with consequent support of G_i is $CCG(G_i)$ and with antecedent support of G_i is $CAG(G_i)$. Suppose $\Omega = \bigcup_i CCG(G_i) \cup \bigcup_i CAG(G_i)$. A coherent configuration satisfies:*

$$\begin{aligned} \forall G \in \Psi, \mathscr{G}(G) = \mathcal{S} &\implies \mathscr{C}(C) = \mathcal{S}, \\ \Omega = \emptyset &\implies GCC(C) = \emptyset, \\ \forall C \in \Omega, \mathscr{C}(C) = \mathcal{P} &\implies \forall G \in GCC(C), \mathscr{G}(G) = \mathcal{U}, \\ \mathscr{C}(C) = \mathcal{D} &\implies \forall G \in \Psi, \mathscr{G}(G) = \mathcal{A}. \end{aligned}$$

(The intuition, approximately, is that if a discharge goal is `Satisfied` then the corresponding commitment must be `Satisfied`. If there are no commitments in Ω then there can be no discharge goals providing consequent support. If a commitment is `Pending` or `Detached` then the corresponding discharge goals must be, respectively, `Suspended` or `Active`.)

A configuration that fails to satisfy any one of the above properties is incoherent.

Definition 40 (Trace convergence to coherent configuration)**.** *If a trace $S_1, S_2, \ldots$ converges to a configuration S_k, and S_k is a coherent configuration, then we say that the trace* converges to a coherent configuration.

We will prove convergence of traces under two assumptions: fairness of agents' actions, and absence of cycling in commitment and goal states. First, the *fairness assumption* is that all agents act towards achieving their commitments and goals; there will be no 'starvation' of a commitment or goal in the system. Hence, all goals and commitments in the system will reach a terminal state, either positive (e.g., `Satisfied`) or negative (e.g., `Failed`).

Second, for convergence we cannot have forever-cycling commitments or goals (e.g., between `Active` and `Suspended`). Informally, if on a trace a specific commitment returns to the same (commitment) state infinitely often, we say the commitment is *cycling*; likewise for goals. Formally:

Definition 41 (Commitment cycling)**.** *Let $C = \mathsf{C}(x, y, p, q)$ be a commitment and τ be a trace of states $\langle S_0, S_1, \ldots \rangle$. Suppose $\mathscr{C}(C) = \sigma$ in some state S_i and in some subsequent state S_j, where $j > i$. If τ contains infinite pairs of $\langle S_i, S_j \rangle$, then we say that C is* cycling *on τ.*

Definition 42 (Goal cycling)**.** *Let $G = \mathsf{G}(x, s, f)$ be a goal and τ be a trace of states $\langle S_0, S_1, \ldots \rangle$. Suppose $\mathscr{G}(G) = \sigma$ in some state S_i and in some subsequent state S_j, where $j > i$. If τ contains infinite pairs of $\langle S_i, S_j \rangle$, then we say that G is* cycling *on τ.*

4.2 Convergence Theorems

This subsection presents the convergence theorems and their proofs. Intuitively, Theorem 1 states that an agent configuration repeatedly becomes coherent if it is incoherent earlier on a trace, assuming the agent follows the practical rules from Section 3, and that no commitment or goal cycles on that trace.

Theorem 1. *Let $\tau = \langle S_0, S_1, \ldots \rangle$ be a trace in which no commitment or goal is cycling. Then for any state S_i in τ, if S_i is not coherent, there is a subsequent state S_j, $j > i$, in τ such that S_j is coherent.*

Proof. The proof considers each possible way that part of the coherence properties can be violated, and shows that necessarily there exists a future state S_j, $j > i$, that is coherent. In turn we treat the three parts of Definition 39, and for each part its four properties.

End goal and commitment: Suppose $C \in maxc(\{\mathcal{C}, \mathcal{D}\})$ is a maximal commitment, $G \in maxg(\mathcal{A})$ is a maximal goal, and Φ is a minimal subset of $CSG(G)$ such that $C_k \in \Phi$ and $\bigwedge_k ant(C_k) \models succ(G)$ in state S_i.

1. $GSC(C) = \emptyset \implies \mathscr{C}(C) \in \{\mathcal{T}, \mathcal{E}, \mathcal{S}, \mathcal{V}\}$: Suppose that in state S_i, $GSC(C)$ becomes empty, that is, all of the goals $G \in GSC(C)$ are either in the failed or terminated state. State S_i is incoherent since it violates: $GSC(C) = \emptyset \implies \mathscr{C}(C) \in \{\mathcal{T}, \mathcal{E}, \mathcal{S}, \mathcal{V}\}$. In state S_i, WITHDRAW OFFER applies. Suppose in some future state S_j, agent x executes this rule, that is, x cancels C. State S_j satisfies: $GSC(C) = \emptyset \implies \mathscr{C}(C) \in \{\mathcal{T}, \mathcal{E}, \mathcal{S}, \mathcal{V}\}$. If agent x does not apply WITHDRAW OFFER, and in state S_j, C expires, satisfies or violates, then state S_j satisfies: $GSC(C) = \emptyset \implies \mathscr{C}(C) \in \{\mathcal{T}, \mathcal{E}, \mathcal{S}, \mathcal{V}\}$.
2. $\mathscr{G}(G) = \mathcal{S} \implies \forall C \in \Phi, \mathscr{C}(C) = \mathcal{D}$: Suppose that in state S_i, $\bigwedge_k ant(C_k)$ holds, which implies that G is satisfied, and all commitments $C_k \in \Phi$ are detached. State S_i satisfies: $\mathscr{G}(G) = \mathcal{S} \implies \forall C \in \Phi, \mathscr{C}(C) = \mathcal{D}$.

3. $\mathscr{G}(G) = \mathcal{A} \implies \forall C \in \Phi, \mathscr{C}(C) = \mathcal{C}$: Suppose that in state S_i, agent x considers and activates G. Further, suppose that sufficient support for that goal is not present in $CSG(G)$, that is, no subset Φ exists. State S_i is incoherent since it violates: $\mathscr{G}(G) = \mathcal{A} \implies \forall C \in \Phi, \mathscr{C}(C) = \mathcal{C}$.

   In state S_i, entice applies. Suppose in some future state S_j, agent x executes this rule and creates a set of commitments such that Φ exists. The resulting state S_j satisfies: $\mathscr{G}(G) = \mathcal{A} \implies \forall C \in \Phi, \mathscr{C}(C) = \mathcal{C}$.

   Suppose agent x does not apply entice, and in state S_j, G terminates or fails. State S_j satisfies: $\mathscr{G}(G) = \mathcal{A} \implies \forall C \in \Phi, \mathscr{C}(C) = \mathcal{C}$.

   Note that due to the fairness assumption, agent x eventually applies entice or terminates G (or G fails).

4. $\forall G \in GSC(C), \mathscr{G}(G) = \mathcal{U} \implies \mathscr{C}(C) \in \{\mathcal{P}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V}\}$: Suppose in state S_i, agent x suspends all goals $G \in GSC(C)$. State S_i is incoherent since it violates: $\forall G \in GSC(C), \mathscr{G}(G) = \mathcal{U} \implies \mathscr{C}(C) \in \{\mathcal{P}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V}\}$.

   In state S_i, suspend offer applies. Suppose in some future state S_j, agent x executes this rule and suspends C. The resulting state S_j satisfies: $\forall G \in GSC(C)$, $\mathscr{G}(G) = \mathcal{U} \implies \mathscr{C}(C) \in \{\mathcal{P}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V}\}$.

   Suppose agent x does not apply suspend offer, and in state S_j, all goals in $GSC(C)$ are suspended, and C expires, detaches, satisfies or violates. State S_j satisfies: $\forall G \in GSC(C), \mathscr{G}(G) = \mathcal{U} \implies \mathscr{C}(C) \in \{\mathcal{P}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V}\}$.

Commitment and means goal: Suppose $C \in maxc(\{\mathcal{C}, \mathcal{D}\})$ is a maximal commitment, and Ψ is a minimal subset of $GAC(C)$ such that $G_k \in \Psi$ and $\bigwedge_k succ(G_k) \models ant(C)$. For each goal $G_k \in GAC(C)$, the set of commitments with antecedent support of G_k is $CAG(G_k)$ and with consequent support of G_k is $CCG(G_k)$. Let $\Omega = \bigcup_k CAG(G_k) \cup \bigcup_k CCG(G_k)$.

1. $\forall G \in \Psi, \mathscr{G}(G) = \mathcal{S} \implies \mathscr{C}(C) = \mathcal{D}$: Suppose in state S_i, all $G_k \in \Psi$ satisfy. This implies that $ant(C)$ holds, that is, C is detached. State S_k satisfies: $\forall G \in \Psi, \mathscr{G}(G) = \mathcal{S} \implies \mathscr{C}(C) = \mathcal{D}$.

2. $\Omega = \emptyset \implies GAC(C) = \emptyset$: Suppose that in state S_i, all commitments in Ω are expired or terminated, that is, Ω is empty. State S_i is incoherent since it violates: $\Omega = \emptyset \implies GAC(C) = \emptyset$.

   In state S_i, abandon means goal applies to each $G_k \in GAC(C)$. Suppose agent y executes this rule and terminates each G_k such that in state S_j all $G_k \in GAC(C)$ are terminated. State S_j satisfies: $\Omega = \emptyset \implies GAC(C) = \emptyset$.

   Suppose agent y does not apply abandon means goal, and all $G_k \in GAC(C)$ eventually fail or satisfy in state S_j, that is, in state S_j, $GAC(C) = \emptyset$. State S_j satisfies: $\Omega = \emptyset \implies GAC(C) = \emptyset$.

3. $\forall C \in \Omega, \mathscr{C}(C) = \mathcal{P} \implies \forall G \in GAC(C), \mathscr{G}(G) = \mathcal{U}$: Suppose in state S_i, all commitments in Ω are suspended. State S_i is incoherent since it violates: $\forall C \in \Omega, \mathscr{C}(C) = \mathcal{P} \implies \forall G \in GAC(C), \mathscr{G}(G) = \mathcal{U}$.

   In state S_i, for each $G_k \in GAC(C)$, suspend means goal applies. Agent y suspends each goal $G_k \in GAC(C)$ over multiple future states. Suppose in a future state S_j, all

G_k are suspended. The resulting state S_j satisfies: $\forall C \in \Omega, \mathscr{C}(C) = \mathcal{P} \implies \forall G \in GAC(C), \mathscr{G}(G) = \mathcal{U}$.

Suppose agent y does not apply SUSPEND MEANS GOAL . Due to the fairness assumption, a commitment cannot remain in pending state forever. In a future state S_j, some $C \in \Omega$ will not be in state pending, that is, $\forall C \in \Omega, \mathscr{C}(C) = P$ will not hold in S_j. Then the state S_j satisfies: $\forall C \in \Omega, \mathscr{C}(C) = \mathcal{P} \implies \forall G \in GAC(C), \mathscr{G}(G) = \mathcal{U}$.

4. $\mathscr{C}(C) = \mathcal{C} \implies \forall G \in \Psi, \mathscr{G}(G) = \mathcal{A}$: Suppose in state S_i, agent creates a maximal commitment $C \in maxc(\{\mathcal{C}, \mathcal{D}\})$. Further, suppose that sufficient antecedent support for C is not present in $GAC(C)$, that is, no subset Ψ exists. State S_i is incoherent since it violates: $\mathscr{C}(C) = \mathcal{C} \implies \forall G \in \Psi, \mathscr{G}(G) = \mathcal{A}$.

   In state S_i, DETACH1 applies. Agent y considers a minimal set of goals Ψ over multiple future states. (Note, by the definition of DETACH1, the agent considers a minimal set.) For each goal in Ψ, DETACH2 applies. Agent x activates each goal $G_k \in \Psi$ over multiple future states. Suppose in a future state S_j, all $G_k \in \Psi$ are active. The resulting state S_j satisfies: $\mathscr{C}(C) = \mathcal{C} \implies \forall G \in \Psi, \mathscr{G}(G) = \mathcal{A}$.

   Suppose agent y does not apply DETACH1. Due to the fairness assumption, C cannot remain in conditional state forever. Suppose in state S_j, C is not conditional. State S_j satisfies: $\mathscr{C}(C) = \mathcal{C} \implies \forall G \in \Psi, \mathscr{G}(G) = \mathcal{A}$.

Commitment and discharge goal: Suppose $C \in maxc(\{\mathcal{C}, \mathcal{D}\})$ is a maximal commitment, and Ψ is a minimal subset of $GCC(C)$ such that $G_k \in \Psi$ and $\bigwedge_k succ(G_k) \models con(C)$. For each goal $G_k \in GCC(C)$, the set of commitments with consequent support of G_k is $CCG(G_k)$ and with antecedent support of G_k is $CAG(G_k)$. Suppose $\Omega = \bigcup_k CCG(G_k) \bigcup_k CAG(G_k)$.

1. $\forall G \in \Psi, \mathscr{G}(G) = \mathcal{S} \implies \mathscr{C}(C) = \mathcal{S}$: Suppose in a future state S_j, all goals Ψ satisfy. Then following the life cycle rule of Definition 29, commitment C satisfies in S_j. State S_j satisfies: $\forall G \in \Psi, \mathscr{G}(G) = \mathcal{S} \implies \mathscr{C}(C) = \mathcal{S}$.
2. $\Omega = \emptyset \implies GCC(C) = \emptyset$: Suppose in a future state S_i, all commitments in Ω are expired or terminated or violated. State S_i is incoherent since it violates: $\Omega = \emptyset \implies GCC(C) = \emptyset$.

   In state S_i, for each $G_k \in GCC(C)$, ABANDON DISCHARGE GOAL applies. Agent x terminates each goal $G_k \in GCC(C)$ over multiple future states. Finally, in a future state S_j, all G_k are terminated. The resulting state S_j satisfies: $\Omega = \emptyset \implies GCC(C) = \emptyset$.

   Suppose agent x does not apply ABANDON DISCHARGE GOAL. Due to the fairness assumption, all $G_k \in GCC(C)$ cannot remain in a non-terminal state forever. Suppose in state S_j, each $G_k \in GCC(C)$ is in terminated, satisfied, or failed state, that is, $GCC(C) = \emptyset$. State S_j satisfies: $\Omega = \emptyset \implies GCC(C) = \emptyset$.
3. $\forall C \in \Omega, \mathscr{C}(C) = \mathcal{P} \implies \forall G \in GCC(C), \mathscr{G}(G) = \mathcal{U}$: Suppose that in state S_i, all commitments in Ω are suspended. State S_i is incoherent since it violates: $\forall C \in \Omega, \mathscr{C}(C) = \mathcal{P} \implies \forall G \in GCC(C), \mathscr{G}(G) = \mathcal{U}$.

   In state S_i, for each $G_i \in GCC(C)$, SUSPEND DISCHARGE GOAL applies. Agent y suspends each goal $G_k \in GCC(C)$ over multiple future states. Finally, in a future state S_j, all G_k are suspended. The resulting state S_j satisfies: $\forall C \in \Omega, \mathscr{C}(C) = \mathcal{P} \implies \forall G \in GCC(C), \mathscr{G}(G) = \mathcal{U}$.

Suppose agent x does not apply SUSPEND DISCHARGE GOAL. Due to the fairness assumption, a commitment cannot remain in pending state forever. In a future state S_j, some $C \in \Omega$ will not be in state pending, that is, $\forall C \in \Omega, \mathscr{C}(C) = P$ will not hold in S_j. State S_j satisfies: $\forall C \in \Omega, \mathscr{C}(C) = \mathcal{P} \implies \forall G \in GCC(C), \mathscr{G}(G) = \mathcal{U}$.

4. $\mathscr{C}(C) = \mathcal{D} \implies \forall G \in \Psi, \mathscr{G}(G) = \mathcal{A}$: Consider a maximal commitment $C \in maxc(\mathcal{D})$ in state S_i for which sufficient consequent support is not present, that is, no subset Ψ exists. State S_i is incoherent since it violates: $\mathscr{C}(C) = \mathcal{D} \implies \forall G \in \Psi, \mathscr{G}(G) = \mathcal{A}$.

   In state S_i, DELIVER1 applies. Agent x considers a minimal set of goals Ψ over multiple future states. For each goal in Ψ, DELIVER2 applies. Agent x activates each goal $G_k \in \Psi$ over multiple future states. Suppose in state S_j, all $G_k \in \Psi$ are active. The resulting state S_j satisfies: $\mathscr{C}(C) = \mathcal{D} \implies \forall G \in \Psi, \mathscr{G}(G) = \mathcal{A}$.

   Suppose agent x does not apply DELIVER1. Due to the fairness assumption C cannot remain in detached state forever. Suppose in state S_j, C is not detached. State S_j satisfies: $\mathscr{C}(C) = \mathcal{D} \implies \forall G \in \Psi, \mathscr{G}(G) = \mathcal{A}$. □

Recall that Theorem 1 applies only to those traces in which no commitment or goal cycles. We now prove a lemma that no commitment can cycle in a trace if no goal is cycling.

Lemma 3. *Let $\tau = \langle S_0, S_1, \ldots \rangle$ be a trace in which no goal is cycling. Then no commitment cycles in τ.*

The goals in the multiagent system can be end goals, or non-end goals, i.e., means or discharge goals. Since we aim to study the properties of a system whose agents follows our practical rules, we assume the agents do so. The proof considers each type of goal.

Proof. Suppose that in τ no goal cycles but some commitment C cycles. According to our semantics, every commitment is linked to some goal: either C arises because of an end goal, or eventually there is a means or discharge goal G which arises from C.

Hence, first suppose G is an end goal of some agent x in the multiagent system, and C is a commitment such that $C \in CSG(G)$. Further, suppose goal G does not cycle but C cycles on the trace τ. The only way a commitment cycles according to our semantics is when it is suspended and then reactivated. Assuming that the agents in the multiagent system follow our practical rules, the debtor of C suspends C following SUSPEND OFFER when G is suspended, and reactivates C using REVIVE if G is subsequently reactivated. However, this implies that the end goal G is cycling, which contradicts the supposition.

Second, G is a means goal related to a commitment C such that $G \in GAC(C)$. Further, suppose G does not cycle, but C cycles on the trace τ. Since commitment C is cycling, it is suspended and then reactivated. When C is suspended, the agent follows SUSPEND MEANS GOAL, and suspends the means goal G. When C is reactivated, the agent follows REACTIVATE MEANS GOAL, and reactivates the means goal. This implies that the means goal G is cycling, which contradicts the supposition.

Third, suppose G is a discharge goal related to a commitment C such that $G \in GCC(C)$. Further, suppose G does not cycle, but C cycles on the trace τ. Since commitment C is cycling, it is suspended and then reactivated. When C is suspended, the agent follows suspend discharge goal, and suspends the discharge goal G. When C is reactivated, the agent follows reactivate discharge goal, and reactivates the means goal. This implies that the means goal G is cycling, which contradicts the supposition. □

The lemma help us prove Theorem 2.

Theorem 2. *Let $\tau = \langle S_0, S_1, \ldots \rangle$ be a trace on which no goal is cycling. Then for any state S_i on τ, if S_i is not coherent, there is a subsequent state S_j, $j > i$, on τ such that S_j is coherent.*

Proof. By Lemma 3, if no goal cycles on τ, then no commitment cycles on τ. By Theorem 1, if no goal and no commitment cycles on τ, then the result follows. □

We can strengthen the result of Theorem 2 to require only end goals to not cycle, in order for convergence to occur. We need a lemma.

Lemma 4. *Let $\tau = \langle S_0, S_1, \ldots \rangle$ be a trace on which no end goal is cycling. Then no goal cycles on τ.*

Proof. Consider a commitment $C = \mathsf{C}(x, y, s, u)$ and the end goal $G = \mathsf{G}(x, s, f)$ from which it arises. By hypothesis, G does not cycle on τ. Suppose that C does cycle on τ. By Lemma 3, this can occur only if there is some non-end goal G' that is cycling. Now consider $G' = \mathsf{G}(x, s', f')$: it is either a means goal for C or a discharge goal for C. Observe from the practical rules that, since G does not cycle, G' cannot cycle. □

Corollary 1. *Let $\tau = \langle S_0, S_1, \ldots \rangle$ be a trace on which no end goal is cycling. Then no commitment cycles on τ.*

Proof. By Lemma 4, if no end goal cycles then no goal cycles. The result follows by Lemma 3. □

Theorem 3. *Let $\tau = \langle S_0, S_1, \ldots \rangle$ be a trace on which no end goal is cycling. Then for any state S_i on τ, if S_i is not coherent, there is a subsequent state S_j, $j > i$, on τ such that S_j is coherent.*

Proof. By Lemma 4, if no end goal cycles on τ, then no goal cycles on τ, and further by Corollary 1, no commitment cycles on τ. By Theorem 1, if no goal and no commitment cycles on τ, then the result follows. □

We reiterate that our convergence properties in the theorems hold only if each agent (1) applies the rules of Section 3.4, (2) acts towards the achievement of its commitments and goals according to the fairness assumption, and (3) eventually applies a rule that is applicable. (If an agent forever chooses to apply no practical rule, then convergence does not occur.) If agents depart from our semantics then the results do not hold.

The results we have proved establish that, granted the conditions above, the operational semantics does not lead to incoherence. We would like to prove a stronger result that agents will successfully achieve their end goals (using commitments and means goals as required). To do so requires additional assumptions about the multiagent system. Specifically, such a theorem rests on assumptions about the collective capability of the agents, the collective persistence of the agents (i.e., informally, not giving up on their goals), and the collective joint feasibility of all their goals. Akin to the argument in Chopra et al. (2014), the proof will recurse through the decomposition of goals. The key idea is that an agent may not have capability to achieve a goal (including a goal for either the antecedent or consequent of a commitment), but can follow the practical rules to have other agents to achieve it.

We remark that a similar 'success theorem' was proved for a single agent by Singh (1994), based on the assumptions of what Singh called know-how (i.e., capability), persistence, and conation (i.e., acting on an intention). Given the assumptions that must be formalized to state and prove such a theorem for the multiagent case, however, we leave it for a future article.

5. Related Work

In this section we survey work pertaining commitment or goal semantics, and indicate the novelty of our contribution. Rodríguez-Aguilar, Sierra, Arcos, López-Sánchez, and Rodríguez (2015) provide a survey of agent coordination infrastructures, noting that "next generation coordination infrastructures must address a number of challenges", including "decision support" to help agents' reason about their goals—in which respect we argue for the saliency of agents' commitments.

5.1 Formalizations of Commitments

The commitment life cycle has been formalized by researchers before us, including by Fornara and Colombetti (2002), Mallya, Yolum, and Singh (2003), and Marengo et al. (2011). The variant we adopt uses state diagrams that include nested states, which simplifies the representation. Further, the states and transitions we adopt accord better with our intuitions. However, our approach could be applied to any of the alternative formulations as well.

Our work does not treat inter-agent communication and messaging, but investigates the semantics of goals and commitments as a basis for cooperation. This semantics could underlie an account of reasoning about communications; indeed, commitments and agent communication have been well explored (Dastani, van der Torre, & Yorke-Smith, 2017; Fornara & Colombetti, 2002; Singh, 1998). Going beyond (only) communication, Dunin-Kęplicz and Verbrugge (2010) formalize the static and dynamic functions of social commitments in teamwork. A difference is that their commitments include an intentional component, whereas the commitments in our approach are purely social entities.

Yolum and Singh (2007) study enacting commitment-based protocols by means of commitment concession. Their work thus addresses the coherence of agents' protocol-enactment policies with the given commitment protocols. Desai, Narendra, and Singh (2008) study the problem of determining whether a given contract is safe and beneficial for an agent. An agent may employ such reasoning to determine whether to, for example, entice another agent or be receptive to enticement from another agent.

Chopra et al. (2010b) formalize the semantic relationship between agents and protocols encoded as goals and commitments, respectively, to verify at design time if a protocol specification (expressed using commitments) supports achieving the goals in an agent specification, and vice versa. In contrast, our semantics applies at runtime, and we propose practical rules of reasoning that agents may follow to achieve coherence between related goals and commitments. Dalpiaz, Chopra, Giorgini, and Mylopoulos (2010) propose a model of agent reasoning based on the pursuit of *variants*—abstract agent strategies for pursuing a goal. We conjecture that Dalpiaz et al.'s approach can be expressed as sets of practical rules, such as those we described above.

Marengo et al. (2011) provide a logical formalization of commitments that include temporal regulations within the antecedent and consequent. They enable an agent to reason about when a temporal commitment is safe for the agent to accept as a debtor. They do not consider goals directly. Marengo et al.'s work complements our work in that, conceivably, an agent may reason about the safety or otherwise of any commitments of which it is the debtor or creditor to decide how to proceed. In other words, an agent could use the more sophisticated reasoning provided by Marengo et al.'s approach to decide which practical rules to execute, for example, to terminate or persist with the goals associated with a commitment.

Chesani, Mello, Montali, and Torroni (2013) provide a formalization of commitments in a first-order event calculus. They define a semantics for commitment operations using event calculus, and use an extended logic programming framework with constraints to define the semantics of commitments. Building on their earlier work, Chesani et al.'s formalization of commitments admits runtime data values within commitments and metric temporal properties over commitments. They further develop monitoring tools and prove properties over the monitoring language.

Chopra and Singh (2009, 2015b) address the problem of ensuring alignment of commitments. Alignment means that whenever a creditor represents a commitment from a debtor, the debtor represents the same commitment (to the same creditor). In effect, the expectations that a creditor has are well-grounded. Of course, the debtor may violate the commitment: alignment does not guarantee success. Chopra and Singh (2009) formalize commitment alignment and show how to achieve it in the face of asynchronous communication by suitably requiring additional messages so that alignment is achieved. Chopra and Singh (2015b) weaken the assumptions further by abolishing the requirement for communication channels that preserve message order.

Alignment relates to an intrinsic philosophical issue of whether one agent's (view of a) commitment is the same as another agent's. Constructs such as an institution or a (central) commitment store have been proposed (Duplessis, Pauchet, Chaignaud, & Kotowicz, 2017; Fornara & Colombetti, 2009). Alignment is concerned with the interaction protocol aspects of the multiagent system architecture in that it shows how the states of agents' public commitments relate to one another in light of observations the agents make and communications they exchange. This article concerns more the internal aspects of the multiagent system architecture in that it relates the goals and commitments of individual agents. Through the agents' commitments, it thus relates the goals (ends and means) of different agents and lends coherence to the computations in a multiagent system.

Two lines of work must be highlighted as closely related to our contribution in this article. Both, in fact, build on our previous, preliminary paper (Telang et al., 2012) and on the set of practical rules formulated in it for commitments and goals.

First, Meneguzzi and colleagues (Meneguzzi et al., 2018; Meneguzzi, Telang, & Yorke-Smith, 2015; Telang, Meneguzzi, & Singh, 2013) explore the use of automated planning to generate commitment-based protocols that achieve the (individual) goals of a set of agents.

Second, Baldoni and colleagues (Baldoni et al., 2015) implement our earlier set of practical rules in JaCaMo+, an extension of JaCaMo in which Jason agents can reason about social relationships represented as commitments. The authors demonstrate how agents implemented in JaCaMo+ are programmed in a high-level manner, while the agent platform exploits our work on the relation between goals and commitments.

5.2 Belief-Desire-Intention Framework and Norms

The Belief-Desire-Intention (BDI) framework of Rao and Georgeff (1992) has been supplemented (or contrasted) with various cognitive and social notions, including for instance shared goals (Grosz & Kraus, 1996), and obligations or norms (Broersen, Dastani, Hulstijn, Huang, & van der Torre, 2001; F. Dignum, Kinny, & Sonenberg, 2002). Our work adopts some of the spirit of the BDI approach, but is not explicitly tied to it. We have used the terminology of beliefs and goals (which can be seen as correlating to Desires in Rao and Georgeff (1992)'s original terminology: see the discussions by, e.g., F. Dignum et al., 2002; Myers and Yorke-Smith, 2007; Braubach and Pokahr, 2009). We have not considered intentions, since planning for achievement of goals and execution of plans is outside our scope. Among many others, Winikoff et al. (2002) treat this aspect. Our setting incorporates social commitments in addition to elements of BDI-like cognitive state treated for instance by Broersen et al. (2001); Myers and Yorke-Smith (2005).

We consider social elements in agent reasoning in the form of social commitments, which carry normative or deontic force in terms of what one agent would bring about for another agent. Another tradition in the literature considers social elements in the form of norms. The two traditions have many points of correspondence (Andrighetto, Governatori, Noriega, & van der Torre, 2013), as we now elaborate.

Some formulations of norms include a notion of sanctioning as a subsidiary. In this line, Singh (2013) treats norms as directed conditional normative relationships with commitments as we have studied here as a proper subclass. That work considers norms as arising in an organizational context, which we have disregarded here. We conjecture that the approach of this article could be readily expanded to tackle (directed conditional) norms.

A number of works provide an operational basis for norm reasoning within the BDI (Meneguzzi, Rodrigues, Oren, Vasconcelos, & Luck, 2015) or other frameworks (Dybalova, Testerink, Dastani, & Logan, 2013; Kollingbaum & Norman, 2003). Like these efforts, our work provides an operational basis, but differs in that we specifically consider commitments. We give detail on three frameworks.

Lee, Padget, Logan, Dybalova, and Alechina (2014) develop N-Jason, an extension to Jason to provide agents with real-time norm compliance, where norms are defined as prohibitions or obligations. N-Jason is a BDI agent framework supporting norm-aware deliberation and run-time norm compliance. The N-Jason execution mechanism schedules intentions with awareness of deadlines, priorities, prohibitions, and obligations. Our work sits at a higher level in that we treat the semantics of goals and commitments, and do not consider scheduling, and our work is not tied to a specific BDI architecture (compare Figure 3).

On a related theme, van Riemsdijk, Dennis, Fisher, and Hindriks (2013) develop a generic execution mechanism that allows agents to adapt their behaviour at run-time according to norms. They use Linear Temporal Logic to define prohibitions and obligations and develop an operational semantics of the execution mechanism using executable temporal logic. In contrast to the above work, we study the interaction of commitments and goals, in fact addressing some of the future work identified by van Riemsdijk et al. (2013): "investigate the effect of complying with norms on goal achievement, and investigate how one can guarantee that such goals are achieved if the agent adapts its behavior to comply with norms (if the goals were achieved in the original agent semantics."

Avali and Huhns (2008) relate an agent's commitments to its beliefs, desires, and intentions using $\mathrm{BDI}_{\mathrm{CTL}^*}$, an extension of CTL* with modal operators for commitments, beliefs, desires, and intentions. El-Menshawy, Bentahar, Qu, and Dssouli (2011) define an extension of CTL with modalities for commitments and their satisfaction or violation, and use model checking to formally verify properties of contracts modelled using commitments. In contrast with these and subsequent similar approaches, we adopt a simpler and more tractable language, we consider commitments and goal life cycles, and propose practical rules that establish how an agent may reason about its goals and commitments systematically and with guarantees of arriving at a coherent state.

5.3 Goals

Among others, Dastani, van Riemsdijk, and Winikoff (2011); van Riemsdijk et al. (2008) and Harland et al. (2014, 2017) propose abstract architectures for goals, on which is based the simplified goal life cycle that we consider. These and other authors formalize the operationalization of goals. In contrast, our work formalizes the combined operational semantics of goals and commitments. A natural extension of our work would be to address the different goal types that van Riemsdijk et al. and Dastani et al. propose.

We have considered each goal to be private to an agent. Work that studies the coordination of agents via shared proattitudes—such as shared goals—include, for example, work by Grosz and Kraus (1996) and Lesser et al. (2004). A practical difference with our approach is that the shared proattitudes approaches violate the heterogeneity and autonomy of agents, if the agents are provided access to each other's goals. Also, if the agents have less than perfect trust in each other, the shared attitude collapses (Singh, 2012). Commitments provide a cleaner interface between agents, specifying precisely what an agent would do for another and thus what the second agent should rely upon from the first.

We have considered each (means) goal in isolation. Thangarajah and Padgham (2011) share a similar conceptualization of goals as us, and treat the interaction between the goals and intentions of a single agent; Thangarajah and Padgham do not consider multiple agents nor commitments.

Kakas, Mancarella, Sadri, Stathis, and Toni (2008) present a modular agent architecture centered on agent state that evolves according to transitions. They consider beliefs, goals, and plans, but do not consider commitments. We develop a coherent operational semantics for goals and commitments between a pair of agents.

The recent work of Cranefield, Winikoff, Dignum, and Dignum (2017) presents a computational mechanism for using values (which could be constructed as norms) to select between hierarchical plans in the context of a BDI-style agent. The semantics of Cranefield et al. is implicit as an extension of AgentSpeak. These authors do not address the interaction of goals and commitments, rather assuming that the top-level goal is the root of the goal-plan tree (compare Harland et al., 2014).

Winikoff (2007) develops a mapping from commitments to BDI-style plans. He modifies SAAPL, an agent programming language, to include commitments in an agent's belief-base and operational semantics update the commitments. Our operational semantics addresses goals (which are more abstract than plans) and commitments. It will be interesting to combine Winikoff's work with ours to develop a comprehensive semantics for commitments, goals, and plans.

Günay, Winikoff, and Yolum (2015) study dynamic protocol generation wherein agents generate commitments to other agents at runtime. The authors propose an algorithm that considers the goals and capabilities of the agent making the commitment, as well as the agent to whom it proposes the commitment, in order to make it more likely that the creditor agent will accept the protocol. Günay et al. (2012) require commitments to be explicitly accepted or rejected. Otherwise, their commitment and goal life cycles are similar to those we proposed previously (Telang et al., 2012) and here, in line with prior work. Günay et al. (2012)'s work can be seen as a precursor to ours, in that they study how to establish commitments, whereas we study the coherent management of commitments and goals, regardless of how they arise.

Günay et al. (2016) extend the Günay's work on commitment protocols, to develop a modelling language for agents with respect to their commitments, capturing also the agent's beliefs and goals. Günay et al. also present a probabilistic model checking approach to verify commitment protocols in terms of compliance and goal satisfaction. We do not develop a modelling language nor treat commitment protocols, but closely study the interaction between goals and commitments in a multiagent system in terms of an operational semantics. We do not rely on model checking but present analytical proofs of convergence properties of goals and commitments under our semantics.

The advantage of model checking a semantics is the automated validation of the formalization and the properties that follow from it. Although computer-aided proofs raise some philosophical questions, their use has become widely accepted. The advantage of analytical proofs, when as in this article they can be derived, is the mathematical certainty and human understandability they provide. Although we do not report a model checking approach here, we have used such an approach in our related work (Telang & Singh, 2012).

5.4 Other Related Work

Telang and Singh (2009) enhance Tropos, an agent-oriented software engineering methodology, with commitments. They describe a methodology that starts from a goal model and derives commitments. Our operational semantics complements Telang and Singh's methodology by providing a formal underpinning for how agents may enact their goals and commitments. Telang and Singh (2012) propose a commitment-based business metamodel, a set of modelling patterns, and an approach for formalizing the business models and verifying message sequence diagrams with respect to the models. Based on the same metamodel, Telang and Singh (2010) abstract patterns from RosettaNet, a leading industry standard for B2B integrations. Our combined operational semantics of commitments and goals can provide a basis for how a business model can be enacted and potentially support the derivation of suitable message sequence diagrams.

Norman and Reed (2010) develop a formal logic of responsibility and delegation, founded on how a description of direct agent responsibility can be formed, and from that a logic semantics of the dynamics of how responsibility can be acquired, transferred and discharged. Delegation can pertain to commitments, and including delegation of commitments is a topic for our future work.

Our research aims at studying open systems. d'Inverno, Luck, Noriega, Rodriguez-Aguilar, and Sierra (2012) present a language, viewed as a 'tower' of four languages, for modelling open systems (electronic institutions), and its operational semantics. However, we focus on the abstractions of goals and commitments, and the interplay between them. Commitments can provide a basis for two of the components of d'Inverno et al.'s approach, namely, role and social mechanisms and interaction mechanisms. Two other components, scenes and institutions, are beyond our scope here. However, they reflect two important differences with the present work. First, d'Inverno et al.'s scenes describe activities organized in a procedural (finite state automaton) representation whereas the present work seeks to be declarative. Second, d'Inverno et al.'s institutions act as entities that can override an agent's actions, meaning that agents are not autonomous, whereas in the present work agents are autonomous. A key feature of electronic institutions as d'Inverno et al. envision is that they provide a social basis for interactions among autonomous agents. We provide a basis for relating goals and commitments that can provide a foundation for how an agent's private goals relate to the agent's commitments, thereby helping flesh out a part of the electronic institution vision.

6. Summary and Future Work

Goals and commitments are sharply complementary. Goals, like other cognitive primitives, are central in modelling and implementing agents, and unsuitable for modelling communication between agents in open multiagent systems. On the contrary, commitments have no direct bearing on the internals of an agent but are crucial to modelling and enacting communication where they provide a high-level notion of compliance. Goals and commitments have the common attribute of being high-level abstractions that help relevant facets respectively of an agent and a multiagent system in nonoperational terms. Because of their importance to the above-mentioned facets, a combined study of goals and commitments is crucial to developing a comprehensive theoretical approach for multiagent systems.

6.1 Contributions Summarized

This article studies the complementary aspects of commitments and goals by establishing an operational semantics of the related life cycles of the two concepts. We have distinguished the purely semantic aspects of their life cycles from the pragmatic aspects of how a cooperative agent may reason, and demonstrated desirable properties such as the convergence of traces of the system. From the viewpoint of agent programming, we have provided a foundational set of rules that is complete in a technical sense; their sufficiency in practice will be found through use.

The importance of our work, even in its previous and preliminary state (Telang et al., 2012), has been recognized by subsequent developments that leverage our practical rules for automatic protocol generation (Meneguzzi et al., 2018, 2013; Meneguzzi, Telang, & Yorke-Smith, 2015) and implement our rules in a practical agent language (Baldoni et al., 2015).

The life cycle rule guides an agent designer in (1) properly encoding beliefs, goals, and commitments, and their interplay, and (2) bringing up the controls an agent may exercise on its goals and commitments through goal and commitment actions, respectively. In addition, the practical rules suggest possible goal and commitment actions that an agent may elect to take given the states of its beliefs, goals, and commitments—and thus capture what strategy it may apply in socially engaging with other agents.

Our contribution is thus not an approach for implementing agents but rather an approach for reasoning about implementations of agents. In particular, Baldoni et al. (2015) have successfully demonstrated how to implement agents based on the earlier version of our approach (Telang et al., 2012). The connection between our proposed approach and existing implementations arises through the life cycles for commitments and goals. We demonstrate our approach with respect to life cycles that we have adopted based on the literature. Variants could be developed for other life cycles. In the same vein, designers may consider different sets of practical rules than we have. For each such selection of life cycles and practical rules, as discussed in Section 2.3, our methodology can be applied to determine whether the rules are convergent and coherent.

6.2 Future Work

Our work carries importance because of its formalization of the intuitive complementarity between goals and commitments. Directions for building on this foundation include considering a hierarchy of prioritized goals or commitments, and extending our semantics to include delegation and assignment of commitments (Norman & Reed, 2010), maintenance goals, shared goals, or plans. We are also interested in examining strong convergence properties for a collaborative multiagent system. Additional cognitive states, such as desires or intentions would also extend the scope of this article.

Our approach assumes propositional goals and commitments. A future direction is to consider enhanced representations that involve decidable fragments of first-order logic. In a first-order representation, the success and failure conditions of goals and, the antecedent and consequent of commitments would be formulas in a first-order logic with parameters. The definition of linked goals would need to be enhanced to consider the parameters in these formulas. For example, to satisfy the commitment $\mathsf{C}(\textsc{buyer}, \textsc{seller}, \mathsf{goods(AI\text{-}Book)}, \mathsf{pay(\$500)})$, the BUYER may create two discharge goals: $\mathsf{G}(\textsc{buyer}, \mathsf{pay(\$200)})$ and $\mathsf{G}(\textsc{buyer}, \mathsf{pay(\$300)})$. In this case, instead of one, there are two discharge goals linked to one commitment. Our set of practical rules would need to be enhanced to consider such possibilities.

The following are two additional specific directions for future work.

6.2.1 Commitment Conflicts

Section 2.3 introduced how the semantics introduced in this article has potential to reduce certain conflicts between commitments, or commitments and goals, to conflicts between single-agent goals. Goal conflict is treated in the literature (Murukannaiah, Kalia, Telang, & Singh, 2015; Thangarajah & Padgham, 2011; Zatelli, Hübner, Ricci, & Bordini, 2016). Research is needed to formalize definitions of commitment conflict, goal conflict, commitment–goal conflict, and to theoretically capture what forms of (commitment) conflicts are mitigated by a system of agents following our semantics for commitment–goal alignment, and further to thoroughly study the topic of commitment conflicts in light of the literature on normative conflicts (Broersen et al., 2001; Castelfranchi, 2000; Chopra & Singh, 2015b; dos Santos, de Oliveira Zahn, Silvestre, da Silva, & Vasconcelos, 2017; Lee et al., 2014).

6.2.2 Verifying Business Models

Although our operational semantics is general to the commitments and goals of a multiagent system, to apply our approach in extensive settings involving multiple commitments and goals will however require additional research, specifically, in analyzing agents with richer decision-making rules, for example, as needed to choose between conflicting commitments or conflicting goals.

Our approach opens the concept of what we might describe as a structurally well-formed *business model*: a system of agents with goals, and commitments between the agents, such that a path exists that satisfies the goals. Systems without such a property correspond to business models that cannot succeed, and should therefore be revised. How such properties can be automatically checked, including in combination with automatic business model generation, is a research topic (Günay et al., 2015; Seqerloo, Amiri, Parsa, & Koupaee, 2019).

Acknowledgments

We thank the JAIR editors and the anonymous reviewers for their very helpful comments. Thanks also to R. Krebbers for valuable suggestions. We also gratefully acknowledge the discussions at the AAMAS 2011 ProMAS workshop where preliminary ideas were presented (Telang et al., 2012). PRT performed part of this work while at NC State University and Cisco Systems Inc. MPS was partially supported by the US Department of Defense under the Science of Security Lablet grant to NC State University. NYS was partially supported by grant number 102853 from the University Research Board, American University of Beirut. NYS thanks the Operations group at the Cambridge Judge Business School and the fellowship at St Edmund's College, Cambridge.

References

Aldewereld, H., & Dignum, V. (2010). OperettA: Organization-oriented development environment. In *Proceedings of 3rd International Workshop on Languages, Methodologies, and Development Tools for Multi-Agent Systems (LADS'10)* (Vol. Lecture Notes in Computer Science 6822, pp. 1–18). Springer.

Andrighetto, G., Governatori, G., Noriega, P., & van der Torre, L. W. N. (Eds.). (2013). *Normative multi-agent systems.* Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi: http://dx.doi.org/10.4230/DFU.Vol4.12111.191

Avali, V. R., & Huhns, M. N. (2008). Commitment-based multiagent decision making. In *Proceedings of 12th International Workshop on Cooperative Information Agents (CIA'08)* (Vol. Lecture Notes in Computer Science 5180, pp. 249–263). Springer.

Baldoni, M., Baroglio, C., Capuzzimati, F., & Micalizio, R. (2015). Programming with commitments and goals in JaCaMo+. In *Proceedings of 14th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'15)* (pp. 1705–1706).

Boella, G., Broersen, J., & van der Torre, L. (2008). Reasoning about constitutive norms, counts-as conditionals, institutions, deadlines and violations. In *Proceedings of 11th Pacific Rim International Conference on Multi-Agents (PRIMA'08)* (pp. 86–97).

Bratman, M. E. (1987). *Intention, plans and practical reason.* Cambridge, MA: Harvard University Press.

Braubach, L., & Pokahr, A. (2009). A property-based approach for characterizing goals. In *Proceedings of 8th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'09)* (pp. 1121–1122).

Broersen, J., Dastani, M., Hulstijn, J., Huang, Z., & van der Torre, L. W. N. (2001). The BOID architecture: Conflicts between beliefs, obligations, intentions and desires. In *Proceedings of 5th International Conference on Autonomous Agents (Agents'01)* (pp. 9–16).

Bulling, N., & Dastani, M. (2016). Norm-based mechanism design. *Artificial Intelligence*, *239*, 97–142.

Castelfranchi, C. (1995). Commitments: From individual intentions to groups and organizations. In *Proceedings of 1st International Conference on Multiagent Systems (ICMAS'95)* (pp. 41–48).

Castelfranchi, C. (2000). Conflict ontology. In H. J. Müller & R. Dieng (Eds.), *Computational conflicts, conflict modeling for distributed intelligent systems* (pp. 21–40). Springer.

Chesani, F., Mello, P., Montali, M., & Torroni, P. (2013). Representing and monitoring social commitments using the event calculus. *Autonomous Agents and Multi-Agent Systems*, *27*, 85–130.

Chopra, A. K., Dalpiaz, F., Aydemir, F. B., Giorgini, P., Mylopoulos, J., & Singh, M. P. (2014). Protos: Foundations for engineering innovative sociotechnical systems. In *Proceedings of 22nd IEEE International Requirements Engineering Conference (RE'14)* (pp. 53–62).

Chopra, A. K., Dalpiaz, F., Giorgini, P., & Mylopoulos, J. (2010a). Modeling and reasoning about service-oriented applications via goals and commitments. In *Proceedings of 22nd International Conference on Advanced Information Systems Engineering (CAiSE'10)* (pp. 417–421).

Chopra, A. K., Dalpiaz, F., Giorgini, P., & Mylopoulos, J. (2010b). Reasoning about agents and protocols via goals and commitments. In *Proceedings of 9th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'10)* (pp. 457–464).

Chopra, A. K., & Singh, M. P. (2009). Multiagent commitment alignment. In *Proceedings of 8th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'09)* (pp. 937–944).

Chopra, A. K., & Singh, M. P. (2015a). Cupid: Commitments in relational algebra. In *Proceedings of 29th AAAI Conference on Artificial Intelligence (AAAI)* (pp. 2052–

2059).

Chopra, A. K., & Singh, M. P. (2015b). Generalized commitment alignment. In *Proceedings of 14th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'15)* (pp. 453–461).

Cranefield, S., Winikoff, M., Dignum, V., & Dignum, F. (2017). No pizza for you: Value-based plan selection in BDI agents. In *Proceedings of 26th International Joint Conference on Artificial Intelligence (IJCAI'17)* (pp. 178–184).

Dalpiaz, F., Chopra, A. K., Giorgini, P., & Mylopoulos, J. (2010). Adaptation in open systems. In *Proceedings of 29th Conference on Conceptual Modeling (ER'10)* (pp. 31–45).

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

Dastani, M., van Riemsdijk, M. B., & Winikoff, M. (2011). Rich goal types in agent programming. In *Proceedings of 10th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'11)* (pp. 405–412).

Davidson, D. (2001). *Essays on actions and events*. Oxford, UK: Clarendon Press.

Desai, N., Chopra, A. K., & Singh, M. P. (2009). Amoeba: A methodology for modeling and evolution of cross-organizational business processes. *ACM Transactions on Software Engineering and Methodology*, *19*(2), 6:1–6:45.

Desai, N., Narendra, N. C., & Singh, M. P. (2008). Checking correctness of business contracts via commitments. In *Proceedings of 7th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'08)* (pp. 787–794).

Dignum, F., Kinny, D., & Sonenberg, E. (2002). From desires, obligations and norms to goals. *Cognitive Science Quarterly*, *2*(3–4), 407–430.

Dignum, V., Dignum, F., & Meyer, J. C. (2004). An agent-mediated approach to the support of knowledge sharing in organizations. *Knowledge Engineering Review*, *19*(2), 147–174.

d'Inverno, M., Luck, M., Noriega, P., Rodriguez-Aguilar, J., & Sierra, C. (2012). Communicating open systems. *Artificial Intelligence*, *186*, 38–94.

dos Santos, J. S., de Oliveira Zahn, J., Silvestre, E. A., da Silva, V. T., & Vasconcelos, W. W. (2017). Detection and resolution of normative conflicts in multi-agent systems: A literature survey. *Autonomous Agents and Multi-Agent Systems*, *31*(6), 1236–1282.

Dunin-Kęplicz, B., & Verbrugge, R. (2010). *Teamwork in multi-agent systems: A formal approach*. Chichester, UK: Wiley.

Duplessis, G. D., Pauchet, A., Chaignaud, N., & Kotowicz, J.-P. (2017). A conventional dialogue model based on dialogue patterns. *International Journal on Artificial Intelligence Tools*, *26*(01), 1760009.

Dybalova, D., Testerink, B., Dastani, M., & Logan, B. (2013). A framework for programming norm-aware multi-agent systems. In *Proceedings of COIN@AAMAS&PRIMA Workshops (COIN'13)* (Vol. Lecture Notes in Computer Science 8386, pp. 364–380). Springer.

El-Menshawy, M., Bentahar, J., Qu, H., & Dssouli, R. (2011). On the verification of social commitments and time. In *Proceedings of 10th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'11)* (pp. 483–490).

Fornara, N., & Colombetti, M. (2002). Operational specification of a commitment-based agent communication language. In *Proceedings of 1st International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'02)* (pp. 535–542).

Fornara, N., & Colombetti, M. (2009). Specifying and enforcing norms in artificial institutions. In *Proceedings of 6th International Workshop on Declarative Agent Languages and Technologies (DALT'09)* (Vol. Lecture Notes in Computer Science 5397, pp. 1–17). Springer.

Fornara, N., Viganò, F., Verdicchio, M., & Colombetti, M. (2008). Artificial institutions: A model of institutional reality for open multiagent systems. *Artificial Intelligence and Law*, *16*(1), 89–105.

Grosz, B., & Kraus, S. (1996). Collaborative plans for complex group action. *Artificial Intelligence*, *86*(2), 269–357.

Günay, A., Liu, Y., & Zhang, J. (2016). Promoca: Probabilistic modeling and analysis of agents in commitment protocols. *Journal of Artificial Intelligence Research*, *57*, 465–508.

Günay, A., Winikoff, M., & Yolum, P. (2012). Commitment protocol generation. In *Proceedings of 10th International Workshop on Declarative Agent Languages and Technologies (DALT'12)* (Vol. Lecture Notes in Computer Science 7784, pp. 67–82). Springer.

Günay, A., Winikoff, M., & Yolum, P. (2015). Dynamically generated commitment protocols in open systems. *Autonomous Agents and Multi-Agent Systems*, *29*(2), 192–229.

Harland, J., Morley, D., Thangarajah, J., & Yorke-Smith, N. (2014). An operational semantics for the goal life-cycle in BDI agents. *Autonomous Agents and Multi-Agent Systems*, *28*(4), 682–719.

Harland, J., Morley, D. N., Thangarajah, J., & Yorke-Smith, N. (2017). Aborting, suspending, and resuming goals and plans in BDI agents. *Autonomous Agents and Multi-Agent Systems*, *31*(2), 288–331.

Kakas, A., Mancarella, P., Sadri, F., Stathis, K., & Toni, F. (2008). Computational logic foundations of KGP agents. *Journal of Artificial Intelligence Research*, *33*, 285–348.

Kollingbaum, M. J., & Norman, T. J. (2003). NoA: A normative agent architecture. In *Proceedings of 18th International Joint Conference on Artificial Intelligence (IJCAI'03)* (pp. 1465–1466).

Lee, J., Padget, J., Logan, B., Dybalova, D., & Alechina, N. (2014). Run-time norm compliance in BDI agents. In *Proceedings of 13th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'14)* (pp. 1581–1582).

Lesser, V., Decker, K., Wagner, T., Carver, N., Garvey, A., Horling, B., ... Zhang, X. (2004). Evolution of the GPGP/TAEMS Domain-Independent Coordination Framework. *Autonomous Agents and Multi-Agent Systems*, *9*(1), 87–143.

Mallya, A. U., Yolum, P., & Singh, M. P. (2003). Resolving commitments among autonomous agents. In *Proceedings of International Workshop on Agent Communication Languages (ACL'03)* (Vol. Lecture Notes in Computer Science 2922, pp. 166–182). Springer.

Marengo, E., Baldoni, M., Baroglio, C., Chopra, A. K., Patti, V., & Singh, M. P. (2011). Commitments with regulations: Reasoning about safety and control in REGULA. In *Proceedings of 10th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'11)* (pp. 467–474).

Maudet, N., & Chaib-Draa, B. (2002). Commitment-based and dialogue-game-based protocols: New trends in agent communication languages. *Knowledge Engineering Review*, *17*, 157–179.

Meneguzzi, F., Magnaguagno, M. C., Singh, M. P., Telang, P. R., & Yorke-Smith, N. (2018). GoCo: Planning expressive commitment protocols. *Autonomous Agents and Multi-Agent Systems*, *32*(4), 459–502.

Meneguzzi, F., Rodrigues, O., Oren, N., Vasconcelos, W. W., & Luck, M. (2015). BDI reasoning with normative considerations. *Engineering Applications of AI*, *43*, 127–146.

Meneguzzi, F., Telang, P. R., & Singh, M. P. (2013). A first-order formalization of commitments and goals for planning. In *Proceedings of 27th AAAI Conference on Artificial Intelligence (AAAI'13)* (pp. 697–703).

Meneguzzi, F., Telang, P. R., & Yorke-Smith, N. (2015). Towards planning uncertain commitment protocols. In *Proceedings of 14th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'15)* (pp. 1681–1682).

Murukannaiah, P. K., Kalia, A. K., Telang, P. R., & Singh, M. P. (2015). Resolving goal conflicts via argumentation-based analysis of competing hypotheses. In *Proceedings of 23rd IEEE International Requirements Engineering Conference (RE'15)* (pp. 156–165).

Myers, K. L., & Yorke-Smith, N. (2005). A cognitive framework for delegation to an assistive user agent. In *Proceedings of AAAI 2005 Fall Symposium on Mixed-Initiative Problem-Solving Assistants* (pp. 94–99). Menlo Park, CA: AAAI Press.

Myers, K. L., & Yorke-Smith, N. (2007). Proactivity in an intentionally helpful personal assistive agent. In *Proceedings of the AAAI 2007 Spring Symposium on Intentions in Intelligent Systems* (pp. 34–37). Menlo Park, CA: AAAI Press.

Norman, T. J., & Reed, C. (2010). A logic of delegation. *Artificial Intelligence*, *174*(1), 51–71.

Rao, A. S., & Georgeff, M. P. (1992). An abstract architecture for rational agents. In *Proceedings of 3rd International Conference on Principles and Knowledge Representation and Reasoning (KR'92)* (pp. 439–449).

Rodríguez-Aguilar, J. A., Sierra, C., Arcos, J. L., López-Sánchez, M., & Rodríguez, I. (2015). Towards next generation coordination infrastructures. *Knowledge Engineering Review*, *30*(4), 435–453.

Sabater-Mir, J., Pinyol, I., Villatoro, D., & Cuni, G. (2007). Towards hybrid experiments on reputation mechanisms: BDI agents and humans in electronic institutions. In *Proceedings of 12th Conference of the Spanish Association for Artificial Intelligence (CAEPIA'07).*

Seqerloo, A. Y., Amiri, M. J., Parsa, S., & Koupaee, M. (2019). Automatic test cases generation from business process models. *Requirements Engineering*, *24*(1), 119–132.

Singh, M. P. (1991). Social and psychological commitments in multiagent systems. In *Proceedings of AAAI 1991 Fall Symposium on Knowledge and Action at Social and Organizational Levels* (pp. 104–106). Menlo Park, CA: AAAI Press.

Singh, M. P. (1994). *Multiagent systems: A theoretical framework for intentions, know-how, and communications* (Vol. Lecture Notes in Computer Science 799). Heidelberg: Springer. (Available at `www.csc.ncsu.edu/faculty/mpsingh/books/MAS/`)

Singh, M. P. (1998). Agent communication languages: Rethinking the principles. *IEEE*

Computer, *31*(12), 40–47.

Singh, M. P. (1999). An ontology for commitments in multiagent systems: Toward a unification of normative concepts. *Artificial Intelligence and Law*, *7*(1), 97–113.

Singh, M. P. (2008). Semantical considerations on dialectical and practical commitments. In *Proceedings of 23rd National Conference on Artificial Intelligence (AAAI'08)* (pp. 176–181).

Singh, M. P. (2012). Commitments in multiagent systems: Some history, some confusions, some controversies, some prospects. In F. Paglieri, L. Tummolini, R. Falcone, & M. Miceli (Eds.), *The goals of cognition: Essays in honor of Cristiano Castelfranchi* (pp. 591–616). London, UK: College Publications.

Singh, M. P. (2013). Norms as a basis for governing sociotechnical systems. *ACM Transactions on Intelligent Systems and Technology*, *5*(1), 21:1–21:23.

Singh, M. P., Chopra, A. K., & Desai, N. (2009). Commitment-based service-oriented architecture. *IEEE Computer*, *42*(11), 72–79.

Telang, P. R., Meneguzzi, F., & Singh, M. P. (2013). Hierarchical planning about goals and commitments. In *Proceedings of 12th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'13)* (pp. 877–884).

Telang, P. R., & Singh, M. P. (2009). Enhancing Tropos with commitments. In *Conceptual modeling: Foundations and applications* (Vol. Lecture Notes in Computer Science 5600, pp. 417–435). Springer.

Telang, P. R., & Singh, M. P. (2010). Abstracting and applying business modeling patterns from RosettaNet. In *Proceedings of 8th International Conference on Service Oriented Computing (ICSOC'10)* (pp. 426–440).

Telang, P. R., & Singh, M. P. (2012). Specifying and verifying cross-organizational business models: An agent-oriented approach. *IEEE Transactions on Services Computing*, *5*(3), 305–318.

Telang, P. R., Singh, M. P., & Yorke-Smith, N. (2012). Relating goal and commitment semantics. In *Proceedings of 9th International Workshop on Programming Multi-Agent Systems (ProMAS'12)* (Vol. Lecture Notes in Computer Science 7217, pp. 22–37). Springer.

Thangarajah, J., Harland, J., Morley, D., & Yorke-Smith, N. (2011). Operational behaviour for executing, suspending and aborting goals in BDI agent systems. In *Proceedings of 8th International Workshop on Declarative Agent Languages and Technologies (DALT'11)* (Vol. Lecture Notes in Computer Science 6618, pp. 1–21). Springer.

Thangarajah, J., & Padgham, L. (2011). Computationally effective reasoning about goal interactions. *Journal of Automated Reasoning*, *47*(1), 17–56.

van Riemsdijk, M. B., Dastani, M., & Winikoff, M. (2008). Goals in agent systems. In *Proceedings of 7th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'08)* (pp. 713–720).

van Riemsdijk, M. B., Dennis, L. A., Fisher, M., & Hindriks, K. V. (2013). Agent reasoning for norm compliance: a semantic approach. In *Proceedings of 12th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'13)* (pp. 499–506).

Verdicchio, M., & Colombetti, M. (2003). A logical model of social commitment for agent communication. In *Proceedings of 2nd International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'03)* (pp. 528–535).

Visser, S., Thangarajah, J., Harland, J., & Dignum, F. (2016). Preference-based reasoning in BDI agent systems. *Autonomous Agents and Multi-Agent Systems*, *30*(2), 291–330.

Winikoff, M. (2007). Implementing commitment-based interactions. In *Proceedings of 6th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'07)* (pp. 873–880).

Winikoff, M., Padgham, L., Harland, J., & Thangarajah, J. (2002). Declarative and procedural goals in intelligent agent systems. In *Proceedings of 8th International Conference on Principles and Knowledge Representation and Reasoning (KR'02)* (pp. 470–481).

Yolum, P., & Singh, M. P. (2007). Enacting protocols by commitment concession. In *Proceedings of 6th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'07)* (pp. 116–123).

Zatelli, M. R., Hübner, J. F., Ricci, A., & Bordini, R. H. (2016). Conflicting goals in agent-oriented programming. In *Proceedings of 6th International Workshop on Programming Based on Actors, Agents, and Decentralized Control (AGERE'16)* (pp. 21–30).