ICT Technical Report ICT-TR-02-2003

The Social Credit Assignment Problem (Extended Version)

Wenji Mao Jonathan Gratch

Abstract *Social credit assignment* is a process of social judgment whereby one singles out individuals to blame or credit for multi-agent activities. Such judgments are a key aspect of social intelligence and underlie social planning, social learning, natural language pragmatics and computational models of emotion. Based on psychological *attribution theory*, this report presents a preliminary computational approach to forming such judgments based on an agent's causal knowledge and conversation interactions.

1. Introduction

In contrast to how causality is used in the physical sciences, people instinctively seek out a human actor for their everyday judgments of credit or blame. Such attributions are fundamental *social* explanations involving judgments of not only causality, but also individual responsibility, free will and mitigating circumstances [Shaver, 1985]. These explanations underlie how we act on and make sense of the social world. They lead to emotional expressions of praise or rage. They justify public applause or prison terms. In short, they lie at the heart of social intelligence.

With the advance of multi-agent systems, user interfaces, and human-like agents, it is increasingly important to model and reason about this uniquely human-centric form of inference. This report lies out a preliminary computational approach to social credit and blame assignment based on psychological attribution theory. We see a number of immediate applications of this model. It can inform social explanations by augmenting traditional causal explanations with attributions of social judgment (e.g., explaining to a student which actor deserves credit or blame for outcomes in a multi-agent training simulation). It can inform social planning by augment traditional causal planners with the ability to reason about which actor has the ability to effect change. It can inform social learning, by distinguishing praiseworthy behavior from blameworthy one and reinforcing the praiseworthy. It can inform theories of natural language as much of human conversation centers around strategies for taking credit or deflecting blame. Finally, it is key for understanding human emotion, as social emotions such as pride, anger or guilt turn on the assessment of praiseworthiness and blameworthiness [Ortony *et al*., 1988; Gratch, 2000].

To be concrete, consider an example from a leadership trainer we are developing [Rickel *et al*., 2002]. The trainee is in command of an infantry platoon, *eagle 2-6*, in peacekeeping operations. His mission is to reinforce another unit, *eagle 1-6*. In route, one of his vehicles seriously injures a civilian and he must balance whether to continue the mission or render aid. Many decisions and outcomes are possible. In our example, the trainee decided to split his forces, ordering his sergeant to send half of his squads to help eagle 1-6. His sergeant responds that this is a bad idea; it will allocate too few forces to either goal, and instead, one squad should be sent ahead to scout the route. The trainee overrules this recommendation. In the end, the trainee finds he has insufficient resources to render aid to the injured civilian in a timely manner. The central question addressed here is to assess who, if anyone deserves blame for this unfortunate outcome, to what extent to blame the responsible party, and how to avoid naïve attributions, such as blaming the squad leaders that actually execute the orders.

Individuals may differ in whom they praise or blame in a specific situation, but psychologists and philosophers agree on the broad features underlying such judgments. Did someone *cause* the outcome? Did she *intend* the act? Did she *know* the consequences? Did she have *choice* or was she *coerced* by another agent? In the example, we may infer from the conversation that there were alternatives and the trainee coerced the sergeant to follow an undesirable course of actions. We can further surmise that the trainee was informed the bad consequence. Baring unknown mitigating factors (e.g. the sergeant al-

ways gave bad advice in the past), we would likely conclude that the trainee is to blame for the delay. This example shows that proper assignment of credit or blame in a social setting must not only consider actions (both physical acts and speech acts) and knowledge state of different actors, but also need to utilize information available to reason about key attributions that contribute to the judgment process.

2. Attribution Theory for Social Judgment

The question of how people assign social credit or blame has been studied extensively in philosophy, law, and social psychology. Traditions differ to the extent that their models are proscriptive (i.e., what are the "ideal" principles of responsibility and "ideal" mechanism for reasoning, for example, legal code or philosophical principles) or normative (i.e., what people actually do in their judgments). As our primary goal is to inform the design of realistic virtual humans that mimic human communicative and social behavior [Gratch *et al*., 2002], our focus is on descriptive rather than proscriptive models and we are particularly influenced by the work of Shaver [1985] and Weiner [1995] as these models are described at a level that is readily adapted to artificial intelligence knowledge representation and reasoning methods.

In Shaver's model, the assignment of credit or blame is a multi-step process initiated by events with positive or negative consequences (see *Figure 1*). First one assesses *causality*, distinguishing between personal versus impersonal causality (i.e., is causal agent a person or a force of nature). If personal, the judgment proceeds by assessing key factors: was it the actor's *intention* to produce the outcome; did the actor *foresee* its occurrence; was the actor forced under *coercion* (e.g., was the actor acting under orders)? As the last step of the process, proper degree of credit or blame is assigned to the responsible agent.

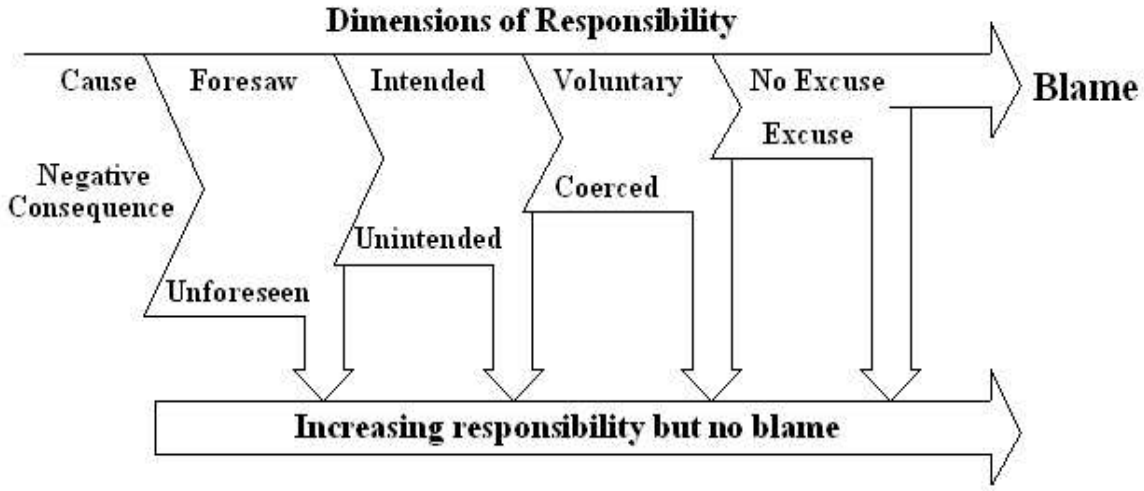

Figure 1. The Process model of blame assignment (adapted and simplified from Shaver [1985])

Weiner [1995] uses terms *locus of causality*, *Controllability* and *intentionality* instead. If the cause of an event is controllable by the actor (as locus of causality), then in the absence of mitigation factors, the actor is regarded as responsible for the outcome. Otherwise, if the cause is controllable by others (as locus of causality), then other agent is regarded as re-

sponsible. Comparing with intention, controllability is more fundamental as one cannot be held responsible for uncontrollable cause but can be held responsible for unintended action and unintended consequence of the action. In the latter case, however, one is not judged as intensely as when the action and the outcome are intended [Weiner, 2001].

These models reduce the problem of assigning credit or blame to the problem of determining whether certain outcome was caused, intended, forseen and/or coerced. Causality and intention map to standard concepts in agent-based systems, particularly frameworks that explicitly represent beliefs, desires and intentions [Bratman, 1987; Grosz and Kraus, 1996]. Coercion requires representation of social relationships and understanding of the extent to which it limits one's range of options. For example, one may be ordered to carry out a task but to satisfy the order, there may be alternatives that vary in blame or creditworthiness.

In modeling realistic human behavior, we cannot assume that a perceiving agent has privileged access to the mental states of other agents (e.g., intention is private to an agent), so deriving attribution variables can be nontrivial. In human social interactions, such variables are gleaned from a variety of sources: from observation of behavior, from statements made through natural language, from knowledge and models built up through past interactions, stereotypes and cultural norms. We show how to infer such information by analyzing natural language and causal evidence, making use of agents' knowledge of actions and consequences as well as commonsense intuition.

This report is organized as follows. In section 3, we construct the computational representation for the work, including representing plan features and attribution variables. Based on the representation, in section 4, we present the commonsense inference from communicative events and plans. Then in section 5, we give the back-tracing algorithm. To illustrate the approach in this report, an example from MRE virtual training scenario is demonstrated in section 6. Finally, in section 7, we summarize the work and raise some future work.

3. From Theory to Computational Approach

To inform social judgments, we need to represent knowledge states of agents and core conceptual variables underlying attribution theory. We also need to discuss how the representational primitives are applied in the attribution process.

3.1 Plan Representation

An *action* consists of a set of propositional preconditions, effects and steps. Each action step is either a *primitive* action (i.e., an action that can be directly executed by an agent) or an *abstract* action. An abstract action may be decomposed hierarchically in multiple ways and each alternative consists of a sequence of primitive or abstract sub-actions. The likelihood of preconditions and effects is represented by utility values, and the desirability of action effects (i.e., effects having positive/negative significance to an agent) is represented by utility values [Blythe, 1999].

A *non-decision node* (or *And node*) is an abstract action that can only be decomposed in one way. A *decision node* (or *Or node*), on the other hand, can be decomposed in more than one way. In a decision node, an agent needs to make a decision and select among different options. If a decision node *A* can be decomposed in different ways $a_1, a_2, \ldots, a_n$, $a_1, a_2, \ldots, a_n$ are choices of action A, and $a_1, a_2, \ldots, a_n$ are *alternatives* each other. Clearly, a primitive action is a non-decision node, while an abstract action can be either a non-decision node or a decision node.

Consequences or outcomes (we use the terms as exchangeable in this report) of actions are represented as a set of primitive action effects. The *consequence set* of an action *A* is defined recursively from leaf nodes (i.e., primitive actions) in plan structure to an action *A* as follows. Consequences of a primitive action are those effects with non-zero utility. For an abstract action, if the abstract action is a non-decision node, then the consequence set of the abstract action is the aggregation of the consequences of its sub-actions. If the abstract action is a decision node, we need to differentiate two kinds of consequences. If a consequence *p* of a decision node occurs among all its choices, we call *p* a *common consequence* of the decision node; otherwise *p* is a *non-common consequence* of the node. The consequence set of a decision node is defined as the set of its common consequences.

In addition, each action step is associated with a *performer* (i.e., the agent that performs the action) and an agent who has *authority* over its execution. The performer cannot execute the action until authorization is given by the authority. This represents the hierarchical organizational structure of social agents.

3.2 Attribution Variables

Now we discuss the *key* conceptual *variables* underlying attribution theory.

Causality refers to the connection between actions and the effects they produce. In our approach, causal knowledge is encoded via *plan representation*. Interdependencies between actions are represented as a set of causal links and threat relations. Each causal link specifies that an effect of an action achieves a particular goal that is a precondition of another action. Threat relations specify that an effect of an action threatens a causal link by making the goal unachievable before it is needed.

Foreseeability refers to an agent's foreknowledge about actions and consequences. We use *know* and *bring-about* to represent foreseeability. If an agent knows an action brings about certain consequence before its execution, then the agent foresees the action brings about the consequence.

Intention is generally conceived as a commitment to work towards certain act or outcome. Intending an act (i.e., *act intention*) is distinguished from intending an outcome of an act (i.e., *outcome intention*) in that the former concerns actions while the latter concerns consequences of actions. Most theories argue that outcome intention rather than act intention is the key factor in determining accountability and intended outcome usually

deserves more elevated accountability judgments [Weiner, 1986, 2001]. We use *intend* and *do* to represent act intention and *intend* and *achieve* for outcome intention. Since our work is applied to rich social context, comparing with [Bratman, 1987; Grosz and Kraus, 1996], we include indirect intentions in our work. For example, an agent intends an action or a consequence, but may not be the actor himself/herself (i.e., by intending another agent to act or achieve the consequence), or an agent intends to act but is coerced to do so.

Similar difference exists in *coercion*. An agent may be coerced to act (i.e., *act coercion*) yet not be coerced to achieve any outcome of the action (i.e., *outcome coercion*), depending on whether the agent has choices in achieving different outcomes among alternatives. It is important to differentiate act coercion and outcome coercion, because it is the latter that actually influence our judgment of behavior, and is used to determine the *responsible agent*. We use *coerced* and *do* to represent act coercion and *coerced* and *achieve* for outcome coercion. In the case of outcome coercion, the responsible agent for a specific outcome is the performer or the authority of an action, but the action may not be the primitive one that directly leads to the outcome.

3.3 Primitives

In order to model the attribution theory, we need to map attribution variables into representational features of an agent's causal interpretation. Here we define a number of specific primitive features that support this mapping.

x and y are different agents. A and B are actions and p is a proposition. The following primitives are adopted in system.

(1) *and-node*(A): A is a non-decision node in plan structure.
(2) *or-node*(A): A is a decision node in plan structure.
(3) *alternative*(A, B): A is an alternative way of acting B.
(4) *effect*(A): Effect set of a primitive action A.
(5) *consq*(A): Consequence set of A.
(6) *common-consq*(A): Common consequence set of A.
(7) *noncom-consq*(A): Non-common consequence set of A.
(8) *performer*(A): performing agent of A.
(9) *authority*(A): authorizing agent of A.
(10) *know*(x, p): x knows p.
(11) *intend*(x, p): x intends p.
(12) *coerced*(x, p, y): x is coerced p by y.
(13) *want*(x, p): x wants p.
(14) *bring-about*(A, p): A brings about p.
(15) *do*(x, A): x does A.
(16) *achieve*(x, p): x achieves p.
(17) *responsible*(p): Responsible agent for p.
(18) *superior*(x, y): x is superior of y.

3.4 Axioms

We identify the interrelations of attribution variables, expressed as *axioms*. The axioms are used either explicitly as *commonsense* inference rules for deriving key attribution values, or implicitly to keep the consistency between different inference rules.

x and y are different agents. A is an action and p is a proposition. The following *axioms* hold from a rational agent's perspective (To simplify the logical expressions, we omit the universal quantifiers in this report, and substitute A for do($*$, A) and p for achieve($*$, p) here).

(1) $\exists y(\text{coerced}(x, A, y)) \Rightarrow \text{intend}(x, A)$
(2) $\text{intend}(x, A) \wedge \neg(\exists y)\text{coerced}(x, A, y) \Rightarrow \exists p(p \in \text{consq}(A) \wedge \text{intend}(x, p))$
(3) $\text{intend}(x, p) \Rightarrow \exists A(p \in \text{consq}(A) \wedge \text{intend}(x, A))$
(4) $\text{intend}(x, A) \wedge p \in \text{consq}(A) \wedge \text{intend}(x, p) \Rightarrow \text{know}(x, \text{bring-about}(A, p))$

The *first* axiom shows that act coercion entails act intention. It means if an agent is coerced an action A by another agent, then the coerced agent intends A. The second and the third axioms show the relations between act intention and outcome intention. The *second* one means if an agent intends an action A and the agent is not coerced to do so (i.e. A is a voluntary act), then the same agent must intend at least one consequence of A. The *third* means if an agent intends a consequence p, the same agent must intend at least one action that has p as a consequence. Note that in both axioms, intending an action or a consequence includes the case that an agent intends another agent to act or achieve the consequence. The *last* one shows the relation between intention and foreseeability. It means if an agent intends an action A to achieve a consequence p of A, the same agent must know that A brings about p.

3.5 Attribution Process and Rules

Social credit assignment focuses on consequences with personal significance to an agent. This evaluation is always from the perspective of a perceiving agent and based on the attribution values acquired by the individual perceiver. As different perceivers have different preferences, different observations, and different knowledge and beliefs, it may well be the case that for the same situation, different perceivers form different judgments.

Nevertheless, the attribution process and rules are general, and applied uniformly to different perceivers. Following Weiner [2001], we use *coercion* to determine the responsible agent for credit or blameworthiness, and *intention* and *foreseeability* in assigning the intensity of credit/blame.

If an action performed by an agent brings about *positive/negative* consequence, and the agent is *not coerced* to achieve the consequence, then *credit/blame* is assigned to the *performer* of the action. Otherwise, assign credit/blame to the *authority*. If the authority is also coerced, the process needs to be traced further to find the *responsible agent* for the

consequence. The *back-tracing algorithm* for finding the responsible agent will be given later.

Rule 1: If <*consequence*> of <*action*> is *positive*/*negative* and
<*performer*> is *not coerced* the <*consequence*>
Then Assign *credit*/*blame* to the <*performer*>
Rule 2: If <*consequence*> of <*action*> is *positive*/*negative* and
<*performer*> is *coerced* the <*consequence*>
Then Assign *credit*/*blame* to the <*responsible* agent>

We adopt a simple categorical model of intensity assignment, though one could readily extend the model to a numeric value by incorporating probabilistic rules of inference. If the responsible agent intends the consequence while acting, the intensity assigned is *high*. If the responsible agent does not foresee the consequence, the intensity is *low*.

4. Commonsense Inference

Judgments of causality, foreseeability, intentionality and coercion are informed by dialogue and causal evidence. Some theories have formally addressed subsets of this judgment task. For example, [Sadek, 1990] addresses the relationship between dialogue and inferences of belief and intention. These theories have not tended to consider coercion. Rather than trying to synthesize and extend such theories, we introduce small number of commonsense rules that, via a justification-based truth maintenance system (*JTMS*), allow agents to make inferences based on this evidence.

4.1 Dialogue Inference

Conversational dialogue between agents is a rich source of information for deriving values of attribution variables. In a conversational dialogue, a *speaker* and a *hearer* take turns alternatively. When a *speech act* [Austin, 1962; Searle, 1969, 1979] is performed, a perceiving agent (which can be one of the participating agents or another agent) makes inferences based on observed conversation and current beliefs. As the conversation proceeds, beliefs are formed and updated accordingly.

Assume conversations between agents are *grounded* [Traum, 1994] and they conform to Grice's maxims of *Quality*[1] and *Relevance*[2] [Grice, 1975]. Background information (agents' social roles, relationship, etc) is also important, for example, an order can be successfully issued only to a subordinate, but a request can be made of any agent.

x and y are different agents. p and q are propositions and t is time. For our purpose, we analyze following speech acts that help infer agents' desires, intentions, foreknowledge and choices in acting.

(1) *inform*(x, y, p, t): x informs y that p at t.

[1] The Quality maxim states that one ought to provide true information in conversation.
[2] The Relevance maxim states that one's contribution to conversation ought to be pertinent in context.

(2) *request*(x, y, p, t): x requests y that p at t.
(3) *order*(x, y, p, t): x orders y that p at t.
(4) *accept*(x, p, t): x accepts p at t.
(5) *reject*(x, p, t): x rejects p at t.
(6) *counter-propose*(x, p, q, t): x counters p and proposes q at t.

We have designed commonsense rules that allow perceiving agents to infer from dialogue patterns. These rules are general. Hence, they can be combined flexibly and applied to variable-length dialogue sequences with multiple participants.

Let z be a perceiving agent. If at time $t1$, a speaker (s) *inform*s a hearer (h) that p, then after $t1$, a perceiving agent can infer that both the speaker and the hearer know that p as long as there is no intervening contradictory belief.

Rule 3: $\text{inform}(s, h, p, t1) \wedge t1<t3 \wedge \neg(\exists t2)(t1<t2<t3 \wedge \text{believe}(z, \neg\text{know}(s, p) \vee \neg\text{know}(h, p), t2)) \Rightarrow \text{believe}(z, \text{know}(s, p) \wedge \text{know}(h, p), t3)$

A *request* gives evidence of the speaker's *desire* (or *want*). An order gives evidence of the speaker's *intend*.

Rule 4: $\text{request}(s, p, t1) \wedge t1<t3 \wedge \neg(\exists t2)(t1<t2<t3 \wedge \text{believe}(z, \neg\text{want}(s, p, t2)) \Rightarrow \text{believe}(z, \text{want}(p), t3)$

Rule 5: $\text{order}(s, p, t1) \wedge t1<t3 \wedge \neg(\exists t2)(t1<t2<t3 \wedge \text{believe}(z, \neg\text{intend}(s, p), t2)) \Rightarrow \text{believe}(z, \text{intend}(s, p), t3)$

The hearer may *accept, reject* or *counter-propose*. If the speaker wants (or intends) and the hearer *accept*s, it can be inferred that the hearer intends. An agent can accept via speech or action execution. If the hearer accepts what the superior wants (or intends), there is evidence of coercion.

Rule 6: $\text{believe}(z, \text{want/intend}(s, p, t1) \wedge \text{accept}(h, p, t2) \wedge \neg\text{superior}(s, h) \wedge t1<t2<t4 \wedge \neg(\exists t3)(t2<t3<t4 \wedge \text{believe}(z, \neg\text{intend}(h, p), t3)) \Rightarrow \text{believe}(z, \text{intend}(h, p), t4)$

Rule 7: $\text{believe}(z, \text{want/intend}(s, p, t1) \wedge \text{accept}(h, p, t2) \wedge \text{superior}(s, h) \wedge t1<t2<t4 \wedge \neg(\exists t3)(t2<t3<t4 \wedge \text{believe}(z, \neg\text{coerced}(h, p, s), t3)) \Rightarrow \text{believe}(z, \text{coerced}(h, p, s), t4)$

In the rules above, if act coercion is true, act intention can be deduced from *Axiom 1*.

If the speaker wants (or intends) and the hearer *reject*s, infer that the hearer does not intend.

Rule 8: $\text{believe}(z, \text{want/intend}(s, p, t1) \wedge \text{reject}(h, p, t2) \wedge t1<t2<t4 \wedge \neg(\exists t3)(t2<t3<t4 \wedge \text{believe}(z, \text{intend}(h, p), t3)) \Rightarrow \text{believe}(z, \neg\text{intend}(h, p), t4)$

If the hearer *counters* acting A and *proposes* acting B instead, both the speaker and the hearer are believed to know that A and B are alternatives. It is also believed that the hearer does not want A and wants B instead.

Rule 9: counter-propose(h, do(h, A), do(h, B), *t1*) $\wedge$ *t1*<*t3* $\wedge$ $\neg(\exists t2)(t1<t2<t3 \wedge$ believe(z, $\neg$know(h, alternative(A, B))$\vee\neg$know(s, alternative(A, B)), *t2*)) $\Rightarrow$ believe(z, know(h, alternative(A, B))$\wedge$know(s, alternative(A, B)), *t3*)

Rule 10: counter-propose(h, p, q, *t1*) $\wedge$ *t1*<*t3* $\wedge$ $\neg(\exists t2)(t1<t2<t3 \wedge$ (believe(z, want(h, p)$\vee\neg$want(h, q), *t2*))) $\Rightarrow$ believe(z, $\neg$want(h, p)$\wedge$want(h, q), *t3*)

If the speaker has *known* that two actions are *alternatives* and still *requests* (or *orders*) one of them, infer that the speaker wants (or intends) the chosen action instead of the alternative. The beliefs that the speaker wants (or intends) the chosen action can be deduced from *Rules 4&5*.

Rule 11: believe(z, know(s, alternative(A, B)), *t1*) $\wedge$ request/order(s, do(h, A), *t2*) $\wedge$ *t1*<*t2*<*t4* $\wedge$ $\neg(\exists t3)(t2<t3<t4 \wedge$ believe(z, want(s, do(h, B)), *t3*)) $\Rightarrow$ believe(z, $\neg$want/intend(s, do(h, B)), *t4*)

4.2 Causal Inference

Causal knowledge encoded in plan representation also helps derive values of attribution variables. Different agent may have access to different plans in memory. While plans are specific to certain domain, the structure and features of plans can be described using domain-independent terms such as action types, alternatives and action effects. We adopt the hierarchical task formalism that differentiates action types, explicitly expresses consequences of alternatives, and separates common consequences of an action from its non-common ones.

An agent's *foreknowledge* can be derived simply by checking primitive action effects. If a consequence p is an effect of a primitive action A, then the agents involved (i.e., the performer and the authority) should know that A brings about p.

Rule 12: p$\in$effect(A) $\Rightarrow$ believe(z, know(performer(A), bring-about(A, p)))
p$\in$effect(A) $\Rightarrow$ believe(z, know(authority(A), bring-about(A, p)))

Outcome intent can be partially inferred from evidence of act intent and comparative features of consequence sets of action alternatives. According to *Axiom 2*, if an agent intends a voluntary action A, the agent must intend at least one consequence of A. If A has only one consequence p, then the agent is believed to intend p. In more general cases, when an action has multiple consequences, in order to identify whether a specific outcome is intended or not, a perceiver may examine *alternatives* the agent intends and does not intend, and compare the consequences of intended and unintended alternatives.

If an agent intends an action A voluntarily and does intend its alternative B, we can infer that the agent either intends (at least) one consequence that only occurs in A or does not

intend (at least) one consequence that only occurs in B, or both. If the consequence set of A is a subset of that of B, the rule can be simplified. As there is no consequence of A not occurring in the consequence set of B, we can infer that the agent does not intend (at least) one consequence that only occurs in B. In particular, if there is only one consequence p of B that does not occur in the consequence set of A, infer that the agent does not intend p.

Rule 13: believe(z, intend(x, A)$\wedge\neg$intend(x, B)$\wedge\neg$coerced(x, A, superior(x))) $\wedge$ alternative(A, B) $\wedge$ consq(A)$\subset$consq(B) $\Rightarrow$ $\exists p$($p\notin$consq(A) $\wedge$ $p\in$consq(B) $\wedge$ believe(z, $\neg$intend(x, p)))

On the other hand, given the same context that an agent intends an action A and does not intend its alternative B, if the consequence set of B is a subset of that of A, infer that the agent intends (at least) one consequence that only occurs in A. In particular, if there is only one consequence p of A that does not occur in the consequence set of B, the agent must intend p.

Rule 14: believe(z, intend(x, A)$\wedge\neg$intend(x, B)$\wedge\neg$coerced(x, A, superior(x))) $\wedge$ alternative(A, B) $\wedge$ consq(B)$\subset$consq(A) $\Rightarrow$ $\exists p$($p\in$consq(A) $\wedge$ $p\notin$consq(B) $\wedge$ believe(z, intend(x, p)))

Outcome coercion can be properly inferred from evidence of act coercion and consequence sets of different action types. In a non-decision node (i.e., *and-node*), if an agent is coerced to act, the agent is also coerced to achieve the consequences of subsequent actions, for the agent has no other choice.

Rule 15: believe(z, coerced(x, A, superior(x))) $\wedge$ and-node(A) $\wedge$ $p\in$consq(A) $\Rightarrow$ believe(z, coerced(x, p, superior(x)))

In a decision node (i.e., *or-node*), however, an agent must make decision amongst multiple choices. Even if an agent is coerced to act, it does not follow that the agent is coerced to achieve a specific consequence of subsequent actions. In order to infer outcome coercion, we examine the choices at a decision node. If an outcome is a common consequence of every alternative, then it is unavoidable and thus outcome coercion is true. Otherwise, if an outcome is a non-common consequence of the alternatives, then the agent has option to choose an alternative to avoid this outcome and thus outcome coercion is false. Our definition of consequence set ensures the consistency when the rules are applied to actions at different levels of plan structure.

Rule 16: believe(z, coerced(x, A, superior(x))) $\wedge$ or-node(A) $\wedge$ $p\in$common-consq(A) $\Rightarrow$ believe(z, coerced(x, p, superior(x)))
believe(z, coerced(x, A, superior(x))) $\wedge$ or-node(A) $\wedge$ $p\in$noncom-consq(A) $\Rightarrow$ believe(z, coerced(x, p, superior(x)))

5. Back-Tracing Algorithm

Judgments of attributions are made after the fact (i.e. actions have been executed and the consequence has occurred). We have developed a *back-tracing algorithm* for evaluating the responsible agent for a specific consequence. The evaluation process starts from the primitive action that directly causes a consequence with positive or negative utility. Since coercion may occur in more than one level in hierarchical plan structure, the process must trace from the primitive action to the higher-level actions. We use a back-tracing algorithm to find the responsible agent. The algorithm takes as input some desirable or undesirable consequence of a primitive action (*step 1*) and works up the task hierarchy. During each pass through the main loop (*step 2*), the algorithm initially assigns default values to the variables (*step 2.2*). Then apply dialog rules to infer variable values at the current level (*step 2.3*). If there is evidence that the performer was coerced to act (*step 2.4*), the algorithm proceeds by applying plan inference rules (*step 2.5*). If there is outcome coercion (*step 2.6*), the authority is deemed responsible (*step 2.7*). If current action is not the root node in plan structure and outcome coercion is true, the algorithm enters next loop and evaluates the next level up in the task hierarchy.

Algorithm (*consequence*, *plan structure*):
1. *parent* = *A*, where *consequence* is an effect of action *A*
2. DO
   - 2.1 *node* = *parent*
   - 2.2 coerced(performer(*node*), *node*, authority(*node*)) = *unknown*
     coerced(performer(*node*), *consequence*, authority(*node*)) = *unknown*
     responsible(*consequence*) = performer(*node*)
   - 2.3 Search dialog history on *node* and apply *dialog inference rules*
   - 2.4 IF coerced(performer(*node*), *node*, authority(*node*)) THEN
   - 2.5 apply *plan inference rules* on *node*
   - 2.6 IF coerced(performer(*node*), *consequence,* authority(*node*)) THEN
   - 2.7 responsible(*consequence*) = authority(*node*)
   - 2.8 *parent* = *P*, where *P* is the *parent* of *node* in *plan structure*

   WHILE *parent* ≠ *root* of *plan structure* AND
   coerced(performer(*node*), *consequence*, authority(*node*)) is *true*
3. RETURN responsible(*consequence*)

After the execution of the algorithm, the responsible agent for the outcome is determined. Meanwhile, through applying inference rules, the algorithm also acquires values of intention and foreknowledge about the agents. The variable values are then used by the attribution rules (*Rules 1&2*) to assign credit or blame to the responsible agent with proper intensity.

Events may lead to more than one desirable/undesirable consequence. For evaluating multiple consequences, we can apply the algorithm the same way, focusing on one consequence each time during its execution. Then, to form an overall judgment, the results can be aggregated and grouped by the responsible agents.

6. Illustrative Example

We are developing this work in the context of the Mission Rehearsal Exercise (MRE) leadership trainer [Rickel *et al*., 2002]. We focus on three social actors, the *student*, the *sergeant* and the *squad leader*, who work as a team in task performance. The student is a human trainee and acts as an authority over the sergeant. The squad leader acts as a subordinate of the sergeant. Conversations between agents are represented via speech acts and a dialogue history as in the *MRE*.

Take the sergeant's perspective as an example. The sergeant perceives the conversation between the actors and task execution (see dialogue segment below). The example is extracted from an actual run of the system. Details on how this negotiation is automatically generated and how natural language is mapped into speech acts can be found in [Traum *et al*., 2003].

Student: Sergeant. Send two squads forward.
Sergeant: That is a bad idea, sir. We shouldn't split our forces. Instead we should send one squad to recon forward.
Student: Send two squads forward.
Sergeant: Against my recommendation, sir. Lopez! Send first and fourth squads to Eagle 1-6's location.
Lopez: Yes, sir. Squads! Mount up!

Dialogue history includes the following acts, ordered by the time the speakers addressed them (*std*, *sgt* and *sld* stand for the student, the sergeant and the squad leader, respectively. *t1<t2<…<t6*).

(1) order(*std*, do(*sgt*, *two-sqds-fwd*), *t1*)
(2) inform(*sgt*, *std*, bring-about(*two-sqds-fwd*, *unit-fractured*), *t2*)
(3) counter-propose(*sgt*,, do(*sgt*, *two-sqds-fwd*), do(*sgt*, *one-sqd-fwd*), *t3*)
(4) order(*std*, do(*sgt*, *two-sqds-fwd*), *t4*)
(5) accept(*sgt*, do(*sgt*, *two-sqds-fwd*), *t5*)
(6) order(*sgt*, do(*sld*, *1st-and-4th-to-celic*), *t6*)
... ...

To simplify the example, we illustrate part of the task structure from *MRE* scenario and evaluate one of the *negative* consequences, though we can generally apply the approach here to more complex judgments. The sergeant has access to a partial plan (see *Figure 2*), where *one squad forward* and *two squads forward* are two choices of action *support eagle-1-6*. *One squad forward* is composed of two primitive actions, *4th squad* (*recon*) *forward* and *remaining* (*squads*) *forward*. *Two squads forward* consists of *1st and 4th* (*squads*) *to celic* and *2nd and 3rd* (*squads*) *to celic*. Two action effects are salient to the sergeant, (*eagle*) *1-6 supported* and *unit fractured*. *1-6 supported* is a desirable team goal. Assume the sergeant assigns negative utility to *unit fractured* and this consequence serves as input to the back-tracing algorithm. We illustrate how to find the blameworthy agent given the sergeant's task knowledge and observations.

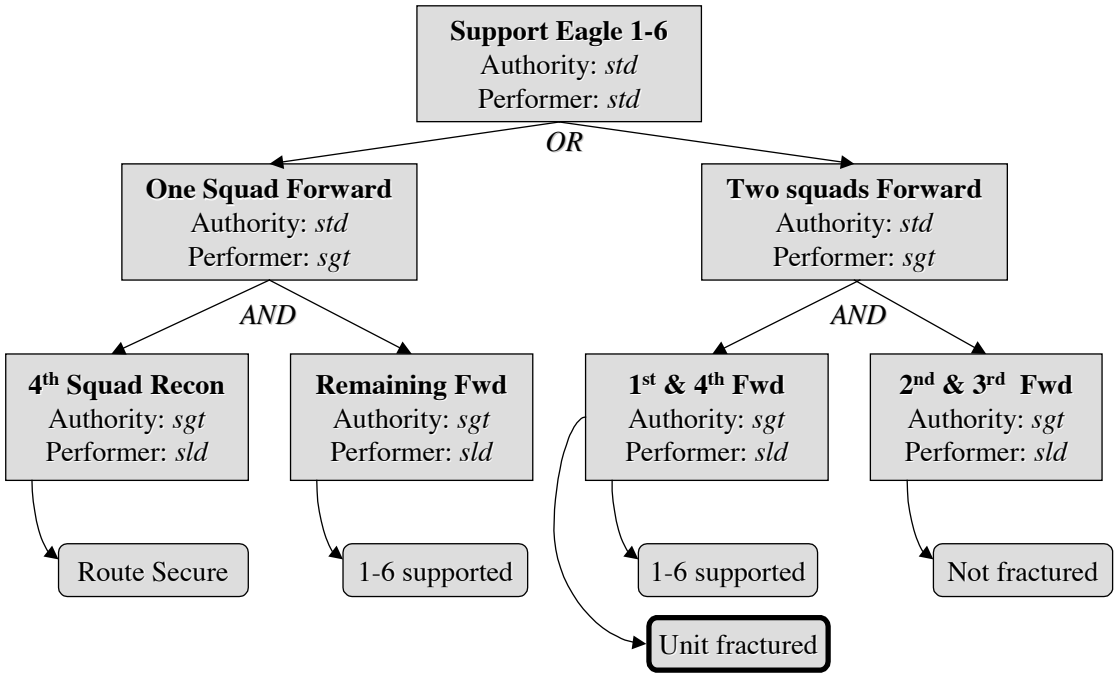

Figure 2. Team plan from the sergeant's perspective

Loop 1: The algorithm starts from primitive action *1st-and-4th-to-celic*, of which *unit-fractured* is an effect. The sergeant perceived that the *squad leader* performed the action.

Step 2.2: Initially, coerced(*sld, 1st-and-4th-to-celic, sgt*) and coerced(*sld, unit-fractured, sgt*) are unknown. Assign the *squad leader* to the responsible agent.

Step 2.3: Relevant dialogue history is *act 6*. Since the sergeant *ordered* the squad leader the act, apply *Rule 5*. The algorithm infers that the sergeant believes he *intended* the squad leader to act. Since the squad leader *accepted* by executing the action and the sergeant is the *superior*, apply *Rule 7*. The sergeant believes that he *coerced* the squad leader to act.

Step 2.4–2.5: Since coerced(*sld*, *1st-and-4th-to-celic*, *sgt*) is true and the primitive action is an *and-node*, apply *Rule 15* . The sergeant believes he coerced the squad leader to fracture the unit. Since *unit-fractured* is an effect of the primitive action, apply *Rule 12*. The sergeant believes that both he and the squad leader *knew* the action bringing about *unit-fractured*.

Step 2.6–2.7: Since coerced(*sld, unit-fractured, sgt*) is true, assign the sergeant to the responsible agent. The *sergeant* believes that he is responsible for *unit-fractured* and he has the *foreknowledge* while acting.

Since parent node is *not* the *root* of plan structure and outcome coercion is *true*, the algorithm enters next loop.

Loop 2: The action is *two-sqds-fwd*, performed by the *sergeant*. Relevant dialogue history is *sequence 1–5*. A variety of beliefs can be inferred from commonsense rules by analyzing the task structure and conversation history. The results are given below.

(1) believe(*sgt*, intend(*std*, do(*sgt*, *two-sqds-fwd*))) (*act 1* or *4*, *rule 5*)
(2) believe(*sgt*, know(*sgt*, bring-about(*two-sqds-fwd*, *unit-fractured*))) (*act 2*, *rule 3*)
(3) believe(*sgt*, know(*std*, bring-about(*two-sqds-fwd*, *unit-fractured*))) (*act 2*, *rule 3*)
(4) believe(*sgt*, know(*sgt*, alternative(*one-sqd-fwd*, *two-sqds-fwd*))) (*act 3*, *rule 9*)
(5) believe(*sgt*, know(*std*, alternative(*one-sqd-fwd*, *two-sqds-fwd*))) (*act 3*, *rule 9*)
(6) believe(*sgt*, ¬want(*sgt*, do(*sgt*, *two-sqds-fwd*))) (*act 3*, *rule 10*)
(7) believe(*sgt*, want(*sgt*, do(*sgt*, *one-sqd-fwd*))) (*act 3*, *rule 10*)
(8) believe(*sgt*, ¬intend(*std*, do(*sgt*, *one-sqd-fwd*))) (*act 4*, *result 5*, *rule 11*)
(9) believe(*sgt*, coerced(*sgt*, *two-sqds-fwd*, *std*)) (*act 5*, *result 1*, *rule 7*)
(10) believe(*sgt*, coerced(*sgt*, *unit-fractured*, *std*)) (*act 5*, *result 9*, *rule 15*)

After *loop 2*, the sergeant believes the student coerced him to fracture the unit (*Result 10*). So the *student* is responsible for the outcome.

Loop 3: The action is *support-eagle-1-6*, performed by the student. There is no relevant dialogue in history. The initial values and the responsible agent are as default. There is no clear evidence of coercion, so the sergeant believes that the *student* is the responsible agent. Parent node is the *root* of plan. The algorithm terminates.

Now the sergeant also believes that the student intended to send two squads forward and did not intend to send one squad forward (*Results 1&8*). Since the consequence set of *one-sqd-fwd* (i.e., *1-6-supported*) is subset of that of *two-sqds-fwd* (i.e., *1-6-supported* and *unit-fractured*), apply *rule 14*. The sergeant believes that the student intended *unit-fractured* and foresaw the outcome (*Result 3*), so the *student* is to blame for *unit-fractured* with *high* intensity.

7. Summary and Future Work

Based on psychological attribution theory, this report presents a preliminary computational approach to social credit assignment. The problem is central in social psychology and social cognition. With the development of human-like agent systems, it is increasingly important for computer-based systems to model this human-centric form of social inference. Our work attempts to help bridge between psychological accounts and computational models by means of AI methods. Rather than impose arbitrary rules on judgment process, our work relies on commonsense heuristics of human inference from conversation communication and causal representation of agents. Our treatments are domain-independent and thus can be used as a general approach to the problem.

This work is still in its early stages. The current implementation has focused on simple commonsense rules in contrast to the more rigorous, often non-monotonic theories typically explored in models of beliefs and intentions. Our sense is that these rules are sufficient for our practical applications, more efficient, though they are less general than those

more formal methods. Our future work must explore more deeply the relationship between these approaches. The model must also be extended before it can be fully integrated into our existing applications. To deal with uncertainty in observations and judgment process, we must incorporate probabilistic reasoning. For modeling more complex multi-agent teamwork, we need to consider joint responsibility and sharing responsibility among teammates (the current model assumes one agent has sole responsibility). Some inference rules are too restrictive and need to make better use of plan knowledge, particularly considering how preconditions and effects indirectly limit one's choices in acting. As our task representation has already encoded information about action preconditions and effects, this should be a natural extension of our existing methods.

Acknowledgement

This work was developed with funds of the U.S. Department of the Army under contract number DAAD 19-99-D-0046. We thank our colleagues on the MRE project for their collaboration.

References

J. Austin. *How to Do Things with Words*. Harvard University Press, 1962.

J. Blythe. Decision-Theoretic Planning. *AI Magazine*, 20(2):37-54, 1999.

M. Bratman. *Intention, Plans and Practical Reason*. Harvard University Press, 1987.

J. Gratch. Emile: Marshall Passions in Training and Education. In: *Proceedings of the 4th International conference on Autonomous Agents*, Barcelona, 2000.

J. Gratch, J. Rickel, E. André, N. Badler, J. Cassell and E. Petajan. Creating Interactive Virtual Humans: Some Assembly Required. *IEEE Intelligent Systems*, 17(4), pp. 54-63, 2002.

H. P. Grice. Logic and Conversation. In: P. Cole and J. Morgan (Eds.), *Syntax and Semantics: Vol 3 Speech Acts*. Academic Press, 1975.

B. Grosz and S. Kraus. Collaborative Plans for Complex Group Action. *Artificial Intelligence*, 86(2): 269-357, 1996.

A. Ortony, G. Clore and A. Collins. *The Cognitive Structure of Emotions*. Cambridge University Press, 1988.

J. Rickel, S. Marsella, J. Gratch, R. Hill, D. Traum and B. Swartout. Toward a New Generation of Virtual Humans for Interactive Experiences. *IEEE Intelligent Systems*, 17(4), pp.32-38, 2002.

J. R. Searle. *Speech Acts*. Cambridge University Press, 1969.

J. R. Searle. *Expression and Meaning*. Cambridge University Press, 1979.

M. D. Sadek. Logical Task Modeling for Man-machine Dialogue. In: *Proceedings of AAAI*, 1990.

K. G. Shaver. *The Attribution Theory of Blame: Causality, Responsibility and Blameworthiness*. Springer-Verlag, 1985.

D. Traum. *A Computational Theory of Grounding in Natural Language Conversation*. Ph.D. Thesis, University of Rochester, 1994.

D. Traum, J. Rickel, J. Gratch and S. Marsella. Negotiation over Tasks in Hybrid Human-agent Teams For Simulation-based Training. In: *Proceedings of the 2nd International conference on Autonomous Agents and Multi-agent Systems*, Melbourn, 2003.

B. Weiner. *An Attributional Theory of Motivation and Emotion*. Springer-Verlag, 1986.

B. Weiner. *The Judgment of Responsibility*. Guilford Press, 1995.

B. Weiner. Responsibility for Social Transgressions: An Attributional Analysis. In: B. F. Malle, L. J. Moses and D. A. Baldwin (Eds.). *Intentions and Intentionality: Foundations of Social Cognition*. The MIT Press, 2001.