On the Revision of Dynamic Intention Structures

Charles L. Ortiz, Jr.
Laboratory for Natural Language Understanding
Nuance Communications
Sunnyvale, CA 94085 U.S.A.
charles.ortiz@nuance.com

Luke Hunsberger
Department of Computer Science
Vassar College
Poughkeepsie, NY 12604 U.S.A.
hunsberg@cs.vassar.edu

Abstract

We developed the theory of Dynamic Intention Structures to represent and reason with incompletely specified and, possibly, mutually dependent intentions as well as the objects referenced within those intentions. The theory of DIS was expressed within a dynamic logic that drew its inspiration from work on Discourse Representation Structures in the linguistics community. In this paper, we extend our earlier work to provide a solution to the problem of intention revision in the context of incompletely specified DISs. We define a syntactically-based intention revision operation that is suitable for the fine-grained revision of the content of intentions.

The belief revision literature has focused, for the most part, on the revision of theories expressed in propositional logic. The problem of revising intentions, however, introduces several challenges involving partiality of content. Intentions typically pass through stages of elaboration (e.g., intending to rent *a* car without yet having decided on a particular car). Further, the process of revising intentions should be possible without requiring that successive elaborations be captured by *rewriting* every intention (i.e., intentions should be *elaboration tolerant* (McCarthy 1988)).

To illustrate the challenges, consider the following example in which an agent, Adam, intends to rent a car as part of his plan for an upcoming trip.

a. Adam intends to get a car by renting it.

b. Adam decides to rent a particular car, "Car39."

c. Adam decides instead to borrow a car.

Ignoring time, sentence (a) might be represented as:

$$intends(Adam, \exists(x)(car(x) \wedge get(x, Rent)))$$

in which the first argument of the predicate, "get", refers to an object and the second to the method or means of performing the get action. This is a simplification that already suffers from elaboration intolerance since means actions can themselves have means actions associated with them, an issue that will be taken up later. Sentence (b) involves the modification of the contents of the original intention to:

$$intends(Adam, get(Car39, Rent))$$

Ideally, this would not require deleting the old intention and replacing it with the above. The problem, here, is that the contents of the original intention are within the scope of both a modal operator and an existential quantifier. The revision corresponding to sentence (c) should produce:

$$intends(Adam, \exists(x)(car(x) \wedge get(x, Borrow)))$$

The difficulty in capturing this change arises from the need to refer to a particular context (e.g., "the trip Adam is planning") and then to state that within that context Adam intends not to rent a car, but to borrow one. Representing this change by a new intention—to borrow a car within that context—would normally result in the addition of that new intention and the deletion of the original intention, given, for example, an auxiliary axiom against renting and borrowing two separate cars for a single trip. This would not suffice, however, since the information regarding the ends action—getting a car—would be lost. Requiring such information to accompany the revision would require re-specifying much of the entire intention, contrary to our aims: we would like to be able to modify the contents "from the outside."

Finally, whereas the above examples involve "local" modifications to existing intentions, global changes that result in the same sorts of modifications should also be supported. For example, suppose that Adam's decision to rent "Car39" for his trip conflicts with a pre-existing intention to rent "Car39" for some other purpose (e.g., to enable his children to move some furniture). In such a case, it would be desirable for Adam's decision to rent "Car39" for his trip, as in sentence (b), to only affect the *choice* of the car to be rented for his children's use. Such fine-grained control over the revision process is not possible within common belief revision frameworks (Gärdenfors 1992).

The DIS Theory. We developed the theory of Dynamic Intention Structures (DIS) to handle incremental modifications to agent intentions (Hunsberger and Ortiz 2008), particularly within collaborative contexts (Hunsberger 1999). Inspired in part by Discourse Representation Theory (DRT) (Kamp and Reyle 1993), each DIS represents the structured content of an intention, while also providing access to its parts. The semantics of each DIS is provided by translating it into a formula in dynamic logic. Although the DIS theory includes *intention-update* operators, they focus on augmentations to existing intentions, not revisions involving local or global modifications. Furthermore, the

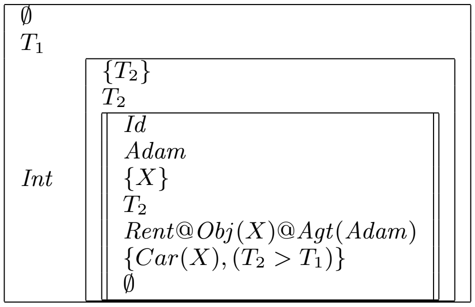

Figure 1: A DIS in "box notation"

DIS theory does not accommodate the calculation of logical consequences. This paper extends DIS theory in these directions, while respecting the need for elaboration tolerance.

A preview of our approach. Our approach involves "unpacking" intentions to enable a finer grained manipulation of propositional content. For example, consider the DIS, $\mathcal{D}$, shown in Fig. 1, which represents Adam's intention at time T_1 to rent a car at some later time T_2. Using an intention identifier, Id, enables subsequent reference to the content of this intention. In this example, Id corresponds to Adam's plan to rent a car.[1] Because the "box notation" can be quite cumbersome, this paper uses an equivalent, but more compact notation in which $\mathcal{D}$ would be notated as:

$$\mathcal{D} = \langle \emptyset, T_1, Int[\langle \{T_2\}, T_2, \langle\!\langle Id, Adam, \{X\}, T_2, \alpha, \mathcal{C}, \emptyset \rangle\!\rangle \rangle] \rangle$$

where: $\alpha = Rent@Obj(X)@Agt(Adam)$;
and: $\mathcal{C} = \{Car(X), (T_2 > T_1)\}$.

In either case, this form of a DIS is called its *canonical form.*

As in the original DIS theory, the semantics of $\mathcal{D}$ is given by translating it into an FOL formula notated as follows:

$$\|\mathcal{D}\|_{w_0} = holds(int(Holds(exists(\{t_2, x\}, \Phi), t_1)), w_0, t_1)$$

where: $\Phi = do(rent@id(id)@obj(x)@agt(adam))$ $\& \; car(x) \; \& \; gt(t_2, t1)$;
and: w_0 is the real world.

Note that $\|.\|$ translates a DIS from its canonical form into FOL. Details on this translation are given later.

Next, to support intention revision, each DIS, $\mathcal{D}$, is first converted into a *predicate form*, $\underline{\mathcal{D}}$, as illustrated below:

$$\underline{\mathcal{D}} = \{id(id), agent(id, adam), var(id, x), time(id, t_2), time_t(id, t_2), var_t(id, t_2), time_c(t_1), act(id, rent@@obj(x)@agt(adam)), constr(id, car(x)), constr(id, gt(t_2, t_1))\}$$

The identifier, id, plays a key role in the predicates in $\underline{\mathcal{D}}$. In particular, it enables subsequent reference to various parts of the content of the intention. As will be seen, for more complex intentions involving action decomposition hierarchies, the structure of such hierarchies is also captured by the predicate form of the corresponding DIS.

The inverse conversion—from predicate form to canonical DIS form—is also defined. If $\mathcal{D}_p$ is the predicate form of a DIS, then $\overline{\mathcal{D}_p}$ is the corresponding DIS in canonical form. Furthermore, $\|\overline{\mathcal{D}_p}\|$ is the translation of this DIS into FOL.

Suppose $\mathcal{D}$ and $\mathcal{D}'$ are two intentions in canonical DIS form, where $\mathcal{D}$ is a pre-existing intention and $\mathcal{D}'$ is a newly adopted intention that conflicts with $\mathcal{D}$. Such a conflict can be determined by comparing $\|\mathcal{D}\|$ and $\|\mathcal{D}'\|$. The process of intention revision generates modifications to the pre-existing intention, $\mathcal{D}$, needed to restore consistency. Our approach begins by converting $\mathcal{D}$ and $\mathcal{D}'$ to their predicate forms, $\underline{\mathcal{D}}$ and $\underline{\mathcal{D}'}$. It then incrementally augments $\underline{\mathcal{D}'}$ with predicates, $\mathcal{P}$, from $\underline{\mathcal{D}}$, generating a maximal set of predicates, $\underline{\mathcal{D}'} \cup \mathcal{P}$, such that $\|\overline{\underline{\mathcal{D}'} \cup \mathcal{P}}\|$ is consistent. To guide the process, the predicates in $\underline{\mathcal{D}}$ are partitioned into *priority classes* such that predicates representing the hierarchical structure of intentions are given highest priority, and predicates involving parameter-binding constraints are given lowest priority.

The DIS Language

The DIS language consists of a set, $Vars$, of variables; a set, $Ct = Agt \cup Acts \cup Obj \cup Index \cup Times$, of constants, where Agt is a set of agent constants, $Acts$ is a set of act type constants, Obj is a set of object constants, $Index$ is a set of unique identifiers, and $Times$ corresponds to the set of positive integers; a set, F, of function symbols including the function "@" to serve as an act-type constructor; a set, P, of predicate symbols ; and $Int(\cdot)$ to represent an intention.

Definition 1 *NATerms is the smallest set such that:*

1. *If $v \in Vars$, then $v \in$ NATerms,*
2. *If $c \in Obj \cup Agt$, then $c \in$ NATerms,*
3. *If f is an n-ary symbol in $F - \{$"@"$\}$ and $t_1, t_2, \ldots, t_n \in$ NATerms, then $f(t_1, \ldots, t_n) \in$ NATerms*

Definition 2 (Act terms) *ATerms is the smallest set such that:*

1. *If $e \in Acts$, then $e \in$ ATerms,*
2. *If $e \in A$, $f \in F$, and $x \in$ NATerms, then $e@f(x) \in$ ATerms. (We say that $f(x)$ is a modifier of act-type e.)*

The act-type constructor, @, allows the construction of more complex act types from simpler ones (Ortiz 1999). For example, suppose $drive@agt(A)@to(Boston)$ represents the act type of agent A driving to Boston. That act type could later be elaborated to any level of detail. For example, $drive@agt(A)@to(Boston)@on(Interstate)$, might represent the act type of agent A driving to Boston via Interstate 95. In this way, the act-type constructor enables the representation of partially specified intentions without committing to a particular predicate arity—such as, $drive(Agent, Object)$—and then incrementally constructing increasingly detailed intentions.

To deal with partiality of action descriptions more systematically, the arguments to act-type modifiers will often be restricted to variables. For example, the preferred description of agent A driving to Boston would be

[1] An extended version of this paper (in preparation) derives a number of rationality theorems and AGM properties, while also examining the side-effects problem and intention overloading.

$do(drive@agt(x)@to(y)) \wedge (x = A) \wedge (y = Boston)$. This technique has the advantage of enabling complex revisions to be performed simply by assigning or de-assigning values to variables.

Definition 3 *The set* $TERMS = ATerms \cup NATerms$.

Definition 4 (Dynamic Intention Structure) *We define the set, DIS, by simultaneous induction as follows.*

1. *If* $R \in P$ *is an n-ary predicate and* $T_1, \ldots, T_n \in$ *TERMS, then* $R(T_1, \ldots, T_n) \in$ *COND.*
2. *If* $T_1, T_2 \in$ *TERMS, then* $T_1 = T_2 \in$ *COND.*
3. *If* $V \in 2^{Vars}, C \in 2^{COND}$*, then* $\langle V, C \rangle \in ExtCOND$.
4. *If* $\phi \in ExtCOND$*, then* $\neg\phi \in$ *COND.*
5. *If* $\phi, \psi \in ExtCOND$*, then* $\phi \Rightarrow \psi \in$ *COND.*
6. *If* $Id \in Index, A \in Agt, V \in 2^{Vars}, T \in Vars \cup Times, Act \in ATerms, C \in 2^{COND}, S \in 2^{Index}$*, then* $\langle\langle Id, A, V, T, Act, C, S \rangle\rangle \in$ *NODES.*
7. *If* $V \in 2^{Vars}, T \in Vars \cup Times, \alpha \in$ *NODES, then* $Int[\langle V, T, \alpha \rangle] \in$ *MODAL.*
8. *If* $V \in 2^{Vars}, T \in Vars \cup Times, \mu \in$ *MODAL, then* $\langle V, T, \mu \rangle \in$ *DIS.*
9. *Nothing else is in COND, ExtCOND, NODES, MODAL or DIS, except as required by (1)-(8) above.*

In (6) above, S represents the set of, possibly empty, means actions for Act. One can think of the nodes comprising an action as representing a plan tree for that action. Each plan tree node includes a set of variables, V, which can be thought of as shared resources for Act and the sub-actions in S. Analogous to programming languages, V can be thought of as a set of variables local to a procedure.

In this paper, we only deal with the case of intentions ranging over actions. For simplicity, we also do not allow logical combinations of intentions (such as, "if p is true then the agent will intend to do α"). However, these sorts of constructions can occur in the FOL knowledge base in which inference takes place. During the revision process, we will assume that all such rules are "protected", that is, are maintained at a higher level of priority than individual intentions. A negated intention is viewed in this paper as simply the absence of an intention in the intention base. In the longer version of this paper, we eliminate these simplifications.

Definition 5 (Intention Base) *An intention base (IB) is any element of* $2^{DIS} \cup 2^{NODES}$.

The extended version of this paper formally describes the following additional restrictions on an intention base: (1) no loops in plan trees; (2) all plan trees must be rooted; and (3) all sub-actions in any DIS must have corresponding NODEs.

Semantics of DISs

The semantics of any DIS in canonical form is specified by translating it into an FOL formula in a meta-language, $\mathcal{L}$. The translation of a DIS, $\mathcal{D}$, relative to a world, w, and an intention base, I, is written $\|\mathcal{D}\|_w^I$.[2]

[2]Reference to the intention base, I, is necessary since the set, S, in a DIS (cf. Rule (6) in Defn. 4) is a set of identifiers that correspond to other NODEs in the intention base.

The specification of the translation function follows a reification strategy similar to that employed in the context of reasoning about knowledge (Moore 1985). The meta-language, $\mathcal{L}$, contains: (1) the usual logical connectives, $\{\wedge, \supset, \sim\}$, that stand for conjunction, implication and negation, respectively; (2) a set of meta-language constants that stand for variables and constants in the object language (i.e., the DIS language); and (3) a set of meta-language functions that stand for predicates and functions in the object language. In addition, $\mathcal{L}$ includes a single predicate symbol, $holds$, that ranges over terms, worlds and times: $holds(p, w, t)$. The term p in a $holds(p, w, t)$ expression has one of two forms: (1) $int(Holds(q, t'))$—with upper-case $Holds$; or (2) having no leading belief or intention operator.

We use abstract syntax for logical operators in $\mathcal{L}$; thus, $holds(p \,\&\, q, w, t)$ stands for $holds(p, w, t) \wedge holds(q, w, t)$. In addition, if V is a set, $\{v_1, \ldots, v_n\}$, we write $exists(V, \phi)$ as shorthand for $exists(v_1, \ldots, exists(v_n, \phi)\ldots)$. Finally, to report that act-type α is performed by doing act-type β, we write: $do(\alpha@method(\beta))$.

The following recursive definition of the translation function for DISs parallels the rules in Defn. 4. It also makes use of the definitions listed in Table 1 and assumes that all variables declared in (sub-)actions are unique. We also assume cross-world identity for constants, terms, predicate and function names: that is, that they have the same denotation in each possible world. We use possible worlds only to provide alternative timelines (futures) for intentions. We assume that there is a function, D, that takes a name in the object language and returns the corresponding name in the meta-language. Because of our assumptions, we will then have $D(T) = t, D(P) = p$, etc., where we adopt the convention that object-language elements are in upper case and meta-language elements are in lower case. Due to space limitations and to simplify the presentation, we skip any steps that involve calls to D and instead replace the constant with the appropriate lower- or upper-case term.

Definition 6 (Translation from DIS to FOL)

1. $\|R(T_1, \ldots, T_n)\|_w^I = r(\|t_1\|_w^I \ldots, \|t_n\|_w^I)$*, r a function*
2. $\|T_1 = T_2\|_w^I = eq(t_1, t_2)$
3. $\|\langle V, C \rangle\|_w^I = exists(v, \|C\|_w^I)$,
4. $\|\neg\phi\|_w^I = not(\|\phi\|_w^I)$
5. $\|\phi \Rightarrow \psi\|_w^I = all(\{v_1, \ldots, v_m\}, \|C_1\|_w^I \& \ldots \& \|C_n\|_w^I \rightarrow \|\psi\|_w^I)$*, where* $\phi = \langle \{V_1, \ldots, V_m\}, \{C_1, \ldots, C_n\} \rangle$
6. *Complex act terms are translated into complex terms:*
   $$\|\langle\langle Id, A, V, T, Act, C, S \rangle\rangle\|_w^I = exists(\|vars^*(Id, I)\|_w^I, do(\alpha) \,\&\, \|cstr^*(Id, I)\|_w^I),$$
   where: $\alpha = act@id(id)@agt(a)@time(t)@tree(Id, I, w)$.
7. $\|Int[\langle V, T, C \rangle]\|_w^I = int(Holds(\|\langle V, C \rangle\|_w^I, t))$,
8. $\|\langle V, T, \mu \rangle\|_w^I = (\exists v_1 \ldots \exists v_n) holds(\|\mu\|_w^I, w, t)$*, where* $V = \{v_1, \ldots, v_n\}$.

We add the following axioms:

$$holds(do(exists(v, do(e))), w, t) \equiv holds(exists(v, do(e)), w, t) \quad (1)$$

$$holds(do(not(x)), w, t) \equiv \neg holds(do(x), w, t) \quad (2)$$

$$holds(do(\alpha@time(t)), w, t') \equiv holds(do(\alpha), w, t) \quad (3)$$

- $vars(Id, I) = V$, where $\langle\langle Id, \_, V, \_, \_, \_, \_\rangle\rangle \in I$
- $cstr(Id, I) = C$, where $\langle\langle Id, \_, \_, \_, \_, C, \_\rangle\rangle \in I$
- $subs(Id, I) = S$, where $\langle\langle Id, \_, \_, \_, \_, \_, S\rangle\rangle \in I$
- $subs^*(Id, I) = subs(Id, I) \cup (\bigcup_{s \in subs(Id,I)} subs^*(s, I))$
- $vars^*(Id, I) = \bigcup_{s \in subs^*(Id,I)} vars(s, I)$
- $cstr^*(Id, I) = \bigcup_{s \in subs^*(Id,I)} cstr(Id, I)$
- $\|\{x_1, \ldots, x_n\}\|^I_w = \{\|x_1\|^I_w, \ldots, \|x_n\|^I_w\}$
- $tree(Id, I, w) = method(act(\|s_1\|^I_w)@tree(s_1, I, w))$
  $@\ldots@method(act(\|s_n\|^I_w)@tree(s_n, I, w))$,
  where: $subs(Id, I, w) = \{s_1, \ldots, s_n\}$

Table 1: Auxiliary functions used in Defn. 6.

We extend $\|.\|$ to any intention base, IS, of DISs:

$$\|IS\|_w = \{\|I\|^I_w \mid I \in IS\}$$

We require that intentions be consistent: for any intention base, IS, it is not the case that both ϕ and $\neg\phi \in \|IS\|^I_{w,t}$. We define consequence via the relation $\models$. In particular, for any $I \in DIS$, $IS \models I$ iff $\|IS\|_w \models \|I\|^I_w$.[3]

The semantics for intention is formalized in FOL by reifying possible worlds (Moore 1985). We adopt the modal logic (System K) (Chellas 1980). Let $acc_i(., ., ., .)$ stand for the accessibility relations for intention. Then:

$$holds(int(a, Holds(p, t')), w, t) \equiv$$
$$\forall w'.acc_i(a, w, w', t) \supset holds(p, w', t')$$

The function acc_i is serial (System K). We adopt a common names assumption (Chellas 1980).

Intention revision

Most approaches to belief revision are founded on the idea of *minimal change*: to revise a set of beliefs, S, with some new p, where p is inconsistent with S, one should make the minimal change necessary to S to accommodate p. Our approach is *syntactic*, assigning greater significance to formulas, and their syntactic form, that appear in a *belief or intention* base (Ginsberg 1986; Nebel 1989; Ortiz 1999) than to the consequential closure of the corresponding base (the resulting *belief set*). Intention revision takes place in two steps within our framework as follows. Let S be an agent's current set of intentions. We translate S into its *predicate form*, as described earlier, that explicitly refers to components of a DIS so that they can be modified according to the minimality criteria above. We use the same meta level language for constants and terms as in $\mathcal{L}$ above, augmented with special predicates to name the components of a DIS (as in $id(id)$, $agent(id, a)$, and so on).

[3] In the extended version of this paper we provide a mapping, $[\![.]\!]$, that takes a representation of an intention in a FOL formula of a particular form and translates it back to canonical form, together with an analog to the revision operation given in the next section that ranges over DISs.

Definition 7 (Predicate form) *If ISB is an IB in canonical form then the translation of ISB into predicate form is (the subscripts Id_c, Id_t, Id_r are mnemonic for, respectively, "context", "top" and "root" node):*

$$\begin{aligned}
\underline{ISB} = \{&id_r(id_r), agent_r(id_r, a_r), var_r(id_r, u_r),\\
&time_r(id_r, t_r), act_r(id_r, at_r), constr_r(id_r, k_r),\\
&sub_r(id_r, b_r), var_t(id_r, u_t), time_t(id_r, t_t),\\
&var_c(id_r, u_c), time_c(id_r, t_c),\\
&\textit{for all } U_r \in V_r, K_r \in C_r, B_r \in S_r, U_t \in V_t, U_c \in V_c\\
&\quad \mid \langle V_c, T_c, int[\langle V_t, T_t,\\
&\qquad \langle\langle Id_r, A_r, V_r, T_r, At_r, C_r, S_r\rangle\rangle\rangle]\rangle \in ISB\}\\
&\cup \underline{nodes(ISB)}
\end{aligned}$$

Where $\underline{nodes(ISB)} =$

$$\begin{aligned}
&\{id(id), agent(id, a), var(id, u), act(id, at), time(id, t),\\
&constr(id, k), sub(id, id') \mid U \in V, K \in C, Id' \in S\\
&\textit{for all } \langle\langle Id, A, V, T, At, C, S\rangle\rangle \in ISB\}
\end{aligned}$$

Definition 8 (Recovering DIS Canonical form) *Let ISB be an IB in predicate form, where, PActs, the set of "positive acts", is the set:*

$\{\alpha \in \textit{ATerms} \mid \alpha \neq \textit{not}(\beta), \textit{for some } \beta \in \textit{ATerms}\}$.

$$\begin{aligned}
\overline{ISB} = \{&\langle V_c, T_c, Int[\langle V_t, T_t,\\
&\quad \langle\langle Id_r, A_r, V_r, T_r, Act_r, C_r, S_r\rangle\rangle\rangle]\rangle\\
&\mid \textit{If } act_r(id_r, act_r) \in ISB \textit{ and } Act_r \in PActs,\\
&\quad \textit{then: } agent_r(id_r, a_r), id_r(id_r), time_r(id_r, t_r),\\
&\qquad time_t(id_r, t_t), time_c(id_r, t_c) \in ISB,\\
&V_r = \{v_r \mid var(id_r, v_r) \in ISB\},\\
&V_t = \{v_t \mid var_t(id_r, v_t) \in ISB\},\\
&V_c = \{v_c \mid var_c(id_r, v_c) \in ISB\},\\
&C_r = \{c_r \mid constr(id_r, c_r) \in ISB\},\\
&S_r = \{s_r \mid sub(id_r, s_r) \in ISB\}\}\\
\cup \{&\langle\langle Id, A, V, T, Act, C, S\rangle\rangle \mid Act \in PActs\ \&\\
&id(id), agent(id, a), time(id, t), act(id, act) \in ISB\\
&\textit{where } V = \{v \mid var(id, v) \in ISB\},\\
&\quad C = \{c \mid constr(id, c) \in ISB\},\\
&\quad S = \{s \mid sub(id, s) \in ISB\}\}
\end{aligned}$$

Note that if an act is not present in the predicate form, $act(Id, Act)$, or it is a negative action ($not(Act)$) then neither it nor the corresponding node will not appear in the DIS.

Let S stand for an IB; to revise S with some ϕ we create a set of equivalence classes on S: $\{S_1, S_2, \ldots, S_n\}$ such that S_1 is meant to correspond to those elements of S that are most important and S_n to those that are least important. Revisions involve either the addition or removal of (sub)actions or constraints from or to an IB.

Definition 9 (Intention revision) *Let I and I' be DISs in predicate form and let S_i be the set of induced equivalence classes on I, $i \geq 1$. The prioritized removal of elements of I that conflict with $\neg\|I'\|$, which we write as $I \bullet I'$, is (Nebel*

1989):

$$
\begin{aligned}
I \bullet I' = \{Y \subseteq I \mid \|\overline{Y}\| \not\vdash \neg\|\overline{I'}\|,\\
Y = \cup_i Y_i, i \geq 1\\
\forall i \geq 1 : (Y_i \subseteq S_i,\\
\forall X : Y_i \subset X \subseteq S_i \rightarrow\\
(\bigcup_{j=1}^{i-1} Y_j \cup X) \vdash \neg\|\overline{I'}\|)\}
\end{aligned}
$$

We can define the operation of intention revision by some I' that is inconsistent with I as:

$$I \star I' = \cap_{(Y \in I \bullet I')} \cup I'$$

This says that we start with I' and augment it with the maximal subset of S_1 such that the result is consistent and where consistency is determined via the translation to FOL. We then repeat the process for each maximal subset of the next equivalence class and stop when no additional elements of S can be added without introducing an inconsistency.

Definition 10 (Priority classes) *Next, we assume that the root nodes of IB are totally ordered, producing the sequence: $[Id_1, Id_2, \ldots, Id_n]$. There are 3n priority classes, with $1 \leq i \leq n$ and $1 \leq j \leq 3$, written as $S_{i,j}(IB)$ and defined as:*

$$
\begin{aligned}
&S_{i,1}(IB) =\\
&\quad \{\widehat{id}(id_j), \widehat{agt}(id_j, a), \widehat{vars}(id_j, v), \widehat{act}(id_j, act),\\
&\quad\quad \widehat{sub}(id_j, id), \widehat{time}(id_j, t) \mid id_j \in subs^*(id_i, IB)\}\\
&S_{i,2}(IB) =\\
&\quad \{\widehat{constr}(id_i, p) \in IB \mid id(id_i) \in IB, p \neq eq(\_, \_)\}\\
&S_{i,3}(IB) =\\
&\quad \{\widehat{constr}(id_i, eq(x, c)) \in IB \mid x \in vars(id_i, IB),\\
&\quad\quad C \in Ct\}
\end{aligned}
$$

where, for any $p \in \{id, agt, vars, \ldots\}$, $\widehat{p}$ is an abbreviation for any of p, p_r, p_t, or p_c. For example, $\widehat{id}$ stands for any of the predicates, id, id_r, id_t, or id_c.

$S_{i,1}$ contains the highest priority predicates that define the structure of the plan tree: the node identifiers, the variables, the actions, the times, and the methods. The maximum subset of the elements of $S_{i,1}$ that are consistent with the new intention, I' (cf. Defn. 9), are included; then the next set, $S_{i,2}$, is considered. $S_{i,2}$ contains all constraints other than variable binding constraints. Again, a maximum consistent subset is added to the maximal set obtained in the previous step. The lowest priority set, $S_{i,3}$, which contains variable binding constraints, is considered in the same way. Finally, note that if an act-type entry conflicts with the revising information, as determined by the $\|.\|$ translation into FOL, then it will not be part of the revised IB. Thus, by Defn. 8, the corresponding sub-tree will be deleted from the canonical DIS, and hence will not appear in the resulting translation.

When revising an intention base, IB, by some I' that is inconsistent with IB, the highest priority is given to the root node/DIS that includes I'. In this way, the intention-revision process would begin by attempting to include as much of the DIS containing I' as possible. Then it would consider the rest of the root nodes/DISs in the totally order list (cf. Defn. 10). To capture this behavior, the identifier for the root node/DIS that contains I' must appear first in the totally ordered list of identifiers.

An extended example of revision

We consider seven steps, at times $t_1 < t_2 < \ldots < t_7$, of adopting and revising intentions. At each time-step, t_i, the canonical form of the intention base is notated $IS(t_i)$.

Step 1: Agent a intends at time t_1 to do a travel-preparation action at some later time t_p.

$$
\begin{aligned}
IS(t_1) = \{\langle\emptyset, T_1, Int[\langle\{T_p\}, T_p,\\
\langle\langle Id_1, A, \emptyset, T_p, \Theta_1, \{T_p > T_1\}, \emptyset\rangle\rangle\rangle]\rangle\}
\end{aligned}
$$

where: $\Theta_1 = Prep@Agt(A)@Time(T_p)$.

The predicate form of this intention is:

$$
\begin{aligned}
\underline{IS(t_1)} \;=\; \{\; &id_r(id_1), agt_r(id_1, a), var_t(id_1, t_p),\\
&time_c(id_1, t_1), time_t(id_1, t_p), time_r(id_1, t_p),\\
&act_r(id_1, \theta_1), constr_r(id_1, gt(t_p, t_1)) \;\}
\end{aligned}
$$

where: $\theta_1 = prep@agt(a)@time(t_p)$

and $gt(t_p, t_1)$ is the metalanguage translation of $T_p > T_1$. The translation to FOL, relative to the real world, w_0, is:

$$
\begin{aligned}
\|IS(t_1)\|_{w_0} = holds(int(Holds(exists(\{t_p\},\\
do(\theta_1@id(id_1))\,\&\, gt(t_p, t_1), t_p)), w_0, t_1)
\end{aligned}
$$

Step 2: Agent a intends to prepare by getting a truck and loading it. The truck, referred to by variable x, is a common resource for the getting and loading actions; it is placed in the parent plan tree node. The notation $IS[t'/t]$ indicates that all instances of t in IS are substituted by t'.[4]

$$
\begin{aligned}
\underline{IS(t_2)} = \underline{IS(t_1)}[t_2/t_1] \star \{&var(id_1, x), id(id_2), id(id_3),\\
&sub_r(id_1, id_2), agt(id_2, a), time(id_2, t_p),\\
&sub_r(id_1, id_3), agt(id_3, a), time(id_3, t_l),\\
&var(id_3, t_l), act(id_2, \theta_2), act(id_3, \theta_3),\\
&constr(id_2, truck(x)), constr(id_3, gt(t_l, t_p))\}
\end{aligned}
$$

where: $\theta_2 = get@agt(a)@obj(x)@time(t_p)$

and: $\theta_3 = load@agt(a)@obj(x)@time(t_l)$.

The result, in canonical form, is:

$$
\begin{aligned}
IS(t_2) = \{\langle\emptyset, T_2, Int[\langle\{T_p\}, T_p, \langle\langle Id_1, A, \{X\}, T_p,\\
\Theta_1, \{T_p > T_2\}, \{Id_2, Id_3\}\rangle\rangle\rangle]\rangle,\\
\langle\langle Id_2, A, \emptyset, T_p, \Theta_2, \{Truck(X)\}, \emptyset\rangle\rangle,\\
\langle\langle Id_3, A, \{T_l\}, T_l, \Theta_3, \{T_l > T_p\}, \emptyset\rangle\rangle\}
\end{aligned}
$$

[4] In the extended paper the persistence of intentions is handled by revising the old IB with the new one (Ginsberg and Smith 1988; Winslett 1988; Ortiz 1999). Also, since no revision takes place until step 4, for simplicity, the changes are just shown as unions.

where: $\Theta_2 = Get@Agt(A)@Obj(X)@Time(T_p)$
and: $\Theta_3 = Load@Agt(A)@Obj(X)@Time(T_l)$.

The translation into FOL yields:

$$\|IS(t_2)\|_{w0} = holds(int(Holds(exists(\{t_p, x, t_l\}, do(\theta_1@id(id_1)@method(\theta_2@id(id_2)) @method(\theta_3@id(id_3))) \ \& \ truck(x) \ \& \ gt(t_l, t_p) \ \& \ gt(t_p, t_2)), t_p)), w_0, t_2)$$

Step 3: Agent A intends to get the truck by renting.

$$\underline{IS(t_3)} = \underline{IS(t_2)}[t_3/t_2] \star \{id(id_4), sub(id_2, id_4), agt(id_4, a), time(id_4, t_p), act(id_4, \theta_4)\}$$

where: $\theta_4 = rent@agt(a)@obj(x)@time(t_p)$.

In canonical form, the result is:

$$IS(t_3) = \{\langle \emptyset, T_3, Int[\langle \{T_p\}, T_p, \langle\langle Id_1, A, \{X\}, T_p, \Theta_1, \{T_p > T3\}, \{Id_2, Id_3\}\rangle\rangle\rangle]\rangle, \langle\langle Id_2, A, \emptyset, T_p, \Theta_2, Truck(X)\}, \{Id_4\}\rangle\rangle, \langle\langle Id_3, A, \{T_l\}, T_l, \Theta_3, \{T_l > T_p\}, \emptyset\rangle\rangle, \langle\langle Id_4, A, \emptyset, T_p, \Theta_4, \emptyset, \emptyset\rangle\rangle\}$$

where: $\Theta_4 = Rent@Agt(A)@Obj(X)@Time(T_p)$.

The translation into FOL yields:

$$\|IS(t_3)\|_{w_0} = holds(int(Holds(exists(\{t_p, x, t_l\} do(\theta_1@id(id_1)@method(\theta_2@id(id_2) @method(\theta_4@id(id_4)))@method(\theta_3@id(id_3))) \ \& \ truck(x) \ \& \ gt(t_l, t_p) \ \& \ gt(t_p, t_3)), t_p)), w_0, t_3).$$

Step 4: The agent intends to rent a car, identified as *car39*. The agent has decided to rent *car39,* which is a car, not a truck. This conflicts with the existing plan to rent a truck, but not with the plan to load whatever vehicle is rented (i.e., X). We assume that the knowledge base contains, with highest priority, the following rule (here, given in FOL; this could also be written in DIS form):

$$holds(car(x) \equiv \neg truck(x), w, t) \tag{4}$$

$$\underline{IS(t_4)} = \underline{IS(t_3)}[t_4/t_3] \star \{constr(id_4, eq(x, car39)), constr(id_4, car(x))\}$$

$$S_{1,1}(IS(t_3)) = \{id_r(id_1), agt_r(id_1, a), var_r(id_1, x), var_t(id_1, t_p), act_r(id_1, \theta_1), time_c(id_1, t_4), time_t(id_1, t_p), time_r(id_1, t_p), id(id_2), sub_r(id_1, id_2), agt(id_2, a), time(id_2, t_p), id(id_3), sub_r(id_1, id_3), agt(id_3, a), time(id_3, t_l), id(id_4), sub(id_2, id_4), agt(id_4, a), time(id_4, t_p), act(id_2, \theta_2), act(id_3, \theta_3), act(id_4, \theta_4), var(id_3, t_l)\}$$

$$S_{1,2}(IS(t_3)) = \{constr_r(id_1, gt(t_p, t_3)), constr(id_2, truck(x)), constr(id_3, gt(t_l, t_p)\},$$

$$S_{1,3}(IS(t_3)) = \emptyset$$

During the revision, all of $S_{1,1}$ will go through. However, $constr(id_2, truck(x))$ (in $S_{1,2}$) conflicts with the revision and, thus, will not be included in $\underline{IS(t4)}$. To see this, we translate to FOL, and then apply axiom (1) to produce:

$$holds(int(Holds(exists(\{t_p, x, t_l\} do(\theta_1@id(id_1) @method(\theta_2@id(id_2)@method(\theta_4@id(id_4))) @method(\theta_3@id(id_3))) \ \& \ truck(x) \ \& \ car(x) \ \& \ eq(x, car39) \ \& \ gt(t_l, t_p) \& gt(t_p, t_4)), t_4)), w_0, t_4)$$

Applying (1), converting "&" to conjunction:

$$\forall w. acc_i(w_0, w, t_4) \supset \exists t_p \exists x \exists t_l. holds(do(\theta_1@id(id_1) @method(\theta_2@id(id_2)@method(\theta_4@id(id_4))) @method(\theta_3@id(id_3))), t_2)), w, t_4) \wedge holds(truck(x), w, t_4) \wedge holds(car(x), w, t_4) \wedge holds(eq(x, car39) \ \& \ gt(t_l, t_p) \ \& \ gt(t_p, t_4), w, t_4)$$

If we eliminate the *holds* expressions by explicitly referring to the accessibility relation, we can see that this is inconsistent, given axiom (4). Hence, the largest subset of $S_{1,2}$ that can be included is:

$$\{constr(id_3, gt(t_l, t_p)), constr(id_1, gt(t_p, t_3)\}$$

The remaining priority classes do not conflict and we arrive at:

$$IS(t_4) = \{\langle \emptyset, T_4, Int[\langle \{T_p\}, T_p, \langle\langle Id_1, A, \{X\}, T_p, \Theta_1, \{T_p > T_4\}, \{Id_2, Id_3\}\rangle\rangle\rangle]\rangle, \langle\langle Id_2, A, \emptyset, T_p, \Theta_2, \emptyset, \{Id_4\}\rangle\rangle, \langle\langle Id_3, A, \{T_l\}, T_l, \Theta_3, \{T_l > T_p\}, \emptyset\rangle\rangle \langle\langle Id_4, A, \emptyset, T_p, \Theta_4, \{X = Car39, Car(Car39)\}, \emptyset\rangle\rangle\}$$

In FOL:

$$\|IS(t_4)\|_{w_0} = holds(int(Holds(exists(\{t_p, x, t_l\}, do(\theta_1@id(id1)@method(\theta_2@id(id_2)@method(\theta_4)) @method(\theta_3@id(id_3))) \ \& \ eq(x, car39) \ \& \ car(x) \ \& \ gt(t_l, t_p) \ \& \ gt(t_p, t_4)), t_p)), w_0, t_4)$$

We also have the following entailments:

$$\|IS(t_4)\|_{w_0} \models holds(int(Holds(do(rent@agt(a)@obj(car39)), t_p)), w_0, t_4) \wedge holds(int(Holds(do(load@agt(a)@obj(car39)), t_l)), w_0, t_4)$$

$$\|IS(t_4)\|_{w_0} \not\models holds(int(Holds(do(load@agt(a) @obj(car39)) \ \& \ truck(car39), t_l)), w_0, t_4)$$

Step 5: The agent decides not to rent.

$$\underline{IS(t_5)} = \underline{IS(t_4)}[t_5/t_4] \star \{act(id_4, not(rent))\}$$

We have that $S_{1,1}(IS(t_4)) = S_{1,1}(IS(t_3))$; hence, by Axiom (2), $act(id_4, rent@agt(a)@obj(x)@time(t_p))$ will not survive the revision. Thus, since the act will not be present, the constraints, $X = Car39$ and $Car(X)$, will not be included in the canonical or FOL forms:

$$\begin{aligned}
&IS(t_5) = \{\langle\emptyset, T_5, Int[\langle\{T_p\}, T_p,\\
&\quad \langle\!\langle Id_1, A, \{X\}, T_p, \Theta_1, \{T_p > T_5\}, \{Id_2, Id_3\}\rangle\!\rangle\rangle]\rangle,\\
&\quad \langle\!\langle Id_2, A, \emptyset, T_p, \Theta_2, \emptyset, \emptyset\rangle\!\rangle,\\
&\quad \langle\!\langle Id_3, A, \{T_l\}, T_l, \Theta_3, \{T_l > T_p\}, \emptyset\rangle\!\rangle\}\\
&\|IS(t_5)\|_{w_0} = holds(int(Holds(exists(\{t_p, x, t_l\},\\
&\quad do(\theta_1@id(id1)\\
&\qquad @method(\theta_2@id(id_2))@method(\theta_3@id(id_3)))\\
&\quad \&\ gt(t_l, t_p)\ \&\ gt(t_p, t_5)), t_p)), w_0, t_5)
\end{aligned}$$

The resulting IB does not entail that the agent intends to load *any* car. Again, DIS provides a fine-grained revision.

Steps 6 and 7: Agent a also had intended to lend a hand truck to someone else (at some future time, t_h). After the above steps, a decides to instead use the hand truck for his own loading action. The above focused on "local" changes to an IB (i.e., changes to an intention that could have rippling effects to other parts of the intention). More interesting is the automatic global change to other intentions in an IB. This can lead to the automatic deletion of *parts* of other existing intentions that conflict with the revision. We make use of:

$$\begin{aligned}
&holds(int(Holds(do(\alpha@\beta), t)), w, t') \supset\\
&\qquad holds(int(Holds(do(\alpha), t)), w, t')\\
&holds(int(Holds(do(\alpha@with(z)), t)), w, t') \equiv\\
&\quad \neg\exists\beta.\beta \neq \alpha\\
&\quad \wedge\ holds(int(Holds(do(\beta@with(z)), t)), w, t')
\end{aligned}$$

That is, one can't use a tool for two different actions.

Consider now adding that agent a intends to help someone with hand truck H42, followed by intending to load with H42. We will assume that the second intention is held at a higher priority than the first one.

$$\begin{aligned}
&\langle\emptyset, T_6, Int[\langle\{T_h\}, T_h, \langle\!\langle Id_6, A, \{T_h\}, T_h,\\
&\quad Help@Agt(a)@With(Z),\\
&\quad \{T_h > T_6, Z = H42, HandTruck(Z)\}, \emptyset\rangle\!\rangle\rangle]\rangle
\end{aligned}$$

If we now revise our intention base with the intention to load Car39 with H42, modifying $IS(t_5)$ so that we have "...$Load@Agt(A)@Obj(X)@With(X)$..." with $Z = H42$ as a constraint, then the constraint, $constr(id_6, eq(z, h42))$ will be removed.

Summary of contributions and related work

As agents form plans, their intentions pass through stages of elaboration; at inception, an intention will typically lack sufficient specificity for execution. In this paper, we have have put forward a theory of intention revision that supports a fine-grained revision of the content of an intention. The revision process makes use of three distinct levels of representation. Canonical Dynamic Intention Structures (DISs) are data structures that explicitly capture the structure of an intention and, from the perspective of an agent designer, can be conveniently visualized in terms of constituent elements, such as resources, actions, constraints and subactions. Canonical DISs are given a formal semantics via a translation into first order logic which is also used for consistency checking during the revision process. To support the augmentation of a DIS with new information, some of which may conflict with either the intention under revision or other intentions in an intention base, we have provided a translation into an equivalent flat representation called *predicate form* which is more amenable to syntactically-based revision approaches based on minimality criteria developed within the belief revision community. Revisions can be expressed through reference to the context of a particular intention together with a description of the new elements of the intention. The revision operation minimally revises the intention as well as the contents of any other intentions that might also conflict with the revision.

Prior work in intention revision has made use of propositional languages to capture content. Much of the prior work has focused on the interaction of intentions with other attitudes such as belief as well as on *when* an intention should be dropped not *how* the contents can be changed (van der Hoek, Jamroga, and Wooldridge 2007). Such approaches provide a less fine-grained framework for revision.

Our approach was motivated by work in DRSs from the linguistics community. However, none of the prior work has examined the revision of previously constructed Discourse Representation Structures (DRSs) with new utterances (for a good overview of Discourse Representation Theory (DRT) see the work of Kamp (1993). Our semantics for DISs, however, is similar to one put forward for DRT (Kamp and Reyle 1993). Alternative proof-theoretic approaches are examined by van Eijck (2005). Kamp (1990) explores the representation, but not the revision, of mental attitudes such as belief. In that work, the semantics of modalities is dealt with in a different manner: instead, ours builds on ideas involving the reification of possible worlds as first proposed by Moore (Moore 1985). Segmented Discourse Representation Structures (Lascarides and Asher 2007) have complex embeddings of "boxes", or discourse structure elements. However, the focus there is on the handling of discourse relations and not on the representation of actions or intentions.

References

Chellas, B. F. 1980. *Modal Logic: An Introduction*. Cambridge University Press.

Gärdenfors, P., ed. 1992. *Belief Revision*. Cambridge University Press.

Ginsberg, M. L., and Smith, D. E. 1988. Reasoning about action I: A possible worlds approach. *Artificial Intelligence* 35:165–195.

Ginsberg, M. L. 1986. Counterfactuals. *Artificial Intelligence* 30:35–79.

Hunsberger, L., and Ortiz, C. 2008. Dynamic Intention Structures I: A theory of intention representation. *Autonomous Agents and Multiagent Systems* 298–326.

Hunsberger, L. 1999. Making SharedPlans more concise and easier to reason about. In *Agent Architectures, Theories and Languages V*, volume 1555 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag. 81–98.

Kamp, H., and Reyle, U. 1993. *From Discourse to logic: Introduction to modeltheoretic semantics of natural language, formal logic, and discourse representation theory*. Kluwer Academic Publishers.

Kamp, H. 1990. *Prolegomena to a Structural Account of Belief and Other Attitudes*. CSLI. 27–90.

Lascarides, A., and Asher, N. 2007. Segmented discourse representation theory: Dynamic semantics with discourse structure. *Computing meaning* 87–124.

McCarthy, J. 1988. Mathematical logic in artifical intelligence. In *Daedalus*, volume (12,45). 297–311.

Moore, R. C. 1985. A formal theory of knowledge and action. In *Formal Theories of the Commonsense World*. Ablex Publishing Corporation.

Nebel, B. 1989. A knowledge level analysis of belief revision. In *Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning*, 301–311.

Ortiz, C. L. 1999. Explanatory update theory: Applications of counterfactual reasoning to causation. *Artificial Intelligence* 108:125–178.

van der Hoek, W.; Jamroga, W.; and Wooldridge, M. 2007. Towards a theory of intention revision. *Synthese* 155(2):265–290.

van Eijck, J. 2005. Discourse representation theory.

Winslett, M. 1988. Reasoning about actions using a possible models approach. In *Proceedings of the National Conference on Artificial Intelligence*, 89–93.