A Declarative Formalization of Knowledge Translation

Saša Buvač and Richard Fikes *

Computer Science Department
Stanford University
Stanford
California 94305-2140
{buvac, fikes}@cs.stanford.edu

Abstract

We describe an interlingua-based methodology for translating encoded knowledge and present a formalism for declaratively specifying vocabulary translations within a predicate logic interlingua. In this paper we (1) use the formalism to provide a semantics for translation, (2) show that the formalism enables translation to be done as deduction by a standard theorem prover, (3) describe a proof technique for determining whether a given set of rules for translating from one vocabulary to another is sufficient for performing that translation for any theory, (4) provide examples of how the formalism can be used to specify various forms of translation, including abstractions and approximations, and (5) describe techniques for precompiling translation rules that translate directly between two given vocabularies.

1 Motivation

Acquiring and representing knowledge is the key to building powerful intelligent systems. Unfortunately, knowledge base construction is difficult and time consuming. The development of most systems requires a new knowledge base to be constructed from scratch. As a result, most systems remain small to medium in size. A promising approach to removing this barrier to the building of intelligent systems is to develop techniques for encoding knowledge in a reusable form so that large portions of a knowledge base for a given application can be assembled from knowledge repositories and other systems.

For encoded knowledge to be incorporated into a system's knowledge base, the knowledge must either be represented in the receiving system's representation language or be translatable in some practical way into that language. Since an important means of achieving efficiency in application systems is to use specialized representation languages that directly support the knowledge processing requirements of the application, we cannot expect a standard knowledge representation language to emerge that would be used generally in application systems. Thus, we are confronted with a *heterogeneous language problem* whose solution requires a capability for translating encoded knowledge among specialized representation languages.

We are addressing this problem by participating in the development of a translation technology for knowledge representation languages based on the use of an *interlingua* for communicating knowledge among systems. Given such an interlingua, a sending system would translate knowledge from its application-specific representation into the interlingua for communication purposes and a receiving system would translate knowledge from the interlingua into its application-specific representation before use. In addition, the interlingua is the language in which libraries would provide reusable knowledge bases.

Translating knowledge from one representation to another involves translating both syntax and vocabulary. Thus, an interlingua must provide a syntax and a vocabulary into and out of which knowledge can be effectively translated. A linear ASCII predicate logic syntax appears to be a suitable interlingua syntax for a broad spectrum of applications. However, no such single interlingua vocabulary is likely to be suitable, since most encoded knowledge uses a rich vocabulary of complex domain-specific terms. An interlingua that requires such terms to be restated in a low-level vocabulary of sets, tuples, integers, etc. will not be practically useful for doing translation in most cases. Thus, we ex-

*The authors would like to thank John McCarthy and Jeff Van Baalen for their valuable comments. This research is supported in part by the Advanced Research Projects Agency, ARPA Order 8607, monitored by NASA Ames Research Center under grant NAG 2-581, by NASA Ames Research Center under grant NCC 2-537, Darpa contract NAG 2-703 and N00014-94-1-0775.

pect that in practice, most interlinguas will be *domain-specific interlinguas* comprised of a general-purpose core logic-based interlingua extended to include a foundation vocabulary and theory for an application domain of interest.

We consider in this paper the problem of translating logical theories from one *representation* to another using a logic-based domain-specific interlingua. By a representation, we mean (1) a declarative representation language; (2) a vocabulary of object, function, and relation constants; and (3) a *base theory* that is considered to be included in any theory expressed in the representation. A representation's base theory is the union of the theory specified by the semantics of the representation language and whatever domain theory is being assumed in the representation.

A straightforward notion of translation might require that translation be *equivalence preserving*, i.e., that the translation of a source theory be logically equivalent to the source theory. However, translation may involve abstracting, approximating, or otherwise transforming the source theory. We describe here a multi-step translation methodology in which the translation steps into and out of the interlingua are assumed to be equivalence preserving, and translation steps within the interlingua may involve arbitrary transformations of the theory being translated, as specified by sets of translation rules.

We assume a first-order logic interlingua with a model-theoretic semantics such as the **K**nowledge **I**nterchange **F**ormat (KIF) [6] being developed in the ARPA Knowledge Sharing Initiative [11]. We assume that the interlingua includes a domain-specific vocabulary and a domain theory. Finally, we assume that any theory representable in the source language is also representable in the interlingua and that any theory representable in the target language is also representable in the interlingua.

The interlingua-based translation process we are considering can be thought of as consisting of four major steps:

1) source language syntax $\rightarrow$ interlingua syntax
2) source representation vocabulary $\rightarrow$ domain-specific interlingua vocabulary
3) domain-specific interlingua vocabulary $\rightarrow$ target representation vocabulary
4) interlingua syntax $\rightarrow$ target language syntax

The first translation step produces a theory in the interlingua using the source representation vocabulary that is logically equivalent to the source theory. Our assumption that any theory representable in the source language is also representable in the interlingua implies that every top level form in the source language which represents a logical sentence has an equivalent representation in the interlingua (i.e., the interlingua is at least as expressive as the source language). Therefore, the first translation step can be performed straightforwardly by an interpreter of a grammar whose rules describe how to parse the source language and how to compose for each parse an equivalent interlingua expression. (See [12] for a description and analysis of such grammar-based translators.)

The last translation step produces a theory in the target language that is logically equivalent to a translation of the source theory in the interlingua. As with the source language, our assumption that any theory representable in the target language is also representable in the interlingua implies that translation from the target language to the interlingua can be specified by a grammar. Such grammars can typically be written so that they are reversible [12] and therefore can be used to recognize interlingua sentences that are translatable into the target language and to construct top level forms in the target language that are equivalent to recognized interlingua sentences. Given such a reversible grammar for the target language, translation from the sublanguage of the interlingua recognizable by the grammar can be performed by an interpreter of the reversible grammar. In addition to interpreting the grammar, the translator out of the interlingua must rewrite interlingua sentences into a syntactic form that is recognizable by the grammar. Van Baalen is developing methods for doing that rewriting [5], and the work reported here may be extendible to that subtask, as discussed in the full version of this paper, [3].

Our primary focus in this paper is on the difficult second and third translation steps that involve translating into and out of the interlingua's domain-specific vocabulary. We present a formalism for specifying such vocabulary translations based on a theory of contexts being developed by McCarthy, et al, [10, 9, 4, 2, 8, 1]. A translation is specified in the formalism as a set of first-order logic sentences each of which describes a rule for deriving a sentence in a target output context that is a translation of a sentence in a source input context. We also explore some properties and uses of the formalism, including

- Using it to provide a formal semantics for translation;
- Noting that the formalism enables translation to be done as deduction by a standard theorem prover;
- Giving a proof procedure for determining whether a given set of rules for translating from one vocabulary to another is sufficient for performing that translation for any theory;
- Providing examples of how the formalism can be used to specify various forms of translation, including abstractions and approximations; and

- Describing techniques for precompiling translation rules that translate directly between two given vocabularies.

2 Defining Translation in Terms of Truth

In this section, we describe a formalism for specifying translations within a first-order logic language and use the formalism to define translation in terms of truth. Intuitively, we will say that a logical theory T' is a translation of logical theory T iff all the sentences in T' are true whenever all the sentences in T are true.

We assume a context formalism along the lines of [2, 4, 10]. A context can be thought of as an object in a first-order language which has an associated *vocabulary* (i.e., a set of non-logical symbols) and denotes a set of models. Formulas in the vocabulary of a context are referred to as *meaningful* in that context. The formalism includes a truth modality $\mathtt{ist}(\kappa, \phi)$ (pronounced "is true"), which holds iff formula ϕ is true in all the models denoted by context κ. Furthermore, we extend the $\mathtt{ist}$ modality to finitely axiomatizable theories as follows. Assume T is a finitely axiomatizable theory. Let A be a finite set of axioms such that $T = Th(A)$. Then $\mathtt{ist}(\kappa, T)$ iff $\mathtt{ist}(\kappa, \bigwedge A)$, i.e., iff the conjunction of the axioms is true in context κ. Note that the choice of A among the various axiomatizations of T is irrelevant.

A particular translation is specified by giving a set of translation rules from a "source" context to a "target" context. The rules can be thought of as describing syntactic transformations, like replacement rules or grammar rules. Formally, the rules enable one to derive sentences in the target context that are equivalent to sentences in the source context.

Definition (translation rule): A translation rule from context κ to context κ' is any axiom which has the form: $\mathtt{ist}(\kappa, \phi) \leftrightarrow \mathtt{ist}(\kappa', \phi')$.

In context formalisms, a *lifting axiom* is defined to be any axiom which relates the truth in one context to the truth in another context. Our translation rules are lifting axioms which have a particular syntactic form. There are cases where other forms of translation rules will be useful; however the study of such more general formulas is beyond the scope of this paper.

Now we can define what we mean by translating a logical theory.

Definition (theory translation): A set of translation rules Tr from context κ to context κ' translates theory T into theory T' iff $Tr \vdash \mathtt{ist}(\kappa, T) \leftrightarrow \mathtt{ist}(\kappa', T')$.

We are assuming a sound and complete logic of contexts (like the one given in [2]). Thus, we could have used $\models$ instead of $\vdash$ in the above formula. Note that translation rules are non-logical axioms. They will be provided by the user in the same manner as the non-logical axioms formalizing a domain.

Translation is typically thought of in terms of individual sentences rather than theories. However, since the top level task we are addressing is the translation of theories, we need only concern ourselves with translating the theory defined by a set of sentences, rather than with translating the sentences themselves, even when the set contains only one sentence.

In addition, defining translation to be of theories rather than of individual sentences allows us to ignore two otherwise problematic issues. First, if we considered translating individual sentences, then what would be the translation of a tautology? Since we are defining translation in terms of truth, and a truth predicate can not distinguish between tautologies, then any tautology would translate into any other tautology. That property does not match our intuitive notion of translation. Also, for the same reason, equivalent sentences such as p, $p \wedge p$, and $p \wedge (q \vee \neg q)$ would have the same translation. Again, that property does not match our intuitive notion of translation, especially when thinking of translating between natural languages.

We avoid these problems by only considering translations of theories. To see that this approach avoids the problem with tautologies, observe that every theory contains all tautologies; i.e., it is not meaningful to talk about the translation of a theory which contains one tautology and excludes the others. The second problem is avoided similarly. Every theory T has the property that if p is in T, then all the propositions q which are equivalent to p are also in T. Therefore, the set of sentences containing p, but not containing $p \wedge p$, is not a theory, and thus it is not meaningful to talk about its translation.

We now state some properties of our translation formalism which we have found to be useful for characterizing particular translations.

Definition (translatable): Theory T is *translatable* from context κ into context κ' by translation rules Tr iff there exists some theory T' such that translation rules Tr from context κ to context κ' translate theory T into theory T'. Context κ is translatable into context κ' by translation rules Tr iff every theory meaningful in context κ is translatable into context κ' by translation rules Tr from context κ to context κ'.

Definition (equivalence preserving): A translation specified by a set of translation rules Tr from context κ into context κ' is *equivalence preserving* iff for any ϕ and ψ for which there exist ϕ' and ψ' such that $Tr \vdash \mathtt{ist}(\kappa, \phi) \leftrightarrow \mathtt{ist}(\kappa', \phi')$ and $Tr \vdash \mathtt{ist}(\kappa, \psi) \leftrightarrow \mathtt{ist}(\kappa', \psi')$, it is the case that $\phi \vdash \psi$ iff $\phi' \vdash \psi'$.

Since translation is defined in terms of truth, translations in our formalism are "almost always" equivalence preserving. The only exception are translations which can not be represented by a function, because they map a single sentence to two non-equivalent sentences.

3 Translation as Deduction

One of the advantages of declaratively representing translation is that a standard theorem prover can be used to derive translations. Moreover, translation axioms specified for individual predicates will allow us to infer the translation of an entire theory using only standard logical deduction. In this section we illustrate this with an example of translating data bases. In this paper we are concerned with expressing translations in logic, and we do not investigate which special deduction systems should be used to reason about them. However, we expect that special purpose reasoners for doing translations can be designed from our logic-based specification.

We assume the following properties hold for data base contexts:

$$\mathtt{ist}(c, \phi \rightarrow \psi) \leftrightarrow (\mathtt{ist}(c, \phi) \rightarrow \mathtt{ist}(c, \psi)) \tag{1}$$

$$\neg\mathtt{ist}(c, \phi x) \leftrightarrow \mathtt{ist}(c, \neg\phi x) \tag{2}$$

$$(\forall x)\mathtt{ist}(c, \phi(x)) \leftrightarrow \mathtt{ist}(c, (\forall x)\phi(x)). \tag{3}$$

The $\Rightarrow$ direction of axiom 2 implies the closed world assumption; it is valid for data base contexts, but poses fairly severe constraints on the semantics of contexts, and will not hold in general for other types of contexts (see [2] for a discussion). Some of these rules can be derived from the others; we list them all for reader's convenience.

Here is a hypothetical example. McCarthy's diary data base and the United Airlines reservations data base use different vocabularies to represent facts about flights. In order to book a flight, McCarthy's artificially intelligent travel agent will need to translate between the two representations.

Assume McCarthy's diary contains the formula

$$\begin{gathered}(\forall d)\, \mathit{Thursday}(d) \rightarrow \\ \mathit{fly}(\mathit{UA}, d, 921, \mathit{San\text{-}Francisco}, 7{:}00, \mathit{LA}, 8{:}21)\end{gathered} \tag{4}$$

expressing the fact that every Thursday McCarthy flies to Los Angeles on flight 921, scheduled to leave San Francisco at 7:00 and to arrive in LA at 8:21. Note that although the term "McCarthy" is not mentioned in the above formula, the entry implicitly pertains to McCarthy since the formula is given in the context of McCarthy's diary.

United Airlines uses a relational data base to store reservation and flight information. The vocabulary of this relational data base contains relations *flight-record* and *passenger-record*. Using the relations in this data base, the fact given by formula 4 is represented by

$$\begin{gathered}(\forall d)\, \mathit{Thursday}(d) \rightarrow \mathit{passenger\text{-}record}(921, d, \mathit{McCarthy}) \\ \mathit{flight\text{-}record}(921, \mathit{San\text{-}Francisco}, 7{:}00, \mathit{LA}, 8{:}21).\end{gathered} \tag{5, 6}$$

Note that since all the flights listed in the United Airlines data base will be United Airlines flights, this information is implicitly assumed by the context of that data base.

If we take *diary*(p) to denote the context of person p's diary, and we take *UA-db* to denote the context of the United Airlines relational data base, then we can use the translation axioms

$$\begin{gathered}\mathtt{ist}(\mathit{diary}(p), \mathit{fly}(\mathit{UA}, d, f, l1, t1, l2, t2)) \leftrightarrow \\ \mathtt{ist}(\mathit{UA\text{-}db}, \mathit{passenger\text{-}record}(f, d, p) \wedge \\ \mathit{flight\text{-}record}(f, l1, t1, l2, t2))\end{gathered} \tag{7}$$

$$\mathtt{ist}(\mathit{diary}(p), \mathit{Thursday}(d)) \leftrightarrow \mathtt{ist}(\mathit{UA\text{-}db}, \mathit{Thursday}(d)) \tag{8}$$

to show that the theory defined by formula 4 in the *diary(McCarthy)* context translates into the theory defined by the conjunction of formulas 5 and 6 in the *UA-db* context.

We proceed to prove this. Applying the quantifier rule given in formula 3 to formula 4 gives

$$\begin{gathered}(\forall d)\mathtt{ist}(\mathit{diary}(\mathit{McCarthy}), \mathit{Thursday}(d) \rightarrow \\ \mathit{fly}(\mathit{UA}, d, 921, \mathit{San\text{-}Francisco}, 7{:}00, \mathit{LA}, 8{:}21)).\end{gathered} \tag{9}$$

From formulas 9 and 1 we can infer

$$\begin{gathered}(\forall d)\mathtt{ist}(\mathit{diary}(\mathit{McCarthy}), \mathit{Thursday}(d)) \rightarrow \\ \mathtt{ist}(\mathit{diary}(\mathit{McCarthy}), \\ \mathit{fly}(\mathit{UA}, d, 921, \mathit{San\text{-}Francisco}, 7{:}00, \mathit{LA}, 8{:}21)),\end{gathered} \tag{10}$$

which in conjunction with translation axioms 7 and 8 gives

$$\begin{gathered}(\forall d)\mathtt{ist}(\mathit{UA\text{-}db}, \mathit{Thursday}(d)) \rightarrow \\ \mathtt{ist}(\mathit{UA\text{-}db}, \mathit{passenger\text{-}record}(921, d, \mathit{McCarthy}) \wedge \\ \mathit{flight\text{-}record}(921, \mathit{San\text{-}Francisco}, 7{:}00, \mathit{LA}, 8{:}21)).\end{gathered} \tag{11}$$

Again, applying the property of `ist` given in formula 1 to formula 11 gives

$$\begin{gathered}(\forall d)\mathtt{ist}(\mathit{UA\text{-}db}, \mathit{Thursday}(d) \rightarrow \\ (\mathit{passenger\text{-}record}(921, d, \mathit{McCarthy}) \wedge \\ \mathit{flight\text{-}record}(921, \mathit{San\text{-}Francisco}, 7{:}00, \mathit{LA}, 8{:}21)),\end{gathered} \tag{12}$$

and, making use of the quantificational property of contexts described in formula 3, we get

$$\begin{gathered}\mathtt{ist}(\mathit{UA\text{-}db}, (\forall d)\, \mathit{Thursday}(d) \rightarrow \\ (\mathit{passenger\text{-}record}(921, d, \mathit{McCarthy}) \wedge\end{gathered} \tag{13}$$

$$\textit{flight-record}(921, \textit{San-Francisco}, 7{:}00, \textit{LA}, 8{:}21)),$$

which is equivalent to the conjunction of formulas 5 and 6. Since all the steps in this proof are reversible, the other direction follows in the same fashion. Therefore, we have proved that

$$\begin{gathered}\texttt{ist}(\textit{diary}(\textit{McCarthy}), (\forall d)\,\textit{Thursday}(d) \rightarrow \\ \textit{fly}(\textit{UA}, d, 921, \textit{San-Francisco}, 7{:}00, \textit{LA}, 8{:}21)) \leftrightarrow \\ \texttt{ist}(\textit{UA-db}, (\forall d)\,\textit{Thursday}(d) \rightarrow \\ \textit{passenger-record}(921, d, \textit{McCarthy}) \\ \wedge \textit{flight-record}(921, \textit{San-Francisco}, 7{:}00, \textit{LA}, 8{:}21)),\end{gathered} \tag{14}$$

which is the desired result. In the introduction we emphasized that our formalism is useful for translating between different vocabularies within a logic interlingua. Consider the *UA-db* context to be the domain-specific interlingua. Then the above example illustrates the second step in the translation process, as described in the introduction: translation from a source representation vocabulary to a domain-specific interlingua vocabulary.

3.1 Proving Translatability

Another advantage of declaratively representing translation in first-order logic is that a standard theorem prover can be used to show that any theory in one context is (or is not, as the case may be) translatable to another context via a given set of translation rules. In the above example, if we assume that context $\textit{diary}(p)$ contains only the non-logical predicates listed in formula 4, then any theory in the context $\textit{diary}(p)$ (for any person p) is translatable into the context *UA-db* by translation rules 7 and 8. Thus, any context $\textit{diary}(p)$ is translatable into the context *UA-db* by translation rules 7 and 8.

Since data bases commonly make a closed world assumption, strong results about their translatability will be provable. To show that the context of one data base is translatable into another by a set of translation rules, we use the following proposition:

Proposition (compositionality of data base translation): If c and c' are data base contexts (i.e., they satisfy formulas 1–3), then context c is translatable to c' via translation rules Tr iff all theories axiomatized by atomic formuli are translatable from c to c' via translation rules Tr.

The proof of the proposition is straightforward by using mathematical induction on formula length and axioms 1–3. It consists of showing that the translatability of the atoms in a formula implies the translatability of the formula itself. Given this proposition, we need only to prove that all the theories axiomatized by atomic formuli are translatable from one context to another in order to prove that the one context is translatable to the other.

We can also use deduction and a theorem prover to show that one context is *not translatable* to another context. This would be done by giving a counter example, i.e., a theory which is not translatable. The above proposition can again simplify this proof, because it tells us that if a counter example exists, then there exists an atomic counter example. Thus, we can restrict the search of the theorem prover to atomic propositions.

Returning to our example, note that translatability does not hold in the other direction: it is not the case that any theory in context *UA-db* is translatable into the $\textit{diary}(p)$ context via the translation axioms 7 and 8. A simple counter example will best illustrate this point. The theory axiomatized by formula 6 is not translatable into $\textit{diary}(p)$ context because the translation axioms tell us only how to translate conjunctions from the *UA-db* context into the $\textit{diary}(p)$ context.

The above proposition does not tell us anything about translatability of contexts which do not make a closed world assumption, i.e. which do not satisfy formula 2. These will, in fact, often not be translatable. In such more general cases we can only prove the following weaker proposition.

Proposition (compositionality of translation): Any theory axiomatized by an (arbitrarily quantified) conjunction of atoms in context c is translatable into context c' via translation rules Tr iff all the theories axiomatized by atomic formuli are translatable from c to c' via translation rules Tr.

This general proposition does not help in translating theories whose axiomatizations include arbitrary disjunctions.

4 Example Forms of Translation

In this section, we give examples of some of the forms of translation which can be represented in our formalism.

4.1 Abstraction

In some cases, one needs to translate a theory from a more expressive context to a less expressive context. Here, we use "more expressive" to mean "having a more detailed ontology" or describing the state of affairs at a "lower level of abstraction".

Consider, for example, two contexts which contain different concepts of snow. The "skier" context distinguishes three kinds of snow, which are represented by $\textit{dry-snow}(x)$, $\textit{wet-snow}(x)$, and $\textit{powder-snow}(x)$. The "tropics" context has only one concept for snow, described by the predicate $\textit{snow}(x)$. The obvious translation rule between the ontologies is:

$$\texttt{ist}(c_{\text{ski}}, \textit{dry-snow}(x) \vee \textit{wet-snow}(x) \vee \textit{powder-snow}(x)) \leftrightarrow \texttt{ist}(c_{\text{tropic}}, \textit{snow}(x)), \tag{15}$$

which translates any of the three types of snow distinguished by the skier into simply snow in the tropics context, and vice versa.

Disregarding for the moment any other predicates which might be present in these contexts, any theory in the tropics context is translatable into the skier context. The converse, however, is false. Only properties true of all three kinds of snow will be translatable from the skier context into the tropics context. For example, the fact that all three kinds of snow are cold is translatable, but the fact that powder snow is best for snow boarding can not be translated from the skier context to the tropics context. Intuitively, the formalism can be thought of as "blocking" any translations which would not be "completely correct" in the sense of equivalence preserving.

4.2 Leaving All Other Predicates Unchanged

Frequently, one wants to specify the translation of only a few predicates while leaving all other predicates unchanged. That is, in such cases, one wants all predicates of the source context to translate into the same predicate in the target context except for a few predicates whose translations are explicitly given.

For example, we might further elaborate the snow example by creating $c_{\text{east-coast-ski}}$, the context of skiing on the East Coast of the United States. The context $c_{\text{east-coast-ski}}$ will be just like the general skiing context c_{ski}, with the exception that the concept of "powder snow" is not meaningful in it. This can be formalized by asserting the translation rule

$$\texttt{ist}(c_{\text{east-coast-ski}}, p(\vec{x})) \leftrightarrow \texttt{ist}(c_{\text{ski}}, p(\vec{x})) \tag{16}$$

for every predicate p except for *powder-snow*(x). It is simple to verify that $c_{\text{east-coast-ski}}$ is translatable into c_{ski}. In the converse direction, all the theories which make no mention of powder snow are translatable from c_{ski} to $c_{\text{east-coast-ski}}$. Furthermore, the translation is equivalence preserving. Alternatively, if one allows second-order axioms, instead of stating the rule 23 for every predicate, we could state a single axiom:

$$(\forall p) p \neq \textit{powder-snow} \rightarrow (\texttt{ist}(c_{\text{east-coast-ski}}, p(\vec{x})) \leftrightarrow \texttt{ist}(c_{\text{ski}}, p(\vec{x}))). \tag{17}$$

Although in this example the predicate *powder-snow* was not translated at all, we could have asserted an arbitrary translation rule which would specify its translation. Translation rules which leave all other predicates unchanged are similar to *default coreference rules* as introduced in [8].

4.3 Approximation

It is often useful to create a context which approximates another context. Consider, for example, the context of laws of electricity c_{electric}, which contains the following formula for computing the resistance of wires connected in series:

$$(\forall w_1, w_2, t) \quad r(\textit{series}(w_1, w_2), t) = r(w_1, t) + r(w_2, t), \tag{18}$$

where the variables w_1 and w_2 are intended to denote particular wires, and t is intended to denote a particular temperature. The formula tells us that the resistance of two wires at temperature t connected in series is the sum of their individual resistances at that temperature. For many applications, one wants to ignore the fact that resistance depends on temperature. In such cases, one would want to use the following formula instead of formula 18:

$$(\forall w_1, w_2) \quad r(\textit{series}(w_1, w_2)) = r(w_1) + r(w_2). \tag{19}$$

We would like to be able to relate the two contexts, so that sentences from one can be translated into the other. This can be accomplished simply, along the lines of [9], by defining the context of temperature t, $c(t)$, for every particular temperature t, and using the translation rule:

$$(\forall w, t, x) \texttt{ist}(c(t), r(w) = x) \leftrightarrow \texttt{ist}(c_{\text{electric}}, r(w, t) = x). \tag{20}$$

Of course, we would also need a translation rule in the style of §4.2, stating that the *series* function remains the same across all contexts.

However, this will not work if the resistances of different wires are known for different temperatures. In such circumstances, one needs to resort to approximation by associating an approximate context $c(t_1, t_2)$ with every temperature interval (t_1, t_2), rather then with every particular temperature. The simplest translation rule is

$$(\forall w, t_1, t_2, x) \texttt{ist}(c(t_1, t_2), r(w, x)) \leftrightarrow (\exists t \in (t_1, t_2)) \texttt{ist}(c_{\text{electric}}, r(w, t) = x), \tag{21}$$

and if we wanted resistance to be a function, we could have added the obvious conjunct to the right hand side. This context is approximate because the information it contains is not as accurate as c_{electric}. For example, if in

c_{electric} the resistance of wire w_1 is 8Ω at some particular temperature t, then when translated into $c(t_1, t_2)$, (where $t_1 \leq t \leq t_2$), all we would know is that there exists some temperature in the interval (t_1, t_2) at which the resistance of w_1 is 8Ω. For practical purposes, we might also want to add a heuristic, instructing the system to always pick the smallest interval in which all the needed resistances are known.

Although adequate for translating resistances of individual wires, the translation rule 21 will not allow us to translate domain facts. Thus, the context c_{electric} is not translatable into the approximate context. For example, using the translation rule 21 and the rule stating that the *series* function remains the same across all contexts, we will not be able to translate formula 18 into formula 19. Intuitively, this is because each application of the formula 21 will introduce a new existential quantifier. In fact, the only way we could obtain formula 19 in an approximate context would be to explicitly add as a translation rule:

$$\begin{gathered}(\forall w_1, w_2, t, t_1, t_2) \\ \mathtt{ist}(c(t_1,t_2), r(\mathit{series}(w_1,w_2)) = r(w_1) + r(w_2)) \leftrightarrow \\ \mathtt{ist}(c_{\text{electric}}, r(\mathit{series}(w_1,w_2),t) = r(w_1,t) + r(w_2,t)),\end{gathered} \tag{22}$$

which explicitly states that formula 18 translates into formula 19.

Note that the translation rule 22 is not equivalence preserving: the sentence $r(\mathit{series}(w_1, w_2)) = r(w_1) + r(w_2)$ maps to many sentences in c_{electric} which are not equivalent. Since the translation is not equivalence preserving, the translation rules encode some "domain information" so that after adding the formula 22 to the c_{electric} context, facts not previously true in that context will become true. Intuitively, this means that the translation really performs some approximation and thus will not "always be correct". We proceed to illustrate this point. Assume that the only axioms in c_{electric} are $r(w_1, 20) = 7\text{k}\Omega$ and $r(w_2, 30) = 3\text{k}\Omega$. Then, in the approximate context $c(20, 30)$, it is true that $r(w_1) = 7\text{k}\Omega$ and $r(w_2) = 3\text{k}\Omega$, which follows from the translation rule 21. Therefore, from formula 19 we get $r(\mathit{series}(w_1, w_2)) = 10\text{k}\Omega$, and, using the translation rule 21, we can prove $(\exists t \in (t_1, t_2)) r(\mathit{series}(w_1, w_2), t) = 10\text{k}\Omega$ in c_{electric}. But this fact was not originally derivable in the c_{electric} context.

Another way of forcing formula 18 to translate into formula 19 would be to define translations as single directional implications. However, this would have other drawbacks. For example, every sentence would translate into any tautology. The analysis of such more general translation axioms is beyond the scope of this paper.

Note that this example suggests an "explosion" of the number of contexts. However, since these contexts need not be represented explicitly by the reasoning system, the explosion should not constitute a problem. Consider the analogy with Peano axioms for natural numbers or Tarski's theory of real fields. Although they represent structures on infinite numbers of objects, both are axiomatizable and thus usable by a theorem prover.

5 Deriving Translation Rules

Representing translation rules declaratively enables new translation rules to be derived from other translation rules. Returning again to the snow example, assume a system is given rules for translating from the skiing context c_{ski} into the tropics context c_{tropic}, and rules for translating from the tropics context into c', a context where "snow" is simply replaced by its definition, leaving all other predicates unchanged. The rule for translating skiing into tropics is given in formula 15; the rule which replaces "snow" by its definition is:

$$\begin{gathered}\mathtt{ist}(c_{\text{tropic}}, \mathit{snow}(x)) \leftrightarrow \\ \mathtt{ist}(c', \mathit{crystallized}(x) \wedge \mathit{frozen}(x) \wedge \mathit{water}(x));\end{gathered} \tag{23}$$

and we will assume rules that leave all other predicates unchanged, as discussed in the previous section. Using logical deduction, a system would be able to derive the rules for translating from the skiing context directly into c'. One application of transitivity (or cut) to the facts 15 and 23 for each implication direction will give us the following desired translation rule:

$$\begin{gathered}\mathtt{ist}(c_{\text{ski}}, \mathit{dry\text{-}snow}(x) \vee \mathit{wet\text{-}snow}(x) \vee \mathit{powder\text{-}snow}(x)) \leftrightarrow \\ \mathtt{ist}(c', \mathit{crystallized}(x) \wedge \mathit{frozen}(x) \wedge \mathit{water}(x)).\end{gathered} \tag{24}$$

Thus, rules could be automatically derived for translating directly between two vocabularies from the rules for translating each of the vocabularies into and out of an interlingua vocabulary. For example, the *Engineering Math Ontology* (EMO), [7], is an example of a domain specific interlingua for the domain of engineering mathematics. Assume that force is represented as a vector in SI units in the EMO. Now, assume we are given two other ontologies which we want to translate between. One uses English units, and the other deals only with a single dimensional space and therefore represents force as a scalar. If translation rules are given for these ontologies into and out of the EMO, a theorem prover could derive translation rules that would translate directly between the ontologies. The derived rules could then be used to perform translation more efficiently by avoiding the steps of translating into and out of the EMO.

We now give a way of characterizing derived translations.

Proposition (translatability): Assume context c_1 is translatable into context c by translation rules Tr_1, and context c is translatable into context c_2 by translation rules Tr_2. Then we can derive translation rules Tr such that context c_1 is translatable into context c_2 by translation rules Tr.

In the snow example, neither the skiing context nor the context c' will be translatable into the other via the rule we have derived. This is because neither the ski context nor the context c' is translatable into the tropics context via our translation rules. Therefore, the above proposition does not apply.

6 Summary

We have described an interlingua-based methodology for translating encoded knowledge and presented a formalism for declaratively specifying vocabulary translations within a predicate logic interlingua. We used the formalism to provide a formal semantics for translation; noted that the formalism enables translation to be done as deduction by a standard theorem prover; described a proof technique for determining whether a given set of rules for translating from one vocabulary to another is sufficient for performing that translation for any theory; provided examples of how the formalism can be used to specify various forms of translation, including abstractions and approximations; and described techniques for precompiling translation rules that translate directly between two given vocabularies.

References

[1] Giuseppe Attardi and Maria Simi. A formalization of viewpoints. *Fundamenta Informaticae*, 23(3), 1995. Also technical report TR-93-062 at the International Computer Science Institute, Berkeley, California 94704-1105.

[2] Saša Buvač, Vanja Buvač, and Ian A. Mason. Metamathematics of contexts. *Fundamenta Informaticae*, 23(3), 1995.

[3] Saša Buvač and Richard Fikes. A declarative formalization of knowledge translation. Technical Report KSL-94-59, Knowledge Systems Laboratory, Stanford University, 1994. A short version of this paper appears in the 1995 AAAI Spring Symposium Series on Information Gathering from Distributed, Heterogeneous Environments.

[4] Saša Buvač and Ian A. Mason. Propositional logic of context. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, 1993.

[5] R. Fikes, M. Cutkosky, T. Gruber, and J. Van Baalen. Knowledge sharing technology project overview. Technical Report KSL 91-71, Knowledge Systems Laboratory, Stanford University, 1991.

[6] Michael R. Genesereth and Richard E. Fikes. Knowledge interchange format, version 3.0, reference manual. Technical Report Logic Group Report Logic-92-1, Stanford University, 1992.

[7] Thomas R. Gruber. An ontology for engineering mathematics. In *Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning*, 1994.

[8] R. V. Guha. *Contexts: A Formalization and Some Applications*. PhD thesis, Stanford University, 1991. Also published as technical report STAN-CS-91-1399-Thesis, and MCC Technical Report Number ACT-CYC-423-91.

[9] John McCarthy. Notes on formalizing context. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, 1993.

[10] John McCarthy and Saša Buvač. Formalizing Context (Expanded Notes). Technical Note STAN-CS-TN-94-13, Stanford University, 1994.

[11] Ramesh S. Patil, Richard E. Fikes, Peter F. Patel-Schneider, Don Mckay, Tim Finin, Thomas Gruber, and Robert Neches. The DARPA knowledge sharing effort: Progress report. In *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning*, 1992.

[12] Jeffrey Van Baalen and Richard E. Fikes. The role of reversible grammars in translating between representation languages. In *Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning*, 1994.

This paper and other papers on context are available to WWW browsers at

http://sail.stanford.edu/buvac/

and by anonymous FTP from sail.stanford.edu in the directory pub/buvac/.