HIGHER-ORDER REPRESENTATION AND REASONING FOR AUTOMATED ONTOLOGY EVOLUTION

Michael Chan, Jos Lehmann and Alan Bundy
School of Informatics, University of Edinburgh, 10 Crichton Street, Edinburgh, U.K.

Keywords: Ontology, Automated ontology evolution, Ontology conflict detection, Higher-order logic, Ontology repair plans, Isabelle/HOL, Development graphs, GALILEO.

Abstract: The GALILEO system aims at realising automated ontology evolution. This is necessary to enable intelligent agents to manipulate their own knowledge autonomously and thus reason and communicate effectively in open, dynamic digital environments characterised by the heterogeneity of data and of representation languages. Our approach is based on patterns of diagnosis of faults detected across multiple ontologies. Such patterns allow to identify the type of repair required when conflicting ontologies yield erroneous inferences. We assume that each ontology is locally consistent, i.e. inconsistency arises only across ontologies when they are merged together. Local consistency avoids the derivation of uninteresting theorems, so the formula for diagnosis can essentially be seen as an open theorem over the ontologies. The system’s application domain is physics; we have adopted a modular formalisation of physics, structured by means of locales in Isabelle, to perform modular higher-order reasoning, and visualised by means of development graphs.

1 INTRODUCTION

Artificial intelligence and, more generally, computer science are presently faced with the challenge that autonomous software agents must be able to manipulate their own knowledge. Such knowledge is typically represented in an ontology that conceptualises the entities of the software’s application domain and allows the software to reason about such entities at a higher level of abstraction than simply the level of data or information. Just like any abstract model, ontologies are limited representations of the world, which is dynamic and inherently complex. If autonomous systems are to deal with such dynamics, they must be able to autonomously update their own ontologies.

The process of updating an ontology in the face of new information is often called ontology evolution. The literature on the subject mostly concentrates on the evolution of ontologies coded in Description Logic for Semantic Web applications (Bundy et al., 2009). The primary accent is on defining logical notions and/or methods to enable a user to maintain the consistency of an ontology either through its lifecycle or in relation to other ontologies. The former case is often related to ontology debugging and yields notions like conservative extensions (Ghiolardi et al., 2006), belief revision (Katsuno and Mendelzon, 1991) interactive ontology evolution (Stojanovic et al., 2002), inconsistency repair (Kalyanpur et al., 2006; Lam et al., 2008; Ovchinnikova and Kühnberger, 2007). The case of multiple ontologies is often related to ontology alignment and yields notions like mapping (Kalfoglou and Schorlemmer, 2003), matching (Doan et al., 2004; Giunchiglia and Shvaiko, 2004), or contextualisation. Despite these valuable research efforts on the dynamics of ontologies, we know of relatively few works that have explicitly considered the problem of applying automated mechanisms to repair locally consistent but globally inconsistent ontologies. In this type of situations belief revision may be insufficient to resolve conflicts between ontologies and the very signature of their representation language may need to be evolved. This opens up many kinds of syntactical manipulations, including splitting a function into parts and changing the arity of a function. An attempt at this kind of automated ontology evolution is described in (McNeill and Bundy, 2007), which investigates an environment in which agents with slightly different ontologies interact with each other. The main goal of the described system, the GALILEO System, is to identify and repair ontological mismatches arising from the heterogeneity in the underlying logical representation, e.g., arity mismatches.
In our view the limited progress in automated ontology evolution described above depends on two circumstances. On the one hand, most of the works address interactive ontology evolution driven by user’s instructions. This choice is a pragmatic one, dictated by the need to support ontology developers in their work, rather than by a quest for automation. In many cases, though, that is not enough because what is actually required is ontology evolution at runtime performed, for instance, by autonomous agents that communicate with each other in heterogeneous environments, including the Semantic Web. On the other hand, the focus on ontologies coded in Description Logics does not allow for a sufficiently generic analysis and resolution of ontological inconsistencies, even when an approach aims at automating, for instance, the integration of changes in ontologies. As a matter of fact, the limited expressivity of first-order or lesser logics constitutes a limit on the possibility of modelling the ontology evolution process in the same language in which the ontology is coded. Being impossible to quantify over, and thus, to reason about, the predicates; the relations and the functions of the ontology, it is very problematic to formalise and implement a sufficiently generic ontology evolution process.

We therefore turned to study automated (as opposed to user-assisted) ontology evolution using higher-order logic (HOL), which provides the benefit of making it possible to express sufficiently generic patterns of evolution. In the framework of the GALILEO system (Bundy and Chan, 2008; Chan and Bundy, 2008), a number of so-called ontology repair plans (ORPs) are being developed and implemented in HOL. These mechanisms compile together patterns for diagnosis of conflicts between ontologies and transformation rules for effecting repairs. For both development and testing, we rely on examples of ontology evolution in physics. Many seminal advances in physics are results of ontology evolution, as physicists revise predictive theories when confronted with conflicting experimental evidence. Therefore, in ORPs developed thus far, one of the ontologies represents a predictive theory; a second ontology represents a sensory or experimental set-up for that theory. When the sensory ontology generates a theorem that contradicts a theorem of the theoretical ontology, an ORP is triggered and amends the two ontologies. ORPs may act either as belief revision mechanisms or as signature revision mechanisms or both. Working in HOL provides the additional benefit of formalising concepts and their relationships with a highly expressive representation. We believe this is desirable, because physics concepts are often naturally represented

Suppose we have an ontology \( O_t \) representing the current state of a predictive physics theory and an ontology \( O_s \) representing some sensory information arising from an experiment. Suppose these two ontologies disagree over the value of some function \( stuff \) when it is applied to a vector of arguments \( \vec{s} \) of type \( \vec{x} \). \( stuff(\vec{s}) \) might, for instance, be the total energy of a ball or the orbit of a planet.

**Trigger:** If \( stuff(\vec{s}) \) has two different values in \( O_t \) and \( O_s \), then the following formula will be triggered, identifying a potential contradiction between theory and experiment.

\[
\begin{align*}
O_t & \models stuff(\vec{s}) = v_1 \\
O_s & \models stuff(\vec{s}) = v_2 \\
O_t & \models v_1 \neq v_2
\end{align*}
\]

where \( O \models \phi \) means that formula \( \phi \) is a theorem of ontology \( O \).

Figure 1: Trigger of the “Where’s My Stuff?” ontology repair plan.

as HOL objects, e.g., the orbit of a star, the rate of change in a quantity, etc.

In this paper we discuss the diagnostic mechanism of the ORP called Where’s my stuff? (WMS) (Bundy and Chan, 2008). WMS is triggered when the predicted value returned by a function, which we call \( stuff \), conflicts with the observed value of the same function. The trigger formulae of WMS are formalised in Figure 1. The purpose of WMS is to amend the signature of two conflicting ontologies by redefining the function that computes the quantity that is subject to contradiction and that instantiates the higher order variable \( stuff \). In practice, WMS deploys an addition-strategy that is quite common in physics. For instance, in order to account for unpredictable yet observed gravitational behaviours in the orbit of a planet or in the stellar orbital velocity in a galaxy, astronomers often postulate the presence of an additional unobserved planet or, resp., of dark matter. Accordingly, WMS redefines the contradictory function (in the examples, the functions orbit, resp., orbital velocity) as the sum of a visible part (i.e. the amount calculated by the original function) and an invisible part (i.e. the amount that can only indirectly be observed). For WMS’s repair operation to be triggered, its diagnostic mechanism must have individuated the function \( stuff \) and assessed a contradiction between the value of \( stuff \) in the theoretical and the sensory ontologies.
The workings of such diagnosis allow us to illus-
trates two points about using a higher-order approach
for ontology evolution:
1. The polymorphism of stuff, as well as of other
symbols like $=, \neq, <, -$, etc. permits the gen-
erality of WMS and its applicability over disparate
cases.
2. The use of a higher-order theorem prover like Is-
abelle (Paulson, 1994), allows reasoning over lo-
cally consistent but globally inconsistent ontolo-
gies that share variables.

More strongly, these two points are important re-
results for both ontology evolution and automated theo-
rem proving, and they represent the main contribution of this paper. They show that current interactive the-
orem proving technology is capable of inferring the trigger formulæ used in ontology repair plans, despite
their problematic features described above.

The rest of the paper is structured as follows: §2
gives an overview of two examples of ontology evolu-
tion in physics that are used in subsequent sections to
evaluate the proposed approach to the representation
and reasoning for ontology evolution; §3 describes
the structure of the ontological representation and the
specific axioms of the theoretical and sensory ontolo-
gies; and, §4 highlights the advantages of detecting
conflicts between ontologies using HOL. Note that
details of ontology repair procedures are not covered
in this paper; we refer interested readers to (Bundy
and Chan, 2008; Chan and Bundy, 2008) for more
complete presentations.

2 TWO EXAMPLES OF
ONTOLOGY EVOLUTION IN
PHYSICS

In this paper, we base our evaluation of the represen-
tation of knowledge and reasoning on two examples
of ontology evolution in physics: the bouncing-ball
paradox and the proposed existence of dark matter.
Both cases can be emulated by WMS.

The bouncing-ball paradox, as described in (diSessa, 1983), involves dropping a ball from a
height above ground and calculating its total energy
as the sum of its kinetic energy, which is a function of
the ball’s velocity, and of its potential energy, which
is a function of its height on the ground. The initial
amount of total energy of the ball will then be zero Joules be-
cause of its zero velocity and zero height at ground
level. The paradox is exactly the contradiction be-
tween the initial and final amounts of total energy of
the ball: the law of conservation of energy requires
such amounts to be the same. WMS emulates the usual
solution to the paradox and adds to the function that
computes the total energy of the ball a third compo-
ment, for elastic energy. This is the type of energy
to which the ball’s kinetic energy is transformed at
the time of impact with the ground. This solution is
equivalent to re-idealising the ball as a spring rather
than as a particle.

In the case of the hypothetic existence of dark matter,
the evidence for it comes from various sources,
for instance, from an anomaly in the orbital veloc-
ties of stars in spiral galaxies identified by Rubin
in 1975. Given the observed distribution of mass in
these galaxies, we can use Newtonian Mechanics to
predict that the orbital velocity of each star should
be inversely proportional to the square root of its
distance from the galactic centre (called its radius). How-
ever, observation of these stars show their orbital
velocities to be roughly constant and independent of
their radius. Figure 2 illustrates the predicted and ac-
tual graphs. In order to account for this discrepancy
it is hypothesed that galaxies contain an invisible halo
of, so called, dark matter which does not radiate and
can only be measured indirectly. Accordingly WMS
adds to the function that computes the stellar orbital
velocity a second component that depends on dark
matter.

This diagram is taken from
http://en.wikipedia.org/wiki/Galaxy_rotation
_problem. The x-axis is the radii of the stars
and the y-axis is their orbital velocities. The
dotted line (A) represents the predicted graph
and the solid line (B) is the actual graph that
is observed.

Figure 2: Predicted vs Observed Stellar Orbital Velocities.
In the next sections we discuss how the physics knowledge underlying these two cases can best be represented in FOL and HOL, respectively, in order for a HOL-theorem prover like Isabelle to diagnose the contradiction between what is expected and what is observed.

3 ONTOLOGICAL REPRESENTATION OF PHYSICS

The language physicists use for expressing relationships between concepts is largely based on mathematics rather than on an expressive logic. It is one of our contributions in this project to provide a logical formalisation of physics formulae and historical examples of ontology evolution. As already mentioned, the need for evolution arises when experimental observations contradict theoretical predictions; thus, to formalise such situations, the predictive theory and sensory data are encapsulated in separate ontologies, which we call \( O_t \) and \( O_s \), respectively. Such modular representation, though basic, provides a range of benefits, including better control of contradiction, more focused effects of repair, variable certainty and increased reusability. To further modularise the existing knowledge representation, the physics and mathematical theories can be partitioned into small ontologies. Each ontology resembles a small context, e.g., some treatment of physics may depend on separate ontologies for a formal theory and for a naive theory of geometry, which only covers 2-D spaces. Certainty factors can be assigned to an ontology, determining the vulnerability of the theory or experiment to repair. For instance, the ontology of a controversial theory could be valued at a lower confidence than an established one. The structure used for storing and managing the ontologies should therefore be able to accommodate a sufficiently large collection of partitioned ontologies because even fundamental ontologies themselves, e.g., Newtonian Mechanics, arithmetic, etc., may be represented as a collection of smaller ontologies. That said, the structure should also support the management of relations and dependencies between ontologies and their terms.

3.1 Modular Representation as a Development Graph

For the storage and management of the collection of ontologies, we use a formal logical representation called development graphs (Autexier et al., 1999), in which nodes and links correspond to ontologies and morphisms, respectively. A logical theory (in our case, an ontology) is characterised by a node, which can be defined to import signatures and axioms from other nodes via definitional links. As will be described later, the ontologies are formalised as locales (Ballarin, 2004) in Isabelle, which are mechanisms for performing modular reasoning; each locale corresponds to a node in the development graph. There are also other types of morphisms implemented, the details of which are not covered in this paper. Development graphs are not only useful for formalisation of ontologies, but also for visualisation of the relations between ontologies and the complete structure. Development graphs are already implemented in HETS (Mossakowski et al., 2007), which is a system for the analysis of various specification languages.

3.1.1 Example Representation

To provide a visualisation of the structure of the ontologies for the model of the bouncing-ball paradox, Figure 3 depicts a development graph containing the relevant ontologies (nodes) and definitional links (arcs); it is an illustration of the development graph visualised in HETS. In this representation, the top ontology BasicPhys contains the fundamental concepts in physics, e.g., time and events. It extends from OrderedReals, which is an internal specification of reals with ordering. The node ClassicalEnergyConv contains types specific to energy conversion, e.g., those for various types of energy, including total energy, kinetic energy, and potential energy, but not the theories describ-
ing the conversion between types of energy. Note that it extends from BasicPhys, so all sorts (types) and operations, if any, are directly imported. Following the path down, the node OtLaws contains the theory of energy conversion for particles (without extent) between kinetic and potential energies for all objects and time moments, i.e. ∀o:Obj,t: Time. \( TE(o,t) = KE(o,t) + PE(o,t) \); that is, it predicts that potential energy can be converted to only kinetic energy because elastic and other types of energy are neglected. Moreover, it contains definitions as well, e.g., ∀o:Obj,t: Time. \( KE(o,t) = \frac{1}{2}Mass(o,t).Vel(o,t)^2 \). The theoretical ontology \( Ot \) extends from OtLaws, which imports the same predictive theory as that in OtLaws. In addition, it contains axioms specifying that the initial velocity is zero and the height is greater than zero. In contrast, the sensory ontology \( Os \) extends from ClassicalEnergyConv. OtLaws contains similar axioms as OtLaws, but at a lower level of generality. The axioms of OtLaws cover only the specific entities and events involved in the experiment, e.g., \( TE(ball,End(drop)) = KE(ball,End(drop)) + PE(ball,End(drop)) \) which restricts the definition to the particular ball being dropped and the particular dropping event involved. Os, unlike \( Ot \), contains axioms specifying that the final height and velocity are both zero.

The development graph of the dark matter example exhibits a similar structure, so it is omitted to avoid repetition.

### 3.2 Axiomatisations of Theoretical and Sensory Ontologies

Representational choices need to be made regarding the axiomatisations of \( O_t \) and \( O_s \). The ontological representation of the predictive theory as \( O_t \) is relatively straightforward as \( O_t \) requires access to the same physics laws as those needed in the case study, which are encoded as axioms; the axioms are contained in \( O_t.Law \), but are exported to \( O_t \). As briefly described already, \( O_t \) has access to the same axiomatised laws in \( O_t \), but with lesser generality; these are contained in \( O_t.Law \), but are exported to \( O_s \). Instead of expressing the laws over the entire relevant domain, the domain of quantification in \( O_s \) is specific to the entities involved in the experiment. Therefore, \( O_s \) makes a lesser commitment than \( O_t \) because it commits itself only to the entities of the experiment. For the bouncing-ball paradox, the axioms of the \( O_t \) and \( O_s \) are:

\[
Ax(O_t) ::= \left\{ \begin{align*}
\forall pPart, t, t_i: & \text{Mom.} \, TE(p,t_i) = TE(p,t_i), \\
\forall pPart, t: & \text{Mom.} \, TE(p,t) = KE(p,t) + PE(p,t), \\
\forall pPart, t: & \text{Mom.} \, KE(p,t) ::= \frac{1}{2}Mass(p,t).Vel(p,t)^2, \\
\forall pPart, t: & \text{Mom.} \, PE(p,t) ::= Mass(p,t).G. \\
& \text{Height}(p,t) \\
\end{align*} \right.
\]

\[
Ax(O_s) ::= \left\{ \begin{align*}
& TE(ball,End(drop)) = KE(ball,End(drop)) + PE(ball,End(drop)), \\
& KE(ball,End(drop)) ::= \frac{1}{2}Mass(ball,End(drop)). \\
& Vel(ball,End(drop))^2, \\
& PE(ball,End(drop)) ::= Mass(ball,End(drop)).G. \\
& \text{Height}(ball,End(drop)) \\
\end{align*} \right.
\]

where \( TE(p,t) \), \( KE(p,t) \), and \( PE(p,t) \) respectively denote the amount of total energy, kinetic energy, and potential energy of an object \( p \) at a time moment \( t \); \( Mass(p,t) \) and \( Vel(p,t) \) respectively denote the mass and velocity of \( p \) at \( t \); \( G \) is the gravitational constant; \( \text{Start}(drop) \) and \( \text{End}(drop) \) respectively denote the start and end of the dropping of the ball.

For the dark matter case study, the axioms of \( O_t \) and \( O_s \) are:

\[
Ax(O_t) ::= \left\{ \begin{align*}
\forall \text{Obj},g: & \text{Galaxy. \, AngVel}(o,g) = \frac{\text{OrbVel}(o,g)}{\text{Rad}(o,g)} \quad (4) \\
\forall \text{Obj},g: & \text{Galaxy. \, Rad}(o,g) > 0, \quad (5) \\
\forall \text{Obj}: & \text{GraphA}(o) = (\text{Rad}(o,MWay), \quad (6) \\
& \text{AngVel}(o,MWay),\text{Rad}(o,MWay)) \\
\end{align*} \right.
\]

\[
Ax(O_s) ::= \left\{ \begin{align*}
\forall \text{Obj}: & \text{AngVel}(o,MWay) = \frac{\text{OrbVel}(o,MWay)}{\text{Rad}(o,MWay)}, \quad (7) \\
\forall \text{Obj}: & \text{Rad}(o,MWay) > 0, \quad (8) \\
\forall \text{Obj}: & \text{GraphB}(o) = (\text{Rad}(o,MWay), \quad (9) \\
& \text{OrbVel}(o,MWay)) \\
\end{align*} \right.
\]

where \( \text{AngVel}(o,g) \) and \( \text{OrbVel}(o,g) \) respectively denote the angular and orbital velocities of an object \( o \) in
the galaxy \( g \); \( \text{Rad}(o, g) \) returns radius between \( o \) and the centre of the galaxy \( g \); and \( \text{MW} \) is our galaxy, Milky Way.

The representation of \( O_t \) (and \( O_s \text{Laws} \)) adopted may appear \textit{ad-hoc} as it requires \( O_t \) to have access to the same laws as \( O_s \) but expressed at the lowest level of generality. However, this reflects a situation where the sensory data in \( O_s \) is interpreted under the context of the theory \( O_t \), so the data would be interpreted using the laws available in \( O_t \). The minimal exportation of the theoretical laws, therefore, involves the same laws but at the lowest level of generality. This interpretation is analogous to a physicist’s making sense of new sensory data in accordance to his/her current physics theory and understanding of the initial experiment setting. Of course, there are other alternative representations that are worthy of further work, each implying a different philosophy of the content of \( O_t \) and of the set of deducible theorems.

(a) \( O_t \) has access to the same axiomatised laws as those imported by \( O_s \) and have the same domain of quantification, so \( O_t \) has access to the same physics as \( O_s \). One shortcoming of this representation is the loss of distinction between predictive theory and mere experimental evidence. Adopting such representation is equivalent to modelling the evolution of a physicist’s ontological understanding of physics when confronted by \textit{two} conflicting sets of data.

(b) \( O_s \) shares only the language of \( O_t \) and does not import any physics laws and definitions, so \( O_s \) is a knowledgebase. In order to derive a contradiction between the two ontologies without losing the distinction between them, the reasoning mechanism must then be able to access both \( O_t \)’s and \( O_s \)’s axioms. This is more general than those described above because, in an environment in which there are multiple theoretical ontologies confronting the same \( O_t \), the deducible theorems then depend on the axioms of the particular theoretical ontology.

(c) \( O_t \) is not committed to share the language or the axioms of \( O_s \), so this is the most general representation. The terms in \( O_t \) and \( O_s \) still need to be related in order to perform reasoning. One approach is to introduce metal-level relations between terms across the two ontologies, perceiving each as a different context (McCarthy and Buvac, 1998). The reasoning mechanism needs to account for both the meta-level relations between terms and object-level formulation, which significantly increases the complexity of the task.

Undoubtedly, representation (c) is the most interesting avenue given the potentially high level of expressivity and generality. It is similar to the current representation in the way that both address the need for interpretation of the data in \( O_t \) in the context of \( O_s \). However, the required axioms for contextual reasoning in the current representation are explicitly encoded in \( O_s \text{Laws} \) at the object-level, which can handle only \( O_s \), whereas in (c) these may be represented at the meta-level, which can handle multiple, arbitrary contexts. That said, we believe the representation we have currently adopted strikes a reasonable balance between expressivity and complexity.

4 REASONING FOR AUTOMATED ONTOLOGY EVOLUTION

The initial reasoning step for ontology evolution is to determine whether a conflict exists between the ontologies and which ontology repair plan should be triggered. Without a robust and correct reasoning mechanism, a repair plan could be triggered to modify an already correct ontology or might not be triggered when it is supposed to be: giving rise to false-positives and false-negatives, respectively. Using the representation described previously in §3, the diagnosis can take place by translating nodes into logical theories in the logic of a particular theorem prover and then attempting to deduce trigger formulae in those theories.

4.1 Detection of Conflicts between Ontologies

If we reason with an inconsistent ontology, then every formula is deducible in multiple ways, leading to an explosion of provable theorems. Therefore, when we prove whether a trigger formula is a theorem of some given ontologies to detect a conflict between them, the ontologies cannot be first merged.

For reasoning in both first-order and higher-order logics, we use the interactive theorem prover Isabelle because its emphasis on proving higher-order theorems. Isabelle, although powerful, does not offer tools tailored for reasoning with modular ontologies. Fortunately, there are at least two workarounds for reasoning modularly without reverting to merging the axioms of the ontologies and giving rise to an inconsistent set of axioms. One is to specify the ontologies as separate Isabelle theories and another is to specify them as separate locales (Ballarin, 2004), which are mechanisms for defining local scopes in a proof. The
locales approach is more attractive because each of our ontologies can be viewed as an individual context and theorems can be proved in the context of a specified locale. That said, each locale corresponds to a node in a development graph as each node represents an ontology, which is analogous to a context. In this section, we present the procedure for the diagnosis of conflict in the bouncing-ball and dark matter case studies. The type of conflict under scrutiny is that for

ified locale. That said, each locale corresponds to a

and theorems can be proved in the context of a spec-

ified as a locale. Locale BasicPhys corresponds to

the node BasicPhys in Figure 3, which represents an

ontology specifying the language of O₁ and O₂:

locale BasicPhys =
  fixes Vel :: "Obj ⇒ Time ⇒ real"
  and Height :: "Obj ⇒ Time ⇒ real"
  and Start :: "Event ⇒ Time ⇒ real"

... The locale ClassicalEnergyConv corresponds to

the node ClassicalEnergyConv in the development

graph, which imports the signature from

BasicPhys and extends it with a language for rep-

resenting various types of energies:

locale ClassicalEnergyConv =
  fixes TE :: "Obj ⇒ Time ⇒ real"
  and KE :: "Obj ⇒ Time ⇒ real"
  and PE :: "Event ⇒ Time ⇒ Energy"
...

Locale Ot Laws corresponds to the node OtLaws in

the development graph and contains the axioms constituting the definitions of total energy (without elasticity), potential energy, kinetic energy, the law of conservation of energy, and the gravitational constant²:

locale OtLaws = BasicPhys +
  assumes te: "TE p t = PE p t + KE p t"
  and pe: "PE p t = Mass p × G × Height p t"
  and ke: "KE p t = 0.5 × Mass p × Vel p t × Vel p t"
...

Locale Ot corresponds to Ot and asserts the values

of the initial velocity, initial height, and mass of the

ball:

locale Ot = OtLaws +
  assumes vinit: "Vel ball (Start drop) = 0"
  and hinit: "Height ball (Start drop) > 0"
  and mass: "Mass ball > 0"

Locale OsLaws corresponds to the node OsLaws and

represents an ontology containing axioms based on

the laws over a specific domain:

locale OsLaws = BasicPhys +
  assumes te: "TE p t = PE p t + KE p t"
  and pe: "PE p t = Mass p × G × Height p t"
  and ke: "KE p t = 0.5 × Mass p × Vel p t × Vel p t"
...

The locale Os corresponds to Os and asserts the values

obtained from observation, i.e. the final values of the velocity, height, and mass of the ball:

¹a::t denotes that a is of type t.

²If F is a function, F x denotes the application of F to x.
can be instantiated with the following substitution:
graphs in the formulation. Formulae (1), (2), and (3)
compared, function objects can be used to represent
scribed earlier, since the two graphs in Figure 2 are
which requires a higher-order representation. As de-
The proposed existence of dark matter is a case study

4.1.2 By Higher-order Proof Calculus

The final proof goal to be discharged is (3), which
can be proved using the instantiation (12):

lemma (in O) lem1: "TE ball (End drop) = 0"
  using mass vfin hfin te pe g by auto

theorem (in O) OtWMS1: "EX (stuff::'?a => '?b) s v1. stuff s = v1"
proof (intro exI) qed (rule lem1)

The second proof goal to be discharged is (2),
which can be proved using the instantiation (11):

lemma (in Os) lem2: "TE ball (End drop) = 0"
  using mass vfin hfin te pe g by auto

theorem (in Os) OsWMS1: "EX (stuff::'?a => '?b) s v2. stuff s = v2"
proof (intro exI) qed (rule lem2)

The first proof goal to be discharged is (1), which

4.1.2 By Higher-order Proof Calculus

The proposed existence of dark matter is a case study
which gives the instantiated form of the trigger for-
mulame as follows:

\begin{align*}
O_t & \vdash \lambda s \in MWay. \langle \text{Rad}(s) \rangle,
O_t & \vdash \lambda s \in MWay. \langle \text{OrbVel}(s) \rangle = \text{Graph}_p \\
O_t & \vdash \lambda s \in MWay. \langle \text{OrbVel}(s) \rangle = \text{Graph}_a \\
O_t & \vdash \text{Graph}_p \neq \text{Graph}_a
\end{align*}

where \( \text{OrbVel}(s) \) is the orbital velocity of star \( s \), \( \text{Rad}(s) \) is the radius of \( s \) from the centre of the Milky Way, and \( MWay \) is our own galaxy, represented as the set of stars it contains. Formula (14) shows the predicted graph, \( \text{Graph}_p \); the orbital velocity decreases roughly inversely with the square root of the radius (see Figure 2). This graph is deduced by Newtonian Mechanics from the observed distribution of the visible stars in the Milky Way. Formula (15) shows the actual observed orbital velocity graph, \( \text{Graph}_a \); it is almost a constant function over most of the values of \( s \) (see Figure 2). Note the use of \( \lambda \)-abstraction to create graph objects as unary functions. These two graphs are unequal (15), within the range of legitimate experimental variation.

The following proof illustrates the power of using
Isabelle’s higher-order proof calculus to detect a con-

A constant for representing the Milky Way:

consts MWay :: Spiral

Locale BasicPhys corresponds to BasicPhys in
the development graph, which represents the ontology
containing only the language of locales \( O_t \) and \( O_s \) in
this particular case study:

locale BasicPhys = 
  fixes OrbVel :: "Obj => Obj set => real"
  and GrphP :: "Obj => real x real"
  ... assumes cab: "\exists P. P \neq \{\} \wedge (\forall x \in P \rightarrow CurveA x \neq CurveB x)"
  and gcp: "GrphP = CurveP"
  and gca: "GrphA = CurveA"
locale OtLaws = BasicPhys +
  assumes radgtzero: "\forall p. g \neq 0" and ovabsov: "\forall OrbVel p g = abs (OrbVel p g)"

Locale \( Ot \) contains the definition of angular velocity
in terms of orbital velocity. It also asserts that its
graph is a plot of the product of angular velocity by radius:
locale Ot = OtLaws +
  assumes avel: "AngVel p g = OrbVel p g / Rad p g"
  and ga: "GrphP p = (Rad p MWay, AngVel p MWay × Rad p MWay)"
locale OsLaws = BasicPhys +
  assumes radgtzero: "∀p. p ∈ MWay → Rad p MWay > 0"
  and ovabsov: "∀p. p ∈ MWay → OrbVel p MWay = abs (OrbVel p MWay)"

Locale Os explicitly asserts that its graph is a plot of the orbital velocity of stars in the Milky Way.

locale Os = OsLaws +
  assumes gb: "GrphA p = (Rad p MWay, OrbVel s g)"

Similar to the previous proof, the first proof goal to be discharged here is (1), which can be proved using the instantiation (14):

lemma (in Ot) lem1: "(λ g s. (Rad s g, OrbVel s g) MWay = GrphP)" 
  apply (simp add: expand_fun_eq)
  ... 

theorem (in Ot) OtWMS1: "∃{stuff::?'a ⇒ ?'b} s v1. stuff s = v1" 
proof (intro ex1) qed (rule lem1) 

The second proof goal to be discharged is (2), which can be proved using the instantiation (15):

lemma (in Os) lem2: "(λ g s. (Rad s g, OrbVel s g)) MWay = GrphA" 
  by (simp add: expand_fun_eq)

theorem (in Os) OsWMS1: "∃{stuff::?'a ⇒ ?'b} s v2. stuff s = v2" 
proof (intro ex1) qed (rule lem2) 

The final proof goal to be discharged is (3), which can be proved using the instantiation (15):

theorem (in Ot) OtWMS2: "GrphA ≠ GrphP" 
  using cab gca gcp by auto

5 DISCUSSION

The two case studies presented have shown the benefits of representing the predictive theory and the sensory data as separate ontologies. By encoding each ontology as an individual locale that is locally consistent, each of the three parts of the WMS trigger formulae is simply an open theorem of the relevant ontology. If the two were merged, then there would be an explosion of uninteresting theorems. Moreover, the case studies have demonstrated the need for higher-order logic and the power of using a higher-order theorem prover such as Isabelle for aiding automated ontology evolution. For example, Isabelle’s polymorphic meta-logic is particularly useful for the detection of the trigger formula because *stuff* has a polymorphic type *a ⇒ b* and, before diagnosis, how it is to be instantiated is not known. An obvious advantage is that the type of *stuff* is a variable, which provides a sufficiently high level of generality in the trigger formula. As shown in the proof for the bouncing-ball paradox, *stuff* is instantiated by *TE* with type *Obj ⇒ Time ⇒ real*, whereas in that for the existence of dark matter, *stuff* is instantiated by *λ s t. (Radius s t, Orb s t)* with type *Obj ⇒ real × real*. Moreover, the proof of the trigger formula (2) requires the comparison using the equality and the inequality operators, which are polymorphic as well. For example, real numbers and functions are compared in the bouncing-ball and the dark matter case studies respectively, so the operators have defined meanings on reals in one scenario and on functions in another.

On the representational aspect, if a less expressive logic, e.g., DL or FOL, was adopted, it would be impossible to reason over function objects. Significant changes to the representation would be required in order to perform the described kind of reasoning. For example, in the dark matter case study, the representation of the function of the orbit of a star could no longer be a functional object, but a (possibly infinite) set of positional points in a 3-D space, which we believe is unnatural.

6 CONCLUSIONS

Further progress in handling automated ontology evolution is now urgent, due to the demand created by multi-agent systems. We have outlined two main challenges to the development of mechanisms supporting automated ontology evolution, i.e. designing a modular representation and performing reasoning across modular ontologies. The latter imposes a relatively greater challenge in our domain as it demands an unusual use of higher-order theorem proving with interactive provers. As described, a formal logical structure is adopted to store and manage ontologies and ontologies themselves are treated as expressive logical theories. Evident by the two described examples from physics, our work is showing the advantages of the unusual use of Isabelle for higher-order reasoning with modular ontologies and the visualisation of the structure as a development graph.
ACKNOWLEDGEMENTS

The research reported in this paper was supported by EPSRC grant EP/E005713/1.

REFERENCES


