Available online at www.sciencedirect.com

SCIENCE DIRECT®

Theoretical Computer Science 354 (2006) 98–117

Theoretical
Computer Science

www.elsevier.com/locate/tcs

Metaphors and heuristic-driven theory projection (HDTP)[☆]

Helmar Gust[a], Kai-Uwe Kühnberger[a,*], Ute Schmid[b]

[a] Institute of Cognitive Science, University of Osnabrück, 49069 Osnabrück, Germany
[b] Department of Information Systems and Applied Computer Science, University of Bamberg, 96045 Bamberg, Germany

Abstract

A classical approach of modeling metaphoric expressions uses a source concept network that is mapped to a target concept network. Both networks are often represented as algebras. In this paper, a representation using the mathematically sound framework of heuristic-driven theory projection (HDTP) is presented which is—although quite different from classical approaches—algebraic in nature, too. HDTP has the advantage that a structural description of source and target can be given and the connection between both domains are more clearly specified. The major aspects of the formal properties of HDTP, the specification of the underlying algorithm HDTP-A, and the development of a formal semantics for analogical reasoning will be discussed. We will apply HDTP to different types of metaphors.

Keywords: Metaphors; Analogies; Anti-unification; HDTP

1. Introduction

Much work has been invested to analyze, model, and conceptualize metaphoric expressions in natural language. Although the interest in this topic is significantly high, the number of existing formal approaches, in particular, the number of approaches, that can be tested on computers, is less developed.[1] The reason for this is mainly based on the fact that formal models for metaphors are relatively hard problems. Even the methodological basis of a model for metaphors is controversial and disputed.

An influential example for a formally more or less spelled-out framework is [18].[2] In this account, Indurkhya represents the target concept network and the source concept network using classical algebras: Pairs $\mathfrak{A} = \langle A, \Omega \rangle$ and $\mathfrak{B} = \langle B, \Sigma \rangle$ where A and B are sets and Ω and Σ are (finite or finitely generated) sets of operations defined on A and B, respectively, are used to represent the source and the target network. The relation $\langle R, \Psi \rangle$ connecting the target and source networks (called metaphorical relation) is considered again as an algebra defined on the product of $\mathfrak{A}$ and $\mathfrak{B}$

[☆] This paper is an extended and rewritten version of [14]. Additionally [12,13,25,15] are related to this work.

[*] Corresponding author.

E-mail addresses: helmar.gust@uos.de (H. Gust), kkuehnbe@uos.de (K.-U. Kühnberger), ute.schmid@wiai.uni-bamberg.de (U. Schmid).

[1] A large portion of work done in this field is related to literary studies, classical linguistics, and cultural studies. The focus of these disciplines is clearly not to develop a formal theory of metaphors.

[2] Examples for further developments of this account are [3,5].

0304-3975/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2005.11.009

with the following characteristics:

$R \subseteq A \times B$,
$\Psi(n) \subseteq \Omega(n) \times \Sigma(n)$ for all arities n and $\Psi^{-1}(\sigma) \neq \emptyset$ for all $\sigma \in \Sigma$,
$\langle R, \Psi\rangle^{-1}(\langle B, \Sigma\rangle) \subseteq \langle A, \Omega\rangle$ is a concept network and finitely generated.

Indurkhya's framework and its successors (compare, for example, [3,5]) model nicely analogies in string domains, in particular proportional analogies of the form $(A : B) :: (C : ?)$ where A, B, and C are given strings of a certain finite alphabet and the question is which string should be chosen for "?", in order to get the same structural relation between strings A and B and strings C and "?". Notice that in examples of proportional analogies usually it is assumed that the source objects A and B are taken from the same domain as the target objects C and "?". For these types of analogies the theory of anti-unification—which also can be considered as an algebraic approach—was successfully applied (cf. [25,15]).

On the other hand, it is not easy to see how Indurkhya-style frameworks can be applied to domains which cannot be represented in such a straightforward algebraic way. For example, how can this framework be used to model analogies that require full first-order logic for a description of the source and target domains? This type of problems also occurs in the framework of anti-unification, because anti-unification is standardly defined on terms but not on complex logical formulas. A further problem is the establishment of an analogical relation between two *different* domains encoded in two *different* theories. Unfortunately the standard spelled-out examples in algebraic approaches (like the one in [18]) do not reflect this fact. Finally, metaphors often require changes and errors in the establishment of the metaphoric relation. Whereas formal languages provide a precisely specified field of investigations, metaphors are notoriously vague and therefore subject to errors and misunderstandings.

We propose a different approach for modeling metaphoric expressions using *heuristic-driven theory projection* (HDTP) [15]. A basic hypothesis of our account is that analogical reasoning plays an important role in the modeling of metaphors: we think that a large number of metaphors can be reduced to analogical reasoning. Hence, the same frameworks that can be used for analogical reasoning can be used for many metaphors as well.[3] A justification of this claim is the fact that many metaphors can be formulated as analogies:

(1) Gills are the lungs of fish.
(2) Gills are to fish as lungs are to mammals.

Although (2) specifies the relation between *gills* and *lungs* explicitly, whereas (1) does not, the meaning of (1) and (2) are equivalent provided appropriate contexts for (1) and (2) are given. It is straightforward to represent (2) as a proportional analogy of the from $(A : B) :: (C : D)$.

A basic idea of HDTP is to use a structural description of both, the target domain and the source domain, in order to achieve an analysis of metaphors. HDTP can be used for generating such a general structural description. Furthermore, HDTP is a generalization of the theory of anti-unification (AU) (cf. [24]) or the computation of the most specific generalization (cf. [23]). The significant differences between AU and HDTP are summarized as follows:

In contrast to classical AU ...

- ... not only terms are generalized but whole theories of given domains, i.e. conjunctions of complex formulas representing axioms of a theory;
- ... an equational theory E is added to handle equality concerning terms (cf. [2]) *and* equivalences concerning formulas. The latter models equivalences of the form $a \leqslant b \;\leftrightarrow\; b \geqslant a$.
- ... HDTP is heuristic-driven: the task to find an generalizations is governed by heuristics implemented in the algorithm HDTP-A.

The remainder of this article is structured as follows: In Section 2, we will discuss similarities and differences between analogies and metaphors. A development of the basic ideas of HDTP together with a spelled-out example of a predictive analogy (Rutherford analogy) will be discussed in Section 3. In Section 4, we will examine the details of HDTP from a syntactic, an algorithmic, and a semantic perspective making the similarities but also the differences to anti-unification precise. In Section 5, we will apply HDTP to metaphors in natural language. Finally, we will conclude this article in Section 6.

[3] An independent support for this claim was recently provided in [10].

2. Metaphors and analogies

2.1. Similarities between metaphors and analogies

Metaphors and analogies occur in a large variety of domains, as well as in quite different forms. Therefore, we restrict our attention to the analysis of metaphors and analogies to computational traditions in artificial intelligence, linguistics, and cognitive science. In order to classify certain aspects and properties of analogies, three types of analogies can be distinguished (cf. [18,25]): First, *proportional analogies* have the general form $(A : B) :: (C : ?)$. These analogies were studied in the domain of natural language (compare (2)), with respect to geometric figures (cf. [6,22]), and in string domains (cf. [17]). Algebraic accounts can be used to characterize proportional analogies (cf. [20]) and to model applications of them (cf. [18]). Second, *predictive analogies* explain a new domain (target) by specifying similarities with a given domain (source), i.e. by transferring information from the source to the target (cf. [9,16]). Examples are metaphoric expressions in natural language as well as more complex conceptualizations of physical correlations between seemingly different domains (compare Section 3 for the Rutherford analogy and [12] for the well-known heat-flow example). Third, *analogical problem solving* can be used to solve a problem by transferring a solution from a well-known domain to an unknown domain. An example is the usage of a LISP program for developing new program code (cf. [1]). It is neither claimed that this classification of analogies is complete, nor that it is the only possible one. Rather such a classification can be useful to specify different properties of analogies.

It is often mentioned that metaphors are strongly connected to analogies (cf. [10,18]). The classification above makes clear that certain metaphors can be considered as proportional analogies and predictive analogies. Dependent on the context one and the same metaphor can be proportional or predictive.

(3) Electrons are the planets of the atom.

Given a situation in which a teacher is lecturing students elementary atom physics, (3) can be interpreted as a predictive analogy: the students learn a new conceptualization of the atom. On the other hand, for scientist (3) can be simply interpreted as a (historically important) proportional analogy.

2.2. Differences between metaphors and analogies

Assume that the metaphoric expression (3) is given establishing an analogy between electrons and planets. In comparison to predictive analogies in naive physics, the metaphoric example (3) is both, simpler and more complicated:

- It is simpler because several, sometimes complicated, laws of physics need not to be formalized in order to get a meaning of the metaphor: in many relevant contexts, a metaphorical relation establishes a simple analogy between particular properties of the involved concepts. Hence, metaphors require just some transferred facts from the source to the target.
- It is more complicated because there is no clear correct or incorrect way of modeling. Dependent on the context, the usage, the intention of the speaker, the background knowledge of the hearer, etc. (3) can mean many different things, whereas in physics, a predictive analogy either *is* in accordance to a given conceptualization of the laws of physics or *is not*.

The two mentioned aspects have certain consequences for our representation. In particular, the modeling of predictive analogies as described in [13,25] cannot be applied without any restrictions to metaphoric expressions. Whereas in the physical realm there are elaborated theories that govern analogical reasoning [4] often no such spelled-out theories are available for metaphoric expressions. For example, the concept of a planet for a physicist is clearly embedded in a physical theory where certain laws govern the behavior of such an entity in a solar system. On the other hand, for a non-specialized native speaker of natural language, the concept of a planet is less clear. The speaker knows examples (like Earth), perhaps she knows that planets occur always together with a sun, the corresponding system forms a central body system, and perhaps she knows that planets usually revolve around this sun. But this seems to be the maximal set of facts that can be assumed concerning the assumptions that govern metaphorical expressions.

[4] Although those theories are considered as qualitative and essentially causal in nature, they give a large amount of information about the involved domains.

3. Using theory projection for predictive analogies

3.1. The Rutherford analogy

Qualitative physics is an interesting case for analogical reasoning because it shows interesting productive aspects of cognition. Fig. 1 gives an analysis of the well-known Rutherford metaphor (3). According to [9], a domain is represented as a structure (a graph) with objects (such as *sun*, *planet-i*), attributes (such as *yellow*(*sun*)), and relations (such as *attracts*(*sun,planet-i*)). Besides the first-order relations, which are defined on objects, there is a second-order relation, defined on relations:

cause(attracts(sun,planet-i), revolves_around(planet-i,sun)).

All entities are represented as nodes in the graph. The arcs represent relations between entities, in other words, their roles (e.g. *S* for *subject* or *O* for *object*).

Analogical reasoning is modeled as a structure preserving partial mapping μ from the source to the target. According to Gentner's principle of systematicity (cf. [9]), mappings of larger structures are preferred. With respect to (3), mapping *sun* to *nucleus* and *planet-i* to *electron-i* results in a large structural congruence between both domains. Because each object can carry over nodes from the source to the target to which it is connected, the causal explanation why a planet revolves around the sun is transferred to the target domain, resulting in the inference that an electron revolves around the nucleus because it is attracted by it. Notice that in the structure mapping engine (SME) in [7] the nodes representing relations (including attributes as 1-ary relations) must be named identically in the source and the target domain forcing μ to be the identity on relations. The systematicity principle translates to the well-known problem of finding the greatest common subgraph of two graphs (cf. [19]).

We think that the roughly described model has certain conceptual deficiencies:

- The whole mapping seems to be rather trivial, because the causal explanation is built into the modeling: *revolves_around*(*planet-i,sun*) and *revolves_around*(*electron-i,nucleus*) are explicitly given in both conceptualizations and guide crucially the transfer mapping. Therefore, metaphor (3) can be used to describe a situation where the hearer of the utterance does not need to perform productive inferences or transfers. Particularly, learning does not seem to play a role. But this conceptualization is not appropriate to model the creative aspects of analogies and metaphors. On the other hand, the transfer of new information from the source to the target, for example, the establishment of a new concept in the target is not possible, although this seems to be crucial for many types of analogies (cf. [18]).
- The explanation is physically wrong, because the two involved *attracts* relations identified in the SME model are just things that are not identical. They correspond to different forces, namely gravity and Coulomb, respectively.
- It is questionable whether Fig. 1 is a natural description of how analogies are used in physics in the described case. What is measurable on the atom side is that the nucleus is heavier than the electron, that the electron has a negative electric charge whereas the nucleus has a positive electric charge, and that the classical plum pudding model (cf. [26,11]) of the atom contradicts experimental experience, namely the scattering experiments by Rutherford. It is not measurable that the electron revolves around the nucleus rather this should be the inference from the data above.
- It is not clear what the semantics of the described representation should be. A match of relations that are similarly labeled does not involve any kind of semantics. Rather it seems to be a match of syntactic strings.

Whereas for certain situations the discussed modeling seems to be an intuitively plausible and correct representation, this does not seem to be true for the general case. In particular, the described representation does not provide a solution for the problem how productive transfers of concepts can be modeled.

3.2. The modeling of the domains

We want to represent the situation where a conceptualization of the solar system can be used to get a new conceptualization of the Rutherford atom model. The solar system is conceptualized using a set of axioms A_S which in turn induces a theory Th_S as depicted in Fig. 2: *planet* and *sun* are considered to be objects. Certain observable properties (or features) are measurable with respect to these objects by performing experiments: the mass of an object, the distance between two objects, and a force between two objects, called gravity, as well as the centrifugal force between two objects—provided an object o_1 is following a circular path around an object o_2. Additionally, certain

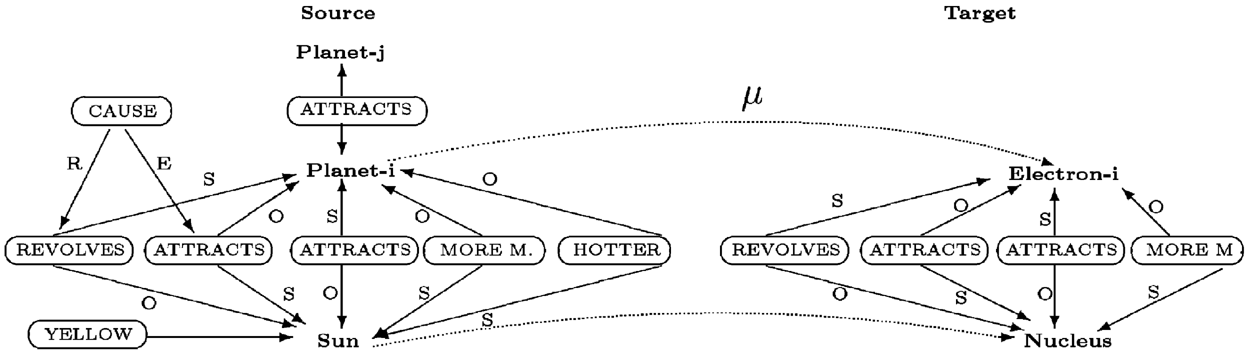

Fig. 1. Structure-mapping for the Rutherford analogy (cf. [9, Fig. 1]).

types
real, object, time

entities
planet : object
sun : object

functions
observable mass: $object \times time \longrightarrow real \times \{kg\}$
observable dist: $object \times object \times time \longrightarrow real \times \{m\}$
observable gravity: $object \times object \times time \longrightarrow real \times \{N\}$
observable centrifugal: $object \times object \times time \longrightarrow real \times \{N\}$

facts
revolves_around(planet, sun)
$mass(sun) > mass(planet)$
$\forall t : time : gravity(planet,sun,t) > 0$
$\forall t : time : dist(planet, sun,t) > 0$

laws
$\forall t : time, o_1 : object, o_2 : object :$
$dist(o_1, o_2, t) > 0 \wedge$
$gravity(o_1, o_2, t) > 0$
$\longrightarrow$
$\exists force : force(o_1, o_2, t) < 0 \wedge$
$force(o_1, o_2, t) = centrifugal(o_1, o_2, t)$

$\forall t : time, o_1 : object, o_2 : object :$
$dist(o_1, o_2, t) > 0 \wedge$
$centrifugal(o_1, o_2, t) < 0$
$\longrightarrow$
$revolves\_around(o_1, o_2)$

Fig. 2. Modeling the physics of a solar system using a theory Th_S.

facts and laws about objects ensure that regularities are given in the source conceptualization governing the behavior of the system.

The atom model is given by a set of axioms A_T (compare Fig. 3). Again these axioms induce a theory Th_T about atoms. The conceptualization of the atom does not contain as much information as the conceptualization of the solar system, because otherwise the establishment of a creative analogy would not be necessary. As objects *electron* and *nucleus* are given as distinct objects. Observable properties are the electric charge of objects as well as masses of objects. Additionally we assume that the Coulomb force between two objects can be measured. Concerning facts governing electron and nucleus, we presuppose that the electron as well as the nucleus have a mass and an electric charge. The latter is the reason why there is a Coulomb force attracting the two objects. Notice that gravity as well as the Coulomb force have the same direction here, i.e. both forces attract electrons and the nucleus (represented by *gravity* (*electron, nucleus, t*) $>$ 0 and *coulomb* (*electron, nucleus, t*) $>$ 0). As long as we are interested in a qualitative analysis of the atom model, it is sufficient to consider only one force, namely the one with the grater magnitude, i.e. the Coulomb force. Last but not least, we are able to perform experiments, in order to test whether analogical transfers yield experimentally valid results. One of these experiments is essentially an abstract representation of the Rutherford experiment, i.e. an experiment that shows that electrons and nucleus have a distance from each other greater than 0.

It is worth noticing that the predicate *revolves_around* has no corresponding predicate in the target domain. Simply transferring this fact to the target would be possible in principal, but there is no way to test in an experiment whether this predicate applies in the target domain. A better modeling is to give an explanation why these concepts can be used in the target domain. This can be achieved by performing an experiment measuring that $dist(electron, nucleus, t) > 0$ and by applying a general (transferred) law from the source that results in the fact $revolves\_around(electron, nucleus)$.

```
types
    real, object, time
entities
    electron: object
    nucleus : object
functions
    observable mass: object × time → real × {kg}
    observable dist: object × object × time → real × {m}
    observable electric_charge: object → real × {eV}
    observable coulomb: object × object × time → real × {N}

facts
    mass(nucleus) > mass(electron)
    electric_charge(electron) < 0
    electric_charge(nucleus) > 0
    ∀t : time : coulomb(electron,nucleus,t) > 0
experiment
    ∀t : time : dist(electron,nucleus, t) > 0
```

Fig. 3. Modeling the physics of the atom model using a theory Th_T.

3.3. The analogical transfer

We will use an extension of the well-known theory of anti-unification to model predictive analogies in naive physics called HDTP. Classical first-order anti-unification is formally founded on the mathematics of term algebras (cf. [23]). We extend this framework to anti-unify not only terms but whole theories (for a detailed discussion compare Section 4).

In HDTP projections are used to find generalizations of the two theories Th_S and Th_T. In order to introduce some important concepts for HDTP, we will first consider the classical case of term anti-unification: anti-unifying two first-order terms t_1 and t_2 of a given term algebra $Term_\Sigma$ means to construct a third term t and two substitutions Θ_1 and Θ_2 such that $t_1 = t\Theta$ and $t_2 = t\Theta$.

Given an equational theory E and a term algebra $Term_\Sigma$ we can introduce the concept of subsumption: A term t subsumes a term s relative to a given equational theory E if the following equivalence holds:

$$s \leqslant_E t \iff \exists\Theta : E \vdash t\Theta = s.$$

A term t is called an anti-instance of a set of terms T if t subsumes all $t' \in T$. An equational theory E allows to express equalities between term expressions. In a concrete situation usually one is confronted with a whole bunch of (possible) anti-instances. Anti-instances can be interpreted as structural descriptions of the underlying terms, because anti-instances are generalizations of those terms, and together with the substitutions the original terms can be re-established. Taking into account the fact that there are in general many possible anti-instances, it is natural to ask which set of anti-instances is most specific, complete, and minimal (cf. [12,27], and Section 4.1). These anti-instances can be identified with *minimal* structural descriptions of certain objects because, intuitively, they are as informative as necessary (completeness) and they are not more informative than necessary (minimality and specificity). Compare Section 4 for a detailed discussion of these concepts.

The sketched first-order case of anti-unification is relatively simple and straightforward. But for our purposes we need a weak form of second-order anti-unification. Consider the following example of an equivalence where the term on the left side is an expression of equational theory E_1 and the term on the right side is an expression of equational theory E_2:

$$f(h(c, h(a, b))) \leftrightarrow g(h(a, b)).$$

Syntactic anti-unification results in the anti-instance $F(h(X, Y))$ where the substitutions Θ_1 and Θ_2 are given as follows (we use capital letters for variables introduced by the anti-unification process):

$$\begin{aligned}\Theta_1/\Theta_2 : \; & F \mapsto f/g,\\ & X \mapsto c/a,\\ & Y \mapsto h(a, b)/b.\end{aligned}$$

The presented anti-instance is a rather weak type of a second-order anti-unification: it can be embedded into a syntactic first-order anti-unification. Nevertheless there are other anti-instances, e.g. $F(h(a, b))$, requiring stronger types of second-order anti-unification. In this case substitutions are given by

$$\Theta_1/\Theta_2 : F \mapsto \lambda y.(f(h(c, y)))/g.$$

Table 1
Anti-instances of our modeling

Source theory Th_S	Target theory Th_T	Generalized theory Th_G
$mass(s) > mass(p)$	$mass(n) > mass(e)$	$mass(Y) > mass(X)$
$rev\_around(p, s)$	$rev\_around(e, n)$	$rev\_around(X, Y)$
$gravity(p, s, t) > 0$	$coulomb(e, n, t) > 0$	$F(X, Y, t) > 0$
$dist(p, s, t) > 0$	$dist(e, n, t) > 0$	$dist(X, Y, t) > 0$

Table 2
Hypotheses of the target domain

Laws	
$\forall t : time, o_1 : object, o_2 : object :$ $dist(o_1, o_2, t) > 0 \wedge$ $coulomb(o_1, o_2, t) > 0$ $\rightarrow$ $\exists force : force(o_1, o_2, t) < 0 \wedge$ $force(o_1, o_2, t) = centrifugal(o_1, o_2, t)$	$\forall t : time, o_1 : object, o_2 : object :$ $dist(o_1, o_2, t) > 0 \wedge$ $centrifugal(o_1, o_2, t) < 0$ $\rightarrow$ $revolves\_around(o_1, o_2)$

This second-order anti-unification can still be embedded into the first-order case provided an equational theory is available in which $\forall y : f(h(c, y)) = f'(y)$ is provable for some (possibly new) function symbol f'. This results in the simple substitution

$$\Theta_1/\Theta_2 : F \mapsto f'/g$$

We can conclude that a certain subclass of second-order anti-unification can be rewritten as first-order E-anti-unifications. In Section 4.2 we will specify this subclass in a more general setting. With regard to the Rutherford analogy the examined weak type of second-order anti-unification is sufficient to model the analogy in an important respect, but we are not able to model the anti-unification of whole theories yet. This is where theory projection is used to generalize the source and the target domain in order to find a generalization. Theory projection is a heuristic-driven association of complex formulas (not just simple terms) of given theories where the anti-unification process is recursively extended from terms to atomic formulas and formulas (cf. Section 4). The task to find appropriate candidates for generalizations is implemented using the algorithm HDTP-A (cf. Section 4.3). Table 1 summarizes the anti-instances of theory projection relative to the theories given in Figs. 2 and 3 (we use e and n as shortcuts for electron resp. nucleus and p and s for planet resp. sun):

Applying appropriate substitutions to the anti-instances yield again theories Th_S and Th_T. The corresponding substitutions Θ_1 and Θ_2 for the anti-unification such that $Source = Th_G\Theta_1$ and $Target = Th_G\Theta_2$ are given by

$$\begin{aligned}\Theta_1/\Theta_2 :\ & X \mapsto planet/electron,\\ & Y \mapsto sun/nucleus,\\ & F \mapsto gravity/coulomb.\end{aligned}$$

Notice that by transferring the laws of the source domain to the target domain we get hypothetical laws in the target domain as well. These laws are not simply mapped one-to-one to the target but governed by the anti-instances. Table 2 specifies the result of this transfer.

A remark concerning the laws of the source domain: these laws are transferred to the target domain with their respective interpretation. Just because we can apply these laws it is possible to *deduce* that an electron is revolving around a nucleus, i.e. we can give an explanation why the electron is revolving. This is a prediction, hence a creative or productive aspect, which follows from the analogical reasoning process. [5] The experiment on the target side

[5] If later an experiment shows that this fact can be falsified (as shown by Niels Bohr), then of course the whole analogy must be challenged. In our contemporary atom model quantum physics effects are responsible for the counter force (of the Coulomb force). Hence only a weaker law can be transferred implying a modification of the source domain. But this is not in focus of this work.

(cf. Fig. 3) can be considered as a test procedure for the transfer. Only those transfers are allowed that are consistent with the observable data.

4. The syntax and semantics of HDTP

In this section, we will discuss the formal details of HDTP. We will start our considerations by examining the basic ideas of first-order anti-unification and its extensions to HDTP. Then we will introduce the underlying algorithm HDTP-A. Finally, we will examine the semantics of an analogical relation between source and target computed by HDTP-A.

4.1. First-order anti-unification

A crucial idea of HDTP is the establishment of a generalization of theories quite similar to the theory of first-order anti-unification establishing a generalization of terms. We will consider first-order anti-unification first, and extend this approach in a second step. First-order anti-unification is crucially based on (sorted) term algebras. The following two definitions make the concept of a signature and the concept of a term algebra precise.

Definition 1. A many-sorted signature $\Sigma = \{Sort_\Sigma, Func_\Sigma, Type_\Sigma\}$ is given by a partially ordered set of sorts $Sort_\Sigma$, a set of function symbols $Func_\Sigma$, and a function $Type_\Sigma : Func_\Sigma \rightarrow Cl(Sort_\Sigma)$ where $Cl(Sort_\Sigma)$ is the closure of sorts under products. [6]

Definition 2. Assume a signature Σ is given. The term algebra $Term(\Sigma, V, C)$ relative to an (infinite) set of sorted variables $V = \{x_1 : s_1, x_2 : s_2, \dots\}$ with $s_i \in Cl(Sort_\Sigma)$ and a (finite) set of sorted constants $C = \{a_1 : s_1, \dots, a_n : s_n\}$ with $s_j \in Cl(Sort_\Sigma)$ is defined as the smallest set such that the following conditions hold:

(1) If $x : s \in V$ is given, then $x : s \in Term(\Sigma, V, C)$.
(2) If $a : s \in C$ is given, then $a : s \in Term(\Sigma, V, C)$.
(3) If $f \in Func_\Sigma$, $Type_\Sigma(f) = s_1 \times s_2 \times \dots \times s_n \rightarrow s$, and $\forall i \in \{1, 2, \dots, n\} : Type_\Sigma(t_i) = s_i$ is given, then $f(t_1, \dots, t_n) \in Term(\Sigma, V, C)$ and furthermore $Type_\Sigma(f(t_1, \dots, t_n)) = s$.

In order to simplify the readability of formal expressions we suppress the coding of the corresponding sorts of variables and constants in the following parts if sort restrictions are clear from the context. Given a term algebra $Term(\Sigma, V, C)$ we can define substitutions on terms.

Definition 3. Assume a term algebra $Term(\Sigma, V, C)$ is given. A substitution on terms is a partial function $\Theta : V \rightarrow Term(\Sigma, V, C)$ mapping variables to terms, formally represented by $\Theta : \{x_1 \mapsto t_1, \dots, x_n \mapsto t_n\}$ (provided $x_i \neq x_j$ for $i, j \in \{1, \dots, n\}$ and $i \neq j$). For a given substitution Θ we can define domain, range, and the value of Θ relative to a given variable v as follows:

$$\begin{array}{lcl} dom(\Theta) & = & \{x \mid x \mapsto t \in \Theta\}, \\ range(\Theta) & = & \{t \mid x \mapsto t \in \Theta\}, \\ value(\Theta, v) & = & t \ \text{ if } \ v \mapsto t \in \Theta. \end{array}$$

Given a term algebra *Term*(Σ, V, C) we call the collection of all possible substitutions *SUB*. In the following we will assume that the variables in $domain(\Theta)$ and the variables occurring in $range(\Theta)$ are disjoint. Using a substitution Θ we are able to replace variables by terms. Such substitutions can be extended to complex terms by the introduction of a function *sub* that recursively applies substitutions Θ.

[6] As usual, we will write the type of a function *f* with domain s' and range s as $s' \rightarrow s$ instead of $s' \times s$.

Definition 4. Assume a substitution $\Theta \in SUB$ is given. We recursively define a function $sub : SUB \times Term(\Sigma, V, C) \rightarrow Term(\Sigma, V, C)$ as follows:

$$sub(\Theta, c) = c \text{ if } c \text{ is a constant,}$$
$$sub(\Theta, v) = value(\Theta, v) \text{ if } v \in dom(\Theta),$$
$$sub(\Theta, f(t_1, \ldots t_n)) = f(sub(\Theta, t_1), \ldots, sub(\Theta, t_n)).$$

In order to preserve readability we simply write $t\Theta$ instead of $sub(\Theta, t)$. As already used in Section 3.3 we define a subsumption relation on terms relative to a given equational theory *E* as follows (cf. [2]):

$$s \leqslant_E t \iff \exists \Theta \in SUB : E \vdash t\Theta = s.$$

If $s \leqslant_E t$ holds then we call *t* a generalization of *s*. The subsumption relation of terms allows us to order terms in a hierarchy according to their specificity. Intuitively an anti-instance of a set of terms $T = \{t_1, \ldots, t_n\}$ is a term *a* together with the corresponding substitutions $(a, \{\Theta_1, \ldots, \Theta_n\})$ that subsumes all elements of *T*. Clearly there are in general many possible anti-instances relative to a given set of terms *T*. In order to get a preferred set of anti-instances we need to require certain conditions on this set of anti-instances: it needs to be complete, minimal, and most specific.

Definition 5. Assume a set of terms $T = \{t_1, \ldots, t_n\}$ where $t_i \in Term(\Sigma, V, C)$ is given. We call a set of anti-instances

$$AI = \{(a_1, \{\Theta_{11}, \ldots, \Theta_{1n}\}), (a_2, \{\Theta_{21}, \ldots, \Theta_{2n}\}), \ldots, (a_k, \{\Theta_{k1}, \ldots, \Theta_{kn}\})\}$$

relative to a given equational theory *E* appropriate if the following three conditions are met:

(i) $\forall i \in \{1, \ldots, k\} \forall j \in \{1, \ldots, n\} : \quad E \vdash a_i \Theta_{ij} = t_j$
(ii) $\forall t \in Term(\Sigma, V, C) : (\forall i \in \{1, \ldots, n\} \exists \Theta_i \in SUB : E \vdash t\Theta_i = t_i)$
$\Rightarrow \exists \Theta \in SUB \, \exists m \in \{1, \ldots, k\} : E \vdash t\Theta = a_m$
(iii) $\neg \exists i, j \in \{1, \ldots, k\} \exists \Theta \in SUB : i \neq j \wedge E \vdash a_i \Theta = a_j$

Appropriate sets of anti-instances can be interpreted as follows: the first condition of Definition 5 establishes the constitutive relation between terms and anti-instances. Appropriate sets of anti-instances must be complete in the sense that every term $t \in T$ can be reconstructed using an anti-instance $a \in AI$ with an appropriate substitution. The second condition ensures that appropriate sets of anti-instances are most specific: if a term *t* subsumes all elements of *T*, then there is an element $(a, \{\Theta_1, \ldots, \Theta_n\})$ such that *t* also subsumes *a*. Finally, the third condition means that appropriate anti-instances must be minimal: anti-instances from *AI* form a minimal basis of anti-instances, i.e. they cannot be gained from each other by applying a further substitution. Appropriate sets of anti-instances generalize the concept of most specific anti-instances in first-order anti-unification *without* equational theory *E*: If *E* is empty, Definition 3.1 reduces to the classical first-order anti-unification on terms for which it is shown in [23] that the set of all most specific anti-instances is a singleton.[7]

4.2. Theory projection

In Section 4.1, we examined how to use anti-unification to calculate an anti-instance of given terms. For the task to find generalizations of the source and the target domains we need an extended approach: not only terms need to be anti-unified, but also whole theories encoded as complex formulas in a first-order language need to be generalized.

The underlying hypothesis is that the source domain is considered to be a rich domain with a large amount of facts and laws determining the conceptualization. In contrast to the source, the target is considered to be a meager domain with only a few background assumptions restricting arbitrariness of the conceptualization. The source or target input is given by a set of facts *Fact* (variable-free literals) and set of laws *Law* (quantified formulas). The resulting theory is defined as

$$Th = \{\phi \quad | \quad Fact \cup Law \vDash \phi\}.$$

[7] In the second-order case, where we wish to anti-unify also higher-order objects like functions and relations this claim does not necessarily hold, but in restricted cases there are only finitely many anti-instances (cf. [27] and Section 4.2). In order to find generalizations of two theories Th_S and Th_T precisely this is required.

Given a theory Th_S (for the source domain) and a theory Th_T (for the target domain) the task is to find possible candidates $\phi \in Th_S$ and $\psi \in Th_T$ such that for a generalization:

$$AI = \{(a_1, \{\Theta_{11}, \Theta_{12}\}), \ldots, (a_k, \{\Theta_{k1}, \Theta_{k2}\})\},$$

we get for $i \in \{1, \ldots, k\} : a_i\Theta_{i1} = \phi$ and $a_i\Theta_{i2} = \psi$. This task can be solved as follows: assume two formulas $\phi \in Th_S$ and $\psi \in Th_T$ are given in languages L_S and L_T, respectively. The generalization is calculated by recursively extending anti-unification on terms to anti-unification on formulas ϕ and ψ. Atomic formulas $R(t_1, \ldots, t_n) \in Th_S$ and $R'(t'_1, \ldots, t'_m) \in Th_T$ can be treated similarly as functional terms defined in Definition 4. For example, a substitution on $R(t_1, \ldots, t_n) \in Th_S$ can be defined as follows:

$$sub(\Theta, R(t_1, \ldots, t_n)) = R(sub(\Theta, t_1), \ldots, sub(\Theta, t_n)).$$

In order to define substitutions on complex formulas $\phi \in Th_S$ and $\psi \in Th_T$, we transform ϕ and ψ into an appropriate normal form $NF(\phi)$ and $NF(\psi)$, e.g. CNF. ϕ and ψ can be generalized if $NF(\phi)$ and $NF(\psi)$ are structurally equal. Then the generalization is simply performed by anti-unifying the corresponding literals.

It remains to show that the described second-order aspects and the usage of an equational theory (cf. Section 3) do not result in infinitely many generalizations in a computation. In order to do this, we define a subclass of first-order E-generalizations that fits our purposes. In the following, we denote with $st(a)$ the set of all subterms occurring in a where a is a term or a formula. Furthermore, a sequence $\{t'_1, \ldots, t'_k\} \subset X$ (where X is a set of terms) is called admissible relative to X, if the variables occurring in $\{t'_1, \ldots, t'_k\}$ are exactly the variables occurring in X.

Definition 6. Assume a term algebra $Term(\Sigma, V, C)$, an equational theory E, and a (complex) term $f(t_1, \ldots, t_n) \in Term(\Sigma, V, C)$ is given. An expansion of Σ and E relative to $f(t_1, \ldots, t_n)$ is defined as the smallest expansion of Σ and E such that the following condition hold:

> For each admissible sequence $\{t'_1, \ldots, t'_k\}$ relative to $st(f(t_1, \ldots, t_n))$ a new function symbol h and an equation $\forall(h(t'_1, \ldots, t'_k) = f(t_1, \ldots, t_n))$ are contained in Σ and E, respectively.

The intuition of Definition 6 is that the added equations describe permuting arguments, removing arguments, and substituting arguments by their subterms. Exactly those second-order problems which are based on these operations can be reduced to the first-order case with equational theory E. A problem that cannot be reduced by this method is, for example, argument introduction. Obviously an expansion of Σ and E relative to $f(t_1, \ldots, t_n)$ results in adding a finite number of equations in solved form.

Definition 7. Assume a set of axioms A is given. An expansion of A relative to a conjunction of formulas $P_1 \wedge \ldots \wedge P_n$ is defined as the smallest expansion such that it holds:

> For each admissible sequence $\{t'_1, \ldots, t'_k\}$ relative to $st(P_1 \wedge \ldots \wedge P_n)$ a new predicate symbol H and a formula $\forall(H(t'_1, \ldots, t'_k) \leftrightarrow P_1 \wedge \ldots \wedge P_n)$ are added to A.

Given a set of axioms A we can use Definition 7 to add a finite number of new formulas, i.e. definitions of new predicates, because there are only finitely many subformulas in A. Then Definition 6 can be used to add a finite number of definitions of new functions, because there are only finitely many subterms in A. These new predicates and functions are now available for the generalization process by first-order anti-unification. Using such newly introduced functions or predicates in first-order anti-unification steps correspond to second-order anti-unification using the original theories: The right-hand sides of the equations or equivalences, respectively, can be used instead of the left-hand sides, but now with the explicit structure changing operations used in the substitutions, which are now second order.[8] A trivial consequence is that for a given term t there are only finitely many s such that $E \vdash t = s$. With respect to Definition 5 this means: if E contains only equations in solved form, then there are only finitely many anti-instances for two terms t and t'. This can be extended to generalizations of atomic formulas. Consequently, a class of second-order generalizations

[8] The algorithmic solution computes the needed new function and predicate definitions only on demand, namely if they are needed to reduce a second-order anti-unification step to the first-order case.

```
$T$ = axioms of the target domain sorted by a heuristics $h$
$S$ = axioms of the source domain
$G$ = empty list of axioms of generalized theory
$\Theta_1 = \Theta_2$ = empty substitution
$Th_T^{A_h} = Th_T$
FOR $\phi \in T$
    $T = normal\_form(T)$
    SELECT $\psi \in S$
        $\psi = normal\_form(\psi)$
        IF not $same\_structure(\phi, \psi)$ REJECT
        SELECT $(\xi, \Theta_1, \Theta_2) \in anti\_instances(\phi, \psi, \Theta_1, \Theta_2)$
            WITH $\xi$ best according to a heuristics $h'$
    IF $h'(\xi) >$ a given threshold
        ADD $\xi$ to $G$
        ADD $\xi\Theta_2$ to $T_T^{A_h}$
        REMOVE $\psi$ from $S$
    ELSE FAIL
END FOR
FOR $\psi \in S$
    $\phi = transfer(\psi, \Theta_1, \Theta_2)$
    IF $T_T^{A_h} \vdash \neg\phi$ CONTINUE
    IF $oracle(\phi) = FALSE$ CONTINUE
    ADD $\phi$ to $Th_T^{A_h}$
    ADD $generalize(\psi, \Theta_1)$ to $G$

END FOR
```

Fig. 4. The algorithm HDTP-A generalizing two theories Th_S and Th_T.

can be reduced to the first-order case (with equational theory E). All problems of the type discussed in this article can be solved with this method.

To summarize the theory so far: the underlying idea is to compute possible generalizations stage-by-stage together with their substitutions governed by a certain heuristics. The restriction to the described subclass of second-order generalizations ensures finiteness of this approach. After the generalization as many source facts and laws as possible are transferred to the target provided the transfer does not result in an inconsistency. The process of the generalization and the transfer of facts and laws we call *theory projection*. The motivation to transfer as many axioms as possible to the target domain is related to the systematicity principle of the SME model (compare Section 3.1) and the metaphorical relation in [18]. The assumption is that the larger the number of (consistent) correspondences is that can be established the more (psychologically) preferred is the interpretation.

4.3. The algorithmic generalization of theories

In this subsection, we will present the algorithm HDTP-A computing generalizations together with their corresponding substitutions provided a conceptualization of the source domain and the target domain is given.[9] The process can roughly be described as depicted in Fig. 4.

The input is given by conceptualizations S inducing a theory Th_S of the source domain, which is coded in a language L_S, and by the conceptualization T inducing a theory Th_T of the target domain, which is coded in a language L_T. The output of the algorithm is a set of generalized axioms G inducing a theory Th_G which is coded in the language $L^+_{S\oplus T}$ generalizing the source domain and the target domain. $L^+_{S\oplus T}$ contains predicates and functions from source and target and a set of new variable symbols. The algorithm chooses an axiom from the target domain T governed by a heuristics h and searches for an axiom from the source domain S to generalize both. If a generalization is found, the resulting generalized axiom is added to the generalized theory. This process is recursively applied until all axioms in the target are generalized.

Finally, remaining axioms from the source can be transferred to the target and the generalized theory, using the computed substitutions as long as the transferred axioms are consistent with the extended target theory $Th_T^{A_h}$. In order

[9] The algorithm described in this subsection is based on the implementation in [12].

to check whether the transferred axioms hold in the intended interpretation of the target, an oracle or test procedure is used representing an experiment. If the transferred axioms do not pass the test, the transfer will be rejected. The described transfer is an important aspect of the algorithm because it allows the introduction of new concepts on the target side. As an example compare [25] where it is shown how to generate a concept like heat-flow on the target side although this concept is not present in the original input.

The algorithm allows the implementation of a variety of different heuristics concerning the selection of axioms from the target domain and the selection of potential candidates for generalization in the source domain. Some possible heuristics h that can be used for choosing axioms are summarized in the following list (which is obviously not complete):

(1) Select axioms from the target domain first.
Remark: The target is considered to be less rich than the source and should be completely covered by the analogy, whereas (usually) not all source axioms will be generalizable.
(2) Select simple axioms first relative to the number of embedded relations, the arity of embedded relations, the number of logical connectives, etc.
Remark: The search space to find corresponding axioms in the source is reduced and the number of possible generalizations is minimized.
(3) Select axioms that maximize the number of shared terms with already generalized axioms.
Remark: This heuristics minimizes the need of additional substitutions.

All these heuristics can always be used and do not change the set of solutions. Only the order by which they are computed is effected. Furthermore, they are independent from each other and simple to compute. We need a further heuristics h' to select an appropriate generalization from the computed generalizations:

(1) Select anti-instances with minimal length of substitutions first.
Remark: The minimization of the global number of substitutions can be approximated by choosing this local heuristics.
(2) Select anti-instances with a minimal number of second-order objects in the substitutions first. [10]
Remark: Second-order substitutions result in a structurally stronger modification.

These heuristics can always be used, but there may be a trade-off between (1) and (2). Again these heuristics do not change the set of solutions. Since the number of alternative solutions even if finite can be rather large, it is important to compute *good* solutions as early as possible. It is not simple to specify what *good* should mean in this context. Usually it is assumed that analogies and metaphors cannot be right or wrong but rather more or less cognitively preferred, i.e. more or less *good*. A cognitive hypothesis is that simpler solutions are (usually) preferred compared to more complex solutions. In this context, simpler means less transformation operations or operations leading to less structural changes. [11]

4.4. Semantics

In this subsection, a semantics for analogical transfer will be established. The idea is to establish a bisimulation-like relation called analogical relation between source and target. [12] This is mirrored by the generalization and the corresponding substitutions of the two input domains source and target.

Definition 8. The language L_S of the source domain and the language L_T of the target domain are standard many-sorted first-order predicate logic languages relative to a given term algebra $Term(\Sigma, V, C)$. Suppressing the sortal

[10] This corresponds to minimizing the use of equations and equivalences introduced by Definitions 6 and 7.

[11] A similar idea is used in [4,5] to compute preferred *Gestalts* of certain similarity patterns. Some empirical evidence is given for minimizing structural changes.

[12] A definition of *cognitive* relation is already given in [18] that is quite close to the classical definition of a bisimulation: structural identity without being isomorphic. Unfortunately, the author does not take into account the processes that are involved in order to establish such a bisimulation, namely the processes having to do with finding generalizations of the two given domains. We will give a precise characterization of the relation between source and target in semantic terms.

restrictions the following sub-languages can be specified:

Terms $t ::= x \mid c \mid f^n(t_1, \dots, t_n)$.
Logical constants $l ::= \wedge \mid \vee \mid \rightarrow \mid \leftrightarrow \mid \neg \mid \forall \mid \exists \mid =$.
Atomic (well-formed) formulas $\alpha ::= t = s \mid R^n(t_1, \dots, t_n)$,
Well-formed formulas $\phi ::= \alpha \mid \phi \wedge \psi \mid \phi \vee \psi \mid \neg\phi \mid \phi \rightarrow \alpha \mid \phi \leftrightarrow \psi \mid \forall x \phi \mid \exists x \phi$.

We suppressed the coding of the sorts $Sort_\Sigma$ in the definition of terms and formulas due to simplify readability. Clearly our present Prolog implementation is based on a language with less expressive power, because of the restriction of Prolog to Horn clauses. For the theoretical investigations concerning formal properties of generalized theories we can drop this restriction and consider as base language full first-order predicate logic.

Theories of the source and target domains are given by axioms specifying facts that hold in a particular domain and rules that can be used to deduce new facts in the domain. The axioms do not specify a complete calculus, but rather an underspecified description of the domain.

Definition 9. A theory *Th* of a language *L* is specified as a consistent and finite set of well-formed formulas of *L* (axioms) of the following form:

Facts: $\alpha ::= R^n(t_1, \dots t_n) \mid t = s \mid \alpha \wedge \beta \mid \alpha \vee \beta$,
Laws: $\phi ::= \phi \rightarrow \psi \mid \forall x \phi \mid \exists x \phi$.

We assume that partial standard first-order predicate logic models with equality are given for the input of source and target domains.

Definition 10. Given a theory *Th* and a model $\mathfrak{M} = \langle D, I \rangle$, truth of formulas is defined as usual: [13]

$$\begin{array}{lll}
\mathfrak{M} \models t = s & \text{iff} & I^+(t) = I^+(s), \\
\mathfrak{M} \models R(t_1, \dots t_n) & \text{iff} & \langle I^+(t_1), \dots, I^+(t_n) \rangle \in R, \\
\mathfrak{M} \models \alpha \wedge \beta & \text{iff} & \mathfrak{M} \models \alpha \text{ and } \mathfrak{M} \models \beta, \\
\mathfrak{M} \models \alpha \vee \beta & \text{iff} & \mathfrak{M} \models \alpha \text{ or } \mathfrak{M} \models \beta, \\
\mathfrak{M} \models \phi \rightarrow \psi & \text{iff} & \mathfrak{M} \not\models \phi \text{ or } \mathfrak{M} \models \psi, \\
\mathfrak{M} \models \forall x \phi & \text{iff} & \text{for all } m \in D : \mathfrak{M} \models \phi(m), \\
\mathfrak{M} \models \exists x \phi & \text{iff} & \text{for some } m \in D : \mathfrak{M} \models \phi(m).
\end{array}$$

The algorithm as described in Section 4.3 takes as input two theories Th_S and Th_T for the source domain and the target domain, respectively. The intuitive idea of the semantics of a metaphor is the establishment of an analogical relation between source and target corresponding to a (psychologically) preferred interpretation of the metaphor. [14] In general, such an analogical relation cannot be established directly, because the two input theories can be quite incoherent and the result would be rather arbitrary. A better approach is to find an analogical relation induced by the algorithm. Let Th_S and Th_T be two input theories (source theory and target theory, respectively). The result of the algorithm HDTP-A_h with heuristics h applied to Th_S and Th_T is a generalized theory Th_G together with two theories $Th_S^{A_h}$ and $Th_T^{A_h}$, which are byproducts of the algorithm. Particularly, the computation of normal forms (cf. Section 4.2) and applications of the equational theory result in a modification of the original input theories Th_S and Th_T. Now it is possible to define the concept of an analogical relation between two theories $Th_S^{A_h}$ and $Th_T^{A_h}$. In the following definition we write A_h instead of HDTP-A$_h$ in order to increase the readability of the formulas.

[13] I^+ denotes the homomorphic extension of *I* to terms.

[14] Notice that an interpretation of a metaphor cannot be right or wrong. Rather it is the case that certain interpretations are psychologically more of less preferred. Examples of empirical results concerning such preferred interpretations of metaphors can be found in [21,5,8].

$$\begin{tikzcd}
\mathfrak{M}_S & R^\star \arrow[l, "\pi_l^{R^\star}"'] \arrow[r, "\pi_r^{R^\star}"] & \mathfrak{M}_T \\
Th_S^{A_h} \oplus Terms_S \arrow[u, "I_S^+"] & R \arrow[l, "\pi_l^R"] \arrow[r, "\pi_r^R"'] & Th_T^{A_h} \oplus Term_T \arrow[u, "I_T^+"']
\end{tikzcd}$$

Fig. 5. Diagrammatic representation of an analogical relation between two theories $Th_S^{A_h} \oplus Terms_S$ and $Th_T^{A_h} \oplus Term_T$ and the induced relation $R^\star$ on the model theoretic level.

Definition 11. Assume two theories $Th_S^{A_h}$ and $Th_T^{A_h}$ together with corresponding models $\mathfrak{M}_S$ for $Th_S^{A_h}$ and $\mathfrak{M}_T$ for $Th_T^{A_h}$ and a coproduct (disjoint union) operation $\oplus$ are given. An analogical relation

$$R \subseteq (Th_S^{A_h} \times Th_T^{A_h}) \oplus (Term(\Sigma_S, V_S, C_S) \times Term(\Sigma_T, V_T, C_T))$$

of theories is a set of pairs $\langle x, y \rangle$ such that it holds:

(i) If $\langle \phi, \psi \rangle$ is a pair of formulas and $\langle \phi, \psi \rangle \in R$, then there exists $g \in Th_G^{A_h}$ such that $(g, \{\Theta_1, \Theta_2\})$ is an anti-instance of ϕ and ψ and

$$Th_T^{A_h} \cup E_T \vdash g\Theta_2 \leftrightarrow \psi \quad \text{and} \quad Th_S^{A_h} \cup E_S \vdash g\Theta_1 \leftrightarrow \phi.$$

(ii) If $\langle t, t' \rangle$ is a pair of terms and $\langle t, t' \rangle \in R$, then there exists $g \in Term_G$ such that $(g, \{\Theta_1, \Theta_2\})$ is an anti-instance of t and t' and

$$E_T \vdash g\Theta_2 = t' \quad \text{and} \quad E_S \vdash g\Theta_1 = t.$$

(iii) There exists a model $\mathfrak{M}_S$ such that $\mathfrak{M}_S \models Th_S^{A_h}$ if and only if there exists a model $\mathfrak{M}_T$ such that $\mathfrak{M}_T \models Th_T^{A_h}$.

The analogical relation R induces a relation $R^\star \subseteq M_S \times M_T$ that respects functions $f \in Func_{\Sigma_{Th_S}}$ and $g \in Func_{\Sigma_{Th_T}}$. In other words, for all terms $t_S \in Term(\Sigma_S, V_S, C_S)$ and $t_T \in Term(\Sigma_T, V_T, C_T)$: if $\langle t_S, t_T \rangle \in R$ then $\langle I_S^+(t_S), I_T^+(t_T) \rangle \in R^\star$. This means that for a given R, the reference determined by models $\mathfrak{M}_S$ and $\mathfrak{M}_T$ coincide not only on the formula level but also on the term level.

Another way to represent the idea behind the concept of an analogical relation can be given by Fig. 5. The relations R and $R^\star$ represent the analogical relation between the source and target theories and the induced relation between the models, respectively. The relation R between theories $Th_S^{A_h} \oplus Terms_S$ and $Th_T^{A_h} \oplus Term_T$ is induced by the algorithm associating terms and formulas of the two theories.[15] The interpretation functions I_S^+ and I_T^+ shift the syntactic association to the semantic level, namely to a relation $R^\star$ between two models. The consistency assumptions for the underlying models ensure that there is an analogical relation between the two domains. Notice that model theoretically, there are many models for $Th_T^{A_h}$ and $Th_S^{A_h}$ provided the underlying axioms are consistent. An analogical relation models these ideas: based on the association of facts and laws of the two domains, it is required that every model making facts and laws in the source true there is a model of the target domain making the corresponding theory true.

We can now examine the complete diagram for heuristic-driven theory projection (Fig. 6) step-by-step.

The core of the diagram: The core is similar to the diagram in Fig. 5, specifying relations R' and $R^\star$ simulating the syntactic correspondence between the conceptualization of source and target (after the algorithm generated associated pairs of facts and laws) and the semantic association between interpretations on the models. Additionally we added A_G—the axioms of the generalized theory of the source and target—together with substitutions Θ_1 and Θ_2 allowing to

[15] $Terms_S$ and $Term_T$ denote the terms occurring in $Th_S^{A_h}$ and $Th_T^{A_h}$, respectively, and terms that can be generated by the underlying equational theories.

$$\begin{tikzcd}
 & & \mathfrak{M}_G & & & \\
 & \mathfrak{M}_S \arrow[ur, "Pr_S"] & R^{\star} \arrow[l, "\pi_l^{R^\star}"'] \arrow[r, "\pi_r^{R^\star}"] & \mathfrak{M}_T \arrow[ul, "Pr_T"'] & & \mathfrak{M}_T^{exp} \arrow[ll, "in_{test}"'] \\
 & & A_G^{A_h} \oplus Term_G \arrow[uu, bend right, "I_G^+ \upharpoonright A_G \oplus Term_G"'] \arrow[dl, "\Theta_1"'] \arrow[dr, "\Theta_2"] & & & \\
Th_S^{A_h} \oplus Terms_S \arrow[uur, "I_S^+"] \arrow[d, "\langle in_{Th_S}, id\rangle"] & A_S^{A_h} \oplus Terms_S \arrow[l, "g_S^{A_h}"'] \arrow[uu, "I_S^+ \upharpoonright A_S^{A_h} \oplus Terms_S"] \arrow[d, "\langle in_{A_S}, id\rangle"] & R \arrow[l, "\pi_l^R"] \arrow[r, "\pi_r^R"'] & A_T^{A_h} \oplus Term_T \arrow[uu, "I_T^+ \upharpoonright A_T^{A_h} \oplus Term_T"] \arrow[r, "g_T^{A_h}"] & Th_T^{A_h} \oplus Term_T \arrow[uul, "I_T^+"'] & Th_{exp} \arrow[l, "in_{exp}"] \arrow[uu, "I_T^+ \upharpoonright exp"'] \\
Th_S \oplus Terms_S & A_S \oplus Terms_S \arrow[l, "\langle g_S, id\rangle"] & & A_T \arrow[u, "in_{A_T}"] \arrow[r, "g_T"'] & Th_T \arrow[u, "in_{Th_T}"'] &
\end{tikzcd}$$

Fig. 6. Diagrammatic representation of theory projection.

regain the source and target from the generalized theory. The left and right lower parts of Fig. 6 represent the (possible) changes of the source and target domains induced by the heuristic association of the algorithm.

Here are some remarks concerning the target side: the input, given by A_T, is a meager conceptualization of the target domain. By possible transfers of facts and laws from the source to the target and possible equivalence transformations of expressions relative to the equational theory E_T (triggered by the algorithm), $A_T^{A_h} \oplus Term_T$ is generated using an injection function (modulo equivalences relative to E_T). The input A_T as well as the modified input $A_T^{A_h} \oplus Term_T$ corresponds to theories Th_T and $Th^{A_h} \oplus Term_T$. The input A_T with its corresponding theory Th_T can be injectively mapped into $A_T^{A_h} \oplus Term_T$ and $Th^{A_h} \oplus Term_T$. Notice that in the source domain the arrows have the inverse direction, because a rich theory about a domain will be restricted to those facts and laws that can be consistently matched with or consistently transferred to the target.

The analogical relation: In Definition 11, an analogical relation between source and target was defined as the following set:

$$R \subseteq (Th_S^{A_h} \times Th_T^{A_h}) \oplus (Term(\Sigma_S, V_S, C_S) \times Term(\Sigma_T, V_T, C_T)).$$

This relation is slightly modified in Fig. 6: R is restricted to the facts, laws, and terms that are induced by the input. The resulting relation R' is specified as follows:

$$R' \subseteq (A_S^{A_h} \times A_T^{A_h}) \oplus (Terms_S \times Term_T).$$

Functions $g_S^{A_h}$ and $g_T^{A_h}$ guarantee that these facts, laws, and terms can be generalized to theories $Th_S^{A_h} \oplus Terms_S$ and $Th_T^{A_h} \oplus Term_T$, respectively.

Experiments/Tests: The test procedure of the algorithm is represented by a theory Th_{exp} specifying the experiments in the target domain. Notice that these experiments are restricted by the condition that every functional expression must correspond to an observable. Mapping the occurring theories to a model theoretic level using an interpretation function I^+ (relative to S, $A_S^{A_h} \oplus Terms_S$, etc.) we can simulate the relational constraints on the syntactic level on the model theoretic level as well. Experiments ensure that the analogical relation is based and tested on laws of nature in the case of qualitative physics. A more general context would require other tests which can be based, for example, on interactions with other agents, on abstract calculations and the like.

The generalized theory and the generalized model: In Fig. 6, $I_G^+ \upharpoonright A_G \oplus Term_G : A_G \oplus Term_G \longrightarrow \mathfrak{M}_G$ is induced by the algorithm HDTP-A and the interpretations I_S^+ and I_T^+. The corresponding homomorphic extension

$I_G^+ : Th_G^{A_h} \oplus Term_G \longrightarrow \mathfrak{M}_G$ is not represented in Fig. 6 due to readability reasons. Furthermore, there are projection mappings $Pr_S : \mathfrak{M}_S \rightarrow \mathfrak{M}_G$ and $Pr_T : \mathfrak{M}_T \rightarrow \mathfrak{M}_G$.

5. The modeling of metaphors

Based on the machinery developed in Section 3, we can apply HDTP to metaphoric expressions. The structure of this section is as follows: first, we will examine certain types of metaphors. Second, we will roughly discuss differences between the Rutherford analogy and the corresponding metaphorical expression. Third, we will model more general metaphoric expressions.

5.1. Types of metaphors

There are many different classifications of metaphors. Additionally there are many closely related concepts to metaphors in linguistic theories like idioms, forms of irony, similes, etc. In the following, we will discuss roughly three types of metaphors. The first type of metaphors connects noun phrases with a form of *to be*. Examples of this type of metaphors are the following ones:

(4)(i) Electrons are the planets of the atom.
(ii) Electricity is the water of an electric circuit.
(iii) Lawyers are sharks.
(iv) Juliet is the sun.

Another type of metaphors assigns a particular attribute to a concept (noun phrase) that typically would not be considered as applicable in a conventional interpretation. Reasons for the non-applicability of the attribute are often sort problems: A liquid can have a color, it can be oily or transparent, it can flow, be cold, or be warm and the like, but usually it cannot be soft like in (5)(i).

(5)(i) A soft wine.
(ii) A cold warrior.
(iii) A warm acknowledgment.

Whereas examples like the ones in (4) and (5) are as simple as possible, this is not in general true for metaphors occurring in poetic contexts. Examples like the often cited poem *Fog* by Carl Sandburg where *fog* is metaphorically correlated to *cats* are more complicated (quoted according to [18]):

The fog comes
on little cat feet.

It sits looking
over harbor and city
on silent haunches
and then moves on.

Another type of expressions which is related to metaphors are idioms. They are usually considered as lexicalized metaphors, i.e. as metaphors that are already transformed into conventionally interpreted expressions. We will not consider language change phenomena like idioms in this paper. Furthermore, examples of irony and similes will not be considered in this section. Last but not least, we do not analyze certain types of unconventional usage of concepts in natural languages as metaphorical. For example, if Tom says to Jim that his fish is in the living room, referring to the wooden fish he bought in Singapore, then this non-conventional usage of the concept fish is not investigated here.

5.2. The Rutherford analogy as a metaphor

Our modeling in Sections 3.2 and 3.3 presupposes a logical reformulation of physical theories—clearly with the restriction that the representation is qualitative not quantitative. These theories were used to generalize laws and facts

Table 3
Modeling of the source and target domain

types	*types*
object	*object*
entities	*entities*
planet: object	*electron: object*
sun: object	*atom: object*
solar_system: object	
facts	*facts*
$revolves\_around(planet, sun)$	$P(electron, atom)$
$central\_body(solar\_system, sun)$	
$R(x, y) \leftrightarrow revolves\_around(x, z) \wedge central\_body(z, y)$	

from the source and the target domains. A physicist clearly has a theory-guided conceptualization when she is trying to understand a metaphor. This theory needs to be taken into account when modeling the process of understanding. Whereas this is a natural setting for modeling predictive analogies in the described situational embedding, this does not seem to be appropriate in the case metaphorical expressions are modeled more generally. Understanding a metaphor like (4)(i), in general, does not presuppose knowledge about facts and laws of physical theories. A rather simple and straightforward solution is just to thin out the used representation for the predictive analogy case.

In our view, metaphors operate on a basis that corresponds to a large extent to lexical meanings of the involved concepts. In particular, what is missing is a *theory* of a particular domain under consideration: the lexical meaning of a concept most often does not involve a spelled-out conceptualization comparable to current scientific theories. Rather one or more preferred properties are often associated with metaphors governing the new non-conventional meaning of the involved target concepts. In order to make metaphor (4)(i) more precise, the concept *planet* is specified by its relation to *sun* which must be available but is not explicitly mentioned. The same holds for the concepts *nucleus* indirectly introduced by the concept *atom*.

- The concept *sun* is a lexicalized entity. As a possible conceptualization, *sun* can have the following properties: it occurs with other objects (like planets) and builds the center of a more complex system that includes *sun* and these other objects. Important is that this system is finite, i.e. that it is nothing that is arbitrarily extended.
- The lexical meaning of *planet* includes a particular relation to another object *sun*, i.e. the concept *planet* is partially defined via a two-ary relation $revolves\_around(x, y)$ together with a certain sort restriction with respect to x and y. The idea is that sort restrictions allows x and y only to be of sort *object* (and not of sort *real* or *time* etc.). Although *sun* is not mentioned in (4)(i) we can assume *revolves_around*(*planet, sun*) in the source domain when background knowledge and an appropriate inference mechanism is available. Hence, the conceptualization of *planet* introduces a concept *sun* and links both concepts together.
- We think that $revolves\_around$ is the preferred property assigned to *planet*. (In other metaphors in which *planet* occurs this can be different. For example, preferred properties could be *heavy*, *round*, etc.)

The linguistic structure of metaphors allows to partially construct the analogical relation R specified in Definition 11. Knowing that planets occur in a solar system we add the idea of a central body system to the source. What we get by a second-order reasoning step is a relation defined by the central body system constituted by *solar system* and *sun* and a *revolves_around* relation between planet and solar system (compare Table 3).

Using this modeling of the domains, it is possible to apply theory projection. HDTP yields the desired generalization establishing the fact $revolving\_around(electron, nucleus)$ in the target domain. To established the generalization we used the syntactical structure of the metaphor and relatively weak assumptions about the lexical meaning of the involved concepts.

5.3. The general case

First, we will examine examples of metaphoric expressions like the ones in (5). This type of metaphors establishes an analogical relation between an adjective and a noun. Again the syntactic structure of such metaphors provide a hint

Table 4
Modeling the source and the target domain of the poem *Fog*

types	*types*
object, space	*object*
entities	*entities*
cat: object	*fog: object*
harbor: object	
haunches: object	
city: object	
feet: object	
a: object	
functions	
over: object → space	
facts	*facts*
$little(feet)$	$moves(fog)$
$looking(a, over(city))$	$P'(fog)$
$looking(a, over(harbor))$	
$moves(a)$	
$sits\_on(a, haunches)$	
$comes\_on(a, feet)$	
$silent(haunches)$	
$P(cat, feet)$	

to which domain the involved concepts belong to: the noun is used as input for the target domain and the adjective is used as input for the source domain. A characteristic feature of expressions like the ones in (5) to be interpreted as a *metaphor*, is that the adjective cannot occur in the target domain with its lexical meaning. But the adjective is not sufficient to determine totally the source domain. It is essential that the source domain can be enriched by background knowledge in order to become compatible with the target domain: For example, in (5)(ii) it is crucial that *cold* is enriched with concepts like *deadhearted* or *insentient*. Clearly not only background knowledge but also an inference mechanism (corresponding to a theory) is necessary to draw this conclusion. Furthermore a sortal restriction applies: the inferred adjectives in the source domain need to be applicable to humans, because else an analogical relation cannot be established. Similarly as in the modeling of the Rutherford metaphor in Section 5.2 the linguistic structure of the metaphors in (5) determine partially the analogical relation *R* from Definition 11, because the adjective on the source side can only be applied to certain nouns coming from the source domain. Given this situation an application of HDTP results in the desired interpretations of the metaphors given in (5).

We will now consider the poem *Fog* by Sandburg mentioned in Section 5.1. The first problem is whether the involved concepts occurring in the poem should be assigned to the source domain or the target domain. A promising strategy yielding an appropriate conceptualization as input for the algorithm HDTP-A is to assign *fog* and *moves* to the target input A_T, and the rest (plus some background knowledge) to the source input A_S. In other words, concepts used as input for the target domain are either the target concept (*fog*) or attributes that are compatible with *fog* relative to their lexical meaning. Again the linguistic analysis of the poem gives a hint how to establish the analogical relation *R*. Last but not least—similar to the case of the Rutherford analogy—we add a predicate P'(*fog*) that will be crucially used in the further procedure. Table 4 summarizes a (slightly simplified) conceptualization of the poem.

The function $over : object \rightarrow space$ is a rough approximation of a modeling of prepositions modifying local attributes. The unspecified relation $P(cat, feet)$ is triggered by the connection between *cat* and *feet* in the first verse of the poem. Applying HDTP (without further background knowledge) results in the association of a/fog together with properties like $moves(a)/moves(fog)$. Although this is in no sense wrong, the usual interpretation of the poem is the association of *cat* and *fog*. In order to establish this analogical relation, we need background knowledge specifying paradigmatic properties and behaviors of cats and additionally an inference mechanism that allows us to deduce that *a* behaves cat-like: it creeps. Then the resulting complex predicate could be generalized and transferred to the target domain resulting in a specification of P'.

6. Conclusions

The examples discussed make crucial assumptions about the involved domains. For the general case of metaphors, it is clear that such assumptions play an important role for the understanding of metaphors: often the lexical meaning of the involved concepts are not sufficient to establish an analogical relation, but more is needed like background knowledge or inference mechanisms deducing new concepts from the input.

Clearly most of the knowledge that is required to understand metaphors is covered by the designated properties of concepts in the source domain. In general, several of these properties needs to be assumed. For example, *planet* could have a similar role *sun* has in the Rutherford analogy when we change the roles like in (6):

(6) An electron in a hydrogen atom is the moon of this atom.

In this example, *planet* needs to be introduced and linked to *moon*. The relevant properties of *planet* change dramatically: now the gravitation center is considered to be the planet. What is needed for an appropriate modeling is a list of designated properties of the involved concepts. Such a list must contain relevant information concerning possible properties of concepts that can play a role in metaphors. Furthermore, it seems to be reasonable to rank these properties according to their importance. Clearly, this is a purely empirical problem, but it is necessary to get the correct input for the machinery.

In this paper, we showed on the one hand that metaphors and analogies are closely related to each other but exhibit significant differences at the other. We introduced the framework HDTP as a means to model generalizations of theories together with the correlated substitutions instantiating the generalized theory with respect to the original domains. HDTP was characterized syntactically, algorithmically, and semantically. We applied HDTP to predictive analogies in qualitative physics, to corresponding metaphors of such predictive analogies, to analogies where the adjective modifies a noun metaphorically, and finally to a (more or less) complex poem. HDTP-A is implemented in PROLOG (cf. [12]). The algorithm HDTP-A solves convincingly the standard problems of analogical reasoning well known from the literature.

References

[1] J. Anderson, R. Thompson, Use of analogy in a production system architecture, in: A. Vosniadou, A. Ortony (Eds.), Similarity and Analogical Reasoning, Cambridge, 1989, pp. 267–297.

[2] J. Burghardt, B. Heinz, Implementing anti-unification modulo equational theory, Tech. Rep., Arbeitspapiere der GMD, 1006, 1996.

[3] M. Dastani, Languages of perception, ILLC Dissertation Series 1998-05, 1998, http://www.cs.uu.nl/mehdi/publications/thesis.ps.ps.

[4] M. Dastani, B. Indurkhya, An algebraic approach to similarity and categorization, in: An Interdisciplinary Workshop on Similarity and Categorization, Edinburgh, Scotland, 1997.

[5] M. Dastani, B. Indurkhya, R. Scha, An algebraic method for solving proportional analogy problems involving sequential patterns, Mind II: Computational Models of Creative Cognition, Dublin, 1997, pp. 1–15.

[6] T. Evans, A program for the solution of a class of geometric-analogy intelligence-questions, in: M. Minsky (Ed.), Semantic Information Processing, MIT press, Cambridge, MA, 1968, pp. 271–353.

[7] B. Falkenhainer, K. Forbus, D. Gentner, The structure-mapping engine: algorithm and example, Artif. Intell. 41 (1989) 1–63.

[8] D. Gentner, The mechanisms of analogical learning, in: S. Vosniadou, A. Ortony (Eds.), Similarity and Analogical Reasoning, Cambridge, 1989, pp. 199–241.

[9] D. Gentner, Structure-mapping: a theoretical framework for analogy, Cognitive Sci. 7 (1983) 155–170.

[10] D. Gentner, B. Bowdle, P. Wolff, C. Boronat, Metaphor is like analogy, in: D. Gentner, K. Holyoak, B. Kokinov (Eds.), The Analogical Mind: Perspectives from Cognitive Science, Cambridge, MA, 2001, 199–253.

[11] D. Griffiths, The Classical Era (1897–1932), New York, 1987.

[12] H. Gust, K.-U. Kühnberger, U. Schmid, Anti-unification of axiomatic systems, Tech. Rep. 2003, http://www.cogsci.uos.de/∼helmar/analogy1.ps/.

[13] H. Gust, K.-U. Kühnberger, U. Schmid, Solving predictive analogy tasks with anti-unification, Proc. Joint Internat. Conf. Cognitive Science 2003 (ICCS ASCS 2003), Sydney.

[14] H. Gust, K.-U. Kühnberger, U. Schmid, Metaphors and anti-unification, in: F. Spoto, G. Scollo, A. Nijholt (Eds.), Proc. Twenty-First Workshop on Language Technology: Algebraic Methods in Language Processing, Verona, Italy, 2003, pp. 111–123.

[15] H. Gust, K.-U. Kühnberger, U. Schmid, Ontological aspects of computing analogies, in: Proc. Sixth Internat. Conf. Cognitive Modeling, Lawrence Earlbaum, Mahwah, NJ, 2004.

[16] J. Hummel, K. Holyoak, Distributed representation of structure: a theory of analogical access and mapping, Psychol. Rev. 104 (3) (1997) 427–466.

[17] D. Hofstadter, The Fluid Analogies Research Group, Fluid Concepts and Creative Analogies, New York, 1995.

[18] B. Indurkhya, Metaphor and Cognition, Kluwer, Dordrecht, The Netherlands, 1992.

[19] B. Jain, F. Wysotzki, Self-organizing recognition and classification of relational structures, in: Proc. 24th Annu. Meeting of the Cognitive Science Society, Fairfax, Virginia, Mahwah, NJ, 2002, pp. 488–493.

[20] F. Klix, Analytische Betrachtungen über Struktur und Funktion von Inferenzen, Z. Psychol. 201 (1993) 393–414.

[21] B. Kokinov, A. Petrov, Integrating memory and reasoning in analogy-making: the AMBR model, in: D. Gentner, K. Holyoak, B. Kokinov (Eds.), The Analogical Mind: Perspectives from Cognitive Science, Cambridge, MA, 2001, pp. 59–124.

[22] S. O'Hara, A model of the redescription process in the context of geometric proportional analogy problems, in: Proc. Internat. Workshop on Analogy and Inductive Inference (AII'92), Springer, Berlin, 1992, pp. 268–293.

[23] G. Plotkin, A note on inductive generalization, Mach. Intell. 5 (1969) 153–163.

[24] J. Reynolds, Transformational systems and the algebraic structure of atomic formulas, Mach. Intell. 5 (1970) 135–151.

[25] U. Schmid, H. Gust, K.-U. Kühnberger, J. Burghardt, An Algebraic Framework for Solving Proportional and Predictive Analogies, in: F. Schmalhofer, R. Young, G. Katz (Eds.), Proc. European Conf. Cognitive Science, Osnabrück, Germany, 2003, Lawrence Earlbaum, NJ, 2003, pp. 295–300.

[26] J. Thomson, On the structure of the atom: an investigation of the stability and periods of oscillation of a number of corpuscles arranged at equal intervals around the circumference of a circle; with application of the results to the theory of atomic structure, Philos. Mag., Series 6, 7(39), (1904) 237–265.

[27] U. Wagner, Combinatorically restricted higher order anti-unification—an application to programming by analogy, Diploma Thesis, Technical University of Berlin, 2002.