Abstract Theorem Proving *

Fausto Giunchiglia
Mechanised Reasoning Group
IRST
Povo, I 38100 Trento
Italy
fausto@irst.uucp

Toby Walsh
Department of Artificial Intelligence
University of Edinburgh
80 South Bridge, Edinburgh
Scotland
Toby_Walsh@uk.ac.edinburgh

Abstract

Informally, **abstraction** can be described as the process of mapping a representation of a problem into a new representation. The aim of the paper is to propose a *theory* of abstraction. The generality of the framework is tested by formalizing and analyzing some work done in the past [Sacerdoti, 1974, Hobbs, 1985, Plaisted, 1981]; its efficacy by giving a procedure which solves the "*false proof*" problem [Plaisted, 1981] by avoiding the use of inconsistent abstract spaces.

1 Introduction

Abstraction has been suggested as a very powerful technique for constraining search in automated reasoning. *Informally*, abstraction can be described as the process of mapping a representation of a problem (also called the "*ground*" representation) into a new representation (also called the "*abstract*" representation) which *preserves certain desirable properties* and *is simpler to handle*. The "desirable properties" amount to requiring that the abstract solution be of help in solving the problem in the original search space. The notion of "simplicity" depends on the application, it may mean decidability or lower complexity. As far as we know, no comprehensive theory of abstraction has been given. The only work in this direction [Plaisted, 1981] is concerned with one form of abstraction and is limited to the area of resolution theorem proving. This has caused the lack of a satisfactory characterization and general understanding of abstraction.

*This work was begun when the first author was working at the Department of Artificial Intelligence at Edinburgh University supported by SERC grant GR/E/4459.8. The second author is supported by a SERC studentship. The research described in this paper owes a lot to the openess and sharing of ideas which exists in the Mathematical Reasoning group. The authors thank Alan Bundy, Enrico Giunchiglia, Alex Simpson and Richard Weyhrauch for the many discussions on the topic. Alan Bundy is also thanked for reading early versions of the paper.

The aim of the work (partially) described in this paper is to provide a theory of abstraction and use it to: **(1)** classify the various forms of abstraction; **(2)** investigate their formal properties and the operations which can be defined on them; **(3)** analyze and classify past work; **(4)** define ways of building "useful abstractions" and **(5)** study how the proof in the abstract space can be used to aid the proof in the ground space. In this paper, for lack of space, only some issues are discussed and proofs are only outlined or not given (for a more complete treatment see [Giunchiglia and Walsh, 1990]).

In our formal framework (section 2), an abstraction is just a mapping between formal systems. As this is a very general concept, we concentrate on the classes of abstraction which preserve provability. This captures most of the relevant previous work in *abstract theorem proving* and *planning* (section 3). In section 4, we investigate the "*false proof*" problem [Plaisted, 1981]; when abstracting a problem, we may throw too much information away and leave an inconsistent abstract space. We prove that this problem cannot be avoided as it is **always** true for the class of abstractions we have come across in abstract theorem proving and planning. However, we are able to propose a (decidable) solution to this problem.

2 The formal framework

Definition 1 (Formal system) *: A* **formal system** Σ *is a triple* $\langle \Lambda, \Delta, \Omega \rangle$*, where* Λ *is the* **Language***,* Ω *is the* **set of axioms** *and* Δ *is the* **Deductive Machinery** *of* Σ*.*

The language Λ is composed of an alphabet, the set of (well formed) terms and the set of well formed formulae (wffs from now on). Ω is a subset of the wffs of Λ. The deductive machinery is a set of rules of inference for deriving theorems from axioms.

Definition 2 (Abstraction) *: If* $\Sigma_1 = \langle \Lambda_{\Sigma_1}, \Omega_{\Sigma_1}, \Delta_{\Sigma_1} \rangle$ *and* $\Sigma_2 = \langle \Lambda_{\Sigma_2}, \Omega_{\Sigma_2}, \Delta_{\Sigma_2} \rangle$ *are two formal systems, an abstraction mapping* f*, written also* $f : \Sigma_1 \mapsto \Sigma_2$*, is a triple of total functions* $\langle f_\Lambda, f_\Omega, f_\Delta \rangle$ *such that:*

$$f_\Lambda : \Lambda_{\Sigma_1} \mapsto \Lambda_{\Sigma_2}$$
$$f_\Omega : \Omega_{\Sigma_1} \mapsto \Omega_{\Sigma_2}$$

$$f_\Delta : \Delta_{\Sigma_1} \mapsto \Delta_{\Sigma_2}$$

If not explicitly stated to the contrary we assume that f_Λ and f_Ω agree on the axioms; that is, for any wff ω, if $\omega \in \Omega$, then $f_\Lambda(\omega) = f_\Omega(\omega)$[1]. When no confusion arises we drop the subfixes. Given a deduction tree Π_{Σ_1} of $\vdash_{\Sigma_1} \varphi_{\Sigma_1}$ in Σ_1, we indicate by $f(\Pi_{\Sigma_1})$ a deduction trees Π_{Σ_2} of $\vdash_{\Sigma_2} f(\varphi_{\Sigma_1})$.

Definition 3 (T*-abstractions) *: An abstraction $f : \Sigma_1 \mapsto \Sigma_2$ is said to be a* [2]

TC-Abstraction *iff, for any wff φ_{Σ_1}, $\vdash_{\Sigma_1} \varphi_{\Sigma_1}$ iff $\vdash_{\Sigma_2} f(\varphi_{\Sigma_1})$;*

TD-Abstraction *iff, for any wff φ_{Σ_1}, if $\vdash_{\Sigma_2} f(\varphi_{\Sigma_1})$ then $\vdash_{\Sigma_1} \varphi_{\Sigma_1}$;*

TI-Abstraction *iff, for any wff φ_{Σ_1}, if $\vdash_{\Sigma_1} \varphi_{\Sigma_1}$ then $\vdash_{\Sigma_2} f(\varphi_{\Sigma_1})$.*

We write *"T*-abstraction"* to mean any of the above abstractions, TH(Σ) to mean the set of wffs provable in Σ and NTH(Σ) to mean the set of wffs which, if added to the axioms of Σ, make it inconsistent. For example, a TI-abstraction can be represeted by the following figure:

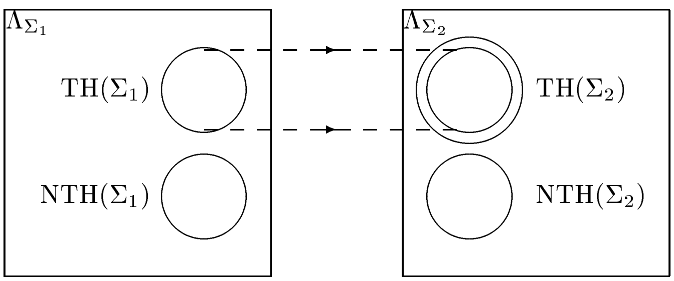

Figure 1: TI-abstraction

TC-abstractions map all the elements of TH(Σ_1) into elements of TH(Σ_2) and these are all and only the elements of TH(Σ_2). Herbrand's theorem can be formalized as a TC-abstraction. TC-abstractions are used, for instance, in decision theory, under the name of *reduction methods*, to prove the decidability of and build deciders for the validity problem for certain subclasses of the first order calculus. The trick is to find a class whose decidability is known and prove that there is a proof of a wff **iff** there is a proof of the "translated" wff in the new class.

In TD-abstractions a subset of the elements of TH(Σ_1) is mapped into TH(Σ_2) and these are all the elements of TH(Σ_2). TD-abstractions are used, for instance, to implement derived inference rules and, as alternatives to TI-abstractions, to overcome some of their problems [Tenenberg, 1987].

In TI-abstractions all the elements of TH(Σ_1) are mapped into a subset of TH(Σ_2). TI-abstractions have been mostly used in "*abstract theorem proving*" [3]. The main idea underlying the use of these abstractions is to prove the abstracted theorem in Σ_2 (which, supposedly, should be simpler than in Σ_1) and then to use the structure of the proof in Σ_2 to shape the proof in Σ_1. The fact that there is a proof in Σ_2 **does not** guarantee that there is a proof in Σ_1.

T*-abstractions are classified on how provability is preserved between the ground space and the abstract space; they are thus useful when the deductive machinery is defined to generate theorems. On the other hand there are formal systems (*ie.* resolution) whose deductive machinery determines inconsistency. In these cases, abstractions must be classified on how inconsistent formal systems are mapped. This requires the definition of new classes of abstractions, called **NT*-abstractions**. Thus, for instance, **NTI-abstractions** are defined as follows [4]:

Definition 4 *: An abstraction $f : \Sigma_1 \mapsto \Sigma_2$ is an* **NTI-Abstraction** *iff, for any wff φ_{Σ_1}, if adding φ_{Σ_1} to the axioms of Σ_1 yields an inconsistent formal system, then adding $f(\varphi_{\Sigma_1})$ to the axioms of Σ_2 yields an inconsistent formal system.*

Various properties, equivalences, and relationships among T*- and NT*-abstractions can be proved [Giunchiglia and Walsh, 1990]. We can draw a similar figure for NTI-abstractions as for TI-abstractions:

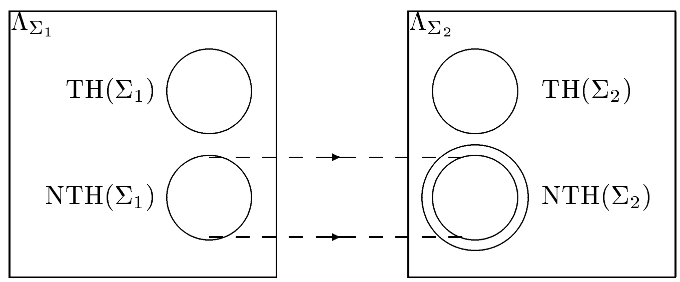

Figure 2: NTI-abstractions (Falseful abstractions)

If a formal system Σ has the "classical" negation, then, for any wff α, $\alpha \in TH(\Sigma)$ iff $\neg\alpha \in NTH(\Sigma)$. Thus trivially:

Corollary 1 *: If Σ_1 and Σ_2 are two formal systems with (classical) negation and if $f : \Sigma_1 \mapsto \Sigma_2$ is a TI-*

[1] To be precise, since we distinguish between wffs occurring as axioms and as anything else, we should consider occurrences of wffs and not wffs. Since, in this paper, for any wff ω, if $\omega \in \Omega$, then $f_\Lambda(\omega) = f_\Omega(\omega)$, to make things simpler, we consider f_Λ and f_Δ to range over wffs.

[2] "T" stands for theorem, "C" for constant, "D" for decreasing and "I" for increasing.

[3] Note that we give the word "*abstraction*" a wider meaning then before (see definition 2).

[4] NTC-abstractions and NTD-abstractions are defined analogously to TC-abstractions and TD-abstractions respectively, but preserving inconsistency instead of theoremhood (see definitions 3, 4).

abstraction then, for any α, if $\alpha \in TH(\Sigma_1)$ then $\neg\alpha \in NTH(\Sigma_1)$ and $\neg f(\alpha) \in NTH(\Sigma_2)$.

In many cases, a refutation system, taking as input a goal formula α (usually automatically) negates it, adds the result to the axioms and tries to prove that the resulting theory is inconsistent. A TI-abstraction will therefore be useful if instead of adding $f(\neg\alpha)$ to the abstract space, we add $\neg f(\alpha)$. Additionally there are TI-abstractions which can be used in resolution-based systems independently of whether the goal or its negation is abstracted.

Definition 5 (Negation preserving) *: An abstraction $f : \Sigma_1 \mapsto \Sigma_2$ is* **negation preserving** *iff $f(\neg\alpha) = \neg f(\alpha)$.*

Theorem 1 *: If $\Sigma_1 = \langle\Lambda_1, \Omega_1, \Delta_1\rangle$ and $\Sigma_2 = \langle\Lambda_2, \Omega_2, \Delta_2\rangle$ are two formal systems with (classical) negation, then a negation preserving abstraction $f : \Sigma_1 \mapsto \Sigma_2$ is a TI-abstraction iff $f' : \Sigma_1' \mapsto \Sigma_2'$ is a NTI-abstraction, where $\Sigma_1' = \langle\Lambda_1, \Omega_1, \Delta_1'\rangle$, $\Sigma_2' = \langle\Lambda_2, \Omega_2, \Delta_2'\rangle$ and Δ_1' and Δ_2' are such that $TH(\Sigma_1') = TH(\Sigma_1)$ and $TH(\Sigma_2') = TH(\Sigma_2)$.*

Examples are $f = f'$ with Σ_1, Σ_2, Σ_1' and Σ_2' being natural deduction, and $f \neq f'$ with Σ_1, Σ_2 being natural deduction and Σ_1', Σ_2' being resolution. As far as we know, all the abstractions proposed to work in resolution systems are negation preserving. However, there are useful abstractions which are not negation preserving (for instance when negation is not part of the language of Σ_1 or Σ_2 [Newell *et al.*, 1963], or only partially preserved by the mapping).

3 Some examples of abstraction

The purpose of these examples, together with providing a rational reconstruction of the work described, is to convince the reader that the framework is very powerful and allows us to present an unified view of the work done in different areas and with different goals. For lack of space, only three examples are reported, more "historical" examples are reported in [Giunchiglia and Walsh, 1990].

Example 1 [Planning]: Abstrips [Sacerdoti, 1974] was one of the first noticeable applications of abstraction. In Abstrips the preconditions to operators were abstracted according to their *criticality*. To formalize Strips-like planning we shall adopt a situation calculus in a natural deduction formal system. Let us consider the abstraction f_{AB} where Σ_1 and Σ_2 are situation calculi with a first order language, Ω consists of frame, operator and theoretic axioms and Δ consists of natural deduction rules of inference. Operators are wffs of the form " $\forall s.(\bigwedge_{1\leq i\leq n} p_i(s) \rightarrow q(f(s)))$ " where p_i is a precondition, s is a state of the world, f is some action, and q describes the new state of the world. Goals are wffs of the form "$\exists s\ r(s)$". f_{AB} is applied to wffs and axioms as follows:

$f_{AB}(\alpha) = \alpha$ if α is an atomic formula.

$f_{AB}(\neg\alpha) = \neg f_{AB}(\alpha)$;

$f_{AB}(\alpha \circ \beta) = f_{AB}(\alpha) \circ f_{AB}(\beta)$, where "$\circ$" is "$\wedge$" or "$\vee$";

$f_{AB}(\sharp x\ \alpha) = \sharp x\ f_{AB}(\alpha)$, where "$\sharp$" is "$\exists$" or "$\forall$";

$f_{AB}(\alpha \rightarrow \beta) = f_{AB}(\alpha) \rightarrow f_{AB}(\beta)$, provided "$\alpha \rightarrow \beta$" is not an operator;

$f_{AB}(\bigwedge_{1\leq i\leq n} p_i(s) \rightarrow r) = \bigwedge_{i\in crit(\kappa)} p_i(s) \rightarrow f_{AB}(r)$, provided that "$\bigwedge_{1\leq i\leq n} p_i(s) \rightarrow r$" is an operator, where $i \in crit(\kappa)$ if the criticality of p_i is greater than κ.

Theorem 2 *: f_{AB} is TI, namely, if $\vdash_{\Sigma_1} \varphi_{\Sigma_1}$, then $\vdash_{\Sigma_2} f_{AB}(\varphi_{\Sigma_1})$.*

Proof[Outline]: By proving that given a deduction tree Π_{Σ_1} of $\vdash_{\Sigma_1} \varphi_{\Sigma_1}$, we can build a deduction tree $\Pi_{\Sigma_2} = f^i_{AB}(\Pi_{\Sigma_1})$ of $\vdash_{\Sigma_2} f_{AB}(\varphi_{\Sigma_1})$. The proof proceeds by induction on the weight N [5] of Π_{Σ_1}. For proofs of length 1, f_{AB} is applied to the single wff; this generates a valid proof in Π_{Σ_2}. Assume that we have a deduction tree up to size N. Any rule application that is not modus ponens involving an operator translates unmodified, in the sense that, for instance, an "$\vee$I" on φ in Π_{Σ_1} becomes an "$\vee$I" on $f_{AB}(\varphi)$ in Π_{Σ_2}. For an operator application, the following transformation is performed:

$$\frac{\dfrac{\Pi_1}{\bigwedge_{1<i<n} p_i} \qquad \bigwedge_{1<i<n} p_i \rightarrow q}{q} \quad \Longrightarrow$$

$$\frac{\dfrac{f_{AB}\left(\dfrac{\Pi_1}{\bigwedge_{1\leq i\leq n} p_i}\right)}{\bigwedge_{i\in crit(\kappa)} p_i} \qquad \bigwedge_{i\in crit(\kappa)} p_i \rightarrow f_{AB}(q)}{f_{AB}(q)}$$

The abstract proof is valid since $f_{AB}(...)$ is a valid deduction tree from the induction hypothesis, and the hypothesis of the (abstract) operator axiom is obtained from "$\bigwedge_{1\leq i\leq n} p_i$" by a (possibly empty) sequence of applications of "and-elimination". $\square$

Note that the abstract proof is longer than the one in the ground space. The purpose of abstracting is not to find these longer proofs; we hope that there are also going to be shorter proofs. These shorter proofs are those that don't try to satisfy p_j for $j \notin crit(\kappa)$. There is no guarantee that there will be a shorter proof than the one exhibited; we will always be able to devise an obtuse theory in which to prove the p_i for $i \in crit(\kappa)$ we have to prove all the other p_j for $j \notin crit(\kappa)$. ♠

[5] The *weight* of a deduction tree is the number of its formula occurrences.

Example 2 [Resolution theorem proving, logic programming]: The work by Plaisted is closest in spirit to ours. Plaisted proposes two classes of abstraction, *ordinary abstractions* and *weak abstractions* [Plaisted, 1981], which map a set of clauses onto a set of clauses and preserve inconsistency. His work is less general than ours as: *he restricts his attention to resolution systems* and *his classes of abstraction fail to capture all NTI-abstraction mappings that preserve inconsistency between resolution systems.* In other words, Plaisted's abstractions are NTI, but not all NTI-abstractions are weak or ordinary [6]. Moreover we claim that our definitions of abstraction are "more natural" in the sense that better reflect and capture the functionalities they are given for.

Ordinary abstractions are described as taking both Σ_1 and Σ_2 to be first order calculi with Λ_Σ allowing clausal form, Δ_Σ being resolution and Ω_Σ being arbitrary. Any ordinary abstraction mapping f maps a clause in Λ_{Σ_1} onto a set of clauses in Λ_{Σ_2} subject to the following conditions:

a) $f(\bot) = \{\bot\}$;

b) if α_3 is a resolvent of α_1 and α_2 in Σ_1, and $\beta_3 \in f(\alpha_3)$ then there exist $\beta_2 \in f(\alpha_2)$ and $\beta_1 \in f(\alpha_1)$ such that a resolvent of β_1 and β_2 subsumes β_3 in Σ_2;

c) if α_1 subsumes α_2 in Σ_1, then for every $\beta_2 \in f(\alpha_2)$ there exists $\beta_1 \in f(\alpha_1)$ such that β_1 subsumes β_2 in Σ_2.

Weak abstractions are identically defined to ordinary abstractions except condition **b)** is weakened to the property that if α_3 is a resolvent of α_1 and α_2 in Σ_1, and $\beta_3 \in f(\alpha_3)$ then there exist $\beta_2 \in f(\alpha_2)$ and $\beta_1 \in f(\alpha_1)$ such that either β_1 subsumes β_3, or β_2 subsumes β_3, or a resolvent of β_1 and β_2 subsumes β_3 in Σ_2.

Theorem 3 : *Weak and ordinary abstractions are NTI.*

Proof: The proof is a corollary to Theorem 2.5 on page 55 of [Plaisted, 1981]. □

Theorem 4 : *There exist NTI-abstractions between resolution systems that are not weak or ordinary abstractions.*

Proof[Outline]: We can find NTI-abstractions that fail every one of the three conditions in the definition of weak and ordinary abstractions. Condition **a)** is failed by the NTI-abstraction f such that, for any wff φ in Σ_1, $f(\varphi) = \{\varphi \vee \bot\}$. The problem with condition **b)** is that we may also need to resolve with an axiom of the theory. Consider, for instance, the abstraction defined by $f(p \vee q) = \{p \vee r\}$ and $f(\varphi) = \{\varphi\}$ otherwise. If Σ_1 contains the axioms, $\neg q$, and $\neg r$ then f is NTI. In particular, $p \vee q$ resolves with $\neg p$ in Σ_1 to give q. However, no clause in the abstraction of $p \vee q$, or $\neg p$ (or their resolvent) subsumes the clause q found in the abstraction of q. For condition **c)**, consider the abstraction defined by $f(p \vee q) = \{r, p \vee q\}$ and $f(\varphi) = \varphi$ otherwise. Now f is NTI. However, f fails condition **c)** of the definition of weak and ordinary abstractions as p subsumes $p \vee q$ but no clause in the abstraction of p subsumes r which is in $f(p \vee q)$. □

The definition of weak and ordinary abstractions could be extended to overcome the first counter-example by replacing condition **a)** with the more general requirement that $\exists \varphi \in f(\bot). \vdash_{\Sigma_1} \neg\varphi$. However, this still leaves useful NTI-abstractions that fail conditions **b)** and **c)**. For example, if $p_0 \leftrightarrow p_i$ for many i we might abstract many clauses of the form $p_i \vee q$ onto the one clause $\{p \vee q\}$. One could argue that ordinary and weak abstractions have the advantage, over NTI-abstractions, that they **always** map into simpler theories, in the sense that there is always an abstract proof that is no longer than the shortest proof of the unabstracted theorem [Plaisted, 1981]. This does not seem a good point since, first of all, we intuitively expect NTI-abstractions (that are not NTC) to have this or similar properties and, second and more importantly, there are NTI-abstractions, which are not weak or ordinary, which build simpler theories (the last example is one possible case). ♠

Example 3 [Common sense reasoning]: In [Hobbs, 1985], Hobbs presents a theory of granularity in which a complex theory is abstracted onto a simpler, more "coarse-grained" theory with a smaller domain; for example, we could map the real world of continuous time and positions onto a (micro)world of discrete time and positions. Hobbs' granularity theory can be formalized as a mapping (let us call it "f_{gran}") that can be proved to be TI. Let us suppose that both Σ_1 and Σ_2 are calculi with a first order language, an arbitrary set of axioms and any complete deductive machinery for first order logic. f_{gran} maps different objects in Σ_1 into (not necessarily different) objects in Σ_1 according to an *indistinguishability relation*, defined by the (second-order) axiom:

$$\forall x, y. x \sim y \;\leftrightarrow\; \forall p \in R. p(x) \leftrightarrow p(y)$$

where R is the subset of the predicates of the theory determined to be *relevant* to the situation at hand[7]. Thus f_{gran} keeps the same logical structure of wffs and translates any constant into its equivalence class, namely $f_{gran}(p(a)) = p(\kappa(a))$ where a is any constant symbol

[6] All Plaisted's examples of abstraction are negation preserving and thus also TI.

[7] As in [Hobbs, 1985], we define indistinguishability for unary predicates; it can, however, be easily generalized to n-ary predicates.

and $\kappa(a)$ is the equivalence class of the constant a wrt the indistinguishability relation; that is $\kappa(x) = \{y : x \sim y\}$.

Theorem 5 *: f_{gran} is TI/NTI.*

Proof By mapping a proof tree Π_{Σ_1} in Σ_1 of φ_{Σ_1} (possibly of $\perp$) onto a proof tree Π_{Σ_2} in Σ_2 of $f(\varphi_{\Sigma_1})$. The proof proceeds by induction on the weight of Π_{Σ_1}. We merely apply f_Λ to every wff in Π_{Σ_1}. □

(Like any TI abstraction, see next section) f_{gran} can map a consistent theory into an inconsistent theory. For example, if the constants a and b are "indistinguishable", then a consistent theory with equality and the axiom $\neg(a = b)$ maps into an inconsistent theory with the axiom $\neg(\{a, b\} = \{a, b\})$. However, the following result holds:

Theorem 6 *: f_{gran} preserves consistency if indistinguishability is defined over all predicates.*

Proof[Outline]: By contradiction. Assume that a consistent theory, Σ_1 maps onto an inconsistent theory, Σ_2. That is, we can find a proof tree, Π_{Σ_2} of $\perp$. We show how you can construct a proof tree, Π_{Σ_1} of $\perp$, contradicting the assumption that Σ_1 is consistent. For every equivalence class, $\kappa(a)$ we pick one member of that class, b; to every wff, φ in Π_{Σ_2} we apply the substitution $\{\kappa(a)/b\}$. This will generate a proof tree, Π_{Σ_1} whose assumptions will either be axioms of Σ_1 or will be derivable from them using the indistinguishability relation and substitution of equivalences. If indistinguishability is not defined over **all** predicates, this last fact will not necessarily be true. □

Note that f_{gran} is a special case of the example of weak/ ordinary abstractions (given in [Plaisted, 1981]) where function symbols are renamed in a systematic (but not necessarily 1-to-1) way. ♠

4 The false proof problem

A major problem with the use of TI-abstractions [8] is that, even if Σ_1 is consistent, Σ_2 may be inconsistent. An example has already been given for f_{gran}. With f_{AB} it is sufficient to consider abstracting the operators "$\alpha_1 \wedge \alpha_2 \rightarrow \alpha_3$" and "$\alpha_1 \wedge \alpha_4 \rightarrow \neg\alpha_3$" onto "$\alpha_1 \rightarrow \alpha_3$" and "$\alpha_1 \rightarrow \neg\alpha_3$" when α_1 is a theorem but α_2 and α_4 are not both theorems. This problem was noticed by Plaisted who called it the *"false proof problem"*. It can be seen as a sympton of the following theorem:

Lemma 1 *: If $f : \Sigma_1 \mapsto \Sigma_2$ is an abstraction and Σ_2 is inconsistent then f is a TI-abstraction.*

[8] Everything stated in this section holds dually for NTI-abstractions.

Lemma 1 holds independently of the consistency of Σ_1. Once f has been proved to be TI it **may** happen that Σ_2 is inconsistent. This is a major blow to the use of TI-abstractions to guide the proof in the ground space. When Σ_2 is inconsistent the structure of the proof in Σ_2 could still be used to shape the proof in Σ_1. However, any wff in Σ_2 is a theorem and thus Σ_2 does not filter out any of the wffs which are not theorems in Σ_1. In a way Σ_2 gives too little information. To make matters worse, in general it is not possible to decide in a finite amount of time whether a formal system is consistent.

When working with a fixed formal system (*ie.* set theory + first order logic) a solution is to build abstractions which are proved a priori to have consistent Σ_2. In many cases, however, (*ie.* planning, logic programming, knowledge based systems), while the set of inference rules of Σ_1 is fixed, its axioms may vary and depend on the application. Tenenberg [Tenenberg, 1987] proposed a solution to the problem in the case of a form of predicate abstractions [9] in a resolution-based system. However, the abstractions he proposes have many drawbacks: the first abstraction is TI but the construction of Ω_{Σ_2} is not decidable (even if recursively enumerable) and it may take an infinite amount of time to generate it; the other two types of abstraction are TD or similar to TD [Giunchiglia and Walsh, 1990] [10]. This means that completeness is lost since there is at least one theorem in Σ_1 whose abstraction is not a theorem in Σ_2. We consider this the one property you do not want to lose.

The ideal solution would be to generalize the concept of abstraction mapping to be parameterized on the axioms of Σ_1 and then to find **sufficient conditions** which guarantee that a TI-abstraction maps Σ_1 into a consistent Σ_2, independently of the axioms of Σ_1 (as long as Σ_1 is consistent). This seems a reasonable request since there are abstractions which, fix Λ_{Σ_1}, Λ_{Σ_2}, Δ_{Σ_1}, Δ_{Σ_2} f_Λ, f_Ω and f_Δ are TI for any choice of the theoretic axioms (this is, for instance, true for the abstractions of the three examples) [11].

Let Λ_{Σ_1} and Λ_{Σ_2} be two languages, Δ_{Σ_1} and Δ_{Σ_2} two deductive machineries. Then, if $f_\Lambda : \Lambda_{\Sigma_1} \mapsto \Lambda_{\Sigma_2}$, $g : \Lambda_{\Sigma_1} \mapsto \Lambda_{\Sigma_2}$ and $f_\Delta : \Delta_{\Sigma_1} \mapsto \Delta_{\Sigma_2}$ are three total functions, $F = \langle f_\Lambda, g, f_\Delta \rangle$ is an abstraction from $\Sigma_1 = \langle \Lambda_{\Sigma_1}, \Lambda_{\Sigma_1}, \Delta_{\Sigma_1} \rangle$ to $\Sigma_2 = \langle \Lambda_{\Sigma_2}, \Lambda_{\Sigma_2}, \Delta_{\Sigma_2} \rangle$. Then for any $\Omega_{\Sigma_1} \subset \Lambda_{\Sigma_1}$, if by "$g \uparrow \Omega_{\Sigma_1}$" we indicate g restricted to apply to Ω_{Σ_1}, $F^{\Omega_{\Sigma_1}} = \langle f_\Lambda, g \uparrow \Omega_{\Sigma_1}, f_\Delta \rangle$ is an abstraction from $\Sigma_1^{\Omega_{\Sigma_1}} = \langle \Lambda_{\Sigma_1}, \Omega_{\Sigma_1}, \Delta_{\Sigma_1} \rangle$ to $\Sigma_2^{\Omega_{\Sigma_1}} = \langle \Lambda_{\Sigma_2}, \Omega_{\Sigma_2}, \Delta_{\Sigma_2} \rangle$, with $\Omega_{\Sigma_2} = g(\Omega_{\Sigma_1})$.

[9] Predicate abstractions are abstractions where distinct predicate symbols in Σ_1 are mapped onto (possibly not distinct) predicate symbols in Σ_2 [Giunchiglia and Walsh, 1990].

[10] Note that it can be proved that, more generally, for **any** TD-abstraction, if Σ_1 is consistent, so is Σ_2.

[11] Of course theory independent TI-abstractions are in general less efficient than the ones geared towards one single theory as they do not exploit the structure of theoretic axioms.

Theorem 7 *: Let Λ_{Σ_1} and Λ_{Σ_2} be two languages, Δ_{Σ_1} and Δ_{Σ_2} two deductive machineries, $f_\Lambda : \Lambda_{\Sigma_1} \mapsto \Lambda_{\Sigma_2}$, $g : \Lambda_{\Sigma_1} \mapsto \Lambda_{\Sigma_2}$ and $f_\Delta : \Delta_{\Sigma_1} \mapsto \Delta_{\Sigma_2}$ three total functions. Then there exists $\Omega_{\Sigma_1} \subset \Lambda_{\Sigma_1}$ such that, if the abstraction $F^{\Omega_{\Sigma_1}} = \langle f_\Lambda, g \uparrow \Omega_{\Sigma_1}, f_\Delta \rangle$ is TI and NTI but not NTC, then $\Sigma_1^{\Omega_{\Sigma_1}}$ is consistent and $\Sigma_2^{\Omega_{\Sigma_1}}$ is inconsistent.*

Proof[Outline]: By constructing Ω_{Σ_1}. Because $F^{\Omega_{\Sigma_1}}$ is NTI- but not NTC-, there exists a wff φ such that adding $F^{\Omega_{\Sigma_1}}(\varphi)$ as an axiom to Σ_2 makes an inconsistent formal system, but that adding φ as an axiom to Σ_1 doesn't. □

Theorem 7 can actually be proved in more powerful forms; however the hypotheses hold for most TI-abstractions. For instance negation preserving abstractions that are TI are also NTI and vice versa (theorem 1). Theorem 7 proves that we cannot find a TI-abstractions which maps a consistent Σ_1 into a consistent Σ_2 independently of the axioms of Σ_1. However, we can find (syntatic characterisations of) subsets of theories for which consistency is guaranteed. A different solution to the false proof problem is to vary the TI-abstraction until we can (decidably) show that Σ_2 is consistent. TI-abstractions applied to the same Σ_1 can be classified into a weak partial order, indicated by "$\sqsubseteq$".

Definition 6 ($\sqsubseteq$) *: If $f_i : \Sigma_1 \mapsto \Sigma_2^i$ and $f_j : \Sigma_1 \mapsto \Sigma_2^j$ are two TI-abstractions then $f_i \sqsubseteq f_j$ iff for all wffs φ_{Σ_1}, if $\vdash_{\Sigma_2^i} f_i(\varphi_{\Sigma_1})$ then $\vdash_{\Sigma_2^j} f_j(\varphi_{\Sigma_1})$. We say that f_i is* **weaker** *than f_j or, dually, that f_j is* **stronger** *than f_i.*

If $f_i \sqsubseteq f_j$, then f_j is stronger than f_i in the sense that there are fewer wffs which are theorems in Σ_2^i and not in Σ_1 than wffs which are theorems in Σ_2^j and not in Σ_1. "$\sqsubseteq$" is in general a weak partial order (respecting transitivity, antisymmetry and reflexivity) but not a total order. If, however, we have a set of totally ordered abstractions then the following result holds:

Theorem 8 *: If $f_1 : \Sigma_1 \mapsto \Sigma_2^1$, ..., $f_n : \Sigma_1 \mapsto \Sigma_2^n$ are TI-abstractions and $f_1 \sqsubseteq ... \sqsubseteq f_n$ ($f_1, ..., f_n$ are totally ordered), then if Σ_2^n is consistent so is Σ_2^i for any $1 \leq i \leq n$.*

Theorem 8 suggests the following process:

- build sets of abstractions, $F_i = \{f_1^i, ..., f_{n_i}^i\}$ where $f_1^i \sqsubseteq ... \sqsubseteq f_{n_i}^i$ and $f_{n_i}^i(\Sigma_1)$ is decidable (*eg.* it is propositional).
- find a set, F_j in which the codomain of the strongest abstraction $f_{n_j}^j(\Sigma_1)$ is consistent. Note that, since $f_{n_j}^j(\Sigma_1)$ is decidable, its consistency or inconsistency can be proved in a finite amount of time.
- starting with the strongest abstraction (that is with $l = n_j$), until $l > 1$ use the proof that the abstracted wff is a theorem in $f_l^j(\Sigma_1)$ to help construct a proof in $f_{l-1}^j(\Sigma_1)$. If, in any of the $f_l^j(\Sigma_1)$, the abstracted wff is not a theorem, then the wff cannot be a theorem in Σ_1 (since f_l^j is a TI-abstraction).

Of course there is no guarantee that all the steps in unabstracting back to Σ_1 will go through or terminate. The overall performance depends on how the various abstractions in the total order are built and on how the process of unabstracting is performed. For instance, computing the consistency of $f_{n_j}^j(\Sigma_1)$ can be optimized by building a very simple, "minimal" $f_{n_j}^j(\Sigma_1)$. Further time can also be saved, when $f_{n_j}^j(\Sigma_1)$ is proved inconsistent by introducing (in an automated way) small variations in $f_{n_j}^j$ that are tuned to the source of the inconsistency.

5 Conclusions

In this paper we have proposed a theory of abstraction which extends the notions of abstraction previously used. We have focused on abstract theorem proving and have suggested that a certain class of provability preserving abstractions, TI- and NTI-abstractions (which are not TC and NTC) are the correct abstractions to use. TC- and NTC- abstractions are in general too strong, and the goal of having "simpler" abstract proofs does not seem achievable except in very special and limited forms (for instance, if $f : \Sigma_1 \mapsto \Sigma_2$ is a TC-/ NTC- abstraction then if Σ_1 is undecidable then Σ_2 cannot be decidable). The dual class of provability preserving abstractions, TD- (and NTD-) abstractions (which are not TC- and NTC-) are of less use as they lose completeness; that is, there is at least one theorem whose abstraction is not a theorem. Unfortunately, TI- (and NTI-) abstractions are subject to the false proof problem; they can map a consistent theory into an inconsistent abstract theory. The last section has proposed a new (and decidable) solution to this problem.

References

[Giunchiglia and Walsh, 1990] F. Giunchiglia and T. Walsh. A Theory of Abstraction. Research paper 516, Dept. of Artificial Intelligence, University of Edinburgh, 1990. Accepted to Journal of Artificial Intelligence.

[Hobbs, 1985] J.R. Hobbs. Granularity. In *Proceedings of the 9th IJCAI*, pages 432–435. International Joint Conference on Artificial Intelligence, 1985.

[Newell *et al.*, 1963] A. Newell, J.C. Shaw, and H.A. Simon. Empirical explorations of the logic theory machine. In Fiegenbaum and Feldman, editors, *Computers & Thought*, pages 134–152. McGraw-Hill, 1963.

[Plaisted, 1981] D.A. Plaisted. Theorem proving with abstraction. *Artificial Intelligence*, 16:47–108, 1981.

[Sacerdoti, 1974] E.D. Sacerdoti. Planning in a hierarchy of abstraction spaces. *Artificial Intelligence*, 5:115–135, 1974.

[Tenenberg, 1987] J.D. Tenenberg. Preserving Consistency across Abstraction Mappings. In *Proceedings of the 10th IJCAI*, pages 1011–1014. International Joint Conference on Artificial Intelligence, 1987.