ELSEVIER

Artificial Intelligence 102 (1998) 163–186

Artificial Intelligence

Value minimization in circumscription ☆

Chitta Baral [a,1], Alfredo Gabaldon [a,*], Alessandro Provetti [b,2]

[a] *Department of Computer Science, University of Texas at El Paso, El Paso, TX 79968, USA*

[b] *D.S.I., Università di Milano, Via Comelico 39, I-20135 Milan, Italy*

Received 2 October 1996; received in revised form 7 May 1998

Abstract

Minimization in circumscription has focussed on minimizing the extent of a set of predicates (with or without priorities among them), or of a formula. Although functions and other constants may be left varying during circumscription, no earlier formalism to the best of our knowledge minimized functions. In this paper we introduce and motivate the notion of *value minimizing* a function in circumscription. Intuitively, value minimizing a function consists in choosing those models where the value of the function is minimal relative to an ordering on its range.

We first give the formulation of value minimization of a single function based on a syntactic transformation and then give a formulation in model-theoretic terms. We then discuss value minimization of a set of functions with and without priorities. We show how Lifschitz's Nested Abnormality Theories can be used to express value minimization, and discuss the prospect of its use for knowledge representation, particularly in formalizing reasoning about actions. © 1998 Published by Elsevier B.V. All rights reserved.

Keywords: Value minimization; Circumscription; Nested abnormality theories; Actions

1. Introduction and motivation

Circumscription [7,10] is one of the earliest logical formalisms used for representing common-sense knowledge. Since it was proposed by McCarthy [9], many extensions of the original circumscription have been proposed. The main technique used in circumscription is *minimization*, which has been mainly used in minimizing the extent

☆ A preliminary version of this work appears in the proceedings of KR-96. Originally submitted as a Research Note.

* Corresponding author. Email: alfredo@cs.utep.edu.

[1] Email: chitta@cs.utep.edu.

[2] Email: provetti@dsi.unimi.it.

PII: S0004-3702(98)00051-4

of a set of predicates—with or without priorities among them, or of a formula. During the minimization some of the other predicates, functions and constants may be allowed to vary. For example, in Baker's solution [3] to the frame problem in theories of action, the function *Result* is varied while the effect of actions are minimized. But, to the best of our knowledge, the idea of *minimizing functions* is not discussed in the circumscription literature.

By minimization of functions we *do not* mean minimizing the extent of a function—the set of inputs where the function is defined. *We mean minimizing the value of the function.* Intuitively, value minimizing a function consists in choosing those models where the value of the function is minimal relative to an ordering on its range.

We first considered minimizing functions when trying to translate specifications in the action description language $\mathcal{L}$ into circumscriptive theories [1]. There, we introduced a function *Sit_map* that maps situations into sequences of actions; these sequences, intuitively, represent the history of the domain up to the situation itself. Assuming that no action has happened unless so suggested by the observations, corresponds to requiring each sequence of actions in the range of *Sit_map* to be *minimal,* meaning that such sequences *cannot be shortened* by dropping some of its elements. This notion of minimality induces a partial order on the space of action sequences. We used this ordering to *minimize Sit_map.*

Even though value minimization of functions and terms is well-studied in conventional linear programming and other areas, we need to formalize it in circumscription to be able to either capture common-sense assumptions—like in the use of *Sit_map* described above, or combine a standard minimum cost criterion with a representation of common-sense knowledge, like in the example discussed below.

1.1. Formalizing flying with the cheapest fare

Often we would like to minimize the cost of our ticket, when flying from one place to another, but subject to certain restrictions. These restrictions could be common-sense restrictions and/or may need representation of defaults. Assume that we have the following predicates and functions:

- *flies*:
  Intuitively, *flies*$(AA, Dallas, Zurich, 101)$ means that AA (American Airlines) has a flight from Dallas to Zurich with the flight number 101.
- *schedule*; *pref_schedule*:
  Intuitively, *schedule*$(101, 10, 22, 1)$ means that flight 101 leaves at 10:00 and reaches its destination at 22:00 and has 1 stop in between.
  Intuitively, *pref_schedule*$(101, 10, 22, 1)$ means that flight 101 which leaves at 10:00 and reaches its destination at 22:00 and has 1 stop in between is one of the preferred ones.
- *lessthaneq*:
  Intuitively, *lessthaneq*(X, Y) means X is less than or equal to Y.
- *cost*:
  Intuitively, *cost*$(101) = 500$ means that the cost of taking flight 101 is \$500.

- *minfare*:
  Intuitively, $minfare(X, Y) = Z$ means that the minimum fare for traveling between X and Y while satisfying the preferences is Z.

Consider the many-sorted theory $T_{lessthaneq}$ that contains the domain closure assumption, the unique names assumption,[3] the facts below and some additional formulas:

$$\begin{array}{l} \mathit{flies}(AA, \mathit{Dallas}, \mathit{Zurich}, 101) \\ \mathit{flies}(AA, \mathit{Dallas}, \mathit{Zurich}, 134) \\ \mathit{flies}(\mathit{Delta}, \mathit{Dallas}, \mathit{Zurich}, 268) \\ \mathit{flies}(\mathit{Continental}, \mathit{Dallas}, \mathit{Zurich}, 361) \\ \mathit{flies}(\mathit{United}, \mathit{Dallas}, \mathit{Zurich}, 471) \\ \mathit{flies}(\mathit{Swissair}, \mathit{Dallas}, \mathit{Zurich}, 531) \\ \mathit{flies}(\mathit{Swissair}, \mathit{Dallas}, \mathit{Zurich}, 561) \\ \mathit{flies}(\mathit{AirIndia}, \mathit{Dallas}, \mathit{Mumbai}, 721) \\ \mathit{flies}(AA, \mathit{Dallas}, \mathit{Mumbai}, 745) \end{array}$$

$$\begin{array}{l} \mathit{schedule}(101, 10, 22, 1) \\ \mathit{schedule}(134, 12, 19, 0) \\ \mathit{schedule}(268, 8, 14, 0) \\ \mathit{schedule}(361, 16, 1, 0) \\ \mathit{schedule}(471, 20, 4, 0) \\ \mathit{schedule}(531, 22, 6, 0) \\ \mathit{schedule}(561, 19, 6, 1) \\ \mathit{schedule}(721, 4, 22, 1) \\ \mathit{schedule}(745, 1, 22, 1) \end{array}$$

$$\begin{array}{l} \mathit{cost}(101) = 500 \\ \mathit{cost}(134) = 550 \\ \mathit{cost}(268) = 600 \\ \mathit{cost}(361) = 620 \\ \mathit{cost}(471) = 580 \end{array}$$

[3] An example of a domain closure axiom for this domain is:

$$(\forall \mathit{flt\#}).\mathit{flt\#} = 101 \vee \cdots \vee \mathit{flt\#} = 561.$$

And an example of a unique names assumption axiom is:

$$AA \neq \mathit{Delta} \neq \mathit{Continental} \neq \mathit{United} \neq \mathit{Swissair}$$

We further discuss these assumptions in Section 2.

$$cost(531) = 630$$
$$cost(561) = 520$$
$$cost(721) = 1100$$
$$cost(745) = 1300$$

$lessthaneq(x, y)$: $x \leqslant y$ and $x, y \in \{0, 500, \ldots, 1300\}$.

We would now like $T_{lessthaneq}$ to contain a definition of *pref_schedule*, as those schedules which leave on or after 6 PM and reach on or after 10 AM, and are non-stop flights. This can be expressed by the following formula:

$$\begin{aligned} & pref\_schedule(flt\#, start, reach, stops) \equiv \\ & \quad [schedule(flt\#, start, reach, stops) \wedge \\ & \quad 18 \leqslant start \leqslant 22 \wedge 4 \leqslant reach \leqslant 10 \wedge stops = 0]. \end{aligned} \tag{1}$$

Note that although the definition of *pref_schedule* is kept as simple as possible, it could be more complicated by involving defaults such as:

> *Normally, John prefers evening non-stop flights that reach in the morning; some exceptions are: the flight does not have a window seat available; it costs over* \$300 *more than a similar* 1*-stop flight and so on.*

To formalize these defaults and their exceptions in an elaboration-tolerant manner, circumscription or another knowledge representation language is needed.

Now the goal is to define *minfare* properly. First, we express what property (besides the value minimization) is satisfied by *minfare*. This is expressed as follows:

$$\begin{aligned} & minfare(from, to) = amount \supset \\ & \quad \exists airline, flt\#, start, reach, stops \\ & \qquad [flies(airline, from, to, flt\#) \wedge \\ & \qquad pref\_schedule(flt\#, start, reach, stops) \wedge cost(flt\#) = amount]. \end{aligned} \tag{2}$$

Let us include this formula also in $T_{lessthaneq}$. Now, performing value minimization of the function *minfare* in the theory $T_{lessthaneq}$ should give us the minimum fare between two places such that there is a flight between them with that fare and with a preferred schedule. Intuitively, in our example, we shall obtain $minfare(Dallas, Zurich) = 580$.

1.2. Organization of the rest of the paper

We will first—in Section 2—give an explicit (syntactic) definition of value minimization and then follow it up with a model-theoretic definition. We will then complete the fare minimization example by specifying the value minimization of *minfare*. In Section 3 we will show how value minimization of functions can be expressed using the standard minimization of *ab* predicates in Lifschitz's Nested Abnormality Theories. In Section 4 we will show how value minimization is useful in formalizing reasoning about actions and narratives. We will then discuss value minimization of multiple functions (in Section 5) and functions encoded as predicates (in Section 6).

2. Value minimization

2.1. Value minimization through syntactic transformation

This section starts with the definition of value minimization—through a syntactic transformation—in most general terms and then proceeds to discuss special versions that are of interest. We will also give an alternative formulation of value minimization in terms of model preference, and use the latter in our proofs.

Here and in the following sections we adopt the following notation: let $T(f; z)$ be a theory where a function symbol f and function/predicate symbols in tuple z appear as free variables. Let $T_{\mathcal{R}}$ be a theory where a *categorical* partial order relation $\mathcal{R}$ is defined, i.e., a predicate $\mathcal{R}$ which captures a partial order on the elements of the universe and is satisfied by only one interpretation. Thence, in the context of a statement involving $T_{\mathcal{R}}$ we can write $\mathcal{R}(x, y)$ with no ambiguity. $\mathcal{R}$ also induces an ordering on function symbols, defined by the following abbreviation (although we write $F(x)$, the arity of F can be greater than one):

$$F \leqslant_{\mathcal{R}} F' \quad \text{stands for} \quad \forall x \mathcal{R}\big(F(x), F'(x)\big)$$

whereas $<_{\mathcal{R}}$ stays for the irreflexive version of $\leqslant_{\mathcal{R}}$; note that this convention is also adopted for other orderings introduced in the rest of the paper. Now it is possible to give the direct definition of value minimization.

Definition 1. The value minimization of a function F in theory $T_{\mathcal{R}}$ with symbols in tuple Z varied is defined as follows:

$$T_{\mathcal{R}}(F; Z) \wedge \neg\exists f, z\big[T_{\mathcal{R}}(f; z) \wedge f <_{\mathcal{R}} F\big]. \tag{3}$$

Before proceeding further, let us comment on the assumptions that are imposed on the structure of the theories $T_{\mathcal{R}}$ which are considered here. In order to compare functions on the base of the values in their range, for each element of the domain there must be a term of $T_{\mathcal{R}}$ which is interpreted onto it; Reiter [11] terms it *domain closure assumption.* Additionally, it can be assumed—although this is not required for $T_{\mathcal{R}}$—that any two constant symbols always denote different elements of the domain, this is termed *unique names assumption.* A theory is said to have *an explicit domain* if it contains axioms for both domain closure and unique names assumptions. Practically, the explicit domain restricts models to domains which are isomorphic to Herbrand's. This approach, which is common in logic programming and deductive databases, seems to us acceptable for our aims.

2.1.1. Term minimization

The value of a function with respect to a finite number of ground terms can also be minimized. For example, if we are only interested in traveling from Dallas to Zurich, we only need to minimize the term *minfare*(*Dallas*, *Zurich*).

We use $\leqslant_{\mathcal{R},t_1,\ldots,t_n}$ to order functions according to $\mathcal{R}$ and their values in $t_1,\ldots,t_n$:

$$F \leqslant_{\mathcal{R},t_1,\ldots,t_n} F' \quad \text{stands for} \quad \bigwedge_{i=1,\ldots,n} \mathcal{R}\big(F(t_i), F'(t_i)\big).$$

Minimization of F relative to terms $t_1,\ldots,t_n$ remains defined as:

$$T_{\mathcal{R}}(F; Z) \wedge \neg\exists f, z\big[T_{\mathcal{R}}(f; z) \wedge f <_{\mathcal{R},t_1,\ldots,t_n} F\big].$$

When explicit domains are considered and $Z = \emptyset$, i.e., nothing is left varying, there is a simpler formula that avoids second-order quantifiers. In fact, let $\bar{c}$ stand for a tuple of object variables $c_1,\ldots,c_k$, where k is the number of terms relative to which we want to minimize F, and let $T_{\mathcal{R}}(\bar{c})$ stand for $T_{\mathcal{R}} \bigwedge_{i=1,\ldots,k} F(t_i) = c_i$. Term minimization is then expressed by:

$$(\forall \bar{c}) T_{\mathcal{R}}(\bar{c}) \supset \left[\forall \bar{c}'\left(T_{\mathcal{R}}(\bar{c}') \bigwedge_{i=1,\ldots,k} \mathcal{R}(c'_i, c_i)\right) \supset \left(\bigwedge_{i=1,\ldots,k} \mathcal{R}(c_i, c'_i)\right)\right].$$

Example 1. Consider the theory $T_{lessthaneq}$ from Section 1. We can now express value minimization of the function *minfare* by the following:

$$T_{lessthaneq}(minfare) \wedge \neg\exists f\big[T_{lessthaneq}(f) \wedge f <_{lessthaneq} minfare\big].$$

Also, value minimization of the term *minfare(Dallas, Zurich)* can be expressed by the following simpler formula:

$$\forall\alpha . T_{lessthaneq}(\alpha) \supset \forall\alpha'\big[T_{lessthaneq}(\alpha') \wedge lessthaneq(\alpha', \alpha) \supset lessthaneq(\alpha, \alpha')\big]$$

where $T_{lessthaneq}(\alpha)$ is defined as

$$T_{lessthaneq} \wedge minfare(Dallas, Zurich) = \alpha.$$

It is easy to see that the resultant theory entails

$$minfare(Dallas, Zurich) = 580.$$

2.2. Model-theoretic definition

In this section we define value minimization in model-theoretic terms.[4] The idea is to define an ordering on models based on their interpretation of function symbols and to strengthen the entailment by considering only models that are minimal with respect to this ordering. Let us start by introducing the following notation. Recall that a structure $\mathcal{I}$ for a language is determined by its universe $|\mathcal{I}|$ and by the interpretations $\mathcal{I}[[C]]$ of all individual, function and predicate constants C in the language.

In Definition 2 below and in the following we refer to an ordering $\mathcal{R}$ which is *external to the theory*, in other words, we refer to the ordering with respect to which we intend to do the minimization, as opposed to its axiomatization within the theory.[5] In terms of the airfares

[4] Please refer to Lifschitz [7] and Shoham [13] for a general discussion and results on preferential models.

[5] This is possible in presence of (i) explicit-domain property (each element of the domain has a name) and (ii) categoricity of the ordering defined within the theory.

example above, the intended external ordering is the usual $\leqslant$ relation on numbers, and its axiomatization is captured by predicate *lessthaneq*. Although we use the same symbol $\mathcal{R}$ to refer to the ordering and to the predicate of its axiomatization, in the former case we use $\mathcal{R}$ with infix notation.

Definition 2 (*Value-minimal*). Let T be a theory, F be a function and Z a tuple of predicate/function constants in the language of T. Let $\mathcal{R}$ [6] be a partial order defined over the elements of the universe. For two models $\mathcal{M}$ and $\mathcal{M}'$ of T, we say $\mathcal{M} \leqslant^{(F;\mathcal{R});Z} \mathcal{M}'$ if

(i) $|\mathcal{M}| = |\mathcal{M}'|$;
(ii) $\mathcal{M}[[\sigma]] = \mathcal{M}'[[\sigma]]$ for each constant σ s.t. $\sigma \neq F$, $\sigma \notin Z$;
(iii) for all x, $\mathcal{M}[[F]](x)\, \mathcal{R}\, \mathcal{M}'[[F]](x)$.

A model $\mathcal{M}$ of T is minimal relative to $\leqslant^{(F;\mathcal{R});Z}$ if there is no model $\mathcal{M}'$ of T such that $\mathcal{M}' <^{(F;\mathcal{R});Z} \mathcal{M}$.

In the case of *term minimization*, condition (iii) in Definition 2 becomes

$$\mathcal{M}[[F]](t_i)\, \mathcal{R}\, \mathcal{M}'[[F]](t_i) \quad (i = 1, \ldots, k)$$

and we write minimality as relative to the ordering $\leqslant^{(F;t_1,\ldots,t_n,\mathcal{R});Z}$ between models. The following example is not about common sense but useful for illustrating the value-minimal ordering on models.

Example 2 (*Exponentiation*). Consider the following two interpretations of function *power* on the domain of naturals with $\leqslant$ as the usual 'less than or equal' relation. By abuse of notation, we define:

$$\mathcal{M}[[power]](x) = x^2, \quad \mathcal{M}'[[power]](x) = x^3.$$

Of course, $\mathcal{M}[[power]](x) \leqslant \mathcal{M}'[[power]](x)$ and it is easy to establish $\mathcal{M} <^{(power,\leqslant)} \mathcal{M}'$.

Finally, let us proceed to state that models of formula (3) are all and only those which are minimal with respect to $\leqslant^{(F;\mathcal{R});Z}$. This result is the counterpart of fundamental Proposition 2.5.1 of [7].

Proposition 1. *An interpretation $\mathcal{M}$ is a model of the value minimization formula* (3):

$$T_{\mathcal{R}}(F;Z) \wedge \neg\exists f, z\big[T_{\mathcal{R}}(f,z) \wedge f <_{\mathcal{R}} F\big]$$

iff $\mathcal{M}$ is a model of $T_{\mathcal{R}}(F;Z)$ and it is minimal relative to $\leqslant^{(F;\mathcal{R});Z}$.

Proof (*Sketh*). A model $\mathcal{M}$ of $T_{\mathcal{R}}(F;Z)$ is minimal relative to $\leqslant^{(F;\mathcal{R});Z}$ iff there is no other model $\mathcal{M}'$ of $T_{\mathcal{R}}(F;Z)$ which differs from $\mathcal{M}$ only in the interpretation of F and the constants in Z, and the following holds: for all x,

$$\mathcal{M}'[[F]](x)\, \mathcal{R}\, \mathcal{M}[[F]](x))$$

[6] In this definition, ordering $\mathcal{R}$ is external to the theory, i.e., $\mathcal{R}$ is not a predicate defined in the theory.

and it is not the case that for all x,

$$\mathcal{M}[[F]](x)\,\mathcal{R}\,\mathcal{M}'[[F]](x)).$$

This is exactly what (3) specifies. □

3. Value minimization using nested abnormality theories

Nested Abnormality Theories (NATs) is a novel circumscription [7,10] technique introduced by Lifschitz [8]. With NATs it is possible to circumscribe several predicates each with respect to only parts of the theory of interest, as opposed to previous techniques such as parallelized and circumscription theories where the circumscription must be done with respect to all of the axioms in the underlying theory (a similar approach was earlier suggested in [12]). Furthermore, complications arising from the interaction of multiple circumscription axioms in a theory can be avoided in NATs with the introduction of blocks. A *block* is characterized by a set of axioms $A_1, \ldots, A_n$—possibly containing the abnormality predicate Ab—which "describe" a set of predicate/function constants $C_1, \ldots, C_m$. The notation for such a theory is

$$\{C_1, \ldots, C_m : A_1, \ldots, A_n\} \tag{4}$$

where each A_i may itself be a block of form (4). The "description" of $C_1, \ldots, C_m$ by a block may depend on other descriptions in embedded blocks.

Interference between circumscription in different blocks is prevented by replacing a predicate Ab with an existentially quantified variable. Lifschitz's idea is to make Ab "local" to the block where it is used, since abnormality predicates play only an auxiliary role, i.e., the interesting consequences of the theory are those which do not contain Ab.

In this section we will show how value minimization can be done by using predicate circumscription within the framework of NATs. The use of NATs is particularly important because it allows us to define the ordering $\mathcal{R}$ in an independent block which will not interfere with the rest of the theory, thus achieving a simpler and better-structured formulation.[7] A formalization of $\mathcal{R}$ and value minimization in a form of circumscription other than NATs will require a careful assessment of priorities of predicates (which are minimized) to avoid undesirable side effects.

Before we show how value minimization can be done using NATs we give a quick overview of NATs and formal definitions of some of the concepts.

3.1. Overview of nested circumscription

The following definitions are from [8]. Let L be a second-order language which does not include Ab. For every natural number k, let L_k be the language obtained by adding the k-ary predicate constant Ab to L. $\{C_1, \ldots, C_m : A_1, \ldots, A_n\}$ is a *block* if each $C_1, \ldots, C_m$ is a function or predicate constant of L, and each $A_1, \ldots, A_n$ is a formula of L_k or a block.

[7] Recall that in Section 2.1 we assumed that the definition of $\mathcal{R}$ is part of $T_{\mathcal{R}}(F; Z)$.

A *Nested Abnormality Theory* is a set of blocks. The semantics of NATs is characterized by a mapping φ from blocks into sentences of L. If A is a formula of language L_k, φA stands for the universal closure of A, otherwise

$$\varphi\{C_1,\ldots,C_m : A_1,\ldots,A_n\} = \exists ab F(ab)$$

where

$$F(Ab) = \mathrm{CIRC}[\varphi A_1 \wedge \cdots \wedge \varphi A_n; Ab; C_1,\ldots,C_m].$$

Recall that CIRC[T; P; Q], means circumscription of the theory T, by minimizing the predicates in P, and varying the objects in Q.

For any NAT T, φT stands for $\{\varphi A \mid A \in T\}$. A *model* of T is a model of φT in the sense of classical logic. A *consequence* of T is a sentence ϕ of language L that is true in all models of T. In this paper, as suggested in [8], we use the abbreviation

$$\{C_1,\ldots,C_m, \min P : A_1,\ldots,A_n\}$$

to denote blocks of the form

$$\{C_1,\ldots,C_m, P : P(x) \supset Ab(x), A_1,\ldots,A_n\}.$$

As the notation suggests, this type of block is used when it is necessary to circumscribe a particular predicate P in a block. In [8] it is shown that

$$\varphi\{C_1,\ldots,C_m, \min P : A_1,\ldots,A_n\}$$

is equivalent to the formula

$$\mathrm{CIRC}[A_1 \wedge \cdots \wedge A_n; P; C_1,\ldots,C_m]$$

when each A_i is a sentence.

Example 3. The transitive closure of a binary predicate P is defined by the following block:

{min P:

$P(x,y) \wedge P(y,z) \supset P(x,z)$

other axioms involving P ...

}

This block cannot be simplified into an equivalent first-order theory.

Example 4. Given an alphabet of action names (defined by property *action*), let a generic plan be just a sequence of actions. The unary predicate *Sequence* and function constant '$\cdot$' characterize generic plans by means of this block:

{min *Sequence*:

$\mathit{Sequence}(\varepsilon)$

$\mathit{Action}(a) \wedge \mathit{Sequence}(\alpha) \supset \mathit{Sequence}(a \cdot \alpha)$

}

3.2. NAT formulation of value minimization

The intuitive idea behind formulating value minimization using NATs is that for value minimizing a function F whose range is ordered by $\mathcal{R}$, we define the Ab predicate such that if $\mathcal{R}(y, F(x))$ for any x and y, then $Ab(x, y)$ is true. Thus when we minimize Ab, we force F to have the minimum value (with respect to $\mathcal{R}$) for all inputs. The following theorem formally states how for each given minimization criterion $\leqslant^{(F;\mathcal{R});Z}$ an equivalent NAT formulation can be constructed.

Theorem 1 (Value-minimal equivalence). *Let T be a NAT that includes the definition of ordering $\mathcal{R}$, and let F be a function constant and Z be a tuple of predicate/function constants in the language of T. Then, for the following NAT:*

$$\left.\begin{array}{l} \{F, Z : \\ \qquad \left.\begin{array}{l} \forall x, y.\ \mathcal{R}(y, F(x)) \supset Ab(x, y) \\ T \end{array}\right\} T_{Ab} \\ \} \end{array}\right\} T_{F,\mathcal{R}}$$

(i) *If $\mathcal{M}$ is a model of T minimal relative to $\leqslant^{(F;\mathcal{R});Z}$ then the interpretation $\mathcal{M}_{Ab}$ obtained by augmenting $\mathcal{M}$ with a predicate Ab defined as follows:*

$$\mathcal{M}_{Ab}[[Ab]] = \big\{(x, y)\colon\ (y, F(x)) \in \mathcal{M}[[\mathcal{R}]]\big\} \tag{5}$$

is a model of $T_{F,\mathcal{R}}$.

(ii) *If $\mathcal{M}_{Ab}$ is a model of $T_{F,\mathcal{R}}$ then the interpretation $\mathcal{M}$ obtained from $\mathcal{M}_{Ab}$ by dropping the extent of Ab is a model of T minimal relative to $\leqslant^{(F;\mathcal{R});Z}$.*

Proof. Consider a theory T as described above.

(i) Let $\mathcal{M}$ be a model of T minimal relative to $\leqslant^{(F;\mathcal{R});Z}$. Let $\mathcal{M}_{Ab}$ be an interpretation obtained as described above.

Clearly, $\mathcal{M}_{Ab}$ is a model of T_{Ab}. Thus, to prove that $\mathcal{M}_{Ab}$ is a model of $T_{F,\mathcal{R}}$ it remains to show that the extent of Ab is minimal, i.e., that there is no model $\mathcal{M}'$ of T_{Ab} s.t.

(a) $|\mathcal{M}'| = |\mathcal{M}_{Ab}|$,

(b) $\mathcal{M}'[[\sigma]] = \mathcal{M}_{Ab}[[\sigma]]$ for every constant σ different from Ab and F and not belonging to Z, and

(c) $\mathcal{M}'[[Ab]] \subset \mathcal{M}_{Ab}[[Ab]]$.

Suppose there is such an $\mathcal{M}'$.

Since $\mathcal{R}$ is reflexive, we have that for all x, $\mathcal{M}'[[F]](x)\,\mathcal{R}\,\mathcal{M}'[[F]](x)$. Then, by the abnormality axiom in T_{Ab} we get that for all x, $(x, \mathcal{M}'[[F]](x)) \in \mathcal{M}'[[Ab]]$. From this and (c) it follows that:

$$\text{for all } x,\ \big(x, \mathcal{M}'[[F]](x)\big) \in \mathcal{M}_{Ab}[[Ab]].$$

By (5), this implies that

$$\text{for all } x,\ \mathcal{M}'[[F]](x)\,\mathcal{R}\,\mathcal{M}_{Ab}[[F]](x)$$

and therefore that

$$\text{for all } x,\ \mathcal{M}'[[F]](x)\,\mathcal{R}\,\mathcal{M}[[F]](x).$$

By Definition 2, this, (a) and (b) imply that

$$\mathcal{M}' \leqslant^{(F;\mathcal{R});Z} \mathcal{M}. \tag{6}$$

Now, we will show that $\mathcal{M} \not\leqslant^{(F;\mathcal{R});Z} \mathcal{M}'$. Suppose that $\mathcal{M} \leqslant^{(F;\mathcal{R});Z} \mathcal{M}'$. Then,

$$\text{for all } x,\ \mathcal{M}[[F]](x)\,\mathcal{R}\,\mathcal{M}'[[F]](x). \tag{7}$$

By (c), there exist x_0, y_0 s.t. $(x_0, y_0) \in \mathcal{M}_{Ab}[[Ab]]$ and

$$(x_0, y_0) \notin \mathcal{M}'[[Ab]]. \tag{8}$$

By (5), $y_0\ \mathcal{R}\ \mathcal{M}_{Ab}[[F]](x_0)$ holds, and therefore $y_0\ \mathcal{R}\ \mathcal{M}[[F]](x_0)$ holds too. From this, (7), and transitivity of $\mathcal{R}$, we have that $y_0\ \mathcal{R}\ \mathcal{M}'[[F]](x_0)$, and by the abnormality axiom in T_{Ab} we get that $(x_0, y_0) \in \mathcal{M}'[[Ab]]$. This contradicts (8). Therefore our assumption that $\mathcal{M} \leqslant^{(F;\mathcal{R});Z} \mathcal{M}'$ is wrong. From this and (6) we have that $\mathcal{M}' <^{(F;\mathcal{R});Z} \mathcal{M}$. But we said $\mathcal{M}$ is a model of T_{Ab} minimal relative to $\leqslant^{(F;\mathcal{R});Z}$. Therefore, there does not exist such an $\mathcal{M}'$ and $\mathcal{M}_{Ab}$ is a model of $T_{F,\mathcal{R}}$.

(ii) Let $\mathcal{M}_{Ab}$ be a model of $T_{F,\mathcal{R}}$. By definition of NATs, $\mathcal{M}_{Ab}$ is also a model of T_{Ab}. We will show, by contradiction, that the interpretation $\mathcal{M}$, obtained from $\mathcal{M}_{Ab}$ by dropping the interpretation of Ab, is a model of T minimal relative to $\leqslant^{(F;\mathcal{R});Z}$.

Suppose that there is a model $\mathcal{M}'$ of T s.t.

$$\mathcal{M}' <^{(F;\mathcal{R});Z} \mathcal{M}. \tag{9}$$

Let $\mathcal{M}'_{Ab}$ be the interpretation obtained from $\mathcal{M}'$ by adding the following interpretation of predicate Ab:

$$\mathcal{M}'_{Ab}[[Ab]] = \{(x, y):\ (y, F(x)) \in \mathcal{M}'[[\mathcal{R}]]\}. \tag{10}$$

From (9) we have that

$$\text{for all } x,\quad \mathcal{M}'_{Ab}[[F]](x)\ \mathcal{R}\ \mathcal{M}[[F]](x). \tag{11}$$

From this and by transitivity of $\mathcal{R}$ the following implication holds:

$$\text{for all } x, y,\quad y\,\mathcal{R}\,\mathcal{M}'_{Ab}[[F]](x)\ \Rightarrow\ y\,\mathcal{R}\,\mathcal{M}[[F]](x).$$

Since we built $\mathcal{M}$ from $\mathcal{M}_{Ab}$ simply by removing the interpretation of Ab, $y\,\mathcal{R}\,\mathcal{M}[[F]](x)$ iff $y\,\mathcal{R}\,\mathcal{M}_{Ab}[[F]](x)$. Thus we have that

$$\text{for all } x, y,\quad y\,\mathcal{R}\,\mathcal{M}'_{Ab}[[F]](x) \Rightarrow y\,\mathcal{R}\,\mathcal{M}_{Ab}[[F]](x).$$

By the abnormality axiom in T_{Ab}, we have that for all x, y,

$$(x, y) \in \mathcal{M}'_{Ab}[[Ab]] \Rightarrow (x, y) \in \mathcal{M}_{Ab}[[Ab]]$$

and therefore

$$\mathcal{M}'_{Ab}[[Ab]] \subseteq \mathcal{M}_{Ab}[[Ab]]. \tag{12}$$

Now, (9) implies that there is an x_0 for which $\mathcal{M}[[F]](x_0)\,\mathcal{R}\,\mathcal{M}'[[F]](x_0)$ does not hold, and hence $\mathcal{M}_{Ab}[[F]](x_0)\,\mathcal{R}\,\mathcal{M}'[[F]](x_0)$ does not hold either. For such an x_0 we have

(a) $(x_0, \mathcal{M}_{Ab}[[F]](x_0)) \notin \mathcal{M}'_{Ab}[[Ab]]$ by (10).
Further,

$$\text{for all } x, \quad \mathcal{M}_{Ab}[[F]](x)\,\mathcal{R}\,\mathcal{M}_{Ab}[[F]](x)$$

holds by reflexivity of $\mathcal{R}$. Hence, by the abnormality axiom in T_{Ab},

$$\text{for all } x \quad \big(x, \mathcal{M}_{Ab}[[F]](x)\big) \in \mathcal{M}_{Ab}[[Ab]].$$

In particular,
(b) $(x_0, \mathcal{M}_{Ab}[[F]](x_0)) \in \mathcal{M}_{Ab}[[Ab]]$ is true.
From (a) and (b) we have that:

$$\mathcal{M}_{Ab}[[Ab]] \not\subseteq \mathcal{M}'_{Ab}[[Ab]].$$

From this and (12) we have that $\mathcal{M}'_{Ab}[[Ab]] \subset \mathcal{M}_{Ab}[[Ab]]$. From this and since $\mathcal{M}'_{Ab}$ has the same universe as $\mathcal{M}_{Ab}$ and differs from it only in the interpretation of F, Z and Ab, we can conclude that $\mathcal{M}_{Ab}$ is not a model of $T_{F,\mathcal{R}}$. This results in a contradiction. Therefore our assumption that there exists $\mathcal{M}'$ such that (9) holds is wrong and $\mathcal{M}$ is a model of T minimal relative to $\leqslant^{(F;\mathcal{R});Z}$. □

3.2.1. Term minimality

The results discussed so far apply equally well to minimizing the interpretation of one or more ground terms: $F(t_1), F(t_2), \ldots, F(t_n)$. Consider an explicit domain NAT T with term ordering $\mathcal{R}$, and let F be a function constant, $t_1, \ldots, t_n$ be ground terms, and Z be a tuple of predicate/function constants in the language of T. The following result is easily established.

Proposition 2 (Term-minimal equivalence). *An interpretation $\mathcal{M}$ is a model of*[8]

$$\begin{aligned}
&\{F, Z : \\
&\qquad \forall y.\, \mathcal{R}(y, F(t_1)) \supset Ab(t_1, y) \\
&\qquad \ldots \\
&\qquad \forall y.\, \mathcal{R}(y, F(t_n)) \supset Ab(t_n, y) \\
&\qquad T \\
&\}
\end{aligned}$$

if and only if $\mathcal{M}$ is a model[9] *of T minimal relative to $\leqslant^{(F,t_1,\ldots,t_n;\mathcal{R});Z}$.*

---

[8] For $n = 1$ we just have $Ab(y)$ in the right hand side of the axiom.

[9] We are abusing the notation slightly here since the predicate Ab, of which $\mathcal{M}$ includes an interpretation, does not appear in T.

3.3. The minimum fare example using NATs

We now give the formulation of value minimization in the minimum fare example using NATs.

$$\{minfare:$$
$$\forall x1, x2, y.\ lessthaneq(y, minfare(x1, x2)) \supset Ab(x1, x2, y)$$
$$T_{lessthaneq}$$
$$\}$$

For the simpler case of characterizing the minimum fare only between Dallas and Zurich the following simpler NAT is sufficient.

$$\{minfare:$$
$$\forall y.\ lessthaneq(y, minfare(Dallas, Zurich)) \supset Ab(y)$$
$$T_{lessthaneq}$$
$$\}$$

3.4. NAT notation for value minimization

Lifschitz's notation is readily adapted to value minimization. We will write

$$\{C_1, \ldots, C_m, \min_{\mathcal{R}} F : A_1, \ldots, A_n, A_{\mathcal{R}}\} \tag{13}$$

where $A_{\mathcal{R}}$ defines [10] $\mathcal{R}$—to denote blocks of the form

$$\{C_1, \ldots, C_m, F : \mathcal{R}(y, F(x)) \supset Ab(x, y), A_1, \ldots, A_n, A_{\mathcal{R}}\}.$$

Intuitively, block (13) refers to a theory consisting of blocks $A_1, \ldots, A_n$ and the block $A_{\mathcal{R}}$ defining $\mathcal{R}$, and where value minimization of function F is performed while predicate/function constants $C_1, \ldots, C_m$ are varying.

4. Applying value minimization to reasoning about actions

In this section, we discuss a recent application of, and motivation for, value minimization in a full-scale knowledge representation problem, namely, reasoning about actual and hypothetical occurrences of actions.

In [1] we introduced a sound and complete formalization of narratives for domains specified in the high-level action description language $\mathcal{L}$ [2]. In the development of this formalization of narratives, we were faced with the problem of minimizing the value

[10] We are assuming that $A_{\mathcal{R}}$ is defined using an NAT block such that statements about $\mathcal{R}$ outside of $A_{\mathcal{R}}$ do not affect the definition of $\mathcal{R}$.

of a particular term: $Sit\_map(S_N)$, that mapped the current-situation symbol S_N onto a sequence of actions

$$A_k \circ \cdots \circ A_2 \circ A_1 \circ \varepsilon$$

which is understood as "A_1 then A_2 then ...etc." These sequences represent the *history* of the domain up to S_N. Thence, minimizing the sequence of actions assigned to S_N was necessary to formalize the assumption from $\mathcal{L}$ that said: "no actions occurred except those needed to explain the facts in the theory".

We now describe the use of value minimization in reasoning about narratives through the axiomatization of a simple $\mathcal{L}$ domain description. Consider the following simple story.

In the initial situation of the domain, F and G were observed to be false, and we do not know about P, i.e., it may be true or false. At a later moment of time, denoted by S_1, F was observed to be true. We also know that action A causes F and that action B causes G when executed in a situation where P holds. Moreover, action B was observed to occur in situation S_1. This information is described in $\mathcal{L}$ as follows:

$$\left.\begin{array}{ll}
\neg F \textbf{ at } S_0 & \text{(a1)} \\
\neg G \textbf{ at } S_0 & \text{(a2)} \\
F \textbf{ at } S_1 & \text{(a3)} \\
S_0 \textbf{ precedes } S_1 & \text{(a4)} \\
B \textbf{ occurs_at } S_1 & \text{(a5)} \\
A \textbf{ causes } F & \text{(a6)} \\
B \textbf{ causes } G \textbf{ causes } P & \text{(a7)}
\end{array}\right\} = D$$

$\mathcal{L}$ incorporates in its semantics the following assumptions:
- values of fluents change only as a result of the execution of actions;
- there are no actions except those mentioned in the domain description;
- the only effects of actions are those described with "**causes**" axioms;
- actions do not overlap or occur simultaneously;
- no actions occur except those needed to explain the facts.

Given the description and the assumptions, one can intuitively conclude that action A occurred in the initial situation (S_0) causing F to become true; also B occurred, as stated in D, but no other action occurred. Still, this domain description allows for two "models", one in which $\neg P$ holds in S_0 and another where P holds in S_0. In the former model, F, $\neg G$ and $\neg P$ all hold in the current situation S_N; in the latter model, F, G and P hold in S_N.

4.1. Axioms for reasoning about narratives

Instead of presenting the complete characterization of domain description D in NATs as described in [1], in this section we discuss a similar but simpler set of axioms. Although these axioms will be particular to D, they usefully illustrate how value minimization was used in the general formalization of narratives.

We will use T_D to denote the NAT characterization of D. The language of T_D includes sorts for actions, fluents, situations and sequences. Possibly indexed letters a, f and s denote variables of the first three sorts respectively. α, β, γ denote sequence variables. We include constants A, B of sort action, F, G, P of sort fluent, and S_0, S_1, S_N of sort situation. Constants S_0 and S_N are called the initial and the current situations respectively. The special symbol ε of sort sequence denotes the empty sequence. We use a function $\circ : \mathit{actions} \times \mathit{sequence} \to \mathit{sequence}$ to build sequences of actions.

Function *Sit_map* is defined from situation constants to sequences of actions, to assign, intuitively, a history to each actual situation. Intuitively, $\mathit{Sit\_map}(S) = A_k \circ \cdots \circ A_1 \circ \varepsilon$ means that the domain of interest was in situation S after sequence $A_k \circ \cdots \circ A_1 \circ \varepsilon$ was executed in the initial situation. The sequence $\mathit{Sit\_map}(S_N)$ is called the *actual sequence*, since it represents the actual sequence of events from the initial situation to the current situation.

T_D includes axioms for the domain closure assumption and the unique names assumption (similar to the ones in [6]), and the definitions of several common relations on sequences, namely prefix, subsequence and concatenate. Relation *Subsequence* is particularly important in as much as it captures the ordering on sequences relative to which value minimization is done. Intuitively, by $\mathit{Subsequence}(\alpha, \beta)$ we mean that α is a subsequence of β in the usual sense, i.e., by removing some elements from β, while preserving the order, it is possible to obtain α. In the NAT syntax, *Subsequence* can be defined in the following *block*: [11]

$$\{\min \mathit{Subsequence}:$$

$$\mathit{Subsequence}(\alpha, \alpha)$$

$$\mathit{Subsequence}(\alpha, \beta) \supset \mathit{Subsequence}(\alpha, a \circ \beta)$$

$$\mathit{Subsequence}(\alpha, \beta) \supset \mathit{Subsequence}(a \circ \alpha, a \circ \beta)$$

$$\}$$

Blocks for predicates *Prefix_eq* and *Concatenate*, with their usual meaning could be defined in a similar fashion; we will assume T_D includes them. Now, we can proceed with the main part of the theory T_D. We start with some axioms describing the actual situations:

$$\mathit{Sit\_map}(S_0) = \varepsilon$$

$$\mathit{Prefix}(\mathit{Sit\_map}(S_0), \mathit{Sit\_map}(S_1))$$

$$\mathit{Prefix\_eq}(B \circ \mathit{Sit\_map}(S_1), \mathit{Sit\_map}(S_N))$$

By the first axiom, the initial situation is mapped onto the empty sequence. This is one of the assumptions of the language $\mathcal{L}$. The second axiom formalizes (a4). The third axiom formalizes (a5), according to which B *fits in* right after the sequence assigned to S_1. Yet the third axiom does not say whether the overall history of the domain, i.e., $\mathit{Sit\_map}(S_N)$, will be equal to what is in $B \circ \mathit{Sit\_map}(S_1)$ or if it will be a longer sequence.

[11] For readers unfamiliar with the NAT syntax it suffices to refer to our overview in Section 3.1.

The following axioms completely characterize the effects of A and B on F and G. The first two axioms encode the effect of actions A and B as specified by (a6) and (a7), respectively. The rest of the axioms encode inertia.

$$Holds(F, A \circ \alpha)$$

$$Holds(P, \alpha) \supset Holds(G, B \circ \alpha)$$

$$Holds(F, \alpha) \equiv Holds(F, B \circ \alpha)$$

$$Holds(G, \alpha) \equiv Holds(G, A \circ \alpha)$$

$$\neg Holds(P, \alpha) \supset [Holds(G, \alpha) \equiv Holds(G, B \circ \alpha)]$$

$$Holds(P, \alpha) \equiv Holds(P, A \circ \alpha)$$

$$Holds(P, \alpha) \equiv Holds(P, B \circ \alpha)$$

Finally, fluent observations as specified by (a1), (a2) and (a3) are encoded by the following axioms:

$$\neg Holds(F, Sit\_map(S_0))$$

$$\neg Holds(G, Sit\_map(S_0))$$

$$Holds(F, Sit\_map(S_1))$$

To completely capture the intuitive meaning of the story in this example, in addition to the above axioms we need to capture the assumption that nothing else happened except the actions necessary to justify the observations. This is where value minimization comes into play: function *Sit_map* must map S_N into the *minimal* (with respect to the subsequence ordering) possible sequence, compatible with the axioms.

Let us first consider the models of T_D as is, i.e., without minimization. For instance, take the model where $Sit\_map(S_N) = B \circ A \circ \cdots \circ A \circ \varepsilon$ and $Sit\_map(S_1) = A \circ A \circ \cdots \circ A \circ \varepsilon$. Compare the model above with that where $Sit\_map(S_N) = B \circ A \circ \varepsilon$ and $Sit\_map(S_1) = A \circ \varepsilon$. In the former model we see several occurrences of A which are *not sanctioned* by the observations. Clearly, minimization of the actual sequence makes the latter model preferred over the former.

4.2. *Value minimization of the actual sequence*

Let $T_D(\alpha)$ denote

$$T_D \wedge Sit\_map(S_N) = \alpha.$$

According to the earlier definition, $Sit\_map(S_N)$ is minimized relative to *Subsequence* by postulating:

$$\forall \alpha T_D(\alpha) \supset \forall \alpha' \big[(T_D(\alpha') \wedge Subsequence(\alpha', \alpha)) \supset Subsequence(\alpha, \alpha')\big]. \tag{14}$$

Let us denote (14) by $\mathcal{T}$. It is easy to see that:

$$\mathcal{T} \models Sit\_map(S_0) = \varepsilon$$

$$\mathcal{T} \models Sit\_map(S_1) = A \circ \varepsilon$$

$$\mathcal{T} \models Sit\_map(S_N) = B \circ A \circ \varepsilon$$

In other words, $\mathcal{T}$ entails that only A and B have occurred, in this order. As far as predicate *Holds*, $\mathcal{T}$ allows two models, mirroring the fact that P is "free" in the initial situation:

$\mathcal{M}_1$	$\mathcal{M}_2$
$\neg Holds(P, Sit\_map(S_0))$	$Holds(P, Sit\_map(S_0))$
$Holds(F, Sit\_map(S_1))$	$Holds(F, Sit\_map(S_1))$
$\neg Holds(G, Sit\_map(S_N))$	$Holds(G, Sit\_map(S_N))$

4.3. The NAT formulation

Example 5. By applying Proposition 2 to theory $\mathcal{T}$ from Section 4.2, we obtain the following NAT formalization:

$$\{Sit\_map:$$

$$Subsequence(\alpha, Sit\_map(S_N)) \supset Ab(\alpha)$$

$$T_D$$

$$\}$$

It is obvious that in the above NAT we accomplish the value minimization of $Sit\_map(S_N)$ by minimizing the predicate Ab. Indeed, suppose an interpretation $\mathcal{I}$ satisfies all the axioms of T_D, and maps $Sit\_map(S_N)$ on the sequence of actions α, while interpretation $\mathcal{I}'$ maps $Sit\_map(S_N)$—other things being equal—onto a super-sequence β of α. As a result, the extent of Ab under $\mathcal{I}$ is a proper subset of the extent of Ab under $\mathcal{I}'$. Therefore, $\mathcal{I}'$ is not a circumscriptive model of the above NAT.

5. Value minimization of multiple functions

In this section we generalize value minimization to the minimization of a collection of functions with or without priorities among them. As in case of parallel and prioritized minimization of predicates, we also often face the necessity of minimizing multiple functions.

5.1. Parallel value minimization

An example of the use of parallel value minimization is of course the combination of our two main examples: a narrative involving the action of traveling from one place to another using the minimum fare ticket.

Here formalizing the narrative would involve minimizing *Sit_map* and formalizing the minimum fare would involve minimizing *minfare*, and since neither has a priority over the other, we would need to minimize them in parallel. We now formally describe parallel value minimization of multiple functions.

Let $\mathcal{R}_1, \ldots, \mathcal{R}_n$ be orderings on the elements of the domain of interest. Let $\vec{F}$ stand for a tuple of function constants $F_1, \ldots, F_n$, and let

$$\vec{F} \leqslant_{\mathcal{R}_1,\ldots,\mathcal{R}_n} \vec{F}' \quad \text{stand for} \quad F_1 \leqslant_{\mathcal{R}_1} F_1' \wedge \cdots \wedge F_n \leqslant_{\mathcal{R}_n} F_n'.$$

Parallel value minimization of a tuple of functions $\vec{F}$ in theory $T_{\mathcal{R}_1,\ldots,\mathcal{R}_n}$ relative to orderings $\mathcal{R}_1, \ldots, \mathcal{R}_n$ with Z varied is explicitly expressed as follows:

$$T_{\mathcal{R}_1,\ldots,\mathcal{R}_n}(\vec{F}, Z) \wedge \neg\exists \vec{f}, z\big[T_{\mathcal{R}_1,\ldots,\mathcal{R}_n}(\vec{f}, z) \wedge \vec{f} <_{\mathcal{R}_1,\ldots,\mathcal{R}_n} \vec{F}\big]. \tag{15}$$

Definition 3. Let T be a theory with functions $\vec{F}$ and orderings $\mathcal{R}_1, \ldots, \mathcal{R}_n$ on the elements of its universe. For two interpretations $\mathcal{M}_1$ and $\mathcal{M}_2$ of T, we say

$$\mathcal{M}_1 \leqslant^{(\vec{F};\mathcal{R}_1,\ldots,\mathcal{R}_n);Z} \mathcal{M}_2$$

if

- $|\mathcal{M}_1| = |\mathcal{M}_2|$,
- $\mathcal{M}_1[[\sigma]] = \mathcal{M}_2[[\sigma]]$ for each constant σ which does not belong to $\vec{F}$ and Z,
- for all x, $\mathcal{M}_1[[F_i]](x)\, \mathcal{R}_i\, \mathcal{M}_2[[F_i]](x)$ for each function F_i in $\vec{F}$ and ordering $\mathcal{R}_i$.

Proposition 1 is straightforwardly generalized to parallel value minimization, thus the following two propositions can be proved.

Proposition 3. *An interpretation $\mathcal{M}$ is a model of (15) iff $\mathcal{M}$ is a model of $T_{\mathcal{R}_1,\ldots,\mathcal{R}_n}(\vec{F}, Z)$ and it is minimal relative to $\leqslant^{(\vec{F},\mathcal{R}_1,\ldots,\mathcal{R}_n);Z}$.*

Again, we describe how to embody T into a NAT whose models coincide with those models of T which are minimal relative to $\leqslant^{(\vec{F};\mathcal{R}_1,\ldots,\mathcal{R}_n);Z}$.

Proposition 4. *If a theory T defines orderings $\mathcal{R}_1, \ldots, \mathcal{R}_n$ and the functions in the tuple $\vec{F}$ belong to its language then a model $\mathcal{M}$ of T is a also a model of*

$$\begin{aligned}
&\{\vec{F}, Z : \\
&\qquad \forall x, y.\ \mathcal{R}_i(y, F_i(x)) \supset Ab(x, y, i)\ \textit{(for each } F_i \textit{ in } \vec{F} \textit{ and each } \mathcal{R}_i\textit{)} \\
&\qquad T \\
&\}
\end{aligned} \tag{16}$$

if and only if $\mathcal{M}$ is minimal relative to $\leqslant^{(\vec{F},\mathcal{R}_1,\ldots,\mathcal{R}_n);Z}$.

5.2. Prioritized value minimization

Let us again consider a theory that has both the formalization of narratives and minimum fare. Now suppose in our minimization of *minfare*(*Dallas*, *Mumbai*) and *Sit_map* we have a higher priority for *minfare*. (We are considering the travel of a graduate student John who does not mind taking extra trouble in order to get the cheapest Dallas–Mumbai ticket.) In other words we would prefer models which cost less between Dallas and Mumbai even if it entails occurrence of extra actions.

Suppose in our narrative John was in Dallas in one situation and in a later situation he was in Mumbai. We would of course like to conclude that John flew from Dallas to Mumbai.

But, at the same time we would like to minimize the cost of the ticket that John bought. Since the latter minimization has a higher priority, John may perform extra actions just so that he will pay less for the Dallas–Mumbai ticket. Examples of such extra actions—from one of the authors' graduate student days—involve buying a coupon from 'The Sharper Image' that gives 25% off the Dallas–Mumbai ticket, or using the triple mileage deal of PANAM to fly to Hawaii and get a free Dallas–Mumbai ticket by only paying for the ticket to Hawaii, which was the cheapest way to fly Dallas–Mumbai in Summer 1990. In both cases, in order to minimize the cost of the ticket John performs extra actions.

Introducing prioritized value minimization that will allow us to capture the above example, and minimize multiple functions, but in a prioritized manner. We follow Lifschitz's notation [7] and consider a tuple $\vec{F}$ of function symbols—which are to be minimized—to be partitioned into disjoint levels [12] (partitions) $\vec{F}^1, \ldots, \vec{F}^n$ of function symbols that have the same priority. Accordingly, we define an ordering over sets of function symbols: $\vec{F} \preceq_{\mathcal{R}_1,\ldots,\mathcal{R}_n} \vec{G}$ stands for

$$\bigwedge_{i=2}^{n}\Big(\bigwedge_{j=1}^{i-1}(\vec{F}^j = \vec{G}^j) \supset (\vec{F}^i \leqslant_{\mathcal{R}_i} \vec{G}^i)\Big)$$

where $\vec{G}$ is a tuple of functions of the same length as $\vec{F}$, with the same arities and with a partition $\vec{G}^1, \ldots, \vec{G}^n$ corresponding to that of $\vec{F}$. Value minimization of a collection of functions $\vec{F}$ with priorities $\vec{F}^1 > \vec{F}^2 > \cdots > \vec{F}^n$ in theory $T_{\mathcal{R}_1,\ldots,\mathcal{R}_n}$ with Z varied is expressed as follows:

$$T_{\mathcal{R}_1,\ldots,\mathcal{R}_n}(\vec{F}, Z) \wedge \neg\exists \vec{f}, z\big[T_{\mathcal{R}_1,\ldots,\mathcal{R}_n}(\vec{f}, z) \wedge \vec{f} \prec_{\mathcal{R}_1,\ldots,\mathcal{R}_n} \vec{F}\big]. \tag{17}$$

Definition 4. Let T be a theory with functions $\vec{F}^1, \ldots, \vec{F}^n$ and orderings $\mathcal{R}_1, \ldots, \mathcal{R}_n$ on the elements of its universe. For two models $\mathcal{M}_1$ and $\mathcal{M}_2$ of T, we say

$$\mathcal{M}_1 \preceq^{(\vec{F}^1 > \cdots > \vec{F}^n; \mathcal{R}_1,\ldots,\mathcal{R}_n):Z} \mathcal{M}_2$$

if

- $|\mathcal{M}_1| = |\mathcal{M}_2|$,
- $\mathcal{M}_1[[\sigma]] = \mathcal{M}_2[[\sigma]]$ for each constant σ which does not belong to $\vec{F}$ and Z,

[12] It is intended that $\vec{F}^1$ has the highest priority and $\vec{F}^n$ the lowest.

– for all x, $\mathcal{M}_1[[F']](x)\,\mathcal{R}_1\,\mathcal{M}_2[[F']](x)$ holds for each F' in $\vec{F}^1$ and ordering $\mathcal{R}_1$,
– for each $i = 2, \ldots, n$, if for each $j = 1, \ldots, i-1$, we have that

$$(\forall x)\mathcal{M}_1[[F]](x) = \mathcal{M}_2[[F]](x)$$

holds for each function F in $\vec{F}^j$, then

$$(\forall x)\mathcal{M}_1[[F']](x)\,\mathcal{R}_i\,\mathcal{M}_2[[F']](x)$$

holds for each F' in $\vec{F}^i$ and ordering $\mathcal{R}_i$.

Proposition 5. *An interpretation $\mathcal{M}$ is a model of (17) iff $\mathcal{M}$ is a model of $T_{\mathcal{R}_1,\ldots,\mathcal{R}_n}(\vec{F}, Z)$ and it is minimal relative to $\preceq^{(\vec{F}^1>\cdots>\vec{F}^n;\mathcal{R}_1,\ldots,\mathcal{R}_n);Z}$.*

Proof. Similar to proof of Proposition 1. □

6. Value minimization of functions encoded as predicates

Although we advocate functions to be directly represented through function constants, sometimes *there may be restrictions in the language that will necessitate encoding functions using predicates.* For instance, function constants are not allowed in basic DATALOG. Also, by not having function constants in the language, we avoid having an infinite universe when it is not needed.

We can reformulate our earlier examples by encoding the function *Sit_map* by a binary sorted predicate *maps*, where *maps*(X, Y) means that the situation constant X is mapped to the action sequence Y. One consequence of this reformulation is that we have to add the following to our theory.

$$maps(X, Y) \wedge maps(X, Z) \supset Y = Z.$$

In this section we discuss value minimization of functions that are encoded as predicates. We say a predicate $P(\vec{x}, \vec{y})$ *encodes* a function in a theory if the theory entails:

for all $\vec{x}, \vec{y}_1, \vec{y}_2$, $P(\vec{x}, \vec{y}_1)$ and $P(\vec{x}, \vec{y}_2)$ implies $\vec{y}_1 = \vec{y}_2$.

Intuitively, we can think of $\vec{x}$ as the *input* and $\vec{y}$ as the *output*, and also speak of *input-arity* and *output-arity* when referring to the length of $\vec{x}$ and $\vec{y}$, respectively. Then, we can consider minimizing the value of the output of P. After this minimization, P will encode a function which maps input tuples into minimal output tuples, with respect to an ordering on the elements of the universe and a theory.

We first consider the simple case when we want to minimize the value of the output as a single term, with respect to an ordering $\mathcal{R}$ on output tuples. We need the following notation: let P and P' be predicates (encoding functions) with the same input and output arities, then:

$$P \leqslant_{\mathcal{R}} P' \quad \text{stands for} \quad (\forall \vec{x}, \vec{y}, \vec{y}')\,.\, P(\vec{x}, \vec{y}) \wedge P'(\vec{x}, \vec{y}') \supset \mathcal{R}(\vec{y}, \vec{y}').$$

Value minimization is achieved in a very similar way as it is with functions: let $T_{\mathcal{R}}(p, z)$ be a theory where a predicate symbol p (that encodes a function) and function/predicate symbols in tuple z appear as free variables, and which includes a definition of predicate $\mathcal{R}$ representing the ordering. Value minimization of the output of P with Z varied is defined as follows:

$$T_{\mathcal{R}}(P, Z) \wedge \neg\exists p, z\big[T_{\mathcal{R}}(p, z) \wedge p <_{\mathcal{R}} P\big]. \tag{18}$$

The corresponding ordering on models is defined as follows:

Definition 5. Let T be a theory, P be a predicate (encoding a function) and Z be a tuple of predicate/function constants in the language of T. Let $\mathcal{R}$ be a partial order defined over tuples of length equal to the output arity of P, formed with elements of the universe of T. For two interpretations $\mathcal{M}$ and $\mathcal{M}'$ of T, we say $\mathcal{M} \leqslant^{(P;\mathcal{R});Z} \mathcal{M}'$ if

(i) $|\mathcal{M}| = |\mathcal{M}'|$;
(ii) $\mathcal{M}[[\sigma]] = \mathcal{M}'[[\sigma]]$ for each constant σ s.t. $\sigma \neq P$, $\sigma \notin Z$;
(iii) for all $\vec{x}, \vec{y}, \vec{y}'$ if $\langle\vec{x}, \vec{y}\rangle \in \mathcal{M}[[P]] \wedge \langle\vec{x}, \vec{y}'\rangle \in \mathcal{M}'[[P]]$ then $\vec{y}\ \mathcal{R}\ \vec{y}'$.

The following is proved by adapting the proof of Proposition 1.

Proposition 6. *An interpretation $\mathcal{M}$ is a model of* (18) *iff $\mathcal{M}$ is a model of $T_{\mathcal{R}}(P, Z)$ and it is minimal relative to $\leqslant^{(P;\mathcal{R});Z}$.*

Consider $T_{\mathcal{R}}$, P and Z as above, the NAT characterization of value minimization of the output of P is given below.

Proposition 7. *An interpretation $\mathcal{M}$ is a model of*

$$
\begin{array}{l}
\{P, Z: \\
\qquad \big(\forall\vec{x}, \vec{y}, \vec{y}'\big).\, P(\vec{x}, \vec{y}) \wedge \mathcal{R}(\vec{y}', \vec{y}) \supset Ab(\vec{x}, \vec{y}') \\
\qquad T_{\mathcal{R}} \\
\}
\end{array}
$$

if and only if $\mathcal{M}$ is a model of T minimal relative to $\leqslant^{(P;\mathcal{R});Z}$.

Proof. Similar to that of Proposition 1. □

Now, a predicate $P(\vec{x}, \vec{y})$ can be considered as encoding multiple functions, each mapping $\vec{x}$ onto a y in $\vec{y}$. (This happens when a relation in a database is associated with several functional dependencies.) When this is the case, we may want to achieve some form of parallel or prioritized value minimization on these implicit functions. Let us introduce the necessary notation to make this precise. Let P and P' be predicates with output arity n and let $\mathcal{R}_1, \ldots, \mathcal{R}_n$ be orderings on the elements of the universe, then:

$P \leqslant_{\mathcal{R}_1,\ldots,\mathcal{R}_n} P'$ stands for

$$
\begin{array}{l}
\big(\forall\vec{x}, y_1, \ldots, y_n, y_1', \ldots, y_n'\big). \\
P\big(\vec{x}, y_1, \ldots, y_n\big) \wedge P'\big(\vec{x}, y_1', \ldots, y_n'\big) \supset \big[\mathcal{R}_1\big(y_1, y_1'\big) \wedge \cdots \wedge \mathcal{R}_n\big(y_n, y_n'\big)\big].
\end{array}
$$

This notation is useful for expressing parallel value minimization on the output elements of a predicate P:

$$T_{\mathcal{R}_1,\ldots,\mathcal{R}_n}(P, Z) \wedge \neg\exists p, z\big[T_{\mathcal{R}_1,\ldots,\mathcal{R}_n}(p, z) \wedge p <_{\mathcal{R}_1,\ldots,\mathcal{R}_n} P\big]. \tag{19}$$

The minimization then is on each element of the output. That is, given an input tuple $\vec{x}$, an output tuple $\vec{y}$ is preferable to a tuple $\vec{y}'$ if each y_i in $\vec{y}$ is preferable to y_i' in $\vec{y}'$. As before, we define an ordering on models so that by choosing a minimal model of a theory in terms of this ordering, we obtain the desired output minimization.

Definition 6. Let T be a theory, P be a predicate with output arity n and Z be a tuple of predicate/function constants in the language of T. Let $\mathcal{R}_1, \ldots, \mathcal{R}_n$ be partial orderings on elements of the universe of T. For two interpretations $\mathcal{M}$ and $\mathcal{M}'$ of T, we say $\mathcal{M} \leqslant^{(P;\mathcal{R}_1,\ldots,\mathcal{R}_n);Z} \mathcal{M}'$ if

(i) $|\mathcal{M}| = |\mathcal{M}'|$;

(ii) $\mathcal{M}[[\sigma]] = \mathcal{M}'[[\sigma]]$ for each constant σ s.t. $\sigma \neq P$, $\sigma \notin Z$;

(iii) for all $\vec{x}, y_1, \ldots, y_n, y_1', \ldots, y_n'$ if

$$\langle\vec{x}, y_1, \ldots, y_n\rangle \in \mathcal{M}[[P]] \wedge \langle\vec{x}, y_1', \ldots, y_n'\rangle \in \mathcal{M}'[[P]]$$

then $y_i \, \mathcal{R}_i \, y_i'$ for each $i = 1, \ldots, n$.

The corresponding NAT characterization is given below.

Proposition 8. *An interpretation $\mathcal{M}$ is a model of*

$$\begin{aligned}
&\{P, Z: \\
&\qquad (\forall\vec{x}, y_1, \ldots, y_n, y_1', \ldots, y_n'). \\
&\qquad\quad P(\vec{x}, y_1, \ldots, y_n) \wedge \mathcal{R}_1(y_1', y_1) \wedge \cdots \wedge \mathcal{R}_n(y_n', y_n) \supset Ab(\vec{x}, y_1', \ldots, y_n') \\
&\qquad T_{\mathcal{R}_1,\ldots,\mathcal{R}_n} \\
&\}
\end{aligned}$$

if and only if $\mathcal{M}$ is a model of T minimal relative to $\leqslant^{(P;\mathcal{R}_1,\ldots,\mathcal{R}_n);Z}$.

Proof. Again it follows by adapting the proof of Proposition 1. $\square$

The natural next generalization is to allow priorities among the output elements. To keep notation simple, let us assume that output terms appear in an order corresponding to the priorities with which they are to be minimized, i.e., the leftmost output term is that with the highest priority. Let P and P' be predicates with the same input and output arities; given the orderings $\mathcal{R}_1, \ldots, \mathcal{R}_n$ we introduce this notation:

$P \preceq_{\mathcal{R}_1,\ldots,\mathcal{R}_n} P'$ stands for

$$\begin{aligned}
&(\forall\vec{x}, y_1, \ldots, y_n, y_1', \ldots, y_n'). \\
&\quad P(\vec{x}, y_1, \ldots, y_n) \wedge P'(\vec{x}, y_1', \ldots, y_n') \supset \bigwedge_{i=1}^{n}\left(\bigwedge_{j=1}^{i-1}(y_j = y_j') \supset \mathcal{R}_i(y_i, y_i')\right).
\end{aligned}$$

Finally, the ordering on models can be defined:

Definition 7. Let T be a theory, P be a predicate with output arity n and Z be a tuple of predicate/function constants in the language of T. Let $\mathcal{R}_1, \ldots, \mathcal{R}_n$ be partial orderings on elements of the universe of T. For two interpretations $\mathcal{M}$ and $\mathcal{M}'$ of T, we say $\mathcal{M} \preceq^{(P;\mathcal{R}_1,\ldots,\mathcal{R}_n);Z} \mathcal{M}'$ if

(i) $|\mathcal{M}| = |\mathcal{M}'|$;

(ii) $\mathcal{M}[[\sigma]] = \mathcal{M}'[[\sigma]]$ for each constant σ s.t. $\sigma \neq P$, $\sigma \notin Z$;

(iii) for all $\vec{x}, y_1, \ldots, y_n, y'_1, \ldots, y'_n$ if

$$\langle \vec{x}, y_1, \ldots, y_n \rangle \in \mathcal{M}[[P]] \wedge \langle \vec{x}, y'_1, \ldots, y'_n \rangle \in \mathcal{M}'[[P]]$$

then $y_1 \; \mathcal{R}_1 \; y'_1$ and for each $i = 2, \ldots, n$, if for $j = 1, \ldots, i-1$, $y_j = y'_j$ then $y_i \; \mathcal{R}_i \; y'_i$.

7. Conclusions

This paper describes a technique for minimizing the value of a function with respect to a background theory and an ordering on the universe induced by the theory itself. As far as we know, this is the first time circumscription is introduced for minimizing a function, although varying a function during minimization of a predicate has been extensively used in Situation Calculus.

We consider the NAT characterization of value minimization as most important for the purposes of knowledge representation, in as much as it allows a value minimization block to be embedded in a larger theory, and allows to define blocks defining the partial orderings $\mathcal{R}$; this is done in a simpler and more structured way than with traditional circumscription.

Notice also that blocks implementing value-minimization of a function are just another kind of block, and therefore can be part of another NAT. It was only for the sake of definition that we described them as outermost with respect to theories. The direct and the model-theoretical definition of value minimization, which are important for understanding this new technique and for proofs, are given.

The usefulness of value minimization in knowledge representation is illustrated by way of a discussion of its application in theories of action from [1] where a formalization of narrative reasoning is developed in the framework of NATs. Finally, the concept of value minimization has been extended to multiple functions with or without prioritizes, even though only strict priorities were considered in this paper.

Acknowledgments

The authors wish to thank Michael Gelfond and Vladimir Lifschitz for their advice and encouragement to develop on Nested Abnormality Theories. The first two authors were supported by the NSF grants IRI-9211662 and IRI-9501577. The third author is grateful to Ann and John Gates for hosting him in El Paso.

References

[1] C. Baral, A. Gabaldon, A. Provetti, Formalizing narratives using nested circumscription, in: Proceedings 14th AAAI National Conference on Artificial Intelligence (AAAI-96), Portland, OR, 1996, pp. 652–657.

[2] C. Baral, M. Gelfond, A. Provetti, Representing actions: laws, observations and hypothesis, J. Logic Programming 31 (1–3) (1997) 201–244.

[3] A. Baker, Nonmonotonic reasoning in the framework of situation calculus, Artificial Intelligence 49 (1991) 5–23.

[4] T. Costello, Non-monotonicity and change, Ph.D. Thesis, Stanford University, 1997.

[5] B. Grosof, Generalizing priorization, in: Proceedings Second International Conference on Principles of Knowledge Representation and Reasoning (KR-91), Cambridge, MA, Morgan Kauffmann, San Mateo, CA, 1991, pp. 289–300.

[6] N. Kartha, V. Lifschitz, Actions with indirect effects (preliminary report), in: Proceedings Fourth International Conference on Principles of Knowledge Representation and Reasoning (KR-94), Bonn, Morgan Kauffmann, San Mateo, CA, 1994, pp. 341–350.

[7] V. Lifschitz, Circumscription, in: Handbook of Logic in Artificial Intelligence and Logic Programming, Oxford University Press, 1993, pp. 297–352.

[8] V. Lifschitz, Nested abnormality theories, Artificial Intelligence 74 (1995) 351–365.

[9] J. McCarthy, Epistemological problems of artificial intelligence, in: Proceedings IJCAI-77, Cambridge, MA, 1977, pp. 1038–1044.

[10] J. McCarthy, Applications of circumscription to formalizing common sense knowledge, Artificial Intelligence 28 (3) (1986) 89–116.

[11] R. Reiter, Equality and domain closure in first-order data bases, J. ACM 27 (2) (1980) 235–249.

[12] E. Sandewall, Filter preferential entailment for the logic of action in almost continuous worlds, in: Proceedings IJCAI-89, Detroit, MI, 1989, pp. 894–899.

[13] Y. Shoham, Nonmonotonic logics: meaning and utility, Proceedings IJCAI-87, Milan, Italy, 1987, pp. 388–393.