# Well Founded Semantics as Two-dimensional Here-and-There

**Pedro Cabalar**
AI Lab. - Dept. of Computer Science
University of Corunna
Corunna, Galicia, SPAIN
cabalar@@dc.fi.udc.es

**Abstract**

This work presents a new logical characterization of well founded semantics that provides, as interesting innovation, a fully semantic-oriented treatment of program connectives without relying on any syntactic restriction or transformation. Our work is inspired by a recent research line which has established a nice correspondence between stable models and a particular minimization for the monotonic logic of *here-and-there*. We identify a generalization of the latter (we have called *two-dimensional* here-and-there) that captures Przymusinki's 3-valued stable models and show that, as in the 2-valued case, the here-and-there version is more economical in the set of models involved in the minimization process.

## Introduction

The main goal of declarative logic programming is to provide a semantic interpretation of a logic program following, as close as possible, standard logical definitions. However, the two most successful declarative semantics for logic programs, *stable models* (Gelfond & Lifschitz 1988) and *well founded semantics* (WFS) (van Gelder, Ross, & Schlipf 1991), achieve this goal only up to a partial extent. Both cases have been traditionally defined in terms of a syntactic transformation (called the *modulo* or *reduct* of the program) and under the assumption of a particular shape for program rules.

Fortunately, this lack of declarativeness has been practically overcome for the stable models semantics, thanks to two different contributions. The work in (Lifschitz 1994) introduced the logic of *Minimal Belief and Negation as Failure* (MBNF), a simplification of (Lin & Shoham 1992), that allows encoding propositional Default Logic in a modal framework. Since stable models can be seen as a subset of Default Logic, MBNF indirectly provided a semantic specification for them that was, in fact, later used for studying properties of logic programs (see for instance (Lifschitz, Tang, & Turner 1999)).

A different research line (Pearce 1997) established an interesting identification of stable models as particular minimal models under Heyting's monotonic logic of *here-and-there* (a stronger version of intuitionistic logic). The result provides the advantages of MBNF, avoiding syntactic dependence, but fits better for stable models, since MBNF uses more complex structures for encoding Default Logic. Besides, as recently proved in (Lifschitz, Pearce, & Valverde 2000), it provides the nice property of capturing strong equivalence of logic programs, that is, two programs behave in the same way, with respect to addition of rules, iff they have the same set of here-and-there models.

Despite of all this work on "declarativeness" for stable models, there is, however, practically no similar approach for WFS. In fact, due to the availability of an efficient computation algorithm for WFS, great part of the literature describes the semantics as the execution of the algorithm itself, which clearly strays from the declarative point of view. The strong relation between stable models and WFS, as proved for instance in (Przymusinski 1994), seems to point out that the achievements of the here-and-there and MBNF encodings could be also obtained for the WFS case. In fact, in (Pearce 1998) a first characterization of WFS using here-and-there was proposed, although, as explained in the conclusions of that work, it was not completely successful, since it finally needed the use of the modulo operation.

In this paper we go one step further, presenting a semantic characterization of WFS that definitively avoids the use of syntactic transformations. To this aim, we use as monotonic basis what we have called the *two-dimensional* extension of here-and-there. The definition of well founded model is then reduced to a simple minimization process among models of this logic. We also show that, as happens with stable models, the here-and-there formulation for WFS is more economical, in the sense of considering less models for the minimization that yields the well founded models.

The methodology we have followed is to extrapolate the study done in (Przymusinski 1994), which shows that WFS can be obtained by generalizing stable models to the 3-valued case, and apply this result to here-and-there, which can also be somehow considered as a 3-valued approach. This informally explains the term of "two-dimensional": we apply 3-valuation to two different "directions."

The paper is organized as follows. In the next section, we present the basic concepts of 3-valued logic used in the paper. After that, we recall the original definition of 3-valued stable models and propose a slight variation which will be more useful for comparison purposes. Next, we present our main contribution, two-dimensional here-and-there, specifying the minimization process that captures WFS. The last two sections respectively include a brief comparison to "usual" here-and-there, and the conclusions and future work.

## Three valued logic: $\mathrm{L}_3$

Given a set of atoms $\Sigma$, called the *signature*, $\mathrm{L}_3$ syntax is formally defined as follows. If $\phi, \psi$ are $\mathrm{L}_3$ formulas and $p$ is any atom then:

$$p, \quad \neg\phi, \quad \phi \vee \psi, \quad \top, \quad \bot, \quad \mathbf{u}, \quad \boldsymbol{l}\phi$$

are also $\mathrm{L}_3$ formulas. The only additions with respect to propositional syntax are the constant $\mathbf{u}$ (read as "undefined") and the unary operator[1] $\boldsymbol{l}$. The formula $\boldsymbol{l}\phi$ is never undefined and points out whether $\phi$ is valuated to true or not. We will intuitively read it as "$\phi$ is believed." Let us include the derived operators $\wedge, \supset, \equiv$, defined in the usual way, and the new conditional:

$$\psi \leftarrow \phi \stackrel{\text{def}}{=} (\boldsymbol{l}\phi \supset \boldsymbol{l}\psi) \wedge (\boldsymbol{l}\neg\psi \supset \boldsymbol{l}\neg\phi)$$

A *literal* is any atom $p$ or its negation $\neg p$. Given a set of atoms $I$, by $\overline{I}$ we denote $\Sigma - I$. A *3-valued interpretation*, or *belief set*, is a pair of sets of atoms $(I^h, I^t)$ satisfying $I^h \subseteq I^t$. Intuitively, atoms in $I^h$ are "believed", atoms in $\overline{I^t}$ are "disbelieved" and all the rest are "undefined." Notice that in the literature, it is perhaps more usual to find the alternative shapes:

- a pair of sets of atoms $(I^+, I^-)$, where $I^+ \cap I^- = \emptyset$,
- a set $L$ of literals which is *consistent* (it contains no pair $p, \neg p$).

It is clear that we may indistinctly use any of the three representations, since they satisfy the correspondence:

$$\begin{array}{lll} I^+ = & I^h & = \{p : p \in L\} \\ I^- = & \overline{I^t} & = \{p : \neg p \in L\} \end{array}$$

However, our choice has not been casual, since $(I^h, I^t)$ is precisely the structure used by here-and-there, and so, will be better for comparison purposes. We say that $I^h$ and $I^t$ are respectively the "here" and "there" components of the belief set.

Two ordering relations among belief sets, $\preceq$ and $\preceq_F$, will be defined:

i) $(I^h, I^t) \preceq (J^h, J^t)$ iff $I^h \subseteq J^h$ and $I^t \subseteq J^t$,

ii) $(I^h, I^t) \preceq_F (J^h, J^t)$ iff $I^h \subseteq J^h$ and $J^t \subseteq I^t$.

In (Przymusinski 1994), these relations receive the names of *standard* and *Fitting's ordering* respectively. The $\preceq$ relation intuitively represents that one belief set is "less believing" than the other, i.e., it has less believed atoms and more disbelieved ones. The other relation, $\preceq_F$, measures the degree of knowledge, so that one belief set contains less knowledge, i.e., more undefined atoms than the other. Therefore, belief sets with shape $(I, I)$, which receive the name of *complete*, are $\preceq_F$-maximal, since they contain no undefined atom.

**Definition 1 ($\mathrm{L}_3$ valuation)**
The *valuation* of a formula $\phi$ with respect to the belief set $B = (I^h, I^t)$, denoted as $\phi^B$, is a value among $\{0, 1/2, 1\}$ (*false*, *unknown* or *true*, respectively), assigned following the next conditions:

1) $p^B = \begin{cases} 1 & \text{if } p \in I^h \\ 0 & \text{if } p \in \overline{I^t} \\ 1/2 & \text{otherwise} \end{cases}$

2) $(\neg\phi)^B = 1 - \phi^B$

3) $(\phi \vee \psi)^B = max(\phi^B, \psi^B)$

4) $\top^B = 1, \ \ \bot^B = 0$ and $\mathbf{u}^B = 1/2$

5) $(\boldsymbol{l}\phi)^B = \begin{cases} 1 & \text{if } \phi^B = 1 \\ 0 & \text{otherwise} \end{cases}$

$\Diamond$

Examining the valuation for the derived '$\leftarrow$' operator, it can be easily seen that coincides with the conditional presented in (Przymusinski 1994):

$$(\psi \leftarrow \phi)^B = \begin{cases} 1 & \text{if } \phi^B \leq \psi^B \\ 0 & \text{otherwise} \end{cases}$$

A belief set $B$ *satisfies* a formula $\phi$ when $\phi^B = 1$. We say that $B$ is a *model* of a theory $T$ when it satisfies all its formulas.

The following proposition asserts that $\mathrm{L}_3$ collapses into classical propositional logic when we only consider complete belief sets.

**Proposition 1** Let $T$ be a $\mathrm{L}_3$ theory and $T'$ a propositional theory, result of replacing in $T$ any expression $\boldsymbol{l}\phi$ by $\phi$. Then, for any set of atoms $I$, the complete belief set $(I, I)$ is a model of $T$ iff $I$ is a 2-valued model of $T'$. $\Diamond$

## Three-valued stable models

In order to deal with logic programs, $\mathrm{L}_3$ syntax needs to be extended with a new unary pseudo-operator *not* used for default negation. This pseudo-operator is not directly interpreted as the rest of connectives, but needs to be ruled out by a particular syntactic transformation, which is only applicable on theories with a restricted shape: logic programs.

A *logic program* is defined as a collection of *rules* which are in their turn defined as follows. If $a_1, \ldots a_k$,

[1] As explained in (Bull & Segerberg 1984), this operator, originally denoted as $\Diamond$, was already present in Lukasiewicz's formulation of 3-valued logic, and was intended for capturing the idea of "necessity."

$b_1, \ldots, b_n$, $c_1, \ldots, c_m$ denote atoms, a *general rule* is an expression:

$$a_1 \vee \cdots \vee a_k \leftarrow b_1 \wedge \cdots \wedge b_n \wedge not\, c_1 \wedge \cdots \wedge not\, c_m \quad (1)$$

with $k > 1, n \geq 0, m \geq 0$. A program is said to be *normal* [2] iff $k = 1$ for all rules and, moreover, is said to be *positive* iff $m = 0$ for all rules.

**Definition 2 (Modulo)**
Given a program $\Pi$ and a belief set $B$, we define $\Pi^B$ (read "$\Pi$ *modulo* $B$") as the program resulting from replacing in $\Pi$ each expression *not* $c_i$ by $\top$ (resp. $\bot$, resp. $\mathbf{u}$) if $c_i^B = 0$ (resp. 1, resp. 1/2). $\Diamond$

The program $\Pi^B$ does not contain default negation, and so, it is a $L_3$ theory. Therefore, it is possible to study the set of $L_3$ models of $\Pi^B$ and define a particular models selection among them.

**Definition 3 (3-valued stable model)**
A *3-valued stable model* $B$ of a program $\Pi$ is any $\preceq$-minimal model of $\Pi^B$. $\Diamond$

In (Przymusinski 1994) it is shown that a positive program has a unique 3-valued stable model. When the program $\Pi$ is normal, $\Pi^B$ is positive, and so, it is possible to define the function $\Gamma(B)$ that denotes the unique 3-valued stable model of $\Pi^B$. So, for normal programs, 3-valued stable models can be characterized as fixpoints of $\Gamma$: $B = \Gamma(B)$.

**Definition 4 (Well founded model)**
A *well founded model* $B$ of a program $\Pi$ is any $\preceq_F$-minimal 3-valued stable model. $\Diamond$

Again, for the case of normal programs, it has also been proved that there exists a $\preceq_F$-minimum 3-valued stable model, i.e., a unique well founded model.

## Avoiding the modulo operation: $NF_3$

As we have seen, the previous definitions do not provide an actual semantic treatment for the default negation. Informally speaking, the reason for this is that the minimization process actually handles two different interpretations: one which we may call the *a priori* assumption (used for ruling out the *not* 's), and the other for interpreting the resulting $L_3$ theory and making the minimization. Therefore, a possibility for providing a semantics for *not* is simultaneously handling two belief sets, rather than one[3]. Let us define a new framework, $NF_3$ (standing for *3-valued negation as failure*), for capturing this idea[4].

The syntax of $NF_3$ is the one from $L_3$ plus the unary operator '*not* '. Notice that we allow now full nesting of this operator with respect to any connective. A $NF_3$ *interpretation*, $M$, is defined as a pair of belief sets $(B^h, B^t)$. Intuitively, $B^t$ will be used for interpreting default negation, whereas $B^h$ will correspond to models of the modulo program. Again, the choice for the superindices is not casual: we say that $B^h$ and $B^t$ are respectively the "here" and "there" components of $M$. Notice that each belief set is in its turn decomposed into two sets of atoms also superindexed by $h$ and $t$. For clarity sake, we will reserve the letters $B$ and $I$ for respectively representing belief sets and sets of atoms, using the notation:

- $B^h = (I^{hh}, I^{ht})$
- $B^t = (I^{th}, I^{tt})$

**Definition 5 ($NF_3$ valuation)**
The *valuation* of a formula $\phi$ with respect to the belief pair $M = (B^h, B^t)$ and $x \in \{h, t\}$, denoted as $\phi^{(M,x)}$, follows the next conditions:

$$1)\ p^{(M,x)} = \begin{cases} 1 & \text{if } p \in I^{xh} \\ 0 & \text{if } p \in \overline{I^{xt}} \\ 1/2 & \text{otherwise} \end{cases}$$

$$2)\ (not\, \phi)^{(M,x)} = (\neg\phi)^{(M,t)}$$

and the rest of operators are directly obtained from $L_3$ valuation (after properly replacing $B$ by $(M, w)$). $\Diamond$

In $NF_3$, a *model* of a theory $T$ is defined as any interpretation $M$, such that $\phi^{(M,h)} = 1$ for all $\phi \in T$. As we can see, all the atoms are initially valuated in the $B^h$ belief set. When we reach a '*not* ' subformula, the selected component changes to $t$, and so, all the atoms in the scope of *not* are valuated in the $B^t$ belief set.

**Proposition 2** An interpretation $(B^h, B^t)$, with $B^h = (I^{hh}, I^{ht})$ and $B^t = (I^{th}, I^{tt})$, satisfies a general rule like (1) iff the following two conditions:

$$\forall b_i \in I^{hh} \text{ and } \forall c_i \notin I^{tt} \Rightarrow \exists a_i \in I^{hh} \qquad \text{(C1)}$$

$$\forall b_i \in I^{ht} \text{ and } \forall c_i \notin I^{th} \Rightarrow \exists a_i \in I^{ht} \qquad \text{(C2)}$$

are true[5].

**Proof**
Conditions C1 and C2 correspond to the unfolding of any $\psi \leftarrow \phi$ into $\boldsymbol{l}\phi \supset \boldsymbol{l}\psi$ and $\boldsymbol{l}\neg\psi \supset \boldsymbol{l}\neg\phi$, respectively. After examining the valuation of formulas, we directly obtain C1 from the first formula and, from the second one we obtain:

$$\forall a_i \notin I^{hh} \Rightarrow \exists b_i \notin I^{ht} \text{ or } \exists c_i \in I^{th}$$

which is simply C2, expressed in its contrapositive shape. $\Diamond$

We prove now that we have actually captured the effect of the modulo transformation:

**Lemma 1** The $L_3$ models $B^h$ of the program $\Pi^{B^t}$ correspond to the $NF_3$-models $(B^h, B^t)$ of $\Pi$ fixing $B^t$.
**Proof**
In $NF_3$, when we study models $B = (B^h, B^t)$ fixing $B^t$, the valuation of '*not* ' formulas is always known. Thus, we have:

[2] Notice that, in fact, the original definition of 3-valued stable models was only presented for normal programs.

[3] This is, for instance, the same idea applied in MBNF in order to encode the justification part of default rules.

[4] $NF_3$ was first introduced in (Cabalar 1999), although it was used there for encoding answer sets semantics.

[5] $\forall$, $\exists$ and $\Rightarrow$ are handled here as metalogical symbols with their usual meaning.

1. $(not\, c_i)^B = 0 = (\bot)^B$ if $c_i \in I^{th}$
2. $(not\, c_i)^B = 1 = (\top)^B$ if $c_i \notin I^{tt}$
3. $(not\, c_i)^B = 1/2 = (\mathbf{u})^B$ if $c_i \in I^{tt} - I^{th}$

which allow making the modulo replacements in $\Pi$ as a $NF_3$ theory. Then, the formulas in $\Pi^{B^t}$ are exclusively valuated in the $B^h$ component, following exactly the same valuation as in $L_3$. $\Diamond$

For commodity sake, we reconsider the $\preceq$ and $\preceq_F$ orderings for $NF_3$-interpretations and define:

i) $(B^h, B^t) \leq_h (C^h, C^t)$ iff $B^t = C^t$ and $B^h \preceq C^h$,

ii) $(B^h, B^t) \leq_t (C^h, C^t)$ iff $B^t \preceq_F C^t$.

that is, in $\leq_h$, we fix the $B^t$ part (the set used for the modulo) and get the $\preceq$-minimal belief set. A minimal model wrt $\leq_h$ (resp. $\leq_t$) is said to be *h-minimal* (resp. *t-minimal*).

Finally, the following result shows that 3-valued stable models and well founded models can be identified as a particular kind of $NF_3$ minimal models, using the $\leq_h$ and $\leq_t$ orderings.

**Theorem 1** A belief set $B$ is a 3-valued stable model of a program $\Pi$ iff $(B, B)$ is $h$-minimal among $NF_3$-models of $\Pi$. Moreover, $B$ is well founded model of $\Pi$ iff $(B, B)$ is $t$-minimal among total $h$-minimal models of $\Pi$.

**Proof**
It is straightforward from lemma 1 and the definitions of $\leq_h$ and $\leq_t$. $\Diamond$

From now on, when talking about 3-valued stable models and well founded models we will use their $NF_3$ characterization obtained above, rather than their original $L_3$ shape.

## Two-dimensional Here-and-There: $HT^2$

An $HT^2$ *formula* is recursively defined as follows. If $p$ is an atom and $\phi, \psi$ are $HT^2$ formulas, then the following expressions are also $HT^2$ formulas:

$$\bot, \quad \top, \quad p, \quad \neg\phi, \quad \phi \vee \psi, \quad not\, \phi, \quad \phi \rightarrow \psi$$

A *one-dimensional world* is any element of the set $\{h, t\}$. We define the accessibility relation $R$ so that $h\ R\ h$, $t\ R\ t$ and $h\ R\ t$. A *two-dimensional world* is a pair $xy$ where $x, y$ are one-dimensional worlds. Two accessibility relations among worlds, $R_1$ and $R_2$, will be defined as follows:

i) $xy\ R_1\ x'y'$ iff $x\ R\ x'$ and $y\ R\ y'$

ii) $xy\ R_2\ xy'$ iff $y \neq y'$

iii) $hy\ R_2\ ty'$ iff $y \neq y'$

Figure 1 graphically shows the three relations, $R$, $R_1$ and $R_2$. Notice that if we make the $y$ components to coincide, that is $hh = ht = h$ and $th = tt = t$, $R_1$ and $R_2$ collapse into $R$.

**Definition 6 ($HT^2$-Interpretation)**
An $HT^2$-*interpretation* is defined as the pair $M = (B^h, B^t)$ where $B^h = (I^{hh}, I^{ht})$ and $B^t = (I^{th}, I^{tt})$ are belief sets satisfying $B^h \preceq B^t$. $\Diamond$

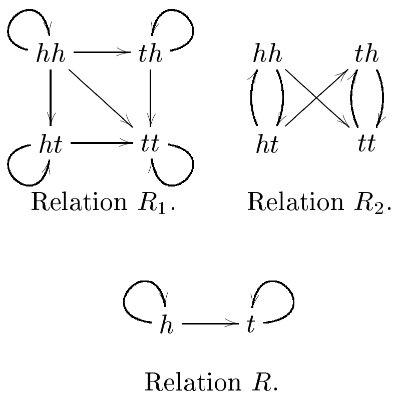


Figure 1: Accessibility relations among worlds.

Notice that the only difference with respect to a $NF_3$-interpretation is that we additionally require $B^h \preceq B^t$. Thus, we will sometimes use $HT^2$-interpretations in the role of $NF_3$ ones. The intuition behind the additional requirement, $B^h \preceq B^t$, is that we have further incremented the uncertainty of a belief set $B$, unfolding it into an *skeptical* set $B^h$ and a *credulous* set $B^t$. Thus, requiring $B^h \preceq B^t$ means that the skeptical set must be "less believing" than the credulous.

An interpretation of shape $(B, B)$ is said to be *total* (our skeptical and credulous viewings coincide). Note that there exists a certain analogy between total interpretation (associated to $x$-coordinate) and complete belief set (associated to $y$-coordinate). Anyway, it is possible to have a total interpretation $(B, B)$ for which $B$ is not complete, and vice versa, to have $B^h, B^t$ complete, but $(B^h, B^t)$ not total. In fact, as we will see later, this last case corresponds to the reduction to 2-valued stable models.

From the definitions of belief set and $\preceq$ it follows:

$I^w \subseteq I^{w'}$ iff $w\ R_1\ w'$

for any $w, w'$ two-dimensional worlds.

**Definition 7 (Satisfaction of a formula)**
We recursively define the *satisfaction* of a formula $\phi$ by an interpretation $M = (B^h, B^t)$ and a two-dimensional world $w$, written $(M, w) \models \phi$, in the following way:

1. $(M, w) \models p$ iff $p \in I^w$
2. $(M, w) \models \neg\phi$ iff $(M, w) \not\models \phi$
3. $(M, w) \models \phi \vee \psi$ iff $(M, w) \models \phi$ or $(M, w) \models \psi$
4. $(M, w) \models \phi \rightarrow \psi$ iff for all $w'$ such that $w\ R_1\ w'$, $(M, w') \models \phi \supset \psi$
5. $(M, w) \models not\ \phi$ iff for all $w'$ such that $w\ R_2\ w'$, $(M, w') \models \neg\phi$

$\Diamond$

Notice that although the semantic structures are almost identical to $NF_3$, the satisfaction relation relies on the usual two truth values – given $M$ and $w$, any formula $\phi$ is either satisfied or not satisfied. We say that an $\text{HT}^2$-interpretation $M$ is a *model* of a theory $T$ iff $(M, hh) \models \phi$ for all $\phi$ in $T$.

**Lemma 2** An $\text{HT}^2$-interpretation $(B^h, B^t)$, with $B^h = (I^{hh}, I^{ht})$ and $B^t = (I^{th}, I^{tt})$, is model of a general rule like (1) iff the following four conditions:

$$\forall b_i \in I^{hh} \text{ and } \forall c_i \notin I^{tt} \Rightarrow \exists a_i \in I^{hh} \qquad \text{(D1)}$$

$$\forall b_i \in I^{ht} \text{ and } \forall c_i \notin I^{th} \Rightarrow \exists a_i \in I^{ht} \qquad \text{(D2)}$$

$$\forall b_i \in I^{th} \text{ and } \forall c_i \notin I^{tt} \Rightarrow \exists a_i \in I^{th} \qquad \text{(D3)}$$

$$\forall b_i \in I^{tt} \text{ and } \forall c_i \notin I^{th} \Rightarrow \exists a_i \in I^{tt} \qquad \text{(D4)}$$

are true.

**Proof**
It is obtained by inspecting the satisfaction of '$\rightarrow$' and '*not*'. Conditions D1, D2, D3 and D4 respectively correspond to the propagation of the implication to worlds $hh$, $ht$, $th$, $tt$, all of them $R_1$-accessible from $hh$. As for each negation *not* $c_i$, it would strictly correspond to ($c_i \notin I^{tt}$ and $c_i \notin I^{ht}$) for D1 and ($c_i \notin I^{th}$ and $c_i \notin I^{hh}$) for D2. However, since $I^{ht} \subseteq I^{tt}$ and $I^{hh} \subseteq I^{th}$, the additional conditions on $I^{hh}$ and $I^{ht}$ have been removed from the antecedents, since they are redundant. $\diamondsuit$

We first directly notice that conditions D1 and D2 are respectively identical to conditions C1 and C2 we obtained for $\text{NF}_3$ satisfaction of (1). This directly means that:

**Proposition 3** If an $\text{HT}^2$-interpretation $M = (B^h, B^t)$ is a $\text{HT}^2$-model of a logic program $\Pi$ then $M$ is a $\text{NF}_3$-model of $\Pi$. $\diamondsuit$

Besides, it can be observed that, for any total interpretation $(B, B)$, conditions D1 and D2 respectively collapse into D3 and D4. This allows proving the following useful result:

**Lemma 3** Given a general logic program $\Pi$, if $(B^h, B^t)$ is an $\text{HT}^2$-model of $\Pi$, then $(B^t, B^t)$ is also $\text{HT}^2$-model of $\Pi$.

**Proof**
First, given any rule (1), from conditions D1-D4 we observe that the atoms $c_i$ are valuated in the there component of the $\text{HT}^2$-interpretation. Therefore, interpretations $(B^h, B^t)$ and $(B^t, B^t)$ will valuate them in the same way. Second, since $(B^h, B^t)$ is model of (1), it satisfies conditions D3 and D4, which are also valuated using exclusively $B^t$. Finally, since $(B^t, B^t)$ is total, D3 and D4 respectively collapse into D1 and D2, and so $(B^t, B^t)$ also satisfies D1 and D2, which means that is model of (1). $\diamondsuit$

Given a logic program $\Pi$ let us define $\text{SUBTOTAL}(\Pi)$ as the set of $\text{NF}_3$-models $(B^h, B^t)$ such that $B^h \preceq B^t$ and $(B^t, B^t)$ is $\text{NF}_3$-model of $\Pi$. As we can see, the first restriction ($B^h \preceq B^t$) directly implies that $\text{NF}_3$-models in $\text{SUBTOTAL}(\Pi)$ have the shape of $\text{HT}^2$-interpretations. Another important remark is that any $\text{NF}_3$-model $M \notin \text{SUBTOTAL}(\Pi)$ is irrelevant for computing the 3-valued stable models. The following theorem identifies the set of $\text{HT}^2$-models of $\Pi$ *exactly as* $\text{SUBTOTAL}(\Pi)$.

**Theorem 2** The pair of belief sets $(B^h, B^t)$ is $\text{HT}^2$-model of a logic program $\Pi$ iff $(B^h, B^t) \in \text{SUBTOTAL}(\Pi)$.

**Proof**
"$\Rightarrow$"
It is straightforward. From lemma 3 we have that $(B^t, B^t)$ is also $\text{HT}^2$-model. From proposition 3, any $\text{HT}^2$-model is $\text{NF}_3$-model. Then, both $(B^h, B^t)$ and $(B^t, B^t)$ are $\text{NF}_3$-models and, since any $\text{HT}^2$-interpretation satisfies $B^h \preceq B^t$, we conclude $(B^h, B^t) \in \text{SUBTOTAL}(\Pi)$.
"$\Leftarrow$"
First note that $(B^h, B^t)$ has the shape of an $\text{HT}^2$-interpretation, since $B^h \preceq B^t$. As $(B^h, B^t)$ is a $\text{NF}_3$-model, for any rule (1), it satisfies conditions C1 and C2, i.e., D1 and D2. However, as $(B^t, B^t)$ is also $\text{NF}_3$-model, it also satisfies D1 and D2 which, since we handle a total interpretation, collapse in their turn into D3 and D4. Finally, this implies that $(B^h, B^t)$ also satisfies D3 and D4, since it has the same there component than $(B^t, B^t)$. $\diamondsuit$

**Theorem 3** The total $h$-minimal $\text{HT}^2$-models of $\Pi$ are exactly its 3-valued stable models.

**Proof** Since $\text{HT}^2$-models of $\Pi$ exactly correspond to $\text{SUBTOTAL}(\Pi)$, and any $\text{NF}_3$-interpretation $M \notin \text{SUBTOTAL}(\Pi)$ is irrelevant for computing 3-valued stable models, we conclude that the sets of models involved in the minimization are identical in both cases. Then, requiring to be total and $h$-minimal is exactly the definition of the minimization done for 3-valued stable model. $\diamondsuit$

As a direct result from this theorem, we can apply exactly the same minimization as in $\text{NF}_3$ for obtaining well founded models in $\text{HT}^2$: they correspond to $t$-minimal models among total $h$-minimal models.

## Relation to "one-dimensional" Here-and-There

In this section we briefly comment what happens when we add to $\text{HT}^2$-interpretations the restriction of completeness of the belief sets. We say that any $\text{HT}^2$-interpretation $(B^h, B^t)$ is *complete* iff both belief sets are complete, that is, it has the shape $B^h = (I^h, I^h)$, $B^t = (I^t, I^t)$. This restriction can also be identified with reducing the set of worlds so that $hh = ht = h$ and $th = tt = t$ which, as explained before, makes both accessibility relations $R_1$ and $R_2$ collapse into the relation $R$ from "one-dimensional" here-and-there.

The condition $B^h \preceq B^t$ directly becomes the more usual $I^h \subseteq I^t$, present in here-and-there. With respect to the ordering relations, notice that $h$-minimality and $t$-minimality become exactly the ones defined in (Pearce 1997) under the same name. Therefore, the $h$-minimality definition allows identifying our total $h$-minimal models as a generalization of the *equilibrium models* presented there, which were proved to be the 2-valued stable models. As for the $t$-minimality, it was also shown in (Pearce 1997) that 2-valued stable models are $t$-minimal. This means that, our second minimization, required for obtaining the well founded models, does not yield any effect on complete stable models.

Notice that, since complete interpretations satisfy $R = R_1 = R_2$, an important property verified for this case is the equivalence:

$$(M, w) \models \phi \rightarrow \bot \quad \text{iff} \quad (M, w) \models \mathit{not}\, \phi \tag{2}$$

which for the two-dimensional case is not necessarily true. In fact, in usual here-and-there, this equivalence allows defining *not* as an operator derived from the implication $\rightarrow$. In $\mathrm{HT}^2$, however, it is necessary to separate the accessibility relations for both operators if we wish to capture WFS. For instance, if we redefine $R_2$ to be equal to $R_1$, a simple logic program like:

$$p \leftarrow \mathit{not}\, p \tag{3}$$

leads to two total models, $B_1^h = B_1^t = (\{p\}, \{p\})$ and $B_2^h = B_2^t = (\emptyset, \{p\})$, but none of them are $h$-minimal. Therefore, with $R_2$ made equal to $R_1$, this program would not have "well founded model." Using the $R_2$ relation proposed in this work, the models below $(B_2^t, B_2^t)$ are ruled out, and so, this model becomes the unique total $h$-minimal model, i.e., the unique well founded model.

## Conclusions and future work

We have presented a fully declarative semantics for WFS that relies on a *two-dimensional* extension of the logic of here-and-there. As a result, we are able, not only to capture well founded semantics for normal programs, but also for any theory with full nesting capability for all the program connectives. The definition of WFS has been reduced to a simple minimization process among models for two-dimensional here-and-there. Besides, the models involved in the minimization process are less than the pairs $(B_1, B_2)$, $B_1$ model of $\Pi^{B_2}$, involved in computing the well founded models, under its traditional definition.

There are, however, some open topics not covered yet. For instance, it remains to be done a comparison between the obtained extension of WFS and other proposed extensions based on the modulo transformation. Besides, we have not covered the case of extended logic programs, that is, programs with "classical" or "explicit" negation, although this extension is currently under study.

Another interesting question that is left open is whether the set of $\mathrm{HT}^2$-models characterize the property of strong equivalence, as happens in the 2-valued case, or not. It is clear that two programs with the same set of $\mathrm{HT}^2$-models will be strongly equivalent. However, the opposite may not necessarily hold and remains to be proved. Finally, one more open line to be studied is how slight changes in the accessibility relations $R_1$ and $R_2$ (respectively used for rule implication and default negation), affect the set of models, since this could lead to alternative semantics different from stable models and WFS.

**Acknowledgements** I would wish to thank to Vladimir Lifschitz for his comments and explanations on his recent joint work with Pearce and Valverde about strong equivalence. Thanks also to Ale Provetti for his review of an early version of this document, and to Ramon P. Otero and David Lorenzo for their useful discussions. This research is partially supported by the Government of Spain, grant PB97-0228, and by a research grant of the Government of Galicia for a stay of the author at the University of Texas at El Paso (april-june, 2000) with the KR group, to which I want to express my gratitude for their kind hospitality.

## References

Bull, R., and Segerberg, K. 1984. Basic modal logic. In Gabbay, D., and Guenthner, F., eds., *Handbook of Philosophical Logic*, volume 2. D. Reidel Publishing Company. 1–88.

Cabalar, P. 1999. Temporal answer sets. In *Proceedings of the Joint Conference on Declarative Programming (APPIA-GULP-PRODE'99).*

Gelfond, M., and Lifschitz, V. 1988. The stable model semantics for logic programming. In Kowalski, R. A., and Bowen, K. A., eds., *Logic Programming: Proc. of the Fifth International Conference and Symposium (Volume 2).* Cambridge, MA: MIT Press. 1070–1080.

Lifschitz, V.; Pearce, D.; and Valverde, A. 2000. Strongly equivalent logic programs. *ACM Transactions on Computational Logic.* (to appear).

Lifschitz, V.; Tang, L. R.; and Turner, H. 1999. Nested expressions in logic programs. *Annals of Mathematics and Artificial Intelligence* 25:369–389.

Lifschitz, V. 1994. Minimal belief and negation as failure. *Artificial Intelligence* 70:53–72.

Lin, F., and Shoham, Y. 1992. A logic of knowledge and justified assumptions. *Artificial Intelligence* 57:271–289.

Pearce, D. 1997. A new logical characterisation of stable models and answer sets. In *Non monotonic extensions of logic programming. Proc. NMELP'96. (LNAI 1216).* Springer-Verlag.

Pearce, D. 1998. Back and forth semantics for normal, disjunctive and extended logic programs. In *Proceedings of the Joint Conference on Declarative Programming (APPIA-GULP-PRODE'98).*

Przymusinski, T. 1994. Well-founded and stationary models of logic programs. *Annals of Mathematics and Artificial Intelligence* 12:141–187.

van Gelder, A.; Ross, K. A.; and Schlipf, J. S. 1991. The well-founded semantics for general logic programs. *Journal of the ACM* 38(3):620–650.