Analysing and Extending Well-Founded and Partial Stable Semantics using Partial Equilibrium Logic

Pedro Cabalar[1] Sergei Odintsov[2] David Pearce[3]
Agustín Valverde[4]

[1]University of Corunna (Spain)

[2]Sobolev Institute of Mathematics (Novosibirsk, Russia)

[3]Universidad Rey Juan Carlos (Spain)

[4]University of Málaga (Spain)

Presented by Axel Polleres[3]

ICLP 2006, Seattle

Outline

1. Introduction
   - Logical foundations of Logic Programming
   - Partial Equilibrium Logic
2. Contributions
   - Correspondence results
   - Strong equivalence
   - Nested logic programs
   - Other results in the paper
3. Conclusions

Outline

1. Introduction
   - Logical foundations of Logic Programming
   - Partial Equilibrium Logic
2. Contributions
   - Correspondence results
   - Strong equivalence
   - Nested logic programs
   - Other results in the paper
3. Conclusions

Fixing logical foundations of LP

- LP definitions rely on:
  syntax transformations (*reduct*) + fixpoint constructions

  Example: M is a stable model [Gelfond & Lifschitz 88] when
  "M is a classical minimal model of Π^M"

- A logical style definition:
  get minimal models inside some (monotonic) logic.

- Advantages:
  - Logically equivalent programs $\Rightarrow$ same minimal models.
  - Full logical interpretation of connectives.

Fixing logical foundations of LP

- LP definitions rely on:
  syntax transformations (*reduct*) + fixpoint constructions

  Example: M is a stable model [Gelfond & Lifschitz 88] when "M is a classical minimal model of Π^M"

- A logical style definition:
  get minimal models inside some (monotonic) logic.

- Advantages:
  - Logically equivalent programs ⇒ same minimal models.
  - Full logical interpretation of connectives.

Fixing logical foundations of LP

- LP definitions rely on:
  syntax transformations (*reduct*) + fixpoint constructions

  Example: M is a partial stable model [Przymusinski 94] when
  "M is a 3-valued minimal model of Π^M"

- A logical style definition:
  get minimal models inside some (monotonic) logic.

- Advantages:
  - Logically equivalent programs ⇒ same minimal models.
  - Full logical interpretation of connectives.

Fixing logical foundations of LP

- LP definitions rely on:
  syntax transformations (*reduct*) + fixpoint constructions

  Example: M is a partial stable model [Przymusinski 94] when "M is a 3-valued minimal model of Π^M"

- A logical style definition:
  get minimal models inside some (monotonic) logic.

- Advantages:
  - Logically equivalent programs ⇒ same minimal models.
  - Full logical interpretation of connectives.

Fixing logical foundations of LP

- LP definitions rely on:
  syntax transformations (*reduct*) + fixpoint constructions

  Example: M is a partial stable model [Przymusinski 94] when "M is a 3-valued minimal model of Π^M"

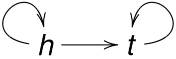
- A logical style definition:
  get minimal models inside some (monotonic) logic.

- Advantages:
  - Logically equivalent programs ⇒ same minimal models.
  - Full logical interpretation of connectives.

Stable models successfully identified

- (Monotonic) intermediate logic of *here-and-there* (*HT*)

  Classical $\subseteq$ *HT* $\subseteq$ Intuitionistic



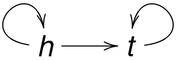
- Pearce's *Equilibrium Logic*: minimal *HT* models
  Intuition: *t* world is fixed (plays the role of "reduct"), *h* world is minimized

- Interesting results:
  - Equilibrium models = stable models [Pearce 97]
  - *HT* captures *strong equivalence* [Lifschitz, Pearce & Valverde 01] (we'll see later...)

Stable models successfully identified

- (Monotonic) intermediate logic of *here-and-there* (*HT*)

  Classical $\subseteq$ *HT* $\subseteq$ Intuitionistic



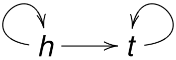
- Pearce's *Equilibrium Logic*: minimal *HT* models
  Intuition: *t* world is fixed (plays the role of "reduct"), *h* world is minimized

- Interesting results:
  - Equilibrium models = stable models [Pearce 97]
  - *HT* captures *strong equivalence* [Lifschitz, Pearce & Valverde 01] (we'll see later. . .)

Stable models successfully identified

- (Monotonic) intermediate logic of *here-and-there* (*HT*)

  Classical $\subseteq$ *HT* $\subseteq$ Intuitionistic



- Pearce's *Equilibrium Logic*: minimal *HT* models
  Intuition: *t* world is fixed (plays the role of "reduct"), *h* world is minimized

- Interesting results:
  - Equilibrium models = stable models [Pearce 97]
  - *HT* captures *strong equivalence* [Lifschitz, Pearce & Valverde 01] (we'll see later. . .)

Outline

1. Introduction
   - Logical foundations of Logic Programming
   - Partial Equilibrium Logic
2. Contributions
   - Correspondence results
   - Strong equivalence
   - Nested logic programs
   - Other results in the paper
3. Conclusions

Logical foundation of *WFS*: recently solved

[Cabalar,Odintsov & Pearce KR'06] *Partial* Equilibrium Logic

1. takes minimal models on monotonic logic HT^2
2. HT^2 classified inside [Došen 86] framework N combined with [Routley & Routley 72] (axioms in the paper).
3. Main idea: each world

   h
   t
   true ⊆ non-false

Logical foundation of *WFS*: recently solved

[Cabalar,Odintsov & Pearce KR'06] *Partial* Equilibrium Logic

1. takes minimal models on monotonic logic HT^2
2. HT^2 classified inside [Došen 86] framework N combined with [Routley & Routley 72] (axioms in the paper).
3. Main idea: each world

h
t
true $\subseteq$ non-false

Logical foundation of *WFS*: recently solved

[Cabalar,Odintsov & Pearce KR'06] *Partial* Equilibrium Logic

1. takes minimal models on monotonic logic HT^2
2. HT^2 classified inside [Došen 86] framework N combined with [Routley & Routley 72] (axioms in the paper).
3. Main idea: each world

h
t
true $\subseteq$ non-false

Logical foundation of *WFS*: recently solved

[Cabalar,Odintsov & Pearce KR'06] *Partial* Equilibrium Logic

1. takes minimal models on monotonic logic HT^2
2. HT^2 classified inside [Došen 86] framework N combined with [Routley & Routley 72] (axioms in the paper).
3. Main idea: each world

h

t

true $\subseteq$ non-false

Logical foundation of *WFS*: recently solved

[Cabalar,Odintsov & Pearce KR'06] *Partial* Equilibrium Logic

1. takes minimal models on monotonic logic HT^2
2. HT^2 classified inside [Došen 86] framework N combined with [Routley & Routley 72] (axioms in the paper).
3. Main idea: each world

h		h'	has now a primed version
t		t'	
true	$\subseteq$	non-false	

Logical foundation of *WFS*: recently solved

[Cabalar,Odintsov & Pearce KR'06] *Partial* Equilibrium Logic

1. takes minimal models on monotonic logic HT^2
2. HT^2 classified inside [Došen 86] framework N
   combined with [Routley & Routley 72] (axioms in the paper).
3. Main idea: each world

h		h'	has now a primed version
t		t'	with the intended meaning
true	$\subseteq$	non-false	

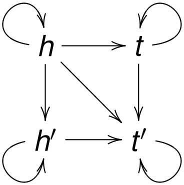
Logical foundation of *WFS*: recently solved

[Cabalar,Odintsov & Pearce KR'06] *Partial* Equilibrium Logic

1. takes minimal models on monotonic logic HT^2
2. HT^2 classified inside [Došen 86] framework N combined with [Routley & Routley 72] (axioms in the paper).
3. Main idea: each world

h		h'	has now a primed version
t		t'	with the intended meaning
true	$\subseteq$	non-false	

Semantics: HT^2 Frames



$\leq$ Accessibility relation like any intermediate logic
$(w \models p$ and $w \leq w')$ implies $w' \models p$

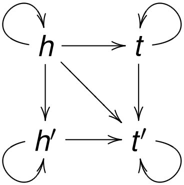
$\leq$ used for implication: $w \models \varphi \rightarrow \psi$ when
$\forall w' \geq w,\ \ w' \models \varphi$ implies $w' \models \psi$

But negation $\neg\phi$ is no longer defined as $\phi \rightarrow \bot$

$*$ *star* function (from Routley semantics)
satisfies: $v \leq w$ iff $w^* \leq v^*$

$w \models \neg\varphi$ when $w^* \not\models \varphi$

Semantics: HT^2 Frames



$\leq$ Accessibility relation like any intermediate logic
$(w \models p \text{ and } w \leq w')$ implies $w' \models p$

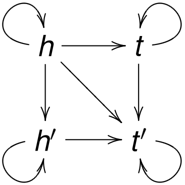
$\leq$ used for implication: $w \models \varphi \rightarrow \psi$ when
$\forall w' \geq w, \quad w' \models \varphi$ implies $w' \models \psi$

But negation $\neg\phi$ is no longer defined as $\phi \rightarrow \bot$

$*$ *star* function (from Routley semantics)
satisfies: $v \leq w$ iff $w^* \leq v^*$

$w \models \neg\varphi$ when $w^* \not\models \varphi$

Semantics: HT^2 Frames



$\leq$ Accessibility relation like any intermediate logic
$(w \models p$ and $w \leq w')$ implies $w' \models p$

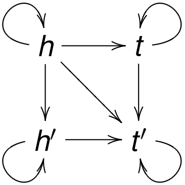
$\leq$ used for implication: $w \models \varphi \rightarrow \psi$ when
$\forall w' \geq w,\ \ w' \models \varphi$ implies $w' \models \psi$

But negation $\neg\phi$ is no longer defined as $\phi \rightarrow \bot$

$*$ *star* function (from Routley semantics)
satisfies: $v \leq w$ iff $w^* \leq v^*$

$w \models \neg\varphi$ when $w^* \not\models \varphi$

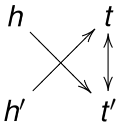
Semantics: HT^2 Frames



$\leq$ Accessibility relation like any intermediate logic
($w \models p$ and $w \leq w'$) implies $w' \models p$

$\leq$ used for implication: $w \models \varphi \rightarrow \psi$ when
$\forall w' \geq w, \;\; w' \models \varphi$ implies $w' \models \psi$

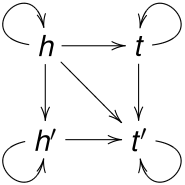
But negation $\neg\phi$ is no longer defined as $\phi \rightarrow \bot$



$*$ *star* function (from Routley semantics)
satisfies: $v \leq w$ iff $w^* \leq v^*$

$w \models \neg\varphi$ when $w^* \not\models \varphi$

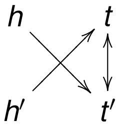
Semantics: HT^2 Frames



$\leq$ Accessibility relation like any intermediate logic
$(w \models p$ and $w \leq w')$ implies $w' \models p$

$\leq$ used for implication: $w \models \varphi \rightarrow \psi$ when
$\forall w' \geq w, \; w' \models \varphi$ implies $w' \models \psi$

But negation $\neg\phi$ is no longer defined as $\phi \rightarrow \bot$



$*$ *star* function (from Routley semantics)
satisfies: $v \leq w$ iff $w^* \leq v^*$

$w \models \neg\varphi$ when $w^* \not\models \varphi$

Partial equilibrium models

- Let H, H', T, T' denote sets of atoms verified at h, h', t, t'.
- A model can be seen as a pair $\langle \mathbf{H}, \mathbf{T} \rangle$ of 3-valued interp. where $\mathbf{H} = (H, H')$ and $\mathbf{T} = (T, T')$.
- Define an ordering among models, $\langle \mathbf{H}_1, \mathbf{T}_1 \rangle \trianglelefteq \langle \mathbf{H}_2, \mathbf{T}_2 \rangle$ if:
  - (i) $\mathbf{T}_1 = \mathbf{T}_2$ (this is fixed)
  - (ii) $\mathbf{H}_1$ less truth than $\mathbf{H}_2$ $(H_1 \subseteq H_2$ and $H'_1 \subseteq H'_2)$.
- $\langle \mathbf{H}, \mathbf{T} \rangle$ is said to be *total* if $\mathbf{H} = \mathbf{T}$.

Definition (Partial equilibrium model)

A model $\mathcal{M}$ of theory Π is a *partial equilibrium (PE) model* of Π if it is total and $\trianglelefteq$-minimal.

Partial equilibrium models

- Let H, H', T, T' denote sets of atoms verified at h, h', t, t'.
- A model can be seen as a pair $\langle \mathbf{H}, \mathbf{T} \rangle$ of 3-valued interp. where $\mathbf{H} = (H, H')$ and $\mathbf{T} = (T, T')$.
- Define an ordering among models, $\langle \mathbf{H}_1, \mathbf{T}_1 \rangle \trianglelefteq \langle \mathbf{H}_2, \mathbf{T}_2 \rangle$ if:
  - (i) $\mathbf{T}_1 = \mathbf{T}_2$ (this is fixed)
  - (ii) $\mathbf{H}_1$ less truth than $\mathbf{H}_2$ $(H_1 \subseteq H_2$ and $H'_1 \subseteq H'_2)$.
- $\langle \mathbf{H}, \mathbf{T} \rangle$ is said to be *total* if $\mathbf{H} = \mathbf{T}$.

Definition (Partial equilibrium model)

A model $\mathcal{M}$ of theory Π is a *partial equilibrium (PE) model* of Π if it is total and $\trianglelefteq$-minimal.

Partial equilibrium models

- Let H, H', T, T' denote sets of atoms verified at h, h', t, t'.
- A model can be seen as a pair $\langle \mathbf{H}, \mathbf{T} \rangle$ of 3-valued interp. where $\mathbf{H} = (H, H')$ and $\mathbf{T} = (T, T')$.
- Define an ordering among models, $\langle \mathbf{H}_1, \mathbf{T}_1 \rangle \trianglelefteq \langle \mathbf{H}_2, \mathbf{T}_2 \rangle$ if:
  (i) $\mathbf{T}_1 = \mathbf{T}_2$ (this is fixed)
  (ii) $\mathbf{H}_1$ less truth than $\mathbf{H}_2$ $(H_1 \subseteq H_2$ and $H'_1 \subseteq H'_2)$.
- $\langle \mathbf{H}, \mathbf{T} \rangle$ is said to be *total* if $\mathbf{H} = \mathbf{T}$.

Definition (Partial equilibrium model)

A model $\mathcal{M}$ of theory Π is a *partial equilibrium (PE) model* of Π if it is total and $\trianglelefteq$-minimal.

Partial equilibrium models

- Let H, H', T, T' denote sets of atoms verified at h, h', t, t'.
- A model can be seen as a pair $\langle \mathbf{H}, \mathbf{T} \rangle$ of 3-valued interp. where $\mathbf{H} = (H, H')$ and $\mathbf{T} = (T, T')$.
- Define an ordering among models, $\langle \mathbf{H}_1, \mathbf{T}_1 \rangle \trianglelefteq \langle \mathbf{H}_2, \mathbf{T}_2 \rangle$ if:
  - (i) $\mathbf{T}_1 = \mathbf{T}_2$ (this is fixed)
  - (ii) $\mathbf{H}_1$ less truth than $\mathbf{H}_2$ $(H_1 \subseteq H_2$ and $H'_1 \subseteq H'_2)$.
- $\langle \mathbf{H}, \mathbf{T} \rangle$ is said to be *total* if $\mathbf{H} = \mathbf{T}$.

Definition (Partial equilibrium model)

A model $\mathcal{M}$ of theory Π is a *partial equilibrium (PE) model* of Π if it is total and $\trianglelefteq$-minimal.

Partial equilibrium models

- Let H, H', T, T' denote sets of atoms verified at h, h', t, t'.
- A model can be seen as a pair $\langle \mathbf{H}, \mathbf{T} \rangle$ of 3-valued interp. where $\mathbf{H} = (H, H')$ and $\mathbf{T} = (T, T')$.
- Define an ordering among models, $\langle \mathbf{H}_1, \mathbf{T}_1 \rangle \trianglelefteq \langle \mathbf{H}_2, \mathbf{T}_2 \rangle$ if:
  (i) $\mathbf{T}_1 = \mathbf{T}_2$ (this is fixed)
  (ii) $\mathbf{H}_1$ less truth than $\mathbf{H}_2$ $\quad (H_1 \subseteq H_2$ and $H'_1 \subseteq H'_2)$.
- $\langle \mathbf{H}, \mathbf{T} \rangle$ is said to be *total* if $\mathbf{H} = \mathbf{T}$.

Definition (Partial equilibrium model)

A model $\mathcal{M}$ of theory Π is a *partial equilibrium (PE) model* of Π if it is total and $\trianglelefteq$-minimal.

Partial equilibrium models

- Let H, H', T, T' denote sets of atoms verified at h, h', t, t'.
- A model can be seen as a pair $\langle \mathbf{H}, \mathbf{T} \rangle$ of 3-valued interp. where $\mathbf{H} = (H, H')$ and $\mathbf{T} = (T, T')$.
- Define an ordering among models, $\langle \mathbf{H}_1, \mathbf{T}_1 \rangle \trianglelefteq \langle \mathbf{H}_2, \mathbf{T}_2 \rangle$ if:
  - (i) $\mathbf{T}_1 = \mathbf{T}_2$ (this is fixed)
  - (ii) $\mathbf{H}_1$ less truth than $\mathbf{H}_2$ $(H_1 \subseteq H_2$ and $H'_1 \subseteq H'_2)$.
- $\langle \mathbf{H}, \mathbf{T} \rangle$ is said to be *total* if $\mathbf{H} = \mathbf{T}$.

Definition (Partial equilibrium model)

A model $\mathcal{M}$ of theory Π is a *partial equilibrium (PE) model* of Π if it is total and $\trianglelefteq$-minimal.

Outline

Partial equilibrium models

Theorem (Corresp. to Partial Stable Models)

For a normal or disjunctive logic program Π, $\langle \mathbf{T}, \mathbf{T} \rangle$ is a partial equilibrium model of Π iff $\mathbf{T}$ is a partial stable model of Π.

Among PE models of a theory Π we define :

- *Well-Founded (WF) model*: minimal information
- *M-equilibrium model*: maximal information
- *L-equilibrium model*: minimal set of undefined atoms

Theorem

For a disjunctive logic program Π, they respectively correspond to well-founded and to M-stable and L-stable models from [Eiter, Leone & Saccà 98].

Partial equilibrium models

Theorem (Corresp. to Partial Stable Models)

For a normal or disjunctive logic program Π, $\langle \mathbf{T}, \mathbf{T} \rangle$ is a partial equilibrium model of Π iff $\mathbf{T}$ is a partial stable model of Π.

Among PE models of a theory Π we define :

- *Well-Founded (WF) model*: minimal information
- *M-equilibrium model*: maximal information
- *L-equilibrium model*: minimal set of undefined atoms

Theorem

For a disjunctive logic program Π, they respectively correspond to well-founded and to M-stable and L-stable models from [Eiter, Leone & Saccà 98].

Partial equilibrium models

Theorem (Corresp. to Partial Stable Models)

For a normal or disjunctive logic program Π, $\langle \mathbf{T}, \mathbf{T} \rangle$ is a partial equilibrium model of Π iff $\mathbf{T}$ is a partial stable model of Π.

Among PE models of a theory Π we define :

- *Well-Founded (WF) model*: minimal information
- *M-equilibrium model*: maximal information
- *L-equilibrium model*: minimal set of undefined atoms

Theorem

For a disjunctive logic program Π, they respectively correspond to well-founded and to M-stable and L-stable models from [Eiter, Leone & Saccà 98].

Outline

Partial Equilibrium Logic and Strong equivalence

Definition (X strong equivalence)

Two theories Π_1, Π_2 are said to be X *strongly equivalent* if for any set of formulas Γ, $\Pi_1 \cup \Gamma$ and $\Pi_2 \cup \Gamma$ have the same models of type X.

Theorem (from KR'06 paper)

Π_1, Π_2 *are PEL strongly equivalent iff they are equivalent in* HT^2.

New results: other model classes captured too

Theorem

Π_1, Π_2 *are WF (resp. M, L) strongly equivalent iff they are equivalent in* HT^2.

Partial Equilibrium Logic and Strong equivalence

Definition (*X* strong equivalence)

Two theories Π_1, Π_2 are said to be X *strongly equivalent* if for any set of formulas Γ, $\Pi_1 \cup \Gamma$ and $\Pi_2 \cup \Gamma$ have the same models of type X.

Theorem (from KR'06 paper)

Π_1, Π_2 *are PEL strongly equivalent iff they are equivalent in* HT^2.

New results: other model classes captured too

Theorem

Π_1, Π_2 *are WF (resp. M, L) strongly equivalent iff they are equivalent in* HT^2.

Partial Equilibrium Logic and Strong equivalence

Definition (X strong equivalence)

Two theories Π_1, Π_2 are said to be X *strongly equivalent* if for any set of formulas Γ, $\Pi_1 \cup \Gamma$ and $\Pi_2 \cup \Gamma$ have the same models of type X.

Theorem (from KR'06 paper)

Π_1, Π_2 *are PEL strongly equivalent iff they are equivalent in* HT^2.

New results: other model classes captured too

Theorem

Π_1, Π_2 *are WF (resp. M, L) strongly equivalent iff they are equivalent in* HT^2.

Partial Equilibrium Logic and Strong equivalence

Definition (X strong equivalence)

Two theories Π_1, Π_2 are said to be X *strongly equivalent* if for any set of formulas Γ, $\Pi_1 \cup \Gamma$ and $\Pi_2 \cup \Gamma$ have the same models of type X.

Theorem (from KR'06 paper)

Π_1, Π_2 *are PEL strongly equivalent iff they are equivalent in* HT^2*.*

New results: other model classes captured too

Theorem

Π_1, Π_2 *are WF (resp. M, L) strongly equivalent iff they are equivalent in* HT^2*.*

Outline

Nested logic programs (NLP)

- Nested expressions: nest $\wedge, \vee, \top, \bot, \neg$ in rule heads and bodies
- Quite common (for rule bodies) in Prolog.
  Example: `a :- \+ (b; c, \+ (d, \+ e)).`
  in logical notation $\neg(b \vee c \wedge \neg(d \wedge \neg e)) \rightarrow a$
- [Lifschitz,Tang,Turner99] (for stable models) NLP unfolded using 12 transformations, which include:
  - Side switching for negation
    $F \wedge \neg\neg G \rightarrow H$ becomes $F \rightarrow H \vee \neg G$
    $F \rightarrow G \vee \neg\neg H$ becomes $F \wedge \neg H \rightarrow G$
- What about PEL and WFS? Do they preserve these transformations? Yes, excepting side switching for negation.

Nested logic programs (NLP)

- Nested expressions: nest $\wedge, \vee, \top, \bot, \neg$ in rule heads and bodies
- Quite common (for rule bodies) in Prolog.
  Example: `a :- \+ (b; c, \+ (d, \+ e)).`
  in logical notation $\neg(b \vee c \wedge \neg(d \wedge \neg e)) \rightarrow a$
- [Lifschitz,Tang,Turner99] (for stable models) NLP unfolded using 12 transformations, which include:
  - Side switching for negation
    $F \wedge \neg\neg G \rightarrow H$ becomes $F \rightarrow H \vee \neg G$
    $F \rightarrow G \vee \neg\neg H$ becomes $F \wedge \neg H \rightarrow G$
- What about PEL and WFS? Do they preserve these transformations? Yes, excepting side switching for negation.

Nested logic programs (NLP)

- Nested expressions: nest $\wedge, \vee, \top, \bot, \neg$ in rule heads and bodies
- Quite common (for rule bodies) in Prolog.
  Example: `a :- \+ (b; c, \+ (d, \+ e)).`
  in logical notation $\neg(b \vee c \wedge \neg(d \wedge \neg e)) \rightarrow a$
- [Lifschitz,Tang,Turner99] (for stable models) NLP unfolded using 12 transformations, which include:
  - Side switching for negation
    $F \wedge \neg\neg G \rightarrow H$ becomes $F \rightarrow H \vee \neg G$
    $F \rightarrow G \vee \neg\neg H$ becomes $F \wedge \neg H \rightarrow G$
- What about PEL and WFS? Do they preserve these transformations? Yes, excepting side switching for negation.

Nested logic programs (NLP)

- Nested expressions: nest $\wedge, \vee, \top, \bot, \neg$ in rule heads and bodies
- Quite common (for rule bodies) in Prolog.
  Example: `a :- \+ (b; c, \+ (d, \+ e)).`
  in logical notation $\neg(b \vee c \wedge \neg(d \wedge \neg e)) \rightarrow a$
- [Lifschitz,Tang,Turner99] (for stable models) NLP unfolded using 12 transformations, which include:
  - Side switching for negation
    $F \wedge \neg\neg G \rightarrow H$ becomes $F \rightarrow H \vee \neg G$
    $F \rightarrow G \vee \neg\neg H$ becomes $F \wedge \neg H \rightarrow G$
- What about PEL and WFS? Do they preserve these transformations? Yes, excepting side switching for negation.

Nested logic programs (NLP)

- Nested expressions: nest $\wedge, \vee, \top, \bot, \neg$ in rule heads and bodies
- Quite common (for rule bodies) in Prolog.
  Example: `a :- \+ (b; c, \+ (d, \+ e)).`
  in logical notation $\neg(b \vee c \wedge \neg(d \wedge \neg e)) \rightarrow a$
- [Lifschitz,Tang,Turner99] (for stable models) NLP unfolded using 12 transformations, which include:
  - Side switching for negation
    $F \wedge \neg\neg G \rightarrow H$ becomes $F \rightarrow H \vee \neg G$
    $F \rightarrow G \vee \neg\neg H$ becomes $F \wedge \neg H \rightarrow G$
- What about PEL and WFS? Do they preserve these transformations? Yes, excepting side switching for negation.

Nested logic programs (NLP)

- Nested expressions: nest $\wedge, \vee, \top, \bot, \neg$ in rule heads and bodies
- Quite common (for rule bodies) in Prolog.
  Example: `a :- \+ (b; c, \+ (d, \+ e)).`
  in logical notation $\neg(b \vee c \wedge \neg(d \wedge \neg e)) \rightarrow a$
- [Lifschitz,Tang,Turner99] (for stable models) NLP unfolded using 12 transformations, which include:
  - Side switching for negation
    $F \wedge \neg\neg G \rightarrow H$ becomes $F \rightarrow H \vee \neg G$
    $F \rightarrow G \vee \neg\neg H$ becomes $F \wedge \neg H \rightarrow G$
- What about PEL and WFS? Do they preserve these transformations? Yes, excepting side switching for negation.

Nested LP for WFS

When restricting to nested expr. in bodies, we obtain rules like:

$$p_1 \wedge \cdots \wedge p_n \wedge \neg q_1 \wedge \cdots \wedge \neg q_m \wedge \neg\neg r_1 \wedge \cdots \wedge \neg\neg r_t \rightarrow s_1 \vee \cdots \vee s_k \quad (1)$$

That is, disjunctive LP with double negation in the body.

Theorem

Let Π be a disjunctive LP with double negation in the body. Let Π' be s.t. we replace each $\neg\neg c$ by $\neg\overline{c}$, plus a rule $\neg c \rightarrow \overline{c}$ per each new $\overline{c}$. Then Π and Π' are strongly equivalent modulo original alphabet.

Nested LP for WFS

When restricting to nested expr. in bodies, we obtain rules like:

$$p_1 \wedge \cdots \wedge p_n \wedge \neg q_1 \wedge \cdots \wedge \neg q_m \wedge \neg\neg r_1 \wedge \cdots \wedge \neg\neg r_t \rightarrow s_1 \vee \cdots \vee s_k \quad (1)$$

That is, disjunctive LP with double negation in the body.

Theorem

Let Π be a disjunctive LP with double negation in the body. Let Π' be s.t. we replace each $\neg\neg c$ by $\neg\overline{c}$, plus a rule $\neg c \rightarrow \overline{c}$ per each new $\overline{c}$. Then Π and Π' are strongly equivalent modulo original alphabet.

Outline

Properties of PEL inference

- Entailment: $\Pi \mid\!\sim \varphi$ if either
  - Π has PEL models and all of them satisfy φ or
  - Π has not PEL models and φ is an HT^2 tautology

Theorem

PEL inference fails cautious monotony, truth by cases, conditionalisation, rationality and weak rationality.
PEL inference satisfies reflexivity, cut, $\vee$ in the antecedent and modus tollens.

Properties of PEL inference

- Entailment: $\Pi \mid\!\sim \varphi$ if either
  - Π has PEL models and all of them satisfy φ or
  - Π has not PEL models and φ is an HT^2 tautology

Theorem

PEL inference fails cautious monotony, truth by cases, conditionalisation, rationality and weak rationality.

PEL inference satisfies reflexivity, cut, $\vee$ in the antecedent and modus tollens.

Properties of PEL inference

- Entailment: $\Pi \mid\!\sim \varphi$ if either
  - Π has PEL models and all of them satisfy φ or
  - Π has not PEL models and φ is an HT^2 tautology

Theorem

PEL inference fails cautious monotony, truth by cases, conditionalisation, rationality and weak rationality.
PEL inference satisfies reflexivity, cut, $\vee$ in the antecedent and modus tollens.

Complexity results

- SAT_{HT^2} is NP-complete; VAL_{HT^2} is coNP-complete
- Checking strong equivalence in PEL = VAL_{HT^2} = coNP-complete
- Existence of partial equilibrium models is Σ_2^P-hard
- The decision problem for equilibrium entailment is Π_2^P-hard

Transformation rules in disjunctive LP

- HT^2 allows analysing which transformations are strongly equivalent
- We have analysed 8 typical transformations for disjunctive LP (see paper): $TAUT$, RED^+, RED^-, $NONMIN$, $GPPE$, $WGPPE$, $CONTRA$, $S-IMP$.
- 3 of them are not sound in PEL ($GPPE$, $S-IMP$, $CONTRA$).

Theorem

D-WFS (resp. STATIC) and PEL are non-comparable (neither stronger or weaker).

Translating PEL into Equilibrium Logic

- [Janhunen et al, ACM TOCL to appear] transformation: obtains partial stable models by translating program (atoms duplicated) and computing stable models
- We generalise this result to translate PEL arbitrary theories into Equilibrium Logic (see paper)

  (The resulting translation of nested implications is not polynomial)

Summary

Partial Equilibrium Logic (PEL):
solid logical foundation for partial stable and well-founded semantics.

1. Strong equivalence under several model classes (WF, M-stable, L-stable) captured.
2. First interpretation of nested expressions for WFS
3. Complexity results similar to Equilibrium Logic
4. Translation of PEL into Equilibrium logic
5. Properties of PEL inference
6. Analysis of transformation rules for disjunctive WFS

Open topics

- Strong equivalence: when it fails, it is not always possible to generate a counterexample in the form of a program yet.
- Study XSB with nested expressions: correct wrt PEL?

Further reading

- P. Cabalar, S. Odintsov & D. Pearce. Logical Foundations of Well-Founded Semantics. In *Proceedings KR 06*.
- P. Cabalar, S. Odintsov & D. Pearce. Strong Negation in Well-Founded and Partial Stable Semantics for Logic Programs. In *Proceedings of IBERAMIA'06*, (LNCS, to appear).
  - Extensions of PEL with strong negation. Comparison to WFSX.
- P. Cabalar, S. Odintsov, D. Pearce & A. Valverde. On the logic and computation of Partial Equilibrium Models. In *Proceedings of JELIA'06*, (LNCS, to appear).
  - Tableaux proof system
  - Splitting theorem for PEL

Transformation rules in disjunctive LP

Why GPPE (unfolding) is not sound? Example:

$$\begin{array}{rcl} & & a \vee b \\ \neg a & \rightarrow & a \\ a \wedge b & \rightarrow & c \end{array}$$

We get 2 PEL models, depending on $a \vee b$:

- When a is true, b and c become false
- When b is true, a gets undefined, and c too (it depends on a)

After applying unfolding on atom b we get the program:

$$\begin{array}{rcl} & & a \vee b \\ \neg a & \rightarrow & a \\ a & \rightarrow & a \vee c \end{array} \qquad \text{(it's a tautology!)}$$

that leaves c false in all PEL models.

Transformation rules in disjunctive LP

Why GPPE (unfolding) is not sound? Example:

$$\begin{array}{rcl} & & a \vee b \\ \neg a & \rightarrow & a \\ a \wedge b & \rightarrow & c \end{array}$$

We get 2 PEL models, depending on $a \vee b$:

- When a is true, b and c become false
- When b is true, a gets undefined, and c too (it depends on a)

After applying unfolding on atom b we get the program:

$$\begin{array}{rcl} & & a \vee b \\ \neg a & \rightarrow & a \\ a & \rightarrow & a \vee c \end{array} \qquad \text{(it's a tautology!)}$$

that leaves c false in all PEL models.

Transformation rules in disjunctive LP

Why GPPE (unfolding) is not sound? Example:

$$\begin{array}{rcl} & & a \vee b \\ \neg a & \rightarrow & a \\ a \wedge b & \rightarrow & c \end{array}$$

We get 2 PEL models, depending on $a \vee b$:

- When a is true, b and c become false
- When b is true, a gets undefined, and c too (it depends on a)

After applying unfolding on atom b we get the program:

$$\begin{array}{rcl} & & a \vee b \\ \neg a & \rightarrow & a \\ a & \rightarrow & a \vee c \end{array} \qquad \text{(it's a tautology!)}$$

that leaves c false in all PEL models.

Transformation rules in disjunctive LP

Why GPPE (unfolding) is not sound? Example:

$$\begin{array}{rcl} & & a \vee b \\ \neg a & \rightarrow & a \\ a \wedge b & \rightarrow & c \end{array}$$

We get 2 PEL models, depending on $a \vee b$:

- When a is true, b and c become false
- When b is true, a gets undefined, and c too (it depends on a)

After applying unfolding on atom b we get the program:

$$\begin{array}{rcl} & & a \vee b \\ \neg a & \rightarrow & a \\ a & \rightarrow & a \vee c \end{array} \qquad \text{(it's a tautology!)}$$

that leaves c false in all PEL models.

Transformation rules in disjunctive LP

Why GPPE (unfolding) is not sound? Example:

$$\begin{array}{rcl} & & a \vee b \\ \neg a & \rightarrow & a \\ a \wedge b & \rightarrow & c \end{array}$$

We get 2 PEL models, depending on $a \vee b$:

- When a is true, b and c become false
- When b is true, a gets undefined, and c too (it depends on a)

After applying unfolding on atom b we get the program:

$$\begin{array}{rcl} & & a \vee b \\ \neg a & \rightarrow & a \\ a & \rightarrow & a \vee c \end{array} \qquad \text{(it's a tautology!)}$$

that leaves c false in all PEL models.