Logic Programming with Preferences using Infinitesimal Logic

Antonis Troumpoukis

National and Kapodistrian University of Athens
Department of Informatics and Telecommunications

PLS9, July 15, 2013

An example

Example

"I want to book a flight from Athens to Boston **and optionally** with Olympic Air"

An example

Example

"I want to book a flight from Athens to Boston **and optionally** with Olympic Air"

- 3 possible answers:

An example

Example

"I want to book a flight from Athens to Boston **and optionally** with Olympic Air"

- 3 possible answers:
  - "yes"

An example

Example

"I want to book a flight from Athens to Boston **and optionally** with Olympic Air"

- 3 possible answers:
  - "yes"
  - "no, no flight from Athens to Boston"

An example

Example

"I want to book a flight from Athens to Boston **and optionally** with Olympic Air"

- 3 possible answers:
  - "yes"
  - "no, no flight from Athens to Boston"
  - "no, no flight from Athens to Boston with Olympic Air"

An example

Example

"I want to book a flight from Athens to Boston **and optionally** with Olympic Air"

- 3 possible answers:
  - "yes"
  - "no, no flight from Athens to Boston"
  - "no, no flight from Athens to Boston with Olympic Air"
- the third answer is "less false" than the second

An example

Example

"I want to book a flight from Athens to Boston **and optionally** with Olympic Air"

- 3 possible answers:
  - "yes"
  - "no, no flight from Athens to Boston"
  - "no, no flight from Athens to Boston with Olympic Air"
- the third answer is "less false" than the second
- we introduce a new truth value: $\mathsf{F}_0 < \mathsf{F}_1 < \mathsf{T}_0$.

An example (cont.)

Example

"I want to have a stopover in Rome **or alternatively** a stopover in London."

An example (cont.)

Example

"I want to have a stopover in Rome **or alternatively** a stopover in London."

- 3 possible answers:

An example (cont.)

Example

"I want to have a stopover in Rome **or alternatively** a stopover in London."

- 3 possible answers:
  - "no"

An example (cont.)

Example

"I want to have a stopover in Rome **or alternatively** a stopover in London."

- 3 possible answers:
  - "no"
  - "yes, with a stopover in Rome"

An example (cont.)

Example

"I want to have a stopover in Rome **or alternatively** a stopover in London."

- 3 possible answers:
  - "no"
  - "yes, with a stopover in Rome"
  - "yes, with a stopover in London"

An example (cont.)

Example

"I want to have a stopover in Rome **or alternatively** a stopover in London."

- 3 possible answers:
  - "no"
  - "yes, with a stopover in Rome"
  - "yes, with a stopover in London"
- the third answer is "less true" than the second

An example (cont.)

Example

"I want to have a stopover in Rome **or alternatively** a stopover in London."

- 3 possible answers:
  - "no"
  - "yes, with a stopover in Rome"
  - "yes, with a stopover in London"
- the third answer is "less true" than the second
- we introduce a new truth value: $\mathsf{F}_0 < \mathsf{F}_1 < \mathsf{T}_1 < \mathsf{T}_0$.

Set of truth values

- By adding more involved preferences we will need...

Set of truth values

- By adding more involved preferences we will need...

An infinite set of truth values, ordered as follows:

$$\mathsf{F}_0 < \mathsf{F}_1 < \cdots < 0 < \cdots < \mathsf{T}_1 < \mathsf{T}_0$$

Set of truth values

- By adding more involved preferences we will need...

An infinite set of truth values, ordered as follows:

$$\mathsf{F}_0 < \mathsf{F}_1 < \cdots < 0 < \cdots < \mathsf{T}_1 < \mathsf{T}_0$$

- truth values “less true” than T and “less false” than F

Set of truth values

- By adding more involved preferences we will need...

An infinite set of truth values, ordered as follows:

$$\mathsf{F}_0 < \mathsf{F}_1 < \cdots < 0 < \cdots < \mathsf{T}_1 < \mathsf{T}_0$$

- truth values “less true” than T and “less false” than F
- first introduced in **[Rondogiannis-Wadge, ACM ToCL 2005]** (LP with negation)

Set of truth values

- By adding more involved preferences we will need...

An infinite set of truth values, ordered as follows:

$$\mathsf{F}_0 < \mathsf{F}_1 < \cdots < 0 < \cdots < \mathsf{T}_1 < \mathsf{T}_0$$

- truth values “less true” than T and “less false” than F
- first introduced in **[Rondogiannis-Wadge, ACM ToCL 2005]** (LP with negation)
- different levels of truth values correspond to different degrees of preferences

An example (cont.)

Example

$$\begin{aligned}
\texttt{desired_flight}(\texttt{F}) \quad &\leftarrow \quad \texttt{from_to}(\texttt{athens}, \texttt{boston}, \texttt{F}) \wedge \\
&\qquad\quad \texttt{has_stopover}(\texttt{F}) \wedge \\
&\qquad\quad \mu\ \texttt{carrier}(\texttt{F}, \texttt{olympic_air}). \\
\texttt{has_stopover}(\texttt{F}) \quad &\leftarrow \quad \texttt{stopover}(\texttt{F}, \texttt{rome}) \vee \\
&\qquad\quad \xi\ \texttt{stopover}(\texttt{F}, \texttt{london}).
\end{aligned}$$

An example (cont.)

Example

$$\begin{aligned}
\texttt{desired_flight(F)} &\leftarrow \texttt{from_to(athens, boston, F)} \wedge \\
&\quad\ \texttt{has_stopover(F)} \wedge \\
&\quad\ \mu\ \texttt{carrier(F, olympic_air)}. \\
\texttt{has_stopover(F)} &\leftarrow \texttt{stopover(F, rome)} \vee \\
&\quad\ \xi\ \texttt{stopover(F, london)}.
\end{aligned}$$

- $\leftarrow \texttt{desired\_flight(F)}$

An example (cont.)

Example

$$
\begin{aligned}
\texttt{desired\_flight(F)} \quad &\leftarrow \quad \texttt{from\_to(athens, boston, F)} \wedge \\
&\qquad\quad \texttt{has\_stopover(F)} \wedge \\
&\qquad\quad \mu\ \texttt{carrier(F, olympic\_air)}. \\
\texttt{has\_stopover(F)} \quad &\leftarrow \quad \texttt{stopover(F, rome)} \vee \\
&\qquad\quad \xi\ \texttt{stopover(F, london)}.
\end{aligned}
$$

- $\leftarrow$ `desired_flight(F)`
  - F_0: some primary requirements are not satisfied

An example (cont.)

Example

$$\texttt{desired\_flight(F)} \leftarrow \texttt{from\_to(athens, boston, F)} \wedge \texttt{has\_stopover(F)} \wedge \mu\ \texttt{carrier(F, olympic\_air)}.$$

$$\texttt{has\_stopover(F)} \leftarrow \texttt{stopover(F, rome)} \vee \xi\ \texttt{stopover(F, london)}.$$

- $\leftarrow \texttt{desired\_flight(F)}$
  - F_0: some primary requirements are not satisfied

An example (cont.)

Example

$$\begin{aligned}
\texttt{desired_flight(F)} &\leftarrow \texttt{from_to(athens, boston, F)} \wedge \\
&\quad\ \texttt{has_stopover(F)} \wedge \\
&\quad\ \mu\ \texttt{carrier(F, olympic_air)}. \\
\texttt{has_stopover(F)} &\leftarrow \texttt{stopover(F, rome)} \vee \\
&\quad\ \xi\ \texttt{stopover(F, london)}.
\end{aligned}$$

- $\leftarrow$ `desired_flight(F)`
  - F_0: some primary requirements are not satisfied
  - F_1: optional requirement is not satisfied

An example (cont.)

Example

$$\texttt{desired\_flight(F)} \leftarrow \texttt{from\_to(athens, boston, F)} \wedge \texttt{has\_stopover(F)} \wedge \mu\ \texttt{carrier(F, olympic\_air)}.$$

$$\texttt{has\_stopover(F)} \leftarrow \texttt{stopover(F, rome)} \vee \xi\ \texttt{stopover(F, london)}.$$

- $\leftarrow$ `desired_flight(F)`
  - F_0: some primary requirements are not satisfied
  - F_1: optional requirement is not satisfied
  - T_1: alternative requirement is satisfied

An example (cont.)

Example

$$\texttt{desired\_flight(F)} \leftarrow \texttt{from\_to(athens, boston, F)} \wedge \texttt{has\_stopover(F)} \wedge \mu\ \texttt{carrier(F, olympic\_air)}.$$

$$\texttt{has\_stopover(F)} \leftarrow \texttt{stopover(F, rome)} \vee \xi\ \texttt{stopover(F, london)}.$$

- $\leftarrow$ `desired_flight(F)`
  - F_0: some primary requirements are not satisfied
  - F_1: optional requirement is not satisfied
  - T_1: alternative requirement is satisfied
  - T_0: all primary requirements are satisfied

An example (cont.)

Example

$$
\begin{aligned}
\texttt{desired\_flight(F)} \quad &\leftarrow \quad \texttt{from\_to(athens, boston, F)} \wedge \\
&\qquad\quad \texttt{has\_stopover(F)} \wedge \\
&\qquad\quad \mu\ \texttt{carrier(F, olympic\_air)}. \\
\texttt{has\_stopover(F)} \quad &\leftarrow \quad \texttt{stopover(F, rome)} \vee \\
&\qquad\quad \xi\ \texttt{stopover(F, london)}.
\end{aligned}
$$

- $\leftarrow$ `desired_flight(F)`
  - F_0: some primary requirements are not satisfied
  - F_1: optional requirement is not satisfied
  - T_1: alternative requirement is satisfied
  - T_0: all primary requirements are satisfied
- μ, ξ first introduced in **[Agarwal, Master's thesis, UVic 2005]** (not in LP setting)

Syntax

Definition

A logic program with preferences is a finite set of rules of one of the two following forms:

- $p \leftarrow \mu^{i_1} q_1 \wedge \cdots \wedge \mu^{i_n} q_n$
- $p \leftarrow \xi^{i_1} q_1 \vee \cdots \vee \xi^{i_n} q_n$

where $p, q_1, \ldots, q_n$ are atomic types.

Syntax

Definition

A logic program with preferences is a finite set of rules of one of the two following forms:

- $p \leftarrow \mu^{i_1} q_1 \wedge \cdots \wedge \mu^{i_n} q_n$
- $p \leftarrow \xi^{i_1} q_1 \vee \cdots \vee \xi^{i_n} q_n$

where $p, q_1, \ldots, q_n$ are atomic types.

Example

$$\mathtt{f(X)} \leftarrow \mathtt{ft(a,b,X)} \wedge \mathtt{p(X)} \wedge \mu\, \mathtt{c(X,o)} \wedge \mu^2\, \mathtt{s(X,w)}.$$

$$\mathtt{p(X)} \leftarrow \mathtt{s(X,r)} \vee \xi\, \mathtt{s(X,l)} \vee \xi^2\, \mathtt{s(X,z)}.$$

Syntax

Definition

A logic program with preferences is a finite set of rules of one of the two following forms:

- $p \leftarrow \mu^{i_1} q_1 \wedge \cdots \wedge \mu^{i_n} q_n$
- $p \leftarrow \xi^{i_1} q_1 \vee \cdots \vee \xi^{i_n} q_n$

where $p, q_1, \ldots, q_n$ are atomic types.

Example

$$\mathtt{f(X)} \leftarrow \mathtt{ft(a,b,X)} \wedge \mathtt{p(X)} \wedge \mu\, \mathtt{c(X,o)} \wedge \mu^2\, \mathtt{s(X,w)}.$$

$$\mathtt{p(X)} \leftarrow \mathtt{s(X,r)} \vee \xi\, \mathtt{s(X,l)} \vee \xi^2\, \mathtt{s(X,z)}.$$

- superscript i of μ^i and ξ^i allows us to express various levels of preferences

Infinitesimal interpretations

Definition

An interpretation I of a program P is a function from B_P to the infinite set of truth values $\{F_0, F_1, \ldots, 0, \ldots, T_1, T_0\}$.

Infinitesimal interpretations

Definition

An interpretation I of a program P is a function from B_P to the infinite set of truth values $\{F_0, F_1, \ldots, 0, \ldots, T_1, T_0\}$.

Definition

Let I be an interpretation of P. We extend I as follows:

- $I(A \wedge B) = \min\{I(A), I(B)\}$
- $I(A \vee B) = \max\{I(A), I(B)\}$
- $I(\mu^k A) = \begin{cases} T_i, & \text{if } I(A) = T_i \\ F_{i+k}, & \text{if } I(A) = F_i \\ 0, & \text{if } I(A) = 0 \end{cases}$
- $I(\xi^k A) = \begin{cases} T_{i+k}, & \text{if } I(A) = T_i \\ F_i, & \text{if } I(A) = F_i \\ 0, & \text{if } I(A) = 0 \end{cases}$
- $I(T) = T_0$

Preference operators

Example

A	μA	ξA
F_i	F_{i+1}	F_i
T_i	T_i	T_{i+1}

Preference operators

Example

A	μA	ξA
F_i	F_{i+1}	F_i
T_i	T_i	T_{i+1}

$$F_0 < \cdots < F_i < F_{i+1} < \cdots < 0 < \cdots < T_{i+1} < T_i < \cdots < T_0$$

Preference operators

Example

A	μA	ξA
F_i	F_{i+1}	F_i
T_i	T_i	T_{i+1}

$$F_0 < \cdots < F_i < F_{i+1} < \cdots < 0 < \cdots < T_{i+1} < T_i < \cdots < T_0$$

- operator μ:

Preference operators

Example

A	μA	ξA
F_i	F_{i+1}	F_i
T_i	T_i	T_{i+1}

$$F_0 < \cdots < F_i < F_{i+1} < \cdots < 0 < \cdots < T_{i+1} < T_i < \cdots < T_0$$

- operator μ:
  - if A is true, then μA as true as A

Preference operators

Example

A	μA	ξA
F_i	F_{i+1}	F_i
T_i	T_i	T_{i+1}

$$F_0 < \cdots < F_i < F_{i+1} < \cdots < 0 < \cdots < T_{i+1} < T_i < \cdots < T_0$$

- operator μ:
  - if A is true, then μA as true as A
  - if A is false, then μA is less false than A

Preference operators

Example

A	μA	ξA
F_i	F_{i+1}	F_i
T_i	T_i	T_{i+1}

$$F_0 < \cdots < F_i < F_{i+1} < \cdots < 0 < \cdots < T_{i+1} < T_i < \cdots < T_0$$

- operator μ:
  - if A is true, then μA as true as A
  - if A is false, then μA is less false than A
  - the absence of an **optional requirement** does not annoy us as much

Preference operators

Example

A	μA	ξA
F_i	F_{i+1}	F_i
T_i	T_i	T_{i+1}

$$F_0 < \cdots < F_i < F_{i+1} < \cdots < 0 < \cdots < T_{i+1} < T_i < \cdots < T_0$$

- operator μ:
  - if A is true, then μA as true as A
  - if A is false, then μA is less false than A
  - the absence of an **optional requirement** does not annoy us as much
- operator ξ:

Preference operators

Example

A	μA	ξA
F_i	F_{i+1}	F_i
T_i	T_i	T_{i+1}

$$F_0 < \cdots < F_i < F_{i+1} < \cdots < 0 < \cdots < T_{i+1} < T_i < \cdots < T_0$$

- operator μ:
  - if A is true, then μA as true as A
  - if A is false, then μA is less false than A
  - the absence of an **optional requirement** does not annoy us as much
- operator ξ:
  - if A is true, then ξA is less true than A

Preference operators

Example

A	μA	ξA
F_i	F_{i+1}	F_i
T_i	T_i	T_{i+1}

$$F_0 < \cdots < F_i < F_{i+1} < \cdots < 0 < \cdots < T_{i+1} < T_i < \cdots < T_0$$

- operator μ:
  - if A is true, then μA as true as A
  - if A is false, then μA is less false than A
  - the absence of an **optional requirement** does not annoy us as much
- operator ξ:
  - if A is true, then ξA is less true than A
  - if A is false, then ξA is as false as A

Preference operators

Example

A	μA	ξA
F_i	F_{i+1}	F_i
T_i	T_i	T_{i+1}

$$F_0 < \cdots < F_i < F_{i+1} < \cdots < 0 < \cdots < T_{i+1} < T_i < \cdots < T_0$$

- operator μ:
  - if A is true, then μA as true as A
  - if A is false, then μA is less false than A
  - the absence of an **optional requirement** does not annoy us as much
- operator ξ:
  - if A is true, then ξA is less true than A
  - if A is false, then ξA is as false as A
  - the presence of an **alternative option** is not as beneficial to us

Infinitesimal models

Definition

Let I be an interpretation of a program P. Then, I satisfies the clause $\mathrm{p} \leftarrow \mathbf{B}$ if $\mathrm{I}(\mathrm{p}) \geq \mathrm{I}(\mathbf{B})$.

Infinitesimal models

Definition

Let I be an interpretation of a program P. Then, I satisfies the clause $\mathrm{p} \leftarrow \mathbf{B}$ if $\mathrm{I}(\mathrm{p}) \geq \mathrm{I}(\mathbf{B})$.

Definition

Let M be an interpretation of a program P. Then, M is a model of P if I satisfies all clauses of P.

Fixed-point semantics

Definition

Let I, J be interpretations of a program P. We write $I \leq J$ if for all $p \in B_P$ it holds $I(p) \leq J(p)$.

Fixed-point semantics

Definition

Let I, J be interpretations of a program P. We write $I \leq J$ if for all $p \in B_P$ it holds $I(p) \leq J(p)$.

Proposition

The set of all interpretations of a program P is a complete lattice under the relation $\leq$.

Fixed-point semantics

Definition

Let I, J be interpretations of a program P. We write $I \leq J$ if for all $p \in B_P$ it holds $I(p) \leq J(p)$.

Proposition

The set of all interpretations of a program P is a complete lattice under the relation $\leq$.

Definition

Let I be an interpretation of a program P. We define the immediate consequence operator T_P of P as follows:

$$T_P(I)(p) = \text{lub}\{I(\mathbf{B}) : (p \leftarrow \mathbf{B}) \in \text{ground}(P)\}$$

Fixed-point semantics (cont.)

Theorem

The T_P operator is $\leq$-monotonic and $\leq$-continuous.

Fixed-point semantics (cont.)

Theorem

The T_P operator is $\leq$-monotonic and $\leq$-continuous.

Theorem

Every logic program with preferences has a $\leq$-minimum model, which is the least fixed-point of T_P.

Fixed-point semantics (cont.)

Theorem

The T_P operator is $\leq$-monotonic and $\leq$-continuous.

Theorem

Every logic program with preferences has a $\leq$-minimum model, which is the least fixed-point of T_P.

$$\bot = \{(p, F_0) : p \in B_P\}$$

Fixed-point semantics (cont.)

Theorem

The T_P operator is $\leq$-monotonic and $\leq$-continuous.

Theorem

Every logic program with preferences has a $\leq$-minimum model, which is the least fixed-point of T_P.

$$\bot = \{(p, F_0) : p \in B_P\}$$

$$\bot,\ T_P(\bot),\ T_P(T_P(\bot)),\ T_P(T_P(T_P(\bot)), \ldots$$

Fixed-point semantics (cont.)

Example

Fixed-point semantics (cont.)

Example

$s \leftarrow \xi\ p.$

$s \leftarrow q \wedge \mu\ r.$

$p \leftarrow q.$

$q.$

Fixed-point semantics (cont.)

Example

$s \leftarrow \xi\ p.$

$s \leftarrow q \wedge \mu\ r.$

$p \leftarrow q.$

$q.$

$$\{(s, F_0), (p, F_0), (q, F_0), (r, F_0)\}$$

Fixed-point semantics (cont.)

Example

$s \leftarrow \xi\ p.$ $\qquad (\xi\ F_0 = F_0)$

$s \leftarrow q \wedge \mu\ r.$ $\qquad (F_0 \wedge \mu\ F_0 = F_0)$

$p \leftarrow q.$

$q.$

$$\{(s, F_0), (p, F_0), (q, F_0), (r, F_0)\}$$

$$\{(s, F_0),$$

Fixed-point semantics (cont.)

Example

$s \leftarrow \xi\ p.$ $\qquad (\xi\ F_0 = F_0)$

$s \leftarrow q \wedge \mu\ r.$ $\qquad (F_0 \wedge \mu\ F_0 = F_0)$

$p \leftarrow q.$ $\qquad (F_0)$

$q.$

$$\{(s, F_0), (p, F_0), (q, F_0), (r, F_0)\}$$

$$\{(s, F_0), (p, F_0),$$

Fixed-point semantics (cont.)

Example

$s \leftarrow \xi\ p.$	$(\xi\ F_0 = F_0)$
$s \leftarrow q \wedge \mu\ r.$	$(F_0 \wedge \mu\ F_0 = F_0)$
$p \leftarrow q.$	(F_0)
$q.$	(T_0)

$$\{(s, F_0), (p, F_0), (q, F_0), (r, F_0)\}$$

$$\{(s, F_0), (p, F_0), (q, T_0),$$

Fixed-point semantics (cont.)

Example

$s \leftarrow \xi\ p.$ $\qquad (\xi\ F_0 = F_0)$

$s \leftarrow q \wedge \mu\ r.$ $\qquad (F_0 \wedge \mu\ F_0 = F_0)$

$p \leftarrow q.$ $\qquad (F_0)$

$q.$ $\qquad (T_0)$

$$\{(s, F_0), (p, F_0), (q, F_0), (r, F_0)\}$$

$$\{(s, F_0), (p, F_0), (q, T_0), (r, F_0)\}$$

Fixed-point semantics (cont.)

Example

$s \leftarrow \xi\ p.$ $\quad (\xi\ F_0 = F_0)$

$s \leftarrow q \wedge \mu\ r.$ $\quad (T_0 \wedge \mu\ F_0 = F_1)$

$p \leftarrow q.$

$q.$

$$\{(s, F_0), (p, F_0), (q, F_0), (r, F_0)\}$$

$$\{(s, F_0), (p, F_0), (q, T_0), (r, F_0)\}$$

$$\{(s, F_1),$$

Fixed-point semantics (cont.)

Example

$s \leftarrow \xi\ p.$ $\quad (\xi\ F_0 = F_0)$

$s \leftarrow q \wedge \mu\ r.$ $\quad (T_0 \wedge \mu\ F_0 = F_1)$

$p \leftarrow q.$ $\quad (T_0)$

$q.$

$$\{(s, F_0), (p, F_0), (q, F_0), (r, F_0)\}$$

$$\{(s, F_0), (p, F_0), (q, T_0), (r, F_0)\}$$

$$\{(s, F_1), (p, T_0),$$

Fixed-point semantics (cont.)

Example

$s \leftarrow \xi\ p.$ $\quad (\xi\ F_0 = F_0)$

$s \leftarrow q \wedge \mu\ r.$ $\quad (T_0 \wedge \mu\ F_0 = F_1)$

$p \leftarrow q.$ $\quad (T_0)$

$q.$ $\quad (T_0)$

$$\{(s, F_0), (p, F_0), (q, F_0), (r, F_0)\}$$

$$\{(s, F_0), (p, F_0), (q, T_0), (r, F_0)\}$$

$$\{(s, F_1), (p, T_0), (q, T_0),$$

Fixed-point semantics (cont.)

Example

$$s \leftarrow \xi\ p. \qquad (\xi\ F_0 = F_0)$$

$$s \leftarrow q \wedge \mu\ r. \qquad (T_0 \wedge \mu\ F_0 = F_1)$$

$$p \leftarrow q. \qquad (T_0)$$

$$q. \qquad (T_0)$$

$$\{(s, F_0), (p, F_0), (q, F_0), (r, F_0)\}$$

$$\{(s, F_0), (p, F_0), (q, T_0), (r, F_0)\}$$

$$\{(s, F_1), (p, T_0), (q, T_0), (r, F_0)\}$$

Fixed-point semantics (cont.)

Example

$s \leftarrow \xi\ p.$ $\quad (\xi\ T_0 = T_1)$

$s \leftarrow q \wedge \mu\ r.$ $\quad (T_0 \wedge \mu\ F_0 = F_1)$

$p \leftarrow q.$

$q.$

$$\{(s, F_0), (p, F_0), (q, F_0), (r, F_0)\}$$

$$\{(s, F_0), (p, F_0), (q, T_0), (r, F_0)\}$$

$$\{(s, F_1), (p, T_0), (q, T_0), (r, F_0)\}$$

$$\{(s, T_1),$$

Fixed-point semantics (cont.)

Example

$s \leftarrow \xi\ p.$ $\quad (\xi\ T_0 = T_1)$

$s \leftarrow q \wedge \mu\ r.$ $\quad (T_0 \wedge \mu\ F_0 = F_1)$

$p \leftarrow q.$ $\quad (T_0)$

$q.$

$$\{(s, F_0), (p, F_0), (q, F_0), (r, F_0)\}$$

$$\{(s, F_0), (p, F_0), (q, T_0), (r, F_0)\}$$

$$\{(s, F_1), (p, T_0), (q, T_0), (r, F_0)\}$$

$$\{(s, T_1), (p, T_0),$$

Fixed-point semantics (cont.)

Example

$s \leftarrow \xi\ p.$	$(\xi\ T_0 = T_1)$
$s \leftarrow q \wedge \mu\ r.$	$(T_0 \wedge \mu\ F_0 = F_1)$
$p \leftarrow q.$	(T_0)
$q.$	(T_0)

$$\{(s, F_0), (p, F_0), (q, F_0), (r, F_0)\}$$

$$\{(s, F_0), (p, F_0), (q, T_0), (r, F_0)\}$$

$$\{(s, F_1), (p, T_0), (q, T_0), (r, F_0)\}$$

$$\{(s, T_1), (p, T_0), (q, T_0),$$

Fixed-point semantics (cont.)

Example

$$
\begin{array}{ll}
\mathsf{s} \leftarrow \xi\ \mathsf{p}. & (\xi\ \mathsf{T}_0 = \mathsf{T}_1) \\
\mathsf{s} \leftarrow \mathsf{q} \wedge \mu\ \mathsf{r}. & (\mathsf{T}_0 \wedge \mu\ \mathsf{F}_0 = \mathsf{F}_1) \\
\mathsf{p} \leftarrow \mathsf{q}. & (\mathsf{T}_0) \\
\mathsf{q}. & (\mathsf{T}_0)
\end{array}
$$

$$\{(\mathsf{s}, \mathsf{F}_0), (\mathsf{p}, \mathsf{F}_0), (\mathsf{q}, \mathsf{F}_0), (\mathsf{r}, \mathsf{F}_0)\}$$

$$\{(\mathsf{s}, \mathsf{F}_0), (\mathsf{p}, \mathsf{F}_0), (\mathsf{q}, \mathsf{T}_0), (\mathsf{r}, \mathsf{F}_0)\}$$

$$\{(\mathsf{s}, \mathsf{F}_1), (\mathsf{p}, \mathsf{T}_0), (\mathsf{q}, \mathsf{T}_0), (\mathsf{r}, \mathsf{F}_0)\}$$

$$\{(\mathsf{s}, \mathsf{T}_1), (\mathsf{p}, \mathsf{T}_0), (\mathsf{q}, \mathsf{T}_0), (\mathsf{r}, \mathsf{F}_0)\}$$

Fixed-point semantics (cont.)

Example

$$\mathsf{s} \leftarrow \xi\ \mathsf{p}. \qquad (\xi\ \mathsf{T}_0 = \mathsf{T}_1)$$

$$\mathsf{s} \leftarrow \mathsf{q} \wedge \mu\ \mathsf{r}. \qquad (\mathsf{T}_0 \wedge \mu\ \mathsf{F}_0 = \mathsf{F}_1)$$

$$\mathsf{p} \leftarrow \mathsf{q}. \qquad (\mathsf{T}_0)$$

$$\mathsf{q}. \qquad (\mathsf{T}_0)$$

$$\{(\mathsf{s}, \mathsf{F}_0), (\mathsf{p}, \mathsf{F}_0), (\mathsf{q}, \mathsf{F}_0), (\mathsf{r}, \mathsf{F}_0)\}$$

$$\{(\mathsf{s}, \mathsf{F}_0), (\mathsf{p}, \mathsf{F}_0), (\mathsf{q}, \mathsf{T}_0), (\mathsf{r}, \mathsf{F}_0)\}$$

$$\{(\mathsf{s}, \mathsf{F}_1), (\mathsf{p}, \mathsf{T}_0), (\mathsf{q}, \mathsf{T}_0), (\mathsf{r}, \mathsf{F}_0)\}$$

$$\{(\mathsf{s}, \mathsf{T}_1), (\mathsf{p}, \mathsf{T}_0), (\mathsf{q}, \mathsf{T}_0), (\mathsf{r}, \mathsf{F}_0)\}$$

$$\{(\mathsf{s}, \mathsf{T}_1), (\mathsf{p}, \mathsf{T}_0), (\mathsf{q}, \mathsf{T}_0), (\mathsf{r}, \mathsf{F}_0)\}$$

Summary

- This work

Summary

- This work
  - **[Rondogiannis-Troumpoukis, JANCL 2013]**

Summary

- This work
  - **[Rondogiannis-Troumpoukis, JANCL 2013]**
  - **[Troumpoukis, Master's thesis, UoA 2012]**

Summary

- This work
  - **[Rondogiannis-Troumpoukis, JANCL 2013]**
  - **[Troumpoukis, Master's thesis, UoA 2012]**
- Future work

Summary

- This work
  - **[Rondogiannis-Troumpoukis, JANCL 2013]**
  - **[Troumpoukis, Master's thesis, UoA 2012]**
- Future work
  - Alterative semantics.

Summary

- This work
  - **[Rondogiannis-Troumpoukis, JANCL 2013]**
  - **[Troumpoukis, Master's thesis, UoA 2012]**
- Future work
  - Alterative semantics.
  - Proof system.

Summary

- This work
  - **[Rondogiannis-Troumpoukis, JANCL 2013]**
  - **[Troumpoukis, Master's thesis, UoA 2012]**
- Future work
  - Alterative semantics.
  - Proof system.
  - More preference operators.

Summary

- This work
  - **[Rondogiannis-Troumpoukis, JANCL 2013]**
  - **[Troumpoukis, Master's thesis, UoA 2012]**
- Future work
  - Alterative semantics.
  - Proof system.
  - More preference operators.
  - Expansion of the set of truth values.

Bibliography

Panos Rondogiannis and Antonis Troumpoukis.

The infinite-valued semantics: overview, recent results and future directions.

Journal of Applied Non-Classical Logics, 23(1-2):213–228, 2013.

Panos Rondogiannis and William W. Wadge.

Minimum model semantics for logic programs with negation-as-failure.

ACM Trans. Comput. Log, 6(2):441–467, 2005.

Antonis Troumpoukis.

Logic programming with preferences using infinitesimal logic.

Master's thesis, University of Athens, 2012.

Ruchi Agarwal.

A framework for expressing prioritized constraints using infinitesimal logic.

Master's thesis, University of Victoria, 2005.