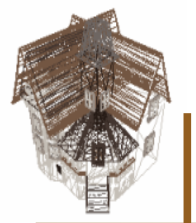


ASP – Answer Set Programming
An operational formalism *(Baral 2003)*

A program Π is a set of expression ρ

$$\rho : L_0 \; or \; L_1 \; or \ldots L_k \leftarrow L_{k+1}, L_{k+2}, \ldots L_m, not \; L_{m+1}, \ldots, not \; L_n$$

where

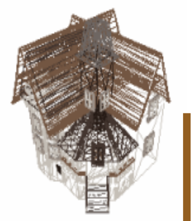
- the L_i are literals ***(atoms or atom negations)***
- The « not » is a negation by failure

Intuitive meaning: for all Herbrand interpretation such that

$\{L_{k+1}, L_{k+2}, \ldots , L_m\}$ is true

while $\{L_{m+1}, \ldots, L_n\}$ failed to be proved

one can derive $\{L_0, L_1, \ldots L_k\}$



An Artificial Agent

$$
\begin{aligned}
&act(P,S,\mathbf{G},\mathbf{A}) \leftarrow person(P),\\
&\quad situation(S),\ goal(G),\ action(A),\\
&\quad will(P,\ S,\ \mathbf{G}),\\
&\quad solve\_goal(P,\ S,\ G,\ \mathbf{A}).
\end{aligned}
$$

Autonomy

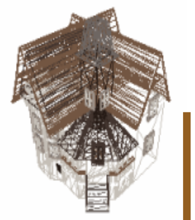
Intelligence

$$\leftarrow act(P,\ S,\ G,\ A),\ act(P,\ S,\ G,\ B),\ A \neq B.$$

An Ethical Artificial Agent

$$\begin{aligned}
&act(P,S,\mathbf{G},\mathbf{A}) \leftarrow person(P),\\
&\quad situation(S),\ goal(G),\ action(A),\\
&\quad will(P,\ S,\ \mathbf{G}),\\
&\quad solve\_goal(P,\ S,\ G,\ \mathbf{A}),\\
&\quad \mathbf{moral(P,\ S,\ G,\ A)}.
\end{aligned}$$

$$\leftarrow act(P,\ S,\ G,\ A),\ act(P,\ S,\ G,\ B),\ A \neq B.$$



An “Aristotelian” Perspective

Predicates:

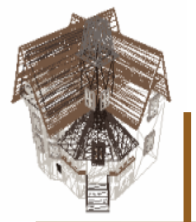
`csq(A,S,C)`: *consequence*

`worse(A,B)`: *comparison of <u>action</u>*

`worst_csq(A,S,C)`: *worst consequence*

*good(P,S,**G**,**A**)* ← *will(P, S, **G**),*
solve_goal(P,S,G,A),worst_csq(A,S,C),
will(P,S,U), solve_goal(P,S,U,B),
csq(B,S,D), worse(D,C).

*bad(P,S,**G**,**A**)* ← *will(P, S, **G**),*
solve_goal(P,S,G,A),worst_csq(A,S,C),
will(P,S,U), solve_goal(P,S,U,B), A ≠ B, csq(B,S,D),
***not** worse(D,C).*
The same action may be both good and bad!

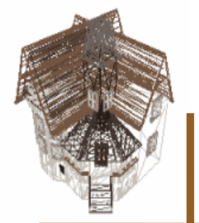


Moral and immoral

$$moral(P,S,\boldsymbol{G},\boldsymbol{A}) \leftarrow \textbf{\textit{not}}\ bad(P,S,G,A).$$

$$moral(P,S,\boldsymbol{G},\boldsymbol{A}) \leftarrow good(P,S,G,A),$$
$$\textbf{\textit{not}}\ immoral(P,S,G,A).$$

$$immoral(P,S,\boldsymbol{G},\boldsymbol{A}) \leftarrow bad(P,S,G,A),$$
$$\textbf{\textit{not}}\ moral(P,S,G,A).$$



The Lying Example

- Three persons: “I”, Peter and Paul
- Two possibilities: tell(P, truth) or tell(P, lie)
- Consequence: tell(“I”, truth) generates a murder

csq(A,S,A) ← .

csq(A,S,B) ← csq(A,S,C), csq(C,S,B).

csq(tell(“I”, truth),s0,murder) ←.

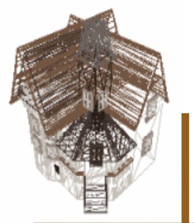
worse(A,B) ← better(B,A), ***not*** *better(A,B).*

worse(A,B) ← worse(A,C), worse(C,B).

better(A, tell(P, lie))←.

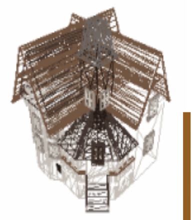
better(A, murder)←.

better(A,A) ←.



The Lying Example

- Half of the answser sets contain: *act("I", answer("I"),s0, tell("I", truth)*
- And half of the answser sets contain: *act("I", answer("I"),s0, tell("I", lie)*
- If we add *worse(murder, lie)* then all the answer sets that contain *act("I", answer("I"),s0, tell("I", truth)* are removed.



Torture example

- Three persons: “I”, Peter and Paul
- Two possibilities: interrogate(P, torture) or interrogate(P, soft)
- Consequence: interrogate(“I”, soft) generates an attack

$csq(A,S,A) \leftarrow .$

$csq(A,S,B) \leftarrow csq(A,S,C),\ csq(C,S,B).$

$csq(interrogate(\text{“}I\text{”},\ soft),s0,attack) \leftarrow.$

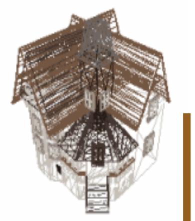
$worse(A,B) \leftarrow better(B,A),\ \mathbf{not}\ \ better(A,B\ ).$

$worse(A,B) \leftarrow worse(A,C),\ worse(C,B).$

$better(A,\ interrogate(P,\ torture))\leftarrow.$

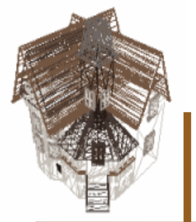
$better(A,\ attack)\leftarrow.$

$better(A,A)\ \leftarrow.$



The Torture Example

- Half of the answser sets contain: *act(“I”, question(“I”),s0, interrogate(“I”, torture)*
- And half of the answser sets contain: *act(“I”, question(“I”),s0, interrogate(“I”, soft)*
- If we add *worse(attack, torture)* then all the answer sets that contain *act(“I”, question(“I”),s0, interrogate(“I”, soft)* are removed.



A Kantian Machine

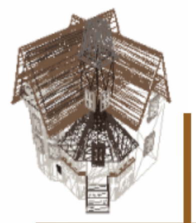
Requirements for a possible society

The maxim of my will has to be universalized – "categorical imperative"

- My acts have to obey to a law
- I must act *by the law* – and not just *in accordance to the law*
- My rule of behavior (my maxim) could be universal

If I adopt a right to lie (even in some conditions), I must conceive a world where everybody could act in the same way, which renders impossible to trust anyone.

In the same way, if I decide to suicide, because I am suffering to much, I must conceive a world…



The Kantian Perspective

$$act(P,S,\mathbf{G},A) \leftarrow person(P),$$
$$situation(S),\ goal(G),\ action(A),$$
$$\boldsymbol{will(P,\ S,\ G)},$$
$$\boldsymbol{solve\_goal(P,\ S,\ G,\ A)},$$
$$\boldsymbol{maxim(P,\ S,\ A)}.$$

"Prudence": pragmatic imperative

"Habileté": problematic imperative

Morality: moral imperative

$$\leftarrow act(P,\ S,\ G,\ A),\ act(P,\ S,\ G,\ B),\ A \neq B.$$

The Kantian Perspective

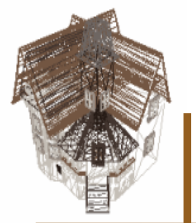
*act(P,S,**G**,**A**) ← person(P),*
situation(S), goal(G), action(A),
***will(P, S, G)**,* ← Prudence
***solve_goal(P, S, G, A)**,* ← “Habileté”
***maxim(P, S, A)**.* ← “Moralité”

← act(P, S, G, A), act(P, S, G, B), A ≠ B.

*The **<u>categorical imperative (morality)</u>***

maxim(P,S,A) ← maxim(“I”, S, B), bind(“I”, B, P, A).

bind(P, tell(P, U), Q, tell(Q, U)) ← .



The Lying Example

- Categorical Imperative

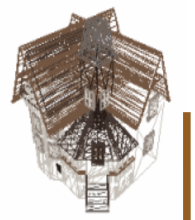
```
maxim_will(P, S, A) :-
          maxim_will(“I”, S, B),
          bind(“I”, B, P, A),
          not maxim_will(P, S, C),
          incompatible(A, C).
```

- Conséquences

```
consequence(A, S, A).
consequence(tell("I", truth), s0, murder).
consequence(tell(peter, truth), s0, murder).
```

- Situation

```
maxim_will(peter, S, tell(peter, lie)) :-
      consequence(tell(peter, truth), S, murder).
```



A Meta-Ethical Requirement

- In a given society, I need to trust at least one person…

$$\mathit{untrust}(P) \leftarrow \mathit{maxim}(P, S, \mathit{tell}(P, \mathit{lie})).$$

$$\mathit{trust}(P) \leftarrow \textbf{\textit{not}}\ \mathit{untrust}(P).$$

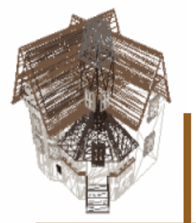
$$\mathit{possible\_society} \leftarrow \mathit{trust}(P).$$

$$\leftarrow \textbf{\textit{not}}\ \mathit{possible\_society}.$$

- The lying example:

  If Paul is lying, there is no real problem

  If I lie, there is no acceptable solution…because I cannot trust anyone in the society.



The Torture Example

- Categorical Imperative

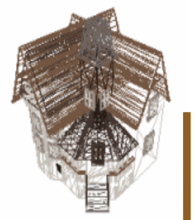
```
maxim_will(P, S, A) :- maxim_will(“I”, S, B),
          bind(“I”, B, P, A),
          not maxim_will(P, S, C), incompatible(A, C).
```

- Conséquences

```
consequence(A, S, A).
consequence(interrogate("I", soft), s0, attack).
consequence(interrogate(peter, soft), s0, attack).
```

- Situation

```
maxim_will(peter, S, interrogate(peter, torture) :-
   consequence(interrogate(peter, soft), S, attack).
```



A Meta-Ethical Requirement

- If P tortures, the tortured person is not considered as a subject, but as a mean to get the truth…

instrumentalize_person(P) ←
maxim(P, S, interrogate(P, torture)).

confidence(P) ← ***not*** *instrumentalize_person(P).*

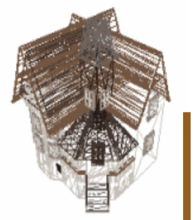
possible_society ← confidence(P).

*←****not*** *possible_society.*

- The torture example:

If Paul tortures, there is no real problem (?)

If I torture, there is no acceptable solution…because I cannot be in confidence with anyone in the society.



The Suicide Example

- Categorical Imperative

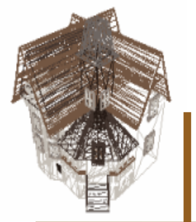
```
maxim_will(P, S, A) :- maxim_will(“I”, S, B),
          bind(“I”, B, P, A),
          not maxim_will(P, S, C), incompatible(A, C).
```

- Conséquences

```
consequence(A, S, A).
consequence(take_care(“I”), s0, loose_dignity(“I”).
consequence(take_care(peter), s0, loose_dignity(peter)).
```

- Situation

```
maxim_will(“I”, S, take_care(“I”)).
maxim_will(“I”, S, keep_dignity(“I”)).
contradictory(keep_dignity(P), loose_dignity(P)).
```



Meta-ethical criteria

- Once being given Kantian requirements of a possible society, it is possible to define formal criteria, which ensure that a set of ethical axioms characterizing the maxims of will gives one (or more) solution(s).

Examples of such formal criteria: (Baral 2003)

Any stratified AnsProlog Program (*i.e. any program whose dependency graph does not contain any negative cycle*) has a unique answer set

Any signed AnsProlog Program has an answer set

Any order consistent AnsProlog Program has an answer set

B. Constant – System of Principles

$\mathit{act}(P,S,\mathbf{G},\mathbf{A}) \leftarrow \mathit{person}(P),$
$\quad \mathit{situation}(S), \mathit{goal}(G), \mathit{action}(A),$
$\quad \mathit{will}(P, S, \mathbf{G}),$
$\quad \mathit{solve\_goal}(P, S, G, \mathbf{A}),$
$\quad \mathbf{\mathit{principle}(P, S, G, A)}.$

$\leftarrow \mathit{act}(P, S, G, A), \mathit{act}(P, S, G, B), A \neq B.$

tell become a ternary term

answer a binary term

B. Constant – System of Principles

*principle(P,S,answer(P,Q),**tell(P,Q,truth)**) ←*
*   **not** ¬principle(P,S,answer(P,Q), tell(P,Q,truth)).*

*principle(P,S,answer(P,Q),**tell(P,Q,lie)**) ←*
*   demerit(Q, tell(P,Q,truth)).*

*¬principle(P,S,answer(P,Q),**tell(P,Q,truth))** ←*
*   principle(P,S,answer(P,Q),tell(P,Q,lie)).*

demerit(Q, tell(P,Q,truth)) ←
*   worst_csq(tell(P,Q,truth), C),*
*   worse(C, tell(P,Q,lie)).*