Or else, what? Imperatives on the borderline of semantics and pragmatics

Frank Veltman

Logic, Language & Computation

November 28th, 2011

Imperatives.

Compare

- Go!
- John had to go.
- You must go.

The last sentence is ambiguous between a *performative* and a *reportative* reading.

I want more than just to explain what it means for a command to be *in force*. How can we model the performative use?

Dynamic semantics

Slogan: You know the meaning of a sentence if you know the change it brings about in the cognitive state of anyone who wants to incorporate the information conveyed by it.

- The meaning $[\varphi]$ of a sentence φ is an operation on cognitive states.

Let S be an cognitive state and φ a sentence with meaning $[\varphi]$. We write

$$S[\varphi]$$

for the cognitive state that results when S is updated with φ.

Key notions

Support Sometimes the information conveyed by φ will already be subsumed by S. In this case, we say that φ *is accepted* in S, or that S *supports* φ, and we write this as $S \models \varphi$. In simple cases this relation can be defined as follows:

- $S \models \varphi$ iff $S[\varphi] = S$

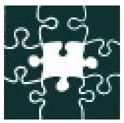
Logical validity An argument is *valid* if updating any state with the premises, yields a state that supports the conclusion.

- $\varphi_1, \ldots, \varphi_n \models \psi$ iff for every state S, $S[\varphi_1] \ldots [\varphi_n] \models \psi$.

Imperatives in dynamic semantics

Basic idea: An imperative α! – if it is accepted – induces a *change of intentions* in the cognitive state of the addressee.

For English α is just an uninflected intransitive verb phrase.



Puzzle 1: Contradiction?

One doctor tells you: *Don't drink milk!*

Another doctor gives the advise: *Drink milk or apple juice!*

Would you trust both and conclude that you should drink apple juice?

Puzzle 2: *A variant of the miners paradox*

If the miners are in shaft A, block shaft A!
If the miners are in shaft B, block shaft B!
The miners are either in shaft A or in shaft B.

$\therefore$ *Block shaft A or shaft B!*

Is this a valid inference?

Some background

"Ten miners are trapped either in shaft A or in shaft B, but we do not know which. Flood waters threaten to flood the shafts. We have enough sandbags to block one of the shafts, but not both. If we block one shaft, all the water will go into the other shaft, killing all miners inside of it. If we block neither shaft, both shafts will fill halfway with water, and just one miner, the lowest in the shaft, will be killed."

Taken from: Kolodny, N. & J. Macfarlane, 'Ifs and Oughts', *The Journal of Philosophy*, 2010, 115-143.

Puzzle 3: pseudo imperatives

- *Stop or I'll shoot you.*

- *Stop and I will make you happy.*

- *Stop and I'll shoot you.*

- *Stop or I will make you happy.*

Puzzle 3: pseudo imperatives

- *Stop or I'll shoot you.*
  (Conditional threat: *if you don't stop, I'll shoot you*)

- *Stop and I will make you happy.*

- *Stop and I'll shoot you.*

- *Stop or I will make you happy.*

Puzzle 3: pseudo imperatives

- *Stop or I'll shoot you.*
  (Conditional threat: *if you don't stop, I'll shoot you*)

- *Stop and I will make you happy.*
  (Conditional promise: *if you stop, I'll make you happy*)

- *Stop and I'll shoot you.*

- *Stop or I will make you happy.*

Puzzle 3: pseudo imperatives

- *Stop or I'll shoot you.*
  (Conditional threat: *if you don't stop, I'll shoot you*)

- *Stop and I will make you happy.*
  (Conditional promise: *if you stop, I'll make you happy*)

- *Stop and I'll shoot you.*
  (Conditional threat: *if you stop, I'll shoot you*)

- *Stop or I will make you happy.*

Puzzle 3: pseudo imperatives

- *Stop or I'll shoot you.*
  (Conditional threat: *if you don't stop, I'll shoot you*)

- *Stop and I will make you happy.*
  (Conditional promise: *if you stop, I'll make you happy*)

- *Stop and I'll shoot you.*
  (Conditional threat: *if you stop. I'll shoot you*)

- *Stop or I will make you happy.* (??)

- *Stop or I will make you happy.* (??)

Why is so difficult to interpret the last example as a conditional promise (*If you don't stop, I'll make you happy*).

Language

Take a language $\mathcal{L}$ of propositional logic (with $\wedge, \vee, \neg$ as logical constants), and add the following clauses:

(i) If φ is a formula of $\mathcal{L}$, then $!\varphi$ is an imperative.

(ii) . . .

Read '$!\varphi$' as *'Make φ true!'*

States

Ingredients: information about the actual world, plans, possible results.

- a *to-do list* is a set of pairs $\langle p, \mathbf{x} \rangle$, with p an atomic sentence and $\mathbf{x} \in \{\mathbf{true}, \mathbf{false}\}$;

- A to-do list l is *consistent* iff there is no p such that both $\langle p, \mathbf{true} \rangle \in l$ and $\langle p, \mathbf{false} \rangle \in l$.

States (continued)

- a *plan* is a set of consistent to-do lists, none of which is a proper subset of another.

- $\{\emptyset\}$ is the *minimal plan*. (It consists of an empty to-do list).

- the empty plan $\emptyset$ is also called the *absurd* plan.

This is a picture of a plan

true	*false*
p	r
q	

true	*false*
p	r
s	

Updating plans *

$$
\begin{array}{llcl}
\textit{atom}: & \Pi{\uparrow}p & = & min\{l' \mid l' \text{ is consistent and} \\
 & & & l' = l \cup \{\langle p, \mathbf{make\,true}\rangle\} \text{ for some list } l \in \Pi\} \\
 & \Pi{\downarrow}p & = & min\{l' \mid l' \text{ is consistent and} \\
 & & & l' = l \cup \{\langle p, \mathbf{make\,false}\rangle\} \text{ for some list } l \in \Pi\} \\
\neg: & \Pi{\uparrow}\neg\varphi & = & \Pi{\downarrow}\varphi \\
 & \Pi{\downarrow}\neg\varphi & = & \Pi{\uparrow}\varphi \\
\wedge: & \Pi{\uparrow}(\varphi \wedge \psi) & = & \Pi{\uparrow}\varphi{\uparrow}\psi \\
 & \Pi{\downarrow}(\varphi \wedge \psi) & = & min(\Pi{\downarrow}\varphi \cup \Pi{\downarrow}\psi) \\
\vee: & \Pi{\uparrow}(\varphi \vee \psi) & = & min(\Pi{\uparrow}\varphi \cup \Pi{\uparrow}\psi) \\
 & \Pi{\downarrow}(\varphi \vee \psi) & = & \Pi{\downarrow}\varphi{\downarrow}\psi
\end{array}
$$

*Let Σ be a set of to-do lists.
Then $min\,\Sigma = \{l \in \Sigma \mid \text{there is no } l' \in \Sigma \text{ such that } l' \subsetneq l\}$

Example

We construct $\{\emptyset\}\uparrow(q \vee r)\uparrow\neg p\uparrow q$.

First, the empty plan $\{\emptyset\}$:

true	*false*

Example

Next, $\{\emptyset\}{\uparrow}(q \vee r)$

true	*false*
q	

true	*false*
r	

Example

Then, $\{\emptyset\}\uparrow(q \vee r)\uparrow\neg p$

true	*false*
q	p

true	*false*
r	p

Example

And finally, $\{\emptyset\}\uparrow(q \vee r)\uparrow\neg p\uparrow q$

true	*false*
q	p

Fact

For plans with *complete* lists, the definition reduces to the well-known eliminative definition:

$$
\begin{array}{rrcl}
\textit{atom}: & \Pi{\uparrow}p & = & \{l \in \Pi \mid \langle p, \mathbf{true}\rangle \in l\} \\
 & \Pi{\downarrow}p & = & \{l \in \Pi \mid \langle p, \mathbf{false}\rangle \in l\} \\
\neg: & \Pi{\uparrow}\neg\varphi & = & \Pi{\downarrow}\varphi \\
 & \Pi{\downarrow}\neg\varphi & = & \Pi{\uparrow}\varphi \\
\wedge: & \Pi{\uparrow}(\varphi \wedge \psi) & = & \Pi{\uparrow}\varphi{\uparrow}\psi \\
 & \Pi{\downarrow}(\varphi \wedge \psi) & = & \Pi{\downarrow}\varphi \cup \Pi{\downarrow}\psi \\
\vee: & \Pi{\uparrow}(\varphi \vee \psi) & = & \Pi{\uparrow}\varphi \cup \Pi{\uparrow}\psi \\
 & \Pi{\downarrow}(\alpha \vee \psi) & = & \Pi{\downarrow}\alpha{\downarrow}\psi
\end{array}
$$

Merging plans

The *merge* $\Pi \sqcup \Pi'$ of two plans Π and Π' is given by the set $min\{l'' \mid l''$ is consistent and $l'' = l \cup l'$ for some $l \in \Pi$ and $l' \in \Pi'\}$

Proposition (decomposition lemma)

$$\text{For every } \varphi, \Pi{\uparrow}\varphi = \Pi \sqcup \{\emptyset\}{\uparrow}\varphi$$

Two more notions

- Π *fits in* Π' iff if for every list $l \in \Pi$ there is some list $l' \in \Pi'$ such that $l \cup l'$ is consistent.

- Π is *compatible with* Π' iff Π is fits in Π' and *vice versa.*

Updating a plan Π with an imperative

(i) $\Pi[!\varphi] = \Pi{\uparrow}\varphi$ if Π is compatible with $\{\emptyset\}{\uparrow}\varphi$.

(ii) $\Pi[!\varphi] = \emptyset$ if Π is not compatible with $\{\emptyset\}{\uparrow}\varphi$.

Example of compatible plans

This is $\{\emptyset\}[!(p \vee q)]$:

true	*false*
p	

true	*false*
q	

It is compatible with $\{\emptyset\}[!\neg(p \wedge q)]$

true	*false*
	p

true	*false*
	q

Example of compatible plans

This is the result: $\{\emptyset\}[!(p \vee q)][!\neg(p \wedge q)]$

true	*false*
p	q

true	*false*
q	p

Contradiction?

One doctor tells you: *Don't drink milk!*

Another doctor gives the advise: *Drink milk or apple juice!*

Would you trust both and conclude that you should drink apple juice?

Example of incompatible plans

The prescription to drink milk or apple juice looks like this

true	*false*
milk	

true	*false*
apple juice	

The prescription not to drink milk gives the plan

true	*false*
	milk

States

(i) a *world* is a function w that assigns to every atomic sentence p one of the truth values $\mathbf{true}$ or $\mathbf{false}$;

(ii) a *state* S is a triple $\langle W, P, R \rangle$ such that

(a) W is a nonempty set of worlds.

(b) P is a function that assigns to every world a plan $P(w)$.

(c) R is a function that assigns to every world $w \in W$ a set of worlds $R(w)$. For every $w' \in R(w)$ there is some list $l \in P(w)$ such that l has been realized in w'.

States

Let $S = \langle W, P, R \rangle$ be a cognitive state.

- If $w \in W$ then for all an agent in that cognitive state S knows, w might be the actual world.

- If $w \in W$, then $P(w)$ is the plan the agent has developed for w. Different worlds may come with different plans.

- If $w \in W$, and $v \in R(w)$ then v is a *possible successor* of w. Every successor of w realises one of the options of the plan for w. (An agent intends to carry out his plans)

Special States

- the *minimal state* is given by the triple $\langle W, P, R\rangle$ such that (i) W is the set of all possible worlds, (ii) for every $w \in W$, $P(w) = \{\emptyset\}$, (iii) for all $w \in W$, $R(w) = W$.

- a state $S = \langle W, P, R\rangle$ is *absurd* iff either (i) $W = \emptyset$, or (ii) there is some $w \in W$ such that $P(w) = \emptyset$, or (iii) there are some $w \in W$, and $l \in P(w)$ for which there is no $v \in R(w)$ such that $l \subseteq v^*$, or (iv) for some w, $R(w) = \emptyset$.

*In this case we say that the plan $P(w)$ is *not executable* in w.

Updating a state S with a descriptive sentence φ

$\langle W, P, R\rangle[\varphi] = \langle W', P', R'\rangle$, where

- $W' = W{\uparrow}\varphi = \{w \in W \mid \varphi$ is true in w$\}$
- $P' = P \restriction W'$
- $R' = R \restriction W'$

Updating a state S with an imperative $!\varphi$

$\langle W, P, R\rangle[!\varphi] = \langle W, P', R'\rangle$, where for every $w \in W$, $P'(w)$ is given by:

$P'(w) = P(w)\uparrow\varphi$, provided that (a) $P(w)$ is compatible with $\{\emptyset\}\uparrow\varphi$, and (b) $P(w)\uparrow\varphi$ is executable in w. Otherwise, $P'(w) = \emptyset$.

$w' \in R'(w)$ iff $w' \in R(w)$ and there is some list $l \in P'(w)$ such that l is realized in w'.

Updating a state S with a formula of the form will φ

If φ is a formula of propositional logic,

$\langle W, P, R\rangle[will\,\varphi] = \langle W, P, R'\rangle$, where R' is given by:

for every $w \in W$, $R'(w) = R(w)\uparrow\varphi = \{v \in R(w) \mid \varphi$ is true in $v\}$.

The Miners Paradox

If the miners are in shaft A, block shaft A!
If the miners are in shaft B, block shaft B!
The miners are either in shaft A or in shaft B.

$\therefore$ *Block shaft A or shaft B!*

Is this a valid inference?

Conditional commands

$\langle W, P, R\rangle[\textit{If } \varphi, !\psi] = \langle W, P', R\rangle$, where for every $w \in W$, $P'(w)$ is given by:

(i) if $w \notin W{\uparrow}\varphi$, then $P'(w) = P(w)$.

(ii) if $w \in W{\uparrow}\varphi$, then $P'(w) = P(w){\uparrow}\psi$, provided that (a) $P(w)$ is compatible with $\{\emptyset\}{\uparrow}\psi$, and (b) $P(w){\uparrow}\psi$ is executable in w; otherwise, $P'(w) = \emptyset$.

The Miners Paradox

Consider the state that you get when you update the minimal state with

$$\textit{in-}A \vee \textit{in-}B$$
$$\neg(\textit{in-}A \wedge \textit{in-}B)$$
$$\textit{If in-}A, !\textit{blocked-}\mathsf{A}$$
$$\textit{If in-}B, !\textit{blocked-}\mathsf{B}$$
$$\textit{If in-}A, !\neg\textit{blocked-}\mathsf{B}$$
$$\textit{If in-}B, !\neg\textit{blocked-}\mathsf{A}$$

In the resulting state $!(A\textit{-blocked} \vee B\textit{-blocked})$ is not acceptable.

Other Mixed Moods

How about

- $!\varphi \vee will\,\psi$
- $!\varphi \wedge will\,\psi$

Here we have to look closer at the way imperatives are processed in particular at the uptake of the imperative.

See to it that p! (the normal case)

true	*false*
p	
...	...

result: p

In many cases (normally?) the speaker wants p to be made true, whereas the hearer prefers $\neg p$ to p, or for some other reason would not by himself choose to make p true.

Gricean maxims for imperatives

Quality^i: A sincere speaker should only assert $!\varphi$ if he or she really wants the hearer to make φ true.

Quantity^i: The speaker should only order (advise, beg, etc.) the hearer to make φ true, if it's really needed, i.e. if it looks like the hearer is not going to make φ true spontaneously.

Wittgenstein, *Tractatus 6.422*

Der erste Gedanke bei der Aufstellung eines etischen Gestzes von der Form ‘du sollst. . . ’ ist: ‘Und was dann wenn ich es nicht tue?’

(When an ethical law of the form, ’Thou shalt ...’ is laid down, one’s first thought is, ’And what if I do not do it?’)

See to it that p!

true	*false*
p	
...	...

result: p

And what if I don't see to it that p?

true	*false*
p	
...	...

result: p

true	*false*
	(p)
...	...

result: $(\neg p)$

How to persuade the hearer

true	*false*
p	
...	...

true	*false*
	(p)
...	...

result: p

$+$

reward

result: $(\neg p)$

How to persuade the hearer

true	*false*
p	
...	...

true	*false*
	p
...	...

result: p

result: $\neg p$

$+$

penalty

How to persuade the hearer

true	*false*
p	
...	...

result: p

$+$

reward

true	*false*
	p
...	...

result: $\neg p$

$+$

penalty

The case of the ten commandments

true	*false*
p	
...	...

true	*false*
	p
...	...

result: p
$+$
Heaven

result: $\neg p$
$+$
Hell

Close the door or I will kick you

true	*false*
close the door	

true	*false*
	close the door

result: the door is closed

result: the door is open

+

I kick you

Close the door and I will kiss you

true	*false*
close the door	

true	*false*
	close the door

result: the door is closed
+
I kiss you

result: de door is open

Close the door and I will kiss you

true	*false*
close the door	

true	*false*
	close the door

result: the door is closed
+
I kiss you

result: de door is open

Compare:

(a) *Close the door. I will kiss you*

(b) *I will kiss you and close the door*

true	*false*
close the door	

true	*false*
	close the door

result: the door is closed
+
I kiss you

result: de door is open

Notice that this *hybrid* state supports

If you close the door, I will kiss you

Close the door and I will kick you

true	*false*
close the door	

true	*false*
	close the door

result: the door is closed
+
I kick you

result: de door is open

Some observations

- Assuming that the speaker really wants the door being closed, there is a direct clash, since (s)he puts a penalty on closing it.

- Compare: *I beg you, please, close the door and I will kick you.*

- In many cases, this is a reaction to the hearer's announcement that (s)he is going to close the door.

Close the door or I will kiss you

true	*false*
close the door	

true	*false*
	close the door

result: the door is closed

result: the door is open

+

I kiss you

Thank You!