Moods and negation

Berislav Žarnić
University of Split

1 Two ways of negating a sentence

According to the received view (for example [9]), there are three logico-semantic moods: indicative, imperative and interrogative, and there are two main components in natural language sentences: modal element (which determines sentence's mood) and sentence radical. On the other hand, it seems that the received view on the sentence components is challenged by the fact that there are two ways to negate a sentence: by negating its content (its radical) and by performing a negative speech act. Alf Ross, who was one of the founders of imperative logic, drew a cognate distinction: there are imperatives with "negated theme of demand" and there are imperatives with "negated factor of demand".

> The use of the imperative mood in colloquial language does not allow this important difference between $I(\bar{x})$ and $\bar{I}(x)$ to be clearly marked. All imperative in the grammatical sense are positive in the sense that they poses a positive factor of demand. For example, "Do not close the door!" can only mean $I(\bar{x})$ not $\bar{I}(x)$. Only by using linguistically indicative mood the difference becomes apparent. For example, "It is your duty not to close the door" $(I(\bar{x}))$, and "It is not your duty to close the door" $(\bar{I}(x))$.
>
> [7] p. 63

Recently, the distinction (between two ways of negating a sentence) as it applies to indicatives has been discussed by Tappenden [10]: in a derivative sense, to deny that S is the same as to assert that $\neg S$, while, in a non-derivative sense, to deny that S is irreducible speech act of "the commitment to the failure to obtain of the conditions that would have to obtain for S to be true".

1.1 Modeling two kinds of negation in dynamic semantics

The second type of negation (e.g. the imperative with negated "factor of demand", the denial as irreducible speech act) is what I call 'negated speech act'. Within the framework of dynamic semantics negated speech act of asserting φ could be modeled as a semantic action which makes asserting $\neg\varphi$ acceptable. Using language of dynamic modal logic (De Rijke [6]), this kind of negation

(negation of a speech act) could be described as

$$con\,(do\,(exp\,(\neg\varphi))) = \{\langle x, y\rangle \mid y \sqsubseteq x \wedge \exists z\,(\langle y, z\rangle \in exp\,(\neg\varphi))\}\,,$$

i.e. it is contingent moving down along information ordering towards an equally or less informative state which enables acceptance of $\neg\varphi$. It is important to note that for the case of 'denial in a non-derivative sense' withdrawal of a sentence does not include acceptance of a new one.

Dynamic modal logic takes a relational approach while update semantics [11] takes the functional one. A loose connection between dynamic modal logic and update-downdate variant of Veltman's system could be established by the following propositions. Let $\sigma[\varphi^{Up}] \neq \emptyset$, $\sigma[\varphi^{Down}] \neq \emptyset$, $\sigma[\varphi^{Test}] \neq \emptyset$:

$$\begin{array}{rcl} \left\langle \sigma, \sigma[\varphi^{Up}]\right\rangle & \in & \|ex(fix(\varphi))\|\,, \\ \left\langle \sigma, \sigma[\varphi^{Down}]\right\rangle & \in & \|con(do(ex\neg\varphi))\|\,, \\ \left\langle \sigma, \sigma[\varphi^{Test}]\right\rangle & \in & \|do\,(\varphi)?\|\,. \end{array}$$

The proposed "downdate" modeling for "negated speech act" shows that its notion depends on the notion of speech act with negated content.

For the purpose of modeling denial as negated speech act of assertion we will introduce downdate function into Veltman's update system and introduce a language that formalizes speech acts performed by uttering indicative sentences (assertion, denial in a non-derivative sense, informative suggestion).

Definition 1 *For a finite set A of propositional letters,$W = \wp A$, $\sigma \subseteq W$ is an informational state over A.*

Definition 2 *If φ is a sentence of classical propositional logic built over finite set A of propositional letters, then φ^{Up}, φ^{Down} and φ^{Test} are sentences of L^{ACT}.*

Definition 3 *Truth definition: [atoms] $h(P, w) = \top$ iff $P \in w$; [compounds] $h(\neg\varphi, w) = \top$ iff it is not the case that $h(\varphi, w) = \top$; $h(\varphi \wedge \psi, w) = \top$ iff $h(\varphi, w) = \top$ and $h(\psi, w) = \top$; $h(\varphi \vee \psi, w) = \top$ iff $h(\varphi, w) = \top$ or $h(\psi, w) = \top$; and so on in the manner of classical propositional logic.*

Definition 4 *Difference relation D between members of W with respect to proposition φ:*

$$D^W\,(\varphi) = \{\langle w, v\rangle \mid w \in W \wedge v \in W \wedge h(\varphi, w) = \bot \wedge h(\varphi, v) = \top\}\,.$$

Definition 5 *Relation μD of minimal difference with respect to proposition φ:*

$$\mu D^W\,(\varphi) = \left\{\langle w, v\rangle \in D^W\,(\varphi) \mid \neg\exists z \exists u\left(\langle z, u\rangle \in D^W\,(\varphi) \wedge |z \vartriangle u| < |w \vartriangle v|\right)\right\},$$

where $\vartriangle$ stands for symmetrical difference.

Example 6 $\mu D^W(\top) = \mu D^W(\bot) = \emptyset$

Definition 7 *Set of "the closest antipodes" for members of σ with respect to φ:*

$$\begin{aligned} {}^{\sigma}\downarrow\varphi\downarrow^{\mu D} &= \left\{v \mid \exists w \left(w \in \sigma \wedge \langle w, v\rangle \in \mu D^{W}(\varphi)\right)\right\} \\ &= mem_2\left((\sigma \times W) \cap \mu D^{W}(\varphi)\right). \end{aligned}$$

Notation 8 *Expressions mem_1 and mem_2 stand for function that delivers first and second members of a binary relation R, i.e. $mem_1(R) = \{x \mid \exists y R(x,y)\}$ and $mem_2(R) = \{y \mid \exists x R(x,y)\}$.*

Definition 9 *Interpretation function $[\cdot]$ takes a state $\sigma \subseteq W$ and a sentence $\varphi \in L^{ACT}$ and delivers a state σ':*

- $\sigma\left[\varphi^{Up}\right] = \{w \in \sigma \mid h(\varphi, w) = \top\}$,

- $\sigma\left[\varphi^{Test}\right] = \begin{cases} \sigma \text{ if } \sigma[\varphi] \neq \emptyset, \\ \emptyset \text{ otherwise,} \end{cases}$

- $\sigma\left[\varphi^{Down}\right] = \begin{cases} \sigma \cup^{\sigma} \downarrow\neg\varphi\downarrow^{\mu D} & \text{if } \sigma[\neg\varphi]^{\ Up} = \emptyset, \\ \sigma \text{ otherwise.} \end{cases}$

Example 10 *Let $w_{pq} = \{p,q\}$, $w_p = \{p\}$, $w_q = \{q\}$, $w_{\emptyset} = \emptyset$. For $A = \{p,q\}$ and $\sigma = \{w_q\}$:*

$$\begin{aligned} \mu D^{W}(p) &= \{\langle w_q, w_{pq}\rangle, \langle w_{\emptyset}, w_p\rangle\}, \\ {}^{\sigma}\downarrow p\downarrow^{\mu D} &= \{w_{pq}\}, \\ \sigma[\neg p]^{\ Down} &= \sigma \cup^{\sigma} \downarrow p\downarrow^{\mu D} = \{w_q, w_{pq}\} \end{aligned}$$

Using the notions of acceptability and acceptance [11], we may introduce four semantic values that a sentence may have at a state:

	ACCEPTABILITY	ACCEPTANCE
$v(\varphi,\sigma)$	$[\varphi]\,\sigma \neq \emptyset$	$[\varphi]\,\sigma = \sigma$
1	Yes.	Yes.
n	Yes.	No.
0	No.	No.
b	No.	Yes.

Propositional negation stays fixed for n and b, while values commute for 1 and 0. The symbols $1, n, 0, b$ are chosen in order to suggest connection to other four-valued logics [3], where n stands for 'neither true nor false' and b stands for 'both true and false'. On the other hand, downdate secures acceptability except for the absurd state.

$v(\varphi,\sigma)$	$v(\neg\varphi,\sigma)$	$v(\neg\varphi,\sigma\left[\varphi^{Down}\right])$
1	0	n
n	n	n
0	1	1
b	b	b

2 Imperatives, commands and permissions

Compared to the language of indicative sentences, imperative language seems to show in a more obvious way the distinction between the speech act, the speech act with negated content and the negated speech act. While the first two are requests, the third is permission. In order to examine the possibility of dynamic modeling of the distinction, we will follow the tradition that links the imperative semantics with action semantics. In particular, we will relay on the following ideas: imperatives contain change expressions, Lemmon [5], the content of imperative is a prescribed action, Belnap [1], Segerberg [8], the semantics of action requires existence of negative condition (counter-state, "null-point", avoidability) Kanger [4], Belnap [1], Von Wright [12]. According to Von Wright, a minimal semantics of action should delineate the following three elements: (1) the initial state, which the agent changes or which would have changed if the agent had not been active, (2) the end-state, which results from the action and (3) the counter-state, which would have resulted from agent's passivity. On that grounds Von Wright developed fourfold classification of actions: producing $(\neg\varphi/\varphi)$, destroying $(\varphi/\neg\varphi)$, sustaining (φ/φ) and suppressing $(\neg\varphi/\neg\varphi)$ state of affairs φ. The classification of actions can be used as the basis of twofold classification of imperatives: 1. complementary imperatives, which are used for requesting production or destruction of a state of affairs: $!(\neg\varphi/\varphi)$, $!(\varphi/\neg\varphi)$, 2. symmetric imperatives, which are used for requesting maintenance or suppression of a state of affairs: $!(\varphi/\varphi)$, $!(\neg\varphi/\neg\varphi)$. To those two a third type of imperatives should be added: "one-sided" imperatives $!(\top/\varphi)$, $!(\top/\neg\varphi)$, which have drawn much attention in the literature.

Example 11 *Let C stand for 'The door is closed'. (i) 'Close the door!' and (ii) 'Don't close the door!' are complementary $!(\neg C/C)$ and symmetric imperatives $!(\neg C/\neg C)$, respectively. Intuitively, they are used for the same kind of speech act, namely, request. Their contents differ and each may be understood as having negated content with respect to the other. On the other hand, permission expressed by (iii) 'You don't have to close the door' or 'You may leave the door open' relate to imperative (i) as a negation of speech act performed by uttering (i).*

Following Von Wright's action semantics, the semantics for imperatives as commanded actions should include: two moments, $before$ and $after$, relation of commanded changes, and set of possible $after$ situations.

Example 12 *The meaning of complementary imperative 'Close the door!' may be depicted by its implications: (initial state) 'The door is open at the moment $before$', (end-state) 'The door shall (ought to) be closed at the moment $after$', (negative condition) 'It is possible that the door will not be closed at the moment $after$', (positive condition) 'It is possible that the door will be closed at the moment $after$'.*

According to the proposed approach, the negated speech act a performed by uttering the sentence φ (where φ stands for a full fledged sentence and not for its radical) is conceived as a semantic action $con(do(exp(\neg\varphi)))$ which enables acceptance of the speech act a with the negated content $\neg\varphi$. In order to apply the approach in the case of speech acts performed by uttering imperatives, one must define the negation of imperatives. As I have argued elsewhere [13], a pair of affirmative and negative imperative comprises a complementary and symmetric imperative, e.g. negation of $!\,(\neg C/C)$ is $!\,(\neg C/\neg C)$, Example 11.

I will formalize imperatives as change expressions [5] having peculiar phenomenology in regards to their "direction of fit with the world"; the left part should fit the world while it is the world that should fit the right part:

$$\overline{!\left(\underbrace{\overbrace{\varphi}^{\text{word to world fit}}}_{\text{initial state}} \;/\; \underbrace{\overbrace{\psi}^{\text{world to word fit}}}_{\text{end state}}\right)}^{\text{commanded change}}.$$

2.1 Language L_{IMP}^{ACT}

2.1.1 Syntax

Definition 13 *Let language L_{PL} of classical propositional logic built over finite set D of propositional letters be given. If $\varphi \in L_{PL}$, then $\cdot\,(\varphi/\top)$ is indicative before-sentence in L_{IMP} and $\cdot\,(\top/\boxdot\,\varphi)$, $\cdot\,(\top/\diamond\,\varphi)$ are indicative after-sentences in L_{IMP}. If $\varphi \in L_{PL}$ and $\psi \in L_{PL}$, then $!\,(\varphi/\psi)$ and $!\,(\top/\varphi)$ are imperative sentences in L_{IMP}. If φ is indicative before-sentence in L_{IMP} and if ψ is imperative sentence on L_{IMP}, then $(\varphi \rightarrow \psi)$ and $(\psi \rightarrow \varphi)$ are conditional imperative sentences in L_{IMP}. Nothing else is a sentence in L_{IMP}.*

Definition 14 *If $\varphi \in L_{IMP}$, then $\varphi^{\,Up}$, $\varphi^{\,Down}$, $\varphi^{\,Test}$ are sentences in the language L_{IMP}^{ACT}.*

2.1.2 Semantics

Definition 15 *Set Σ of cognitive motivational states is the set constructed in the following way:*

- D is a finite set of propositional letters,
- $W = \wp D$ is a set of bare situations,
- $Moments = \{before, after\}$ is a set of moments,
- $Init = W \times \{before\}$ is a set of initial situations,
- $Res = W \times \{after\}$ is a set of resulting situations,
- $Changes = Init \times Res$ is a set of commanded changes,
- $\Sigma = Changes \times Res$ is set of cognitive-motivational states.

Definition 16 *For $\varphi \in L_{IMP}$, $|\tau|_X^t$ is set of φ bare situations in $X \subseteq W$ coupled with moment $t \in Moments$:*

$$|\varphi|_X^t = \{\langle w, t\rangle \mid w \in X \wedge t \in Moments \wedge h(\varphi, w) = \top\}.$$

Definition 17 *Intension $\|\varphi/\psi\|$ of a change expression (φ/ψ) is the set*

$$\|\varphi/\psi\| = |\varphi|_W^{before} \times |\psi|_W^{after}.$$

Definition 18 *For $\varphi \in L_{IMP}$, $|\varphi|_{\mu D^\rho}^t$ is set of φ bare situations in W minimally differing from bare situations in ρ sharing the same moment t:*

$$\begin{aligned} |\varphi|_{\mu D^\rho}^{before} &= {}^{mem_1(mem_1(\rho))}\downarrow\varphi\downarrow^{\mu D} \\ &= \left\{v \mid \exists w \left(w \in mem_1(mem_1(\rho)) \wedge \langle w, v\rangle \in \mu D^W(\varphi)\right)\right\}, \end{aligned}$$

and

$$\begin{aligned} |\varphi|_{\mu D^\rho}^{after} &= {}^{mem_1(mem_2(\rho))}\downarrow\varphi\downarrow^{\mu D} \\ &= \left\{v \mid \exists w \left(w \in mem_1(mem_2(\rho)) \wedge \langle w, v\rangle \in \mu D^W(\varphi)\right)\right\}. \end{aligned}$$

Definition 19 *Set Φ of absurd states:*

$$\Phi = \{\langle \rho, \pi\rangle \mid \rho = \emptyset \vee \neg mem_2(\rho) \subseteq \pi\}.$$

Definition 20 *For $\langle \rho_1, \pi_1\rangle \in \Sigma$ and $\langle \rho_2, \pi_2\rangle \in \Sigma$, operation $\Cup$ of merging structures is defined as: $\langle \rho_1, \pi_1\rangle \Cup \langle \rho_2, \pi_2\rangle = \langle \rho_1 \cup \rho_2, \pi_1 \cup \pi_2\rangle$.*

Definition 21 *Interpretation function $[\cdot]$ for the language L_{IMP}^{ACT} is function from $\Sigma \times L_{IMP}^{ACT}$ into Σ such that:*

- $$\begin{aligned} \langle \rho, \pi\rangle[\varphi_1] \ldots [\varphi_n] = \langle \rho, \pi\rangle[\varphi_1; \ldots; \varphi_n] = \\ (((\langle \rho, \pi\rangle[\varphi_1]) \ldots)[\varphi_{n-1}])[\varphi_n], \end{aligned}$$

  for $\varphi_1, \ldots, \varphi_n \in L_{IMP}^{ACT}$,

- $\langle \rho, \pi\rangle[!(\top/\varphi)^{Up}] = \begin{cases} \langle \rho \cap \|\top/\varphi\|, \pi\rangle \text{ if } |\varphi|_{mem_1(\pi)}^{after} \subset \pi, \\ 1 \text{ otherwise;} \end{cases}$

- $\langle \rho, \pi\rangle[\cdot(\varphi/\top)^{Up}] = \langle \rho \cap \|\varphi/\top\|, \pi\rangle$;

- $\langle \rho, \pi\rangle[(\top/\boxdot\varphi)^{Up}] = \langle \rho \cap \|\top/\varphi\|, \pi \cap |\varphi|_W^{after}\rangle$;

- $\langle \rho, \pi\rangle\left[(\cdot(\varphi/\top) \rightarrow !(\top/\psi))^{Up}\right] = \begin{cases} \langle \rho, \pi\rangle\left[!(\top/\psi)^{Up}\right] \\ \text{if } \langle \rho, \pi\rangle\left[\cdot(\varphi/\top)^{Up}\right] = \langle \rho, \pi\rangle, \\ \langle \rho, \pi\rangle\left[\cdot(\neg\varphi/\top)^{Up}\right] \\ \Cup \\ \langle \rho, \pi\rangle\left[\cdot(\varphi/\top)^{Up}\right]\left[!(\top/\psi)^{Up}\right] \\ \text{otherwise;} \end{cases}$

- $\langle \rho, \pi \rangle [!(\top/\varphi)^{Down}] = \begin{cases} \left\langle \begin{array}{l} \rho \cup \left(mem_1(\rho) \times |\neg\varphi|^{after}_{\mu D^\rho} \right), \\ \pi \cup |\varphi|^{after}_{\mu D^\rho} \cup |\neg\varphi|^{after}_{\mu D^\rho} \end{array} \right\rangle \\ \text{if } \langle \rho, \pi \rangle [!(\top/\neg\varphi)^{Up}] \in \Phi, \\ \langle \rho, \pi \rangle \text{ otherwise;} \end{cases}$

- $\langle \rho, \pi \rangle [\cdot(\varphi/\top)^{Down}] = \begin{cases} \langle \rho \cup \left(|\neg\varphi|^{before}_{\mu D^\rho} \times mem_2(\rho) \right), \pi \rangle \\ \text{if } \langle \rho, \pi \rangle [\cdot(\neg\varphi/\top)^{Up}] \in \Phi, \\ \langle \rho, \pi \rangle \text{ otherwise;} \end{cases}$

- $\langle \rho, \pi \rangle [\cdot\, (\top/ \boxdot\, \varphi)^{Down}] = \begin{cases} \langle \rho \cup \left(mem_1(\rho) \times |\neg\varphi|^{after}_{\mu D^\rho} \right), \pi \cup |\neg\varphi|^{after}_{\mu D^\rho} \rangle \\ \text{if } \langle \rho, \pi \rangle [\cdot\, (\top/ \boxdot\, \neg\varphi)^{Up}] \in \Phi, \\ \langle \rho, \pi \rangle \text{ otherwise;} \end{cases}$

- $\langle \rho, \pi \rangle \left[(\cdot(\varphi/\top) \to !(\top/\psi))^{Down} \right] = \begin{cases} \langle \rho, \pi \rangle \left[!\, (\top/\psi)^{Down} \right] \\ \text{if } \langle \rho, \pi \rangle \, [\cdot(\varphi/\top) \to !(\top/\neg\psi)]^{Up} \in \Phi, \\ \langle \rho, \pi \rangle \text{ otherwise;} \end{cases}$

- $\langle \rho, \pi \rangle [\varphi^{Test}] = \begin{cases} \langle \rho, \pi \rangle \text{ if } \langle \rho, \pi \rangle [(\varphi)^{Up}] \notin \Phi, \\ \langle \emptyset, \emptyset \rangle \text{ otherwise,} \end{cases}$ where $\varphi^{Test} \in L^{ACT}_{IMP}$;

- $\langle \rho, \pi \rangle [!\, (\varphi/\psi)^{Up}] = \langle \rho, \pi \rangle [\cdot\, (\varphi/\top)^{Up}] \left[!\, (\top/\psi)^{Up} \right]$;

- $\langle \rho, \pi \rangle [!\, (\varphi/\psi)^{Down}] = \langle \rho, \pi \rangle [\cdot\, (\varphi/\top)^{Down}] \left[!\, (\top/\psi)^{Down} \right]$;

- $\langle \rho, \pi \rangle [\cdot\, (\top/ \diamond\, \varphi)^{Up}] = \langle \rho, \pi \rangle [\cdot\, (\top/ \boxdot\, \varphi)^{Test}]$;

- $\langle \rho, \pi \rangle [\cdot\, (\top/ \diamond\, \varphi)^{Down}] = \langle \rho, \pi \rangle [\cdot\, (\top/ \boxdot\, \varphi)^{Down}]$;

- $\langle \rho, \pi \rangle \left[(!(\top/\varphi) \to \cdot(\psi/\top))^{Up} \right] = \langle \rho, \pi \rangle \left[(\cdot(\neg\psi/\top) \to !(\top/\neg\varphi))^{Up} \right]$;

- $\langle \rho, \pi \rangle \left[(!(\top/\varphi) \to \cdot(\psi/\top))^{Down} \right] = \langle \rho, \pi \rangle \left[(\cdot(\neg\psi/\top) \to !(\top/\neg\varphi))^{Down} \right]$.

Almost all of the proposed interpretations require explanation, which will be omitted here due to the limited space. I will briefly comment only on imperatives. Von Wright's "three points of action semantics" are built in the update semantics for the complementary and symmetric imperatives as commanded actions. Information on initial state is encoded into the set $mem_1(\rho)$, information on the end state is encoded into the set $mem_2(\rho)$, information on the counter state (which would have or could have resulted if the agent had refrained from performing commanded action) is encoded in the set π, which also encodes information on the possibility of end-state (π shows avoidability and possibility of the end-state).

Example 22 *Implications listed in Example 12 hold (for discussion of varieties of relations of meaning inclusion that can be distinguished within dynamic semantics see [2]). If* $\langle \rho, \pi \rangle \left[! \left(\neg C / C \right)^{Up} \right] = \langle \rho, \pi \rangle$*, then*

$$\langle \rho, \pi \rangle \left[\cdot \left(\neg C / \top \right)^{Up} ; ! \left(\top / C \right)^{Up} ; \cdot \left(\top / \diamond \neg C \right)^{Up} ; \cdot \left(\top / \diamond C \right)^{Up} \right] = \langle \rho, \pi \rangle \, .$$

Pragmatics within semantics and universality of logic The language L_{IMP}^{ACT} and interpretation function $[\cdot]$ provide an uncommon approach which drags the pragmatics into the syntax of the formal language and, consequently, it equates pragmatic effects with semantic actions. It is the speech act that gets a formal translation and not the sentence by whose utterance it is performed. If this approach is sound, then logic might claim universality of its scope over the language. Pragmatics might lie within the scope of logic.

Example 23 *Command 'Close the door!' is formalized as* $! \left(\neg C / C \right)^{Up}$*; permission 'You don't have to close the door.' ('You may leave the door open.') as* $! \left(\neg C / C \right)^{Down}$*; suggestion "Maybe you should close the door' as* $! \left(\neg C / C \right)^{Test}$*.*

Example 24 *The puzzle of distribution of permission over disjunction has been much discussed in the literature: (i) 'You may see to it that A or B' intuitively implies (ii) 'You may see to it that A' and (iii) 'You may see to it that B'. On the proposed approach (i) is translated as* $! \left(\top / \neg A \wedge \neg B \right)^{Down}$ *and interpreted as cancellation of (iv) 'See to it that both $\neg A$ and $\neg B$'. In the same vein, (ii) and (iii) are translated as* $! \left(\top / \neg A \right)^{Down}$ *and* $! \left(\top / \neg B \right)^{Down}$*, respectively. If* $\langle \rho, \pi \rangle \left[! \left(\top / \neg A \wedge \neg B \right)^{Up} \right] = \langle \rho, \pi \rangle$*, then*

$$\begin{aligned} \langle \rho, \pi \rangle \left[! \left(\top / \neg A \wedge \neg B \right)^{Down} \right] &= \\ = \langle \rho, \pi \rangle \left[! \left(\top / \neg A \wedge \neg B \right)^{Down} \right] \left[! \left(\top / \neg A \right)^{Down} \right] &= \\ = \langle \rho, \pi \rangle \left[! \left(\top / \neg A \wedge \neg B \right)^{Down} \right] \left[! \left(\top / \neg B \right)^{Down} \right] &. \end{aligned}$$

2.2 Expressive completeness

There are several interesting questions that arise at the interface between natural language and its logical formalization. In the natural language there are three kinds of imperatives (the complementary or produce imperatives, symmetric or sustain imperatives, and "one-sided" or "see to it that" imperatives). Therefore, the translation of natural language sentences will yield a proper subset of imperative sentences in L_{IMP}. Namely, we will find only $! \left(\neg \varphi / \varphi \right)$, $! \left(\varphi / \varphi \right)$ and $! \left(\top / \varphi \right)$ types of sentences in the subset. Is the subset strong enough to generate each non-absurd cognitive-motivational state? If not, is it so that there are some obstacles to communication that are inherent in the language itself? Further, do negated speech acts add expressive power to the language?

The answer to the first question is affirmative and it is negative to the second question within the framework of language L_{IMP}^{ACT} and its semantics. Theorem 25 shows that each non-absurd cognitive-motivational state $\sigma \in \Sigma - \Phi$ may be generated using a proper subset of language L_{IMP}^{ACT} in which only "non-negated speech acts" occur.

Theorem 25 *For each $\langle \rho, \pi \rangle \in \Sigma - \Phi$ there are $\varphi_1, ..., \varphi_n \in L_{IMP}$ such that $\langle Changes, Res \rangle \left[(\varphi_1)^{Up}; ...; (\varphi_n)^{Up} \right] = \langle \rho, \pi \rangle$.*

Proof. Proof is given by construction of the required text. Let

$$mem_1 (mem_1 (\rho)) = \{w_1, ..., w_n\} .$$

The construction may proceed in three steps. In the first step, the first members of ρ are cut out of $\langle Changes, Res \rangle$ using sentence

$$\cdot (nf (mem_1 (mem_1 (\rho))) / \top)^{Up}$$

and obtaining $\langle mem_1 (\rho) \times Res, Res \rangle$, Proposition 30. In the second step, a sequence of sentences $s (w_1)^{Up}; ...; s (w_n)^{Up}$ is applied to $\langle mem_1 (\rho) \times Res, Res \rangle$ (where each sentence $s (w_i)$ is either a conditional imperative or a tautology) yielding:

$$\langle mem_1 (\rho) \times Res, Res \rangle \left[s (w_1)^{Up} \right] ... \left[s (w_n)^{Up} \right] = \langle \rho, Res \rangle ,$$

Proposition 33. In the third step, application of $\cdot (\top / \boxdot\, nf(mem_1(\pi)))^{Up}$ gives the desired result: $\langle \rho, Res \rangle \left[\cdot (\top / \boxdot\, nf(mem_1(\pi)))^{Up} \right] = \langle \rho, \pi \rangle$, Proposition 34. The text

$$\cdot (nf((mem_1(mem_1(\rho)))/\top)^{Up}; s(w_1)^{Up}; ...; s(w_n)^{Up}; \cdot (\top / \boxdot\, nf(mem_1(\pi)))^{Up}$$

is an instance that proves that each non-absurd state may be generated by text of L_{IMP}^{ACT}. ■

Corollary 26 *Let $L_{\to stit}^{ACT} \subset L_{IMP}^{ACT}$ be a language comprising only sentences of the form: $\cdot (\varphi / \top)^{Up}$, $(\cdot (\varphi / \top) \to ! (\top / \psi))^{Up}$, $\cdot (\top / \boxdot\, \varphi)^{Up}$. Language $L_{\to stit}^{ACT}$ is expressively complete with respect to set $\Sigma - \Phi$ of non-absurd states.*

Definition 27 (Literals λ) *Given $l_1, ..., l_n$ list of all propositional letters in D, $w_1, ..., w_m$ list of all situations in $S \subseteq W$, $\wp D = W$, $1 \leq i \leq n$, $1 \leq j \leq m$, literals λ_j^i are defined by:*

$$\lambda_{l_i}^{w_j} = \begin{cases} l_i \text{ if } l_i \in w_j , \\ \neg l_i \text{ if } l_i \notin w_j . \end{cases}$$

Definition 28 (Adequate description) *Function nf delivers a disjunctive normal form for the set S with respect to given lists of letters $l_1, ..., l_n$ and situations $w_1, ..., w_m$:*

$$\begin{aligned} nf(S) &= \left(\left(\lambda_{l_1}^{w_1} \wedge ... \wedge \lambda_{l_n}^{w_1}\right) \vee ... \vee \left(\lambda_{l_1}^{w_m} \wedge ... \wedge \lambda_{l_n}^{w_m}\right)\right) \\ &= nf\left(w_1\right) \vee ... \vee nf\left(w_m\right). \end{aligned}$$

Proposition 29 *For $S \subseteq W$, $\left|nf\left(S\right)\right|_W^t = S \times \{t\}$.*

Proof. The proof is straightforward and only right to left direction will be shown. Suppose for some arbitrary v that $\langle v, t\rangle \in S \times \{t\}$. Obviously,

$$h\left(nf\left(v\right), v\right) = \top.$$

By Definition 3, $h\left(nf\left(S\right), v\right) = \top$. By Definition 16, $\langle v, t\rangle \in \left|nf\left(S\right)\right|_W^t$. ■

Proposition 30

$$\langle Changes, Res\rangle \left[\cdot\left(nf\left(mem_1\left(mem_1\left(\rho\right)\right)\right)/\top\right)^{Up}\right] = \langle mem_1\left(\rho\right) \times Res, Res\rangle$$

Proof. By Definition 17,

$$\left\|nf\left(mem_1\left(mem_1\left(\rho\right)\right)\right)/\top\right\| = \left|nf\left(mem_1\left(mem_1\left(\rho\right)\right)\right)\right|_W^{before} \times \left|\top\right|_W^{after}.$$

The fact that $\left|\top\right|_W^{after} = Res$ together with an application of Proposition 29, i.e.

$$\left|nf\left(mem_1\left(mem_1\left(\rho\right)\right)\right)\right|_W^{before} = mem_1\left(\rho\right),$$

give the desired result. ■

Definition 31 *Function $ex_\rho^{\langle w,before\rangle}$ delivers set of resulting situations "visible" from situation $\langle w, before\rangle$:*

$$ex_\rho^{\langle w,before\rangle} = mem_2\left(\left(\left\{\langle w, before\rangle\right\} \times Res\right) \cap \rho\right).$$

Definition 32 *Let $\rho \subseteq \pi$. For each situation $\langle w, before\rangle \in mem_1\left(\rho\right)$, function s delivers a sentence from L_{IMP}:*

$$s(w) = \begin{cases} \cdot\left(\top/\top\right) \ \text{if } ex_\rho^{\langle w,before\rangle} = \pi, \\ \left(\cdot\left(nf\left(w\right)/\top\right) \to! \left(\top/nf\left(mem_1\left(ex_\rho^{\langle w,before\rangle}\right)\right)\right)\right) \ \text{otherwise.} \end{cases}$$

Proposition 33 *Let $\{w_1, ..., w_n\} = mem_1\left(\rho\right)$. Then*

$$\langle Changes, Res\rangle \left[\cdot\left(nf\left(mem_1\left(mem_1\left(\rho\right)\right)\right)/\top\right)^{Up}; s(w_1)^{Up}; ...; s\left(w_n\right)^{Up}\right] = \langle \rho, Res\rangle.$$

Proof. By Proposition 30,

$$\langle Changes, Res\rangle \left[\cdot \left(nf\left(mem_1\left(mem_1\left(\rho\right)\right)\right)/\top\right)^{Up}; s(w_1)^{Up}; ...; s\left(w_n\right)^{Up}\right]$$
$$= \langle mem_1\left(\rho\right) \times Res, Res\rangle \left[s(w_1)^{Up}; ...; s\left(w_n\right)^{Up}\right].$$

There are two cases to examine.

First, for $n = 1$ let $mem_1\left(\rho\right) = \{\langle w, before\rangle\}$. Therefore,

$$\rho = \{\langle w, before\rangle\} \times mem_2\left(\rho\right).$$

There are two subcases. If $ex_\rho^{\langle w, before\rangle} = Res$, then $s\left(w_1\right) = \cdot\left(\top/\top\right)$ and obviously

$$\langle mem_1\left(\rho\right) \times Res, Res\rangle \left[\cdot\left(\top/\top\right)^{Up}\right] = \langle\rho, Res\rangle.$$

In the second subcase, $ex_\rho^{\langle w, before\rangle} \subset Res$. Then

$$s\left(w\right) = \left(\cdot\left(nf\left(w\right)/\top\right) \to! \left(\top/nf\left(mem_1\left(ex_\rho^{\langle w, before\rangle}\right)\right)\right)\right).$$

Since $\left|nf\left(w\right)\right|_W^{before} = \{\langle w, before\rangle\} = mem_1\left(\rho\right)$, the conditional has the following impact:

$$\sigma \left[\left(\cdot\left(nf\left(w\right)/\top\right) \to! \left(\top/nf\left(mem_1\left(ex_\rho^{\langle w, before\rangle}\right)\right)\right)\right)^{Up}\right] =$$
$$= \sigma \left[!\left(\top/nf\left(mem_1\left(ex_\rho^{\langle w, before\rangle}\right)\right)\right)^{Up}\right],$$

where $\sigma = \langle\{\langle w, before\rangle\} \times Res, Res\rangle$. Since

$$\left|nf\left(mem_1\left(ex_\rho^{\langle w, before\rangle}\right)\right)\right|_W^{after} = mem_2\left(\rho\right),$$

we get the required result.

For the second case, when $n > 1$ we have to show that semantic impact (if any) of $s\left(w_i\right)$ is localized to $\langle w_i, before\rangle$ generating $\{\langle w_i, before\rangle\} \times ex_\rho^{\langle w_i, before\rangle}$ and leaving everything else as it is. In other words, we have to show that for each $w_i \in mem_1\left(\rho\right)$,

$$\langle mem_1\left(\rho\right) \times Res, Res\rangle \left[s(w_i)^{Up}\right] =$$
$$= \left\langle \begin{array}{l} \left(mem_1\left(\rho\right) - \{\langle w_i, before\rangle\}\right) \times Res \\ \cup \\ \{\langle w_i, before\rangle\} \times ex_\rho^{\langle w_i, before\rangle} \end{array}, Res \right\rangle.$$

There are two cases to examine. First, if $ex_\rho^{\langle w_i, before\rangle} = Res$, then $s\left(w_i\right) = \cdot\left(\top/\top\right)$ and

$$\langle mem_1\left(\rho\right) \times Res, Res\rangle \left[\cdot\left(\top/\top\right)^{Up}\right] = \langle mem_1\left(\rho\right) \times Res, Res\rangle.$$

In the second subcase: $ex_{\rho}^{\langle w_i, before\rangle} \subset Res$. The fact that $n > 1$ guarantees

$$\langle mem_1(\rho) \times Res, Res\rangle \left[\cdot (nf(w_i)/\top)^{Up}\right] \neq \langle mem_1(\rho) \times Res, Res\rangle$$

since $|nf(w_i)|_W^{before} \neq mem_1(\rho)$. Let σ stand for $\langle mem_1(\rho) \times Res, Res\rangle$. Then the conditional

$$s(w_i) = \left(\cdot (nf(w_i)/\top) \rightarrow ! \left(\top / nf\left(mem_1\left(ex_{\rho}^{\langle w_i, before\rangle}\right)\right)\right)\right)$$

has the following impact:

$$\sigma\left[\left(\cdot (nf(w_i)/\top) \rightarrow ! \left(\top / nf\left(mem_1\left(ex_{\rho}^{\langle w_i, before\rangle}\right)\right)\right)\right)^{Up}\right] =$$

$$= \begin{cases} \sigma\left[\cdot(\neg nf(w_i)/\top)^{Up}\right] \\ \Cup \\ \sigma\left[\cdot(nf(w_i)/\top)^{Up}\right]\left[!\left(\top / nf\left(mem_1\left(ex_{\rho}^{\langle w_i, before\rangle}\right)\right)\right)^{Up}\right] \end{cases}$$

$$= \begin{cases} \langle (mem_1(\rho) - \{\langle w_i, before\rangle\}) \times Res, Res\rangle \\ \Cup \\ \left\langle \{\langle w_i, before\rangle\} \times ex_{\rho}^{\langle w_i, before\rangle}, Res\right\rangle \end{cases}$$

$$= \left\langle \begin{array}{l} (mem_1(\rho) - \{\langle w_i, before\rangle\}) \times Res \\ \cup \\ \{\langle w_i, before\rangle\} \times ex_{\rho}^{\langle w_i, before\rangle} \end{array}, Res \right\rangle.$$

The sequence $s(w_1)^{Up}; ...; s(w_n)^{Up}$ of *up*-functions generates the desired:

$$\left\langle \bigcap_{i \leq n} \left(\begin{array}{l} ((mem_1(\rho) - \{\langle w_i, before\rangle\}) \times Res) \\ \cup \\ \left(\{\langle w_i, before\rangle\} \times ex_{\rho}^{\langle w_i, before\rangle}\right) \end{array} \right), Res \right\rangle =$$

$$= \langle \rho, Res\rangle.$$

■

Proposition 34 $\langle \rho, Res\rangle \left[\cdot (\top / \boxdot\, nf(mem_1(\pi)))^{Up}\right] = \langle \rho, \pi\rangle$

Proof. Routine. ■

References

[1] Belnap, Nuel and Perloff, Michael (1988). Seeing to it that: a canonical form of agentives. *Theoria* **54**: 175–199

[2] Benthem, Johan van (1993). *Exploring Logical Dynamics.* CSLI Publications, Stanford

[3] Fox, John (1990) Motivation and demotivation of a four-valued logic. *Notre Dame Journal of Formal Logic* **31**: 76–80

[4] Kanger, Stig (1972). Law and logic. *Theoria* **38**: 105–129

[5] Lemmon, Edward (1965). Deontic logic and the logic of imperatives. *Logique et Analyse* **8**: 39-71.

[6] de Rijke, Maarten (1998). A system of dynamic modal logic. *Journal of Philosophical Logic* **27**: 109–142

[7] Ross, Alf (1941). Imperatives and logic. *Theoria* **7**: 53–71

[8] Segerberg, Krister (1990). Validity and satisfaction in imperative logic. *Notre Dame Journal of Formal Logic* **31**: 203–221

[9] Stenius, Erik (1967). Mood and language game. Synthese, 19: 27-52.

[10] Tappenden, Jamie (1999). Negation, denial and language change in philosophical logic. In: D. Gabbay and H. Wansing, (eds.) *What is Negation?* pp. 261-298, Kluwer

[11] Veltman, Frank (1996). Defaults in update semantics. *Journal of Philosophical Logic* **25**: 221–261

[12] Wright, Georg Henrik von (1963). *Norm and Action: A Logical Inquiry.* Routledge and Kegan Paul

[13] Zarnic, Berislav (2003). Imperative negation and dynamic semantics. In: *Meaning: the Dynamic Turn.* J. Peregrin (editor). pp. 201–211, Elsevier