# Imperatives in discourse

## 1 CROSS-LINGUISTIC CHARACTERISTICS

(**SO**) SUBJECT (*EXP) & OBJECT (variable scope)

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| (1P) | a. | *An*i | *byl*o | *wesol*o. | | | | (P = Polish) |
| | | Ann.*DAT* | was.*N* | cheerful*ly* | | | | |
| | | Ann (EXP) was (= felt) cheerful. | | | | | | |
| | b. | *An*ia | *byl*a | *wesol*a | (*choc* | *byl*o | jej | *smutn*o). |
| | | Ann.*NOM* | was.*F* | cheerful.*F* | (though | was.*N* | her.*DAT* | sad*ly*) |
| | | Ann was being (= acted) cheerful (though she felt sad). | | | | | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| (2P) | a.* | Badz | *wesol*o! | /✓*Niech* | ci | *bedzie* | *wesol*o! |
| | | be.*IMP*.2SG | cheerful*ly* | / OPT | you.*DAT* | be.FUT.3SN | cheerful*ly* |
| | | | | Have a good time! (*lit.* May you…) | | | |
| | b. | Badz | *wesol*a! | | | | |
| | | be.*IMP*.2SG | cheerful.*F* | | | | |
| | | Be (= act) cheerful! | | | | | |

(3E) a. [Find] my assistant! — O > IMP
b. [Find me an assistant]! — IMP > O

| | | | | |
|---|---|---|---|---|
| (3K) | a. | *Ikiurtiga* | *naniguk*! | (K = Kalaallisut) |
| | | *ikiur-ti-ga* | *nani-gi-*uk | O > IMP |
| | | help-tv\agt-1SG.SG | find-IMP-2SG.*3SG* | |
| | b. | *Ikiur-ti-*ssaq-*n*nik | *nani-*ssi-*gi-*t! | IMP > O |
| | | help-tv\agt-*exp/des*-1SG.SG.*MOD* | find-*antip*-IMP-2SG | |

(**I**) INDIRECT DIRECTIVES — (a) |= (b)

(4E) a. Ann said to Ole: "*Wash* up!"
b. Ann told Ole *to* wash up.

| | | | | | |
|---|---|---|---|---|---|
| (4K) | a. | *Aanip* | *Ole* | *uqarvigaa*: | "*Irruigit*!" |
| | | *Aani-p* | *Ole* | *uqar-vigi-pa-a*: | "*irrui-*gi-*t*" |
| | | A.-ERG | Ole | say-iv\tv-DEC.TV-3SG.3SG | "wash.up-*IMP*-2SG" |
| | b. | *Aanip* | *Ole* | *uqarvigaa* | *irruiqqullugu*. |
| | | *Aani-p* | *Ole* | *uqar-vigi-pa-a* | *irrui-*qqu-*llu-gu* |
| | | A.-ERG | Ole | say-iv\tv-DEC.TV-3SG.3SG | wash.up-*tell*-ELA$_T$-3SG$_\perp$ |
| | b′. | *Aanip* | *Ole* | *irruiqquaa*. | (b) = (b′) |
| | | *Aani-p* | *Ole* | *irrui-*qqu-*pa-a* | |
| | | A.-ERG | Ole | wash.up-*tell*-DEC.TV-3SG.3SG | |

(¬) NEGATIVE DIRECTIVES (a) |= (b)

(5E) a. Ann said to me: "Do*n't* wash up!"
b. Ann told me *not* to wash up.

(5K) a. *Aanip* *uqarvigaanga*: "*Irruiqinak*!"
*Aani-p* *uqar-vigi-pa-anga*: "*irrui*-qina-*k*"
A.-ERG say-iv\tv-DEC.TV-3SG.1SG "wash.up-$IMP_{\notin}$-2SG"
b. *Aanip* *uqarvigaanga* *irruiqqunanga*.
*Aani-p* *uqar-vigi-pa-anga* *irrui-qqu*-na-*nga*
A.-ERG say-iv\tv-DEC.TV-3SG.1SG wash.up-tell-$NON_{T}$-1SG
b′. *Aanip* *irruiqqunngilaanga*
*Aani-p* *irrui-qqu*-nngit-la-*anga*
A.-ERG wash.up-tell-*not*-$DEC_{\notin}$-3SG.1SG
i. Ann told me not to wash up. = (b)
ii. Ann didn't tell me to wash up.

(→) CONDITIONAL/ANTECEDENT DIRECTIVES

(6E) *If* my son shows up, *help* him!

(6K) *Irnira* *takkuppat*, *ikiuruk*!
*irniq-ga* *takkug*-pp-*at* *ikiur*-gi-*uk*!
son-1SG.SG show.up-$HYP_{\perp}$-3$SG_{\perp}$ help-*IMP*-2SG.3SG

(7P) a. Jezeli *podejdziesz* to *ci* *to* *dam.*
if come.closer.NPST.2SG *then* you.D this.A give.NPST.1SG
If you come closer, then I'll give it to you.
b. *Podej*dz to *ci* *to* *dam.*
come.closer.*IMP*.2SG *then* you.D this.A give.NPST.1SG
Come closer; if you do, then I'll give it to you.

[*Harry Potter*: Prof. McGonagal to Hagrid, who's sobbing loudly when orphaned baby Harry is about to be left with his Muggle relatives.]

(8E) a. *Get* a grip on yourself *or* we'll be found!
b. *Leave* your bike here *and* it'll be stolen!
c.*We'll be found *or/and* get a grip on yourself!

(8K) a. *Iqqissigit*! *Taamaaliunngikkuit*
*iqqissi*-gi-*t*! *taama-iliur-nngit*-gu-*vit*
calm.down-*IMP*-2SG thus-do-not-$HYP_{T}$-2SG
*takuniqassaagut*
*taku-niqar-ssa-pu-gut*
see-passive-exp/des-DEC.IV-1PL
Calm down! If you don't, we'll be seen. (*lit*. [I] expect us to …)

(∧) 'AND': ASSERTION ~ DIRECTIVE

• assertion (A = Mom, B = Dad) (B) |= (A)

(9E) A: Today Ole set the table
B: Right, he set the table *and* washed up.

(9K) A: *Ole ullumi nirriviliurpuq.*
*Ole ulluq-mi nirrivik-liur-pu-q*
Ole day-LOC table-make-DEC.IV-3SG
B. *Suu, nirriviliurlunilu irruisimavuq.*
*suu nirrivik-liur-llu-ni*=lu *irrui-sima-pu-q.*
right, table-make-ELA$_T$-3SG$_T$=*and* wash.up-prf-DEC.IV-3SG

• directive (A = Mom, B = Dad, O = Ole, C = Grandma) (B) |≈ (A)

(10E) A: Ole, set the table!
B: Right, set the table *and* wash up!
O to C: I'm told to set the table *and* wash up!

(10K) A: *Ole, nirriviliurit*!
*Ole, nirrivik-liur*-gi-*t*
Ole, table-make-*IMP*-2SG
B. *Suu, nirriviliurlutillu irruissaatit*!
*suu, nirrivik-liur-llu-tit*=lu *irrui-ssa-pu-tit*
right, table-make-ELA$_T$-2SG=*and* wash.up-exp/des-DEC.IV-2SG
O to C: *Nirriviliurlungalu irruissaangaguuq*!
*nirrivik-liur-llu-nga*=lu *irrui-ssa-pu-nga=guuq*
table-make-ELA$_T$-1SG=*and* wash.up-exp/des-DEC.IV-1SG=RPT

(∨) 'OR': ASSERTION ≠ DIRECTIVE

• assertion (A = Mom, B = Dad) (B) |= (A)

[Every day two children help at dinner—one sets the table, the other one washes up. B knows Ole was on the team today, but not what his task was.]

(11E) B: Today Ole must've set the table or washed up.
A: Right, (he didn't wash up but) he set the table.

(11K) B: *Ole nirriviliurunarpuq irruigunarpurluunniit.*
*Ole nirrivik-liur-gunar-pu-q irrui-gunar-pu-q=luunniit*
Ole table-make-likely-DEC.IV-3SG wash.up-likely-DEC.IV-3SG=or
A: *Suu*, (*irruinngikkaluarluni*) *nirriviliursimavuq.*
*suu*, (*irrui-nngit-galuar-llu-ni*) *nirrivik-liur-sima-pu-q*
right, (wash.up-not-rem-ELA$_T$-3SG$_T$) table-make-prf-DEC.IV-3SG

• directive (A = Mom, B = Dad, O = Ole) (B) |≠ (A)

(12E) A: Ole, set the table or wash up!
B: ? Right, wash up!
O: No, Mom didn't say that. If I want to, she said, I can just set the table. I am not going to wash up.

(12K) A: *Ole*, *nirriviliurlutilluunniit* *irruigit*!
*Ole*, *nirrivik-liur-llu-tit=luunniit* *irrui-gi-t*
Ole, table-make-ELA$_{T}$-2SG=or wash.up-IMP-2SG
B: ? *Suu*, *irruigit*!
*suu*, *irrui-gi-t*
right, wash.up-IMP-2SG
O: *Naamik*, *anaana* *taama* *uqanngilaq.*
*naamik* *anaana* *taama* *uqar-nngit-la-q*
no, mom thus say-not-DEC$_{\notin}$-3SG
*Piumagumaguuq* *nirriviliuinnarsinnaavunga.*
*pi-juma-gu-ma=guuq* *nirrivik-liur-innar-sinnaa-pu-nga*
v-want-HYP$_{T}$-1SG=RPT table-make-just-possible-DEC.IV-1SG
*Irruinianngilanga*.
*irrui-niar-nngit-la-nga*
wash.up-intend-not-DEC$_{\notin}$-1SG

cf. Polish, intensional verbs (*look for*, *need*, *want*, …), and questions:

**(**13P**)** A. *Nakryj* *do* *stolu* albo *zmyj* *naczynia*!
set.IMP.2SG to table.G *or*/*OR wash.up.IMP.2SG dishes.A
B. #*Tak jest,* *nakryj* *do* *stolu*!
right, set.IMP.2SG to table

**(**14P**)** A. *Ania* poszukuje *kucharki* albo *sprzataczki*,
Ann look.for.NPST.3SG cook.G *or* maid.G
*wszystko* *jej* *jedno*.
all her.D one
Ann is looking for a maid or a cook, she doesn't care which.
B. #*Tak jest, ona* *poszukuje* *kucharki*.
right, she look.for.NPST cook.G
?Right, she's looking for a cook.

**(**15P**)** A. *Ania* *poszukuje* *kucharki* czy tez *sprzataczki*.
Ann look.for.NPST.3SG cook.G OR *else* maid.G
Ann is looking for a cook or a maid (I don't know which)
B. *Poszukuje* *kucharki*.
look.for.NPST cook.G

**(16P)** A. Czy *poszukujesz* *kucharki* albo *sprzataczki*?
OR look.for.NPST.2SG cook.G *or* maid.G
Are you looking for a cook or a maid?
B. ✓*Tak* (, *poszukuje*). | *Nie*, (*nie*=*poszukuje*)
yes (, look.for.NPST.1SG) | no (not=look.for.NPST.1SG)
Yes (, I am) | No (. I am not)

**(17P)** A. Czy *poszukujesz* *kucharki* czy *sprzataczki*?
OR look.for.NPST.2SG cook.G OR maid.G
Are you looking for a cook or a maid?
B. ✓(*Poszukuje*) *kucharki*.
(look.for.NPST.1SG) cook.G
(I am looking for) a cook.
✓(*Poszukuje*) *sprzataczki*.
(look.for.NPST.1SG) maid.G
(I am looking for) a maid.
B′. #*Tak* (, *poszukuje*). | # *Nie*, (*nie*=*poszukuje*)
yes (, look.for.NPST.1SG) | no (not=look.for.NPST.1SG)
Yes (, I am) | No (. I am not)

(18) a. Post the letter! ***Ross' paradox***
b. Post the letter *or* burn it!

*Pippi Longstocking* (Ch. 8 'Pippi entertains two burglars', simplified)
(19) [There was a loud knock on the door.]
a. "*Isirit*!" K
*isir-gi-t*
come.in-IMP-2SG
"Come in!" (conventional response) E
b. "*Isirit* *imaluunniit* *isiqinak*!, K
*isir-gi-t* imaluunniit *isir-qina-k*
come.in-IMP-2SG OR come.in-IMP$_{\notin}$-2SG
*illit* *pi-juma-saq-n-nik*!"
2SG v-want-tv\rn-2SG.SG-MOD

"*Wejdz* albo *nie*(=*wchodz*), P
come.in.IMP.2SG *or* not(=come.in.IMP.2SG)
*jak* *wolisz.*
how/as prefer.NPST.2SG
"Come in *or* don't, as you wish!" (said Pippi) E

## (**D**) DISCOURSE ANAPHORA

(20) Context (myth): A man, carried off by breaking ice and adopted by a bear, is homesick. One day the bear jumps into the sea and calls back:

i. *Angirlarsiruit*
*angirlar-sir-gu-vit*
go.home-long.to-HYP$_{\top}$-2SG

| | | |
|---|---|---|
| *malillunga* | *immamut* | *pissigit*! |
| *malig-llu-nga* | *imaq-mut* | pissig-gi-*t* |
| follow-ELA$_{\top}$-1SG | sea-SG.DAT | *jump-IMP*-2SG |

If you long to go home, *jump* into the sea after me!

ii.

| | |
|---|---|
| *Pissinngikkuit* | *angirlarnavianngilatit.* |
| pissig-*nngit-gu-vit* | *angirlar-navianngit-la-tit* |
| *jump*-not-HYP$_{\top}$-2SG | go.home-certain.not.to-DEC$_{\notin}$-2SG |

If you don't *jump*, you certainly won't go home.

On contact with the sea the man becomes a seagull & is now told:

| | | |
|---|---|---|
| iii. *Siqungirit!* | / | iii′. *Uiqinak*! |
| siqungir-gi-*t* | / | uit-qina-*k* |
| *shut.eyes-IMP*-2SG | / | *open.eyes-IMP*$_{\notin}$-2SG |
| *Shut* your eyes! | / | Don't *open* your eyes! |

iv.

| | |
|---|---|
| *Uikkaluaruit* | *angirlarnavianngilatit.* |
| uit-galuar-gu-*vit* | *angirlar*-navianngit-*la-tit.* |
| *open.eyes*-rem-HYP$_{\top}$-2SG | go.home-certain.not.to-DEC$_{\notin}$-2SG |

If you *open* your eyes, you certainly won't go home.

(21) i.

| | | | |
|---|---|---|---|
| *Ilaanni* | *Ole* | *qasigissamik* | *puisisimavuq.* |
| *ilaanni* | *Ole* | qasigiaq-mik | puisi-*g-sima-pu-q* |
| once | Ole | *harbour.seal-MOD* | *seal*-catch-prf-DEC.IV-3SG |

Once Ole caught *a harbour seal*.

ii.

| | | |
|---|---|---|
| *Tikikkami* | *nulii* | *uqarvigaa*: |
| *tikit-ga-mi* | *nuli-i* | *uqar-vigi-pa-a* |
| come-FCT$_{\top}$-3SG$_{\top}$ | wife-3SG$_{\top}$.SG | say-to-DEC.TV-3SG.3SG |

When he came home, he said to his wife:

iii.

| | | |
|---|---|---|
| "*Amia* | *panirsimappat* | *tuniguk*!" |
| *amiq*-a | *panir-sima-pp-at* | tuni-gi-*uk* |
| skin-*3SG*$_{\perp}$.SG | dry-prf-HYP$_{\perp}$-3SG$_{\perp}$ | *sell-IMP*-2SG.3SG |

"When *the* skin has dried *sell* it!"

iv.

| | |
|---|---|
| *Nulia* | *taamaaliurpuq.* |
| *nuli-a* | taama=*iliur*-pu-*q* |
| wife-3SG$_{\perp}$.SG | *thus*=do-DEC.IV-3SG |

His wife did *so/that*.

## 2 IMPERATIVES AS PLAN UPDATES

• BASIC IDEA (Mastop 2005)
Speech affects addressee's information and action plan—specifically, a declarative updates his *information*, whereas an imperative updates his *plan*.

• CH. 4: UPDATE WITH PLANS[1] ($UP_1$, atemporal)

**Definition 4.3.** (Syntax)

i. $UP_0 = P \cup A$,
   where $P = \{set, wsh, opn, ent, \dots\}$ is a set of atomic declaratives
   $A = \{set!, wsh!, opn!, ent!, \dots\}$ is a set of atomic imperatives

ii. $UP_0 \subseteq UP_1$
   $\neg\phi \in UP_1$ if $\phi \in UP_1$
   $\phi \wedge \psi, \phi \vee \psi \in UP_1$ if $\phi, \psi \in UP_1$

iii. $\phi \rightarrow \psi := \neg\phi \vee (\phi \wedge \psi)$ if $\phi$ contains only declarative atoms

| e.g. | English/Kalaallisut | $UP_1$ |
|---|---|---|
| (22) | A: Ole set the table. | *set* |
| | B: Right, he set the table and washed up. | $set \wedge wsh$ |
| | C: #True, he didn't wash up. | $\neg wsh$ |
| (23) | A: Ole set the table or washed up. | $set \vee wsh$ |
| | B: Right, he set the table. | *set* |
| | C: True, he didn't wash up. | $\neg wsh$ |
| | C′: Indeed, he also washed up. | *wsh* |
| (24) | A: Set the table! | *set*! |
| | B: Right, set the table and wash up! | $set! \wedge wsh!$ |
| (25) | A: Set the table or wash up! | $set! \vee wsh!$ |
| | B: ?Right, set the table! | *set*! |
| (26) i. | Ole either has or has not washed up. | $wsh \vee \neg wsh$ |
| ii. | If he has, pay him! | $wsh \rightarrow pay!$ |
| iii. | if he hasn't, don't! | $\neg wsh \rightarrow \neg pay!$ |
| (27) i. | Don't open your eyes! | $\neg opn!$ |
| ii. | If you open your eyes you won't go home! | $opn \rightarrow \neg ghm$ |

**P1** (puzzle): $p \rightarrow a := \neg p \vee (p \wedge a)$, but in every lg. *$(p \wedge a)$, e.g. *(8Ec)

[1] This is my name for Mastop's system, to facilitate comparison with *UC*.

**Definition 4.4** (situations & plans)

- $s \subseteq P \times \{1, 0\}$ is a *situation*
  $S := Pow(P \times \{0, 1\})$ is the set of situations
- $\pi \subseteq A \times \{\text{Y}, \text{N}\}$ is a *plan*
  $\Pi := Pow(A \times \{\text{Y}, \text{N}\})$ is the set of plans.

**e.g.**

- $s_1 = \{\langle set, 1\rangle, \langle wsh, 0\rangle\}$
  is a situation where *set* is true, and *wsh*, false
- $\pi_1 = \{\langle set!, \text{Y}\rangle, \langle wsh!, \text{N}\rangle\}$
  is a plan to do *set*! and not do *wsh*!
- $\varnothing$
  is the maximally unspecified situation or plan

**Definition 4.6** (possibilities & states)

- $(s, \pi) \in S \times \Pi$ is a *possibility*
- $\sigma \subseteq S \times \Pi$ is a (*cognitive*) *state*
  $\Sigma := Pow(S \times \Pi)$ is the set of (cognitive) states

**e.g.**

- $\sigma_1 = \{(\{\langle wsh, 1\rangle\}, \{\langle pay!, \text{Y}\rangle\}),$
  $(\{\langle wsh, 0\rangle\}, \{\langle pay!, \text{Y}\rangle\})\}$
  exp. of $\sigma_1$ doesn't know whether *wsh* is true or false, plans to do *pay*! in either case
- $\sigma_2 = \{(\{\langle wsh, 1\rangle\}, \{\langle pay!, \text{Y}\rangle\}),$
  $(\{\langle wsh, 1\rangle\}, \{\langle pay!, \text{N}\rangle\})\}$
  exp. of $\sigma_2$ knows that *wsh* is true, undecided whether or not to do *pay*!
- $\sigma_3 = \{(\{\langle wsh, 1\rangle\}, \{\langle pay!, \text{Y}\rangle\}),$
  $(\{\langle wsh, 0\rangle\}, \{\langle pay!, \text{N}\rangle\})\}$
  exp of $\sigma_3$ doesn't know whether *wsh* is true or false, plans to do *pay*! if *wsh* is true, not do *pay*! if *wsh* is false
- $\{(\varnothing, \varnothing)\}$ — maximally unspecified state; $\varnothing$ — absurd state (update ends)

**Definition 4.7** (consistency & quandry)

- $s$ is *consistent*, $\checkmark s$, iff $\neg\exists p \in P$: $\langle p, 1\rangle \in s \wedge \langle p, 0\rangle \in s$
  $\pi$ is *consistent*, $\checkmark \pi$, iff $\neg\exists a \in A$: $\langle a, \text{Y}\rangle \in \pi \wedge \langle a, \text{N}\rangle \in \pi$
- A possibility $(s, \pi)$ is a *quandry* iff $s$ or $\pi$ is not consistent
  A state $\sigma$ is *quandry free* iff there is no quandry in $\sigma$

**e.g.**

(28) Poland during WWII. If a Jew is found, he's shot. If a Pole hides a Jew in his home, then the entire Polish family is shot too. Consider a Polish person whose conscience dictates (i) and (ii):

i. If you see a child in danger, help it. *see* → *hlp*!
ii. If you love your family, don't help Jews. *lov* → ¬*hlp*!
iii. So, if you love your family and see a Jewish child in danger, ??…

- $\sigma_4 = \{(\{\langle lov, 1\rangle, \langle see, 0\rangle\}, \{\langle hlp!, \text{N}\rangle\}),$
  $(\{\langle lov, 1\rangle, \langle see, 1\rangle\}, \{\boldsymbol{\langle hlp!, \text{Y}\rangle}, \boldsymbol{\langle hlp!, \text{N}\rangle}\}), \ldots\}$
  exp of $\sigma_4$ knows that *lov* is true, doesn't know whether *see* is true or false, plans not to do *hlp*! if *see* is false, is in a quandry if *see* is true.
- $\sigma_5 = \{(\{\boldsymbol{\langle see, 0\rangle}, \boldsymbol{\langle see, 1\rangle}, \ldots\}, \ldots), \ldots\}$
  exp of $\sigma_5$ is in a quandry whether *see* is true or false

**Definition 4.5** (expansions). For any $p \in P$, $a \in A$:

- $s \oplus p \;=\; s \cup \{\langle p, 1\rangle\}$
  $s \ominus p \;=\; s \cup \{\langle p, 0\rangle\}$
- $\pi \oplus a \;=\; \pi \cup \{\langle a, \text{Y}\rangle\}$
  $\pi \ominus a \;=\; \pi \cup \{\langle a, \text{N}\rangle\}$

**Definition 4.8** (positive update ↑, negative update ↓). If quandry free,

***p.*** $\sigma \uparrow p \;=\; \{(s \oplus p, \pi) | (s, \pi) \in \sigma \;\&\; \checkmark(s \oplus p)\}$
$\sigma \downarrow p \;=\; \{(s \ominus p, \pi) | (s, \pi) \in \sigma \;\&\; \checkmark(s \ominus p)\}$
***a.*** $\sigma \uparrow a \;=\; \{(s, \pi \oplus a) | (s, \pi) \in \sigma\}$
$\sigma \downarrow a \;=\; \{(s, \pi \ominus a) | (s, \pi) \in \sigma\}$
**¬.** $\sigma \uparrow \neg\phi \;=\; \sigma \downarrow \phi$
$\sigma \downarrow \neg\phi \;=\; \sigma \uparrow \phi$
**∨.** $\sigma \uparrow (\phi \vee \psi) \;=\; (\sigma \uparrow \phi) \cup (\sigma \uparrow \psi)$
$\sigma \downarrow (\phi \vee \psi) \;=\; (\sigma \downarrow \phi) \downarrow \psi$
**∧.** $\sigma \uparrow (\phi \wedge \psi) \;=\; (\sigma \uparrow \phi) \uparrow \psi$
$\sigma \downarrow (\phi \wedge \psi) \;=\; (\sigma \downarrow \phi) \cup (\sigma \downarrow \psi)$

Otherwise, the output is ∅.

**P2** (problem):
This approach to imperatives conflates *semantics* (understanding the imperative) with *decision making* (deciding whether to comply)—e.g. in (12) it is not clear how this approach could model Ole's refusal to comply with Dad's command.

SAMPLE DERIVATIONS:

$\sigma_0 \;=\; \{\langle\varnothing, \varnothing\rangle\}$

**(22)** i. Ole set the table.

| | | |
|---|---|---|
| $\sigma_1$ | $:= \sigma_0 \uparrow set$ | |
| | $= \{(s \oplus set, \pi) \mid (s, \pi) \in \sigma_0 \;\&\; \checkmark(s \oplus set)\}$ | $\uparrow p$ |
| | $= \{(s \cup \{\langle set, 1\rangle\}, \pi) \mid (s, \pi) \in \sigma_0 \;\&\; \checkmark(s \cup \{\langle set, 1\rangle\})\}$ | $\oplus p$ |
| | $= \{(\{\langle set, 1\rangle\}, \varnothing)\}$ | $\sigma_0$ |

ii. Right, he set the table and washed up.

| | | |
|---|---|---|
| $\sigma_2$ | $:= \sigma_1 \uparrow (set \wedge wsh)$ | |
| | $= (\sigma_1 \uparrow set) \uparrow wsh$ | $\wedge$ |
| | $= \{(s \oplus set, \pi) \mid (s, \pi) \in \sigma_1 \;\&\; \checkmark(s \oplus set)\} \uparrow wsh$ | $\uparrow p$ |
| | $= \sigma_1 \uparrow wsh$ | $\oplus p$, $\sigma_1$ |
| | $= \{(s \oplus wsh, \pi) \mid (s, \pi) \in \sigma_1 \;\&\; \checkmark(s \oplus wsh)\}$ | $\uparrow p$ |
| | $= \{(\{\langle set, 1\rangle. \langle wsh, 1\rangle\}, \varnothing)\}$ | $\oplus p$, $\sigma_1$ |

C: #True, he didn't wash up.

| | | |
|---|---|---|
| $\sigma_3$ | $:= \sigma_2 \uparrow \neg wsh$ | |
| | $= \sigma_2 \downarrow wsh$ | $\neg$ |
| | $= \{(s \ominus wsh, \pi) \mid (s, \pi) \in \sigma_2 \;\&\; \checkmark(s \ominus wsh)\}$ | $\downarrow p$ |
| | $= \{(s \cup \{\langle wsh, 0\rangle\}, \pi) \mid (s, \pi) \in \sigma_2 \;\&\; \checkmark(s \cup \{\langle wsh, 0\rangle\})\}$ | $\ominus p$ |
| | $= \varnothing$ | $\sigma_2$, $\checkmark s$ |

**(28)** i. If you see a child in danger, help it!

| | | |
|---|---|---|
| $\sigma_1$ | $:= \sigma_0 \uparrow\; see \rightarrow hlp!$ | |
| | $= \sigma_0 \uparrow \neg see \vee (see \wedge hlp!)$ | $\rightarrow$ |
| | $= (\sigma_0 \uparrow \neg see) \cup (\sigma_0 \uparrow see \wedge hlp!)$ | $\vee$ |
| | $= (\sigma_0 \downarrow see) \cup (\sigma_0 \uparrow see) \uparrow hlp!$ | $\neg$, $\wedge$ |
| | $= \{(\{\langle see, 0\rangle\}, \varnothing)\} \cup \{(\{\langle see, 1\rangle\}, \varnothing)\} \uparrow help!$ | as above |
| | $= \{(\{\langle see, 0\rangle\}, \varnothing)\} \cup \{(\{\langle see, 1\rangle\}, \{\langle hlp!, \text{Y}\rangle\})\}$ | $\uparrow a$, $\oplus a$ |
| | $= \{(\{\langle see, 0\rangle\}, \varnothing), (\{\langle see, 1\rangle\}, \{\langle hlp!, \text{Y}\rangle\})\}$ | $\cup$ |

ii. If you love your family, don't help Jews!

| | | |
|---|---|---|
| $\sigma_2$ | $:= \sigma_1 \uparrow lov \rightarrow \neg hlp!$ | |
| | $= \varnothing$, | |
| because | $\sigma_1 \uparrow \neg lov \vee (lov \wedge \neg hlp!)$ | $\rightarrow$ |
| | $= (\sigma_1 \uparrow \neg lov) \cup (\sigma_1 \uparrow lov \wedge \neg hlp!)$ | $\vee$ |
| | $= (\sigma_1 \downarrow lov) \cup (\sigma_1 \uparrow lov) \downarrow hlp!$ | $\neg$, $\wedge$, $\neg$ |
| | $= \{(\{\langle see, 0\rangle, \langle lov, 0\rangle\}, \varnothing), (\{\langle see, 1\rangle, \langle lov, 0\rangle\}, \{\langle hlp!, \text{Y}\rangle\})\}$ | |
| | $\cup\; \{(\{\langle see, 0\rangle, \langle lov, 1\rangle\}, \{\langle hlp!, \text{N}\rangle\}),$ | $\downarrow a$, $\ominus a$, … |
| | $\quad (\{\langle see, 1\rangle, \langle lov, 1\rangle\}, \{\boldsymbol{\langle hlp!, \text{Y}\rangle, \langle hlp!, \text{N}\rangle}\})\}$ | |

| OVERVIEW OF PREDICTIONS: | | **MB *intuition*** |
|---|---|---|
| $\sigma_0$ | $= \{\langle\varnothing, \varnothing\rangle\}$ | |
| **(22)** | A: Ole set the table. | |
| $\sigma_1$ | $:= \sigma_0 \uparrow \mathit{set}$ | ✓(intuitive) |
| | $= \{(\{\langle \mathit{set}, 1\rangle\}, \varnothing)\}$ | |
| | B: Right, he set the table and washed up. | |
| $\sigma_2$ | $:= \sigma_1 \uparrow (\mathit{set} \wedge \mathit{wsh})$ | ✓ |
| | $= \{(\{\langle \mathit{set}, 1\rangle. \langle \mathit{wsh}, 1\rangle\}, \varnothing)\}$ | |
| | C: #True, he didn't wash up. | |
| $\sigma_3$ | $:= \sigma_2 \uparrow \neg \mathit{wsh}$ | ✓ |
| | $= \varnothing$ | |
| **(23)** | A: Ole set the table or washed up. | |
| $\sigma_1$ | $:= \sigma_0 \uparrow (\mathit{set} \vee \mathit{wsh})$ | ✓ |
| | $= \{(\{\langle \mathit{set}, 1\rangle\}, \varnothing). (\{\langle \mathit{wsh}, 1\rangle\}, \varnothing)\}$ | |
| | B: Right, he set the table. | |
| $\sigma_2$ | $:= \sigma_1 \uparrow \mathit{set}$ | ✓ |
| | $= \{(\{\langle \mathit{set}, 1\rangle\}, \varnothing), (\{\langle \mathit{set}, 1\rangle, \langle \mathit{wsh}, 1\rangle\}, \varnothing)\}$ | (see C–C″) |
| | C: True, he didn't wash up. | |
| $\sigma_3$ | $:= \boldsymbol{\sigma_2} \uparrow \neg \mathit{wsh}$ | ✓ |
| | $= \{(\{\langle \mathit{set}, 1\rangle, \langle \mathit{wsh}, 0\rangle\}, \varnothing)\}$ | $= \boldsymbol{\sigma_1} \uparrow \neg \mathit{wsh}$ |
| | C′: Indeed, he also washed up. | |
| $\sigma'_3$ | $:= \boldsymbol{\sigma_2} \uparrow \mathit{wsh}$ | ✓ |
| | $= \{(\{\langle \mathit{set}, 1\rangle, \langle \mathit{wsh}, 1\rangle\}, \varnothing)\}$ | $= \boldsymbol{\sigma_1} \uparrow \mathit{wsh}$ |
| | C″: #Right, he didn't set the table. | |
| $\sigma''_3$ | $:= \boldsymbol{\sigma_2} \uparrow \neg \mathit{set}$ | ✓ |
| | $= \varnothing$ | $\neq \boldsymbol{\sigma_1} \uparrow \neg \mathit{set}$ |
| **(24)** | A: Set the table! | |
| $\sigma_1$ | $:= \sigma_0 \uparrow \mathit{set}!$ | ? (questionable, see **P2**) |
| | $= \{(\varnothing, \{\langle \mathit{set}, \mathrm{Y}\rangle\})\}$ | |
| | B: Right, set the table and wash up! | |
| $\sigma_2$ | $:= \sigma_1 \uparrow (\mathit{set}! \wedge \mathit{wsh}!)$ | ? (**P2**) |
| | $= \{(\varnothing, \{\langle \mathit{set}!, \mathrm{Y}\rangle, \langle \mathit{wsh}!, \mathrm{Y}\rangle\})\}$ | |
| | C. #Don't wash up! | |
| $\sigma_3$ | $:= \boldsymbol{\sigma_2} \uparrow \neg \mathit{wsh}!$ | ? (**P2**) |
| | $= \varnothing$ | $\neq \boldsymbol{\sigma}_1 \uparrow \neg \mathit{wsh}!$ |

**(25)** A. Set the table or wash up!

$\sigma_1$ := $\sigma_0 \uparrow (set! \vee wsh!)$ ? (**P2**)

= $\{(\varnothing, \{\langle set!, \text{Y}\rangle\}), (\varnothing, \{\langle wsh!, \text{Y}\rangle\})\}$

B. ?Right, set the table!

$\sigma_2$ := $\sigma_1 \uparrow set!$ × (counter-intuitive)

= $\{(\varnothing, \{\langle set!, \text{Y}\rangle\}), (\varnothing, \{\langle set!, \text{Y}\rangle, \langle wsh!, \text{Y}\rangle\})\}$

**(26)** i. Ole either has or has not washed up.

$\sigma_1$ := $\sigma_0 \uparrow (wsh \vee \neg wsh)$ ✓

= $\{(\{\langle wsh, 1\rangle\}, \varnothing), (\{\langle wsh, 0\rangle\}, \varnothing)\}$

ii. If he has, pay him!

$\sigma_2$ := $\sigma_1 \uparrow (wsh \rightarrow pay!)$ ? (**P1**, **P2**)

= $\{(\{\langle wsh, 1\rangle\}, \{\langle pay!, \text{Y}\rangle\}), (\{\langle wsh, 0\rangle\}, \varnothing)\}$

iii. If he hasn't, don't!

$\sigma_3$ := $\sigma_2 \uparrow (\neg wsh \rightarrow \neg pay!)$ ? (**P1**, **P2**)

= $\{(\{\langle wsh, 1\rangle\}, \{\langle pay!, \text{Y}\rangle\}),$
$(\{\langle wsh, 0\rangle\}, \{\langle pay!, \text{N}\rangle\})\}$

**(25)** i. Don't open your eyes!

$\sigma_1$ := $\sigma_0 \uparrow \neg opn!$

= $\{(\varnothing, \{\langle opn!, \text{N}\rangle\})\}$ ? (**P2**)

ii. If you open your eyes, you won't go home.

$\sigma_2$ := $\sigma_1 \uparrow (opn \rightarrow \neg ghm)$ ×

= $\{(\{\langle opn, 0\rangle\}, \{\langle opn!, \text{N}\rangle\}),$
$(\{\langle opn, 1\rangle, \langle ghm, 0\rangle\}, \{\langle opn!, \text{N}\rangle\})\}$

**(26)** i. If you see a child in danger, help him.

$\sigma_1$ := $\sigma_0 \uparrow (see \rightarrow hlp!)$ ? (**P1**, **P2**)

= $\{(\{\langle see, 1\rangle\}, \{\langle hlp!, \text{Y}\rangle\}), (\{\langle see, 0\rangle\}, \varnothing)\}$

ii. If you love your family, don't help Jews.

$\sigma_2$ := $\sigma_1 \uparrow (lov \rightarrow \neg hlp!)$

= $\varnothing$ ✓/? (**P1**)

# 3 UPDATE WITH ACTION PLANS (Mastop 2005, Ch. 5)

NOTE: In what follows I simplify Mastop's ch. 5 system to focus on temporal issues. I haven't been able to apply his ch. 5 system as it stands (the definitions don't add up to a coherent story) so I have done my best to correct his definitions along the lines of what I take to be his basic ideas (e.g. as indicated below, my definition 1 is based on Mastop's definitions A.10–A.12 on pp. 167–172). Finally, I have modified his terminology to conform to English usage, and his notation, to make it easier to understand.

**Definition 1** (Vocabulary & syntax) (cf. A.10-12)

0. $N = \{al, ed, \dots\}$
   $V^0 = \{set, wsh, out, \dots\}$
   $V^1 = \{pay, find, lv, \dots\}$
   $V^2 = \{hand, give, send, \dots\}$

i. $\beta\alpha_1\dots\alpha_{n+1} \in UP_0$ if $\beta \in V^n$ & $\alpha_1,\dots, \alpha_{n+1} \in N$
   $\beta\alpha_1\dots\alpha_n! \in UP_0$ if $\beta \in V^n$ & $\alpha_1,\dots, \alpha_n \in N$

ii. $UP_0 \subseteq UP_1$
   $\neg\phi \in UP_1$ if $\phi \in UP_1$
   $\phi \wedge \psi, \phi \vee \psi \in UP_1$ if $\phi, \psi \in UP_1$

ii′. $\phi \to \psi := \neg\phi \vee (\phi \wedge \psi)$ if $\phi$ does not contain any '!'

iii. $UP_1 \subseteq UP_2$
   $\mathbf{P}\phi, \mathbf{F}\phi, \mathbf{T}\phi \in UP_2$ if $\phi \in UP_2$

| **e.g.** | | English | $UP_2$ |
|---|---|---|---|
| (1) | A: | Set the table! | *set*! |
| | B: | Right, set the table and wash up! | *set*! ∧ *wsh*! |
| (2) | i. | Today Ed helped Jo (a bit). | **TP***hlp jo ed* |
| | ii. | He washed up. | **P***wsh ed* |

[Jo to Ed]

| | | | |
|---|---|---|---|
| (3) | i. | Don't go out! | ¬*out*! |
| | ii. | If you do, I'll leave you. | **F**(*out ed* → *lv ed jo*) |
| (3′) | i. | Don't go out today! | **T**¬*out*! |
| | ii. | If you do, I'll leave you. | **F**(*out ed* → *lv ed jo*) |
| (4) | i. | Ed either did or did not wash up. | **P**(*wsh ed* ∨ ¬*wsh ed*) |
| | ii. | If he did, pay him! | **P**(*wsh ed* → *pay ed*!) |
| | iii. | if he didn't, don't! | **P**(¬*wsh ed* → ¬*pay ed*!) |

**Definition 2** ($UP_2$ models) (cf. A.1–2)

A $UP_2$-model is a tuple $\langle D, \mathcal{A}, \mathcal{E}, \text{EXE}, \langle T, <, \subseteq\rangle, [\![\cdot]\!]\rangle$, where

a. $D$ is a set of *individuals*, incl. *humans* $H \subseteq D$
   $\mathcal{A}$ is a set of *types of planned actions*
   $\mathcal{E}$ is a set of *types of realized events*
   $\text{EXE}: \mathcal{A} \times H \rightarrow \mathcal{E}$ (event type: execution of $A \in \mathcal{A}$ by human $h \in H$)

b. $T$ is a set of *times*, including *days* $T_0 \subseteq T$
   $t \subseteq \text{PST}\ t'$ iff $t < t'$ (past of $t'$)
   $t \subseteq \text{FUT}\ t'$ iff $t' < t$ (future of $t'$)
   $t \subseteq \text{TOD}\ t'$ iff $\exists t''(t'' \in T_0\ \&\ t \subseteq t''\ \&\ t' \subseteq t'')$ (today of $t'$)

c. $[\![\alpha]\!] \in D$ if $\alpha \in N$
   $[\![\beta]\!]: D^n \rightarrow \mathcal{A}$ if $\beta \in V^n$

**Definition 3** (situations & plans) (cf. A.3)

- $s \subseteq \mathcal{E} \times T \times \{1, 0\}$ is a *situation*
  $S := Pow(\mathcal{E} \times T \times \{0, 1\})$ is the set of situations
- $\pi \subseteq \mathcal{A} \times T \times \{\text{Y}, \text{N}\}$ is a *plan*
  $\Pi := Pow(\mathcal{A} \times T \times \{\text{Y}, \text{N}\})$ is the set of plans

**e.g.**

- $s_1 = \{\langle \text{EXE}([\![pay]\!]([\![ed]\!]), [\![al]\!]), t_{-1}, 1\rangle\}$
  is a situation where $[\![al]\!]$ executes an action of type $[\![pay]\!]([\![ed]\!])$ in $t_{-1}$
- $\pi_1 = \{\langle [\![pay]\!]([\![ed]\!]), t_1, \text{Y}\rangle, \langle [\![wsh]\!], t_2, \text{N}\rangle\}$
  is a plan to execute a $[\![pay]\!]([\![ed]\!])$-action in $t_1$, no $[\![wsh]\!]$-action in $t_2$
- $\varnothing$ is the maximally unspecified situation (or plan)

**Definition 4** (temporal restriction) (cf. A.15)

- $(s|t) := \{\langle E, t', 1\rangle |\ t' \leq t\} \cup \{\langle E, t', 0\rangle |\ t' \leq t\}$ is situation $s$ up to $t$
  $(t|\pi) := \{\langle A, t', \text{Y}\rangle |\ t < t'\} \cup \{\langle A, t', \text{N}\rangle |\ t < t'\}$ is plan $\pi$ from $t$ onward
- situation $s$ is *t-restricted* iff $s = (s|t)$, plan $\pi$ is *t-restricted* iff $\pi = (t|\pi)$

**e.g.** if $t_{-1} \leq t_0 < t_1 < t_2$

- $s_1 = \{\langle \text{EXE}([\![set]\!], [\![ed]\!]), t_{-1}, 1\rangle, \langle \text{EXE}([\![wsh]\!], [\![ed]\!]), t_0, 0\rangle\}$
  $= (s_1|t_0)$
  $\neq (s_1|t_{-1}) = \{\langle \text{EXE}([\![set]\!], [\![ed]\!]), t_{-1}, 1\rangle\}$
- $\pi_1 = \{\langle [\![set]\!], t_1, \text{Y}\rangle, \langle [\![wsh]\!], t_2, \text{N}\rangle\}$
  $= (t_0|\pi)$
  $\neq (t_1|\pi) = \{\langle [\![wsh]\!], t_2, \text{N}\rangle\}$
- $\varnothing = (t|\varnothing) = (\varnothing|t)$ for all $t \in T$

**Definition 5** (possibilities, states, contexts) (cf. A.13, 18)

- $(s, \pi) \in S \times \Pi$ is a *possibility*
- $\sigma \subseteq S \times \Pi$ is a (*cognitive*) *state*
  $\Sigma := Pow(S \times \Pi)$ is the set of (cognitive) states
- $c \in \Sigma \times H \times T \times T$ is a *context*

**e.g.**

- $\sigma_1 = \{(\{\langle \text{EXE}([\![wsh]\!], [\![ed]\!]), t_{-1}, 1\rangle\}, \{\langle [\![pay]\!]([\![ed]\!]), t_1, \text{Y}\rangle\}),$
  $(\{\langle \text{EXE}([\![wsh]\!], [\![ed]\!]), t_{-1}, 0\rangle\}, \{\langle [\![pay]\!]([\![ed]\!]), t_1, \text{N}\rangle\})\}$
  exp. of $\sigma_1$ doesn't know whether $[\![ed]\!]$ executed a $[\![wsh]\!]$-action in $t_{-1}$, plans to execute $[\![pay]\!]([\![ed]\!])$ in $t_1$ if $[\![ed]\!]$ did execute $[\![wsh]\!]$ in $t_{-1}$, and plans not to execute $[\![pay]\!]([\![ed]\!])$ in $t_1$ if $[\![ed]\!]$ did not execute $[\![wsh]\!]$ in $t_{-1}$
- $\{(\varnothing, \varnothing)\}$ $\varnothing$
  maximally unspecified state absurd state
- $\langle \sigma_1, [\![al]\!], t_0, \text{PST } t_0\rangle$
  is a context where $[\![al]\!]$ is in the cognitive state $\sigma_1$, his now is $t_0$, and the temporal location frame is the (entire) past of $t_0$
- $\langle \varnothing, h, t, t'\rangle$ is an impossible context, for all $h, t, t'$ (update stops)

**Definition 6** (consistency & quandry) (cf. A.4)

- a situation $s$ is *consistent*, $\checkmark s$,
  iff $\neg\exists E, t, t'\colon \langle E, t, 1\rangle \in s \wedge t \subseteq t' \wedge \langle E, t', 0\rangle \in s$
- a plan $\pi$ is *consistent*, $\checkmark\pi$,
  iff $\neg\exists A, t, t'\colon \langle A, t, \text{Y}\rangle \in \pi \wedge t \subseteq t' \wedge \langle A, t', \text{N}\rangle \in \pi$
- possibility $(s, \pi)$ is a *quandry* iff $s$ or $\pi$ is not consistent
- a context $c = \langle \sigma, h, t, t'\rangle$ is *quandry free* iff there is no quandry in $\sigma$

**Definition 7** (expansions) (cf. A.7, 19)

- $s \oplus \langle E, t'\rangle \;=\; s \cup \{\langle E, t', 1\rangle\}$
  $s \ominus \langle E, t'\rangle \;=\; s \cup \{\langle E, t', 0\rangle\}$
- $\pi \oplus \langle A, t'\rangle \;=\; \pi \cup \{\langle A, t', \text{Y}\rangle\}$
  $\pi \ominus \langle A, t'\rangle \;=\; \pi \cup \{\langle A, t', \text{N}\rangle\}$
- $\sigma \uparrow_t \langle E, t'\rangle$
  $= \{(s \oplus \langle E, t'\rangle, \pi) | \; (s, \pi) \in \sigma \;\&\; s = (s|t) \;\&\; t' \leq t \;\&\; \checkmark(s \oplus \langle E, t'\rangle)\}$
  $\sigma \downarrow_t \langle E, t'\rangle$
  $= \{(s \ominus \langle E, t'\rangle, \pi) | \; (s, \pi) \in \sigma \;\&\; s = (s|t) \;\&\; t' \leq t \;\&\; \checkmark(s \ominus \langle E, t'\rangle)\}$
- $\sigma \uparrow_t \langle A, t'\rangle$
  $= \{(s, \pi \oplus \langle A, t'\rangle) | \; (s, \pi) \in \sigma \;\&\; \pi = (t|\pi) \;\&\; t < t'\}$
  $\sigma \downarrow_t \langle A, t'\rangle$
  $= \{(s, \pi \ominus \langle A, t'\rangle) | \; (s, \pi) \in \sigma \;\&\; \pi = (t|\pi) \;\&\; t < t'\}$

**Definition 8** (positive update $c[\cdot]^+$, negative update $c[\cdot]^-$) (cf. A.19–21)
Let $c = \langle\sigma, h, t, t'\rangle$. If quandry free,

$$
\begin{array}{lll}
\textbf{\textit{E}}. & c[\beta\alpha_1\ldots\alpha_{n+1}]^+ & = \langle(\sigma \uparrow_t \langle \text{EXE}([\![\beta]\!]([\![\alpha_1]\!],\ldots[\![\alpha_n]\!]), [\![\alpha_{n+1}]\!]), t'\rangle), h, t, t'\rangle \\
 & c[\beta\alpha_1\ldots\alpha_{n+1}]^- & = \langle(\sigma \downarrow_t \langle \text{EXE}([\![\beta]\!]([\![\alpha_1]\!],\ldots[\![\alpha_n]\!]), [\![\alpha_{n+1}]\!]), t'\rangle), h, t, t'\rangle \\
\textbf{\textit{A}}. & c[\beta\alpha_1\ldots\alpha_n!]^+ & = \langle(\sigma \uparrow_t \langle [\![\beta]\!]([\![\alpha_1]\!],\ldots[\![\alpha_n]\!]), t'\rangle), h, t, t'\rangle \\
 & c[\beta\alpha_1\ldots\alpha_n!]^- & = \langle(\sigma \downarrow_t \langle [\![\beta]\!]([\![\alpha_1]\!],\ldots[\![\alpha_n]\!]), t'\rangle), h, t, t'\rangle \\
\neg. & c[\neg\phi]^+ & = c[\phi]^- \\
 & c[\neg\phi]^- & = c[\phi]^+ \\
\vee. & c[\phi \vee \psi]^+ & = \langle(c[\phi]^+)_1 \cup (c[\psi]^+)_1, h, t, t'\rangle \\
 & c[\phi \vee \psi]^- & = c[\phi]^-[\psi]^- \\
\wedge. & c[\phi \wedge \psi]^+ & = c[\phi]^+[\psi]^+ \\
 & c[\phi \wedge \psi]^- & = \langle(c[\phi]^-)_1 \cup (c[\psi]^-)_1, h, t, t'\rangle \\
\textbf{P}. & \langle\sigma, h, t, t'\rangle[\textbf{P}\phi]^+ & = \langle\sigma, h, t, \text{PST } t\rangle[\phi]^+ \\
 & \langle\sigma, h, t, t'\rangle[\textbf{P}\phi]^- & = \langle\sigma, h, t, \text{PST } t\rangle[\phi]^- \\
\textbf{F}. & \langle\sigma, h, t, t'\rangle[\textbf{F}\phi]^+ & = \langle\sigma, h, t, \text{FUT } t\rangle[\phi]^+ \\
 & \langle\sigma, h, t, t'\rangle[\textbf{F}\phi]^- & = \langle\sigma, h, t, \text{FUT } t\rangle[\phi]^- \\
\textbf{T}. & \langle\sigma, h, t, t'\rangle[\textbf{T}\phi]^+ & = \langle\sigma, h, t, t' \cap \text{TOD } t\rangle[\phi]^+ \\
 & \langle\sigma, h, t, t'\rangle[\textbf{T}\phi]^- & = \langle\sigma, h, t, t' \cap \text{TOD } t\rangle[\phi]^-
\end{array}
$$

Otherwise, the output context is $\langle\varnothing, h, t, t'\rangle$ (absurd context, update stops).

SAMPLE DERIVATIONS (see also app. C)

$c_0 = \langle\sigma_0, [\![ed]\!], t_0, t_1\rangle$
where $\sigma_0 = \{\langle\varnothing, \varnothing\rangle\}$ & $t_0 < t_1$

**(1)** Mom to Ed: Set the table!

$$
\begin{array}{lll}
c_1 & := c_0[set!]^+ & \\
 & = \langle(\sigma_0 \uparrow_{t0} \langle [\![set]\!], t_1\rangle), [\![ed]\!], t_0, t_1\rangle & \textbf{\textit{A}}
\end{array}
$$

where

$$
\begin{array}{lll}
\sigma_1 & := \sigma_0 \uparrow_{t0} \langle [\![set]\!], t_1\rangle & \\
 & = \{(s, \pi \oplus \langle [\![set]\!], t_1\rangle) | (s, \pi) \in \sigma_0 \ \& \ \pi = (t_0|\pi) \ \& \ t_0 < t_1\} & \sigma\uparrow \\
 & = \{(s, \pi \cup \{\langle [\![set]\!], t_1, \text{Y}\rangle\}) | (s, \pi) \in \sigma_0 \ \& \ \pi = (t_0|\pi) \ \& \ t_0 < t_1\} & \pi\oplus \\
 & = \{(\varnothing, \{\langle [\![set]\!], t_1, \text{Y}\rangle\})\} & \sigma_0, (t|\pi)
\end{array}
$$

Dad: Right, set the table and wash up!

$$
\begin{array}{lll}
c_2 & := c_1[set! \wedge wsh!]^+ & \\
 & = c_1[set!]^+[wsh!]^+ & \wedge \\
 & = \langle(\sigma_1 \uparrow_{t0} \langle [\![wsh]\!], t_1\rangle), [\![ed]\!], t_0, t_1\rangle & \text{as above}
\end{array}
$$

where

$$
\begin{array}{lll}
\sigma_2 & := \sigma_1 \uparrow_{t0} \langle [\![wsh]\!], t_1\rangle & \\
 & = \{(\varnothing, \{\langle [\![set]\!], t_1, \text{Y}\rangle, \langle [\![wsh]\!], t_1, \text{Y}\rangle\})\} & \text{as above}
\end{array}
$$

OVERVIEW OF PREDICTIONS    ***MB intuitions***

• Initial context from addressee's perspective

$c_0 \;=\; \langle \sigma_0, \llbracket ed \rrbracket, t_0, t_1 \rangle$

where    *stipulation* 1:

$\sigma_0 \;=\; \{\langle \varnothing, \varnothing \rangle\} \;\&\; t_0 < t_1$    $t_0 < t_1$

**(1)** Mom to Ed: Set the table!

$c_1 \;:=\; c_0[set!]^+$

$\quad = \;\langle(\sigma_1, \llbracket ed \rrbracket, t_0, t_1 \rangle$    ✓, but need

where    stipulation 1

$\sigma_1 \;=\; \{(\varnothing,\; \{\langle \llbracket set \rrbracket, t_1, \textsc{y} \rangle\})\}$

Dad to Ed: Right, set the table and wash up!

$c_2 \;:=\; c_1[set! \wedge wsh!]^+$

$\quad = \;\langle \sigma_2, \llbracket ed \rrbracket, t_0, t_1 \rangle$

where    ?, same frame

$\sigma_2 \;=\; \{(\varnothing,\; \{\langle \llbracket set \rrbracket, t_1, \textsc{y} \rangle, \langle \llbracket wsh \rrbracket, t_1, \textsc{y} \rangle\})\}$    ($t_1$) for both?

• Initial context from addressee's perspective

$c_0 \;=\; \langle \sigma_0, \llbracket al \rrbracket, t_0, t_{-1} \rangle$

where    *stipulation* 2:

$\sigma_0 \;=\; \{\langle \varnothing, \varnothing \rangle\} \;\&\; t_{-1} \cap \textsc{tod}\; t_0 \cap \textsc{pst}\; t_0 \in T$    $t_{-1} \ldots \in T$

**(2)** i. Today Ed helped Jo (a bit).

$c_1 \;:=\; \langle \sigma_0, \llbracket al \rrbracket, t_0, t_{-1} \rangle[\mathbf{TP}hlp\; jo\; ed]^+$

$\quad = \;\langle \sigma_1, \llbracket al \rrbracket, t_0, t_{-1} \cap \textsc{tod}\; t_0 \cap \textsc{pst}\; t_0 \rangle$    ✓, but need

where    stipulation 2

$\sigma_1 \;=\; \{(\{\langle \textsc{exe}(\llbracket hlp \rrbracket(\llbracket jo \rrbracket), \llbracket ed \rrbracket), t_{-1} \cap \textsc{tod}\; t_0 \cap \textsc{pst}\; t_0, 1 \rangle\}, \varnothing)\}$

ii. He washed up.

$c_2 \;:=\; \langle \sigma_1, \llbracket al \rrbracket, t_0, t_{-1} \cap \textsc{tod}\; t_0 \cap \textsc{pst}\; t_0 \rangle[\mathbf{P}wsh\; ed]^+$

$\quad = \;\langle \sigma_2, \llbracket al \rrbracket, t_0, \textsc{pst}\; t_0 \rangle$    × (tmp. update)

where

$\sigma_2 \;=\; \{(\{\langle \textsc{exe}(\llbracket hlp \rrbracket(\llbracket jo \rrbracket), \llbracket ed \rrbracket), t_{-1} \cap \textsc{tod}\; t_0 \cap \textsc{pst}\; t_0, 1 \rangle,$

$\qquad \langle \textsc{exe}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \textsc{pst}\; t_0, 1 \rangle\}, \varnothing)\}$    × (tmp. frame)

Re: *stipulation* 2

This analysis wrongly predicts # (2′), because $\textsc{yst}\; t_0 \cap \textsc{tod}\; t_0 \notin T$:

**(2′)** i. Yesterday Ed didn't help Jo.

ii. But today he did help her (a bit).

• Initial context from addressee's perspective — ***MB intuitions***

$c_0 = \langle \sigma_0, [\![ed]\!], t_0, t_1 \rangle$

where

$\sigma_0 = \{\langle \varnothing, \varnothing \rangle\}$ & $t_0 < t_1$ — *stipulation* 1 again

**(3)** i. Don't go out!

$c_1 := c_0[\neg out!]^+$

$= \langle \sigma_1, [\![ed]\!], t_0, t_1 \rangle$ — × IMP as plan upd., if spkr has no authority over addressee

where

$\sigma_1 = \{(\varnothing, \{\langle [\![out]\!], t_1, \mathrm{N} \rangle\})\}$

ii. If you do, I'll leave you.

$c_2 := c_1[\mathbf{F}(out\ ed \rightarrow lv\ ed\ jo)]^+$

$= \langle \varnothing, [\![ed]\!], t_0, \mathrm{FUT}\ t_0 \rangle$ — ×××, crazy result

• Initial context from addressee's perspective

$c_0 = \langle \sigma_0, [\![al]\!], t_0, t_1 \rangle$

where

$\sigma_0 = \{\langle \varnothing, \varnothing \rangle\}$

**(4)** i. Ed either did or did not wash up.

$c_1 := \langle \sigma_0, [\![al]\!], t_0, t_1 \rangle[\mathbf{P}(wsh\ ed \vee \neg wsh\ ed)]^+$

$= \langle \sigma_1, [\![al]\!], t_0, \mathrm{PST}\ t_0 \rangle$

where

$\sigma_1 = \{(\{\langle \mathrm{EXE}([\![wsh]\!], [\![ed]\!]), \mathrm{PST}\ t_0, 1 \rangle\}, \varnothing),$ — ✓ frame for $^+$-fact

$(\{\langle \mathrm{EXE}([\![wsh]\!], [\![ed]\!]), \mathrm{PST}\ t_0, 0 \rangle\}, \varnothing)\}$ — × frame for $^-$-fact

ii. If he did, pay him!

$c_2 := \langle \sigma_1, [\![al]\!], t_0, \mathrm{PST}\ t_0 \rangle[\mathbf{P}(wsh\ ed \rightarrow pay\ ed!)]^+$

$= \langle \sigma_2, [\![al]\!], t_0, \mathrm{PST}\ t_0 \rangle$ — ××, predict (ii) ≡ (ii′)

where — ii′. He has never washed up.

$\sigma_2 = \{(\{\langle \mathrm{EXE}([\![wsh]\!], [\![ed]\!]), \mathrm{PST}\ t_0, 0 \rangle\}, \varnothing)\}$

iii. If he didn't, don't!

$c_3 := \langle \sigma_2, [\![al]\!], t_0, \mathrm{PST}\ t_0 \rangle[\mathbf{P}(\neg wsh\ ed \rightarrow \neg pay\ ed!)]^+$ — ×××, more craziness

$= \langle \varnothing, [\![al]\!], t_0, \mathrm{PST}\ t_0 \rangle$

***MB conclusion***: Time to look for another approach to imperatives.

## A. UPDATE WITH PLANS ($UP_1$)

**Definition 4.3.** (Syntax)

i. $UP_0 = P \cup A$,
where $P = \{set, wsh, opn, ent, \ldots\}$ is a set of atomic declaratives
$A = \{set!, wsh!, opn!, ent!, \ldots\}$ is a set of atomic imperatives

ii. $UP_0 \subseteq UP_1$
$\neg\phi \in UP_1$ if $\phi \in UP_1$
$\phi \wedge \psi, \phi \vee \psi \in UP_1$ if $\phi, \psi \in UP_1$

iii. $\phi \rightarrow \psi := \neg\phi \vee (\phi \wedge \psi)$ if $\phi$ contains only declarative atoms

**Definition 4.4** (situations & plans)

- $s \subseteq P \times \{1, 0\}$ is a *situation*
  $S := Pow(P \times \{0, 1\})$ is the set of situations
- $\pi \subseteq A \times \{\text{Y}, \text{N}\}$ is a *plan*
  $\Pi := Pow(A \times \{\text{Y}, \text{N}\})$ is the set of plans

**Definition 4.6** (possibilities & states)

- $(s, \pi) \in S \times \Pi$ is a *possibility*
- $\sigma \subseteq S \times \Pi$ is a (*cognitive*) *state*
  $\Sigma := Pow(S \times \Pi)$ is the set of (cognitive) states

**Definition 4.7** (consistency & quandry)

- $s$ is *consistent*, $\checkmark s$, iff $\neg\exists p \in P: \langle p, 1\rangle \in s \wedge \langle p, 0\rangle \in s$
  $\pi$ is *consistent*, $\checkmark\pi$, iff $\neg\exists a \in A: \langle a, \text{Y}\rangle \in \pi \wedge \langle a, \text{N}\rangle \in \pi$
- A possibility $(s, \pi)$ is a *quandry* iff $s$ or $\pi$ is not consistent
  A state $\sigma$ is *quandry free* iff there is no quandry in $\sigma$

**Definition 4.5** (expansions). For any $p \in P$, $a \in A$:

- $s \oplus p = s \cup \{\langle p, 1\rangle\}$
  $s \ominus p = s \cup \{\langle p, 0\rangle\}$
- $\pi \oplus a = \pi \cup \{\langle a, \text{Y}\rangle\}$
  $\pi \ominus a = \pi \cup \{\langle a, \text{N}\rangle\}$

**Definition 4.8** (positive update $\uparrow$, negative update $\downarrow$). If quandry free,

| | | | |
|---|---|---|---|
| ***p.*** | $\sigma \uparrow p$ | = | $\{(s \oplus p, \pi) \mid (s, \pi) \in \sigma \ \& \ \checkmark(s \oplus p)\}$ |
| | $\sigma \downarrow p$ | = | $\{(s \ominus p, \pi) \mid (s, \pi) \in \sigma \ \& \ \checkmark(s \ominus p)\}$ |
| ***a.*** | $\sigma \uparrow a$ | = | $\{(s, \pi \oplus a) \mid (s, \pi) \in \sigma\}$ |
| | $\sigma \downarrow a$ | = | $\{(s, \pi \ominus a) \mid (s, \pi) \in \sigma\}$ |
| **¬.** | $\sigma \uparrow \neg\phi$ | = | $\sigma \downarrow \phi$ |
| | $\sigma \downarrow \neg\phi$ | = | $\sigma \uparrow \phi$ |
| **∨.** | $\sigma \uparrow (\phi \vee \psi)$ | = | $(\sigma \uparrow \phi) \cup (\sigma \uparrow \psi)$ |
| | $\sigma \downarrow (\phi \vee \psi)$ | = | $(\sigma \downarrow \phi) \downarrow \psi$ |
| **∧.** | $\sigma \uparrow (\phi \wedge \psi)$ | = | $(\sigma \uparrow \phi) \uparrow \psi$ |
| | $\sigma \downarrow (\phi \wedge \psi)$ | = | $(\sigma \downarrow \phi) \cup (\sigma \downarrow \psi)$ |

Otherwise, the output is $\varnothing$.

**Definition 4.9** (acceptance & entailment)

- $\phi$ is *accepted* in $\sigma$, $\sigma \Vdash \phi$, iff $\sigma \subseteq (\sigma \uparrow \phi)$
- $\phi_1, \ldots \phi_n \Vdash \psi$, iff $\forall\sigma$: $\sigma \uparrow \phi_1 \ \ldots \uparrow \phi_n \Vdash \psi$

## B. UPDATE WITH ACTION PLANS ($UP_2$)

**Definition 1** (Syntax) (cf. A.10-12)

i. $\beta\alpha_1\ldots\alpha_{n+1} \in UP_0$ if $\beta \in V^n$ & $\alpha_1,\ldots, \alpha_{n+1} \in N$
   $\beta\alpha_1\ldots\alpha_n! \in UP_0$ if $\beta \in V^n$ & $\alpha_1,\ldots, \alpha_n \in N$

ii. $UP_0 \subseteq UP_1$
   $\neg\phi \in UP_1$ if $\phi \in UP_1$
   $\phi \wedge \psi, \phi \vee \psi \in UP_1$ if $\phi, \psi \in UP_1$

ii′. $\phi \rightarrow \psi := \neg\phi \vee (\phi \wedge \psi)$ if $\phi$ does not contain any '!'

iii. $UP_1 \subseteq UP_2$
   $\mathbf{P}\phi, \mathbf{F}\phi, \mathbf{T}\phi \in UP_2$ if $\phi \in UP_2$

**Definition 2** ($UP_2$ models) (cf. A.1–2)

A $UP_2$-model is a tuple $\langle D, \mathcal{A}, \mathcal{E}, \text{EXE}, \langle T, <, \subseteq\rangle, [\![\cdot]\!]\rangle$, where

a. $D$ is a set of *individuals*, incl. *humans* $H \subseteq D$
   $\mathcal{A}$ is a set of *types of planned actions*
   $\mathcal{E}$ is a set of *types of realized events*
   $\text{EXE}: \mathcal{A} \times H \rightarrow \mathcal{E}$ (event type: execution of $A \in \mathcal{A}$ by human $h \in H$)

b. $T$ is a set of *times*, including *days* $T_0 \subseteq T$
   $t \subseteq \text{PST}\ t'$ iff $t < t'$ (past of $t'$)
   $t \subseteq \text{FUT}\ t'$ iff $t' < t$ (future of $t'$)
   $t \subseteq \text{TOD}\ t'$ iff $\exists t''(t'' \in T_0\ \&\ t \subseteq t''\ \&\ t' \subseteq t'')$ (today of $t'$)

c. $[\![\alpha]\!] \in D$ if $\alpha \in N$
   $[\![\beta]\!]: D^n \rightarrow \mathcal{A}$ if $\beta \in V^n$

**Definition 3** (situations & plans) (cf. A.3)

- $s \subseteq \mathcal{E} \times T \times \{1, 0\}$ is a *situation*
  $S := Pow(\mathcal{E} \times T \times \{0, 1\})$ is the set of situations
- $\pi \subseteq \mathcal{A} \times T \times \{\text{Y}, \text{N}\}$ is a *plan*
  $\Pi := Pow(\mathcal{A} \times T \times \{\text{Y}, \text{N}\})$ is the set of plans

**Definition 4** (temporal restriction) (cf. A.15)

- $(s|t) := \{\langle E, t', 1\rangle | \ t' \leq t\} \cup \{\langle E, t', 0\rangle | \ t' \leq t\}$ is situation $s$ up to $t$
- $(t|\pi) := \{\langle A, t', \text{Y}\rangle | \ t < t'\} \cup \{\langle A, t', \text{N}\rangle | \ t < t'\}$ is plan $\pi$ from $t$ onward

**Definition 5** (possibilities, states & contexts) (cf. A.13, 18)

- $(s, \pi) \in S \times \Pi$ is a *possibility*
- $\sigma \subseteq S \times \Pi$ is a (*cognitive*) *state*
  $\Sigma := Pow(S \times \Pi)$ is the set of (cognitive) states
- $c \in \Sigma \times H \times T \times T$ is a *context*

**Definition 6** (consistency & quandry) (cf. A.4)

- a situation $s$ is *consistent*, $\checkmark s$,
  iff $\neg\exists E, t, t'$: $\langle E, t, 1\rangle \in s \wedge t \subseteq t' \wedge \langle E, t', 0\rangle \in s$
- a plan $\pi$ is *consistent*, $\checkmark\pi$,
  iff $\neg\exists A, t, t'$: $\langle A, t, \text{Y}\rangle \in \pi \wedge t \subseteq t' \wedge \langle A, t', \text{N}\rangle \in \pi$
- possibility $(s, \pi)$ is a *quandry* iff $s$ or $\pi$ is not consistent
- a context $c = \langle\sigma, h, t, t'\rangle$ is *quandry free* iff there is no quandry in $\sigma$

**Definition 7** (expansions) (cf. A.7, 19)

- $s \oplus \langle E, t'\rangle = s \cup \{\langle E, t', 1\rangle\}$
  $s \ominus \langle E, t'\rangle = s \cup \{\langle E, t', 0\rangle\}$
- $\pi \oplus \langle A, t'\rangle = \pi \cup \{\langle A, t', \text{Y}\rangle\}$
  $\pi \ominus \langle A, t'\rangle = \pi \cup \{\langle A, t', \text{N}\rangle\}$
- $\sigma \uparrow_t \langle E, t'\rangle$
  $= \{(s \oplus \langle E, t'\rangle, \pi)|\ (s, \pi) \in \sigma\ \&\ s = (s|t)\ \&\ t' \leq t\ \&\ \checkmark(s \oplus \langle E, t'\rangle)\}$
  $\sigma \downarrow_t \langle E, t'\rangle$
  $= \{(s \ominus \langle E, t'\rangle, \pi)|\ (s, \pi) \in \sigma\ \&\ s = (s|t)\ \&\ t' \leq t\ \&\ \checkmark(s \ominus \langle E, t'\rangle)\}$
- $\sigma \uparrow_t \langle A, t'\rangle$
  $= \{(s, \pi \oplus \langle A, t'\rangle)|\ (s, \pi) \in \sigma\ \&\ \pi = (t|\pi)\ \&\ t < t'\}$
  $\sigma \downarrow_t \langle A, t'\rangle$
  $= \{(s, \pi \ominus \langle A, t'\rangle)|\ (s, \pi) \in \sigma\ \&\ \pi = (t|\pi)\ \&\ t < t'\}$

**Definition 8** (positive update $c[\cdot]^+$, negative update $c[\cdot]^-$) (cf. A.19–21)

Let $c = \langle\sigma, h, t, t'\rangle$. If quandry free,

| | | | |
|---|---|---|---|
| ***E*.** | $c[\beta\alpha_1\ldots\alpha_{n+1}]^+$ | = | $\langle(\sigma \uparrow_t \langle \text{EXE}([\![\beta]\!]([\![\alpha_1]\!],\ldots[\![\alpha_n]\!]), [\![\alpha_{n+1}]\!]), t'\rangle), h, t, t'\rangle$ |
| | $c[\beta\alpha_1\ldots\alpha_{n+1}]^-$ | = | $\langle(\sigma \downarrow_t \langle \text{EXE}([\![\beta]\!]([\![\alpha_1]\!],\ldots[\![\alpha_n]\!]), [\![\alpha_{n+1}]\!]), t'\rangle), h, t, t'\rangle$ |
| ***A*.** | $c[\beta\alpha_1\ldots\alpha_n!]^+$ | = | $\langle(\sigma \uparrow_t \langle[\![\beta]\!]([\![\alpha_1]\!],\ldots[\![\alpha_n]\!]), t'\rangle), h, t, t'\rangle$ |
| | $c[\beta\alpha_1\ldots\alpha_n!]^-$ | = | $\langle(\sigma \downarrow_t \langle[\![\beta]\!]([\![\alpha_1]\!],\ldots[\![\alpha_n]\!]), t'\rangle), h, t, t'\rangle$ |
| **¬.** | $c[\neg\phi]^+$ | = | $c[\phi]^-$ |
| | $c[\neg\phi]^-$ | = | $c[\phi]^+$ |
| **∨.** | $c[\phi \vee \psi]^+$ | = | $\langle(c[\phi]^+)_1 \cup (c[\psi]^+)_1, h, t, t'\rangle$ |
| | $c[\phi \vee \psi]^-$ | = | $c[\phi]^-[\psi]^-$ |
| **∧.** | $c[\phi \wedge \psi]^+$ | = | $c[\phi]^+[\psi]^+$ |
| | $c[\phi \wedge \psi]^-$ | = | $\langle(c[\phi]^-)_1 \cup (c[\psi]^-)_1, h, t, t'\rangle$ |
| **P.** | $\langle\sigma, h, t, t'\rangle[\mathbf{P}\phi]^+$ | = | $\langle\sigma, h, t, \text{PST}\ t\rangle[\phi]^+$ |
| | $\langle\sigma, h, t, t'\rangle[\mathbf{P}\phi]^-$ | = | $\langle\sigma, h, t, \text{PST}\ t\rangle[\phi]^-$ |
| **F.** | $\langle\sigma, h, t, t'\rangle[\mathbf{F}\phi]^+$ | = | $\langle\sigma, h, t, \text{FUT}\ t\rangle[\phi]^+$ |
| | $\langle\sigma, h, t, t'\rangle[\mathbf{F}\phi]^-$ | = | $\langle\sigma, h, t, \text{FUT}\ t\rangle[\phi]^-$ |
| **T.** | $\langle\sigma, h, t, t'\rangle[\mathbf{T}\phi]^+$ | = | $\langle\sigma, h, t, t' \cap \text{TOD}\ t\rangle[\phi]^+$ |
| | $\langle\sigma, h, t, t'\rangle[\mathbf{T}\phi]^-$ | = | $\langle\sigma, h, t, t' \cap \text{TOD}\ t\rangle[\phi]^-$ |

Otherwise, the output context is $\langle\varnothing, h, t, t'\rangle$ (absurd context, update stops).

## C. FROM ENGLISH TO $UP_2$

X to Al:

$c_0 = \langle \sigma_0, [\![al]\!], t_0, t_{-1} \rangle$

where $\sigma_0 = \{\langle \varnothing, \varnothing \rangle\}$ & $t_{-1} \cap \text{TOD } t_0 < t_0$

**(2)** i. Today Ed helped Jo (a bit).

$$
\begin{aligned}
c_1 &:= \langle \sigma_0, [\![al]\!], t_0, t_{-1}\rangle[\mathbf{TP}\mathit{hlp\ jo\ ed}]^+ \\
&= \langle \sigma_0, [\![al]\!], t_0, t_{-1} \cap \text{TOD } t_0\rangle[\mathbf{P}\mathit{hlp\ jo\ ed}]^+ && \mathbf{T} \\
&= \langle \sigma_0, [\![al]\!], t_0, t_{-1} \cap \text{TOD } t_0 \cap \text{PST } t_0\rangle[\mathit{hlp\ jo\ ed}]^+ && \mathbf{P} \\
&= \langle (\sigma_0 \uparrow_{t0} \langle \text{EXE}([\![hlp]\!]([\![jo]\!]), [\![ed]\!]), t_{-1} \cap \text{TOD } t_0 \cap \text{PST } t_0\rangle), && \boldsymbol{E} \\
&\qquad [\![al]\!], t_0, t_{-1} \cap \text{TOD } t_0 \cap \text{PST } t_0\rangle
\end{aligned}
$$

where

$$
\begin{aligned}
\sigma_1 &:= \sigma_0 \uparrow_{t0} \langle \text{EXE}([\![hlp]\!]([\![jo]\!]), [\![ed]\!]), t_{-1} \cap \text{TOD } t_0 \cap \text{PST } t_0\rangle \\
&= \{(s \oplus \langle \text{EXE}([\![hlp]\!]([\![jo]\!]), [\![ed]\!]), t_{-1} \cap \text{TOD } t_0 \cap \text{PST } t_0\rangle, \pi) | && \sigma\uparrow \\
&\qquad (s, \pi) \in \sigma_0 \ \&\ s = (s|t_0) \ \&\ t_{-1} < t_0 \\
&\qquad \checkmark(s \oplus \langle \text{EXE}([\![hlp]\!]([\![jo]\!]), [\![ed]\!]), t_{-1} \cap \text{TOD } t_0 \cap \text{PST } t_0\rangle)\} \\
&= \{(s \cup \{\langle \text{EXE}([\![hlp]\!]([\![jo]\!]), [\![ed]\!]), t_{-1} \cap \text{TOD } t_0 \cap \text{PST } t_0, 1\rangle\}, \pi) | && s\oplus \\
&\qquad (s, \pi) \in \sigma_0 \ \&\ s = (s|t_0) \ \&\ t_{-1} < t_0 \\
&\qquad \checkmark(s \cup \{\langle \text{EXE}([\![hlp]\!]([\![jo]\!]), [\![ed]\!]), t_{-1} \cap \text{TOD } t_0 \cap \text{PST } t_0, 1\rangle\})\} \\
&= \{(\{\langle \text{EXE}([\![hlp]\!]([\![jo]\!]), [\![ed]\!]), t_{-1} \cap \text{TOD } t_0 \cap \text{PST } t_0, 1\rangle\}, \varnothing)\} && \sigma_0
\end{aligned}
$$

ii. He washed up.

$$
\begin{aligned}
c_2 &:= \langle \sigma_1, [\![al]\!], t_0, t_{-1} \cap \text{TOD } t_0 \cap \text{PST } t_0\rangle[\mathbf{P}\mathit{wsh\ ed}]^+ \\
&= \langle \sigma_1, [\![al]\!], t_0, \text{PST } t_0\rangle[\mathit{wsh\ ed}]^+ && \mathbf{P}\text{ (oops!)} \\
&= \langle (\sigma_1 \uparrow_{t0} \langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST } t_0\rangle), [\![al]\!], t_0, \text{PST } t_0\rangle && \boldsymbol{E}
\end{aligned}
$$

where

$$
\begin{aligned}
\sigma_2 &:= \sigma_1 \uparrow_{t0} \langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST } t_0\rangle \\
&= \{(s \oplus \langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST } t_0\rangle, \pi) | && \sigma\uparrow \\
&\qquad (s, \pi) \in \sigma_1 \ \&\ s = (s|t_0) \ \&\ \text{PST } t_0 < t_0 \\
&\qquad \&\ \checkmark(s \oplus \langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST } t_0\rangle)\} \\
&= \{(s \cup \{\langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST } t_0, 1\rangle\}, \pi) | && s\oplus \\
&\qquad (s, \pi) \in \sigma_1 \ \&\ s = (s|t_0) \\
&\qquad \&\ \checkmark(s \cup \{\langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST } t_0, 1\rangle\})\} \\
&= \{(\{\langle \text{EXE}([\![hlp]\!]([\![jo]\!]), [\![ed]\!]), t_{-1} \cap \text{TOD } t_0 \cap \text{PST } t_0, 1\rangle, && \sigma_1 \\
&\qquad \langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST } t_0, 1\rangle\}, \varnothing)\}
\end{aligned}
$$

Jo to Ed:

$c_0 \ = \ \langle \sigma_0, \llbracket ed \rrbracket, t_0, t_1 \rangle$

where $\sigma_0 = \ \{\langle \varnothing, \varnothing \rangle\}$ & $t_0 < t_1$

**(3)** i. Don't go out!

$c_1 \ := \ c_0[\neg out!]^+$

$\quad = \ c_0[out!]^-$ $\hfill \neg$

$\quad = \ \langle (\sigma_0 \downarrow_{t0} \langle \llbracket out \rrbracket, t_1 \rangle), \llbracket ed \rrbracket, t_0, t_1 \rangle$ $\hfill \boldsymbol{A}$

where

$\sigma_1 \ := \ \sigma_0 \downarrow_{t0} \langle \llbracket out \rrbracket, t_1 \rangle$

$\quad = \ \{(s, \pi \ominus \langle \llbracket out \rrbracket, t_1 \rangle) | \ (s, \pi) \in \sigma_0 \ \& \ \pi = (t_0|\pi) \ \& \ t_0 < t_1\}$ $\hfill \sigma\downarrow$

$\quad = \ \{(s, \pi \cup \{\langle \llbracket out \rrbracket, t_1, \mathrm{N} \rangle\}) | \ (s, \pi) \in \sigma_0 \ \& \ \pi = (t_0|\pi) \ \& \ t_0 < t_1\}$ $\hfill \pi\ominus$

$\quad = \ \{(\varnothing, \ \{\langle \llbracket out \rrbracket, t_1, \mathrm{N} \rangle\})\}$ $\hfill \sigma_0, (t|\pi)$

ii. If you go out, I'll leave you.

$c_{11} := \langle \sigma_1, \llbracket ed \rrbracket, t_0, \mathrm{FUT}\ t_0 \rangle$

$c_2 \ := \ c_1[\mathbf{F}(out\ ed \to lv\ ed\ jo)]^+$

$\quad = \ c_{11}[out\ ed \to lv\ ed\ jo]^+$ $\hfill$ **F** (oops!)

$\quad = \ c_{11}[\neg out\ ed \vee (out\ ed \wedge lv\ ed\ jo)]^+$ $\hfill \to$

$\quad = \ \langle (c_{11}[\neg out\ ed]^+)_1 \cup (c_{11}[out\ ed \wedge lv\ ed\ jo]^+)_1, \llbracket ed \rrbracket, t_0, \mathrm{FUT}\ t_0 \rangle$ $\hfill \vee$

$\quad = \ \langle (c_{11}[out\ ed]^-)_1 \cup (c_{11}[out\ ed]^+[lv\ ed\ jo]^+)_1, \llbracket ed \rrbracket, t_0, \mathrm{FUT}\ t_0 \rangle$ $\hfill \neg, \wedge$

$\quad =: \ \langle (c_{11}[out\ ed]^-)_1 \cup (\langle \sigma_{12}, \llbracket ed \rrbracket, t_0, \mathrm{FUT}\ t_0 \rangle [lv\ ed\ jo]^+)_1,$ $\hfill \sigma_{12}$

$\qquad \llbracket ed \rrbracket, t_0, \mathrm{FUT}\ t_0 \rangle$

$\quad =: \ \langle \sigma_{21} \cup \sigma_{22}, \llbracket ed \rrbracket, t_0, \mathrm{FUT}\ t_0 \rangle$ $\hfill \sigma_{21}, \sigma_{22}$

where

$\sigma_{21} = (c_{11}[out\ ed]^-)_1$

$\quad = \ \sigma_1 \downarrow_{t0} \langle \mathrm{EXE}(\llbracket out \rrbracket, \llbracket ed \rrbracket), \mathrm{FUT}\ t_0 \rangle$ $\hfill \boldsymbol{E}, c_{11}$

$\quad = \ \{(s \ominus \langle \mathrm{EXE}(\llbracket out \rrbracket, \llbracket ed \rrbracket), \mathrm{FUT}\ t_0 \rangle, \pi) |$

$\qquad (s, \pi) \in \sigma_1 \ \& \ s = (s|t_0) \ \& \ \mathrm{FUT}\ t_0 \leq t_0 \ \& \ \ldots\}$ $\hfill \sigma\downarrow$ (oops!)

$\quad = \ \varnothing$ $\hfill \{-|-\}$

$\sigma_{12} = (c_{11}[out\ ed]^+)_1$

$\quad = \ \sigma_1 \uparrow_{t0} \langle \mathrm{EXE}(\llbracket out \rrbracket, \llbracket ed \rrbracket), \mathrm{FUT}\ t_0 \rangle$ $\hfill \boldsymbol{E}, c_{11}$

$\quad = \ \{(s \oplus \langle \mathrm{EXE}(\llbracket out \rrbracket, \llbracket ed \rrbracket), \mathrm{FUT}\ t_0 \rangle, \pi) |$

$\qquad (s, \pi) \in \sigma_1 \ \& \ s = (s|t_0) \ \& \ \mathrm{FUT}\ t_0 \leq t_0 \ \& \ \ldots\}$ $\hfill \sigma\uparrow$ (oops!)

$\quad = \ \varnothing$ $\hfill \{-|-\}$

$\sigma_{22} = (\langle \sigma_{12}, \llbracket ed \rrbracket, t_0, \mathrm{FUT}\ t_0 \rangle [lv\ ed\ jo]^+)_1$

$\quad = \ \varnothing \uparrow_{t0} \langle \mathrm{EXE}(\llbracket lv \rrbracket(\llbracket ed \rrbracket), \llbracket jo \rrbracket), \mathrm{FUT}\ t_0 \rangle$ $\hfill \boldsymbol{E}, \sigma_{12}$

$\quad = \ \varnothing$ $\hfill \sigma\uparrow$

Jo to Ed:

$c_0 \;=\; \langle \sigma_0, [\![ed]\!], t_0, t_1 \rangle$

where $\sigma_0 = \;\{\langle \varnothing, \varnothing \rangle\}\;\&\; t_0 < t_1 \cap \text{TOD}\; t_0$

**(3′)** i. Don't go out today!

$c'_1 := c_0[\mathbf{T}\neg out!]^+$

$\quad = \langle \sigma_0, [\![ed]\!], t_0, t_1 \cap \text{TOD}\; t_0 \rangle[\neg out!]^+$ **T**

$\quad = \langle \sigma_0, [\![ed]\!], t_0, t_1 \cap \text{TOD}\; t_0 \rangle[out!]^-$ ¬

$\quad = \langle (\sigma_0 \downarrow_{t0} \langle [\![out]\!], t_1 \cap \text{TOD}\; t_0 \rangle), [\![ed]\!], t_0, t_1 \cap \text{TOD}\; t_0 \rangle$ ***A***

where

$\sigma'_1 := \sigma_0 \downarrow_{t0} \langle [\![out]\!], t_1 \cap \text{TOD}\; t_0 \rangle$

$\quad = \{(s, \pi \ominus \langle [\![out]\!], t_1 \cap \text{TOD}\; t_0 \rangle)|$
$\quad\quad (s, \pi) \in \sigma_0 \;\&\; \pi = (t_0|\pi) \;\&\; t_0 < t_1 \cap \text{TOD}\; t_0\}$ $\sigma\!\downarrow$

$\quad = \{(s, \pi \cup \{\langle [\![out]\!], t_1 \cap \text{TOD}\; t_0, \text{N} \rangle\})|$
$\quad\quad (s, \pi) \in \sigma_0 \;\&\; \pi = (t_0|\pi) \;\&\; t_0 < t_1 \cap \text{TOD}\; t_0\}$ $\pi\ominus$

$\quad = \{(\varnothing, \{\langle [\![out]\!], t_1 \cap \text{TOD}\; t_0, \text{N} \rangle\})\}$ $\sigma_0, (t|\pi)$

ii. If you go out, I'll leave you.

$c'_{11} := \langle \sigma'_1, [\![ed]\!], t_0, \text{FUT}\; t_0 \rangle$

$c'_2 := c'_1[\mathbf{F}(out\; ed \rightarrow lv\; ed\; jo)]^+$

$\quad = c'_{11}[out\; ed \rightarrow lv\; ed\; jo]^+$ **F** (oops!)

$\quad = c'_{11}[\neg out\; ed \vee (out\; ed \wedge lv\; ed\; jo)]^+$ **→**

$\quad = \langle (c'_{11}[\neg out\; ed]^+)_1 \cup (c'_{11}[out\; ed \wedge lv\; ed\; jo]^+)_1, [\![ed]\!], t_0, \text{FUT}\; t_0 \rangle$ **∨**

$\quad = \langle (c'_{11}[out\; ed]^-)_1 \cup (c'_{11}[out\; ed]^+[lv\; ed\; jo]^+)_1, [\![ed]\!], t_0, \text{FUT}\; t_0 \rangle$ **¬, ∧**

$\quad =: \langle (c'_{11}[out\; ed]^-)_1 \cup (\langle \sigma'_{12}, [\![ed]\!], t_0, \text{FUT}\; t_0 \rangle[lv\; ed\; jo]^+)_1,$ $\sigma'_{12}$
$\quad\quad [\![ed]\!], t_0, \text{FUT}\; t_0 \rangle$

$\quad =: \langle \sigma'_{21} \cup \sigma'_{22}, [\![ed]\!], t_0, \text{FUT}\; t_0 \rangle$ $\sigma'_{21}, \sigma'_{22}$

where

$\sigma'_{21} = (c'_{11}[out\; ed]^-)_1$

$\quad = \{(s \ominus \langle \text{EXE}([\![out]\!], [\![ed]\!]), \text{FUT}\; t_0 \rangle, \pi)|$
$\quad\quad (s, \pi) \in \sigma_1 \;\&\; s = (s|t_0) \;\&\; \text{FUT}\; t_0 \le t_0 \;\&\; \ldots\}$ ***E***, $c'_{11}$, $\sigma\!\downarrow$ (oops!)

$\quad = \varnothing$ {–|–}

$\sigma'_{12} = (c_{11}[out\; ed]^+)_1$

$\quad = \{(s \oplus \langle \text{EXE}([\![out]\!], [\![ed]\!]), \text{FUT}\; t_0 \rangle, \pi)|$
$\quad\quad (s, \pi) \in \sigma_1 \;\&\; s = (s|t_0) \;\&\; \text{FUT}\; t_0 \le t_0 \;\&\; \ldots\}$ ***E***, $c'_{11}$, $\sigma\!\uparrow$ (oops!)

$\quad = \varnothing$ {–|–}

$\sigma'_{22} = (\langle \sigma'_{12}, [\![ed]\!], t_0, \text{FUT}\; t_0 \rangle[lv\; ed\; jo]^+)_1$

$\quad = \varnothing$ ***E***, $\sigma\!\uparrow$, $\sigma'_{12}$

X to Al:

$c_0 \;=\; \langle \sigma_0, \llbracket al \rrbracket, t_0, t_1 \rangle$

where $\sigma_0 = \{\langle \varnothing, \varnothing \rangle\}$

**(4)** i. Ed either did or did not wash up.

$$
\begin{array}{lll}
c_1 & := \langle \sigma_0, \llbracket al \rrbracket, t_0, t_1 \rangle [\mathbf{P}(\mathit{wsh\ ed} \vee \neg \mathit{wsh\ ed})]^{+} & \\
& = \langle \sigma_0, \llbracket al \rrbracket, t_0, \text{PST } t_0 \rangle [(\mathit{wsh\ ed} \vee \neg \mathit{wsh\ ed})]^{+} & \mathbf{P} \\
& = \langle (\langle \sigma_0, t_0, \text{PST } t_0 \rangle [\mathit{wsh\ ed}]^{+})_1 \cup (\langle \sigma_0, t_0, \text{PST } t_0 \rangle [\neg \mathit{wsh\ ed})]^{+})_1, & \\
& \quad \llbracket al \rrbracket, t_0, \text{PST } t_0 \rangle & \vee \\
& = \langle (\langle \sigma_0, t_0, \text{PST } t_0 \rangle [\mathit{wsh\ ed}]^{+})_1 \cup (\langle \sigma_0, t_0, \text{PST } t_0 \rangle [\mathit{wsh\ ed})]^{-})_1, & \\
& \quad \llbracket al \rrbracket, t_0, \text{PST } t_0 \rangle & \neg \\
& = \langle (\sigma_0 \uparrow_{t0} \langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0 \rangle) & \\
& \quad\; \cup (\sigma_0 \downarrow_{t0} \langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0 \rangle) & \\
& \quad \llbracket al \rrbracket, t_0, \text{PST } t_0 \rangle & \boldsymbol{E}
\end{array}
$$

where

$$
\begin{array}{lll}
\sigma_1 & := (\sigma_0 \uparrow_{t0} \langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0 \rangle) & \\
& \quad \cup (\sigma_0 \downarrow_{t0} \langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0 \rangle) & \\
& = \{(s \oplus \langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0 \rangle, \pi) | & \sigma\uparrow, \sigma\downarrow \\
& \quad (s, \pi) \in \sigma_0 \;\&\; s = (s|t_0) \;\&\; \text{PST } t_0 < t_0 & \\
& \quad \&\; \checkmark(s \oplus \langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0 \rangle)\} & \\
& \quad \cup \{(s \ominus \langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0 \rangle, \pi) | & \\
& \qquad (s, \pi) \in \sigma_0 \;\&\; s = (s|t_0) \;\&\; \text{PST } t_0 < t_0 & \\
& \qquad \&\; \checkmark(s \ominus \langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0 \rangle)\} & \\
& = \{(s \cup \{\langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0, 1 \rangle\}, \pi) | & s\oplus, s\ominus \\
& \quad (s, \pi) \in \sigma_0 \;\&\; s = (s|t_0) \;\&\; t_{-1} < t_0 & \\
& \quad \&\; \checkmark(s \cup \{\langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0, 1 \rangle\})\} & \\
& \quad \cup \{(s \cup \{\langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0, 0 \rangle\}, \pi) | & \\
& \qquad (s, \pi) \in \sigma_0 \;\&\; s = (s|t_0) \;\&\; t_{-1} < t_0\} & \\
& \qquad \&\; \checkmark(s \cup \{\langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0, 0 \rangle\})\} & \\
& = \{(\{\langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0, 1 \rangle\}, \varnothing)\} & \sigma_0 \\
& \quad \cup \{(\{\langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0, 0 \rangle\}, \varnothing)\} & \\
& = \{(\{\langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0, 1 \rangle\}, \varnothing), & \\
& \qquad (\{\langle \text{EXE}(\llbracket wsh \rrbracket, \llbracket ed \rrbracket), \text{PST } t_0, 0 \rangle\}, \varnothing)\} & \cup
\end{array}
$$

ii. If he did, pay him!

$c_{11} := \langle \sigma_1, [\![al]\!], t_0, \text{PST } t_0 \rangle$

$c_2 := c_1[\mathbf{P}(wsh\ ed \rightarrow pay\ ed!)]^+$

$= c_{11}[wsh\ ed \rightarrow pay\ ed!]^+$ — **P**

$= c_{11}[\neg wsh\ ed \vee (wsh\ ed \wedge pay\ ed!)]^+$ — $\rightarrow$

$= \langle (c_{11}[\neg wsh\ ed]^+)_1 \cup (c_{11}[wsh\ ed \wedge pay\ ed\ !]^+)_1, [\![al]\!], t_0, \text{PST } t_0 \rangle$ — $\vee$

$= \langle (c_{11}[wsh\ ed]^-)_1 \cup (c_{11}[wsh\ ed]^+[pay\ ed!]^+)_1, [\![al]\!], t_0, \text{PST } t_0 \rangle$ — $\neg, \wedge$

$=: \langle (c_{11}[wsh\ ed]^-)_1 \cup (\langle \sigma'_1, [\![al]\!], t_0, \text{PST } t_0 \rangle [pay\ ed!]^+)_1, [\![al]\!], t_0, \text{PST } t_0 \rangle$ — $\sigma'_1$

$=: \langle \sigma_2 \cup \sigma'_2, [\![al]\!], t_0, \text{PST } t_0 \rangle$ — $\sigma_2, \sigma'_2$

$= \langle \sigma_2, [\![al]\!], t_0, \text{PST } t_0 \rangle$ — see below

where

$\sigma_2 := (c_{11}[wsh\ ed]^-)_1$

$= \sigma_1 \downarrow_{t0} \langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST } t_0 \rangle$ — $\boldsymbol{E}, c_{11}$

$= \{(s \ominus \langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST } t_0 \rangle, \pi) |$
$(s, \pi) \in \sigma_1 \ \&\ s = (s|t_0) \ \&\ \text{PST } t_0 \leq t_0$
$\&\ \checkmark(s \ominus \langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST } t_0 \rangle \}$ — $\sigma\!\downarrow$

$= \{(\{\langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST } t_0, 0 \rangle\}, \varnothing)\}$ — $s\ominus, \sigma_1$

$\sigma'_1 := (c_{11}[wsh\ ed]^+)_1$

$\sigma'_2 := (\langle \sigma'_1, [\![al]\!], t_0, \text{PST } t_0 \rangle [pay\ ed!]^+)_1$

$= \sigma'_1 \uparrow_{t0} \langle [\![pay]\!]([\![ed]\!]), \text{PST } t_0 \rangle$ — $\boldsymbol{A}$

$= \{(s, \pi \oplus \langle [\![pay]\!]([\![ed]\!]), \text{PST } t_0 \rangle) |$
$(s, \pi) \in \sigma'_1 \ \&\ \pi = (t_0|\pi) \ \&\ t_0 < \text{PST } t_0\}$ — $\sigma\!\uparrow$

$= \varnothing$ — PST

iii. If he didn't, don't!

$c_3$ := $c_2[\mathbf{P}(\neg wsh\ ed \to \neg pay\ ed!)]^+$
= $c_2[\neg wsh\ ed \to \neg pay\ ed!]^+$ **P**
= $c_2[\neg\neg wsh\ ed \vee (\neg wsh\ ed \wedge \neg pay\ ed!)]^+$ **→**
= $\langle (c_2[\neg\neg wsh\ ed]^+)_1 \cup (c_2[\neg wsh\ ed \wedge \neg pay\ ed\ !]^+)_1,$ **∨**
$[\![al]\!], t_0, \text{PST}\ t_0\rangle$
= $\langle (c_2[\neg wsh\ ed]^-)_1 \cup (c_2[\neg wsh\ ed]^+[\neg pay\ ed!]^+)_1,$ **¬, ∧**
$[\![al]\!], t_0, \text{PST}\ t_0\rangle$
= $\langle (c_2[wsh\ ed]^+)_1 \cup (c_2[wsh\ ed]^-[pay\ ed!]^-)_1,$ **¬**
$[\![al]\!], t_0, \text{PST}\ t_0\rangle$
=: $\langle (c_2[wsh\ ed]^+)_1 \cup (\langle \sigma'_2, [\![al]\!], t_0, \text{PST}\ t_0\rangle[pay\ ed!]^-)_1,$ $\sigma'_2$
$[\![al]\!], t_0, \text{PST}\ t_0\rangle$
=: $\langle \sigma_3 \cup \sigma'_3, [\![al]\!], t_0, \text{PST}\ t_0\rangle$ $\sigma_3, \sigma'_3$
= $\langle \varnothing, [\![al]\!], t_0, \text{PST}\ t_0\rangle$ see below

where

$\sigma_3$ := $(c_2[wsh\ ed]^+)_1$
= $\sigma_2 \uparrow_{t0} \langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST}\ t_0\rangle$ ***E***, $c_2$
= $\{(s \oplus \langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST}\ t_0\rangle, \pi) |$
$(s, \pi) \in \sigma_2\ \&\ s = (s|t_0)\ \&\ \text{PST}\ t_0 \leq t_0$
$\&\ \checkmark(s \oplus \langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST}\ t_0\rangle\}$ $\sigma\uparrow$
= $\{(s \cup \{\langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST}\ t_0, 1\rangle\}, \pi) |$
$(s, \pi) \in \sigma_2\ \&\ s = (s|t_0)\ \&\ \text{PST}\ t_0 \leq t_0$
$\&\ \checkmark(\ s \cup \{\langle \text{EXE}([\![wsh]\!], [\![ed]\!]), \text{PST}\ t_0, 1\rangle\})\}$ $s\oplus$
= $\varnothing$ $\sigma_2$

$\sigma'_2$ := $(c_2[wsh\ ed]^-)_1$

$\sigma'_3$ := $(\langle \sigma'_2, [\![al]\!], t_0, \text{PST}\ t_0\rangle[pay\ ed!]^-)_1$
= $\sigma'_2 \downarrow_{t0} \langle [\![pay]\!]([\![ed]\!]), \text{PST}\ t_0\rangle$ ***A***
= $\{(s, \pi \ominus \langle [\![pay]\!]([\![ed]\!]), \text{PST}\ t_0\rangle) |$ $\sigma\downarrow$
$(s, \pi) \in \sigma'_2\ \&\ \pi = (t_0|\pi)\ \&\ t_0 < \text{PST}\ t_0\}$
= $\varnothing$ PST

# Modals and imperatives: Appendix to Susan's presentation of Aloni 2007

• LOGIC OF ALTERNATIVES (see cut-&-paste handout)

Notation: $s[\![\phi]\!]^{M,g} := \{w|\ M, w, s \models_g \phi\}$

$[\![\phi]\!]^{M,g} := \{w|\ \exists s\!: M, w, s \models_g \phi\}$

• EXISTENTIAL GENERALIZATION

(1) Somebody smokes.

(1a) $\exists x\, A(x)$

(1b) $\exists p(p \wedge p = \exists x\, A(x))$

(1c) $\exists p(p \wedge \exists x(p = A(x))$

FACT 1a.$\models$

$M, w, s \models_g \exists x\, A(x)$ iff $[\![A]\!](w) \neq \varnothing$

PROOF. (1) iff (5)

1. $M, w, s \models_g \exists x\, A(x)$
2. $\exists d \in D$: $M, w, s \models_{g[x/d]} A(x)$ — D2.$\exists x$
3. $\exists d \in D$: $g[x/d](x) \in [\![A]\!](w)$ — D2.$R$
4. $\exists d \in D$: $d \in [\![A]\!](w)$ — $g[x/d]$
5. $[\![A]\!](w) \neq \varnothing$ — $\varnothing$

FACT 1a.$s[\![\cdot]\!]$

$s[\![\exists x\, A(x)]\!]^{M,g}$

$= \{w|\ [\![A]\!](w) \neq \varnothing\}$

FACT 1a.$[\![\cdot]\!]$

$[\![\exists x\, A(x)]\!]^{M,g}$

$= \{w|\ [\![A]\!](w) \neq \varnothing$

FACT 1a.ALT[2]

$\text{ALT}(\exists x\, A(x))_{M,g} = \{[\![\exists x\, A(x)]\!]^{M,g}\}\backslash\{\varnothing\}$

PROOF

1. $\text{ALT}(\exists x\, A(x))_{M,g}$
2. $\{\{w|\ M, w, s \models_g \exists x\, A(x)\}|\ s \in P^{n(\exists x\, A(x))}\}\backslash\{\varnothing\}$ — D4
3. $\{\{w|\ M, w, s \models_g \exists x\, A(x)\}|\ s \in P^{0}\}\backslash\{\varnothing\}$ — D1
4. $\{\{w|\ [\![A]\!](w) \neq \varnothing\}\}\backslash\{\varnothing\}$ — F1a.$\models$
5. $\{[\![\exists x\, A(x)]\!]^{M,g}\}\backslash\{\varnothing\}$ — F1a.$[\![\cdot]\!]$

[2] Aloni's def. 4 says: '…\∅'. But $A\backslash\varnothing = A$, for any set $A$. I think she means '…\{∅}'.

(1b) $\exists p(p \wedge p = \exists x\, A(x))$

FACT 1b.|=
$M, w, s \models_g \exists p(p \wedge p = \exists x\, A(x))$
iff $\exists s'$: $s = (\llbracket \exists x\, A(x) \rrbracket^{M,g})s'$ & $w \in \llbracket \exists x\, A(x) \rrbracket^{M,g}$

PROOF. (1) iff (5)

1. $M, w, s \models_g \exists p(p \wedge p = \exists x\, A(x))$
2. $\exists q, s'$: $s = qs'$ — D2.$\exists p$
   & $M, w, s' \models_{g[p/q]} p \wedge p = \exists x\, A(x)$
3. $\exists q, s'$: $s = qs'$ — D2.$\wedge$, D1
   & $\exists r'$: $s' = r'$
      & $M, w, r' \models_{g[p/q]} p$
      & $M, w, r' \models_{g[p/q]} p = \exists x\, A(x)$
4. $\exists q, s'$: $s = qs'$ — $s' = r'$
   & $M, w, s' \models_{g[p/q]} p$
   & $M, w, s' \models_{g[p/q]} p = \exists x\, A(x)$
5. $\exists q, s'$: $s = qs'$ — D2.$p$, =
   & $w \in g[p/q](p)$
   & $\exists r \in P^{n(p)} \exists r' \in P^{n(\exists x\, A(x))} \forall v$: $M, v, rs' \models_{g[p/q]} p$
      iff $M, v, r's' \models_{g[p/q]} \exists x\, A(x)$
6. $\exists q, s'$: $s = qs'$ — D1, {–|–}
   & $w \in g[p/q](p)$
   & $\{v \mid M, v, s' \models_{g[p/q]} p\} = \{v \mid M, v, s' \models_{g[p/q]} \exists x\, A(x)\}$
7. $\exists q, s'$: $s = qs'$ & $w \in q$ — D2.$p$, $g$[
   & $\{v \mid v \in q\} = \{v \mid M, v, s' \models_{g[p/q]} \exists x\, A(x)\}$
8. $\exists q, s'$: $s = qs'$ & $w \in q$ — D2.$\exists x$
   & $q = \{v \mid \exists d \in D(M, v, s' \models_{g[p/q][x/d]} A(x))\}$
9. $\exists q, s'$: $s = qs'$ & $w \in q$ — D2.$R$, $g$[
   & $q = \{v \mid \exists d \in D$: $d \in \llbracket A \rrbracket(v))\}$
10. $\exists q, s'$: $s = qs'$ & $w \in q$ — simplify
   & $q = \{v \mid \llbracket A \rrbracket(v) \neq \emptyset\}$
11. $\exists q, s'$: $s = qs'$ & $w \in q$ & $q = \llbracket \exists x\, A(x) \rrbracket^{M,g}$ — F1a.$\llbracket \cdot \rrbracket$
12. $\exists s'$: $s = (\llbracket \exists x\, A(x) \rrbracket^{M,g})s'$ & $w \in \llbracket \exists x\, A(x) \rrbracket^{M,g}$ — elimin. $q$

FACT 1b.$s[\![\cdot]\!]$

$s[\![\exists p(p \wedge p = \exists x\, A(x))]\!]^{M,g}$

| | | |
|---|---|---|
| $=$ | $[\![\exists x\, A(x)]\!]^{M,g}$ | if $\exists s'$: $s = ([\![\exists x\, A(x)]\!]^{M,g})s'$ |
| | $\varnothing$ | otherwise |

PROOF

$s[\![\exists p(p \wedge p = \exists x\, A(x))]\!]^{M,g}$

| | | | |
|---|---|---|---|
| $=$ | $\{w \mid M, w, s \models_g \exists p(p \wedge p = \exists x\, A(x))\}$ | | $s[\![\cdot]\!]$ |
| $=$ | $\{w \mid \exists s'$: $s = ([\![\exists x\, A(x)]\!]^{M,g})s'\ \&\ w \in [\![\exists x\, A(x)]\!]^{M,g}\}$ | | F1b.$\models$ |
| $=$ | $[\![\exists x\, A(x)]\!]^{M,g}$ | if $\exists s'$: $s = ([\![\exists x\, A(x)]\!]^{M,g})s'$ | $\{-\|-\}$ |
| | $\varnothing$ | otherwise | |

FACT 1b.$[\![\cdot]\!]$

$[\![\exists p(p \wedge p = \exists x\, A(x))]\!]^{M,g}$

$= \quad [\![\exists x\, A(x)]\!]^{M,g}$

PROOF

$[\![\exists p(p \wedge p = \exists x\, A(x))]\!]^{M,g}$

| | | |
|---|---|---|
| $=$ | $\{w \mid \exists s$: $M, w, s \models_g \exists p(p \wedge p = \exists x\, A(x))\}$ | $[\![\cdot]\!]$ |
| $=$ | $\{w \mid \exists s, s'$: $s = ([\![\exists x\, A(x)]\!]^{M,g})s'\ \&\ w \in [\![\exists x\, A(x)]\!]^{M,g}\}$ | F1b.$\models$ |
| $=$ | $[\![\exists x\, A(x)]\!]^{M,g}$ | $\{-\|-\}$ |

FACT 1b.ALT

$\text{ALT}(\exists p(p \wedge p = \exists x\, A(x)))_{M,g}$

$= \quad \{[\![\exists x\, A(x)]\!]^{M,g}\}\backslash\{\varnothing\}$

PROOF

| | | |
|---|---|---|
| 1. | $\text{ALT}(\exists p(p \wedge p = \exists x\, A(x)))_{M,g}$ | |
| 2. | $\{\{w \mid M, w, s \models_g \exists p(p \wedge p = \exists x\, A(x))\} \mid s \in P^1\}\backslash\{\varnothing\}$ | D4, D1 |
| 3. | $\{s[\![\exists p(p \wedge p = \exists x\, A(x))]\!]^{M,g} \mid s \in P^1\}\backslash\{\varnothing\}$ | $s[\![\cdot]\!]$ |
| 4. | $\{[\![\exists x\, A(x)]\!]^{M,g}\}\backslash\{\varnothing\}$ | F1b.$s[\![\cdot]\!]$, $[\![\cdot]\!]$ |

(1c) $\exists p(p \wedge \exists x(p = A(x)))$

FACT 1c.|=
$M, w, s \models_g \exists p(p \wedge \exists x(p = A(x)))$
iff $\exists d \exists s'$: $s = (\llbracket \exists x\ A(x) \rrbracket^{M, g[x/d]})s'$ & $w \in \llbracket \exists x\ A(x) \rrbracket^{M, g[x/d]}$

PROOF. (1) iff (5)

| | | |
|---|---|---|
| 1. | $M, w, s \models_g \exists p(p \wedge \exists x(p = A(x)))$ | |
| 2. | $\exists q, s'$: $s = qs'$ | D2.$\exists p$ |
| | & $M, w, s' \models_{g[p/q]} p \wedge \exists x(p = A(x))$ | |
| 3. | $\exists q, s'$: $s = qs'$ | D2.$\wedge$, D1 |
| | & $\exists r'$: $s' = r'$ | |
| | & $M, w, r' \models_{g[p/q]} p$ | |
| | & $M, w, r' \models_{g[p/q]} \exists x(p = A(x))$ | |
| 4. | $\exists q, s'$: $s = qs'$ | $s' = r'$ |
| | & $M, w, s' \models_{g[p/q]} p$ | |
| | & $M, w, s' \models_{g[p/q]} \exists x(p = A(x))$ | |
| 5. | $\exists q, s'$: $s = qs'$ | D2.$p$, $\exists$ |
| | & $w \in g[p/q](p)$ | |
| | & $\exists d \in D$: $M, w, s' \models_{g[p/q][x/d]} p = A(x)$ | |
| 6. | $\exists q, s'$: $s = qs'$ | $g[$, D2.= |
| | & $w \in q$ | D1 |
| | & $\exists d \in D \forall v$: $M, v, s' \models_{g[p/q][x/d]} p$ | |
| | iff $M, v, s' \models_{g[p/q][x/d]} A(x)$ | |
| 7. | $\exists q, s'$: $s = qs'$ | D2.$p$, $R$, |
| | & $w \in q$ | $g[$ |
| | & $\exists d \in D \forall v$: $v \in q$ iff $d \in \llbracket A \rrbracket(v)$ | |
| 8. | $\exists d, q, s'$: $s = qs'$ & $w \in q$ & $q = \{v \mid d \in \llbracket A \rrbracket(v)\}$ | rearrange |
| 9. | $\exists d, q, s'$: $s = qs'$ & $w \in q$ & $q = \llbracket A(x) \rrbracket^{M, g[x/d]}$ | F1a.$\llbracket\cdot\rrbracket$ |
| 10. | $\exists d, s'$: $s = (\llbracket \exists x\ A(x) \rrbracket^{M, g[x/d]})s'$ & $w \in \llbracket A(x) \rrbracket^{M, g[x/d]}$ | elimin. $q$ |

FACT 1c.$s\llbracket\cdot\rrbracket$
$s\llbracket \exists p(p \wedge \exists x(p = A(x))) \rrbracket^{M, g}$
= $\{q\}$ if $\exists d, s'$: $s = qs'$ & $q = \llbracket A(x) \rrbracket^{M, g[x/d]}$
$\varnothing$ if $\neg\exists d, s'$: $s = (\llbracket A(x) \rrbracket^{M, g[x/d]})s'$

FACT 1b.$\llbracket\cdot\rrbracket$
$\llbracket \exists p(p \wedge \exists x(p = A(x))) \rrbracket^{M, g}$
= $\llbracket \exists x\ A(x) \rrbracket^{M, g}$

FACT 1b.ALT
ALT$(\exists p(p \wedge \exists x(p = A(x))))_{M,g}$
$= \quad \{\{[\![A(x)]\!]^{M, g[x/d]}\} \mid d \in D\} \backslash \{\varnothing\}$

PROOF

1. ALT$(\exists p(p \wedge \exists x(p = A(x)))_{M,g}$
2. $\{\{w \mid M, w, s \models_g \exists p(p \wedge \exists x(p = A(x))\} \mid s \in P^1\} \backslash \{\varnothing\}$ — D4, D1
3. $\{s[\![\exists p(p \wedge \exists x(p = A(x)))]\!]^{M,g} \mid s \in P^1\} \backslash \{\varnothing\}$ — $s[\![\cdot]\!]$
4. $\{\{[\![A(x)]\!]^{M, g[x/d]}\} \mid d \in D\} \backslash \{\varnothing\}$ — F1c.$s[\![\cdot]\!]$, $[\![\cdot]\!]$

• DISJUNCTION

(2) Michael went to the beach or he went to the cinema.
(2a) $b \vee c$ (2b) $\exists p(p \wedge p = b \vee c)$
(2c) $\exists p(p \wedge (p = b \vee p = c))$

FACT 2a.$\models$
$M, w, s \models_g b \vee c$
iff $w \in [\![b]\!]^M \cup [\![c]\!]^M$

PROOF. (1) iff (4)

1. $M, w, s \models_g b \vee c$
2. $M, w, s \models_g b$ or $M, w, s \models_g c$ — D2.$\vee$, D1
3. $w \in [\![b]\!]$ or $w \in [\![c]\!]$ — D2.$R$
4. $w \in [\![b]\!] \cup [\![c]\!]$ — $\cup$

FACT 2a.$s[\![\cdot]\!]$
$s[\![b \vee c]\!]^{M,g}$
$= \quad [\![b]\!] \cup [\![c]\!]$

FACT 2a.$[\![\cdot]\!]$
$[\![b \vee c]\!]^{M,g}$
$= \quad [\![b]\!] \cup [\![c]\!]$

FACT 2a.ALT
ALT$(b \vee c)_{M,g}$
$= \quad \{[\![b \vee c]\!]^{M,g}\} \backslash \{\varnothing\}$

PROOF

1. ALT$(b \vee c)_{M,g}$
2. $\{\{w \mid M, w, s \models_g b \vee c\} \mid s \in P^0\} \backslash \{\varnothing\}$ — D4, D1
4. $\{\{w \mid w \in [\![b]\!] \cup [\![c]\!]\}\} \backslash \{\varnothing\}$ — F2a.$\models$
5. $\{[\![b \vee c]\!]^{M,g}\} \backslash \{\varnothing\}$ — F2a.$[\![\cdot]\!]$

(2c) $\exists p(p \wedge (p = b \vee p = c))$

FACT 2c.|=
$M, w, s \models_g \exists p(p \wedge (p = b \vee p = c))$
iff $\exists q \exists s'$: $s = qs'$ & $w \in q$ & $q \in \{[\![b]\!], [\![c]\!]\}$

PROOF. (1) iff (5)

| | | |
|---|---|---|
| 1. | $M, w, s \models_g \exists p(p \wedge (p = b \vee p = c))$ | |
| 2. | $\exists q, s'$: $s = qs'$ | D2.$\exists p$ |
| | & $M, w, s' \models_{g[p/q]} p \wedge (p = b \vee p = c)$ | |
| 3. | $\exists q, s'$: $s = qs'$ | D2.$\wedge$, D1 |
| | & $\exists r'$: $s' = r'$ | |
| | & $M, w, r' \models_{g[p/q]} p$ | |
| | & $M, w, r' \models_{g[p/q]} p = b \vee p = c$ | |
| 4. | $\exists q, s'$: $s = qs'$ | $s' = r'$ |
| | & $M, w, s' \models_{g[p/q]} p$ | |
| | & $M, w, s' \models_{g[p/q]} p = b \vee p = c$ | |
| 5. | $\exists q, s'$: $s = qs'$ & $w \in q$ | D2.$p$, $g$[ |
| | & ($M, w, s' \models_{g[p/q]} p = b$ or $M, w, s' \models_{g[p/q]} p = c$) | D2.$\vee$, D1 |
| 6. | $\exists q, s'$: $s = qs'$ & $w \in q$ | D2.=, {-\|-} |
| | & $\{v \mid M, v, s' \models_{g[p/q]} p\} = \{v \mid M, v, s' \models b\}$ | |
| | or $\{v \mid M, v, s' \models_{g[p/q]} p\} = \{v \mid M, v, s' \models c\}$ | |
| 7. | $\exists q, s'$: $s = qs'$ & $w \in q$ | D2.$p$, $R$ |
| | & $\{v \mid v \in q\} = \{v \mid v \in [\![b]\!]\}$ | |
| | or $\{v \mid v \in q\} = \{v \mid v \in [\![c]\!]\}$ | |
| 8. | $\exists q, s'$: $s = qs'$ & $w \in q$ & ($q = [\![b]\!]$ or $q = [\![c]\!]$) | {-\|-} |
| 9. | $\exists q, s'$: $s = qs'$ & $w \in q$ & $q \in \{[\![b]\!], [\![c]\!]\}$ | simplify |

FACT 2c.$s[\![\cdot]\!]$
$s[\![\exists p(p \wedge (p = b \vee p = c))]\!]^{M,g}$
$= \{q\}$ if $\exists s'$: $s = qs'$ & $q \in \{[\![b]\!], [\![c]\!]\}$
$\ \ \ \varnothing$ if $\neg\exists s'$: $s = qs'$ & $q \in \{[\![b]\!], [\![c]\!]\}$

FACT 2c.$[\![\cdot]\!]$
$[\![\exists p(p \wedge \exists x(p = a \vee p = c))]\!]^{M,g}$
$= [\![b \vee c]\!]^{M,g}$

FACT 2c.ALT
ALT$(\exists p(p \wedge (p = b \vee p = c)))_{M,g}$
$= \{[\![b]\!], [\![c]\!]\}$

• NEGATION

(3) Michael didn't go either to the beach or to the cinema.
(3c) $\neg\exists p(p \wedge (p = b \vee p = c))$

FACT 3c.|=
$M, w, s \models_g \neg\exists p(p \wedge (p = b \vee p = c))$
iff $w \notin [\![b \vee c]\!]$

PROOF. (1) iff (12)

| | | |
|---|---|---|
| 1. | $M, w, s \models_g \neg\exists p(p \wedge (p = b \vee p = c))$ | |
| 2. | $\neg\exists q \in P^1: M, w, qs \models_g \exists p(p \wedge (p = b \vee p = c))$ | D2.¬, D1 |
| 3. | $\neg\exists q: M, w, s \models_{g[p/q]} p \wedge (p = b \vee p = c)$ | D2.∃ |
| 4. | $\neg\exists q: M, w, s \models_{g[p/q]} p$ & $M, w, s \models_{g[p/q]} p = b \vee p = c$ | D2.∧, D1 |
| 5. | $\neg\exists q: w \in q$ & $(M, w, s \models_{g[p/q]} p = b$ or $M, w, s \models_{g[p/q]} p = c)$ | D2.$p$, $g$[ ; D2.∨, D1 |
| 6. | $\neg\exists q: w \in q$ & $\{v \mid M, v, s \models_{g[p/q]} p\} = \{v \mid M, v, s \models b\}$ or $\{v \mid M, v, s \models_{g[p/q]} p\} = \{v \mid M, v, s \models c\}$ | D2.=, {-\|-} |
| 7. | $\neg\exists q: w \in q$ & $\{v \mid v \in q\} = \{v \mid v \in [\![b]\!]\}$ or $\{v \mid v \in q\} = \{v \mid v \in [\![c]\!]\}$ | D2.$p$, $R$ |
| 8. | $\neg\exists q: w \in q$ & $(q = [\![b]\!]$ or $q = [\![c]\!])$ | {-\|-} |
| 9. | $\neg\exists q: w \in q$ & $q \in \{[\![b]\!], [\![c]\!]\}$ | simplify |
| 10. | $w \notin [\![b]\!]$ & $w \notin [\![c]\!]$ | simplify |
| 11. | $w \notin [\![b]\!] \cup [\![c]\!]$ | simplify |
| 12. | $w \notin [\![b \vee c]\!]^{M,g}$ | F2a.$[\![\cdot]\!]$ |

FACT 3c.($s$)$[\![\cdot]\!]$
$s[\![\neg\exists p(p \wedge (p = b \vee p = c))]\!]^{M,g}$
$= [\![\neg\exists p(p \wedge (p = b \vee p = c))]\!]^{M,g}$
$= W \setminus [\![b \vee c]\!]^{M,g}$

FACT 3c.ALT.
ALT$(\neg\exists p(p \wedge (p = b \vee p = c)))_{M,g}$
$= \{W \setminus [\![b \vee c]\!]^{M,g}\}$

• MODALS (free choice readings)

(4) You may go to the beach or go to the cinema.
(4c) MAY $\exists p(p \wedge (p = b \vee p = c))$

FACT 4c.|=
$M, w, s \models_g$ MAY $\exists p(p \wedge (p = b \vee p = c))$
iff $(\exists v \in [\![b]\!]: wRv)$ & $(\exists v \in [\![c]\!]: wRv)$

PROOF. (1) iff (4)

| | | |
|---|---|---|
| 1. | $M, w, s \models_g$ MAY $\exists p(p \wedge (p = b \vee p = c))$ | |
| 2. | $\forall q \in \text{ALT}(\exists p(p \wedge (p = b \vee p = c)))_{M,g} \exists v \in q: wRv$ | D5.MAY |
| 3. | $\forall q \in \{[\![b]\!], [\![c]\!]\} \exists v \in q: wRv$ | F2c.ALT |
| 4. | $(\exists v \in [\![b]\!]: wRv)$ & $(\exists v \in [\![c]\!]: wRv)$ | simplify |

FACT 4c.$[\![\cdot]\!]$
$[\![\text{MAY } \exists p(p \wedge (p = b \vee p = c))]\!]^{M,g}$
= $\{w \mid (\exists v \in [\![b]\!]: wRv)$ & $(\exists v \in [\![c]\!]: wRv)\}$

Compare:
(4d) $\neg$MUST $\neg\exists p(p \wedge (p = b \vee p = c))$

FACT 4d.|=
$M, w, s \models_g \neg$MUST $\neg\exists p(p \wedge (p = b \vee p = c))$
iff $\exists v: wRv$ & $v \in [\![b \vee c]\!]$

(5) You must go to the beach or go to the cinema.
(5c) MUST $\exists p(p \wedge (p = b \vee p = c))$

FACT 5c.|=
$M, w, s \models_g$ MUST $\exists p(p \wedge (p = b \vee p = c))$
iff $(\forall v: wRv \rightarrow v \in [\![b]\!])$ or $(\forall v: wRv \rightarrow v \in [\![c]\!])$

PROOF. (1) iff (4)

| | | |
|---|---|---|
| 1. | $M, w, s \models_g$ MUST $\exists p(p \wedge (p = b \vee p = c))$ | |
| 2. | $\exists q \in \text{ALT}(\exists p(p \wedge (p = b \vee p = c)))_{M,g} \forall v: wRv \rightarrow v \in q$ | D5.MUST |
| 3. | $\exists q \in \{[\![b]\!], [\![c]\!]\} \forall v: wRv \rightarrow v \in q$ | F2c.ALT |
| 4. | $(\forall v: wRv \rightarrow v \in [\![b]\!])$ or $(\forall v: wRv \rightarrow v \in [\![c]\!])$ | simplify |

FACT 5c.$[\![\cdot]\!]$
$[\![\text{MUST } \exists p(p \wedge (p = b \vee p = c))]\!]^{M,g}$
= $\{w \mid (\forall v: wRv \rightarrow v \in [\![b]\!])$ or $(\forall v: wRv \rightarrow v \in [\![c]\!])\}$

# Free Choice Permission (Kamp 1973)

## 1 INTRODUCTION

• ENTAILMENT PUZZLE
In context A disjunction *entails* (permits more than) either disjunct, e.g.

A: Father to Michael:
(1) You may go to the beach or go to the cinema.
(2) You may go to the beach.

But in context B disjunction is *entailed* by (asserts less than) either disjunct

B: Michael's little sister in response to his question: 'What did Dad say?'
(1) You may go to the beach or go to the cinema (I am not sure which).
(2) You may go to the beach.

• Kamp 1973, sec. 2–3: PERMISSION $\not\subseteq$ ASSERTION
Permission (salient reading in context A) is not assertion (salient in ctx B). Any theory that treats permission as assertion makes wrong predictions about entailment (e.g. extending (2) $\models$ (1) in context B to context A).

• Kamp 1973, sec. 4–5: TOWARD A THEORY OF PERMISSION
**Basic idea**: Assertion narrows down the class of worlds we may be in (*common ground*), whereas permission lifts a ban and thereby expands the class of compliant worlds (*compliance sphere*). A disjunctive assertion narrows the input common ground less than either of the disjuncts. In contrast, a disjunctive (free choice) permission lifts more input bans and thereby expands the compliance sphere more than either of the disjuncts. In both cases the disjunction operator, $\vee$, is interpreted as set-theoretic union.

## 2 PERMISSION $\not\subseteq$ ASSERTION

- "one principle […] which appears to be sound on any plausible understanding of *ought* or *obligatory* [is] the inference rule
  (5) $\dfrac{\phi \leftrightarrow \psi}{O\phi \leftrightarrow O\psi}$
  […] In addition I adopt for [the intended interpretation]…axiom
  (6) $O(p \wedge q) \rightarrow Op$
  which, ostensibly, says that if the conjunction of $p$ and $q$ is obligatory then so is $p$ by itself." (Kamp 1973:59)

The following logic implements these minimal assumptions. To facilitate comparison with Aloni 2007, I write **must** and **may** for Kamp's *O* and *P*.

• DEONTIC LOGIC ($DL_1$) ~ Kamp's

**D1** (syntax)

0. $Con_0 \;=\; \{a, b, \ldots\}$
   $Con_1 \;=\; \{\textbf{must}, \textbf{may}, \ldots\}$

i. $Con_0 \subseteq Fml$

ii. $\neg\phi \in Fml$ if $\phi \in Fml$

iii. $\phi \wedge \psi \in Fml$ if $\phi, \psi \in Fml$

iv. $\phi \vee \psi \in Fml$ if $\phi, \psi \in Fml$

v. $\mu\phi \in Fml$ if $\mu \in Con_1$ & $\phi \in Fml$

**D2** (models)

A $DL_1$-model is a pair $M = \langle W, [\![\cdot]\!]\rangle$, where

i. $W$ is a non-empty set (of possible worlds)

ii. $[\![\cdot]\!]$ is an interpretation function such that:

- $[\![\phi]\!] \in Pow(W)$ if $\phi \in Con_0$
- $[\![\mu]\!] \in Pow(W)^{Pow(W)}$ if $\mu \in Con_1$ (5)
- $[\![\textbf{must}]\!](q \cap r) \subseteq [\![\textbf{must}]\!](q)$ if $q, r \subseteq W$ (6)
- $[\![\textbf{may}]\!](q) = W \backslash [\![\textbf{must}]\!](W \backslash q)$ if $q \subseteq W$ (7)

**D3** (semantics)

ii. $[\![\neg\phi]\!] \;=\; W \backslash [\![\phi]\!]$

iii. $[\![\phi \wedge \psi]\!] \;=\; [\![\phi]\!] \cap [\![\psi]\!]$

iv. $[\![\phi \vee \psi]\!] \;=\; [\![\phi]\!] \cup [\![\psi]\!]$

v. $[\![\mu\phi]\!] \;=\; [\![\mu]\!]([\![\phi]\!])$

**D4** (truth and entailment)

i. $\models_w \phi$ iff $w \in [\![\phi]\!]$

ii. $\phi \models \psi$ iff for every $DL_1$-model $\langle W, [\![\cdot]\!]\rangle$, $[\![\phi]\!] \subseteq [\![\psi]\!]$

Kamp observes that $DL_1$ and any other implementation of (5)–(7) predicts
• WRONG ENTAILMENT FOR PERMISSION

**F.**|= (entailment fact)
**may** $\phi$ |= **may** $(\phi \vee \psi)$

PROOF.
Let $M = \langle W, [\![\cdot]\!]\rangle$ be any $DL_1$-model. Then $[\![\phi]\!], [\![\psi]\!] \subseteq W$ for all $\phi, \psi \in Fml$ (by D2 and D3). Therefore:

1. $[\![\mathbf{must}]\!]((W\backslash[\![\phi]\!]) \cap (W\backslash[\![\psi]\!])) \subseteq [\![\mathbf{must}]\!](W\backslash[\![\phi]\!])$ D2.ii.**must**

Hence

2. $W\backslash[\![\mathbf{must}]\!](W\backslash[\![\phi]\!]) \subseteq W\backslash[\![\mathbf{must}]\!]((W\backslash[\![\phi]\!]) \cap (W\backslash[\![\psi]\!]))$ \, ⊆[3]
3. $W\backslash[\![\mathbf{must}]\!](W\backslash[\![\phi]\!]) \subseteq W\backslash[\![\mathbf{must}]\!](W\backslash([\![\phi]\!] \cup [\![\psi]\!]))$ \, ∩, ∪[4]
4. $W\backslash[\![\mathbf{must}]\!](W\backslash[\![\phi]\!]) \subseteq W\backslash[\![\mathbf{must}]\!](W\backslash[\![\phi \vee \psi]\!])$ D3.∨
5. $[\![\mathbf{may}]\!]([\![\phi]\!]) \subseteq [\![\mathbf{may}]\!]([\![\phi \vee \psi]\!])$ D2.ii.**may**
6. $[\![\mathbf{may}\ \phi]\!] \subseteq [\![\mathbf{may}\ (\phi \vee \psi)]\!]$ D3.$\mu$

Since this holds to any $DL_1$-model, D4 yields: **may** $\phi$ |= **may** $(\phi \vee \psi)$.

• Cf. Aloni (2007): PERMISSION ⊆ ASSERTION, BUT NO **F.**|=
4 reasons:
– Kamp '73 assumes *surface-faithful* translation, Aloni '07 does not, e.g.

(1) You may go to the beach or go to the cinema.
($1_K$) **may**$(b \vee c)$ Kamp 1973
($1_A$) MAY $\exists p(p \wedge (p = b \vee p = c))$ Aloni 2007

– Kamp '73 and Aloni '07 disagree on the key assumptions: (5) modals denote propositional functions; (6) if $(\phi \wedge \psi)$ is a deontic necessity, then so is $\psi$; and (7) deontic possibility is the dual of deontic necessity:

| | Kamp | Aloni |
|---|---|---|
| (5) | ✓: $[\![\phi]\!] = [\![\psi]\!] \rightarrow [\![\mathbf{may}\ \phi]\!] = [\![\mathbf{may}\ \psi]\!]$ | ×: $[\![$MAY $\phi]\!]$ funct. of ALT$(\phi)_{M,g}$ |
| (6) | ✓: **must** $(\phi \wedge \psi)$ \|= **must** $\psi$ | ×: MUST $(\phi \wedge \psi)$ $\not\models$ MUST $\psi$ |
| (7) | ✓: $[\![\mathbf{may}\ \phi]\!] = [\![\neg\ \mathbf{must}\ \neg\psi]\!]$ | ×: $[\![$MAY $\phi]\!] \neq [\![\neg$MUST $\neg\phi]\!]$ |

• TRUTH VALUE PROBLEM

Assimilating permission to assertion also predicts that permissions (e.g. (1) in context A) are true or false, like assertions ((1) in context B). According to my (MB) intuitions, permissions have no truth value (like directives).

---

[3] $A, B \subseteq U \;\&\; A \subseteq B \rightarrow U\backslash B \subseteq U\backslash A$
[4] $A, B \subseteq U \rightarrow (U\backslash A) \cap (U\backslash B) = U\backslash(A \cup B)$

## 3 DEONTIC LOGIC WITH PERMISSION ($DL_2$)

**D1** (syntax)

0. $Con_0 = \{a, b, \ldots\}$ $Con_1 = \{\textbf{must}, \textbf{may}, \ldots\}$ $Con_2 = \{\textbf{may}_2, \ldots\}$

i. $Con_0 \subseteq Fml$

ii. $\neg\phi \in Fml$ if $\phi \in Fml$

iii. $\phi \wedge \psi \in Fml$ if $\phi, \psi \in Fml$

iv. $\phi \vee \psi \in Fml$ if $\phi, \psi \in Fml$

v. $\mu\phi \in Fml$ if $\mu \in Con_1 \cup Con_2$ & $\phi \in Fml$

**D2** (models)

A $DL_2$-model is a triple $M = \langle W, B, [\![\cdot]\!]\rangle$, where

i. $W$ is a non-empty set (of possible worlds)

ii. $\varnothing \subset B \subseteq Pow(W)$ is a set (of standing bans) s.t. $p, q \in B \rightarrow p \not\subseteq q$. For any $p \subseteq W$, $p_B := \{q \in B|\ q \subseteq p\}$ (bans in $B$ that can be lifted by $p$).

iii. $[\![\cdot]\!]$ is an interpretation function such that

- $[\![\phi]\!] \in Pow(W)$ if $\phi \in Con_0$
- $[\![\mu]\!] \in Pow(W)^{Pow(W)}$ if $\mu \in Con_1 \cup Con_2$ (5)
- $[\![\textbf{must}]\!](p \cap q) \subseteq [\![\textbf{must}]\!](p)$ if $p, q \subseteq W$ (6)
- $[\![\textbf{may}]\!](p) = W \backslash [\![\textbf{must}]\!](W \backslash p)$ if $p \subseteq W$ (7)
- $[\![\textbf{may}_2]\!](p) = W \backslash \cup(B \backslash p_B)$ if $p \subseteq W$ & $p_B \neq \varnothing$
  $= \varnothing$ if $p \subseteq W$ & $p_B = \varnothing$

**D3** (proposition expressed)

ii. $[\![\neg\phi]\!] = W \backslash [\![\phi]\!]$

iii. $[\![\phi \wedge \psi]\!] = [\![\phi]\!] \cap [\![\psi]\!]$

iv. $[\![\phi \vee \psi]\!] = [\![\phi]\!] \cup [\![\psi]\!]$

v. $[\![\mu\phi]\!] = [\![\mu]\!]([\![\phi]\!])$

**D4** (update semantics). For any $p, q \in Pow(W) \backslash \{\varnothing\}$,

i. $\langle p, q\rangle[\phi] = \langle p \cap [\![\phi]\!], q\rangle$

ii. $\langle p, q\rangle[\neg\phi] = \langle p \cap [\![\neg\phi]\!]), q\rangle$

iii. $\langle p, q\rangle[\phi \wedge \psi] = \langle p, q\rangle[\phi][\psi]$

iv. $\langle p, q\rangle[\phi \vee \psi] = \langle p \cap [\![\phi \vee \psi]\!], q\rangle$

v. $\langle p, q\rangle[\mu\phi] = \langle p \cap [\![\mu\phi]\!], q\rangle$ if $\mu \in Con_1$
$= \langle p, q \cup [\![\mu\phi]\!]\rangle$ if $\mu \in Con_2$

**D5** (informativity, empowerment, and truth)

- $\phi$ is *informative* iff for some $DL_2$-model $M$, $p$ & $q$: $\varnothing \subset (\langle p, q\rangle[\phi])_1 \subset p$
- $\phi$ is *empowering* iff for some $DL_2$-model $M$, $p$ & $q$: $\varnothing \subset p \subset (\langle p, q\rangle[\phi])_2$
- given $p$, $\phi$ is *true* in $w$, $p \models_w \phi$, iff $\phi$ is informative & $\exists q$: $w \in (\langle p, q\rangle[\phi])_1$

**D6** (entailment) ($|=_B$ for Kamp's *p-entails*)

- $\phi \models \psi$ iff ϕ and ψ are informative and for all $M = \langle W, B, [\![\cdot]\!]\rangle$ & $p, q \in Pow(W)\backslash\{\varnothing\}$: $(\langle p, q\rangle[\phi])_1 \subseteq (\langle p, q\rangle[\psi])_1$
- $\phi \models_B \psi$ iff ϕ and ψ are empowering and for all $M = \langle W, B, [\![\cdot]\!]\rangle$ & $p, q \in Pow(W)\backslash\{\varnothing\}$: $[\![\phi]\!]_B \neq \varnothing$ & $[\![\psi]\!]_B \neq \varnothing \rightarrow$ $(\langle p, q\rangle[\psi])_2 \subseteq (\langle p, q\rangle[\phi])_2$

- SOLUTION TO ENTAILMENT PUZZLE

In context A disjunction (1) *B-entails* (permits more/is more empowering than) disjunct (2):

A. Father to Michael:

(1) You may go to the beach or go to the cinema.
(1′) $\mathbf{may}_2\ (b \vee c)$

(2) You may go to the beach.
(2′) $\mathbf{may}_2\ b$

*DL*$_2$-facts:

**F1′**.$[\![\cdot]\!]$ If $[\![b]\!], [\![c]\!] \in B$, then $W\backslash\cup(B\backslash\{[\![b]\!], [\![c]\!]\}) \subseteq [\![\mathbf{may}_2\ (b \vee c)]\!]$
**F2′**.$[\![\cdot]\!]$ If $[\![b]\!] \in B$, then $[\![\mathbf{may}_2\ b]\!] = W\backslash\cup(B\backslash\{[\![b]\!]\})$
**F′**.⊆ If $[\![b]\!], [\![c]\!] \in B$, then $[\![\mathbf{may}_2\ b]\!] \subseteq [\![\mathbf{may}_2\ b \vee c]\!]$
**F**.$|=_B$ (*DL*$_2$-fact ?) $\mathbf{may}_2\ (b \vee c) \models_B \mathbf{may}_2\ b$

In context B disjunction (1) is *entailed* by (asserts less/is less informative than) disjunct (2)

B. Michael's little sister in response to his question: 'What did Dad say?'

(1) You may go to the beach or go to the cinema (I am not sure which).
(1″) $\mathbf{may}\ (b \vee c)$

(2) You may go to the beach.
(2″) $\mathbf{may}\ b$

*DL*$_2$-facts = *DL*$_1$-facts:

**F**1″.$[\![\cdot]\!]$ $[\![\mathbf{may}\ (b \vee c)]\!] = W\backslash[\![\mathbf{must}]\!](W\backslash([\![b]\!] \cup [\![c]\!]))$
**F**2″.$[\![\cdot]\!]$ $[\![\mathbf{may}\ b]\!] = W\backslash[\![\mathbf{must}]\!](W\backslash[\![b]\!])$
**F″**.⊆ $[\![\mathbf{may}\ b]\!] \subseteq [\![\mathbf{may}\ (b \vee c)]\!]$
**F**.|= $\mathbf{may}\ b \models \mathbf{may}\ (b \vee c)$ (cf. proof in sec. 2)

• SAMPLE PROOFS

**F1′**. Given $[\![b]\!], [\![c]\!] \in B$,
$W \backslash \cup(B \backslash \{[\![a]\!], [\![b]\!]\}) \subseteq [\![\mathbf{may}_2\ (b \vee c)]\!]$

PROOF. Given $[\![b]\!], [\![c]\!] \in B$, we infer (1):

| | |
|---|---|
| 1. $\{[\![b]\!], [\![c]\!]\} \subseteq \{q \in B \mid q \subseteq [\![b]\!] \cup [\![c]\!]\} \subseteq Pow(W)$ | $\subseteq$, D2.iii |

Hence:

| | |
|---|---|
| 2. $B \backslash \{q \in B \mid q \subseteq [\![b]\!] \cup [\![c]\!]\} \subseteq B \backslash \{[\![b]\!], [\![c]\!]\} \subseteq Pow(W)$ | $\backslash, \subseteq$ |
| 3. $\cup(B \backslash \{q \in B \mid q \subseteq [\![b]\!] \cup [\![c]\!]\}) \subseteq \cup(B \backslash \{[\![b]\!], [\![c]\!]\}) \subseteq W$ | $\cup, \subseteq$ |
| 4. $W \backslash \cup(B \backslash \{[\![b]\!], [\![c]\!]\}) \subseteq W \backslash \cup(B \backslash \{q \in B \mid q \subseteq [\![b]\!] \cup [\![c]\!]\})$ | $\backslash, \subseteq$ |
| 5. $W \backslash \cup(B \backslash \{[\![b]\!], [\![c]\!]\}) \subseteq W \backslash \cup(B \backslash ([\![b]\!] \cup [\![c]\!])_B)$ | D2.ii.$p_B$ |
| 6. $W \backslash \cup(B \backslash \{[\![b]\!], [\![c]\!]\}) \subseteq [\![\mathbf{may}_2]\!]([\![b]\!] \cup [\![c]\!])$ | D2.iii.$\mathbf{may}_2$ |
| 7. $W \backslash \cup(B \backslash \{[\![b]\!], [\![c]\!]\}) \subseteq [\![\mathbf{may}_2]\!]([\![b \vee c]\!])$ | D3.$\vee$ |
| 8. $W \backslash \cup(B \backslash \{[\![b]\!], [\![c]\!]\}) \subseteq [\![\mathbf{may}_2\ (b \vee c)]\!]$ | D3.$\mu$ |

**F′**.$\subseteq$. Given $\{[\![b]\!], [\![c]\!]\} = [\![b \vee c]\!]_B$,
$[\![\mathbf{may}_2\ b]\!] \subseteq [\![\mathbf{may}_2\ b \vee c]\!]$

PROOF. We first note that (1) implies (4):

| | |
|---|---|
| 1. $\{[\![b]\!]\} \subseteq \{[\![b]\!], [\![c]\!]\} \subseteq Pow(W)$ | $\subseteq$, D2.iii |
| 2. $B \backslash \{[\![b]\!], [\![c]\!]\} \subseteq B \backslash \{[\![b]\!]\} \subseteq Pow(W)$ | $\backslash, \subseteq$ |
| 3. $\cup(B \backslash \{[\![b]\!], [\![c]\!]\}) \subseteq \cup(B \backslash \{[\![b]\!]\}) \subseteq W$ | $\cup, \subseteq$ |
| 4. $W \backslash \cup(B \backslash \{[\![b]\!]\}) \subseteq W \backslash \cup(B \backslash \{[\![b]\!], [\![c]\!]\})$ | $\backslash, \subseteq$ |

Hence, given $[\![b]\!], [\![c]\!] \in B$,

| | |
|---|---|
| 5. $[\![\mathbf{may}_2\ b]\!] \subseteq [\![\mathbf{may}_2\ b \vee c]\!]$ | F1′, F2′ |