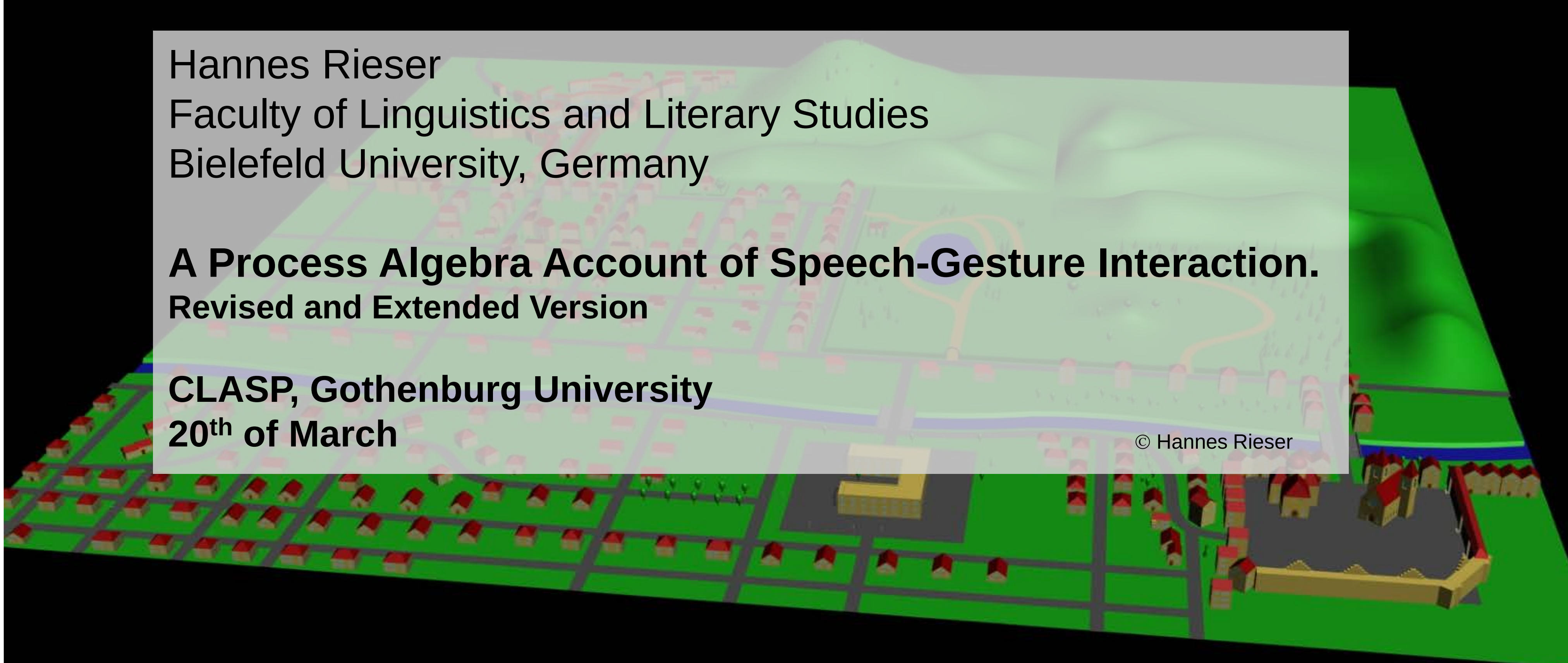


Hannes Rieser
Faculty of Linguistics and Literary Studies
Bielefeld University, Germany

A Process Algebra Account of Speech-Gesture Interaction.

Revised and Extended Version

CLASP, Gothenburg University
20th of March

© Hannes Rieser

Contents

Introduction and Idea of Paper

Facts (SaGA rated corpus-data, Lücking et al. 2012):

Iconic and referential gestures

- are semantically related with the speech they accompany
- do not perfectly synchronize with semantic coordination point in speech
  - *contra* McNeill (1992)
  - modern speech-gesture interaction research (Alahverdzhieva and Lascarides 2010, Giorgolo (2010), Lascarides, A. and Stone, M. 2009, Lücking 2013, Röpke 2011, Röpke, Hahn, and Rieser 2013 among others) has not developed general solutions for this problem
- come before relevant speech, after it or overlap
- can be independent/provide independent content

Introduction and Idea of Paper

Claim: description of speech-gesture coordination cannot be given solely in a naïve compositional way

Caveat: composition does play a role finally, when the speech-gesture contact points have been identified

Given asynchrony, a lot of work has to be done before the speech-gesture contact points are identified.
Parallel problem also treated here: incrementality and scope resolution

Introduction and Idea of Paper

Proposed solution for the asynchrony problem:

- view gesture and speech as independent processes interacting if semantically apt
- move to methodology working with a process ontology instead of a purely domain-of-objects one as in model theory

used: ψ-calculus (Johansson, 2010, Bengtson et al. 2011)

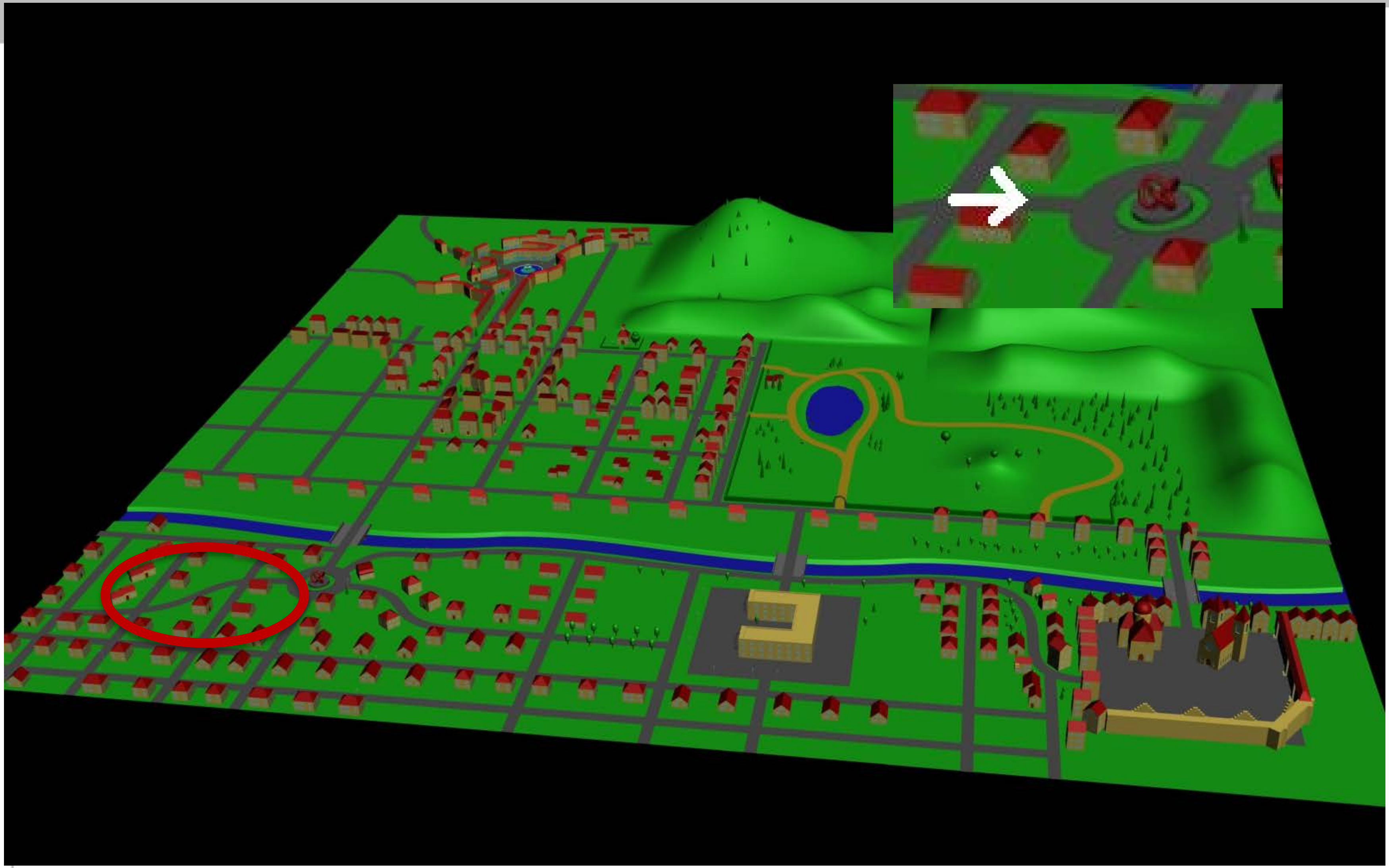
- Process Algebra
  - recent extension of Milner's π-calculus (Milner,1999, Parrow, 2001, Sangiorgi and Walker, 2001),
  - works with processes (agents) and data structures
    - transmitted between agents *via* structured channels using input-output facilities

Aside: gesture and speech are NOT bisimilar processes, no identity

Contents

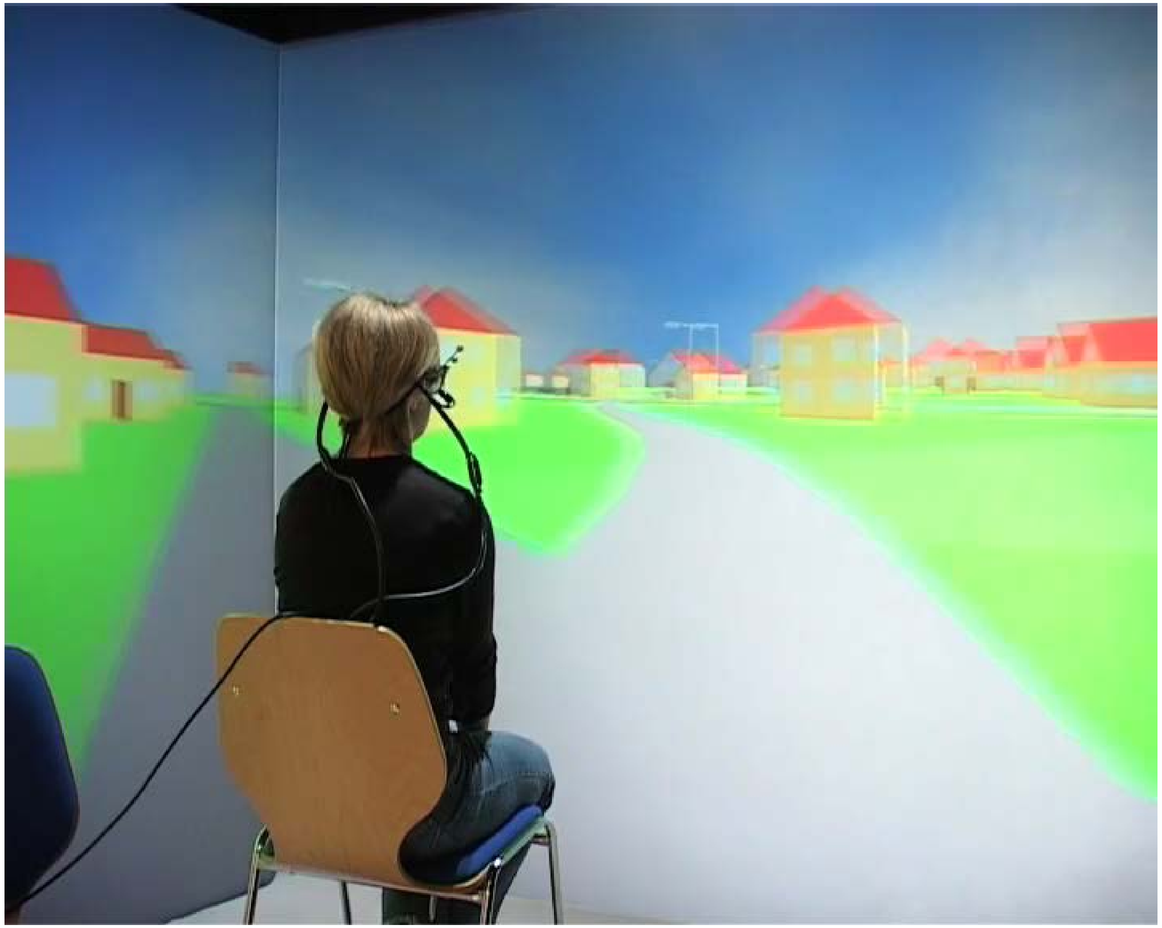
Speech-gesture Asynchrony

Stimulus: VR-rendering of domain model for route-description dialogues



Speech-gesture Asynchrony

Stimulus: VR-rendering of domain model for route-description dialogues



Speech-gesture Asynchrony

Route-g. Ok, well the tour starts and yes it is such a wormed street follow it
|- WINDING-GESTURE$_{RG}$ -|
Follower Yes

Route-g. And then there is a sculpture it is kind of red looks like two interwined bretzels
|-TWO-BRETZELS GESTURE$_{RG}$-|
Follower

Route-g. or so

Follower aha

Route-g. Towards it and it is kind of round and it goes in a circle well there is a circle
|- GESTURE$_{RG}$ -| |- GESTURE$_{RG}$ -| |- GESTURE$_{RG}$ -|
around it

Follower Yes

Route-g. And then one passes its right side
|- GESTURERG -| |- GESTURERG -| BODY MOVE TO PANTOMIME ONLOOKER POSITION
AND DIRECTIONS ABORTED
Follower Hold on well you-CUT OFF well you walk now
|- WINDING-GESTURE$_{F}$ ----
into this street and then where is the sculpture is it at the front or to the
--------------|- WINDING-GESTURE-HELD --
|-GESTURE$_{F-LH}$-| |-GESTURE$_{F-LH}$-| |-GESTURE$_{F-LH}$-| |-GESTURE$_{F-LH}$-|
right
- winding-gesture-held --
|-GESTURE$_{F-LH}$-|
Route-g. Yes it comes at you notice it immediately
|- TWO-HANDED-GESTURE$_{RG}$ -|
Follower OK
- WINDING-GESTURE-HELD --
Route-g. When you notice the wormed street.
|- WINDING GESTURE -|
Follower OK
- WINDING-GESTURE-HELD --

Speech-gesture Asynchrony: Stimulus, Cave-Data, Dialogue Example, Annotation

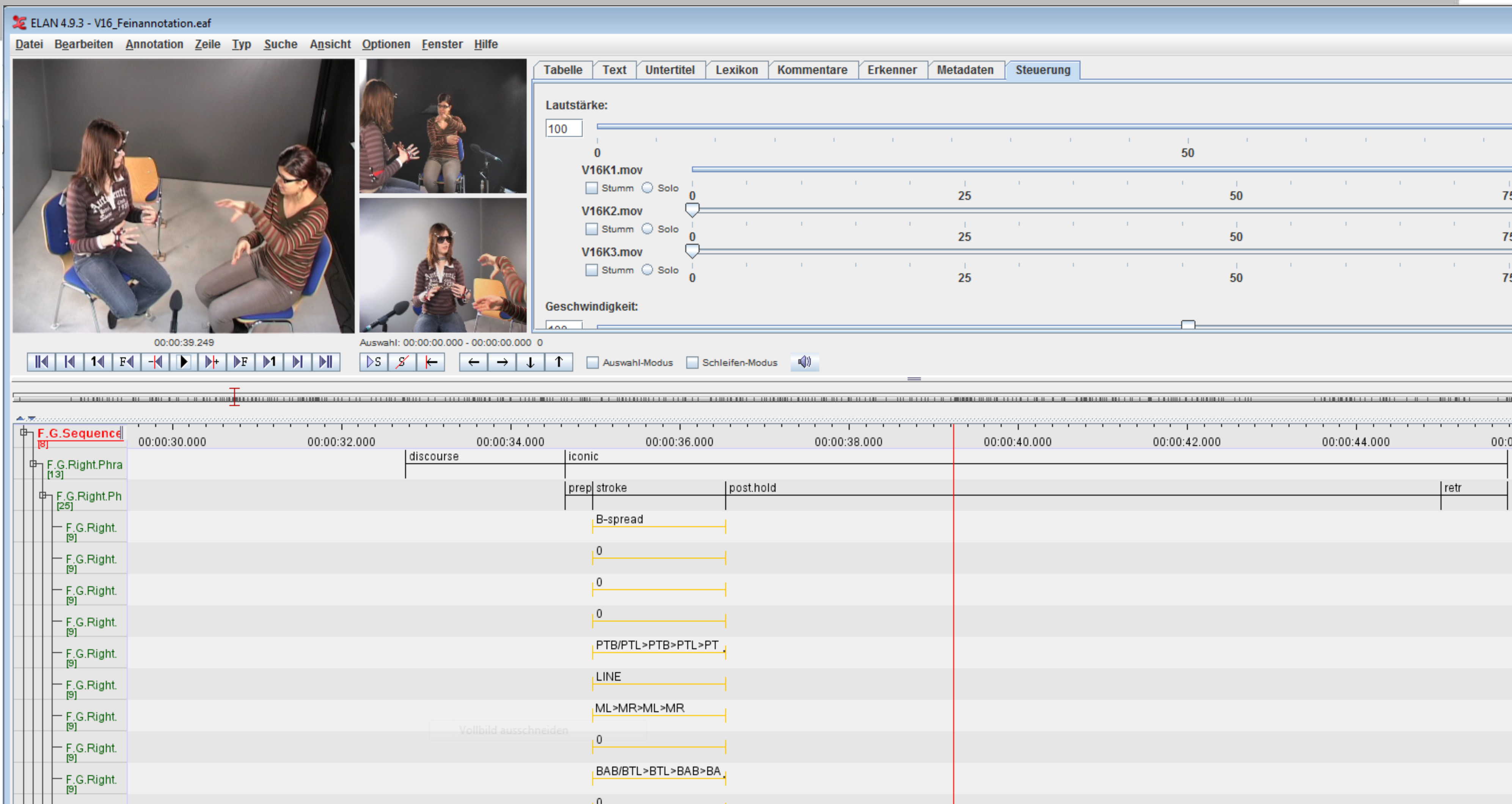
Speech-gesture Asynchrony

Dialogue Example (Route description V16, SaGA-Corpus)

```
22 Follower                              Hold on well you-CUT OFF well you walk now
23                                                                  |- WINDING-GESTURE_F ----
24          into this street and then where is the sculpture is it at the front      or to the
25          ---------------|- WINDING-GESTURE-HELD ---------------------------------------------
                                |-GESTURE_F-LH-|   |-GESTURE_F-LH-|   |-GESTURE_F-LH-| |-GESTURE_F-LH-|
26          right
27          - winding-gesture-held ---------------------------------------------------------------
            |-GESTURE_F-LH-|
28 Route-g. Yes it comes at            you notice it immediately
29          |- TWO-HANDED-GESTURE_RG -|
30 Follower OK
31          - WINDING-GESTURE-HELD ---------------------------------------------------------------
32 Route-g. When you notice the wormed street.
33          |- WINDING GESTURE -|
34 Follower OK
35          - WINDING-GESTURE-HELD ---------------------------------------------------------------
```

Speech-gesture Asynchrony: Stimulus, Cave-Data, Dialogue Example, Annotation

Speech-gesture Asynchrony: Stimulus, Cave-Data, Dialogue Example, Annotation



ELAN-annotation of central example (still)

Speech-gesture Asynchrony: Stimulus, Cave-Data, Dialogue Example, Annotation

Observations

Follower's contribution:

- winding gesture starts with "well" line (22) and continues up to "street" and is held until line (34)
- no verbal indication of street's bends, instead Follower's winding gesture plus the word "street"
- the two information processes/agents together produce multi-modally "wormed street"/"geschlängelte Straße"

Route-giver's contribution:

- Uses a winding gesture in lines 2 and 33.

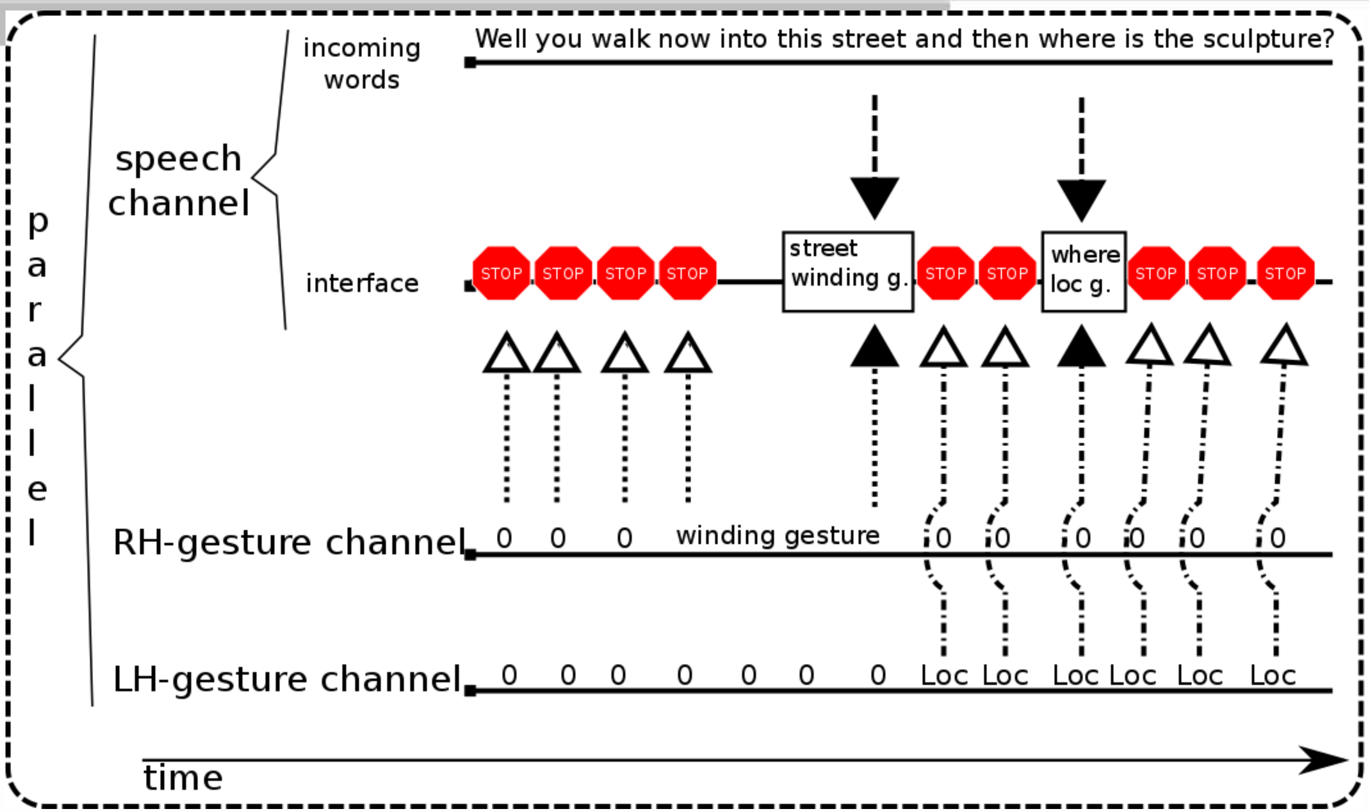
Gestural "contact" in lines 31 to 35:

- Follower holds the winding gesture
- Route-g. produces a winding gesture

Contents

Speech-gesture Asynchrony: Solution Intuitions

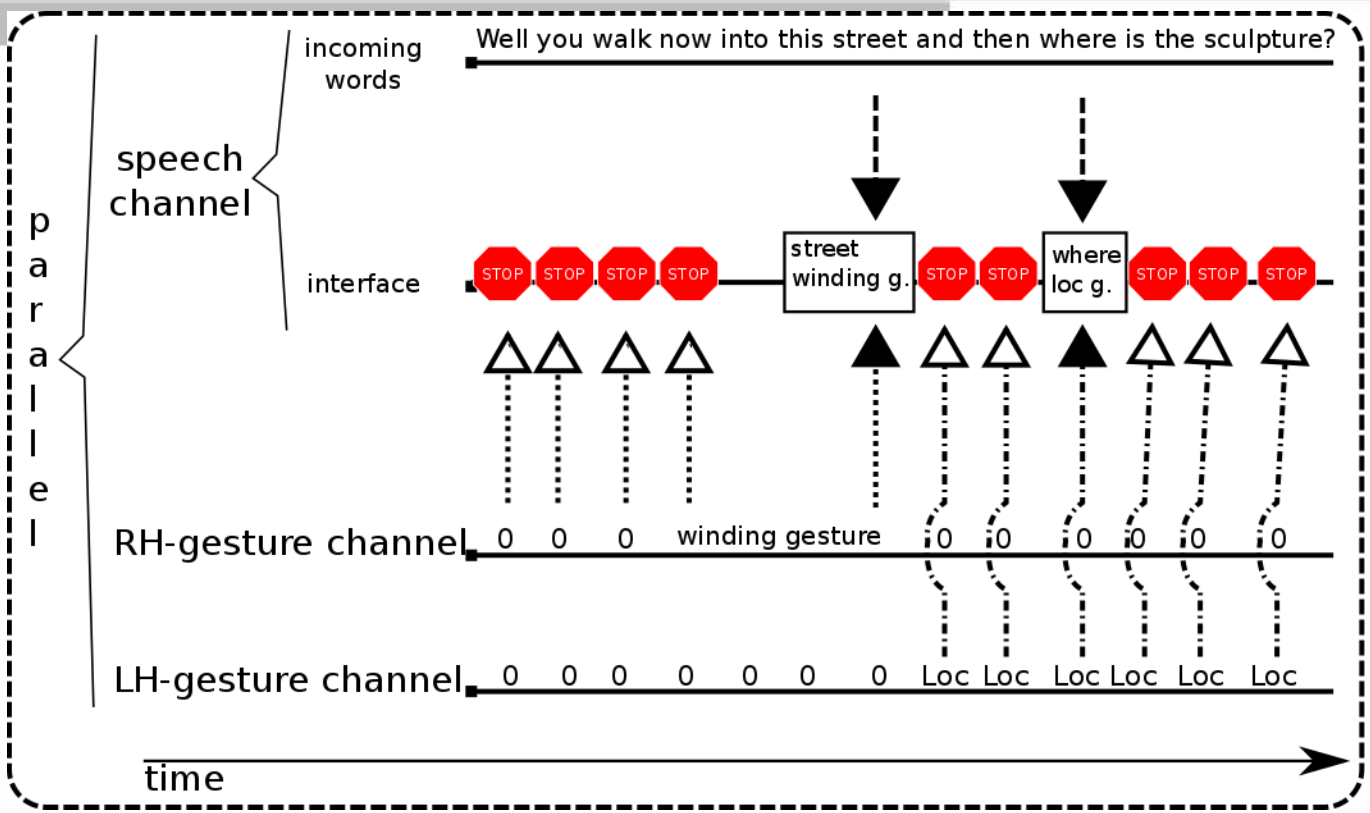
Didactic picture of channel interaction between speech and gesture



The winding gesture is taken to cooperate with the noun “street” (preferred reading).

Speech-gesture Asynchrony: Solution Intuitions

Didactic picture of channel interaction between speech and gesture

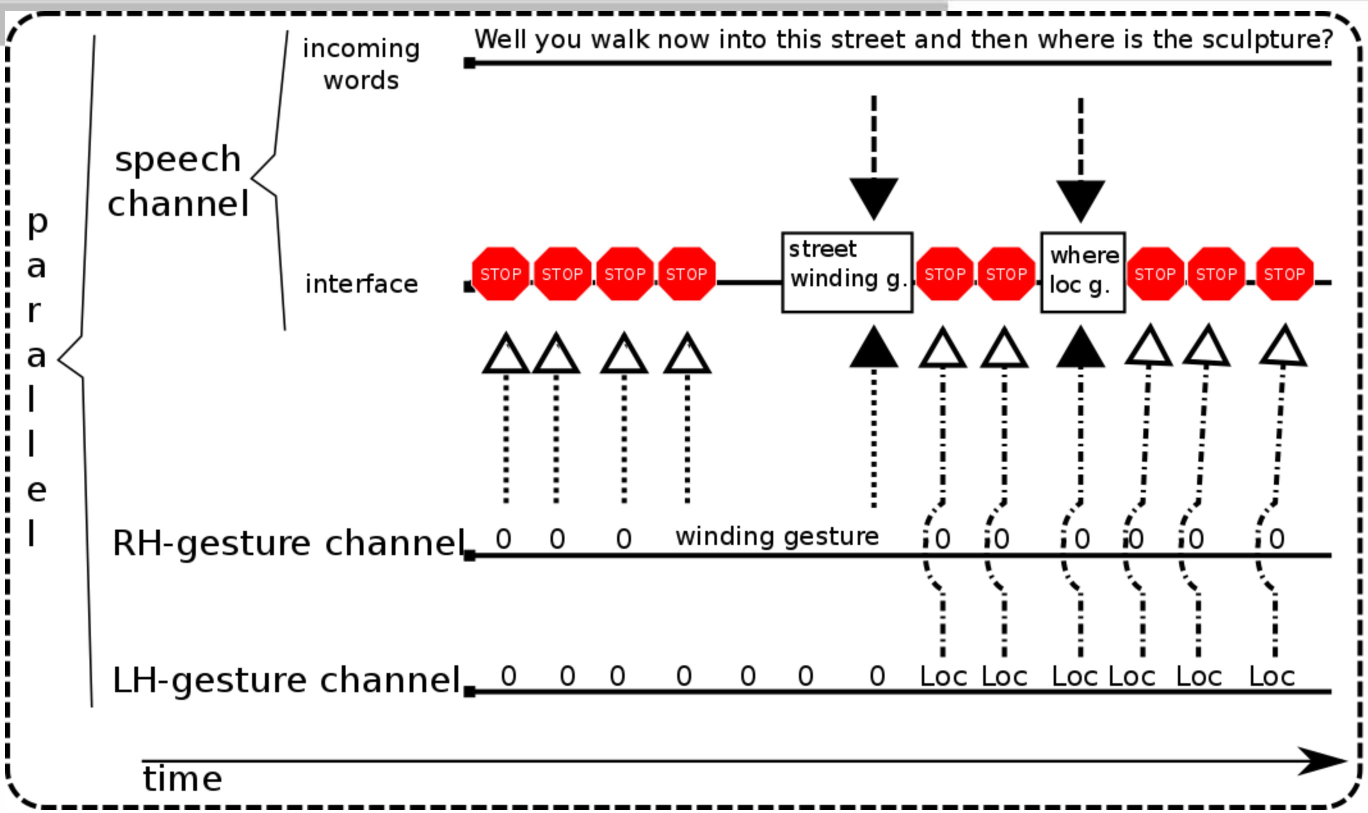


Assumptions:

- channels for speech, LH- and RH-gesture
- independent production of information on channels speech, RH, and LH

Speech-gesture Asynchrony: Solution Intuitions

Didactic picture of channel interaction between speech and gesture

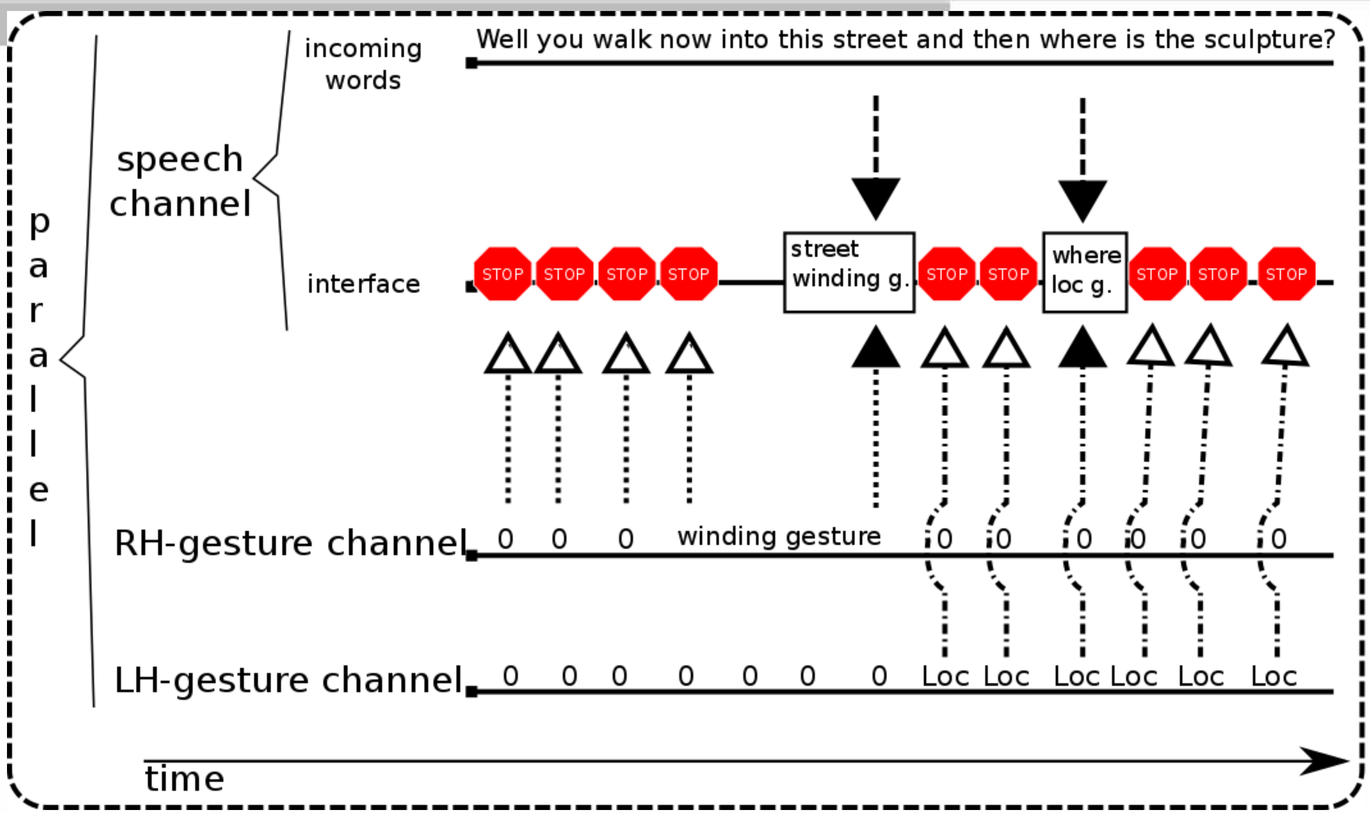


Assumptions:

- winding gesture's stroke information denied access to interface channel
- winding gesture waits until it can interact

Speech-gesture Asynchrony: Solution Intuitions

Didactic picture of channel interaction between speech and gesture

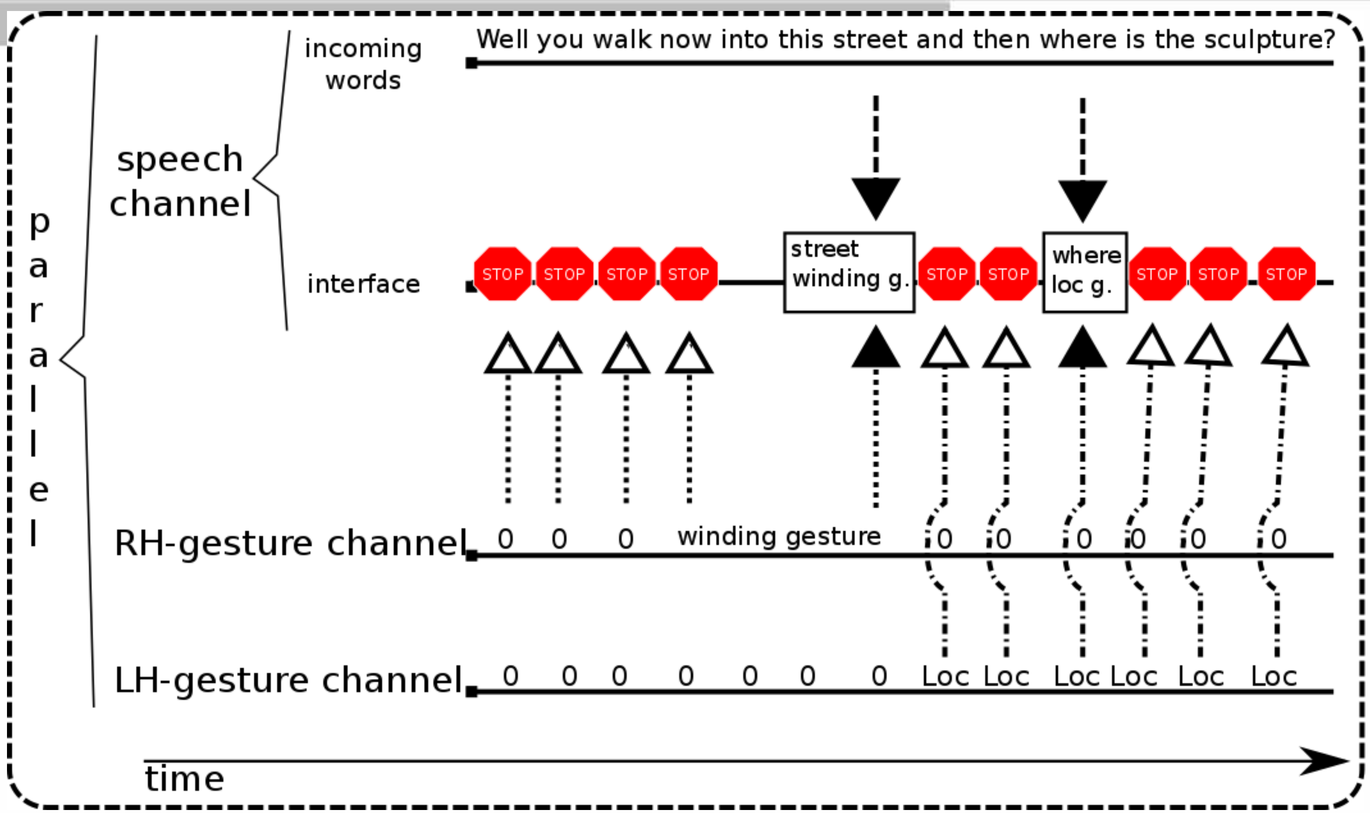


How done?

- access properties of winding gesture's and "street"'s semantics implemented by typing
- here Montague grammar/typed λ-calculus, alternatives: Combinatory Logics, HPSG's AVMs, other typing regimes

Speech-gesture Asynchrony: Solution Intuitions

Didactic picture of channel interaction between speech and gesture

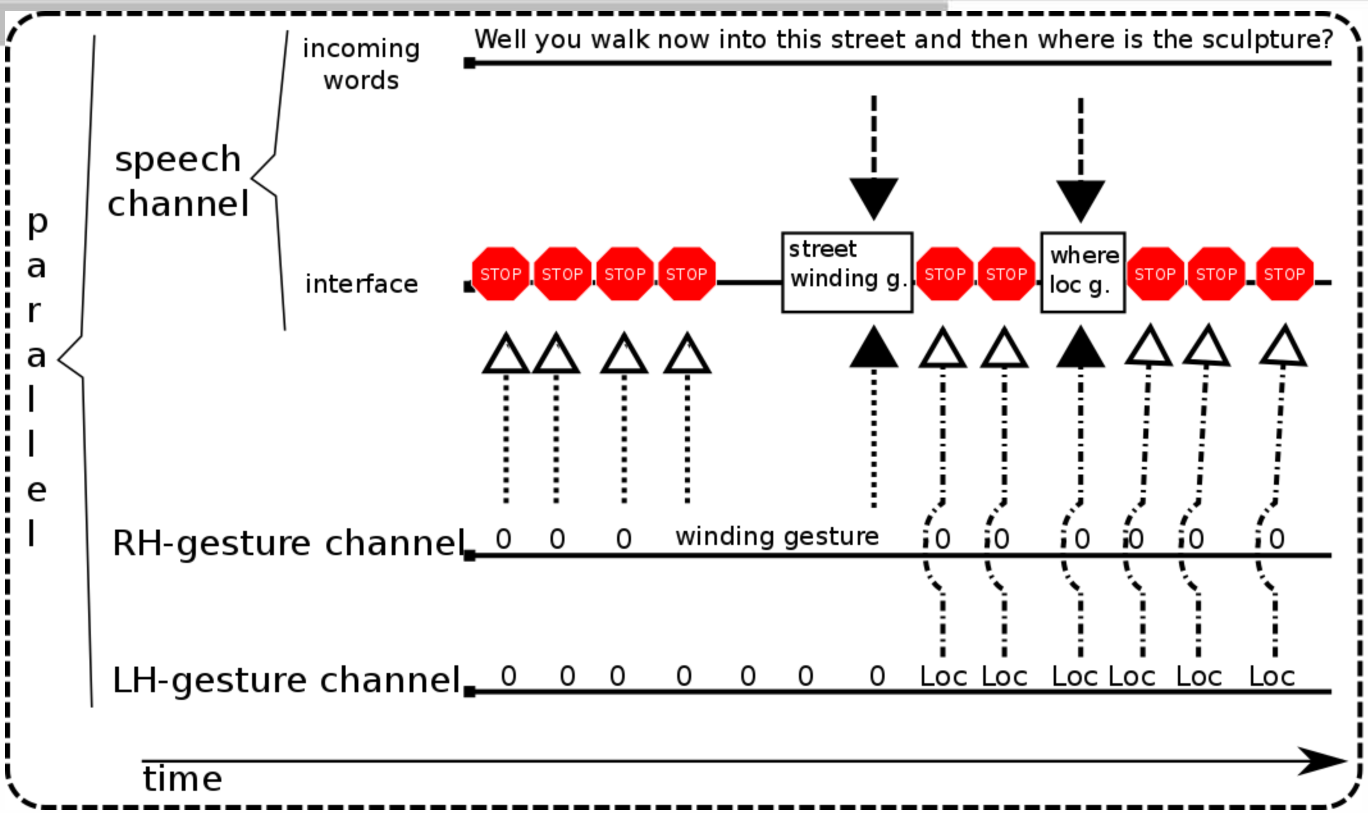


Assumptions:

- after the interaction, winding gesture's interfacing possibilities are set to 0.

Speech-gesture Asynchrony: Solution Intuitions

Didactic picture of channel interaction between speech and gesture



Technical problem:

- Fuse channel technology and typed λ-calculus

Attempted in my λ-ψ-hybrid calculus

Speech-gesture Asynchrony: Solution Intuitions

Cooper’s Conjecture: The Event Case

R. Cooper’s idea (Ghent July 2016):

- What the Follower has in mind is the event of walking into the street.
- Gesture indicates a property of the event.
- Event going in bends.

Accounts for the length of the stroke: from “you” to “into” (“Du” …”rein”).

Contents

ψ-calculus Essentials and How to Capture the Main Intuitions with them

Recap and Outlook

Empirical observations provide evidence for

- channels on which information (data, agents or procedures) can be sent
- processes operating concurrently
- communication among processes and

- indication when a process is active (waits, acts) or comes to its end

Communication: organised mainly with

  - input- output mechanism
  - parallel processes

Blocking of information:

- achieved with typing of data structures and processes.

ψ-calculus Essentials and How to Capture the Main Intuitions with them

What do we have in Process Algebra ψ to model these intuitions?

We have parameters, operators on these, frames and agents.

ψ-calculus Essentials and How to Capture the Main Intuitions with them

Parameters indicating data types given in definition 1

Definition 1

T the (data) terms, ranged over by M, N

C the conditions, ranged over by φ

A the assertions, ranged over by (capital) Ψ

(Bengtson et al. 2011, pp. 4-14)

Data terms:

- can come from any (higher order) logic
- here typed λ-calculus chosen
- reasons:
  - semantics tradition
  - well researched into model- and proof-theoretically (cf. Barendregt et al. 2013, Cresswell 1973)
  - developing hybrid λ-ψ-functor-argument-structures

ψ-calculus Essentials and How to Capture the Main Intuitions with them

The equivariant operators come in definition 2

(“equivariance” for capturing α-equivalence)

Definition 2

$\leftrightarrow: T \times T \rightarrow C$	Channel Equivalence
$\otimes: A \times A \rightarrow A$	Composition
$\vdash \subseteq A \times C$	Entailment

ψ-calculus Essentials and How to Capture the Main Intuitions with them

The ψ-calculus agents/processes, indicated by P, Q…

Definition 3

0	Inert agent
$\overline{M}N.P$	Output process
$\underline{M}(\lambda x)N.P$	Input process, written as "$\underline{M}(x)N.P$"
τ	Silent agent
case $\varphi_1: P_1 \parallel \ldots \parallel \varphi_n: P_n$	Case construct
$(\nu a)P$	Restriction on process P
$P \mid Q$	Parallel processes
$!P$	Replication process: $!P =_{def} P \mid !P$
$(\vert\Psi\vert)$	Assertion
.	Compositional operator

Auxiliary symbols:

<, >	Brackets enclosing typed λ-expressions

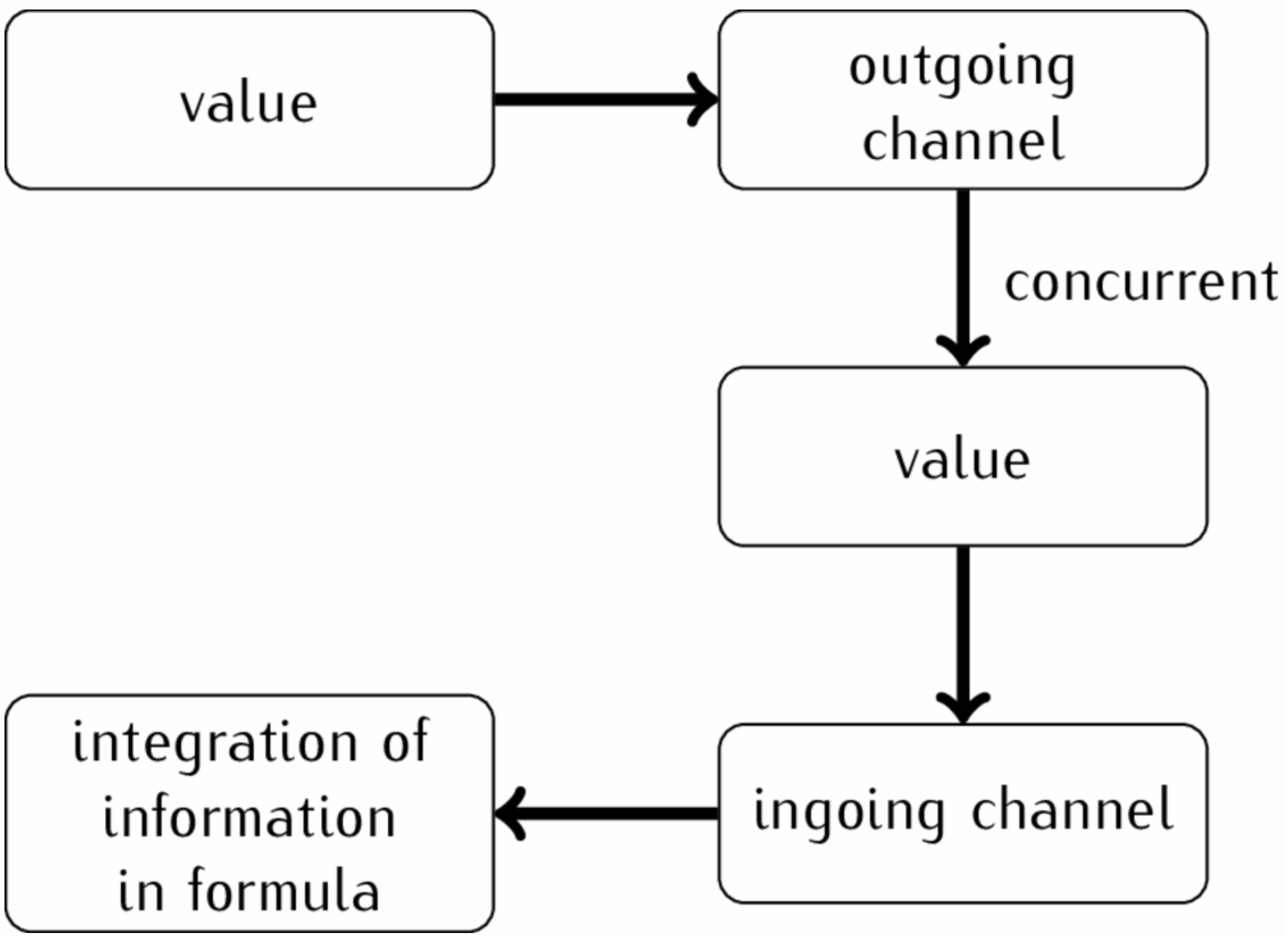
ψ-calculus Essentials and How to Capture the Main Intuitions with them

The ψ-calculus agents/processes, indicated by P, Q…

Mainly used in this talk: 0 agent, $\overline{M}N.P$, $\underline{M}(x)N.P$, P | Q, !P, "."

ψ-calculus Essentials and How to Capture the Main Intuitions with them

i-o-facility through Channel Communication



ψ-calculus Essentials and How to Capture the Main Intuitions with them

Structural Operational Semantics for Parallel Agents

Transition Rules:

Schema

$$\frac{\text{premises}}{\text{conclusion}}$$

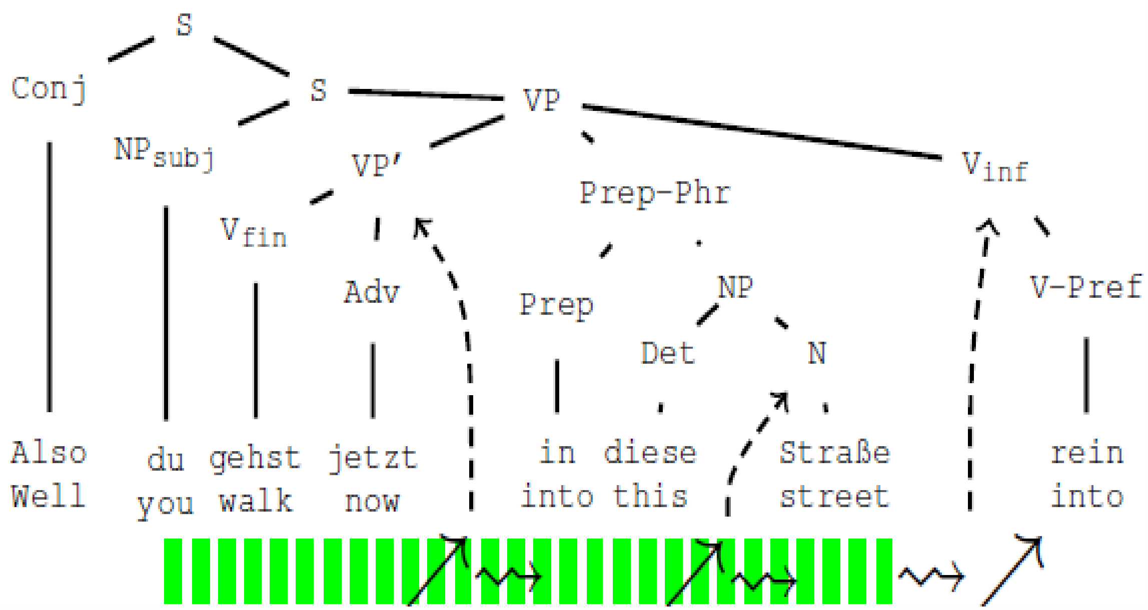
COM (simplified)

$$\frac{P \xrightarrow{\bar{a}b} P' \qquad Q \xrightarrow{\underline{a}b} Q'}{P|Q \xrightarrow{\tau} P'|Q'}$$

Contents

A λ-ψ-modelling of the Dialogue Example

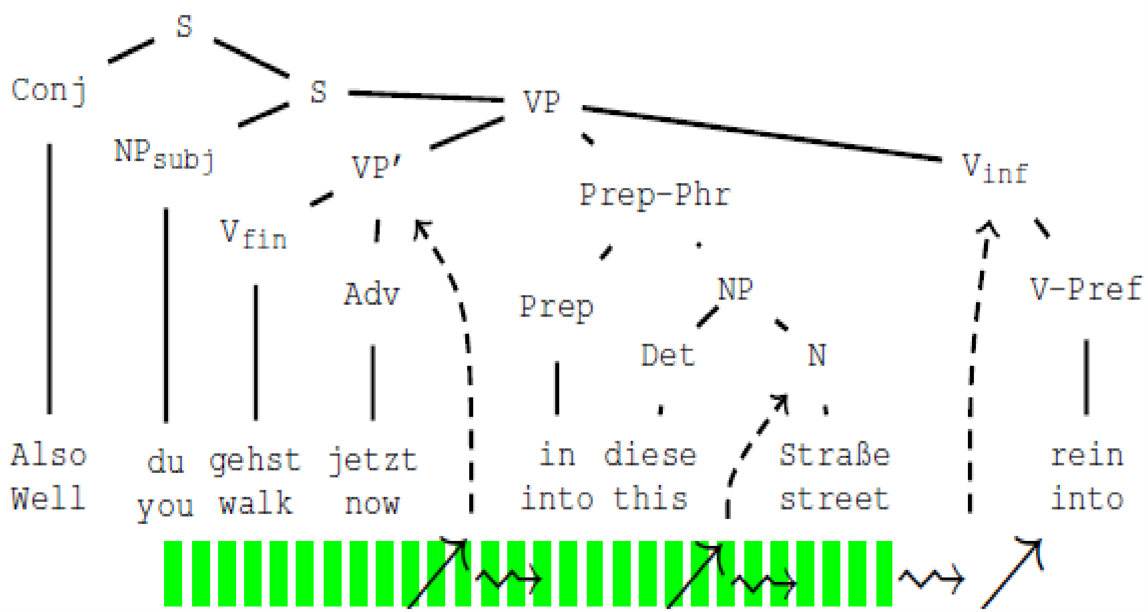
Empirical Foundations



- Syntax of the Follower's clarification request
- Winding gesture starts before possible integration points with speech and continues afterwards
- Gesture stroke: marked with a green dashed line

A λ-ψ-modelling of the Dialogue Example

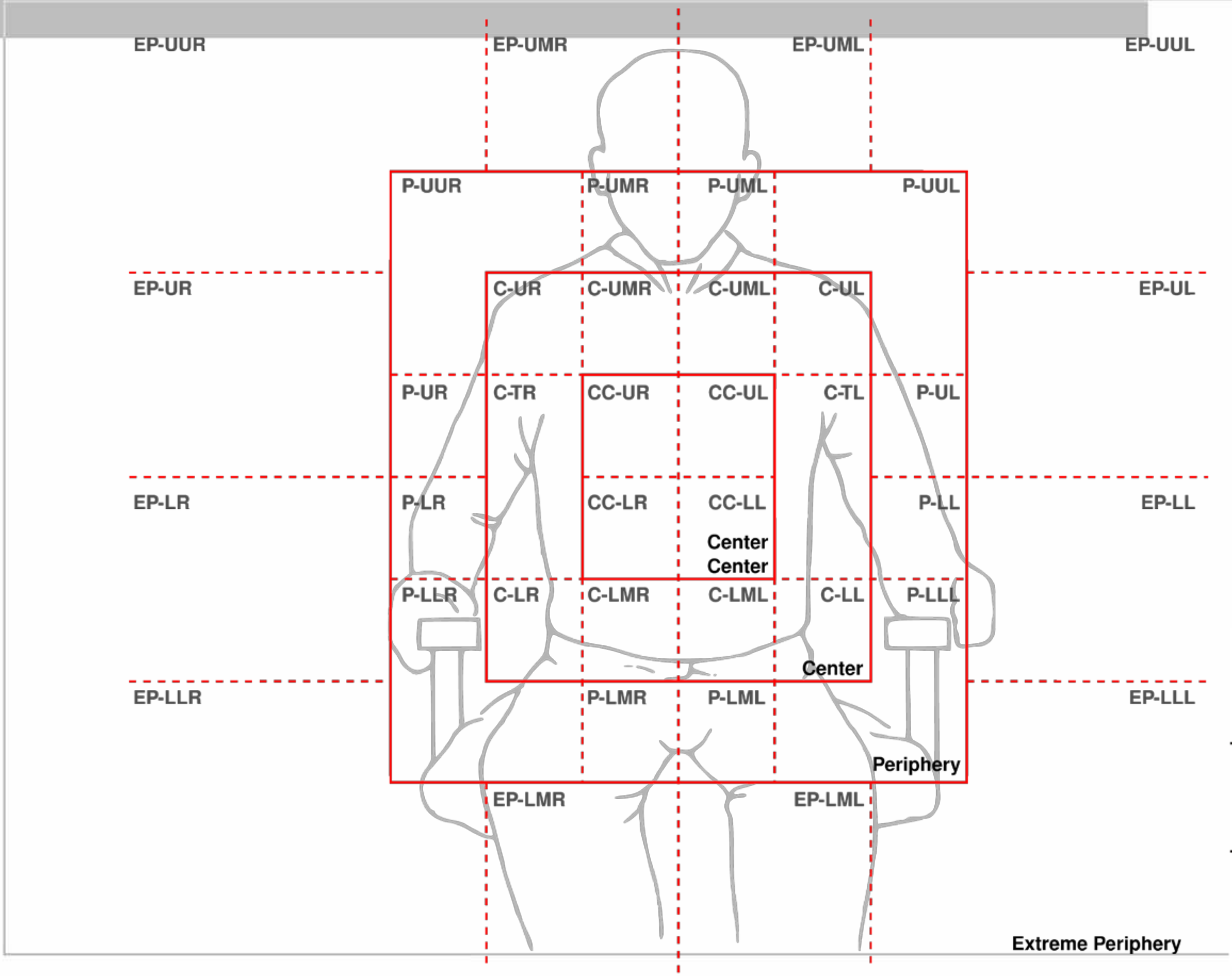
Empirical Foundations

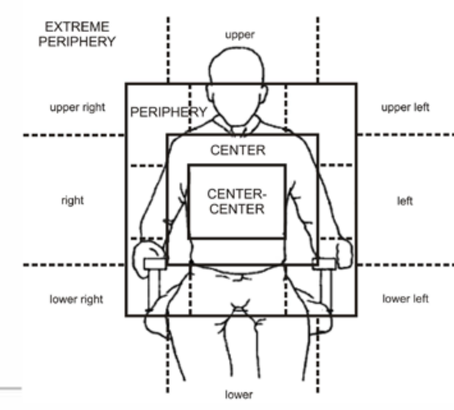


- Syntax structure: “German Satzklammer” (didactic English translation)
- Possible speech-gesture interface points “walk now”, “street” and “into”: marked with dashed arrows
- Winding arrow: indicates winding of gesture.

A λ-ψ-modelling of the Dialogue Example

Empirical Foundations





A λ-ψ-modelling of the Dialogue Example

Empirical Foundations

…	…
F.Right.Hand.Shape	B-spread
F.Right.Palm.Direction	PTB/PTL>PTB>PTL>PTB>PTB/PTL
F.Right.Palm.Movement	LINE
F.Right.Path.of.Palm.Direction	ML>MR>ML>MR
F.Right.Back.of.Hand.Direction	BAB/BTL>BTL>BAB>BAB/BTL
…	…
F.Right.Path.of.Wrist.Location	LINE
F.Right.WL.Movement.Direction	MF/ML
F.Right.Wrist.Position	C-RT
F.Right.Wrist.Distance	D-CE
...	...

Follower’s AVM of gesture annotation. Represented is stroke of gesture. Translated as “winding”

A λ-ψ-modelling of the Dialogue Example

Empirical Foundations

…	…
RG.Right.Hand.Shape	B-spread
RG.Right.Palm.Direction	PTL>PTB/PTL>PTL
RG.Right.Back.of.Hand.Direction	BAB>BAB/BTL>BAB/BUP
RG.Right.PathOfBack.of.Hand.Direction	MF>ML>MF>MR/MU
…	…
RG.Right.Wrist.Position	CC
RG.Right.Wrist.Distance	D-CE
…	…

Routegiver's AVM of first winding-gesture annotation. Represented is stroke of gesture. Translated as "winding"

A λ-ψ-modelling of the Dialogue Example

Empirical Foundations

…	…
RG.Right.Hand.Shape	B-spread
RG.Right.Palm.Direction	PTB/PTL>PTL>PTL
RG.Right.Path.Of.Palm.Direction	ARC
RG.Right.Back.of.Hand.Direction	BAB/BTL>BAB
RG.Right.Path.Of.Back.of.Hand.Direction	ARC>LINE
RG.Right.Back.of.Hand.Movement.Direction	MF>ML>MF>MR/MU
RG.Right.Path.Of.WristLocation	LINE>LINE
RG.Right.WristLocation.MovementDirection	ML>MF
RG.Right.Wrist.Position	CC
RG.Right.Wrist.Distance	D-CE
…	…

Routegiver’s AVM of second winding-gesture annotation. Represented is stroke of gesture. Translated as “winding”

Contents

A λ-ψ-modelling of the Dialogue Example

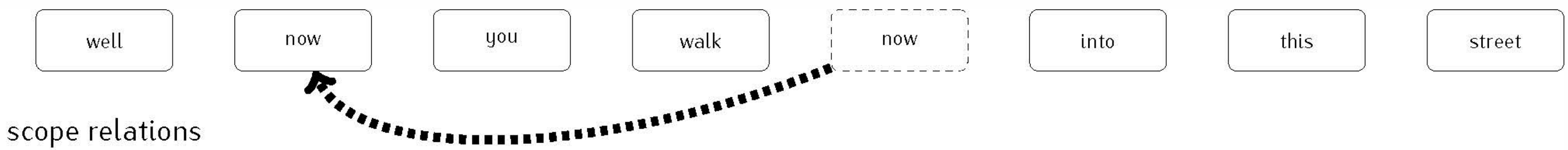
The λ-ψ-calculus Tailored to Suit

well | you | walk | now | into | this | street

incoming words

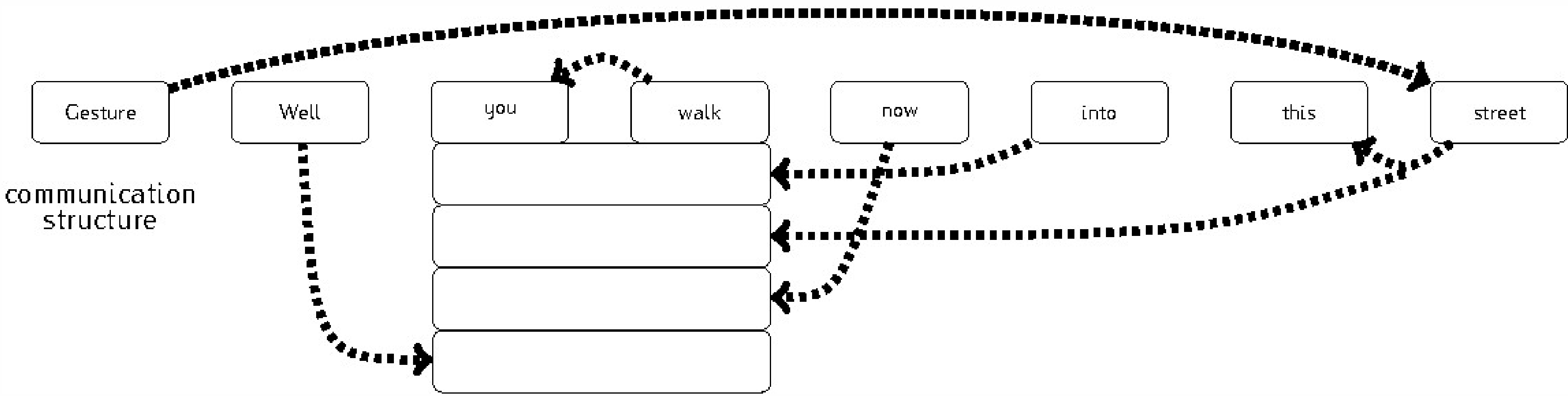
A λ-ψ-modelling of the Dialogue Example

The λ-ψ-calculus Tailored to Suit



A λ-ψ-modelling of the Dialogue Example

The λ-ψ-calculus Tailored to Suit



A λ-ψ-modelling of the Dialogue Example

The λ-ψ-calculus Tailored to Suit

Problem: Respect the order of the incoming gesture and words and model correct scope relations

The λ-ψ-calculus Tailored to Suit:

Definitions 1-3 in section 4 define a family of ψ-calculi with various open slots

A λ-ψ-modelling of the Dialogue Example

The λ-ψ-calculus Tailored to Suit

Now: set up a λ-ψ- calculus to model the asynchronous speech-gesture communication of the dialogue example

Cresswellian type structure

$T =_{def} N \cup Var \in Cat_i = \{p_{<0,0>}, u_1, f_{<0,1>}, f'_{<0,e,1>}, r_{<0,e',1>}, pr_{<0,0>}\} \cup$
$\{e_1, e_1', winding'_{<0,1>}, well'_{<0,0>}, you'_1, walk'_{<0,e,1>}, now'_{<0,0>}, into'_{<0,e',1>},$
$this'_{<1,<0,1>,1>}, street'_{<0,1>}\} \cup Var \in C$

$C =_{def} \{fb_{<0,<0,1>,1>} \leftrightarrow b_{<0,<0,1>,1>} \leftrightarrow b'_{<0,<0,1>,1>}, we_{<0,0>} \leftrightarrow we'_{<0,0>}, w_{<0,e,1>} \leftrightarrow$
$w'_{<0,e,1>}, i_{<0,e',1>} \leftrightarrow i'_{<0,e',1>} \leftrightarrow in_{<0,e',1>}, ts_1 \leftrightarrow ts'_1, nw_{<0,0>} \leftrightarrow nw'_{<0,0>} \leftrightarrow n_{<0,0>},$
$s_{<0,1>} \leftrightarrow s'_{<0,1>}\}$

$A =_{def}$ $\{1\}$

$\otimes =_{def}$ $\lambda\Psi 1, \Psi 2.\ 1$

$1 \vdash$ $\{x \leftrightarrow y \mid x, y \in C \text{ and } x, y \in Cat_i\}$

Contents

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

What the Speech-Gesture-Interaction Agent SGIA does:

- handles incrementality
- implements the intuitively correct scopes
- achieves the speech-gesture integration to produce a multi-modal meaning

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

“Street” solution

$$\mathrm{SGIA} =_{def} !\overline{fb}. <\lambda f\, \lambda u(f(u) \wedge \mathrm{winding}'(u))>.0 \mid \underline{we}\,(we'). <\lambda p(\mathrm{well}'(p))(we')>.0 \mid \underline{w}\,(w').\ \underline{i}\,(i').\ \underline{ts}\,(ts').\ \underline{nw}\,(nw').\ \overline{pr}.\ nw' <<< \lambda f\, \lambda ru\,(\lambda x\,(f(x, you') \wedge r(x, u))e)w'>i'>ts'>.0 \mid \overline{w}. < \mathrm{walk}'>.0 \mid \overline{n}. <\lambda p\ \mathrm{now}'(p)>.0 \mid \overline{in}.\ \mathrm{into}'.0 \mid \underline{s}(s').\overline{ts}. <\lambda g\,(\mathrm{this}'\ x\,(g(x)))s'.0> \mid \underline{b}(b').\ \overline{s}. <b' <\lambda x(\mathrm{street}'(x))>>.0$$

How SGIA works:

- Winding gesture: produced concurrently with the words <”well”, “you”, “walk”, “now”, “into”, “this”, “street”>

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

“Street” solution

$$\mathrm{SGIA} =_{def} !\overline{fb}.\ <\lambda f\ \lambda u(f(u) \wedge \mathrm{winding'}(u))>.0 \mid \underline{we}\ (we').\ <\lambda p(\mathrm{well'}(p))(we')>.0 \mid \underline{w}\ (w').\ \underline{i}\ (i').\ \underline{ts}\ (ts').\ \underline{nw}\ (nw').\ \overline{pr}.\ nw'<<< \lambda f\ \lambda ru\ (\lambda x\ (f(x, \mathrm{you'}) \wedge r(x, u))e)w'>i'>ts'>.0 \mid \overline{w}.< \mathrm{walk'}>.0 \mid \overline{n}.<\lambda p\ \mathrm{now'}(p)>.0 \mid \overline{in}.\ \mathrm{into'}.0 \mid \underline{s}(s').\overline{ts}.\ <\lambda g\ (\mathrm{this'}\ x\ (g(x)))s'.0> \mid \cancel{\underline{b}(b')}.\ \overline{s}.\ <\lambda u(\mathrm{street'}(u) \wedge \mathrm{winding'}(u))>.0$$

How SGIA works:

- Composable winding gesture representation sends its information to “street”, yielding “winding street”:

$$\underline{b}(b').\ \overline{s}.\ <b'\ <\lambda x(\mathrm{street'}(x))>>.0$$

$$\cancel{\underline{b}(b')}.\ \overline{s}.\ <\lambda f\ \lambda u(f(u) \wedge \mathrm{winding'}(u))\ \lambda x(\mathrm{street'}(x))>.0$$

$$\cancel{\underline{b}(b')}.\ \overline{s}.\ <\lambda u(\mathrm{street'}(u) \wedge \mathrm{winding'}(u))>.0$$

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

"Street" solution

SGIA $=_{def}$!$\overline{\text{fb}}$. <λf λu(f(u) ∧ winding'(u))>.0 | we (we'). <λp(well'(p))(we')>.0| w (w'). i (i'). ts (ts'). nw (nw'). $\overline{\text{pr}}$. nw'<<< λf λru (λx (f(x, you') ∧ r(x, u))e)w'>i'>ts'>.0 | $\overline{\text{w}}$. < walk'>.0 | $\overline{\text{n}}$. <λp now'(p)>.0 | $\overline{\text{in}}$. into'.0| s~~(s')~~.$\overline{\text{ts}}$. this' x (street'(x) ∧ winding'(x)).0 | b~~(b'). $\overline{\text{s}}$. <λu(street'(u) ∧ winding'(u))>~~.0

How SGIA works:

- Property "winding street" sends its information to "this" and we get the referring expression "this winding street":

s~~(s')~~.$\overline{\text{ts}}$. λg (this' x (g(x))) λu(street'(u) ∧ winding'(u)).0

s~~(s')~~.$\overline{\text{ts}}$. this' x (street'(x) ∧ winding'(x)).0

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

"Street" solution

SGIA $=_{def}$ $!\overline{fb}$. <λf λu(f(u) ∧ winding'(u))>.0 | $\underline{we}$ (we'). <λp(well'(p))(we')>.0| $\underline{w}$ (w'). $\underline{i}$ (i'). $\underline{ts}$ (ts'). $\underline{nw}$ (nw'). $\overline{pr}$. nw'<<< λf λru (λx (f(x, you') ∧ r(x, u))e)w'>i'>ts'>.0 | $\overline{w}$. < walk'>.0 | $\overline{n}$. <λp now'(p)>.0 | $\overline{in}$. into'.0| ~~$\underline{s}$(s')~~.$\overline{ts}$. this' x (street'(x) ∧ winding'(x)).0 | ~~$\underline{b}$(b'). $\overline{s}$. <λu(street'(u) ∧ winding'(u))>~~.0

How SGIA works:

- Information tied to "you": propositional function:

| $\underline{w}$ (w'). $\underline{i}$ (i'). $\underline{ts}$ (ts'). $\underline{nw}$ (nw'). $\overline{pr}$. nw'<<< λf λru (λx (f(x, you') ∧ r(x,

u))e)w'>i'>ts'>.0

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

“Street” solution

SGIA $=_{def}$ $!\overline{fb}$. <λf λu(f(u) ∧ winding’(u))>.0 | $\underline{we}$ (we‘). <λp(well’(p))(we’)>.0| $\underline{w}$ (w’). $\underline{i}$ (i’). $\underline{ts}$ (ts′). $\underline{nw}$ (nw’). $\overline{pr}$. nw’<<< λf λru (λx (f(x, you’) ∧ r(x, u))e)w‘>i‘>ts‘>.0 | $\overline{w}$. < walk’>.0 | $\overline{n}$. <λp now’(p)>.0 | $\overline{in}$. into‘.0| $\underline{s}$~~(s’)~~.$\overline{ts}$. this’ x (street’(x) ∧ winding’(x)).0 | $\underline{b}$~~(b’). $\overline{s}$. <λu(street’(u) ∧ winding’(u))>~~.0

How SGIA works:

needs a lot of constant information

- a relation “walk” defined on an event *e* and a subject “you”:

| $\overline{w}$. < walk’>.0

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

“Street” solution

$$SGIA =_{def}\ !\overline{fb}.<\lambda f\,\lambda u(f(u) \wedge winding'(u))>.0 \mid \underline{we}\,(we').<\lambda p(well'(p))(we')>.0 \mid \underline{w}\,(\cancel{w'}).\ \underline{i}\,(i').\ \underline{ts}\,(ts').\ \underline{nw}\,(nw').\ \overline{pr}.\ nw'<<<\lambda f\,\lambda ru\,(\lambda x\,(f(x, you') \wedge r(x, u))e)\ walk'>i'>ts'>.0 \mid \cancel{\overline{w}.< walk'>}.0 \mid \overline{n}.<\lambda p\ now'(p)>.0 \mid \overline{in}.\ into'.0 \mid \underline{s}\cancel{(s')}.\overline{ts}.\ this'\ x\ (street'(x) \wedge winding'(x)).0 \mid \cancel{\underline{b}(b').\ \overline{s}.\ <\lambda u(street'(u) \wedge winding'(u))>}.0$$

How SGIA works:

needs a lot of constant information

- a relation “walk” defined on an event *e* and a subject “you”

$\mid \overline{w}.< walk'>.0$

$\cancel{\underline{w}}\ (\cancel{w'}).\ \underline{i}\,(i').\ \underline{ts}\,(ts').\ \underline{nw}\,(nw').\ \overline{pr}.\ nw'<<<\lambda f\,\lambda ru\,(\lambda x\,(f(x, you') \wedge r(x, u))e)\ walk'>i'>ts'>.0$

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

"Street" solution

$$\mathrm{SGIA} =_{def}\ !\overline{fb}.\ <\lambda f\ \lambda u(f(u) \wedge \mathrm{winding}'(u))>.0 \mid \underline{we}\ (we').\ <\lambda p(\mathrm{well}'(p))(we')>.0 \mid \underline{w}\ (\cancel{w'}).\ \underline{i}\ (i').\ \underline{ts}\ (ts').\ \underline{nw}\ (nw').\ \overline{pr}.\ nw'<<< \lambda ru\ (\lambda x\ (\mathrm{walk}'(x, \mathrm{you}') \wedge r(x, u))e) >i'>ts'>.0 \mid \cancel{\overline{w}.\ < \mathrm{walk}'>}.0 \mid \overline{n}.\ <\lambda p\ \mathrm{now}'(p)>.0 \mid \overline{in}.\ \mathrm{into}'.0 \mid \underline{s}(\cancel{s'}).\overline{ts}.\ \mathrm{this}'\ x\ (\mathrm{street}'(x) \wedge \mathrm{winding}'(x)).0 \mid \cancel{\underline{b}(b').\ \overline{s}.\ <\lambda u(\mathrm{street}'(u) \wedge \mathrm{winding}'(u))>}.0$$

How SGIA works:

needs a lot of constant information

- a relation "walk" defined on an event *e* and a subject "you"

$$\mid \overline{w}.< \mathrm{walk}'>.0$$

$$\cancel{\underline{w}\ (w')}.\ \underline{i}\ (i').\ \underline{ts}\ (ts').\ \underline{nw}\ (nw').\ \overline{pr}.\ nw'<<< \lambda ru\ (\lambda x\ (\mathrm{walk}'(x, \mathrm{you}') \wedge r(x, u))e) >i'>ts'>.0$$

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

"Street" solution

SGIA $=_{def}$!$\overline{\text{fb}}$. <λf λu(f(u) ∧ winding'(u))>.0 | we (we'). <λp(well'(p))(we')>.0| w (~~w'~~). i ~~(i')~~. ts (ts′). nw (nw'). $\overline{\text{pr}}$. nw'<<< λru (λx (walk'(x, you') ∧ r(x, u))e) >into'>ts'>.0 | ~~$\overline{\text{w}}$. < walk'>~~.0 | $\overline{\text{n}}$. <λp now'(p)>.0 | $\overline{\text{in}}$. into'.0| s~~(s')~~.$\overline{\text{ts}}$. this' x (street'(x) ∧ winding'(x)).0 | b~~(b'). $\overline{\text{s}}$. <λu(street'(u) ∧ winding'(u))>~~.0

How SGIA works:

| $\overline{\text{w}}$. < walk'>.0

~~w (w'~~). i (i'). ts (ts′). nw (nw'). $\overline{\text{pr}}$. nw'<<< λru (λx (walk'(x, you') ∧ r(x, u))e) >i'>ts'>.0

= i (i'). ts (ts′). nw (nw'). $\overline{\text{pr}}$. nw'<<< λru (λx (walk'(x, you') ∧ r(x, u))e) >i'>ts'>.0

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

"Street" solution

$$\mathrm{SGIA} =_{def}\ !\overline{fb}.\ <\lambda f\ \lambda u(f(u) \wedge \mathrm{winding}'(u))>.0 \mid \underline{we}\ (we').\ <\lambda p(\mathrm{well}'(p))(we')>.0 \mid \underline{w}\ (\cancel{w'}).\ \underline{i}\ \cancel{(i')}.\ \underline{ts}\ (ts').\ \underline{nw}\ (nw').\ \overline{pr}.\ nw'<<<\ \lambda ru\ (\lambda x\ (\mathrm{walk}'(x, you') \wedge r(x, u))e) >\mathrm{into}'>ts'>.0 \mid \cancel{\overline{w}.\ <\ \mathrm{walk}'>}.0 \mid \overline{n}.\ <\lambda p\ \mathrm{now}'(p)>.0 \mid \cancel{\overline{in}.\ \mathrm{into}'}.0 \mid \underline{s}\ \cancel{(s')}.\overline{ts}.\ \mathrm{this}'\ x\ (\mathrm{street}'(x) \wedge \mathrm{winding}'(x)).0 \mid \cancel{\underline{b}(b').\ \overline{s}.\ <\lambda u(\mathrm{street}'(u) \wedge \mathrm{winding}'(u))>}.0$$

How SGIA works:

- a relation "into" defined on an event *e:*

$$\cancel{\overline{in}.\ \mathrm{into}'}.0$$

$$\underline{i}\ \cancel{(i')}.\ \underline{ts}\ (ts').\ \underline{nw}\ (nw').\ \overline{pr}.\ nw'<<<\ \lambda ru\ (\lambda x\ (\mathrm{walk}'(x, you') \wedge r(x, u))e)$$

$$>\mathrm{into}'>ts'>.0$$

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

"Street" solution

SGIA $=_{def}$ $!\overline{\text{fb}}$. <λf λu(f(u) ∧ winding'(u))>.0 **|** $\underline{\text{we}}$ (we'). <λp(well'(p))(we')>.0**|** $\underline{\text{w}}$ (~~w'~~). $\underline{\text{i}}$ ~~(i')~~. $\underline{\text{ts}}$ (ts'). $\underline{\text{nw}}$ (nw'). $\overline{\text{pr}}$. nw'<<< λu (λx (walk'(x, you') ∧ into'(x, u))e) >>ts'>.0 **|** ~~$\overline{\text{w}}$. < walk'>~~.0 **|** $\overline{\text{n}}$. <λp now'(p)>.0 **|** ~~$\overline{\text{in}}$. into'~~.0**|** $\underline{\text{s}}$~~(s')~~.$\overline{\text{ts}}$. this' x (street'(x) ∧ winding'(x)).0 **|** $\underline{\text{b}}$~~(b'). $\overline{\text{s}}$. <λu(street'(u) ∧ winding'(u))>~~.0

How SGIA works:

- a relation "into" defined on an event *e:*

~~$\overline{\text{in}}$. into'~~.0

$\underline{\text{i}}$ ~~(i')~~. $\underline{\text{ts}}$ (ts'). $\underline{\text{nw}}$ (nw'). $\overline{\text{pr}}$. nw'<<< λu (λx (walk'(x, you') ∧ into' (x, u))e)

>>ts'>.0

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

"Street" solution

$$\mathrm{SGIA} =_{def} !\overline{fb}.\ \langle \lambda f\ \lambda u(f(u) \wedge \mathrm{winding}'(u))\rangle.0 \ |\ \underline{we}\ (we').\ \langle \lambda p(\mathrm{well}'(p))(we')\rangle.0 \ |\ \underline{w}\ (\cancel{w'}).\ \cancel{\underline{i}\ (i')}.\ \cancel{\underline{ts}\ (ts')}.\ \underline{nw}\ (nw').\ \overline{pr}.\ nw'\langle\langle\langle \lambda u\ (\lambda x\ (\mathrm{walk}'(x, \mathrm{you}') \wedge \mathrm{into}'(x, u))e)\ \rangle\rangle \mathrm{this}'\ x\ (\mathrm{street}'(x) \wedge \mathrm{winding}'(x))\rangle.0 \ |\ \cancel{\overline{w}.\ \langle\ \mathrm{walk}'\rangle}.0 \ |\ \overline{n}.\ \langle \lambda p\ \mathrm{now}'(p)\rangle.0 \ \cancel{|\ \overline{in}.\ \mathrm{into}'}.0 \ |\ \underline{s}(s').\cancel{\overline{ts}}.\ \cancel{\mathrm{this}'\ x\ (\mathrm{street}'(x) \wedge \mathrm{winding}'(x))}.0 \ |\ \underline{b}(b').\ \cancel{\overline{s}.\ \langle \lambda u(\mathrm{street}'(u) \wedge \mathrm{winding}'(u))\rangle}.0$$

How SGIA works:

- the referring expression "this winding street"

$\cancel{\overline{ts}}.\ \mathrm{this}'\ x\ (\mathrm{street}'(x) \wedge \mathrm{winding}'(x)).0$

$\cancel{\underline{ts}\ (ts')}.\ \underline{nw}\ (nw').\ \overline{pr}.\ nw'\langle\langle \lambda u\ (\lambda x\ (\mathrm{walk}'(x, \mathrm{you}') \wedge \mathrm{into}'(x, u))e)\ \langle \mathrm{this}'\ x\ (\mathrm{street}'(x) \wedge \mathrm{winding}'(x))\rangle\rangle.0$

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

“Street” solution

SGIA $=_{def}$!$\overline{\text{fb}}$. <λf λu(f(u) ∧ winding’(u))>.0 | $\underline{\text{we}}$ (we‘). <λp(well’(p))(we’)>.0| $\underline{\text{w}}$ (~~w’~~). ~~$\underline{\text{i}}$ (i’)~~. ~~$\underline{\text{ts}}$ (ts’)~~. $\underline{\text{nw}}$ (nw’). $\overline{\text{pr}}$. nw’<<< (λx (walk’(x, you’) ∧ into‘(x, this’ x (street’(x) ∧ winding’(x))))e) >>>.0 | ~~$\overline{\text{w}}$. < walk’>~~.0 | $\overline{\text{n}}$. <λp now’(p)>.0 | ~~$\overline{\text{in}}$. into‘~~.0| $\underline{\text{s}}$(~~s’~~).$\overline{\text{ts}}$. ~~this’ x (street’(x) ∧ winding’(x))~~.0 | $\underline{\text{b}}$~~(b’). $\overline{\text{s}}$. <λu(street’(u) ∧ winding’(u))>~~.0

How SGIA works:

- the referring expression “this winding street”

~~$\underline{\text{ts}}$ (ts’)~~. $\underline{\text{nw}}$ (nw’). $\overline{\text{pr}}$. nw’< λu (λx (walk’(x, you’) ∧ into‘(x, u))e) <<this’ x

(street’(x) ∧ winding’(x))>>>.0

= ~~$\underline{\text{ts}}$ (ts’)~~. $\underline{\text{nw}}$ (nw’). $\overline{\text{pr}}$. nw’< (λx (walk’(x, you’) ∧ into‘(x, this’ x

(street’(x) ∧ winding’(x))))e) >.0

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

"Street" solution

SGIA $=_{def}$!$\overline{fb}$. <λf λu(f(u) ∧ winding'(u))>.0 | $\underline{we}$ (we'). <λp(well'(p))(we')>.0| $\underline{w}$ (~~w'~~). ~~$\underline{i}$ (i')~~. ~~$\underline{ts}$ (ts')~~. $\underline{nw}$ (nw'). $\overline{pr}$. nw'<<< (walk'(e, you') ∧ into'(e, this' x (street'(x) ∧ winding'(x)))) >>>.0 | ~~$\overline{w}$. < walk'>~~.0 | $\overline{n}$. <λp now'(p)>.0 | ~~$\overline{in}$. into'~~.0| $\underline{s}$~~(s')~~.~~$\overline{ts}$. this' x (street'(x) ∧ winding'(x))~~.0 | ~~$\underline{b}$(b'). $\overline{s}$. <λu(street'(u) ∧ winding'(u))>~~.0

How SGIA works:

- event introduction:

  $\underline{nw}$ (nw'). $\overline{pr}$. nw'<<< (walk'(e, you') ∧ into'(e, this' x (street'(x) ∧ winding'(x)))) >>>.0

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

“Street” solution

SGIA $=_{def}$!$\overline{fb}$. <λf λu(f(u) ∧ winding’(u))>.0 | $\underline{we}$ (we‘). <λp(well’(p))(we’)>.0| $\underline{w}$ (~~w’~~). ~~$\underline{i}$ (i’)~~. ~~$\underline{ts}$ (ts’)~~. $\underline{nw}$ (nw’). $\overline{pr}$. nw’<<< (walk’(e, you’) ∧ into‘(e, this’ x (street’(x) ∧ winding’(x)))) >>>.0 | ~~$\overline{w}$. < walk’>~~.0 | $\overline{n}$. <λp now’(p)>.0 ~~| $\overline{in}$. into‘~~.0| ~~$\underline{s}$(s’)~~.~~$\overline{ts}$~~. ~~this’ x (street’(x) ∧ winding’(x))~~.0 | ~~$\underline{b}$(b’). $\overline{s}$. <λu(street’(u) ∧ winding’(u))>~~.0

How SGIA works:

- Resulting term: (walk’(e, you’) ∧ into‘(e, this’ x (street’(x) ∧ winding’(x)))) is the proposition
- “now”: $\overline{n}$. <λp now’(p)>.0 looks for and with which it combines to yield another proposition:

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

“Street” solution

$$\mathrm{SGIA} =_{def}\ !\overline{fb}.\ <\lambda f\, \lambda u(f(u) \wedge \mathrm{winding}'(u))>.0 \mid \underline{we}\ (we').\ <\lambda p(\mathrm{well}'(p))(we')>.0 \mid \underline{w}\ (\cancel{w'}).\ \cancel{\underline{i}\ (i')}.\ \cancel{\underline{ts}\ (ts')}.\ \cancel{\underline{nw}\ (nw')}.\ \overline{pr}.\ \lambda p\ \mathrm{now}'(p)<<< (\mathrm{walk}'(e, you') \wedge \mathrm{into}'(e, \mathrm{this}'\ x\ (\mathrm{street}'(x) \wedge \mathrm{winding}'(x)))) >>>.0 \mid \cancel{\overline{w}.\ < \mathrm{walk}'>}.0 \mid \cancel{\overline{n}.\ <\lambda p\ \mathrm{now}'(p)>}.0 \mid \cancel{\overline{in}.\ \mathrm{into}'}.0 \mid \underline{s}(s').\overline{ts}.\ \cancel{\mathrm{this}'\ x\ (\mathrm{street}'(x) \wedge \mathrm{winding}'(x))}.0 \mid \underline{b}(b').\ \cancel{\overline{s}.\ <\lambda u(\mathrm{street}'(u) \wedge \mathrm{winding}'(u))>}.0$$

How SGIA works:

$$\cancel{\overline{n}.\ <\lambda p\ \mathrm{now}'(p)>}.0$$

$$\cancel{\underline{nw}\ (nw')}.\ \overline{pr}.\ \lambda p\ \mathrm{now}'(p)\ (\mathrm{walk}'(e, you') \wedge \mathrm{into}'(e, \mathrm{this}'\ x\ (\mathrm{street}'(x) \wedge \mathrm{winding}'(x)))).0$$

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

“Street” solution

SGIA $=_{def}$!$\overline{\text{fb}}$. <λf λu(f(u) ∧ winding’(u))>.0 | $\underline{\text{we}}$ (we‘). <λp(well’(p))(we’)>.0| $\underline{\text{w}}$ (~~w~~’). ~~$\underline{\text{i}}$ (i’)~~. ~~$\underline{\text{ts}}$ (ts’)~~. ~~$\underline{\text{nw}}$ (nw’)~~. $\overline{\text{pr}}$. <now’(walk’(e, you’) ∧ into‘(e, this’ x (street’(x) ∧ winding’(x)))) >.0 | ~~$\overline{\text{w}}$. < walk’>~~.0 | ~~$\overline{\text{n}}$. <λp now’(p)>~~.0 ~~| $\overline{\text{in}}$. into‘~~.0| $\underline{\text{s}}$~~(s’)~~.~~$\overline{\text{ts}}$~~. ~~this’ x (street’(x) ∧ winding’(x))~~.0 | $\underline{\text{b}}$~~(b’). $\overline{\text{s}}$. <λu(street’(u) ∧ winding’(u))>~~.0

How SGIA works:

~~$\overline{\text{n}}$. <λp now’(p)>~~.0

~~$\underline{\text{nw}}$ (nw’)~~. $\overline{\text{pr}}$. <now’(walk’(e, you’) ∧ into‘(e, this’ x (street’(x) ∧ winding’(x))))>.0

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

“Street” solution

SGIA $=_{def}$!$\overline{\text{fb}}$. <λf λu(f(u) ∧ winding’(u))>.0 | $\underline{\text{we}}$ (we‘). <λp(well’(p))(we’)>.0| $\underline{\text{w}}$ (~~w’~~). ~~$\underline{\text{i}}$ (i’)~~. ~~$\underline{\text{ts}}$ (ts’)~~. ~~$\underline{\text{nw}}$ (nw’)~~. $\overline{\text{pr}}$. <now’(walk’(e, you’) ∧ into‘(e, this’ x (street’(x) ∧ winding’(x)))) >.0 | ~~$\overline{\text{w}}$. < walk’>~~.0 | ~~$\overline{\text{n}}$. <λp now’(p)>~~.0 | ~~$\overline{\text{in}}$. into‘~~.0| $\underline{\text{s}}$~~(s’)~~.~~$\overline{\text{ts}}$~~. ~~this’ x (street’(x) ∧ winding’(x))~~.0 | $\underline{\text{b}}$~~(b’). $\overline{\text{s}}$. <λu(street’(u) ∧ winding’(u))>~~.0

How SGIA works:

- New proposition: put on an outgoing channel, combines with “well”, again generating a proposition:

  ~~$\overline{\text{pr}}$.~~ <now’(walk’(e, you’) ∧ into‘(e, this’ x (street’(x) ∧ winding’(x)))) >.0
  ~~$\underline{\text{we}}$ (we‘)~~. <λp(well’(p))(we’)>.0

= <λp(well’(p))(now’(walk’(e, you’) ∧ into‘(e, this’ x (street’(x) ∧ winding’(x)))))>.0

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

“Street” solution

SGIA $=_{def}$!fb. <λf λu(f(u) ∧ winding’(u))>.0 | ~~we (we‘)~~. <(well’(now’(walk’(e, you’) ∧ into‘(e, this’ x (street’(x) ∧ winding’(x))))))))>.0| w (~~w’~~). ~~i (i’)~~. ~~ts (ts’)~~. ~~nw (nw’)~~. ~~pr. <now’(walk’(e, you’) ∧ into‘(x, this’ x (street’(x) ∧ winding’(x)))) >~~.0 | ~~w. < walk’>~~.0 | ~~n. <λp now’(p)>~~.0 | ~~in. into‘~~.0| s(s’).ts. ~~this’ x (street’(x) ∧ winding’(x))~~.0 | b~~(b’). s. <λu(street’(u) ∧ winding’(u))>~~.0

How SGIA works:

~~pr.~~ <now’(walk’(e, you’) ∧ into‘(e, this’ x (street’(x) ∧ winding’(x)))) >.0

~~we (we‘)~~. <λp(well’(p))(we’)>.0

= <λp(well’(p))(now’(walk’(e, you’) ∧ into‘(e, this’ x (street’(x) ∧

winding’(x)))))>.0

(well’(now’(walk’(e, you’) ∧ into‘(e, this’ x (street’(x) ∧ winding’(x)))))).0

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

“Street” solution

SGIA $=_{def}$!$\overline{\text{fb}}$. <λf λu(f(u) ∧ winding’(u))>.0 | ~~we (we‘)~~. <(well’(now’(walk’(e, you’) ∧ into‘(x, this’ x (street’(x) ∧ winding’(x)))))))>.0| w (~~w’~~). ~~i (i’)~~. ~~ts (ts’)~~. ~~nw (nw’)~~. ~~pr. <now’(walk’(e, you’) ∧ into‘(x, this’ x (street’(x) ∧ winding’(x)))) >~~.0 | ~~w. < walk’>~~.0 | ~~n. <λp now’(p)>~~.0 | ~~in. into‘~~.0| s(~~s’~~).~~ts. this’ x (street’(x) ∧ winding’(x))~~.0 | b(b’). ~~s. <λu(street’(u) ∧ winding’(u))>~~.0

How SGIA works:

- winding gesture continues to be held. So we have at the end of the process communication:

!$\overline{\text{fb}}$. <λf λu(f(u) ∧ winding’(u))>.0 | well’(now’(walk’(e, you’) ∧ into‘(e, this’ x (street’(x) ∧ winding’(x))))).0 |

| .0 null-agent due to propositional function
| .0 null-agent due to walk’
| .0 null-agent due to now’

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

"Street" solution

SGIA $=_{def}$!$\overline{\text{fb}}$. <λf λu(f(u) ∧ winding'(u))>.0 | ~~$\underline{\text{we}}$ (we')~~. <(well'(now'(walk'(e, you') ∧ into'(x, this' x (street'(x) ∧ winding'(x)))))))>.0| $\underline{\text{w}}$ (~~w'~~). ~~$\underline{\text{i}}$ (i')~~. ~~$\underline{\text{ts}}$ (ts')~~. $\underline{\text{nw}}$ ~~(nw')~~. ~~$\overline{\text{pr}}$. <now'(walk'(e, you') ∧ into'(x, this' x (street'(x) ∧ winding'(x)))) >~~.0 | ~~$\overline{\text{w}}$. < walk'>~~.0 | ~~$\overline{\text{n}}$. <λp now'(p)>~~.0 | ~~$\overline{\text{in}}$. into'~~.0| ~~$\underline{\text{s}}$(s')~~.~~$\overline{\text{ts}}$~~. ~~this' x (street'(x) ∧ winding'(x))~~.0 | $\underline{\text{b}}$(b'). ~~$\overline{\text{s}}$. <λu(street'(u) ∧ winding'(u))>~~.0

!$\overline{\text{fb}}$. <λf λu(f(u) ∧ winding'(u))>.0 | well'(now'(walk'(e, you') ∧ into'(x, this' x (street'(x) ∧ winding'(x))))).0 |

| .0 null-agent due to propositional function
| .0 null-agent due to walk'
| .0 null-agent due to now'
| .0 null-agent due to into'
| .0 null-agent due to this' winding' street'
| .0 null-agent due to winding' street'

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

Cooper's Conjecture: Event Reading Idea

Idea:

Gesture is modifying the event in the propositional function

Consequences:

- Extend the propositional function
- Change the type of the gesture channel
- Change the speech-gesture contact point (interacting earlier on)

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

Cooper’s Conjecture: Event Reading Idea

SGIA $=_{def}$!$\overline{\text{fb}}$. <λu(winding’(u)>.0 **|** $\underline{\text{we}}$ (we‘). <λp(well’(p))(w’)>.0**|**$\underline{\text{wind}}$(wind’). $\underline{\text{w}}$ (w’). $\underline{\text{i}}$ (i’). $\underline{\text{ts}}$ (ts′). $\underline{\text{nw}}$ (nw’). $\overline{\text{pr}}$. nw’<<< λgfru∃e(f’(e, you’) ∧ r(e, u) ∧ g(e)) wind’)w‘>i‘>ts‘>.0 **|** $\overline{\text{w}}$. < walk’>.0 **|** $\overline{\text{n}}$. <λp now’(p).0> **|** $\overline{\text{in}}$. into‘.0**|** $\underline{\text{s}}$(s’).$\overline{\text{ts}}$. <<λg (this x (g(x)) s’>.0**|** $\underline{\text{b}}$(b’). $\overline{\text{s}}$. <b’ <λx(street’(x))>>.0

Winding’ sends its information via $\overline{\text{fb}}$ and $\underline{\text{wind}}$

- to the event e in the propositional function
- λgfru∃e(f’(e, you’) ∧ r(e, u) ∧ g(e)) wind’)w‘>i‘>ts‘>.0 and
- a winding event winding’(e) is created.

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

Modelling of Context

Contexts are called “environments”. $\Psi \rhd P \xrightarrow{\alpha} P'$

In an environment that asserts Ψ P can perform an action α leading to P’.

“$\rhd$” behaves similar to Situation Semantic’s “$\models$”.

Environmental assertions Ψ express effect of environment upon agent:

- conditions on agents and channels
- enabling interactions between parallel agents
- enabling conditions for case

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

Structural Operational Semantics for Parallel Agents Reconsidered

First step in derivation of example:

$$\Psi \otimes \Psi_P \otimes \Psi_Q \vdash fb \leftrightarrow b$$

$$\frac{\Psi_Q \otimes \Psi \rhd P \xrightarrow{\overline{fb}(<\lambda f\, \lambda u(f(u) \wedge \text{ winding}'(u))>)} P' \quad \Psi_P \otimes \Psi \rhd Q \xrightarrow{\underline{b}\,(b').\bar{s}.<b'<\text{street}'(x)\gg} Q'}{\Psi \rhd P|Q \xrightarrow{\tau} P'|Q'}$$

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

Exploiting Non-determinacy

Assume gesture attachment as indicated in the intuitive syntax picture, i.e. the winding gesture can go with

- “street”
- with the preposition “into”

- or (by Cooper’s Conjecture) with the whole event as expressed in the propositional function

then we can use the case construct to express non-determinacy.

News from λ-ψ: The Speech-Gesture-Interaction Agent SGIA

Exploiting Non-determinacy

Assume for simplicity that φ_i T, so all readings are equally probable.

Then we could have

Case T: $SGIA_{street`}$
T: $SGIA_{into'}$
T: $SGIA_e$

Gesture representation and channel conditions would have to be adjusted accordingly. Observe that we get different agents.

Also possible: Rank the φ_is.

- Probabilistic extensions of process algebras
- Process algebras using priorities.

Contents

Conclusion and Suggestions for Further Research

How to fit in dialogue:

Sketch of how to treat back-channels as agents/processes.

Example from Esghi, Howes, Gregoromichelaki, Hough, and Purver (2015):

A: The doctor
B: Chorlton?
A: No, Fitzgerald.
B: uh-huh.
A: he examined me.

- We have a clarification dialogue for the reference of “the doctor”.
- We need A- and B-processes, which are allowed to interact i-o-wise.
    - We have to make sure that we get alternating dialogue contributions.

Conclusion and Suggestions for Further Research

A	B
$\overline{\mathrm{Out}}.\ \langle \iota x\ \exists s(\mathrm{doctor}'(x,s)) \rangle$	$\underline{\mathrm{In}}$. thd. case: $(d \in D_B = \mathrm{thd})$: $\overline{\mathrm{Out}}.\ \langle$”mhm”$\rangle$.0 $\neg\ (d = \mathrm{thd})$: $\overline{\mathrm{Out}}.\ \langle ?(\mathrm{Chorlton} = \mathrm{thd}) \rangle$
$\underline{\mathrm{In}}$. repl. case: repl = “mhm”: $\overline{\mathrm{Out}}. < \lambda x\ (he = x\ \wedge\ \mathrm{examined}(x, A)) > .0$ repl = ?(Chorlton = thd): case: $\iota x\ \exists s\big(\mathrm{doctor}'(x,s)\big) = \mathrm{Chorlton}$: $\overline{\mathrm{Out}}. < \text{"mhm"} > .0$ $\neg\ (\iota x\ \exists s\big(\mathrm{doctor}'(x,s)\big) = \mathrm{Chorlton})$. $\overline{\mathrm{Out}}.\ \langle$”no”$\rangle$.0	$\underline{\mathrm{In}}$. rep. case: (rep = “no”): $\neg\ (\mathrm{Chorlton} = \mathrm{thd})$
$\overline{\mathrm{Out}}. < \iota x\ \exists s\big(\mathrm{doctor}'(x,s)\big) = \mathrm{Fitzgerald} >$	$\underline{\mathrm{In}}$. thd. case: $\iota x\ \exists s\big(\mathrm{doctor}'(x,s)\big) = \mathrm{Fitzgerald}$. $\overline{\mathrm{Out}}.\ \langle$”uh-huh”$\rangle$
$\underline{\mathrm{In}}$.repl. case: repl = ”uh-huh”: $\overline{\mathrm{Out}}. < \lambda x\ (he = x \wedge \mathrm{examined}(x, A)) > .0$	

Conclusion and Suggestions for Further Research

- Grounding (including larger context) would have to be achieved *via* a common-belief and individual-belief model and a domain D starting with CPs A, B and the doctor (all done incrementally).

- Observe that a generalization is still missing here:
  It would have to be "reference resolution" or some such using a mechanism like
  - R. Cooper's IntegrateOwn/OtherAcknowledgement or
  - J. Ginzburgh's QUD

References

Alahverdzhieva, K. and Lascarides, A. (2010). Analysing language and co-verbal gesture in constraint-based grammars. In Müller, St., editor, *Proceedings of the 17th International Conference on Head-Driven Phase Structure Grammar (HPSG)*, pp. 5–25, Paris.

Asher, N. and Lascarides, A. (2003). Logics of Conversation. CUP

Asher, N. and Lascarides, A. (2008). Commitments, Beliefs and Intentions in Dialogue. *Proceedings of Londial 2008, the 12th Workshop on the Semantics and Pragmatics of Dialogue*, 35-43

Barendregt, H., Dekkers, W., Statman, R. (2013). Lambda Calculus with Types. ASL, Cambridge: CUP

Bengtson, J., Johansson, M., Parrow, J., and Björn, V. (2011). PSI-CALCULI: A FRAMEWORK FOR MOBILE PROCESSES WITH NOMINAL DATA AND LOGIC. *Logical Methods in Computer Science*. Vol. 7 (1:11), 2011, pp. 1-44

Cooper, R. (2015). Type theory and language. From perception to linguistic communication. Gothenburg University, Draft, June 11, 2015

Cresswell, M. J. (1973). Logics and Languages. Methuen & Co Ltd: London

Ginzburg, J. (2012). The interactive stance. Meaning for conversation. Clarendon Press: Oxford.

Giorgolo, G. (2010). Space and Time in Our Hands. UIL-OTS, Universiteit Utrecht, 2010

Hahn, F., and Rieser, H. (2012). Non-compositional Gestures. Talk given at ESSLLI 2012.

Hahn, F., Lawler, I., and Rieser, H. (2015). Iconic gestures are context-dependent on speech: Evidence from corpus data and first steps towards a formal model. Accepted for GESPIN 2015.

Hennessy, M. 2002. A fully abstract denotational semantics for the π-calculus. Theoretical Computer Science 278, 53-89

Johansson, M. 2010. *Psi-calculi: a framework for mobile process calculi. Cook your own correct process calculus - just add data and logic.* Acta Universitatis Upsaliensis. *Uppsala*. *Dissertations from the Faculty of Science and Technology* 94. 184 pp. Uppsala.

Kendon, A. (2004). *Gesture: Visible Action as Utterance.* Cambridge: CUP. Berlin/Heidelberg: Springer.

Kempson, R., Cann, R., Gregoromichelaki, E., Chatzikyriakidis, St. (2016). Language as Mechanisms for Interaction. To appear with *Theoretical Linguistics*.

Lascarides, A. and Stone, M. (2006). Formal semantics of iconic gesture. In Schlangen D. and Fernández, R., editors, Proceedings of the 10thWorkshop on the Semantics and Pragmatics of Dialogue, Brandial’06, Potsdam. Universitätsverlag Potsdam, pp. 64–71.

Lascarides, A. and Stone, M. (2009). A formal semantic analysis of gesture. Journal of Semantics, 26(4), pp. 393–449.

Lücking, A. (2013). *Ikonische Gesten. Grundzüge einer linguistischen Theorie.* De Gruyter Mouton, Germany.

Lücking, A., Bergmann, K., Hahn, F., Kopp, S., and Rieser, H. (2012). Data-based Analysis of Speech and Gesture: The Bielefeld Speech and Gesture Alignment Corpus (SaGA) and its Applications. *Journal on Multimodal User Interfaces* 7(1-2), pp. 5-18.

Lücking A., Pfeiffer T., Rieser H. (2015). Pointing and reference reconsidered. *Journal of Pragmatics 77*, pp. 56-79.

McNeill, D. (1992). *Hand and Mind: What Gestures Reveal about Thought.* Chicago: Chicago University Press.

McNeill, D. (2005). *Gesture and Thought.* Chicago: Chicago University Press.

Milner, R. (1999). *Communicating and mobile systems: the π-calculus*. Cambridge: CUP.

Parrow, J. (2001). An introduction to the π-calculus. In Bergstra, J. A., Ponse, A., and Smolka, S. A., editors, *Handbook of Process Algebra*. Amsterdam: Elsevier, pp. 479–545.

Poesio, M. and Rieser, H. (2010). Completions, Coordination, and Alignment in Dialogue. *Dialogue and Discourse*, *1*, 1–89

Poesio, M. and Rieser, H. (2011). An Incremental Model of Anaphora and Reference Resolution Based on Resource Situations. *Dialogue and Discourse*, *2*(1), 235–277

Rieser, H. (2010). On factoring out a gesture typology from the bielefeld speech-and-gesture-alignment corpus (saga). In: Kopp, S., Wachsmuth, I. (eds.) Gesture in Embodied Communication and Human-Computer Interaction. Springer, Berlin/Heidelberg

Rieser, H. (2013). Speech-gesture interfaces. An overview. In *Abstracts of the 35th Annual Conference of the German Linguistic Society (DGfS)*, pp. 282–283.

Rieser, H. (2014). Gesture and Speech as Autonomous Communicating Processes. Talk at the Stuttgart Workshop on "Embodied meaning goes public". Stuttgart University, December 2014

Rieser, H. (2015). When Hands Talk to Mouth. Gesture and Speech as Autonomous Communicating Processes. *Proceedings of Semdial 2015*, Gothenburg

Röpke, I. (2011). Watching the growth point grow. In *Proceedings of the Second Conference on Gesture and Speech in Interaction* (GESPIN) 2011.

Röpke, I., Hahn, F., and Rieser, H. (2013). Interface Constructions for Gestures Accompanying Verb Phrases. In *Abstracts of the 35th Annual Conference of the German Linguistic Society* (DGfS), Potsdam, pp. 295–296.

Sangiorgi, D. and Walker, D. (2001). The π-calculus. A Theory of Mobile Processes. Cambridge: CUP.