Introducing Dialogue Games

Lecture 4

Paul Piwek

The Open University, UK

ESSLLI 2007 Dublin 13 – 17 August

Overview Today & Friday

Today: descriptive dialogue games

- Task-oriented dialogue game for two virtual robots (Power, 1979)
- Generic dialogue game (Ginzburg, to appear)

Friday: Dialogue game rules

- Comparisons
- Critique (Grice)
- Questions – The End

Descriptive approaches

Ultimate aim: throw light on/model *naturally occurring dialogue* (rather than prescribing how a dialogue ought to be conducted)

- Power’s perspective − Computational models for addressing questions in Theoretical Psychology: e.g., how do linguistic acts fit into goal-directed behaviour?

- Ginzburg’s perspective − Linguistics: how to characterize well-formed, that is coherent, dialogues

Power (1974, 1979)

- “A Computer model of conversation” and “The organisation of purposeful dialogues”
- Purposeful: “A speaker is not just putting a meaning into words, he [sic] is also trying thereby to do something to achieve a purpose. And a hearer has not understood a remark unless he has perceived its purpose as well as its literal meaning.”

Power (1974, 1979)

1. Devise a computational model of a speaker.
2. Explore the consequences and limitations of that model by having two copies of the model engage in conversations.
3. Why? “[computer models] give quick accurate feedback on the coherence, completeness, and detailed consequences of a theory. Theories of cognitive processes are necessarily complicated, and are likely to be too complicated for the unaided human intelligence to cope with”

The model/program

1. Data structures representing a world (with its own objects, laws, etc.) that the agents inhabit;
2. Robots, each with a *mind* and certain *capabilities*;
3. A "chairman" who arranges time sharing between the agents;
4. Functions that print out what is happening.

The program – World

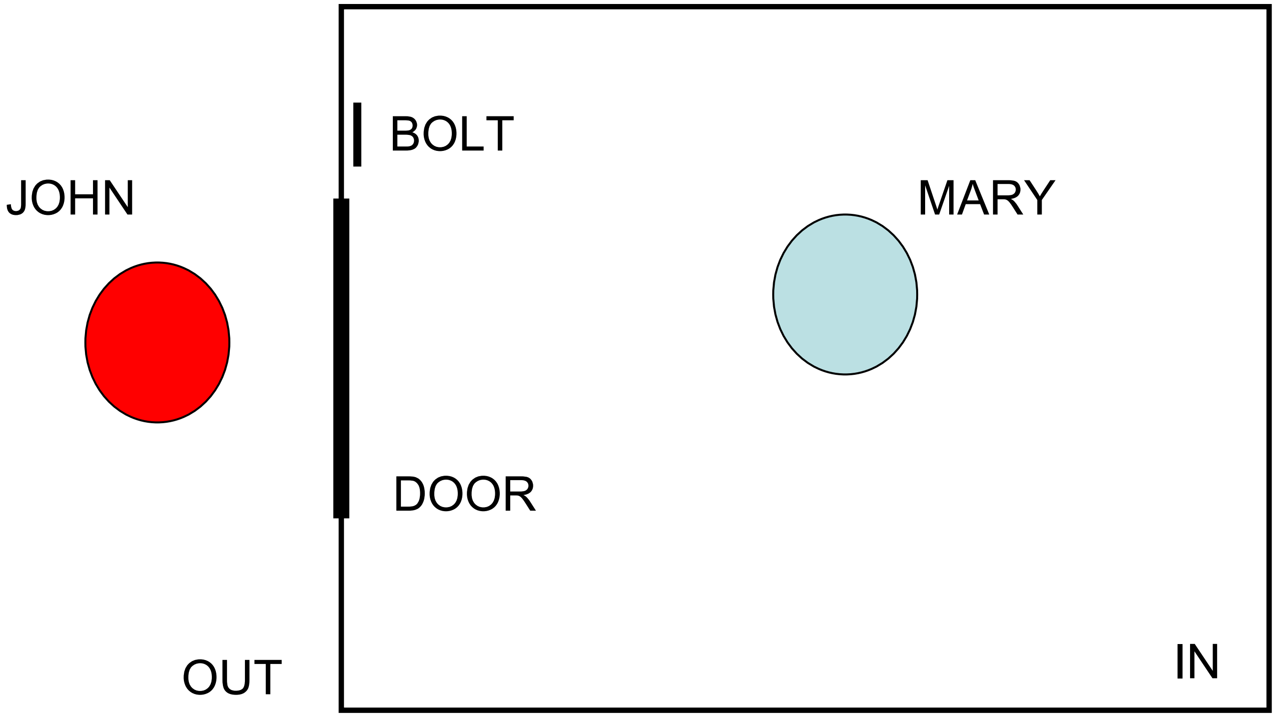
- <u>Objects</u> = John, Mary, a door, (a bolt)
- <u>State</u> = position of objects:
  - For robots: IN/OUT
  - For the door: OPEN/SHUT
  - For the bolt: UP/DOWN
- <u>Actions</u>
  - For robots: MOVE (i.e., try to move)
  - For the door: PUSH
  - For the bolt: SLIDE

The program – World

“Laws of nature”:

- If robot MOVE, then position robot changes provided that door OPEN.
- If robot PUSH door, then position door changes provided that bolt UP.
- If robot SLIDES bolt, then position bolt changes provided that robot is IN.

The program – World



Program – Robots

- Each robot's **mind** consists of:
  - Planning tree (populated with goals)
    - John's goal: JOHN IN
    - Mary's goal JOHN IN
  - Procedure execution stack
  - Beliefs (about regularities in the world)
    - John:
      - If a robot MOVES, nothing happens
      - If a robot PUSHES the door, the door changes position
      - If a robot SLIDES the bolt, nothing happens
    - Mary:
      - If a robot MOVES, it changes position provided the door is OPEN.
      - If a robot PUSHES the door, the door changes position.
      - If a robot SLIDES the bolt, nothing happens.

Program – Robots

- Each robot has **capabilities** for:
  - Perception:
    - John: SEE
    - Mary: -
  - Action
    - John: MOVE, SLIDE
    - Mary: MOVE, SLIDE, PUSH
  - Planning and Conversational Procedures

A conversation - 1

- John: Mary
- Mary: Yes
- John: I want to suggest a goal
- Mary: Go ahead.
- John: Will you help me get in?
- Mary: By all means.
- John: Shall we make a plan?
- Mary: John.
- John: Yes.

A conversation - 2

- Mary: May I ask you something?
- John: Go ahead.
- Mary: Are you in?
- John: No.
- Mary: Shall we make a plan?
- John: Okay.
- Mary: John.
- John: Yes.
- Mary: May I ask you something.
- John: Go ahead.

A conversation - 3

- Mary: Can you move?
- John: Yes.
- Mary: John.
- John: Yes.
- Mary: May I ask you something.
- John: Go ahead.

A conversation - 4

- Mary: Is the door open?
- John: No.
- Mary: I suggest that we get the door open and then you move.
- John: Mary
- Mary: Yes
- John: I want to explain something.
- Mary Go ahead.
- John: If one moves, nothing happens.

A conversation - 5

- Mary: I disagree. If one moves when the door is open one changes position.
- John: I see.
- Mary: I suggest that we get the door open and then you move.
- John: All right.
- Mary: Shall we make a plan?
- John: Okay.
- Mary: I suggest that I push the door.

A conversation - 6

- John: All right.

  Mary pushes the door and the state of the world changes

- Mary: I want to tell you something
- John: Go ahead.
- Mary: I have pushed the door.
- John: I see
- Mary: Let’s asses the results of my action

A conversation - 7

- John: Okay.
- Mary: Nothing has happened.
- John: Mary.
- Mary: Yes.
- John: I want to tell you something.
- Mary: Go ahead.
- John: The door is open.
- Mary: I see. The door has changed position.

A conversation - 8

- John: Yes.
- Mary: The door is now open.
- John: Right.

John moves and the state of the world changes

- John: I want to tell you something.
- Mary: Go ahead

A conversation - 9

- John: I have moved.
- Mary: I see.
- John: Let’s assess the result of my action.
- Mary: Okay.
- John: I have changed position.
- Mary: John.
- John: Yes.
- Mary: I want to tell you something.

A conversation – 10

- John: Go ahead.
- Mary: You are out.
- John: I disagree. I have changed position.
- Mary: Yes.
- John: I am in now.
- Mary: right.

➤ ➤ ➤

Planning and Action

Planning and Action

John: Mary
Mary: Yes
John: I want to suggest a goal
Mary: Go ahead.
John: Will you help me get in?
Mary: By all means.

Planning and Action

$[\text{JOHN IN}]_{\text{both}}$

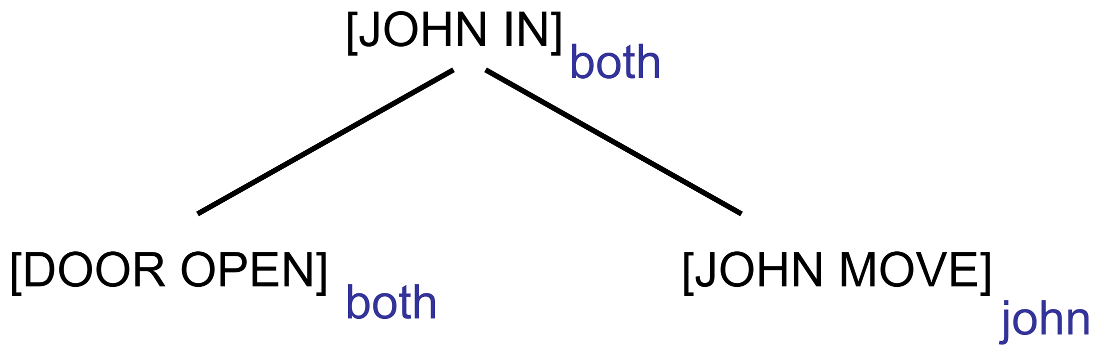
John: Mary
Mary: Yes
John: I want to suggest a goal
Mary: Go ahead.
John: Will you help me get in?
Mary: By all means.

Planning and Action

$[\text{JOHN IN}]_{\text{both}}$

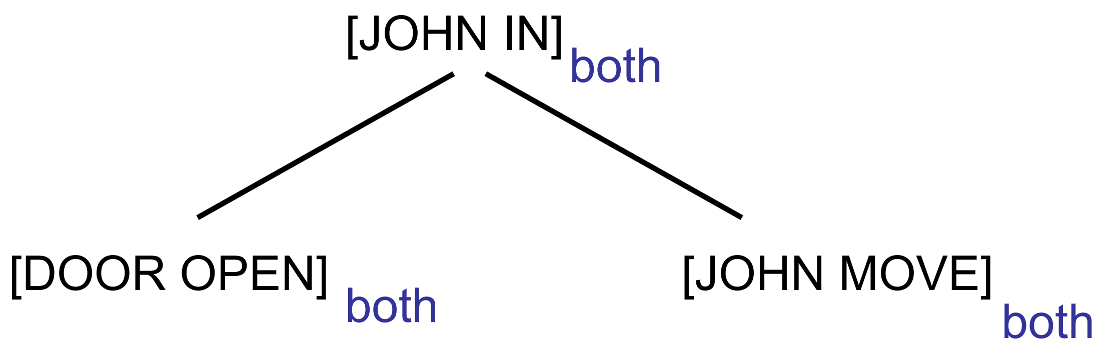
Mary: I suggest that we get the door open and then you move.
John: All right.

Planning and Action



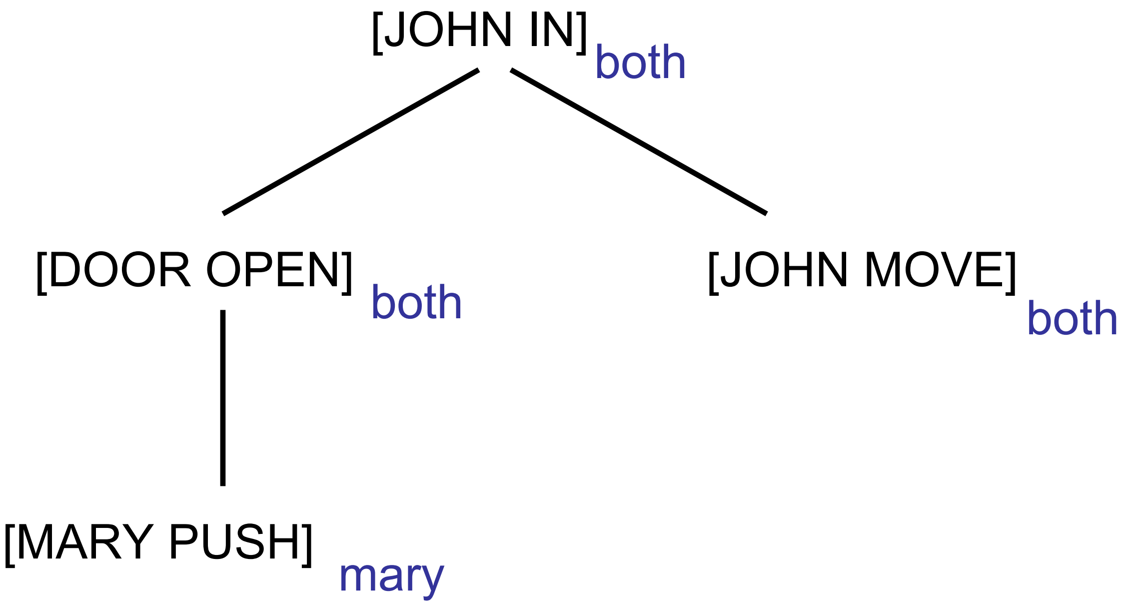
Mary: I suggest that we get the door open and then you move.
John: All right.

Planning and Action



Mary: I suggest that I push the door
John: All right

Planning and Action



Mary: I suggest that I push the door
John: All right

Procedures

- **Action and Observation**
  - SEE
  - PUSH

- **Planning**
  – CHOOSEGOAL
  – ACHIEVEGOAL
  – FINDPLAN
  – …

- **Conversational**
  – ANNOUNCE
  – AGREEGOAL
  – AGREEPLAN
  – ASK
  – TELL
  – DISCUSS
  – ASSESS

Single Agent Planning Procedures

- ACHIEVEGOAL
- FINDPLAN

ACHIEVEGOAL M $\Rightarrow$ R(esult)

Start with G:=M

0. Identify current goal G
1. If action G, check whether it can be performed directly. If so, perform it, remove it, and go to 0.
2. If state G, test whether it holds.
   - If so, remove it and
     - If main goal, exit and report *success*; else return to 0.
   - Else, FINDPLAN P to achieve G. If there is no plan, remove G and its sister(s) and record failure, and go to 0. If G is main, return *fail*.
4. Attach P to G and return to 0.

FINDPLAN G $\Rightarrow$ P(lan)

0. Identify the type of object, call it T, in G (e.g, for JOHN IN, T:=ROBOT)
1. Find a belief B (If action X, then change Y provided that state Z) in which T is the Y-entry. (E.g., If a ROBOT MOVES, ROBOT changes position provided DOOR is OPEN). Report failure if no belief could be found.
2. Replace ROBOT in B with the name of the agent.
3. Check whether you can perform X. If not, report failure.
4. Test whether the "provided that" clause Z is satisfied.
5. If Z is already satisfied, P := X, else, P:= Z, X

Joint planning procedures

Revised ACHIEVEGOAL

1.
   a) If action G is assigned to your partner, wait until he tells you he performed it.
   b) If action G is assigned to you, perform it. Then tell your partner that you performed it and initiate a conversation to assess the result (assuming the supergoal of G is joint).

   Finally, remove G and go to 0.

3. If G is your responsibility, select a plan to achieve it. If G is joint, have a conversation to agree on the plan (call AGREEPLAN which will in turn call FINDPLAN).

Revised FINDPLAN

- If the goal is joint, find the robot who can perform the plan to achieve the goal. (For calls from AGREEPLAN)

Calling Conversational Procedures

- CHOOSEGOAL

0. Select a main goal and call it M
1. Try to achieve main goal M on your own (individual ACHIEVEGOAL).
2. If result of ACHIEVEGOAL is succeed, go to 4.
3. Secure cooperation through AGREEGOAL. If answer A “yes”, label M as a joint responsibility and call joint ACHIEVEGOAL. If the answer A is “no”, go to 4.
4. do nothing (control back to the chairman)

Conversational Procedure AGREEGOAL (S1,S2,G)$\Rightarrow$A

- S1 composes sentence S asking for help with G
- S2 decodes S, obtaining a value for G. If it is identical with his own main goal, he gives A the value “yes”, if not, “no”. If A is “yes”, the variable A of CHOOSEGOAL is also set to “yes”. S2 utters A and exits from AGREEGOAL
- S1 reads A and exits returning A to the procedure which called AGREEGOAL.

John decides on a goal (JOHN IN) and tries to find a way to achieve it.

CHOOSEGOAL	R = ? A = ?

JOHN

MARY

John fails to achieve the goal on his own.

ACHIEVEGOAL R = FAIL

CHOOSEGOAL R = ?
A = ?

JOHN MARY

Now he has to arrange a call of AGREEGOAL to make it a joint goal.

CHOOSEGOAL	R = FAIL A = ?	
JOHN		MARY

John produces a call by name: “**Mary**”

CHOOSEGOAL

R = FAIL
A = ?

JOHN

MARY

Chairman gives control to Mary

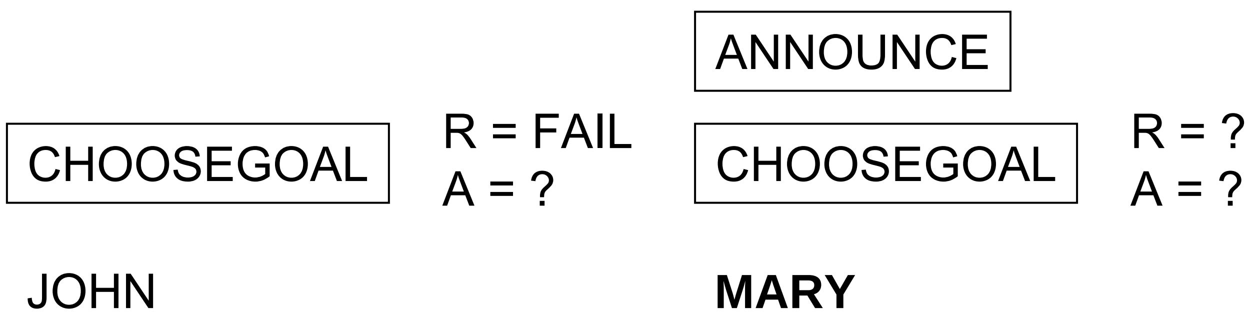
| CHOOSEGOAL |

R = FAIL
A = ?

JOHN

MARY

Mary calls CHOOSEGOAL (on awakening), but without executing it, and then responds to the call by name (“**Yes**”) and starts ANNOUNCE in the role of S2.



ANNOUNCE (S1,S2,C)

1. S1 composes a sentence U to announce C (another conv. procedure), and utters it
2. S2 decodes U obtaining C. He then checks that he is ready to call C. If not, he suspends ANNOUNCE, until he is ready. Then he calls C, in place of ANNOUNCE, taking the role of S2 in C, and utters “go ahead” or “okay”.
3. S1 reads S2’s response, checks that S2 is ready, and then calls C in place of ANNOUNCE, taking the role of S1 in C.

Control is back with John, who now also starts announce and produces the first part, e.g., "**I want to suggest a goal**".

JOHN		MARY	
ANNOUNCE		ANNOUNCE	
CHOOSEGOAL	R = FAIL A = ?	CHOOSEGOAL	R = ? A = ?

JOHN MARY

Control is back with Mary; from “**I want to suggest a goal**” she understands that C = AGREEGOAL.

ANNOUNCE

CHOOSEGOAL

R = FAIL
A = ?

JOHN

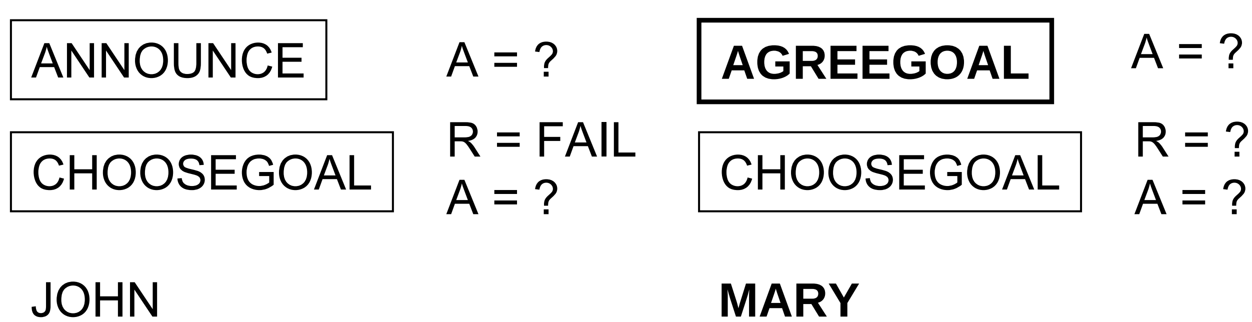
ANNOUNCE

CHOOSEGOAL

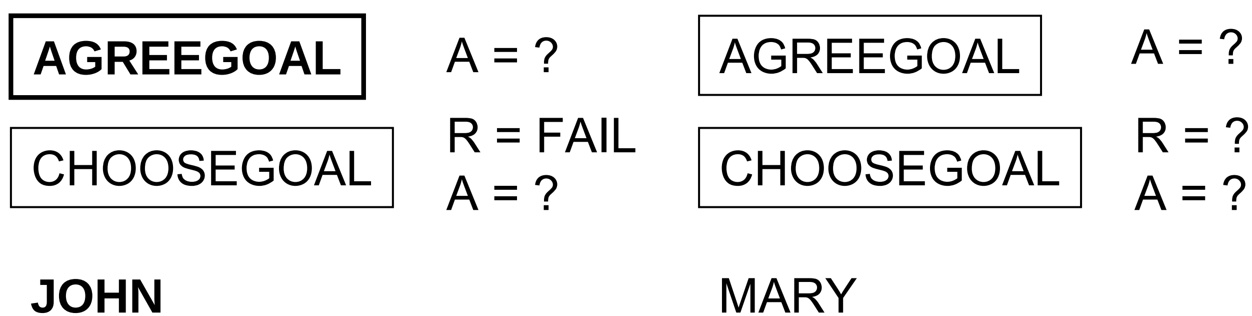
R = ?
A = ?

MARY

She starts the procedure AGREEGOAL, replacing ANNOUNCE and provides feedback, e.g., “**go ahead**”.



John reads Mary’s response, then calls AGREEGOAL with himself as the first speaker.



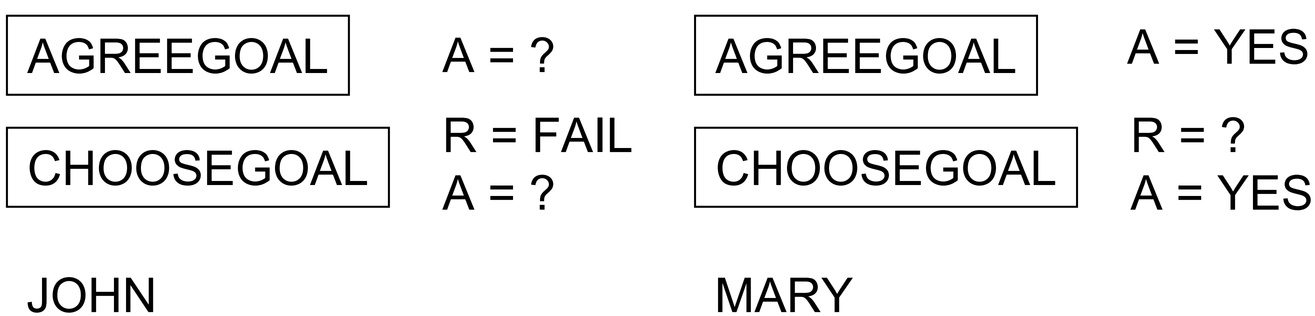
John produces the first part: “**Will you help me get in?**”

AGREEGOAL	A = ?	AGREEGOAL	A = ?
CHOOSEGOAL	R = FAIL A = ?	CHOOSEGOAL	R = ? A = ?
JOHN		MARY	

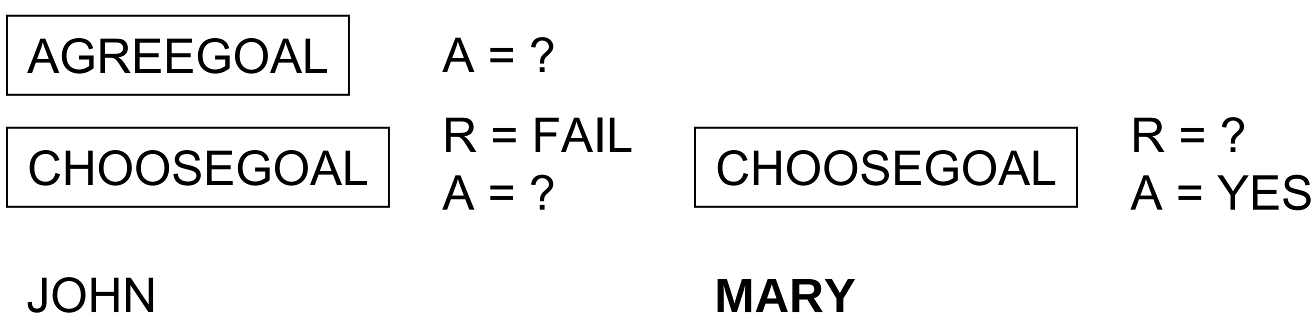
Mary decodes “Will you help me get in?”. She compares the goal with her own goal. Since they are identical she set A to “YES” and responds with “**By all means**”.

AGREEGOAL	A = ?	AGREEGOAL	A = ?
CHOOSEGOAL	R = FAIL A = ?	CHOOSEGOAL	R = ? A = ?
JOHN		**MARY**	

Mary decodes “Will you help me get in?”. She compares the goal with her own goal. Since they are identical she set A to “YES” and responds with “**By all means**”.



A of CHOOSEGOAL has been updated, and the main goal is now labelled as a joint responsibility. She then exit AGREEGOAL.



John reads A, and returns it to the procedure that called AGREEGOAL and then exits AGREEGOAL.

AGREEGOAL A = ?

CHOOSEGOAL R = FAIL A = ?

CHOOSEGOAL R = ? A = YES

JOHN

MARY

John reads A, and returns it to the procedure that called AGREEGOAL and then exits AGREEGOAL.

AGREEGOAL	A = YES		
CHOOSEGOAL	R = FAIL A = ?	CHOOSEGOAL	R = ? A = YES
JOHN		MARY	

John reads A, and returns it to the procedure that called AGREEGOAL and then exits AGREEGOAL.

AGREEGOAL	A = YES		
CHOOSEGOAL	R = FAIL A = YES	CHOOSEGOAL	R = ? A = YES
JOHN		MARY	

John now also labels the goal “JOHN IN” as joint responsibility.

CHOOSEGOAL	R = FAIL A = YES	CHOOSEGOAL	R = ? A = YES
JOHN		MARY	

Since the initiative is still with John, he calls achieve goal again.

ACHIEVEGOAL

CHOOSEGOAL

R = FAIL
A = YES

CHOOSEGOAL

R = ?
A = YES

JOHN

MARY

From ACHIEGOAL: If G is a joint responsibility, initiate a conversation to agree a plan P to achieve it. So now we get again announce part 1 “Shall we make a plan?” **…**

ACHIEVEGOAL

CHOOSEGOAL

R = FAIL
A = YES

CHOOSEGOAL

R = ?
A = YES

JOHN

MARY

Conversational Procedure ASK(S1,S2,Q)

- S1 composes S which expresses question Q and utters it.
- S2 reads U and obtains a value for Q. He records that S1 cannot see the object in Q, and then inspects the world to see if Q is true. If he finds no information, he says “I don’t know”, otherwise “yes” or “no” as appropriate.
- S1 reads S2’s reply and updates his world model appropriately. If it is “I don’t know, he records that S2 cannot see the object mentioned in Q”.

In a nutshell

- A program for simulating dialogues between agents;
- Based on traditional AI planning notions (goal, belief, plan, …)
- Tight integration of conversational and other planning procedures, e.g.:
  - ACHIEVEGOAL calls AGREEGOAL
  - ASK calls SEE

In a nutshell

- Novel (in 1979): the notion of a “dialogue state”: control stack + planning tree. (Note that Power does not mention beliefs, although these can change in the course of a conversation).
- The point of an utterance is modelled in terms of the control stack. It traces the motivation for/rationale behind an utterance.

Limitations

- **“Fault 1”**: The model does not capture how utterances achieve a goal. It does explain why we don’t say “Will you put the fire on?” if, for example:
  - John has just asked Mary to put the fire on and she agreed.
  - Bill has just asked Mary to put the fire on and she agreed.
  - Mary has just announced that she will put the fire on.
  - Mary is walking towards the fire with a box of matches.

Limitations

- Mary has just said that she doesn’t want the fire on.
- Mary, who is a child, is not allowed to put the fire on. John disagrees with this rule but is sure that Mary will obey it.

Limitations

- A request for X to do A causes X to do A first by causing X to INTEND to do A.
- Principles of rational action
  - Don’t try to achieve goals that are already achieved
  - Don’t try to achieve goals that are known to be unachievable.
- These issues are dealt with to some extent in part II of Power’s paper.

Limitations

- **“Fault 2”**: relations between elements of the dialogue state are not represented explicitly.
- For instance, there is no explicit representation of the notion of a candidate plan: P is a candidate plan for G. These are simply variables of different procedures. This makes it difficult to respond to unexpected turns (the problem of flexibility).

Further issues

- The shape of a dialogue depends on the planning strategy: left-branching depth-first. What if agents have different strategies?
- We need to assume that at the beginning of a dialogue both interlocutors are running CHOOSEGOAL. What if someone barges in in the middle of an activity?
- No parallelism is allowed (by the chairman).
- “Cognitive” actions (SEE, ASK, …) are strictly separated ontologically from physical actions (MOVE, …) that change the world.

Ginzburg (forthcoming)

- Chapter 4 (“Basic Interaction in Dialogue”) “of Semantics and Interaction in Dialogue”
- “the dialogue analyst describes conventionally acceptable moves and the effects they give rise to among conversation participants in terms of information states”

Ginzburg (forthcoming)

- Methodology similar to that of the traditional linguist (e.g., syntactician):
  - Syntax: The theory assigns structure to sentences and explains ill-formedness of certain sequences (of words) in terms of the impossibility to assign a valid structure to them.
  - Dialogue: The theory assign sequences of information states to certain sequences of moves (effects) and rules out others as incoherent, because there does not exist a corresponding sequence of information states.

Ginzburg (forthcoming)

- Methodology in practice:
  - Compare and contrast sequences of moves that are intuitively coherent and those that are not, and show that the theory only models the coherent ones.

Info. States and Conv. Rules

- A notion of an **information state**. Let S be the set of all possible information states.
- A notion of a **conversational rule** *r* that maps $X \subseteq S \Rightarrow Y \subseteq S$. (Preconditions to Effects)
- The ordered pair of information states <x,y> is a pairwise-r-coherent, iff $x \in X$ and $y \in Y$.
- IS-R-coherent($<x_1,\ldots,x_n>$) iff for each pair $<x_i,x_{i+1}>$ (with $0<i<n+1$) $\exists r \in Y$ such that the pair is pairwise-r-coherent.

Warning: notation and formulation of principles is different (though hopefully equivalent in spirit) to that of Ginzburg Chapter 4

Info. States and Conv. Rules

- Rule composition: If r: $A \subseteq S \Rightarrow B \subseteq S$ and r': $C \subseteq S \Rightarrow D \subseteq S$, then $r \circ r' = A \subseteq S \Rightarrow D \subseteq S$, provided that: $B \subseteq C$.

- *But where are the dialogue moves?*

Moves

- For each information state $s \in S$:
  $\exists M$: s.Moves = M.
- If M is non-empty, we also have an m such that s.LatestMove = m
- Non-empty sequence M is R-coherent iff $\exists s_1,\ldots,s_n$ $(n>1)$ such that s_1.Moves = empty & s_n.Moves = M & IS-R-coherent($<s_1,\ldots,s_n>$).

Moves

- **Pairwise Coherence** (Ginzburg formulation): m_1 and m_2 are R-pairwise-coherent given a set of rules R *iff* there exist S_0, r_1 (S_0,S_1), S_1.latestMove=m_1, there are S_i, r_i $(1 \leq i \leq k)$ such that: r_{i+1} (S_i,S_{i+1}) and S_k.LatestMove = m_2.
- Compare

Moves

- **Pairwise Coherence** (*Ginzburg formulation*): m_1 and m_2 are R-pairwise-coherent given a set of rules R *iff* there exist S_0, r_1 (S_0,S_1), S_1.latestMove=m_1, there are S_i, r_i $(1 \leq i \leq k)$ such that: r_{i+1} (S_i,S_{i+1}) and S_k.LatestMove = m_2.
- Compare Intermediate moves between m_1 and m_2 , if for any S_i LatestMove is not equal to m_1 or m_2.

Notational Convention

- We write

  $\{Attr_1 = In\text{-}value_1, \ldots, Attr_n = value_n\} \Rightarrow$
  $\{Attr'_1 = value'_1, \ldots, Attr'_n = value'_n\}$

  for

  $X \subseteq S \Rightarrow Y \subseteq S$ such that X consists of exactly those $x \in S$ for which $x.Attr_1 = In\text{-}value_1, \ldots, x.Attr_n = value_n$ and Y consists of exactly those $y \in S$ for which $y.Attr'_1 = value'_1, \ldots, y.Attr'_n = value'_n$

This is a simplified notation replacing the record types in Ginzburg (forthcoming)

Two kinds of rules

- <u>Update rules:</u>

  $A \Rightarrow A'$ if

  (Moves = M) $\in$ A and (Moves = M) $\in$ A'

- <u>Generation/reaction rules:</u>

  $A \Rightarrow A'$ if

  (Moves = M) $\in$ A and (Moves = M+m) $\in$ A'

Note that this distinction is not made in Ginzburg Chapter 4 (but see Piwek 1998)

Information states

- *Dialogue Gameboards (DGB)*
- Facts: set of commonly agreed upon facts (Ginzburg argues that each interlocutor maintains their own repr. of FACTS). Closed under conjunction/disjunction.
- QUD (‘questions under discussion’): partially ordered set that specifies the currently discussed questions
- MAX-QUD: discourse topic
- Moves: content of the moves made
- LatestMove

Assumption: MOVES contains propositions characterizing the linguistic sign (illocution?) of utterances

Rules

- Greeting
- Parting
- Disengaged
- Free Speech
- QSpec
- Ask QUD incrementation
- Assert QUD incrementation
- Assertion checking
- Accept move
- Confirm move
- Fact update/QUD downdate
- Question Introduction Appropriateness Condition (QIAC)
- QCoord
- Initiating Move (relative to Private)

Overview Today & Friday

Today: descriptive dialogue games

- Task-oriented dialogue game for two virtual robots (Power, 1979)
- Generic dialogue game (Ginzburg, to appear)

Tomorrow: Dialogue game rules

- Ginzburg continued
- Comparisons
- Critique (Grice)
- Questions – The End