MULTIMODAL SEMANTIC SIMULATIONS OF LINGUISTICALLY UNDERSPECIFIED MOTION EVENTS

Nikhil Krishnaswamy and James Pustejovsky, Brandeis University

August 5, 2016, Spatial Cognition 2016, Philadelphia, PA, USA

Foundations

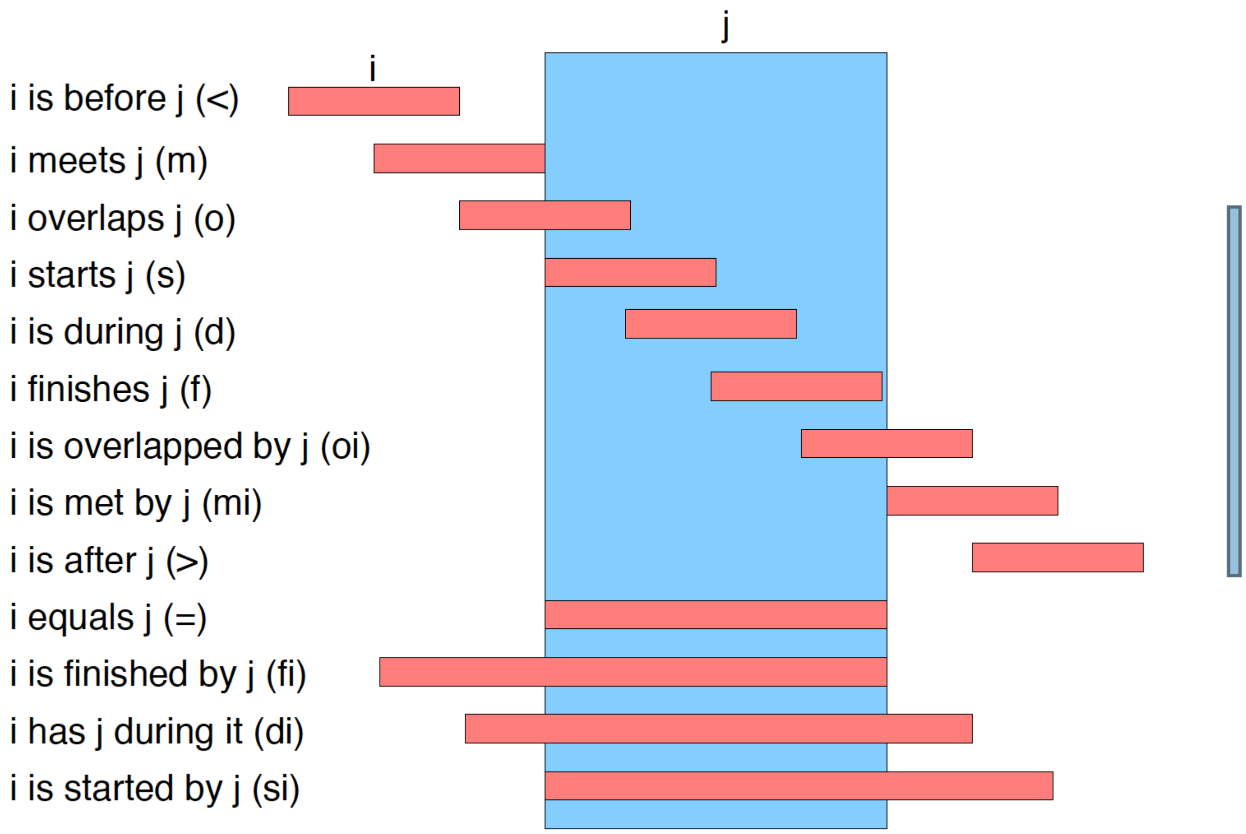
- Remarkable number of concepts in human mental model
- Mental models are adaptable
  - Can make sense of new situations, contexts, and kinds of knowledge
  - Can be revised based on new experience
- Mental models are embodied and multimodal
  - Embodiment maps concepts between domains
  - Modalities (perceptual and effector) constitute aspects of representation
- "Simulation": mental instantiation of an utterance, based on embodiment

Nikhil Krishnaswamy | nkrishna@brandeis.edu

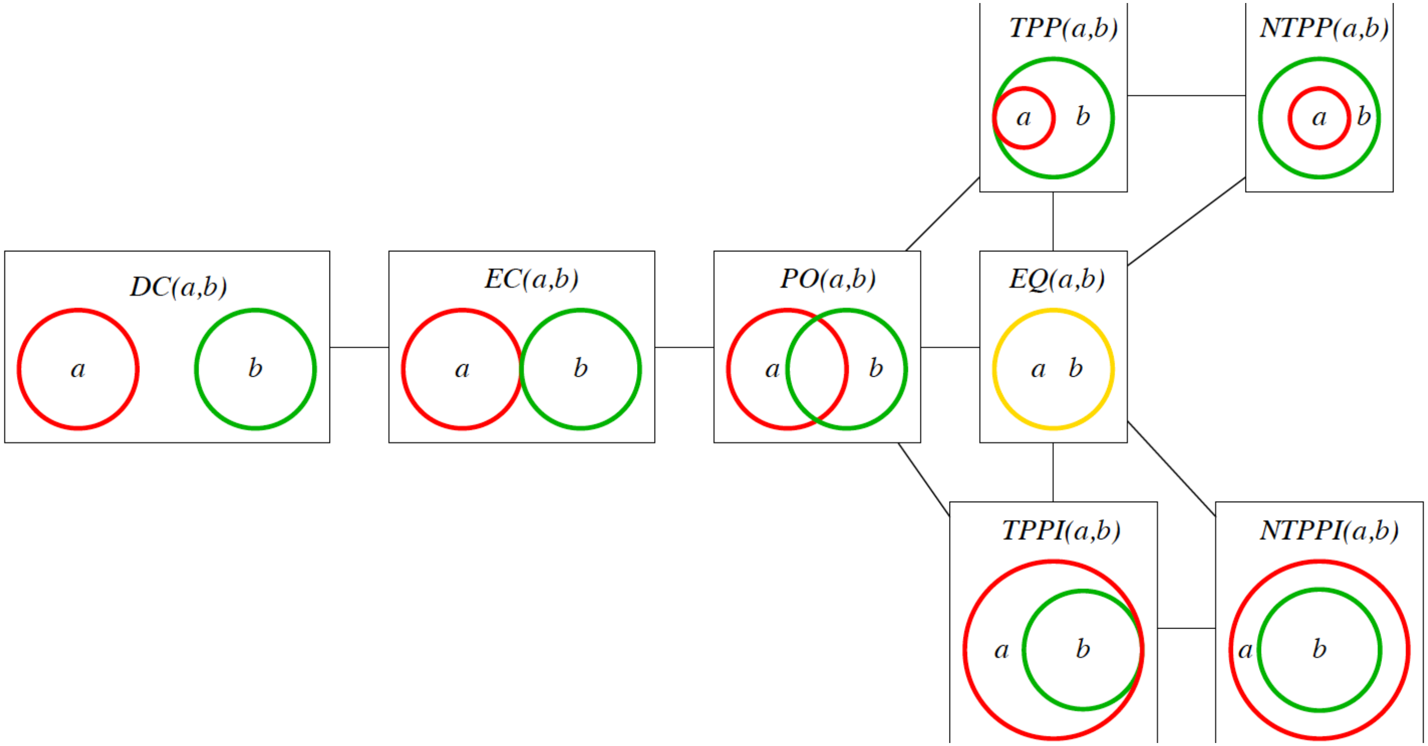
Past/Related Research

- Spatial/temporal algebraic interval logic
  - Allen Temporal Relations (Allen, 1983)
  - Region Connection Calculus (RCC8) (Randell et al., 1992)
    - RCC-3D (Albath, et al., 2010)
- Generative Lexicon, DITL (Pustejovsky, 1995; Pustejovsky and Moszkowicz, 2011)
- Static scene generation
  - WordsEye (Coyne and Sproat, 2001)
  - LEONARD (Siskind, 2001)
  - Stanford NLP Group (Chang et al., 2015)
- QSR/Game AI approaches to scenario-based simulation (Forbus et al., 2001; Dill, 2011)
- Spatial constraint mapping to animation (Bindiganavale and Badler, 1998)

Allen Temporal Relations



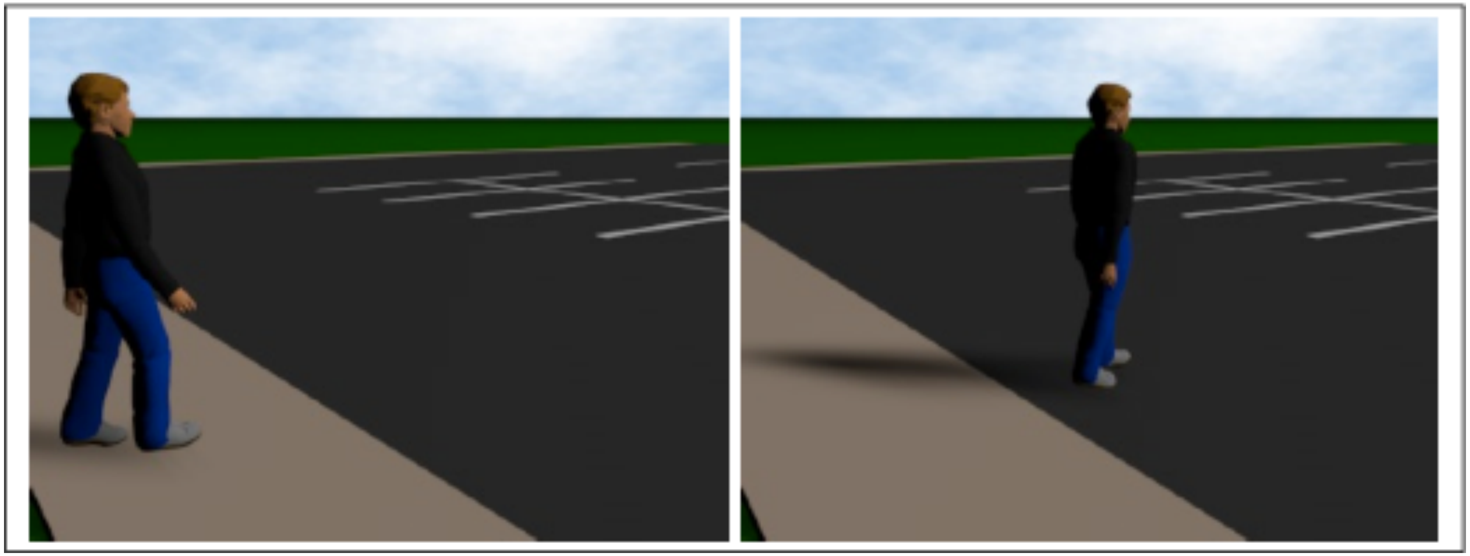
Region Connection Calculus



WordsEye

Generated from text: It is morning. It is partly cloudy. the ground is shiny. the grey shiny hill is is on the ground. The 10 foot tall grey cat is 8 feet in the hill. the tiny plane is 7 feet above the hill...

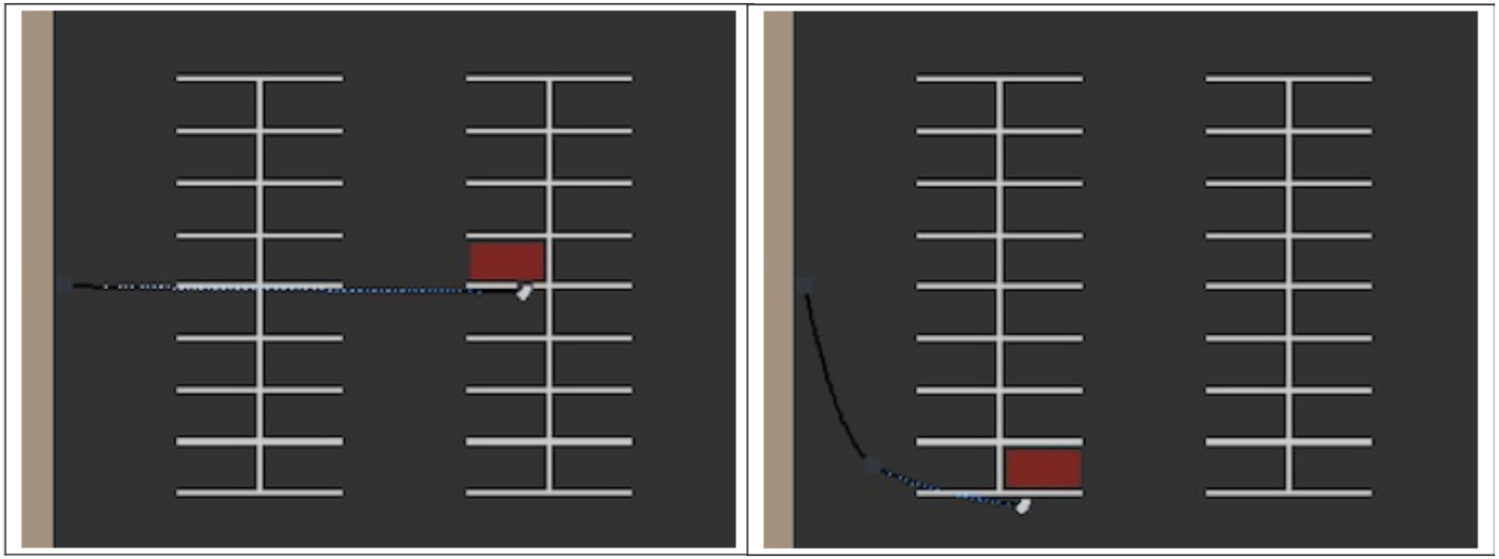
Nikhil Krishnaswamy | nkrishna@brandeis.edu



"Enter$_P$ the parking lot"

Path depends on bounds of parking lot

"Enter" is a **path** verb
(Pustejovsky and Moszkowicz, 2011)



"Hurry$_m$ to the car"

Path depends on location of car

"Hurry" is a **manner of motion** verb
(Pustejovsky and Moszkowicz, 2011)

 Nikhil Krishnaswamy | nkrishna@brandeis.edu

Events as Programs

- **Path** verbs **designate** a distinguished value in the state-to-state location change
  - Change in value is **tested**
- **Manner** of motion verbs **iterate** a state-to-state location change
  - Change in value is **assigned**/reassigned
- Verbs can be realized as **programs** enacted over arguments (Naumann, 1999)

Events as Programs

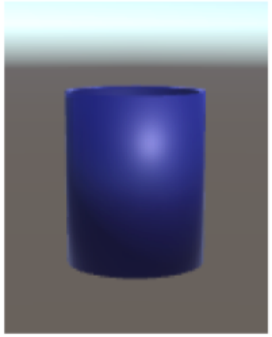
- Programs are compositional
  - Program's linguistic representation can be broken down into subevents
- Simple programs
  - translocate, rotate, grasp, hold, release, etc.
- Complex programs
  - put, stack, etc.

$put(A, B)$
a. Given $\mathcal{C}$ being satisfied (A is clear, within reach, etc), then *grasp* A, and **while** *hold* A, *move* A until at position B.
b. $\mathcal{C}?; grasp(A); (hold(A)?; move(A))^*; on(A, B)?; ungrasp(A); \neg hold(A)?$

- VoxML: Visual Object Concept Modeling Language (Pustejovsky and Krishnaswamy, 2016)
- Annotation and modeling language for "voxemes"
  - Visual instantiation of a lexeme
- Scaffold for mapping from lexical information to simulated objects and operationalized behaviors
- Encodes afforded behaviors for each object
  - Gibsonian - afforded by object structure (e.g. grasp, move, lift) (Gibson, 1977; 1979)
  - Telic - goal-directed, purposeful (e.g. drink from) (Pustejovsky, 1995)

VoxML

$$
\left[\begin{array}{l}
\textbf{cup} \\
\text{LEX} = \left[\begin{array}{l} \text{PRED} = \textbf{cup} \\ \text{TYPE} = \textbf{physobj} \end{array}\right] \\
\text{TYPE} = \left[\begin{array}{l} \text{HEAD} = \textbf{cylindroid[1]} \\ \text{COMPONENTS} = \textbf{surface,interior} \\ \text{CONCAVITY} = \textbf{concave} \\ \text{ROTATSYM} = \{Y\} \\ \text{REFLECTSYM} = \{XY, YZ\} \end{array}\right] \\
\text{HABITAT} = \left[\begin{array}{l} \text{INTR} = [2]\left[\begin{array}{l} \text{UP} = align(Y, \mathcal{E}_Y) \\ \text{TOP} = top(+Y) \end{array}\right] \\ \text{EXTR} = \ldots \end{array}\right] \\
\text{AFFORD\_STR} = \left[\begin{array}{l} A_1 = H[2] \rightarrow [put(x, on([1]))]support([1], x) \\ A_2 = H[2] \rightarrow [put(x, in([1]))]contain([1], x) \\ A_3 = H[2] \rightarrow [grasp(x, [1])] \end{array}\right] \\
\text{EMBODIMENT} = \left[\begin{array}{l} \text{SCALE} = \textbf{<agent} \\ \text{MOVABLE} = \textbf{true} \end{array}\right]
\end{array}\right]
$$



VoxML

$$\begin{bmatrix} \textbf{put} \\ \text{LEX} = \begin{bmatrix} \text{PRED} = \textbf{put} \\ \text{TYPE} = \textbf{transition\_event} \end{bmatrix} \\ \text{TYPE} = \begin{bmatrix} \text{HEAD} = \textbf{transition} \\ \text{ARGS} = \begin{bmatrix} \text{A}_1 = \textbf{x:agent} \\ \text{A}_2 = \textbf{y:physobj} \\ \text{A}_3 = \textbf{z:location} \end{bmatrix} \\ \text{BODY} = \begin{bmatrix} \text{E}_1 = grasp(x,y) \\ \text{E}_2 = [while(hold(x,y), move(y))] \\ \text{E}_3 = [at(y,z) \rightarrow ungrasp(x,y)] \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \textbf{in} \\ \text{LEX} = \begin{bmatrix} \text{PRED} = \textbf{in} \end{bmatrix} \\ \text{TYPE} = \begin{bmatrix} \text{CLASS} = \textbf{config} \\ \text{VALUE} = \textbf{ProperPart} \mid\mid \textbf{PO} \\ \text{ARGS} = \begin{bmatrix} \text{A}_1 = \textbf{x:3D} \\ \text{A}_2 = \textbf{y:3D} \end{bmatrix} \\ \text{CONSTR} = \textbf{...} \end{bmatrix} \end{bmatrix}$$

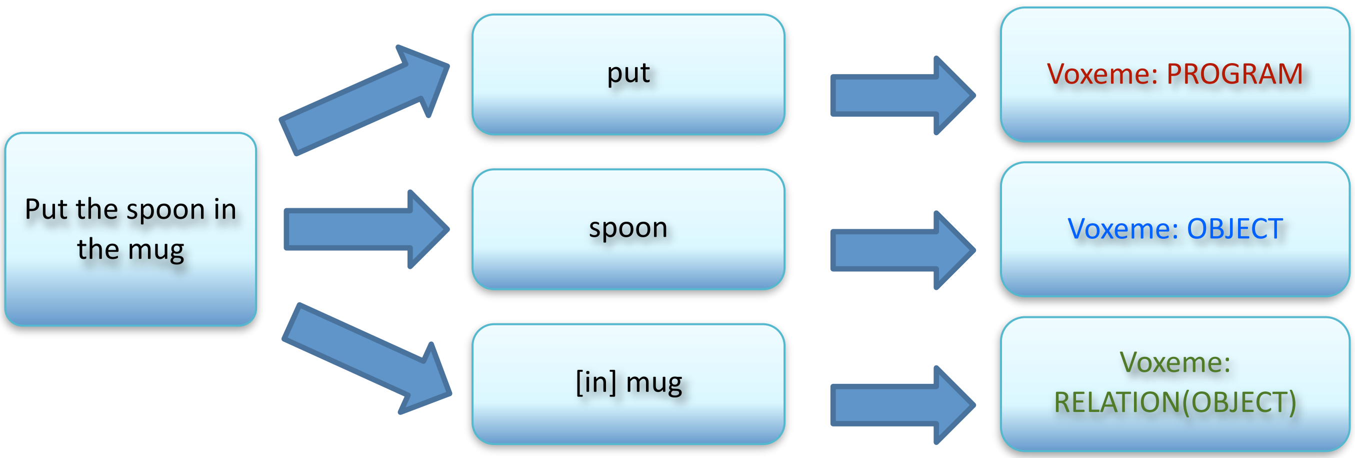
VoxSim: Software Architecture

We begin by inputting a sentence in plain English

Put the spoon in the mug

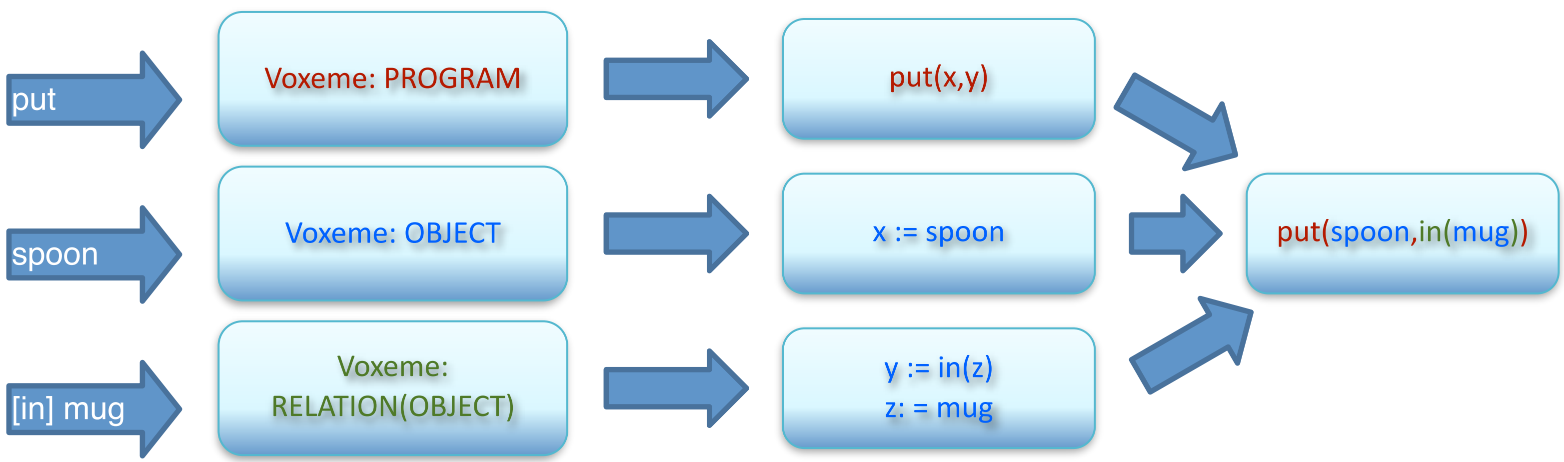
VoxSim: Software Architecture

From a dependency parse, we extract labeled entities in the scene, and verbs those entities may afford



VoxSim: Software Architecture

Resolve the parsed sentence into a predicate-logic formula



Nikhil Krishnaswamy | nkrishna@brandeis.edu

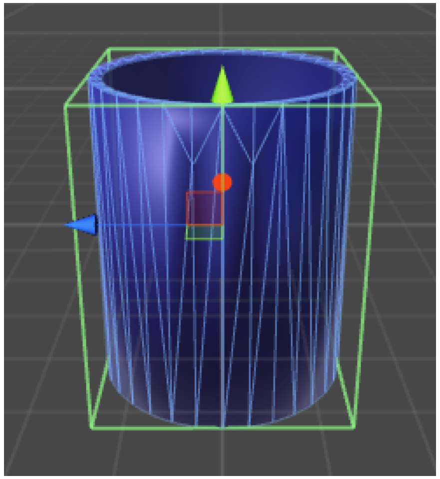
Each predicate is operationalized according to its type structure

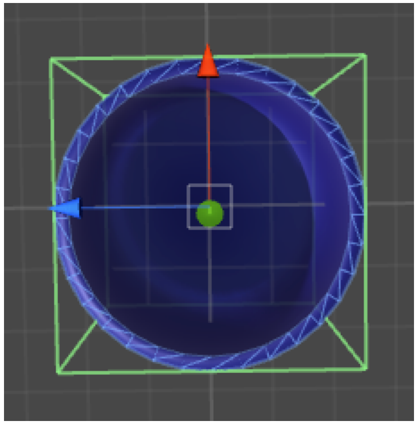
put(spoon,in(mug))

- in(z): takes object, outputs location
- put(x,y): path verb
  - while(!at(y), move(x))

Semantic Processing

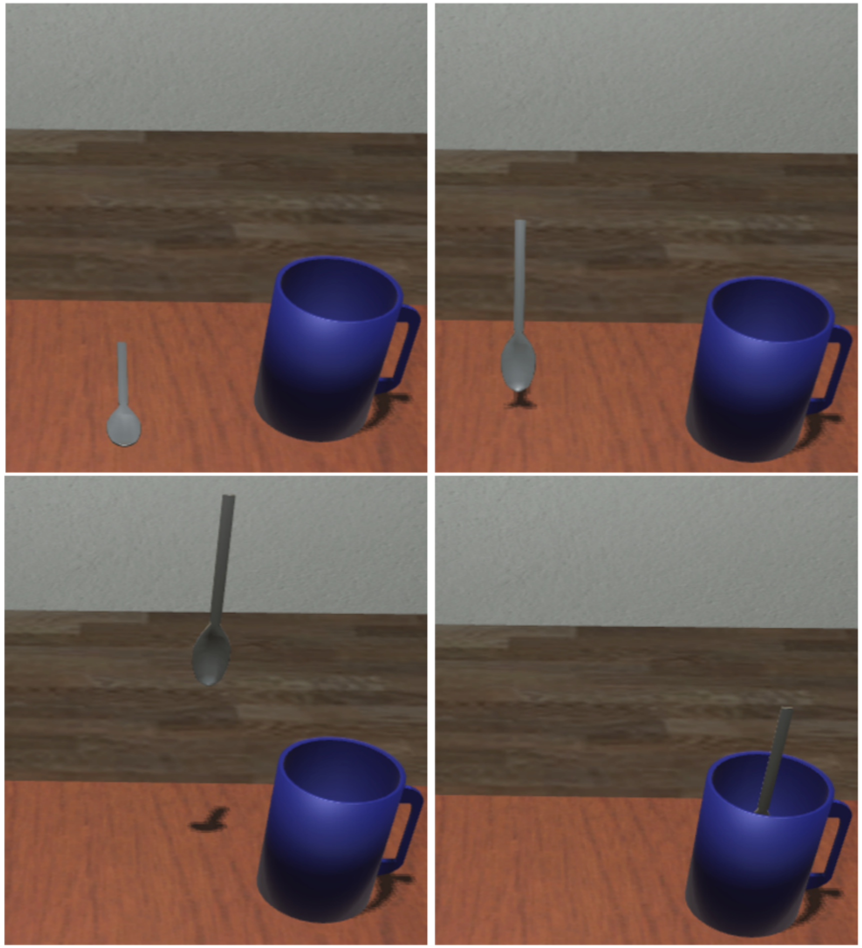
- Object bounds may not contour to geometry
  - e.g. Concave objects
- Semantic information imposes further constraints
- “in cup”: (PO | TPP | NTPP) with area denoted by cup’s interior
  - Interpenetrates bounds, but not geometry



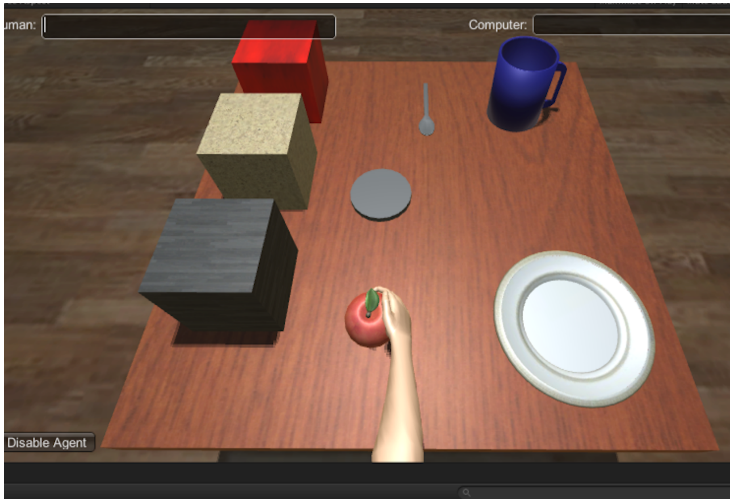


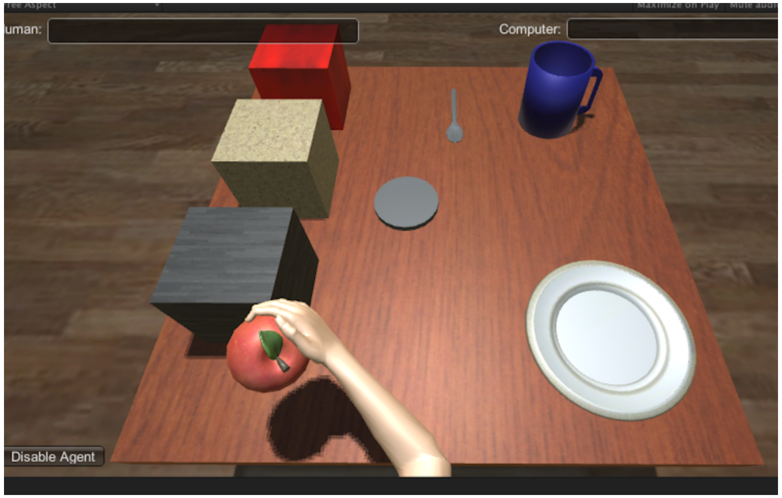
Semantic Processing

- Can test be satisfied with current object configuration?
- Can test be satisfied by reorienting objects?
- Can test be satisfied at all?



Rig Attachment





- Temporary parent-child relationship between joint on rig and manipulated object
- Allows agent and object to move together
- “Object model” + “Action model” = “Event model”

Demo

- Platform for incorporating motion/dynamic semantics into visualization
  - Visualization → Simulation → Minimal Model
- Runtime visualization generation necessitates assigning values in the simulation to parameters unspecified in minimal model
  - e.g. speed, direction, etc.
- Complete set of primitive programs in a particular domain unknown

 Nikhil Krishnaswamy | nkrishna@brandeis.edu

Future Work

- Monte-Carlo simulation generation with multiple evaluation tasks
  - Given visualization with randomly-assigned underspecified variables, choose best description
  - Given description, choose best visualization from randomly-generated set
  - Automatic evaluation of actual simulation result vs. DITL-derived satisfaction conditions
- Corpus building for linked videos and simulations with event labels for machine learning of event classification

Acknowledgments

SIFT Smart Information Flow Technologies

DARPA

Department of the Navy ONR Science & Technology

Brandeis University Student Workers
Jessica Huynh
Paul Kang
Subahu Rayamajhi
Amy Wu
Beverly Lum
Victoria Tran