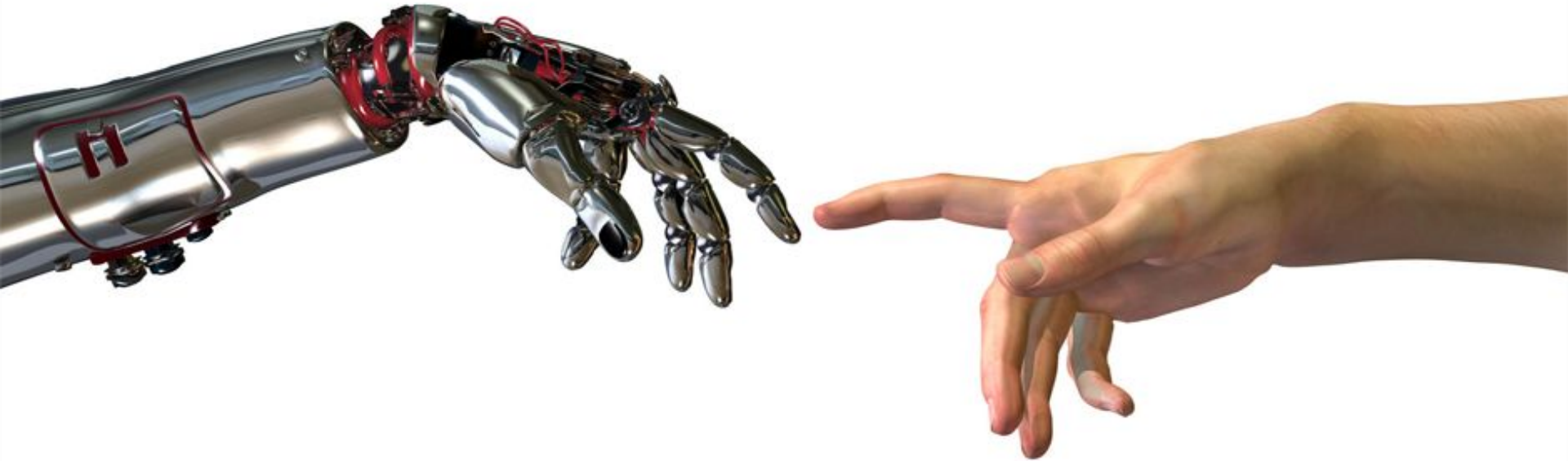
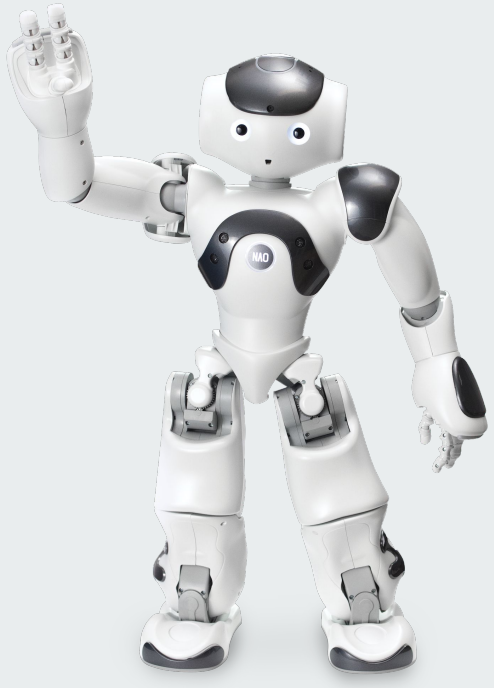
Optimizing HRI Interactions for NAO

Leonard Ramsey and Zoya Yeprem



Introduction



Motivation

- Robots and Humans
- Robot companions are starting to be explored more
  - (e.g. Friends, Caretakers, Teachers)
  - Robots can consistently provide source of happiness and compassion that humans may not always be capable of
- Not many applications of formal methods for robots operating in non-safety critical applications

Problem Statement

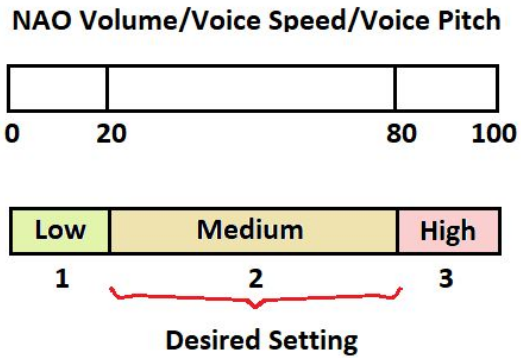
Robots should be capable of optimally engaging with human users

iPal companion

NAO teacher

Assumptions

- We will focus on optimal speech only and did not consider movement
- Robot mood will correlate with human mood
- There are three considered states volume, pitch, and speed (1/low, 2/medium, 3/high). We will consider “2” to be the ideal state.
- We do not directly consider changes in the human mood in our model.



Approach

General Approach

1. Determine properties that should hold for system
2. Construct transition system that replicates possible interaction with human
3. Verify properties in transition system
4. Generate policy that follows properties
5. Implement on NAO

General Approach

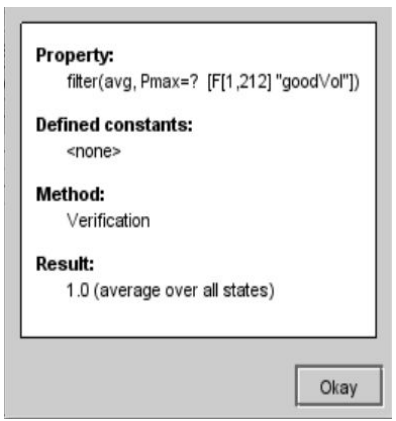
1. **Determine properties that should hold for system**
2. **Construct transition system that replicates possible interaction with human**
3. **Verify properties in transition system**
4. Generate policy that follows properties
5. * Implement on NAO

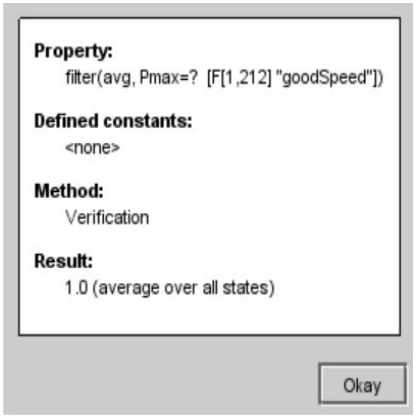
Properties

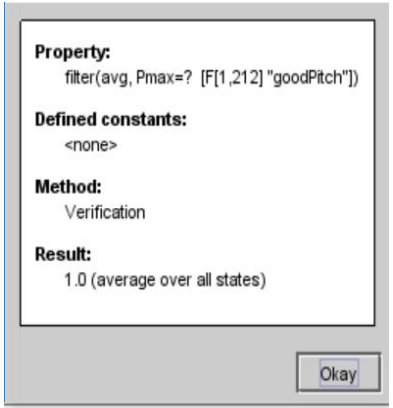
- filter(avg, Pmax=? [F[1,212] "goodVol"])
- filter(avg, Pmax=? [F[1,212] "goodSpeed"])
- filter(avg, Pmax=? [F[1,212] "goodPitch"])

```
label "goodVol" = (p4=2);
 // Volume being in desired range (2)
label "goodSpeed" = (p2=2);
// Speed being in desired range (2)
label "goodPitch" = (p3=2);
// Voice Pitch being in desired range (2)
```

Results of Verification







* For our model we found that it takes at least 212 steps to guarantee with probability 1.0 that the model reaches these properties.

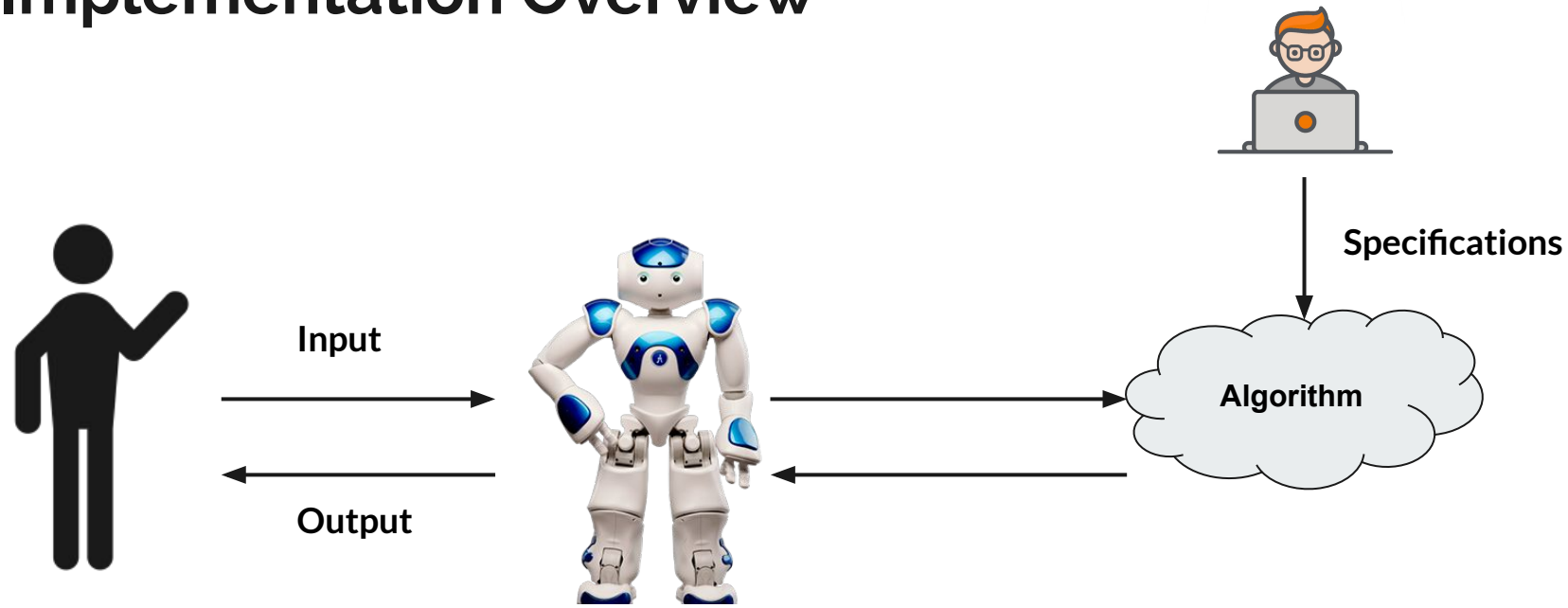
Model Overview

```
module Mood
        p1: [0..3] init 0;

        [] p1=0 -> 0.25 : (p1'=1) + 0.5 : (p1'=2) + 0.25 : (p1'=3);
        [] p1=1 -> 0.7 : (p1'=1) + 0.15 : (p1'=2) + 0.15 : (p1'=3);
        [] p1=2 -> 0.25 : (p1'=1) + 0.5 : (p1'=2) + 0.25 : (p1'=3);
        [] p1=3 -> 0.15 : (p1'=1) + 0.15 : (p1'=2) + 0.7 : (p1'=3);

endmodule

module Speed = Mood [p1=p2] endmodule
module Pitch = Mood [p1=p3] endmodule
module Volume = Mood [p1=p4] endmodule
```

Built Model

States: 256
Initial states: 1
Transitions: 3072

Implementation Overview



Conclusion

Conclusion

- There are not many applications for non-safety critical systems
- So far, we have verified that our model satisfies properties relevant and important to this application
- Next, we will look into using a solver to generate outputs for the NAO robot and complete an implementation

Questions?