Towards Real-Time Natural Language Corrections for Assistive Robots

Alexander Broad[1,4] , Jacob Arkin[2] , Nathan Ratliff[3], Thomas Howard[2] and Brenna Argall[1,4]

[1]Northwestern University, Evanston, IL 60208 [2]University of Rochester, Rochester, NY 14627
[3]Lula Robotics Inc., Seattle, WA, 98102 [4]Rehabilitation Institute of Chicago, Chicago, IL, 60611

Motivation

- Human operators should be able to modify properties of how their robotic counterpart achieves a goal on the fly
- Autonomously designed trajectories are often not optimal from a safety perspective and do not take into account a user's preferences
- Natural language is an intuitive and extensive communication scheme
- Particularly important for users with motor impairments

Figure 1. The MICO robotic arm on a power wheelchair with user. http://www.kinovarobotics.com/

Approach

- Generalizable natural language interface provides corrective instructions to an assistive robotic manipulator in real-time (Fig. 1)
- Distributed Correspondence Graph (DCG) [1] assigns semantic meaning to user utterances
- Desired corrections alter the behavior of the robotic manipulator by treating the modifications as constraints on the motion generation paradigm

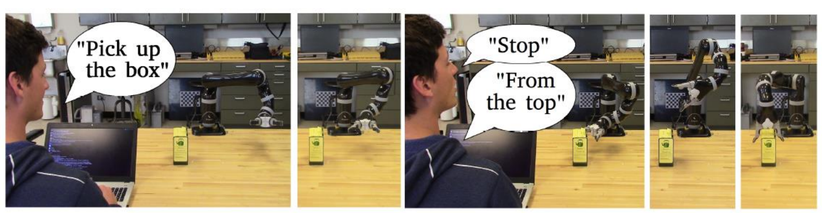

Figure 1. Example workflow. From left to right : (1) User issues initial command. (2) Arm begins to move and user recognizes undesirable action. (3) User stops arm and provides desired correction to motion. (4) Arm begins moving again, this time with new planning parameters. (5) Arm completes the task with user correction applied.

[1] T.M. Howard, S. Tellex, and N. Roy. "A Natural Language Planner Interface for Mobile Manipulators". In: Proceedings of the Int. Conf. on Robotics and Automation (ICRA). IEEE, May 2014, pp. 6652– 6659.

Distributed Correspondence Graph

- Goal : Find the most likely trajectory, $x_t^*(\mathrm{t})$, from a natural language correction (Λ) in the context of the perceived environment (Υ)

$$\Phi^* = \arg\max_{\phi_{ij} \in \Phi} \prod_i \prod_j p\left(\phi_{ij} | \gamma_{ij}, \lambda_i, \Gamma_{c_{ij}}, \Upsilon\right)$$

- First : Generate a parse tree of the correction (Fig. 2)
- Second : Use hierarchical structure to develop DCG model (Fig. 3)

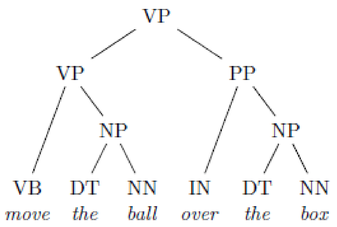

Figure 2. DCG for the instruction "move the ball over the box."

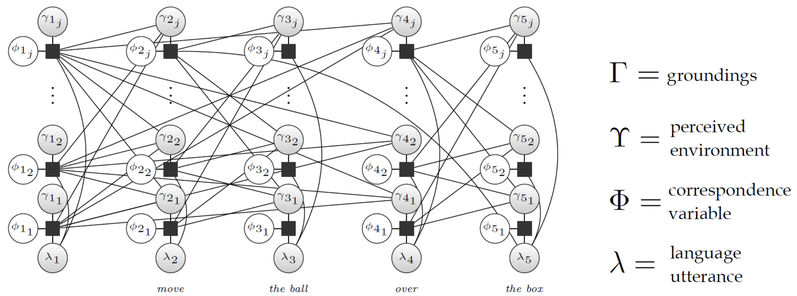

Figure 3. DCG for the instruction "move the ball over the box."

Dimensions of Correction

1) The **speed** with which the manipulator executes a trajectory
2) The **orientation** of the manipulator's end effector
3) The **position of the end effector** relative to other objects in the environment
4) The **gross contact point** during grasping motions

Pilot Study

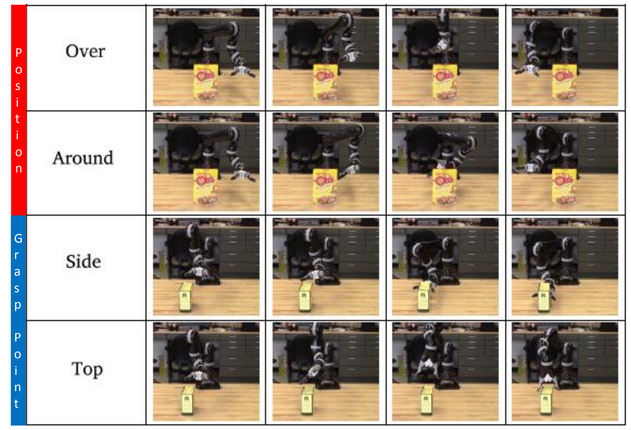

Figure 4. Example trajectory corrections

- Three non-expert users
- End-to-end system using a MICO robotic arm
- Analyze points of failure and direct future work

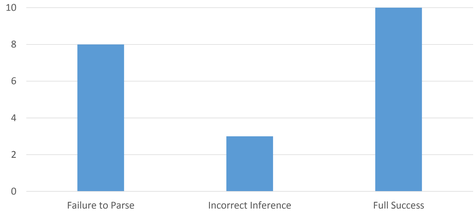

Figure 5. Failure and Success Counts

	Run-Time (seconds)
DCG Inference	0.0236
Planning	0.9 - 1.5

Table 1. System Run-Time

Discussion and Future Work

- Study provides strong evidence that our approach is effective and efficient.
- In failed trials, the main source of error was the size of our language corpus which can be easily extended.
- In future work we hope to show that this method can be used to learn user preferences over time.
- We also expect to include additional dimensions of correction including differentiating between nearby objects and desired grasping strength.