There is **funding available for 2 PhD/MS students** for 2 years in a NSF / National Robotics Initiative project called *"Improved safety and reliability of robotic systems by faults/anomalies detection from uninterpreted signals of computation graphs"*. This is the original proposal: http://purl.org/censi/research/preprints/fault_proposal/nri_fault_project_description.pdf. The basic theme is using learning/identification techniques to detect faults and anomalies with "zero design effort", i.e. without adding any complexity to the existing system.
The following are two examples of possible projects for students that highlight the CS and control sides of the project.

If interested, please get in touch with Andrea Censi (censi@mit.edu).

Project #1 (computer science side)

Disciplines: robotics, software engineering, machine learning, identification, cloud computing
Main scientific question: *To what degree we can detect and faults in a robotic system, without knowing anything in advance about the system?*
Starting approach: Learn low-level models of the sensorimotor dynamics from the signals that are available for logging in the system's computation graph (i.e. "topics" in a ROS applications). Modify the computation graph to introduce supervisory components to detect and mitigate faults based on learned models. Demonstrate system on actual platform doing service-robot-like tasks; offload computation/storage to cloud.
Main platforms: Kuka Youbot w/cameras,range-finders, etc; cloud computing using Amazon EC2
Skills required or to be acquired: GNU/Linux, Python programming, ROS.
Reading materials:
1) *Bootstrapping bilinear models of simple Vehicles* http://purl.org/censi/research/2013-jbds-sub.pdf
2) *Fault detection and isolation from uninterpreted data in robotic sensorimotor cascades* http://purl.org/censi/research/2012-fault.pdf
3) Chapter 2 of http://purl.org/censi/2012/phd

Project #2 (control theory/applied math side)

Disciplines: robotics, identification, group theory, machine learning, deep learning
Main scientific question: *How complex should a learning agent be to solve the "brain in a vat" bootstrapping challenge of learning to use any robotic body?*
Starting approach: Solve the learning problem in a very large but limited domain, namely the "set of all robots" R. Construct R from basic pieces (the "canonical" robotic sensors and actuators) plus composition operations. Further enlarge the set by allowing various "nuisances" that can act on the system (both noninvertible, like discretization, sampling, delays, noise; and invertible, like illumination and everything that can be modeled as a group action). Construct the set of learning agents similarly with "dual operators" that allow learning/rejecting/marginalizing those nuisances, and "decomposing" the compositions. Obtain control-theory-like guarantees on the closed-loop performance on given tasks. Demonstrate on physical platform.
Skills required or to be acquired: Python programming, scientific data analysis
Reading materials:
1) *Bootstrapping bilinear models of simple Vehicles* http://purl.org/censi/research/2013-jbds-sub.pdf
2) *Calibration by correlation by metric embedding from nonmetric similarities* purl.org/censi/research/preprints/camera_calibration/2012-camera_calibration.pdf
3) Chapter 2 + Part 3 of http://purl.org/censi/2012/phd