Robot Simulation Environments

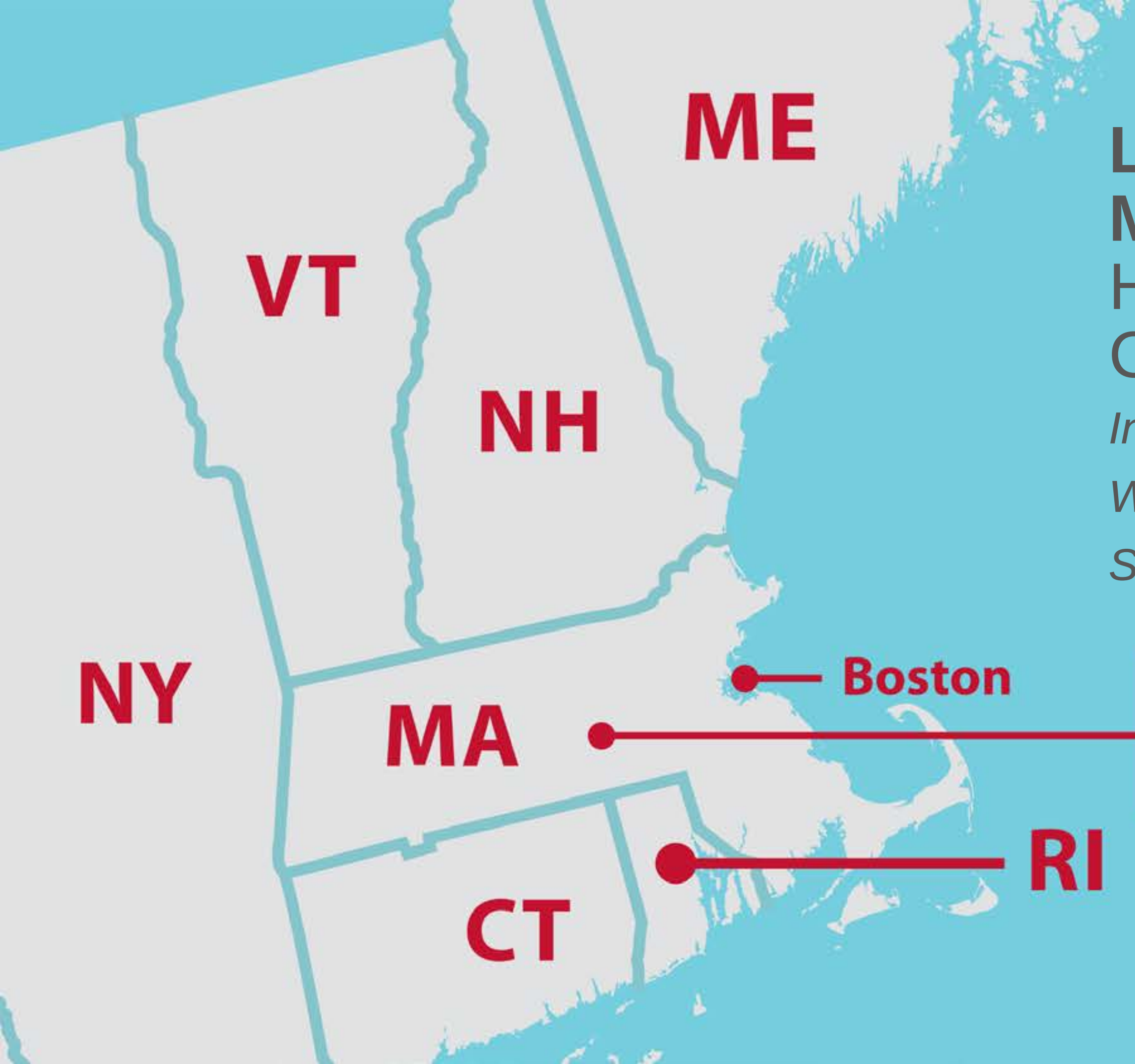
Gregory S. Fischer
Adnan Munawar

gfischer@wpi.edu amunawar@wpi.edu
http://aimlab.wpi.edu
@mrirobot

Agenda

1. Introduction
    a. Simulation Environments
    b. Modeling the dVRK
    c. Kinematic Parameter Identification
    d. Dynamic Parameter Identification
2. Asynchronous Multi-Body Framework (AMBF)
    a. Brief Intro & Components of AMBF
    b. The AMBF Format for Robot and Scene Description
    c. Graphical Assistance for Designing Robots and Scenes
    d. Real-Time Simulation and Control of Complex Surgical Robots
    e. Communication Payloads
    f. Support for Multiple Users Interacting with Simulation using Simulated End-Effectors
    g. Multi-port Camera Control and VR
3. A Python Client For Object Based Interaction
    a. GYM Compatible Interface for Training NN and RL Agents using TensorFlow of Theanos

INTRODUCTION

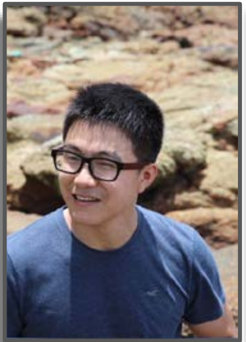


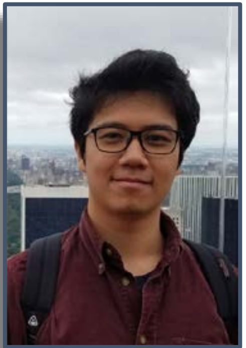
Located in Worcester, Massachusetts (the Heart of the Commonwealth)

Including locations at Gateway Park Worcester and the Seaport District in South Boston

WPI AIM Lab (dVRK Group)

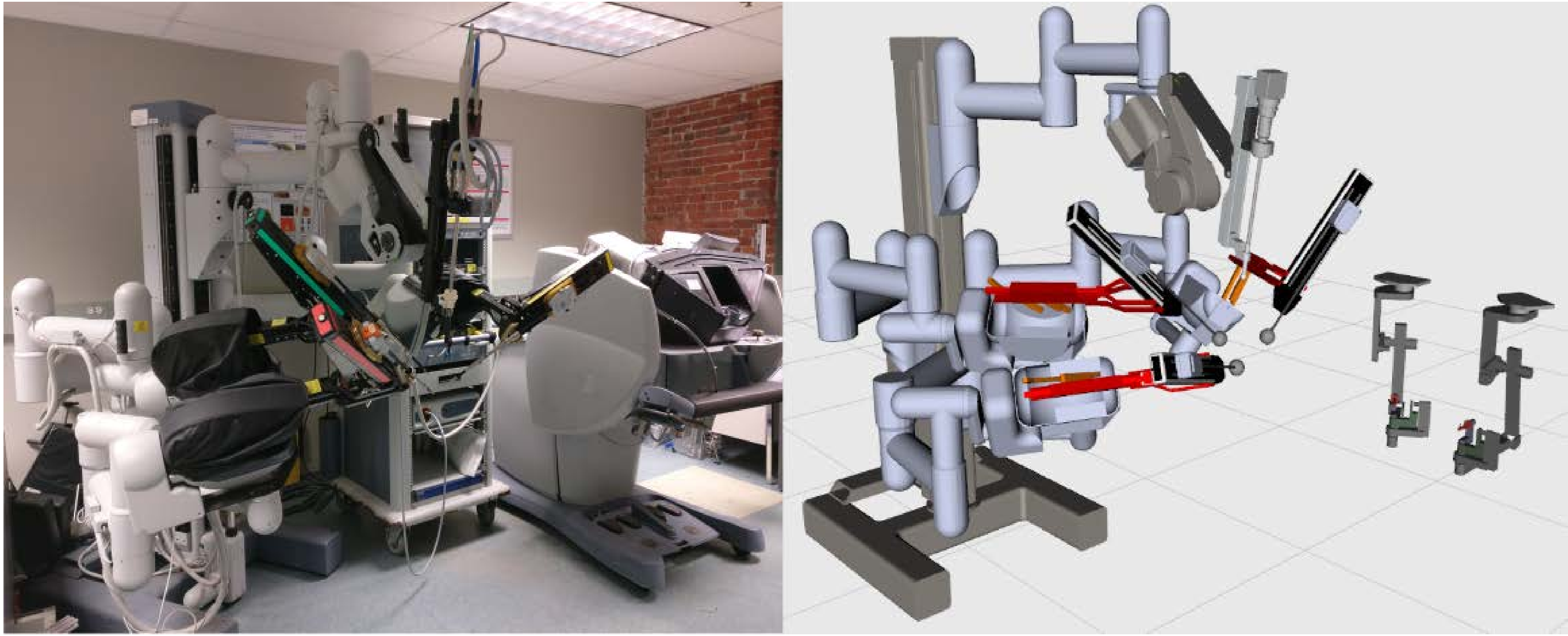
Greg Fischer



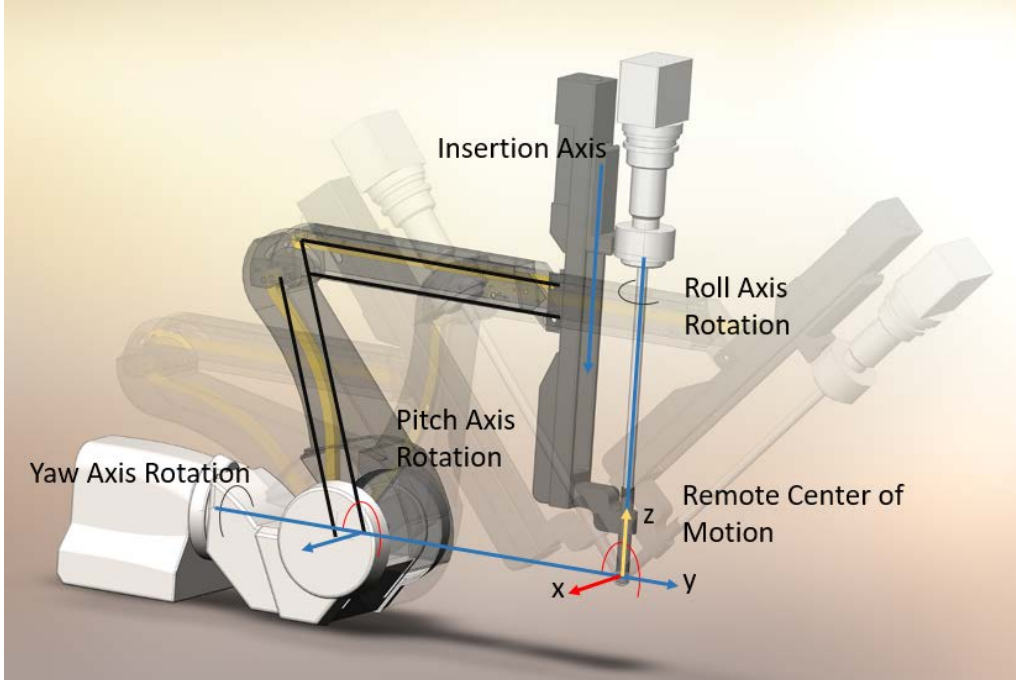


Adnan Munawar, Yan Wang, Radian Azhar, Ham Nuttaworn, Ankur Agarwal, Anna Novoseltseva

Model & ROS-Based Simulator for the dVRK



Development of Simulation Environment



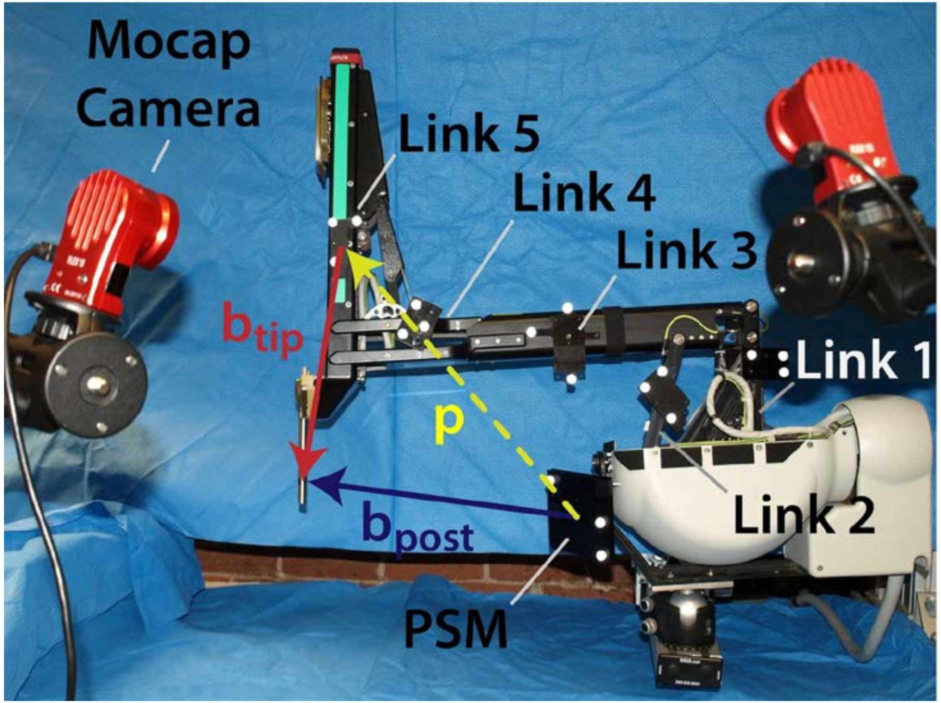
Procedure:

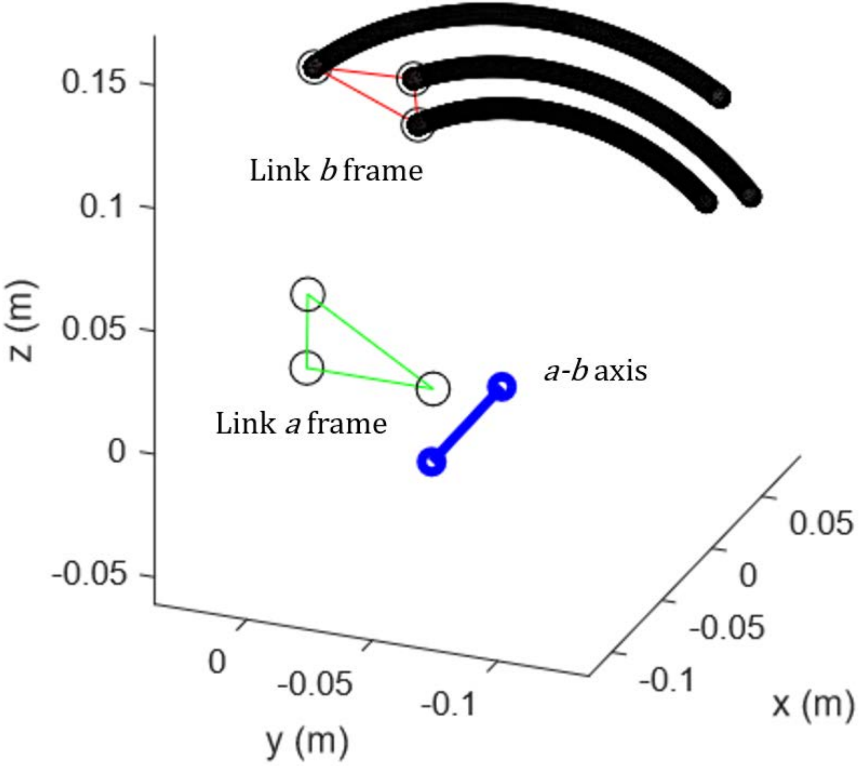
- Parameter Identification
- CAD and URDF export
- Parallel Linkage
- Modifications
- Gazebo Plugin and Control

Dynamic Parameter Estimation for Improved Haptics and Simulation

(Radian and Yan)

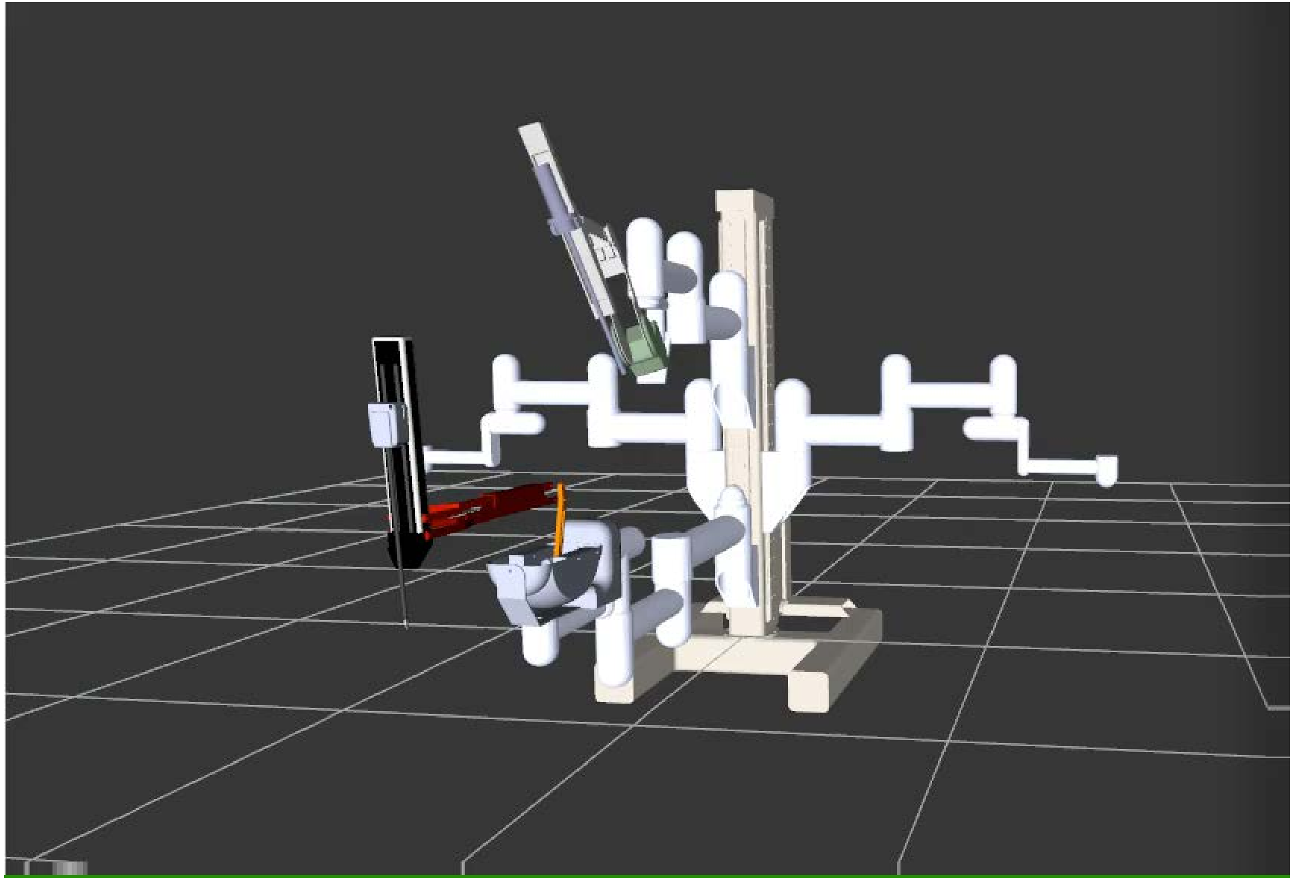
Kinematic Parameters of PSM/ECM Links





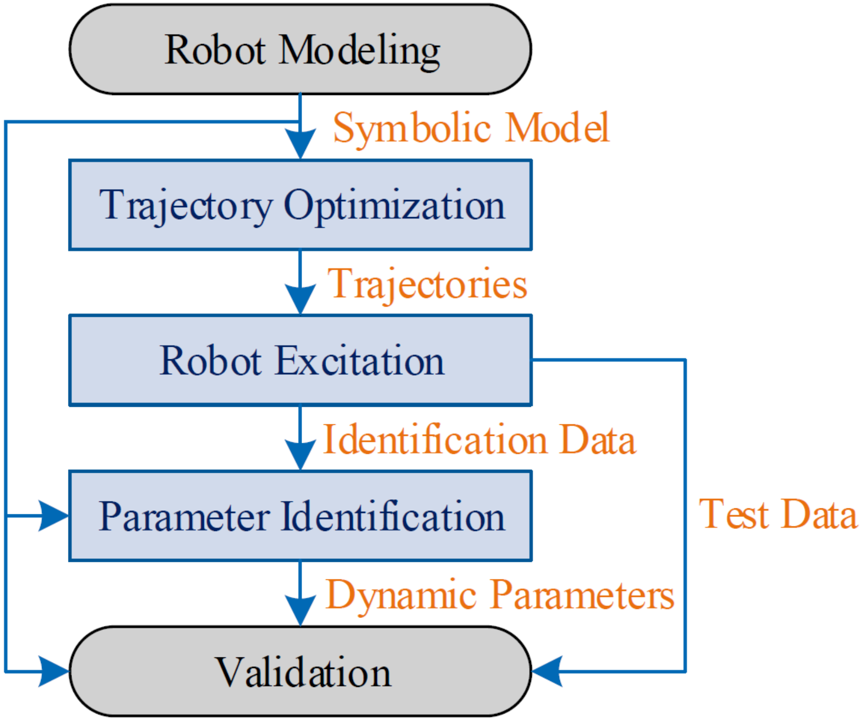
Accurate CAD Modeling of the daVinci

ROS-Based Modeling dVRK Patient-Side Manipulators



Our dVRK Setup

Dynamic Parameter Identification



Link inertia: $m_k, [l_x \quad l_y \quad l_z],$
$[L_{xx} \quad L_{xy} \quad L_{xz} \quad L_{yy} \quad L_{yz} \quad L_{zz}]$

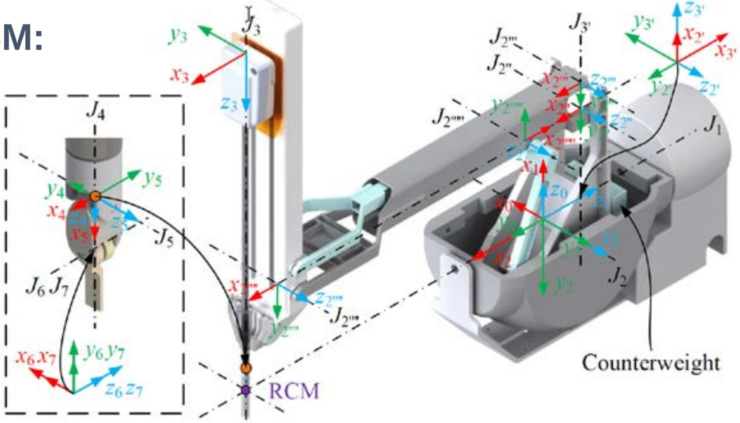
Motor inertia: $\boldsymbol{\tau}_m(\ddot{q}_a) = \boldsymbol{I}_m \ddot{q}_a$

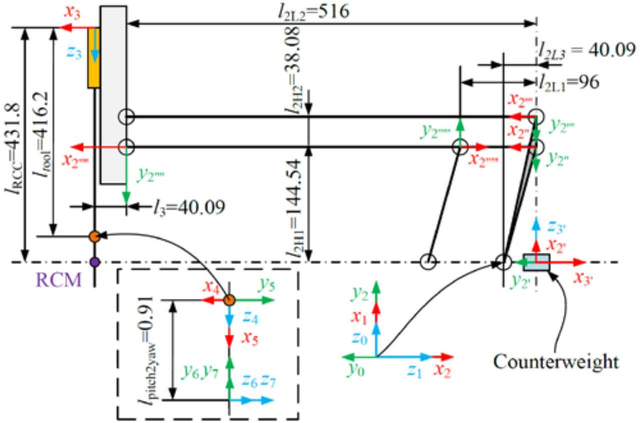
Friction: $\boldsymbol{\tau}_f(\dot{q}) = \boldsymbol{F}_v \dot{q} + \boldsymbol{F}_c \mathrm{sgn}(\dot{q}) + \boldsymbol{F}_o$

Spring: $\boldsymbol{\tau}_s(q) = \boldsymbol{K}_s \Delta \boldsymbol{l}_s$

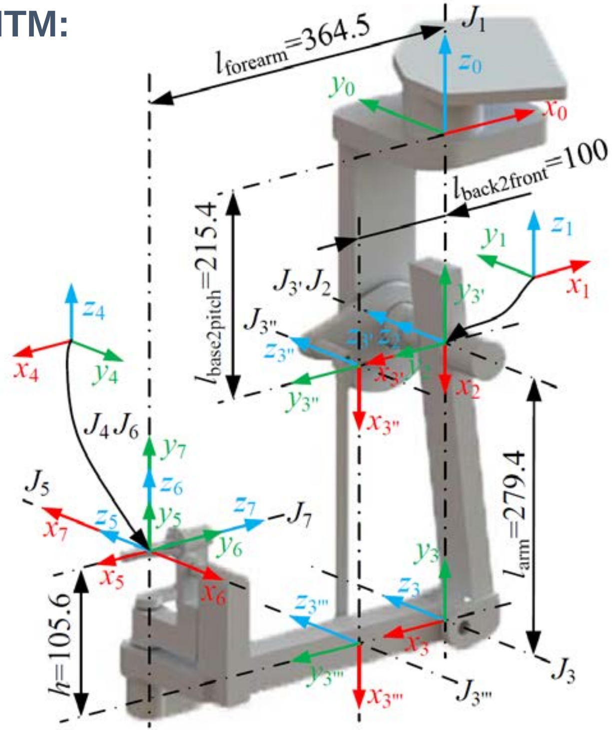
Frame Definitions & Kinematic Model:

PSM:



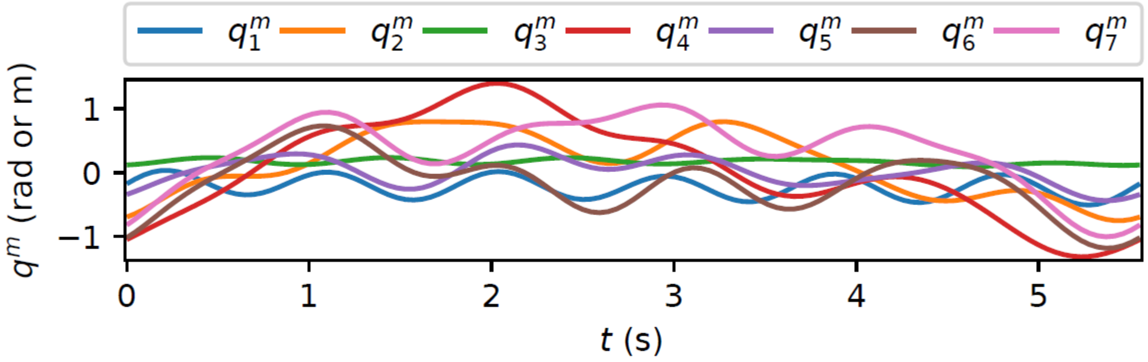


MTM:

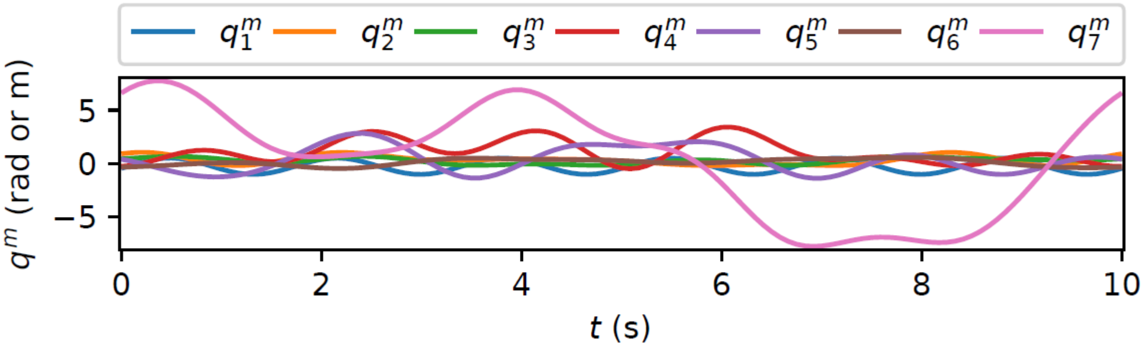


Trajectory Optimization:

PSM:

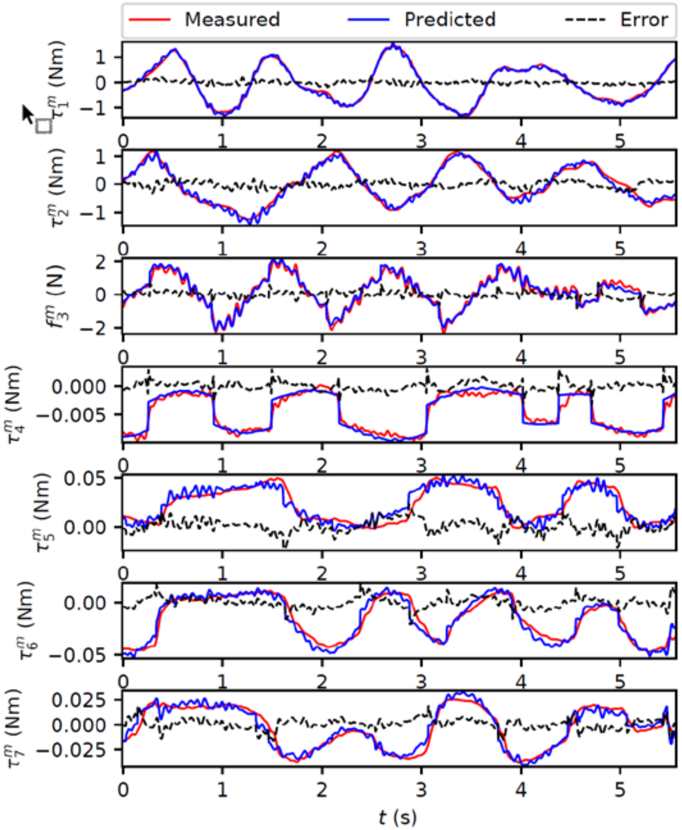


MTM:

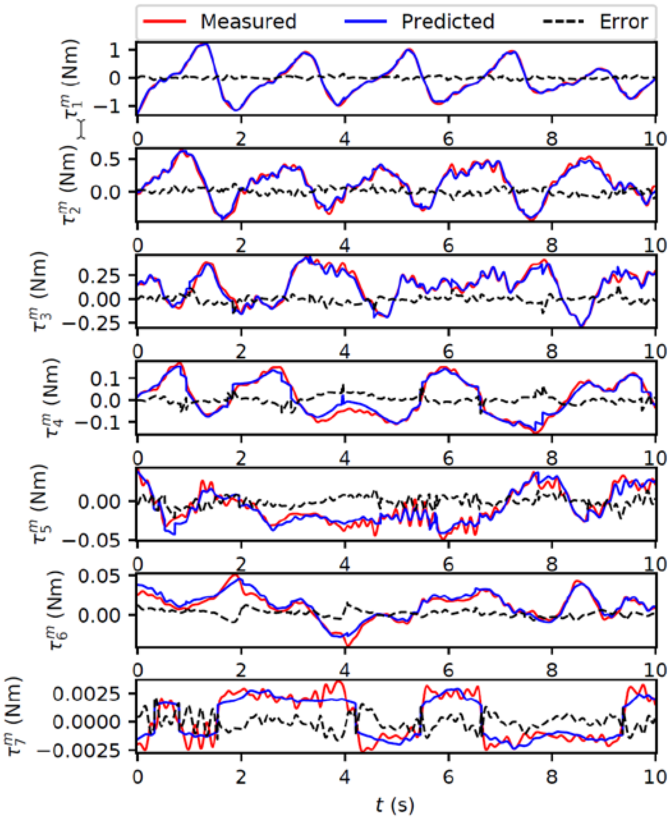


Validation: Predicted vs Measured Joint Torques

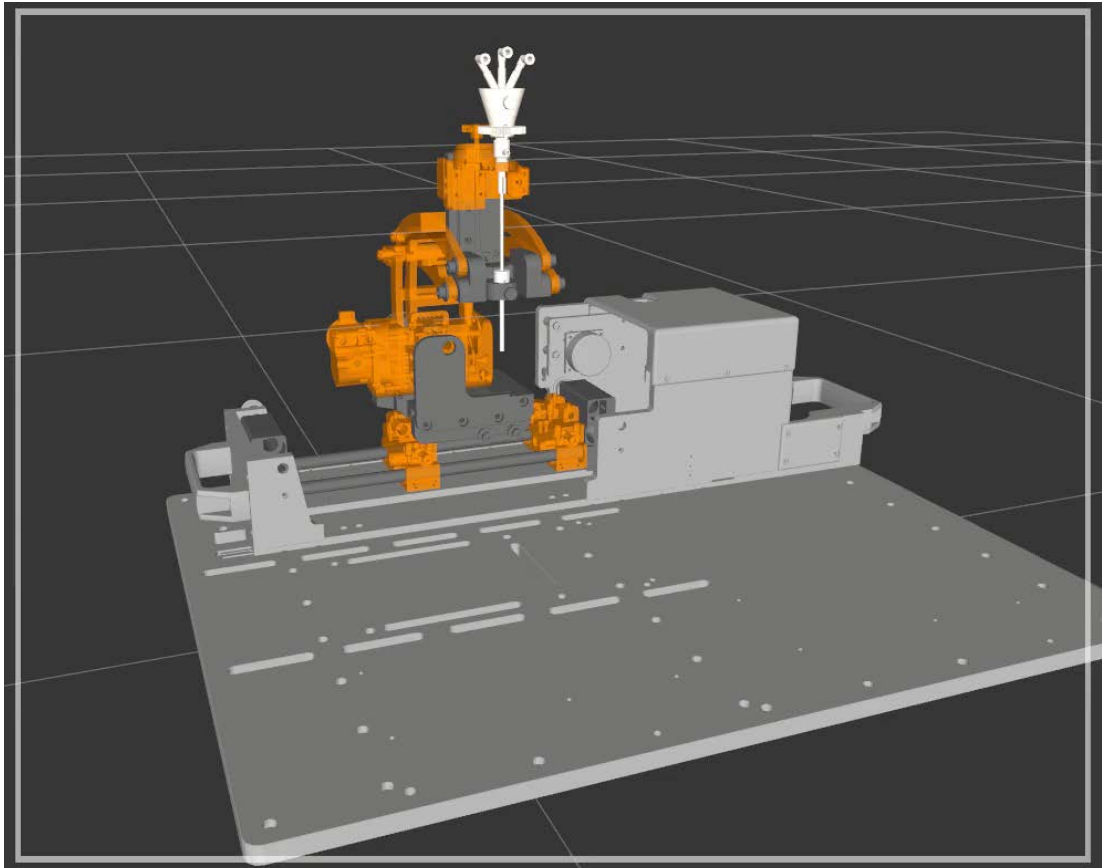
PSM:



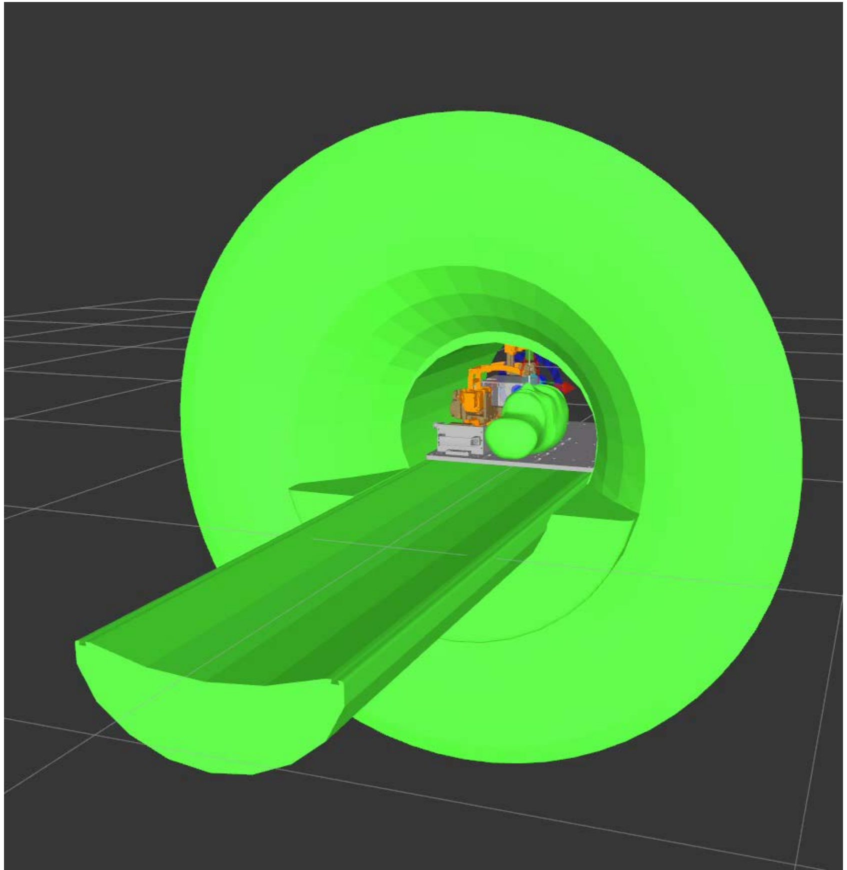
MTM:



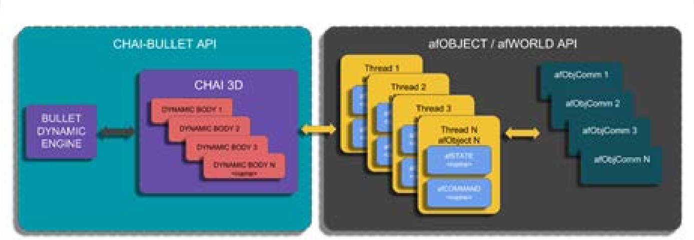
Applying rviz Simulation to our Neurosurgery System



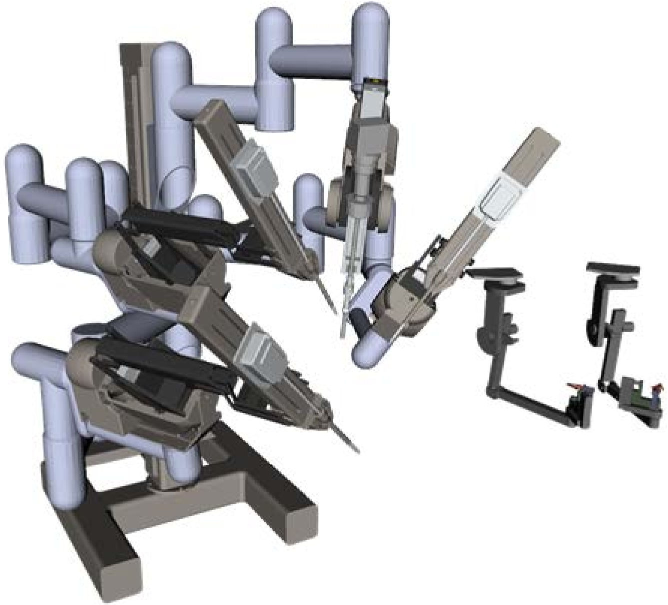
Collision Detection Based Workspace Analysis in MRI Bore



Simulation Environments (Gazebo, Matlab, AMBF)



https://github.com/WPI-AIM/ambf



https://github.com/WPI-AIM/dvrk_env

SECTION 2

The Asynchronous Multi-Body Framework

Existing Robot Simulators

- RViz: Go to ROS Simulator (Purely Kinematics)
- Gazebo: Popular with ROS Community (ODE, BULLET, DARTSim, …)
- MuJoCo (Popular with Machine Learning and RL Community)
- VRep (A very capable simulator by Copella Robotics)

- Key Limitations:
  - Across the Board Support for Closed Loop Mechanisms.
  - Inflexibility with Input Devices and handling increasing load from simulated objects
  - Except Rviz, communication pipeline appears to be mostly an after thought.
  - Name-spacing, the heart and soul of ROS, is a severely underrated feature often ignored in Robot Specification

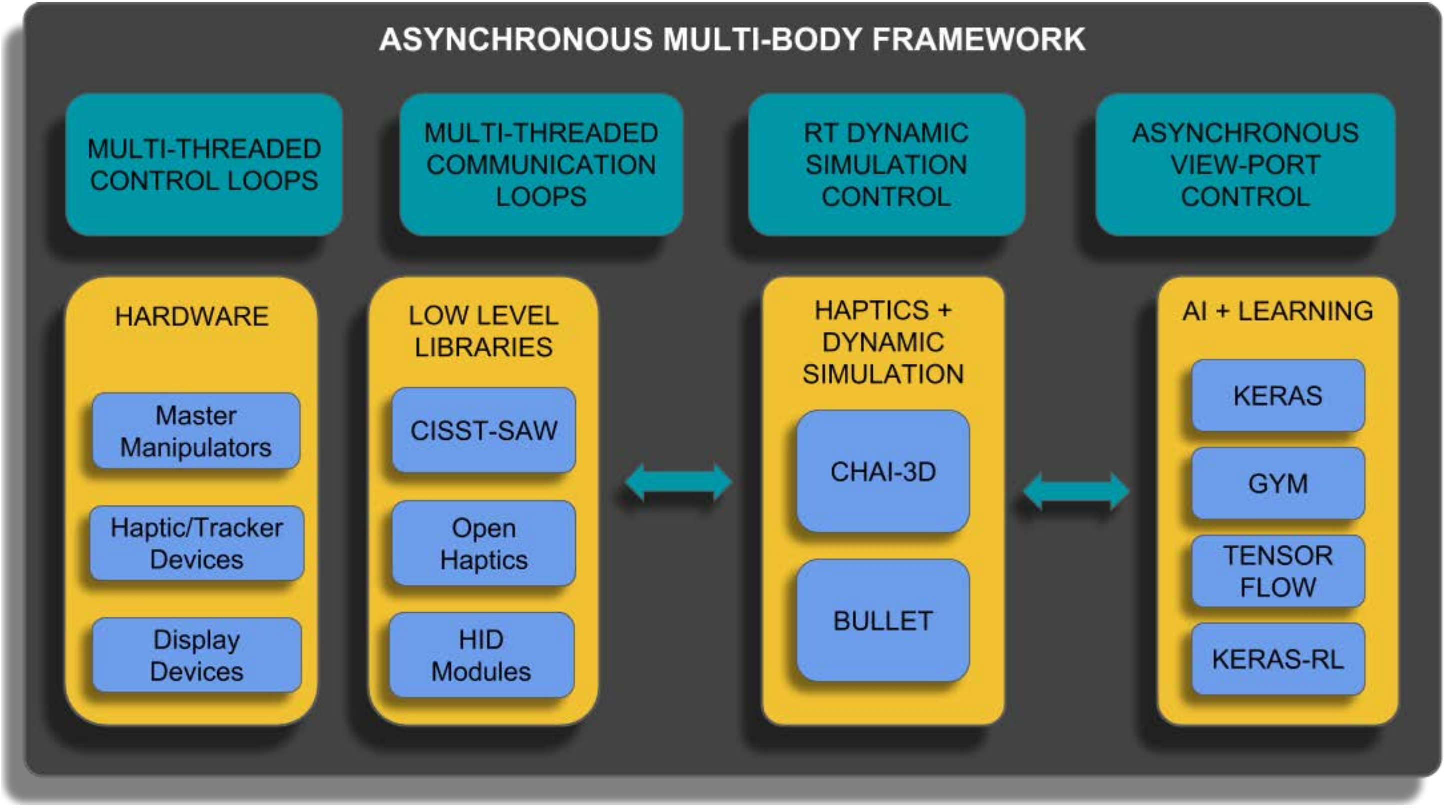
Key Limitations

AMBF Framework and Components

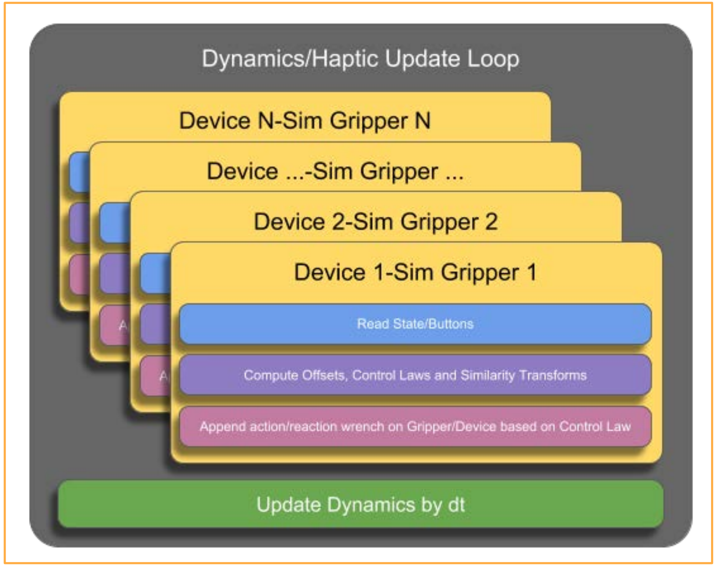
BLENDER AMBF ADDON	AMBF FRAMEWORK LIBRARY	AMBF ROS MODULES
URDF2AMBF ADDON	AMBF SIMULATOR	AMBF IPC COMMUNICATION PIPELINE

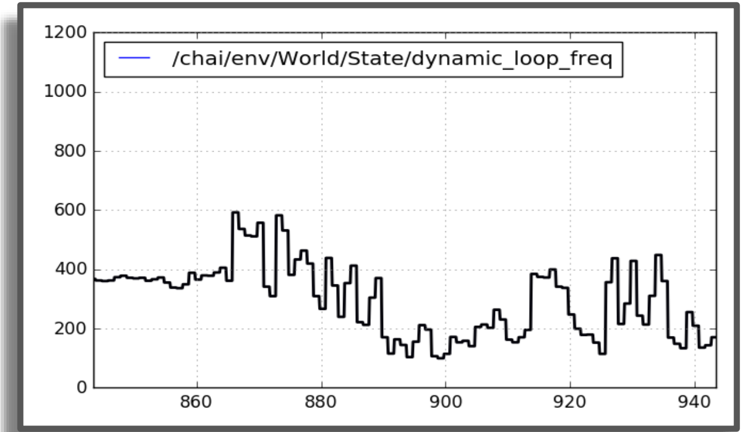
https://github.com/WPI-AIM/ambf

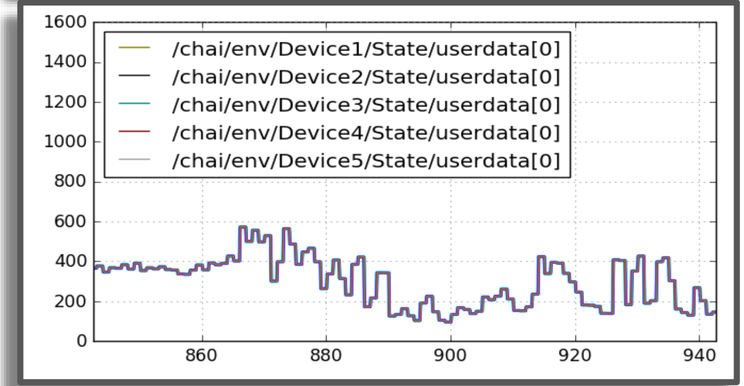
The AMBF Framework Library



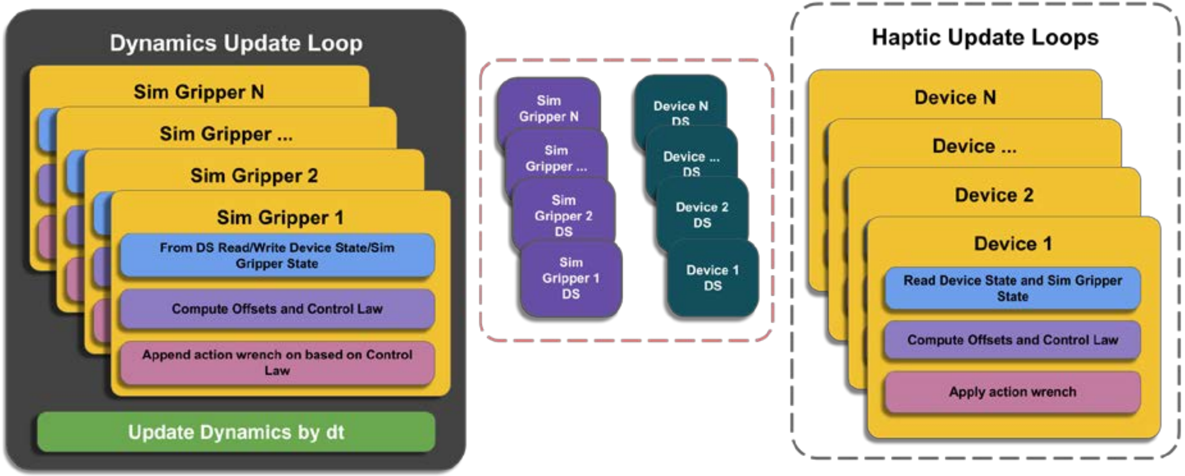
Sequential Device Control

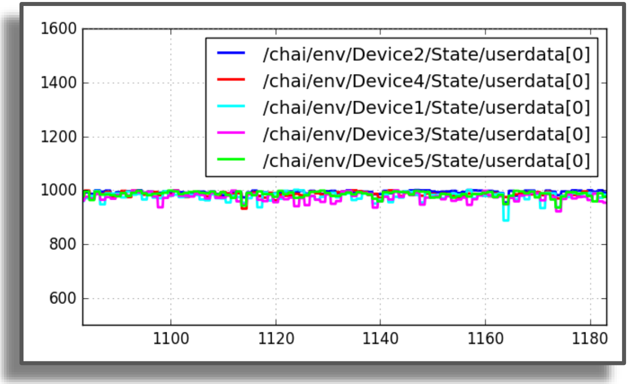


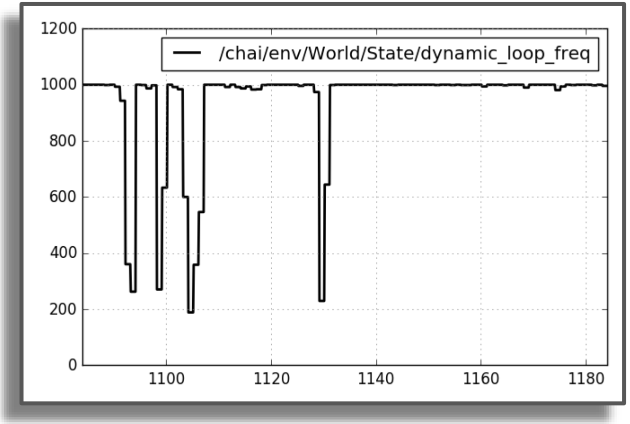




Asynchronous Device Control

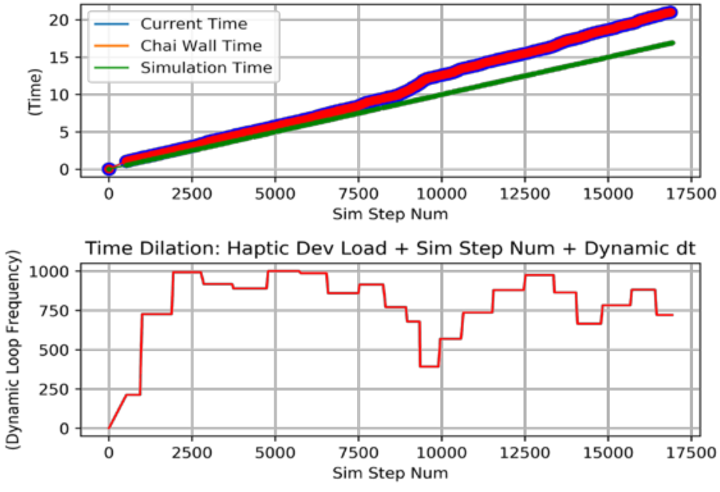




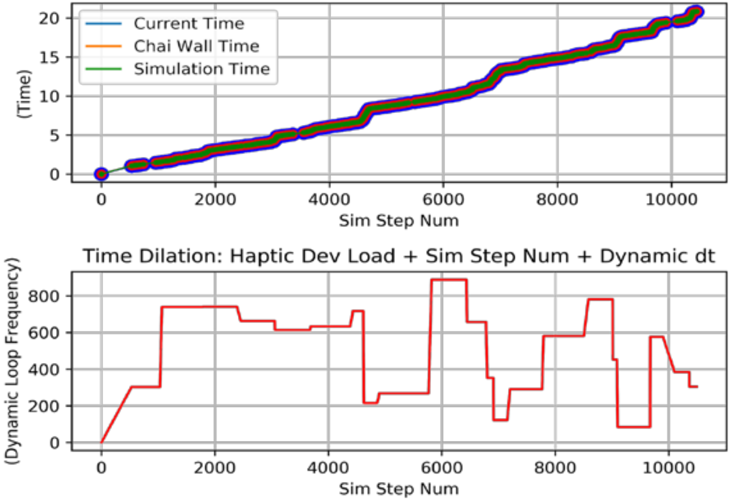


Tracking Time in a Dynamic-Haptic Simulation

- We make distinction between Haptic Update Rate and Dynamic Update Rate
- Haptic Update Rate is the rate of reading/writing io (reading state data and updating forces of the devices)
- Dynamic Update Rate is the update rate of stepping the dynamic solver



Fixed Time Step

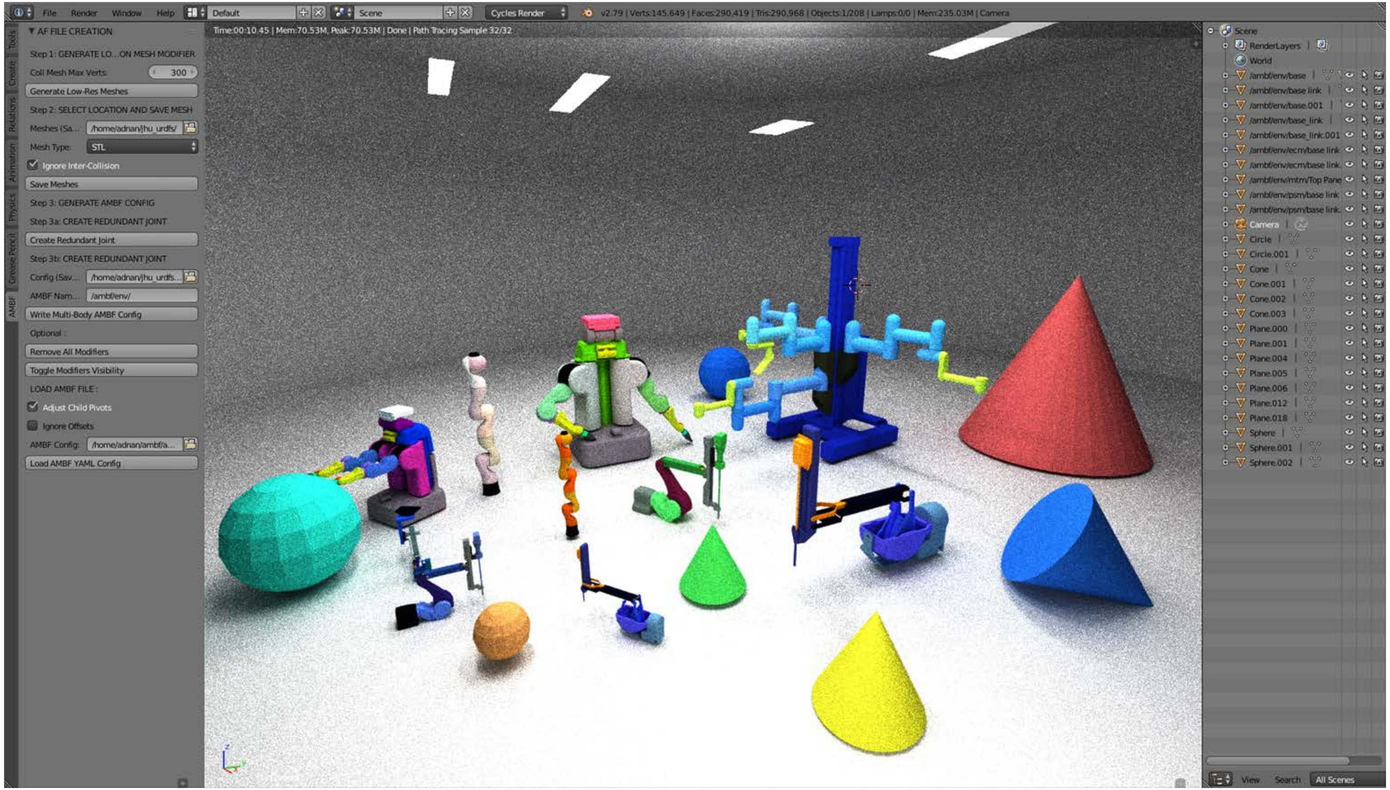


Dynamic Time Step

URDF 2 AMBF Converter

- Convenient script to convert existing URDF Models:
    - https://github.com/WPI-AIM/urdf_2_ambf
- Indigenous XML Tree parsing
- Fixing the Limitations due to distributed definition of URDF format by segregating visual offsets and placing them in joint information
- Overriding intermediate static links by warning the user and using super low inertial values.

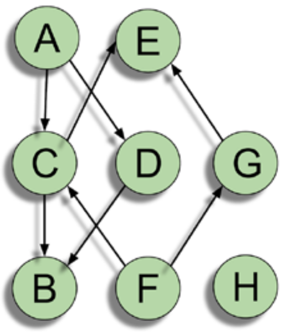
Blender AMBF Addon



https://github.com/WPI-AIM/ambf_addon

The AMBF Format

Densely Connected Body Tree



Bodies	Parents	Children
A	-	C, E, D, B
B	D, A, C, F	-
C	A, F	B
D	A	B
E	C, A, G, F	-
F	-	C, E, G
G	F	E
H	-	-

bodies: [Body A, Body B]
joints: [Joint A-B]
camera: [Camera Top]
lights: [Bulb]

high resolution path: ./meshes/high_res
low resolution path: ./meshes/high_res
namespace: /ambf/env/human/

Joint A-B:
- **parent:** Body A
- **child:** Body B
- **parent axis:** {x: 0, y: 0, z: 1}
- **parent pivot:** {x: 0, y: 0.1, z: 0.5}
- **child axis:** {x: 1, y: 0, z: 0}
- **child pivot:** {x: 0, y: -0.1, z: 0}
- **type:** revolute
- **controller:** {P: 10, I: 0, D: 0.1}

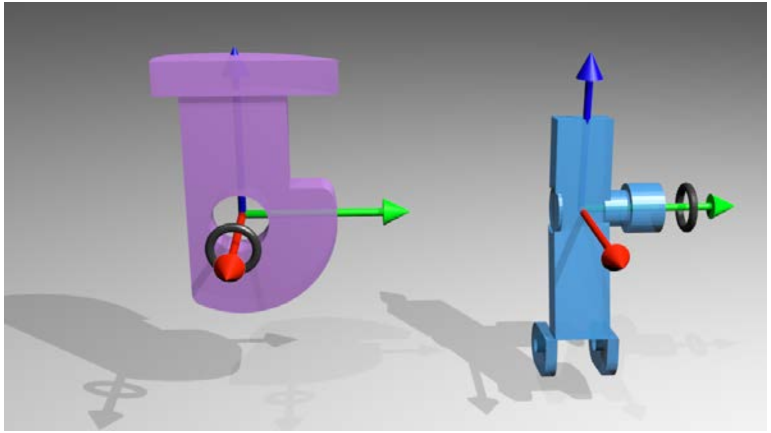
Body A:
- **name:** Torso
- **location:**
  - **position:** {x: 0, y: 0, z: 0}
  - **orientation:** {r: 0, p: 0, y: 0}
- **mass:** 45
- **inertia:** {ix: 6, iy: 10, iz: 8}
- **mesh:** Torso.stl

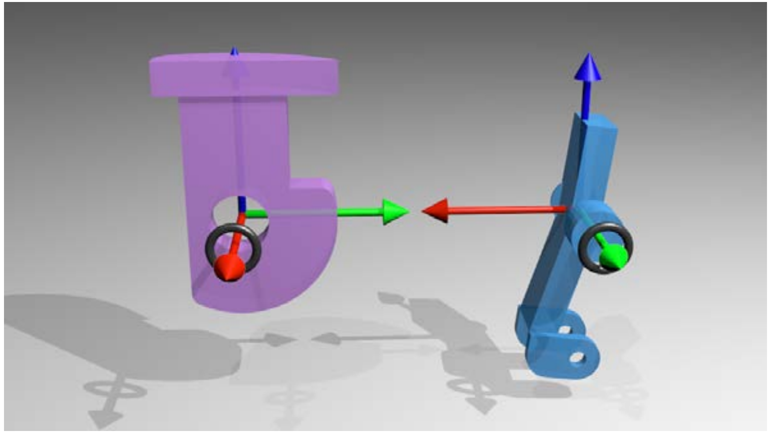
Body B:
- **name:** Head
- **inertial offset:**
  - **position:** {x: 0, y: 0, z: 0}
- **namespace:** /movable/
- **mass:** 10
- **linear controller:** {P: 5, D: 0.1}
- **mesh:** Head.stl

Bulb:
- **location:** {x: 0.5, y: -0.5, z: 2.5}
- **direction:** {x: 0, y: 0, z: -1.0}
- **spot exponent:** 0.3
- **shadow quality:** 5
- **cutoff angle:** 0.7

Camera Top:
- **location:** {x: 3.0, y: -3.0, z: 2.0}
- **look at:** {x: 0.0, y: 0.0, z: -0.5}
- **up:** {x: 0.0, y: 0.0, z: 1.0}
- **clipping:** {near: 0.01, far: 10.0}
- **field view angle:** 0.7
- **controlling devices:** [Falcon]

Intuitive Joint Constraint Definition





Directly use constraint axes and offsets rather than an intermediate constraint frame with fixed RPY

Additional Features

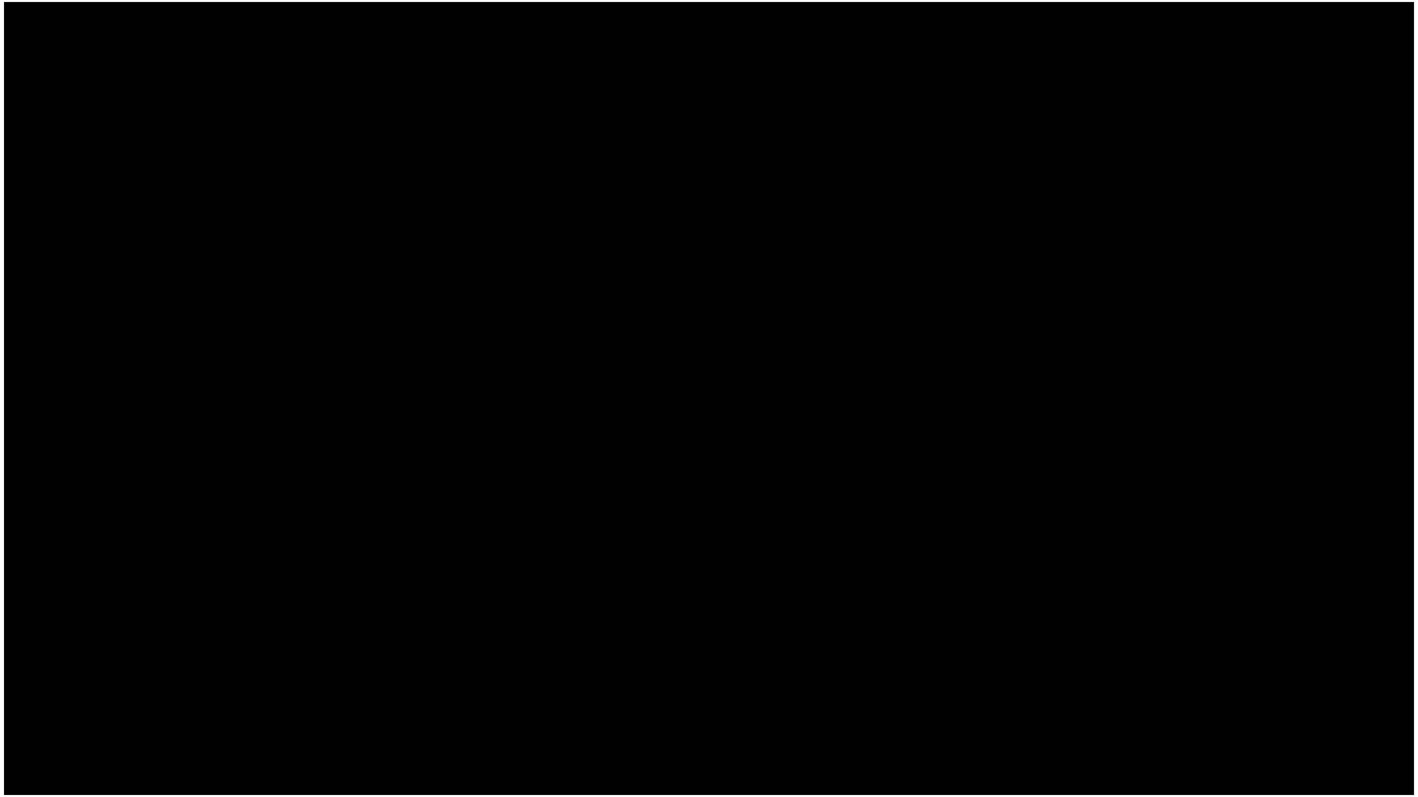
- Adding cameras adds viewports and new windows
- Specifying Stereo View Params for VR
- Setting monitor for each camera
- Binding Input Devices to Camera Frames
- Parenting supported across different AMBF config files
- Breaking down a single robot into multiple AMBF Config Files
- Intuitive Collision Grouping
- Support for Global Parameters that can ben overridden by Local Parameters
- Support for Sensors (Proximity Only for now)
- Support for Soft-Bodies

Communication Payloads

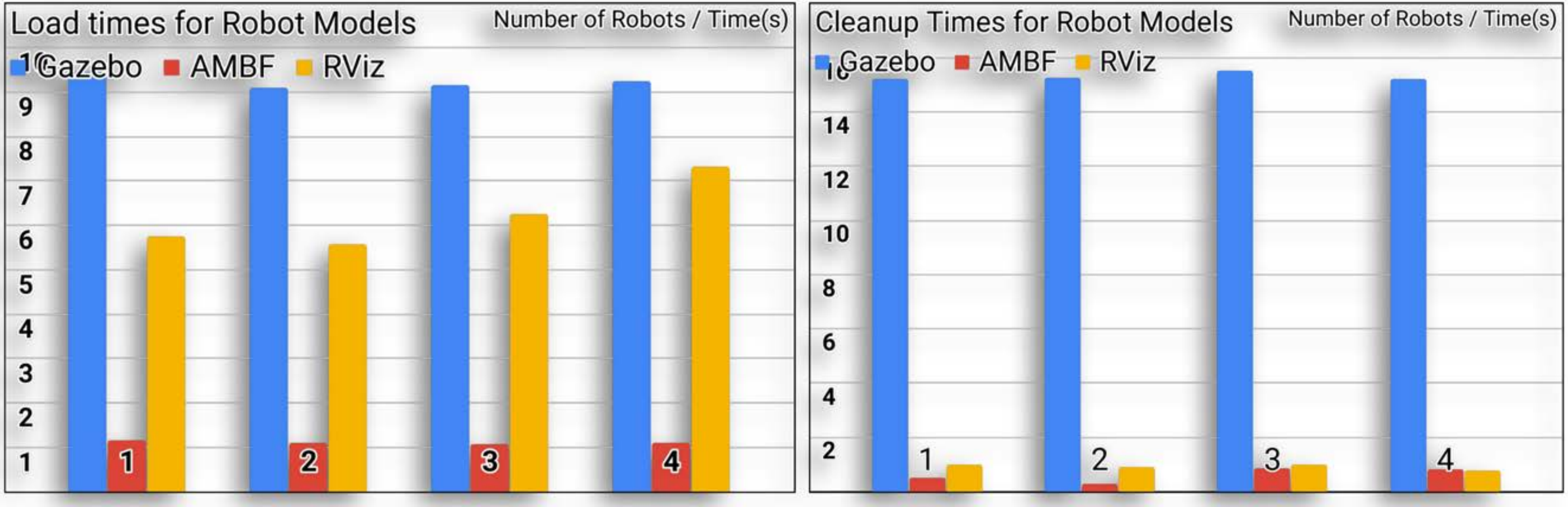
afWorld Instance	
State	**Command**
Dynamic Loop Frequency	Enable Step Throttling
Number of Devices	Step Clock
Simulation Step Number	Number of Steps to Skip
Wall Time	-
Simulation Time	-

afObject Instance	
State	**Command**
Header	Header
Inertial Data	Enable Position Controller
Pose	Pose Command
Accumulated Wrench	Wrench Command
Children Names	-
Joint Names	Joint Commands
Joint Positions	Joint Position Control Mask

The AMBF Simulator



Relative Startup Speed of AMBF Simulator

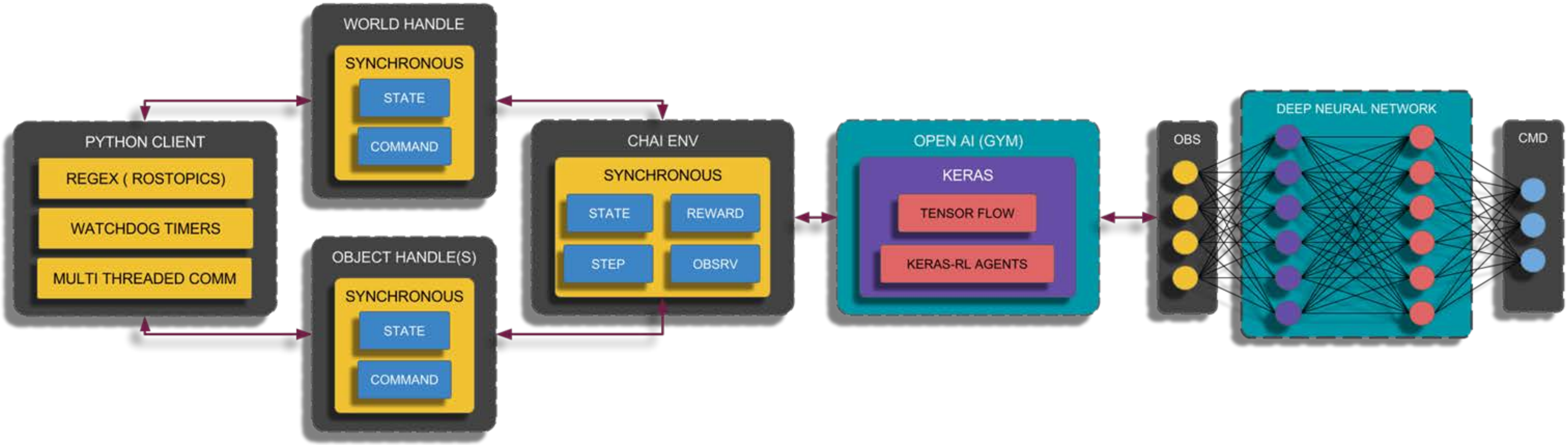


SECTION 2

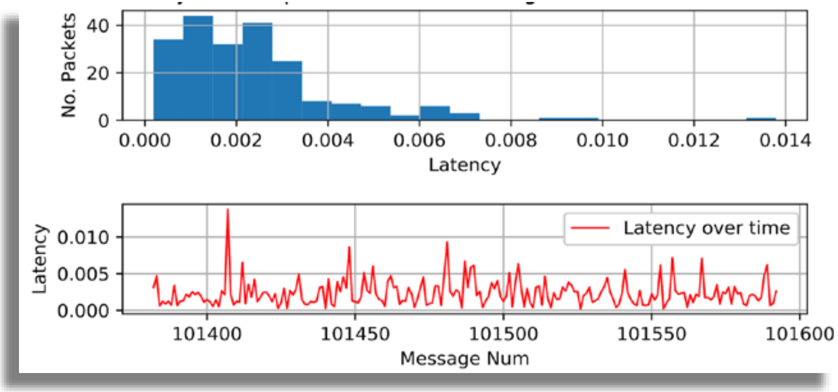
A Python Client for Training NN and RL Agents

The Coordination Client (Python)

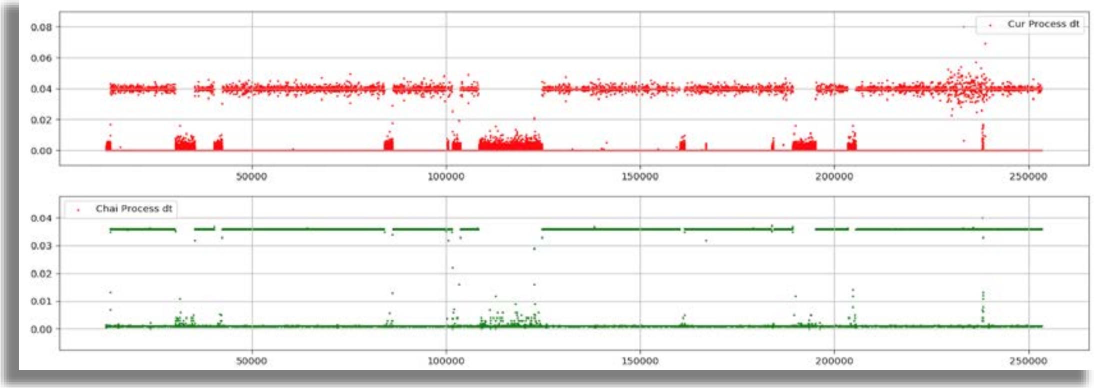
- Asynchronous client for bi-directional communication with AMBF Simulator
- Direct interface with Keras using Command-State Interface (Tensorflow and Theano)
- Control of simulation stepping and throttling
- Probing of afObjects using human readable strings.
- Watchdog timers for resetting commands



Python Clients Communication Performance



Histogram of the time difference between the embedded time of a received packet and the current time for synchronous communication using Step Throttling



Comparing the time offset between each subsequently received packet from the previous packet vs difference between subsequent read times

Thank you!

Links to Software Repositories:

AMBF: https://github.com/WPI-AIM/ambf

URDF2AMBF: https://github.com/WPI-AIM/urdf_2_ambf

Blender AMBF ADDON: https://github.com/WPI-AIM/ambf_addon

DVRK_ENV: https://github.com/WPI-AIM/dvrk_env

Contact Email:

gfischer@wpi.edu

amunawar@wpi.edu

References:

http://aimlab.wpi.edu/publications/