Checkpoint presentation. Project #24

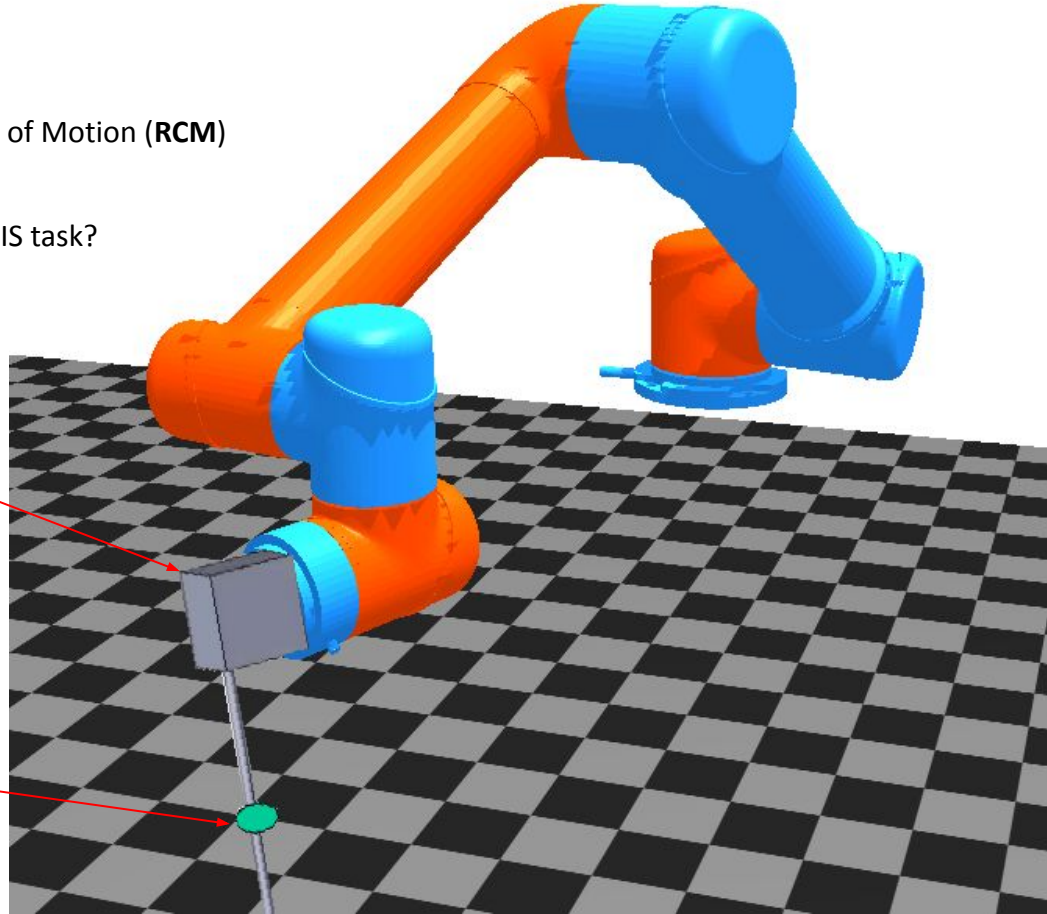
Evaluation of Virtual Remote Center of Motion for Minimally Invasive Surgery

Team members: Yernar Zhetpissov
Mentors: Adnan Munawar, Peter Kazanzides

04/02/2022 Computer Integrated Surgery II

Project background

- Minimally invasive surgery (**MIS**) requires Remote Center of Motion (**RCM**)
- Not everyone has access to special MIS surgical systems
  - daVinci
- Feasibility of robot manipulators (e.g. UR5) applied for MIS task?
- Dynamic simulator **AMBF**



Objectives

1. Implement virtual RCM for MIS in AMBF simulator

2. Evaluate it (collision during bimanual tasks, accuracy, joint limitations)

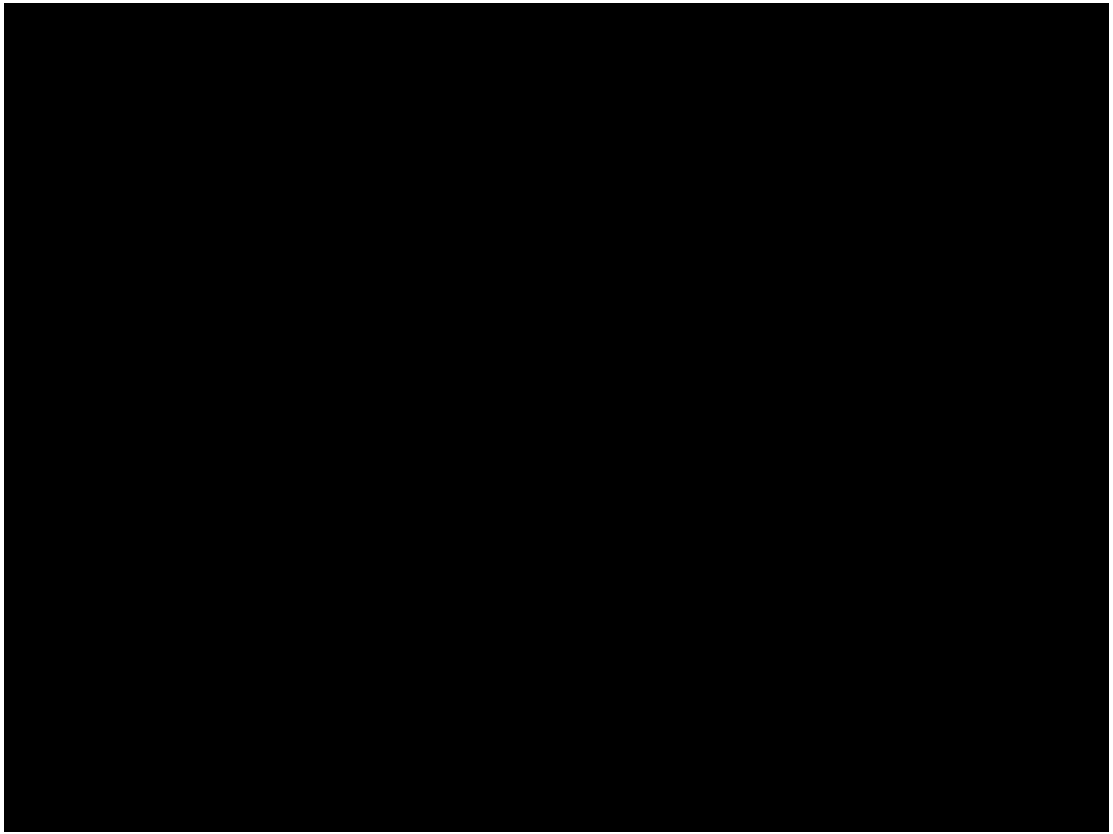
Importance:

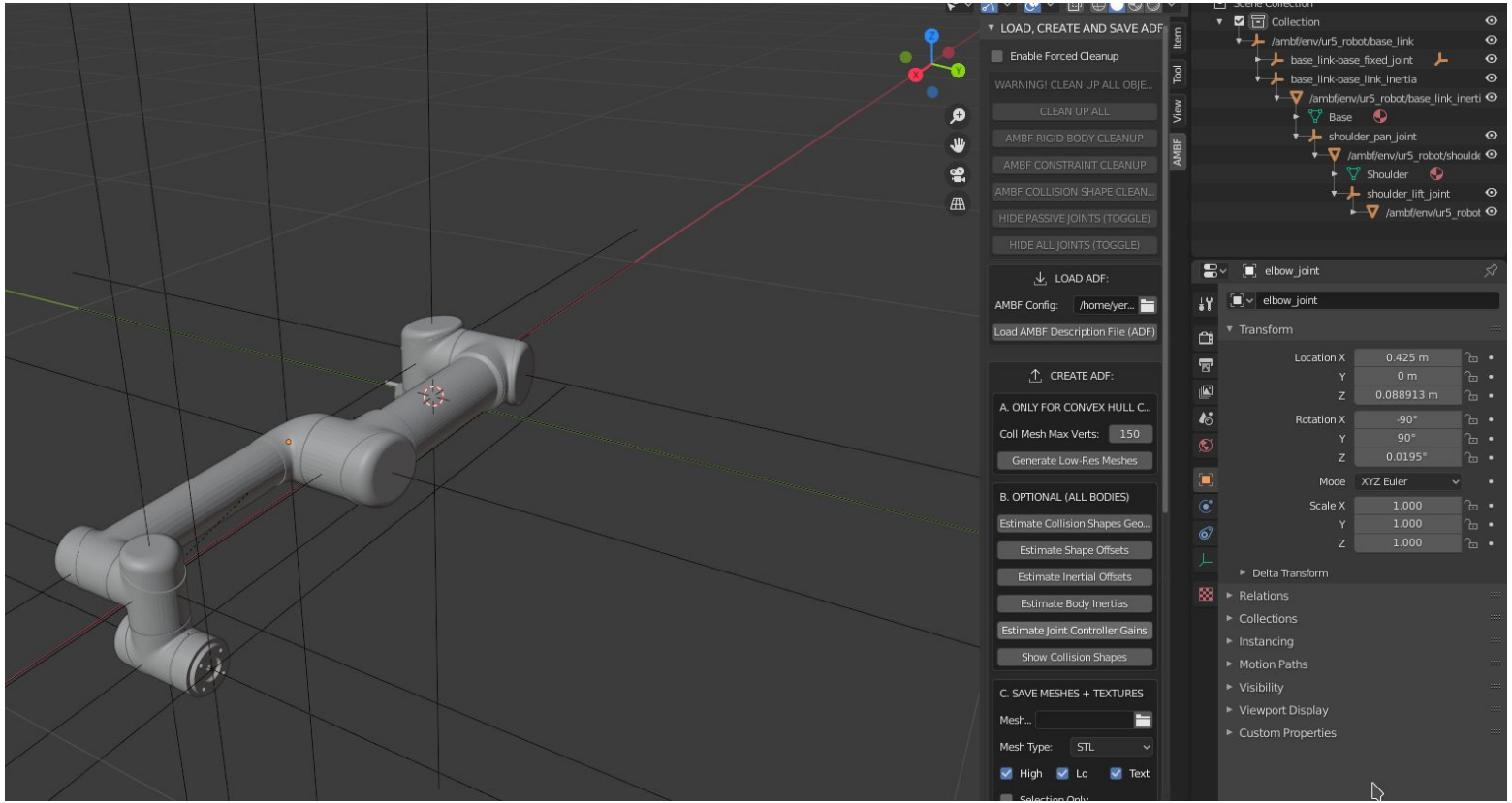
- The project would help to evaluate capabilities and constraints of manipulator arms applied in (bimanual) MIS tasks
- Would open the door for MIS related simulations using manipulator arms

Progress and Technical approach

DONE:

- UR5 imported to AMBF
- Forward and Inverse Kinematics implemented using KDL library





UR5 model import to AMBF config file

KDL - Kinematics and Dynamics Library, an application independent framework, useful for computations on kinematic chains

```
ur5_fk::ur5_fk () {

  //Initializing UR5 chain using DH parameters
  KDL::Chain chain();
  chain.addSegment(KDL::Segment(KDL::Joint(KDL::Joint::RotZ),KDL::Frame::DH(0, M_PI/2, 0.089159, 0)));
  chain.addSegment(KDL::Segment(KDL::Joint(KDL::Joint::RotZ),KDL::Frame::DH(-0.425, 0, 0, 0)));
  chain.addSegment(KDL::Segment(KDL::Joint(KDL::Joint::RotZ),KDL::Frame::DH(-0.39225, 0, 0, 0)));
  chain.addSegment(KDL::Segment(KDL::Joint(KDL::Joint::RotZ),KDL::Frame::DH(0, M_PI/2, 0.10915, 0)));
  chain.addSegment(KDL::Segment(KDL::Joint(KDL::Joint::RotZ),KDL::Frame::DH(0, -M_PI/2, 0.09465, 0)));
  chain.addSegment(KDL::Segment(KDL::Joint(KDL::Joint::RotZ),KDL::Frame::DH(0, 0, 0.0823, 0)));

  kdl_fk_pos = new KDL::ChainFkSolverPos_recursive( chain );
  _kdl_ik_vel = new KDL::ChainIkSolverVel_pinv(chain);
  kdl_ik_pos = new KDL::ChainIkSolverPos_NR( chain, *kdl_fk_pos, *_kdl_ik_vel);

}
```

Main Page | Modules | Namespaces | Classes | Files

Kinematic Families

All classes to support kinematic families. More...

Classes

class	**KDL::Chain** This class encapsulates a **serial** kinematic interconnection structure. It is built out of segments. More...
class	**KDL::ChainFkSolverAcc** This **abstract** class encapsulates a solver for the forward acceleration kinematics for a **KDL::Chain**. More...
class	**KDL::ChainFkSolverPos** This **abstract** class encapsulates a solver for the forward position kinematics for a **KDL::Chain**. More...
class	**KDL::ChainFkSolverPos_recursive**
class	**KDL::ChainFkSolverVel** This **abstract** class encapsulates a solver for the forward velocity kinematics for a **KDL::Chain**. More...
class	**KDL::ChainFkSolverVel_recursive**
class	**KDL::ChainIkSolverAcc** This **abstract** class encapsulates the inverse acceleration solver for a **KDL::Chain**. More...
class	**KDL::ChainIkSolverPos** This **abstract** class encapsulates the inverse position solver for a **KDL::Chain**. More...
class	**KDL::ChainIkSolverPos_LMA** Solver for the inverse position kinematics that uses Levenberg-Marquardt. More...
class	**KDL::ChainIkSolverPos_NR**
class	**KDL::ChainIkSolverPos_NR_JL**
class	**KDL::ChainIkSolverVel** This **abstract** class encapsulates the inverse velocity solver for a **KDL::Chain**. More...
class	**KDL::ChainIkSolverVel_pinv**
class	**KDL::ChainIkSolverVel_pinv_givens**
class	**KDL::ChainIkSolverVel_pinv_nso**
class	**KDL::ChainIkSolverVel_wdls**

Next steps

1. Study constrained non-linear optimization (Chapter 5 of Scientific computing : an introductory survey by Michael T. Heath)
2. Implement vRCM using sawConstrainedController package
3. Evaluate implemented virtual RCM using recorded positions of instruments while performing bi-manual MIS task

To achieve the virtual RCM point, we use a constrained optimization formulation based on [11]. In particular, we optimize the following cost function:

$$||J_h(q)\Delta q - Gf|| \quad (2)$$

subject to the constraint:

$$||P_{cl} + J_{cl}(q)\Delta q - P_o|| \leq \epsilon \quad (3)$$

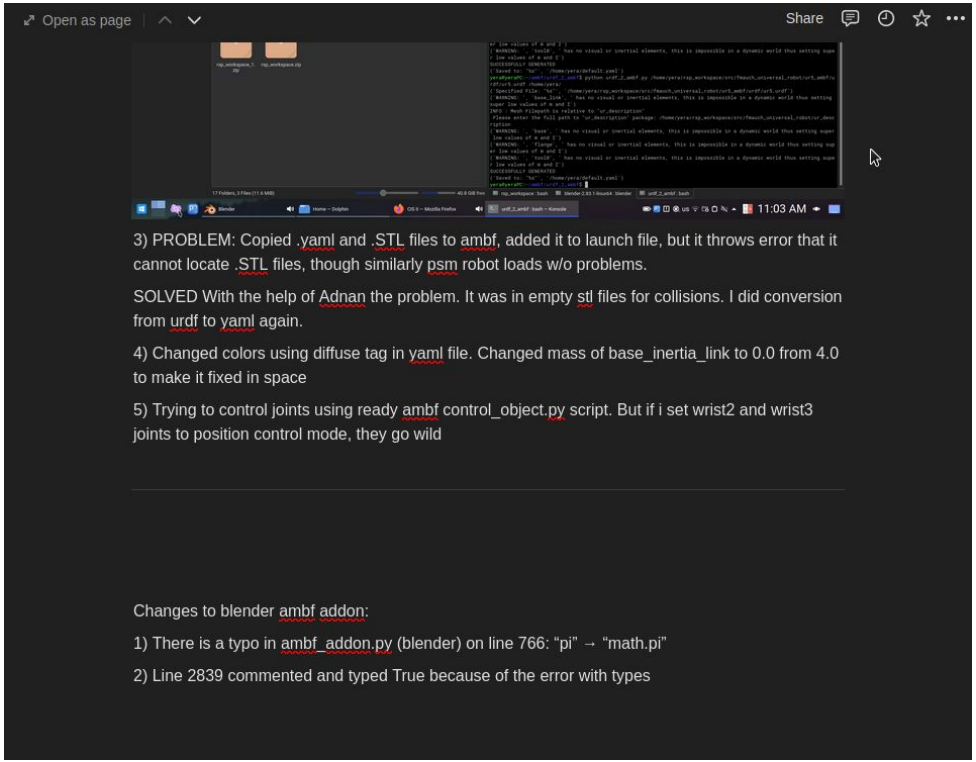
Here, P_{cl} is a point on the robot tool that is closest to the virtual RCM point, P_o. $J_h(q)$ and $J_{cl}(q)$ are the manipulator Jacobians resolved at the handle and at P_{cl}, respectively, and q is the vector of joint positions (see Fig. 5). The output of this optimization is the desired joint velocity vector, Δq, which becomes the input to an inner velocity control loop.

B. Mitchell et al., "Development and Application of a New Steady-Hand Manipulator for Retinal Surgery"

Documentation

! Development project

1. Software architecture (code uploaded to github)
2. RCM implementation background
3. Literature review (vRCM papers and nonlinear constraints optimization)



Problems

- Solving problem with importing robot to AMBF
- Learning as I am doing, depending on ROS learning
- Management problems emerging from that

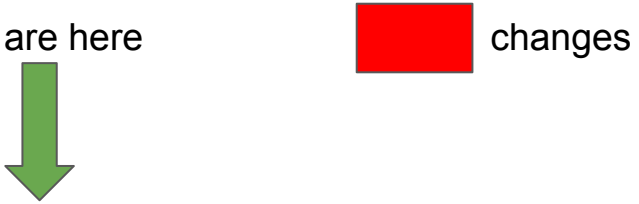
Deliverables

min: UR5+dVrk instrument kinematics models and RCM plugins for AMBF (most of the work)

expected: Evaluation report using recorded bag data on UR5

max: Demonstration on real UR5 with mock instrument driver attached (high probability of being cancelled/shifted beyond the class timing)

Milestones+Schedule (updated)



		Weeks 3-5	Week 6	Week 7	Week 8	Week 9	Week10	Week11	Week12	Week13	Week14
Setup	AMBF Installation										
	Study AMBF wiki										
UR5 model	UR5 CAD model import										
	Forward & inverse Kinematics										
UR5 control+RCM	study sawConstraintsControl pkg										
	integration of UR5 control driver with AMBF										
	RCM implementation										
Evaluation	Testing stage										
	Evaluation stage										
	Evaluation report										
Demo on real UR5 (might be dropped)											

Dependencies (updated)

1. Linux workstation (**done**)
2. Recorded Cartesian motion of two da Vinci instruments working on a bimanual task, such as the suturing task in the AccelNet Surgical Robotics Challenge. (**Ready recorded data to be provided**)
3. Robot and daVinci instruments (provided by prof.Kazanzides)
4. 3D printer might be needed for maximum deliverable (unresolved). **Resolving plan:** tape the driver on the manipulator tightly

Project management

- Progress tracking Zoom meetings with mentors (~~every 2 weeks~~ every week, Thursday)
  - Peter Kazanzides (RCM)
  - Adnan Munawar (AMBF+implementation)
- Questions over e-mail when necessary before meetings
- Implementation management via Github

Thank you!