HOCHSCHULE RAVENSBURG-WEINGARTEN UNIVERSITY OF APPLIED SCIENCES

Mujiburrehman Gouri

RWU

NEURA ROBOTICS

Dynamic Load Carrying Capacity of a Robot Manipulator

Metzingen, Germany

May 5, 2021

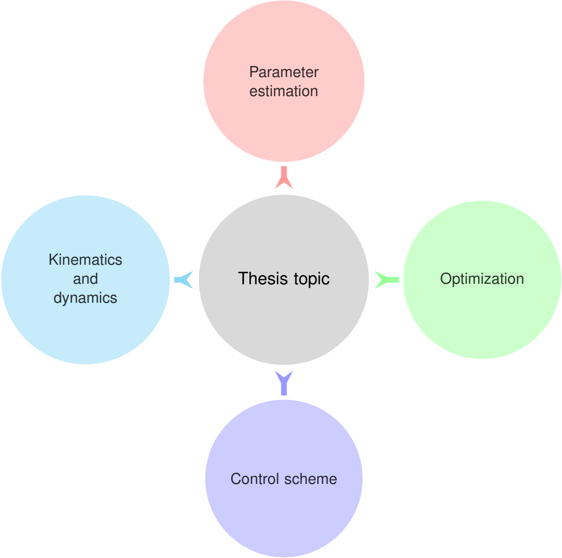
Prof. Dr.-Ing. Lothar Berger
Prof. Dr.-Ing. Raphael Ruf
Hochschule Ravensburg-Weingarten

Dipl.-Ing. Mavarick Ho
Neura Robotics GmbH

Overview

Introduction

Topic of investigation



What is DLCC

Figure: LARA robot

- Maximum load robot can carry
- Joint torque decides the load carrying capacity of the robot
- Dynamic load depends upon the end-effector motion
- Payload capacity of LARA
- Payload capacity changes for configuration close or away from robot

What is DLCC

Figure: LARA robot

- Maximum load robot can carry
- Joint torque decides the load carrying capacity of the robot
- Dynamic load depends upon the end-effector motion
- Payload capacity of LARA
- Payload capacity changes for configuration close or away from robot

What is DLCC

Figure: LARA robot

- Maximum load robot can carry
- Joint torque decides the load carrying capacity of the robot
- Dynamic load depends upon the end-effector motion
- Payload capacity of LARA
- Payload capacity changes for configuration close or away from robot

What is DLCC

Figure: LARA robot

- Maximum load robot can carry
- Joint torque decides the load carrying capacity of the robot
- Dynamic load depends upon the end-effector motion
- Payload capacity of LARA
- Payload capacity changes for configuration close or away from robot

What is DLCC

Figure: LARA robot

- Maximum load robot can carry
- Joint torque decides the load carrying capacity of the robot
- Dynamic load depends upon the end-effector motion
- Payload capacity of LARA
- Payload capacity changes for configuration close or away from robot

What is the Goal

- Incorporate robot dynamics in trajectory planning
- $\tau = H(q)\ddot{q} + c(q, \dot{q}) + \tau_g(q)$
- Robot motion can have joint jerks for payload exceeding limits
- Minimizing the joint jerks reduces the wear and tear
- Optimization problem is formulated

What is the Goal

- Incorporate robot dynamics in trajectory planning
- $\tau = H(q)\ddot{q} + c(q, \dot{q}) + \tau_g(q)$
- Robot motion can have joint jerks for payload exceeding limits
- Minimizing the joint jerks reduces the wear and tear
- Optimization problem is formulated

What is the Goal

- Incorporate robot dynamics in trajectory planning
- $\tau = H(q)\ddot{q} + c(q, \dot{q}) + \tau_g(q)$
- Robot motion can have joint jerks for payload exceeding limits
- Minimizing the joint jerks reduces the wear and tear
- Optimization problem is formulated

What is the Goal

- Incorporate robot dynamics in trajectory planning
- $\tau = H(q)\ddot{q} + c(q, \dot{q}) + \tau_g(q)$
- Robot motion can have joint jerks for payload exceeding limits
- Minimizing the joint jerks reduces the wear and tear
- Optimization problem is formulated

Problem statement

Mass: To obtain the physical properties of the end-effector object

Optimize: To optimize the trajectory for objective function

Control: To track the optimized trajectory

Topic importance

Why mass?	Precise motion in task space
Why Optimize?	Trajectory with minimum energy and reduced wear-tear
Why Control?	Follow the desired trajectory

Methodology

Inertial parameter estimation

A rigid body in space has following physical parameters,

1. Mass
2. Centre of mass
3. Moment of inertia

Inertial parameter estimation

A rigid body in space has following physical parameters,

1. Mass
2. Centre of mass
3. Moment of inertia

Inertial parameter estimation

A rigid body in space has following physical parameters,

1. Mass
2. Centre of mass
3. Moment of inertia

Mass estimation

1. Joint torque without mass (τ_1)
2. Fix the joint position
3. Joint torque with mass (τ_2)
4. Change in torque $\delta\tau = \tau_1 - \tau_2$
5. Jacobian (J) Wrench (f) calculation $f = {J^{T}}^{-1}\delta\tau$
6. $\text{mass} = \dfrac{f}{gravity}$

Centre of mass

1. Compute wrench (f) for different position
2. Fill the data matrix with end-effector force (F) and torque(M)
3. Avoid singular joint positions
4. Least squares estimation for centre of mass (r)

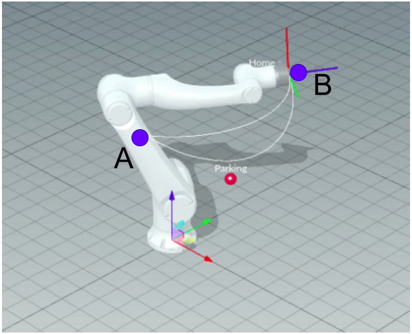
$$r = (F^T F)^{-1} F^T M$$

Optimization

Optimization problem is formulated for trajectory planning

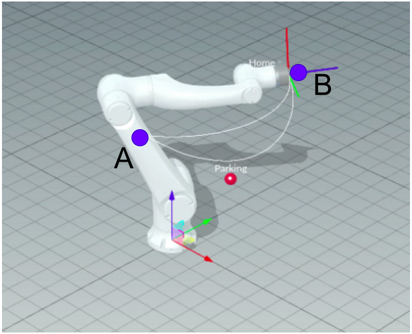
1. Cost functional formulation

Cost functional



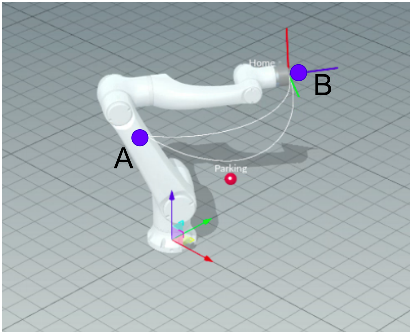
- Robot motion from initial to goal position
- First objective is to track the desired trajectory
- Secondary objective is to minimize the energy

Cost functional



- Robot motion from initial to goal position
- First objective is to track the desired trajectory
- Secondary objective is to minimize the energy

Cost functional



- Robot motion from initial to goal position
- First objective is to track the desired trajectory
- Secondary objective is to minimize the energy

First objective

- Tracking of desired (x_d) and current position (x_c) as $x_d - x_c$
- Nonlinear functions and inverse kinematics
- Tracking of velocity profile $\dot{x}_d - \dot{x}_c$
- Using jacobian(J), $\dot{x} = J_{(q)}\dot{q}$

First objective

- Tracking of desired (x_d) and current position (x_c) as $x_d - x_c$
- Nonlinear functions and inverse kinematics
- Tracking of velocity profile $\dot{x}_d - \dot{x}_c$
- Using jacobian(J), $\dot{x} = J_{(q)}\dot{q}$

First objective

- Tracking of desired (x_d) and current position (x_c) as $x_d - x_c$
- Nonlinear functions and inverse kinematics
- Tracking of velocity profile $\dot{x_d} - \dot{x_c}$
- Using jacobian(J), $\dot{x} = J_{(q)}\dot{q}$

First objective

- Tracking of desired (x_d) and current position (x_c) as $x_d - x_c$
- Nonlinear functions and inverse kinematics
- Tracking of velocity profile $\dot{x_d} - \dot{x_c}$
- Using jacobian(J), $\dot{x} = J_{(q)}\dot{q}$

Second objective

- Kinetic energy of the robot, $K.E = \frac{1}{2}\dot{q}^T H \dot{q}$
- Complex nonlinear terms and computation effort
- Joint acceleration for minimum energy, $K.E \approx \sum_{n=1}^{N}(\ddot{q}^T \ddot{q})$

Second objective

- Kinetic energy of the robot, $K.E = \frac{1}{2}\dot{q}^T H \dot{q}$
- Complex nonlinear terms and computation effort
- Joint acceleration for minimum energy, $K.E \approx \sum_{n=1}^{N}(\ddot{q}^T \ddot{q})$

Second objective

- Kinetic energy of the robot, $K.E = \dfrac{1}{2}\dot{q}^T H \dot{q}$
- Complex nonlinear terms and computation effort
- Joint acceleration for minimum energy, $K.E \approx \sum_{n=1}^{N}(\ddot{q}^T \ddot{q})$

Problem formulation

Cost functional $f(p) = w_1 \sum_{n=1}^{N} (\dot{x}_d - \dot{x}_c) + w_2 \sum_{n=1}^{N} (\ddot{q}^T \ddot{q})$

Analytic gradient of the cost function

Numerical gradient by finite-difference

Weight matrix to prioritize the cost function

Inequality constraint on $\ddot{q}$ $\dot{q}$ and q

Equality constraint on desired position

Solve the optimization

Ensure joint torque limits

Results

Inertial parameter estimation

- Friction is absent in robot simulation, $I_F = 0$
- Friction is present in robot hardware, $I_F \neq 0$
- $\tau = H(q)\ddot{q} + c(q, \dot{q}) + \tau_g(q) + \mathbf{I_F}$

Parameter (unit)		Actual value	Estimated value
mass (kg)		5	**5.15**
center of mass (m)	**x**	0.04	0.0527
	y	0.04	0.0542
	z	0.0	-0.0004

Table: Robot simulation

Parameter (unit)		Actual value	Estimated value
mass (kg)		4	**23.7**
center of mass (m)	**x**	0.15	0.1698
	y	0.0	0.3842
	z	0.075	0.00198

Table: Robot hardware

Inertial parameter estimation

- Friction is absent in robot simulation, $l_F = 0$
- Friction is present in robot hardware, $l_F \neq 0$
- $\tau = H(q)\ddot{q} + c(q,\dot{q}) + \tau_g(q) + \mathbf{l_F}$

Parameter (unit)		Actual value	Estimated value
mass (kg)		5	**5.15**
center of mass (m)	**x**	0.04	0.0527
	y	0.04	0.0542
	z	0.0	-0.0004

Table: Robot simulation

Parameter (unit)		Actual value	Estimated value
mass (kg)		4	**23.7**
center of mass (m)	**x**	0.15	0.1698
	y	0.0	0.3842
	z	0.075	0.00198

Table: Robot hardware

Inertial parameter estimation

- Friction is absent in robot simulation, $I_F = 0$
- Friction is present in robot hardware, $I_F \neq 0$
- $\tau = H(q)\ddot{q} + c(q,\dot{q}) + \tau_g(q) + \mathbf{I_F}$

Parameter (unit)		Actual value	Estimated value
mass (kg)		5	**5.15**
center of mass (m)	**x**	0.04	0.0527
	y	0.04	0.0542
	z	0.0	-0.0004

Table: Robot simulation

Parameter (unit)		Actual value	Estimated value
mass (kg)		4	**23.7**
center of mass (m)	**x**	0.15	0.1698
	y	0.0	0.3842
	z	0.075	0.00198

Table: Robot hardware

Inertial parameter estimation

- Friction is absent in robot simulation, $I_F = 0$
- Friction is present in robot hardware, $I_F \neq 0$
- $\tau = H(q)\ddot{q} + c(q,\dot{q}) + \tau_g(q) + \mathbf{I_F}$

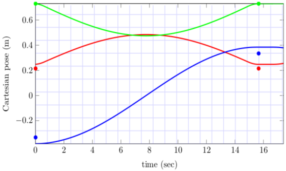
Parameter (unit)		Actual value	Estimated value
mass (kg)		5	**5.15**
center of mass (m)	**x**	0.04	0.0527
	y	0.04	0.0542
	z	0.0	-0.0004

Table: Robot simulation

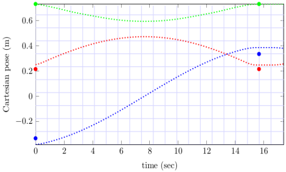
Parameter (unit)		Actual value	Estimated value
mass (kg)		4	**23.7**
center of mass (m)	**x**	0.15	0.1698
	y	0.0	0.3842
	z	0.075	0.00198

Table: Robot hardware

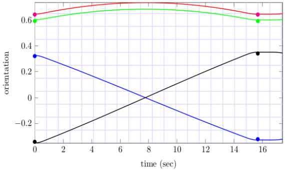
Optimization



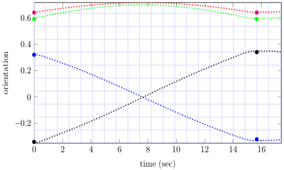
(a) input cartesian pose of the end-effector in x (—), y (—), and z (—).



(b) optimized cartesian pose of the end-effector in x (—), y (—), and z (—).

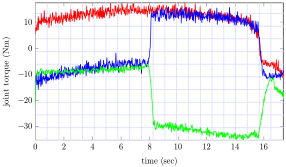


(c) input end-effector orientation qx (—), qy (—), qz (—), and w(—).

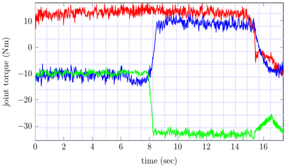


(d) optimized end-effector orientation qx (—), qy (—), qz (—), and w(—).

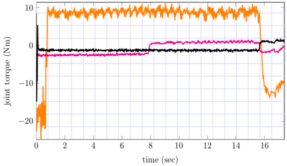
Figure: cartesian pose



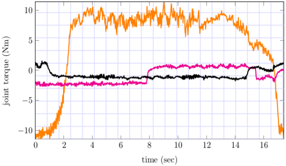
(a) input joint torque for joint 1 (—), joint 2 (—), and joint 3 (—).



(b) optimized joint torque for joint 1 (—), joint 2 (—), and joint 3 (—).



(c) input joint torque for joint 4 (—), joint 5 (—), and joint 6 (—).



(d) optimized joint torque for joint 4 (—), joint 5 (—), and joint 6 (—).

Figure: joint torque

Trajectory before optimization

Trajectory after optimization

Conclusion

Key findings

- Mass of the object is estimated using wrench analysis
- Torque due to friction (I_F) in robot joints affects the estimation
- Desired trajectory is followed with minimum accleration
- Proposed formulation is faster to converge $\sim 300 - 600$ ms

Key findings

- Mass of the object is estimated using wrench analysis
- Torque due to friction (I_F) in robot joints affects the estimation
- Desired trajectory is followed with minimum accleration
- Proposed formulation is faster to converge $\sim 300 - 600$ ms

Key findings

- Mass of the object is estimated using wrench analysis
- Torque due to friction (I_F) in robot joints affects the estimation
- Desired trajectory is followed with minimum accleration
- Proposed formulation is faster to converge $\sim 300 - 600$ ms

Key findings

- Mass of the object is estimated using wrench analysis
- Torque due to friction (I_F) in robot joints affects the estimation
- Desired trajectory is followed with minimum accleration
- Proposed formulation is faster to converge $\sim 300 - 600$ ms

limitations

- Does not directly follow the cartesian position, $x_d - x_c$
- The proposed approach only consider the joint acceleration
- Approximate solution to the given optimization problem
- Requires pre-planned motion to start the optimization

limitations

- Does not directly follow the cartesian position, $x_d - x_c$
- The proposed approach only consider the joint acceleration
- Approximate solution to the given optimization problem
- Requires pre-planned motion to start the optimization

limitations

- Does not directly follow the cartesian position, $x_d - x_c$
- The proposed approach only consider the joint acceleration
- Approximate solution to the given optimization problem
- Requires pre-planned motion to start the optimization

limitations

- Does not directly follow the cartesian position, $x_d - x_c$
- The proposed approach only consider the joint acceleration
- Approximate solution to the given optimization problem
- Requires pre-planned motion to start the optimization

Thank you for your attention!