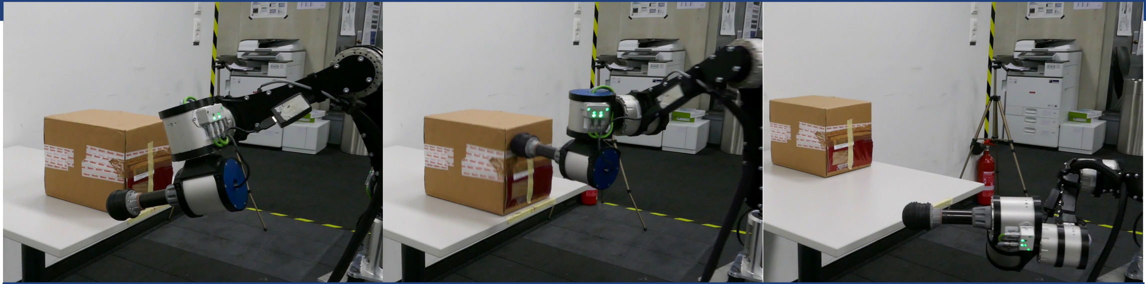


Contact-Implicit Trajectory Optimization for Dynamic Object Manipulation

Jean-Pierre Sleiman, Jan Carius, Ruben Grandia, Martin Wermelinger, Marco Hutter

Robotics Systems Lab, ETH

Introduction

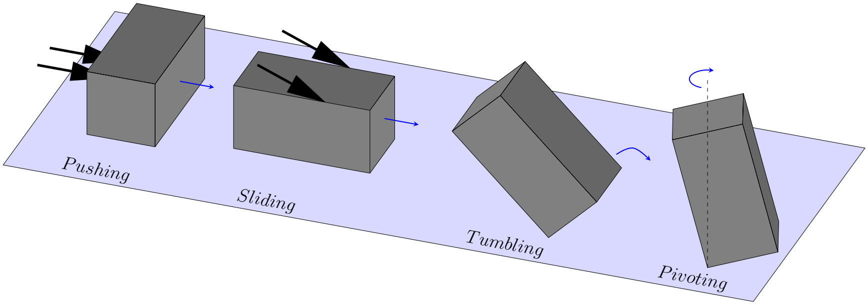
There has been a shift in attention from industrial robotics towards the development of robots that are capable of a more dexterous interaction with their environment.

Locomotion

Dynamic Manipulation

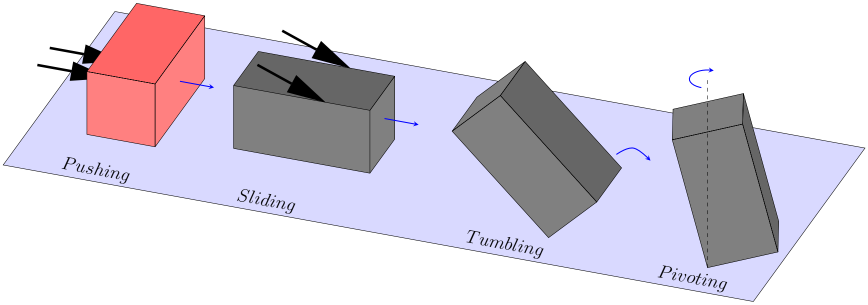
Dynamic Manipulation

Nonprehensile (graspless) manipulation includes phases where the manipulator loses possibility of contact with the object before task completion

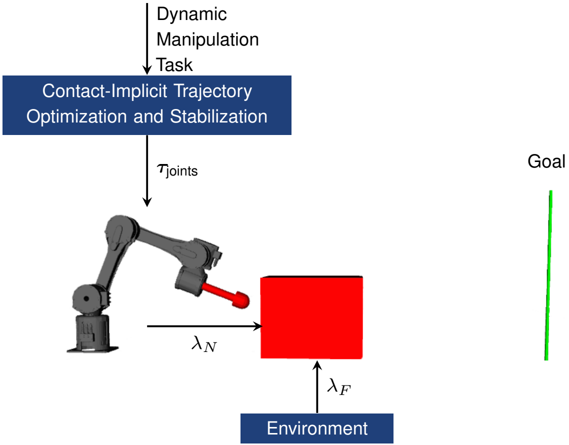


Dynamic Manipulation

Nonprehensile (graspless) manipulation includes phases where the manipulator loses possibility of contact with the object before task completion

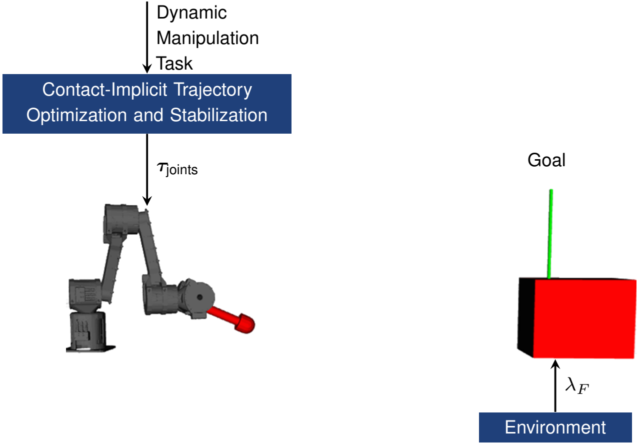


Motivation & Problem Statement



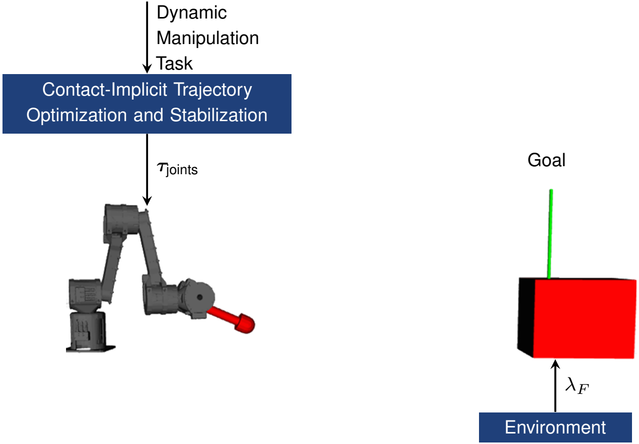
Motivation & Problem Statement

1. Solve the optimal control problem efficiently for hopes of applying it within an MPC framework



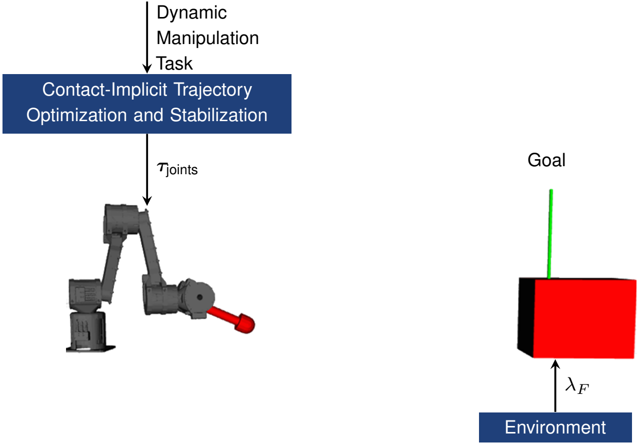
Motivation & Problem Statement

1. Solve the optimal control problem efficiently for hopes of applying it within an MPC framework
2. Ensure dynamic feasibility and physically consistent contact behavior
3. Avoid specifying a hand-crafted contact schedule



Motivation & Problem Statement

1. Solve the optimal control problem efficiently for hopes of applying it within an MPC framework
2. Ensure dynamic feasibility and physically consistent contact behavior
3. Avoid specifying a hand-crafted contact schedule
4. Ensure ease of transferability onto real hardware without any post-optimization modifications



Optimal Control Problem

Trajectory optimization techniques aim to solve the following nonlinear optimal control (NLOC) problem:

$$
\begin{cases}
\min\limits_{\boldsymbol{\tau}(\cdot)} \quad J(\boldsymbol{x}(t), \boldsymbol{\tau}(t), t) = m(\boldsymbol{x}(T)) + \int\limits_0^T L(\boldsymbol{\tau}(t), \boldsymbol{x}(t))\mathrm{d}t \\
\boldsymbol{s.t.} \quad \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{\tau}(t)) \qquad \textbf{(nonlinear, smooth ODEs)} \\
\qquad \boldsymbol{h}_{min} \leq \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{\tau}(t)) \leq \boldsymbol{h}_{max} \qquad \textbf{(inequality constraints)} \\
\qquad \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{\tau}(t)) = 0 \qquad \textbf{(equality constraints)} \\
\forall t \in [0, T]
\end{cases}
$$

Optimal Control Problem

Trajectory optimization techniques aim to solve the following nonlinear optimal control (NLOC) problem:

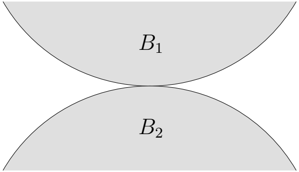
$$\begin{cases} \min\limits_{\boldsymbol{\tau}(\cdot)} \quad J(\boldsymbol{x}(t), \boldsymbol{\tau}(t), t) = m(\boldsymbol{x}(T)) + \int_0^T L(\boldsymbol{\tau}(t), \boldsymbol{x}(t))\mathrm{d}t \\ \boldsymbol{s.t.} \quad \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{\tau}(t)) \\ \qquad \boldsymbol{h}_{min} \leq \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{\tau}(t)) \leq \boldsymbol{h}_{max} \\ \qquad \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{\tau}(t)) = 0 \\ \forall t \in [0, T] \end{cases}$$

How to incorporate the multi-contact behavior into the dynamics?

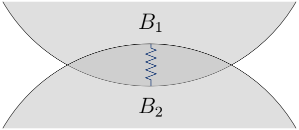
Non-Smooth Contact Dynamics

Modeling

A reliable model describing multi-body contacts has to be chosen before simulating the system dynamics



(a)



(b)

Contact dynamics can be modeled using either **(a)** a hard contact-model or **(b)** a soft contact-model

Optimal Control Problem

Trajectory optimization techniques aim to solve the following nonlinear optimal control (NLOC) problem:

$$\begin{cases} \min\limits_{\boldsymbol{\tau}(\cdot)} \quad J(\boldsymbol{x}(t), \boldsymbol{\tau}(t), t) = m(\boldsymbol{x}(T)) + \int\limits_0^T L(\boldsymbol{\tau}(t), \boldsymbol{x}(t))\mathrm{d}t \\ \boldsymbol{s.t.} \quad \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{\tau}(t)) \\ \qquad \boldsymbol{h}_{min} \leq \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{\tau}(t)) \leq \boldsymbol{h}_{max} \\ \qquad \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{\tau}(t)) = 0 \\ \forall t \in [0, T] \end{cases}$$

→ *How to simulate the hybrid dynamics resulting from a hard contact-model?*

Non-Smooth Contact Dynamics

Simulation

Two prominent schemes used for simulating hybrid dynamical systems are: **event-driven** and **time-stepping** techniques

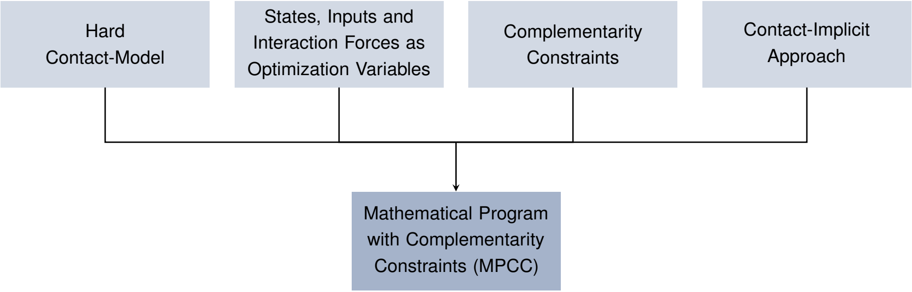
Courtesy of Blender Guru and Phymec

The forward simulation of multi-contact dynamics can be eventually formulated as a so-called **linear complementarity problem (LCP)**

Non-Smooth Contact Dynamics

Trajectory Optimization

Unlike the **multi-phase approach**, **contact-implicit optimization (CIO)** does not require a prespecified contact schedule!



Multiple Shooting

Transcription Process

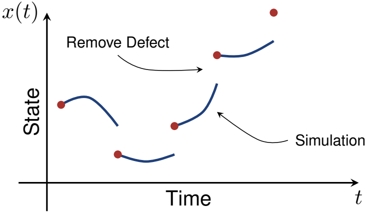
- Integral expressions replaced with Riemann sums, defines the cost function
- System dynamics integrated with a Runge-Kutta scheme, defines the state transition function

↓

Multi-staged Program

$$
\begin{cases}
\min\limits_{z_1,\ldots,z_N} \quad F_N(\boldsymbol{z}_N) + \sum\limits_{k=1}^{N-1} F(\boldsymbol{z}_k) & \\
s.t \quad \boldsymbol{E}_k \boldsymbol{z}_{k+1} = \boldsymbol{c}(\boldsymbol{z}_k) & \forall\ k = 1, \ldots, N-1 \\
\qquad \boldsymbol{S}_1 \boldsymbol{z}_1 = \boldsymbol{z}_{init} & \\
\qquad \boldsymbol{S}_N\ \boldsymbol{z}_N = \boldsymbol{z}_{final} & \\
\qquad \underline{\boldsymbol{z}}_k \leq \boldsymbol{z}_k \leq \bar{\boldsymbol{z}}_k & \forall\ k = 1, \ldots, N \\
\qquad \underline{\boldsymbol{h}}_k \leq \boldsymbol{h}(\boldsymbol{z}_k) \leq \bar{\boldsymbol{h}}_k & \forall\ k = 1, \ldots, N
\end{cases}
$$

Discretize



Optimize ⟶ FORCES PRO

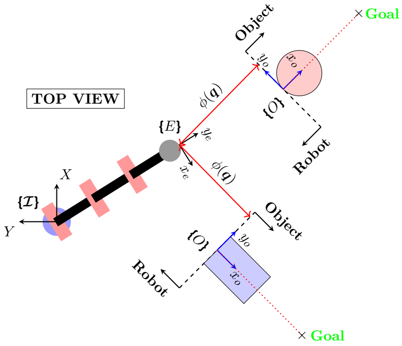
Contact-Implicit Optimization

1. The cost function minimizes energy and penalizes the robot's wrist joint-velocities

$$J(\boldsymbol{z}_k) = \sum_{k=1}^{N} \Delta t \cdot \left(\frac{\boldsymbol{\tau}_k^T \boldsymbol{R} \boldsymbol{\tau}_k}{\tau_{max}^2} + \frac{\dot{\boldsymbol{q}}_{r_k}^T \boldsymbol{Q} \dot{\boldsymbol{q}}_{r_k}}{\dot{q}_{max}^2} \right)$$

2. The manipulation task is specified in terms of final boundary conditions in the optimization program

3. The signed distance function $\phi(\boldsymbol{q})$ is defined such that a negative distance indicates that the end-effector is penetrating the object



Contact-Implicit Optimization

- Normal contact force λ_N, contact impulse Λ_N, and percussion P_N between robot and object
- Frictional force λ_F, frictional impulse Λ_F, and percussion P_F between environment and object
- Variables λ_n and λ_f defined in relation to the percussions: $P_N = \Delta t \cdot \lambda_n$ & $P_F = \Delta t \cdot \lambda_f$

Underactuated Dynamics & Impulse-Momentum Equations

$$\begin{cases} \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{S}^T \boldsymbol{\tau} + \boldsymbol{J}^T(\boldsymbol{q}) \cdot \begin{bmatrix} \lambda_N \\ \lambda_F \end{bmatrix} \\ \boldsymbol{M}(\boldsymbol{q})(\dot{\boldsymbol{q}}^+ - \dot{\boldsymbol{q}}^-) = \boldsymbol{J}^T(\boldsymbol{q}) \cdot \begin{bmatrix} \Lambda_N \\ \Lambda_F \end{bmatrix} \end{cases}$$

$$\Downarrow$$

$$\begin{cases} \dot{\boldsymbol{q}}_{k+1} = \dot{\boldsymbol{q}}_k + \Delta t \cdot \boldsymbol{M}_k^{-1} \left(\boldsymbol{S}^T \boldsymbol{\tau}_k - \boldsymbol{h}_k + \boldsymbol{J}_k^T \cdot \begin{bmatrix} \lambda_{n_k} \\ \lambda_{f_k} \end{bmatrix} \right) \\ \boldsymbol{q}_{k+1} = \boldsymbol{q}_k + \Delta t \cdot \dot{\boldsymbol{q}}_{k+1} \end{cases}$$

Complementarity Constraints

$$0 \leq \phi(\boldsymbol{q}_k) \perp \lambda_{n_k} \geq 0$$

$$\Updownarrow$$

$$\begin{cases} \phi(\boldsymbol{q}_k) \geq 0 & \text{(no penetration)} \\ \lambda_{n_k} \geq 0 & \text{(unilateral contact force)} \\ \lambda_{n_k} \cdot \phi(\boldsymbol{q}_k) = 0 & \text{(no force at a distance)} \end{cases}$$

Contact-Implicit Optimization

- Normal contact force λ_N, contact impulse Λ_N, and percussion P_N between robot and object
- Frictional force λ_F, frictional impulse Λ_F, and percussion P_F between environment and object
- Variables λ_n and λ_f defined in relation to the percussions: $P_N = \Delta t \cdot \lambda_n \;\&\; P_F = \Delta t \cdot \lambda_f$

Underactuated Dynamics & Impulse-Momentum Equations

$$\begin{cases} \boldsymbol{M}(\boldsymbol{q})\ddot{q} + \boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{S}^T\boldsymbol{\tau} + \boldsymbol{J}^T(\boldsymbol{q}) \cdot \begin{bmatrix} \lambda_N \\ \lambda_F \end{bmatrix} \\ \boldsymbol{M}(\boldsymbol{q})(\dot{\boldsymbol{q}}^+ - \dot{\boldsymbol{q}}^-) = \boldsymbol{J}^T(\boldsymbol{q}) \cdot \begin{bmatrix} \Lambda_N \\ \Lambda_F \end{bmatrix} \end{cases}$$

$$\Downarrow$$

$$\begin{cases} \dot{\boldsymbol{q}}_{k+1} = \dot{\boldsymbol{q}}_k + \Delta t \cdot \boldsymbol{M}_k^{-1} \left(\boldsymbol{S}^T \boldsymbol{\tau}_k - \boldsymbol{h}_k + \boldsymbol{J}_k^T \cdot \begin{bmatrix} \lambda_{n_k} \\ \lambda_{f_k} \end{bmatrix} \right) \\ \boldsymbol{q}_{k+1} = \boldsymbol{q}_k + \Delta t \cdot \dot{\boldsymbol{q}}_{k+1} \end{cases}$$

Complementarity Constraints

$$0 \leq \phi(\boldsymbol{q}_k) \perp \lambda_{n_k} \geq 0$$

$$\Updownarrow$$

$$\begin{cases} \phi(\boldsymbol{q}_k) \geq 0 & \text{(no penetration)} \\ \lambda_{n_k} \geq 0 & \text{(unilateral contact force)} \\ \lambda_{n_k} \cdot \phi(\boldsymbol{q}_k) = 0 & \text{(no force at a distance)} \end{cases}$$

Contact-Implicit Optimization

- Normal contact force λ_N, contact impulse Λ_N, and percussion P_N between robot and object
- Frictional force λ_F, frictional impulse Λ_F, and percussion P_F between environment and object
- Variables λ_n and λ_f defined in relation to the percussions: $P_N = \Delta t \cdot \lambda_n$ & $P_F = \Delta t \cdot \lambda_f$

Underactuated Dynamics & Impulse-Momentum Equations

$$\begin{cases} \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{S}^T \boldsymbol{\tau} + \boldsymbol{J}^T(\boldsymbol{q}) \cdot \begin{bmatrix} \lambda_N \\ \lambda_F \end{bmatrix} \\ \boldsymbol{M}(\boldsymbol{q})(\dot{\boldsymbol{q}}^+ - \dot{\boldsymbol{q}}^-) = \boldsymbol{J}^T(\boldsymbol{q}) \cdot \begin{bmatrix} \Lambda_N \\ \Lambda_F \end{bmatrix} \end{cases}$$

$$\Downarrow$$

$$\begin{cases} \dot{\boldsymbol{q}}_{k+1} = \dot{\boldsymbol{q}}_k + \Delta t \cdot \boldsymbol{M}_k^{-1} \left(\boldsymbol{S}^T \boldsymbol{\tau}_k - \boldsymbol{h}_k + \boldsymbol{J}_k^T \cdot \begin{bmatrix} \lambda_{n_k} \\ \lambda_{f_k} \end{bmatrix} \right) \\ \boldsymbol{q}_{k+1} = \boldsymbol{q}_k + \Delta t \cdot \dot{\boldsymbol{q}}_{k+1} \end{cases}$$

Complementarity Constraints

$$0 \leq \phi(\boldsymbol{q}_k) \perp \lambda_{n_k} \geq 0$$

$$\Updownarrow$$

$$\begin{cases} \phi(\boldsymbol{q}_k) \geq 0 & \text{(no penetration)} \\ \lambda_{n_k} \geq 0 & \text{(unilateral contact force)} \\ \lambda_{n_k} \cdot \phi(\boldsymbol{q}_k) = 0 & \text{(no force at a distance)} \end{cases}$$

Contact-Implicit Optimization

- Positive scalar quantity v_d signifying the speed along the desired direction of motion
- Pre- and post-impact relative separation velocities γ^- / γ^+
- Newton's restitution coefficient ϵ

Newton's Restitution Law of Impacts

$$\phi(\boldsymbol{q}) = 0: \qquad 0 \leq P_N \perp \left(\gamma^+ + \epsilon\gamma^-\right) \geq 0$$

$$\Downarrow$$

$$P_{N_k} \cdot (\gamma_{k+1} + \epsilon\gamma_k) = 0$$

$$\Delta t \cdot \lambda_{n_k} \cdot \left(\dot{\phi}_{k+1} + \epsilon\dot{\phi}_k\right) = 0$$

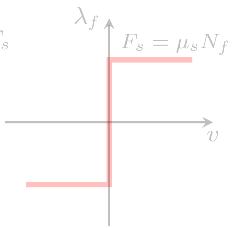
$$\Delta t \cdot \lambda_{n_k} \cdot (\boldsymbol{J}_N(\boldsymbol{q}_{k+1})\dot{\boldsymbol{q}}_{k+1} + \epsilon\boldsymbol{J}_N(\boldsymbol{q}_k)\dot{\boldsymbol{q}}_k) = 0$$

Coulomb Friction Model

$$\begin{cases} \lambda_f = \lambda_n & \text{if } v_d = 0 \;\&\; \lambda_n < F_s \\ \lambda_f = F_s & \text{if } v_d > 0 \end{cases}$$

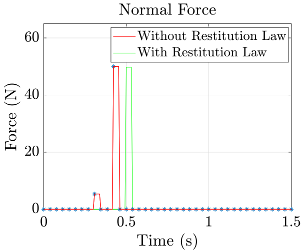
$$\Uparrow$$

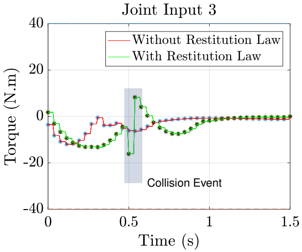
$$\begin{cases} 0 \leq v_{dir} \perp (F_s - F_f) \geq 0 \\ (F_s - F_f)(\lambda - F_f) = 0 \end{cases}$$



Without Newton’s Restitution Law

With Newton’s Restitution Law





Contact-Implicit Optimization

- Positive scalar quantity v_d signifying the speed along the desired direction of motion
- Pre- and post-impact relative separation velocities γ^- / γ^+
- Newton's restitution coefficient ϵ

Newton's Restitution Law of Impacts

$$\phi(\boldsymbol{q}) = 0 : \qquad 0 \leq P_N \perp \left(\gamma^+ + \epsilon\gamma^-\right) \geq 0$$

$$\Downarrow$$

$$P_{N_k} \cdot (\gamma_{k+1} + \epsilon\gamma_k) = 0$$

$$\Delta t \cdot \lambda_{n_k} \cdot \left(\dot{\phi}_{k+1} + \epsilon\dot{\phi}_k\right) = 0$$

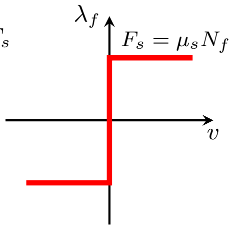
$$\Delta t \cdot \lambda_{n_k} \cdot (\boldsymbol{J}_N(\boldsymbol{q}_{k+1})\dot{\boldsymbol{q}}_{k+1} + \epsilon\boldsymbol{J}_N(\boldsymbol{q}_k)\dot{\boldsymbol{q}}_k) = 0$$

Coulomb Friction

$$\begin{cases} \lambda_f = \lambda_n & \text{if } v_d = 0 \;\&\; \lambda_n < F_s \\ \lambda_f = F_s & \text{if } v_d > 0 \end{cases}$$

$$\Uparrow$$

$$\begin{cases} 0 \leq v_{dir} \perp (F_s - F_f) \geq 0 \\ (F_s - F_f)(\lambda - F_f) = 0 \end{cases}$$

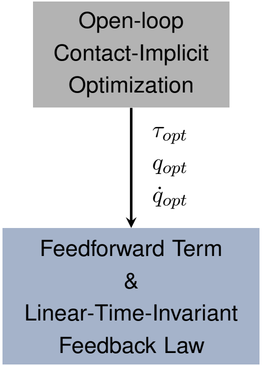


Contact-Implicit Optimization

The optimal input sequence τ_{opt_k} is an open-loop control policy. Therefore, **trajectory stabilization** is still needed.

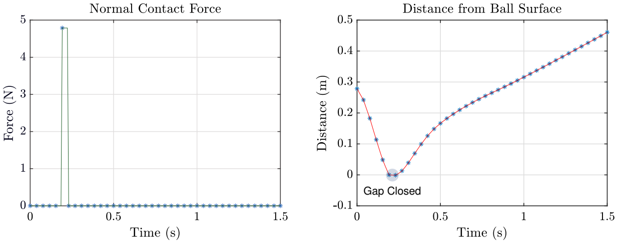
$$\tau_i(t) = \underbrace{\tau_{opt_i}(t)}_{feedforward\ term} + \underbrace{k_{p_i}\left(q_{opt_i}(t) - q_i(t)\right) + k_{d_i}\left(\dot{q}_{opt_i}(t) - \dot{q}_i(t)\right)}_{feedback\ term}$$

where $\tau_{opt}(t)$ is obtained by a zero-order-hold while $q_{opt}(t)$ and $\dot{q}_{opt}(t)$ by linear interpolation



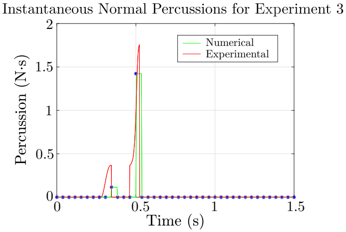
Robot-Door Manipulation Task

Robot-Ball Manipulation Task



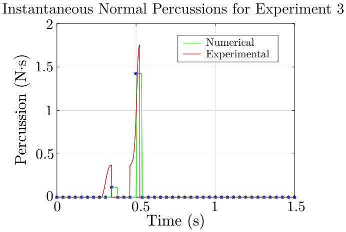
Robot-Block Manipulation Task

- Compatibility between optimal contact force and measured one



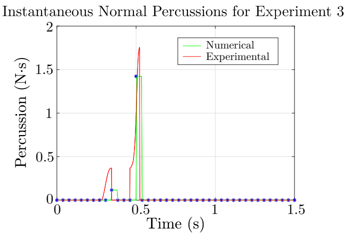
Robot-Block Manipulation Task

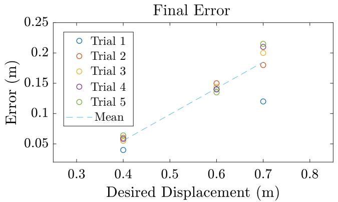
- Compatibility between optimal contact force and measured one
- Convergence rate two orders of magnitude higher than that of previous works



Robot-Block Manipulation Task

- Compatibility between optimal contact force and measured one
- Convergence rate two orders of magnitude higher than that of previous works
- Satisfaction of manipulation task with inaccuracy proportional to desired displacement





Thank you for your attention!

ETH zürich

RSL Robotic Systems Lab

FNSNF
Swiss National Science Foundation

ANYmal RESEARCH

Any Questions?