MICRO-507: Legged Robots

Gait and Trajectory Optimization for Legged Systems Through Phase-Based End-Effector Parameterization

By Alexander W. Winkler, C. Dario Bellicoso, Marco Hutter, and Jonas Buchli

ETH Zurich, Robotic Systems Lab, "ETH Zurich Testing ANYmal C's Advanced Locomotion Capabilities in Unstructured Natural Terrain," ANYbotics, last modified June 3, 2020, https://www.anybotics.com/news/advancing-legged-robotics-research/.

Théo Heng
Aubin Sabatier
Luca Sidoti Pinto

École polytechnique fédérale de Lausanne

26 November 2024

Motivation

Dynamic Models	Advantages	Limitations
Linear Inverted Pendulum (LIP)	- Simplifies CoM dynamics - Fast for predefined tasks	- Restricted base motions - No flight phases or uneven terrain
Full Rigid-Body Dynamics	- High accuracy - Captures vertical and complex motions	- Computationally expensive - Complex to model
Centroidal Dynamics	- Balances simplicity and feasibility - Handles flight phases	- Assumes minimal limb mass deviations - Limited on uneven terrain

➢ Require predefining some aspects of the motion (gait sequence, foothold placement, or step timings)

Contact Schedule Optimization	Advantages	Limitations
Integer Programming	- Optimizes contact schedules - Useful for predefined tasks	- Relies on heuristics - Computationally costly for large variables
Soft-Contact Models	- Models smooth force transitions - Handles contact dynamics	- Stiffness affects convergence - Unsuitable for highly uneven terrain
Linear Complementary Problem (LCP)	- Ensures feasible constraints - Dynamic motion compatibility	- High computational cost - Scaling issues with multi-legged systems

➢ Struggle to balance computational efficiency and adaptability

New Approach

- A **single optimization formulation** that determines:
  - Gait sequence (e.g., walk, trot, jump)
  - Step timings
  - Footholds
  - Body motion in 6 dimensions (position, orientation)
  - Forces at each foot

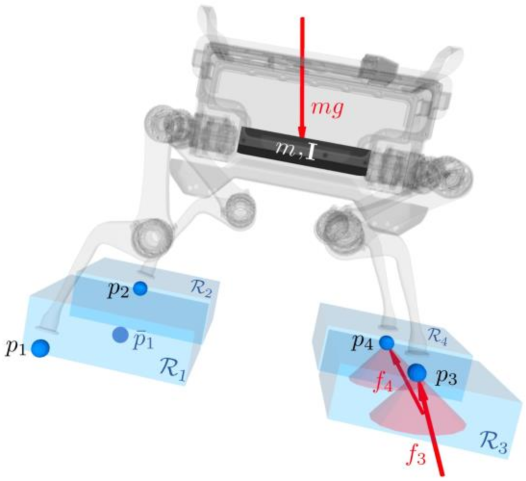
- Requires only the **start & goal position**, nb of steps, and terrain data
- Generates motion plans for arbitrary terrains and robot morphologies
- Maintains computational efficiency for practical online deployment

Demo video


Gait and Trajectory Optimization for Legged Systems through Phase-based Endeffector Parameterization
A. W. Winkler, C. D. Bellicoso, M. Hutter, J. Buchli
Robotics and Automation Letters (RA-L), 2018
A D R L
RSL
Robotic Systems Lab

Methodology

- Optimization Model
  - Nonlinear Programming (NLP)
  - Decision variables: body motion, feet trajectories, and contact forces
  - Constraints: friction cone, kinematic restrictions, and centroidal dynamics



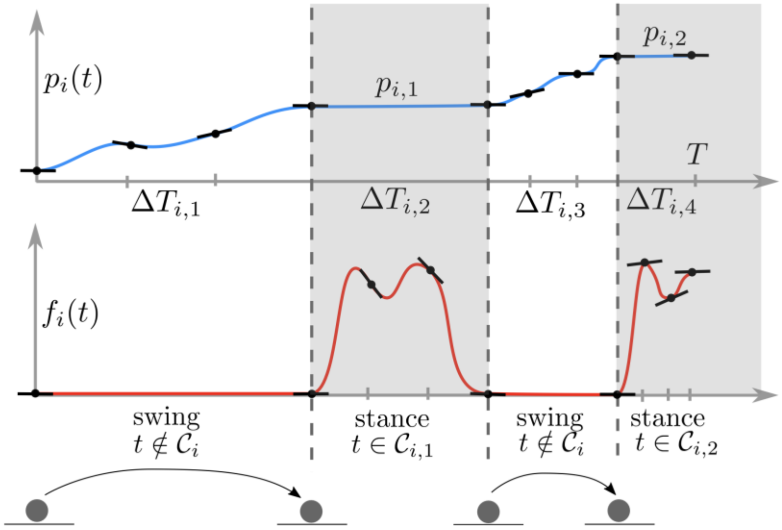
$$
\begin{array}{lll}
\text{find} & \mathbf{r}(t) \in \mathbb{R}^3 & \text{(CoM linear position)} \\
 & \boldsymbol{\theta}(t) \in \mathbb{R}^3 & \text{(base euler angles)} \\
 & \text{for every foot } i: & \\
 & \quad \Delta T_{i,1} \ldots, \Delta T_{i,2n_{s,i}} \in \mathbb{R} & \text{(phase durations)} \\
 & \quad \mathbf{p}_i(t, \Delta T_{i,1}, \ldots) \in \mathbb{R}^3 & \text{(foot position)} \\
 & \quad \mathbf{f}_i(t, \Delta T_{i,1}, \ldots) \in \mathbb{R}^3 & \text{(force at foot)} \\
\text{s.t.} & [\mathbf{r}, \boldsymbol{\theta}](t{=}0) = [\mathbf{r}_0, \boldsymbol{\theta}_0] & \text{(initial state)} \\
 & \mathbf{r}(t{=}T) = \mathbf{r}_g & \text{(desired goal)} \\
 & [\ddot{\mathbf{r}}, \dot{\boldsymbol{\omega}}]^T = \mathbf{f}_d(\mathbf{r}, \mathbf{p}_1, \ldots, \mathbf{f}_1, \ldots) & \text{(dynamic model)} \\
 & \text{for every foot } i: & \\
 & \quad \mathbf{p}_i(t) \in \mathcal{R}_i(\mathbf{r}, \boldsymbol{\theta}), & \text{(kinematic model)} \\
 & \quad \text{if foot } i \text{ in contact}: & \\
 & \quad\quad \dot{\mathbf{p}}_i(t \in \mathcal{C}_i) = \mathbf{0} & \text{(no slip)} \\
 & \quad\quad p_i^z(t \in \mathcal{C}_i) = h_{terrain}(\mathbf{p}_i^{xy}) & \text{(terrain height)} \\
 & \quad\quad \mathbf{f}_i(t \in \mathcal{C}_i) \cdot \mathbf{n}(\mathbf{p}_i^{xy}) \geq 0 & \text{(pushing force)} \\
 & \quad\quad \mathbf{f}_i(t \in \mathcal{C}_i) \in \mathcal{F}(\mu, \mathbf{n}, \mathbf{p}_i^{xy}) & \text{(friction cone)} \\
 & \quad \text{if foot } i \text{ in air}: & \\
 & \quad\quad \mathbf{f}_i(t \notin \mathcal{C}_i) = \mathbf{0} & \text{(no force in air)} \\
 & \quad \sum_{j=1}^{2n_{s,i}} \Delta T_{i,j} = T & \text{(total duration)}
\end{array}
$$

Phase-Based Parameterization

Challenge: Traditional methods require predefined gaits and foot sequences alternating between **Swing** phase and **Stance** phase

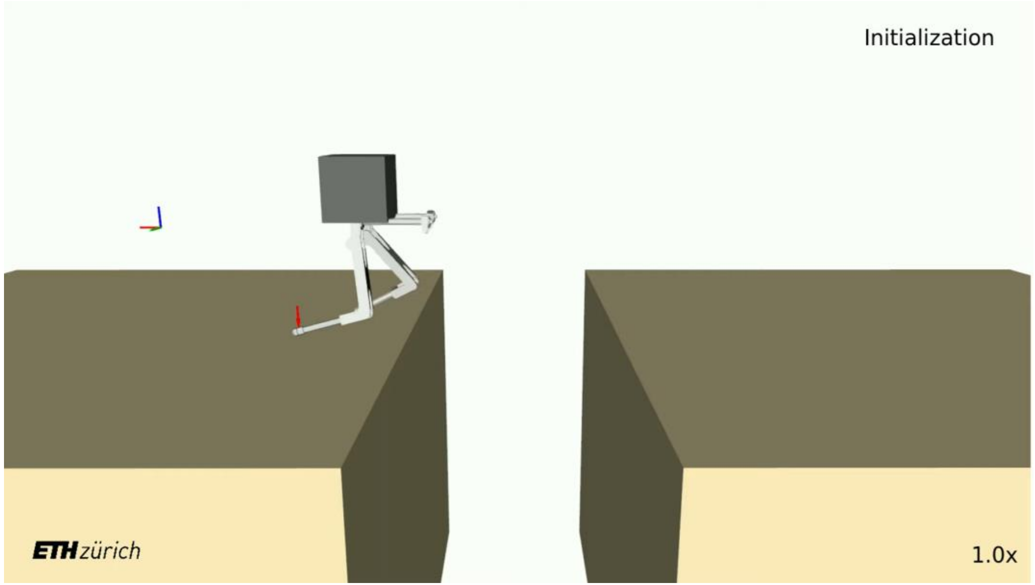
Solution:

- These phases are represented using **polynomials**
- Position and forces during each phase are described by cubic functions
- The algorithm optimizes the phases durations (ΔTi,j) and sequence to generate arbitrary gaits



- ➢ Automatically adapts the gait to the terrain:
  - Inserts flight phases when needed (e.g., for jumping gaps)
  - Adjusts timing for rough terrain or steep inclines

Biped Gap Crossing



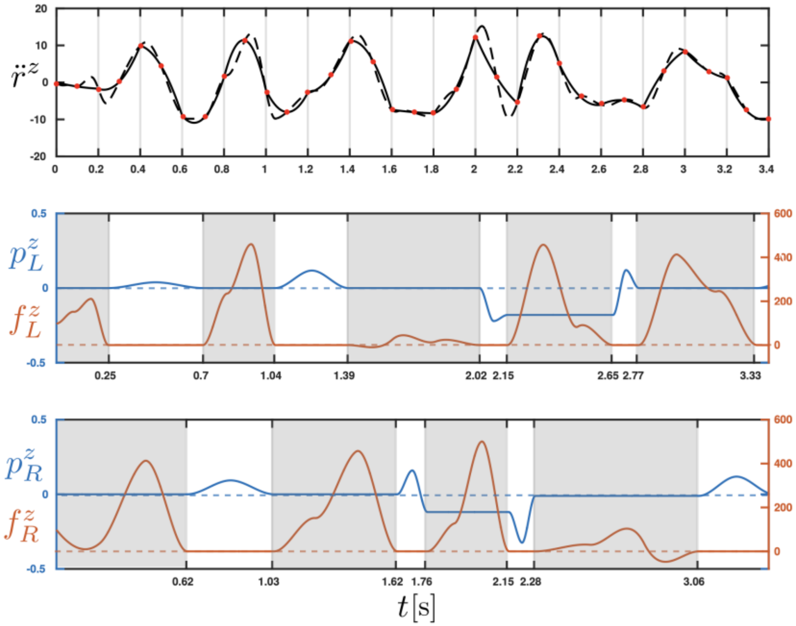


Figure: A generated motion plan for bipedal robot crossing a wide gap

Limitations

- Trade-offs between speed and solver efficiency

- **Simplified Dynamics Model**
  - Model assumes light, immobile limbs
  - May lead to inaccuracies for heavy or highly dynamic limbs
- **Terrain Constraint Gaps**
  - Only applied at specific points along the trajectory
  - May lead to collisions on irregular or complex surfaces
- **Local Minima in Optimization**
  - The solver can get stuck in suboptimal solutions
  - Can limit motion quality

- **Possible improvements**
  - Refining the dynamics model
  - Increasing the frequency of terrain constraint checks
  - Fixing step timings to reduce the problem’s complexity for the solver

Exam Questions

Question 1: What are the main advantages of using a centroidal dynamics model in this work compared to other dynamic models like the Linear Inverted Pendulum (LIP)?

Answer 1:

The centroidal dynamics model offers a middle ground between simplicity and complexity. Unlike the LIP model, which restricts vertical movement and assumes fixed contact points, the centroidal dynamics model:

- Accounts for vertical movements and changes in body orientation, enabling a wider range of dynamic motions.
- Explicitly models individual contact forces, which allows enforcement of friction cone constraints and ensures physically feasible motion.
- Supports the generation of motions with full-flight phases, essential for highly dynamic tasks such as jumping.

Question 2: How does the optimization framework handle transitions between contact and flight phases?

Answer 2:

The framework uses a continuous parameterization of foot forces and positions, where forces are automatically set to zero during flight phases and non-zero during contact. This approach avoids the complexity of integer programming and enables seamless transitions between contact and flight.