Imperial College London

HiPEDS

Towards a Framework for Nonlinear Predictive Control using Derivative-Free Optimization

Ian McInerney[a], Lucian Nita[a], Yuanbo Nie[b], Alberto Oliveri[c], Eric C. Kerrigan[a,b]

[a]Department of Electrical and Electronic Engineering, Imperial College London
[b]Department of Aeronautics, Imperial College London
[c]Department of Electrical, Electronic, Telecommunications Engineering and Naval Architecture, University of Genoa

Presented at the 7th IFAC Conference on Nonlinear Predictive Control

Derivative-Free Framework Overview

- Structured search using the Mesh Adaptive Direct Search (MADS) algorithm
- Linear complexity growth with size of search space
- Augment dynamics with integral cost and constraint violation as states
- Single-shooting simulation of dynamics for each candidate input trajectory

Outline

Nonlinear Model Predictive Control

$$
\begin{aligned}
\min_{x,u,t_f} \quad & \Phi(x(t_f), u(t_f), t_f) + \int_{t_0}^{t_f} L(x(t), u(t), t)dt \\
\text{s.t.} \quad & f(x(t), \dot{x}(t), u(t), t) = 0, \quad \forall t \in [t_0, t_f] \\
& g(x(t), u(t), t) \leq 0 \\
& h(x(0), u(0), t_0, x(t_f), u(t_f), t_f) = 0
\end{aligned}
$$

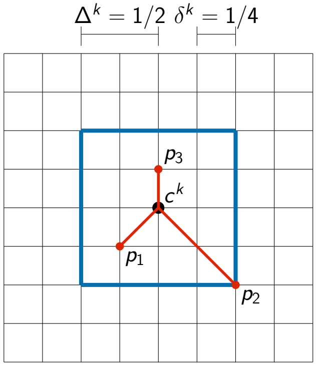
- Continuous-time problem formulation
- Optimize over x, u and t_f
- Continuous-time dynamics $f(\cdot)$
- Arbitrary path constraints $g(\cdot)$
- Boundary constraints $h(\cdot)$

Why Derivative-Free Optimization?

- First/Second-order methods require derivatives of dynamics+cost+constraints
- Computing derivatives for complex non-linear models/black-box models can be difficult/inaccurate
- Derivative-free optimization only requires simulation models

MADS - Overview

- Define a cost function $\mathcal{F}$
- Choose $n + 1$ poll points on the *mesh* in a *frame* surrounding c^k to test
- Choose c_{k+1} as poll point with lowest cost
- Distance between mesh points δ shrinks if no lower cost is found
- Can evaluate all poll points in parallel



MADS - Progressive Barrier Constraints

- Relax constraints and measure their violation
- Keep both a feasible and infeasible iterate and poll at both

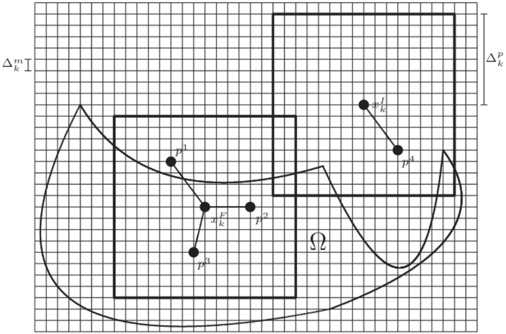


Image from C. Audet and J. E. Dennis, 'A Progressive Barrier for Derivative-Free Nonlinear Programming', SIAM Journal on Optimization, vol. 20, no. 1, pp. 445–472, Jan. 2009.

Framework Overview

- Search space is the input trajectories u and t_f
- Augment dynamics with states for the Lagrangian term and violation measure for the path constraints
- Single-shooting simulation of the augmented dynamics at each poll point
- Use progressive-barrier constraints on the path constraint violation measure

Augmented Dynamics

Cost

Introduce state for Lagrange cost with dynamics

$$\dot{l}(t) = L(x(t), u(t), t)$$

Path constraints

Use $L1$ penalty

$$v_i = \int_0^{t_f} \max\{0, g_i(x(t), u(t), t)\} dt.$$

Introduce state for violation measurement with dynamics

$$\dot{v}(t) = g^+(x(t), u(t), t)$$

Overall Blackbox Function

Let: c be the point in the search space being evaluated

1: Construct the input trajectory u from c

2: Simulate the augmented dynamics using an appropriate solver for the differential equations

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} f(x(t), \dot{x}(t), u(t), t) \\ L(x(t), u(t), t) - \dot{l}(t) \\ g^{+}(x(t), u(t), t) - \dot{v}(t) \end{bmatrix}$$

3: Compute the violation of the boundary conditions

$$v_b = \sum_i \rho_{b_i} |h_i(x(0), u(0), t_0, x(t_f), u(t_f), t_f)|$$

4: Compute the overall constraint violation

$$\mathcal{H} \leftarrow v_b^2 + \sum_i \rho_i (v_i(t_f))^2$$

5: Compute the cost function value

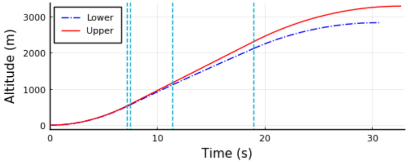
$$\mathcal{F} \leftarrow \Phi(x(t_f), u(t_f), t_f) + l(t_f)$$

Numerical Example - Setup

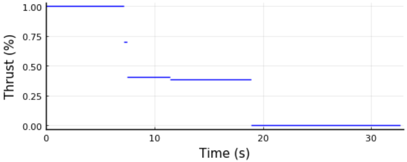
- Make a rocket reach an apogee of 10,000 feet (3,048 meters)
- Rocket's drag coefficient and specific impulse are uncertain in bounded ranges
- By system monotonicity, simulate trajectories with both lower bounds and both upper bounds
- Piecewise-constant input thrust trajectory with variable switching times
- Path constraint to limit velocity to ≤ 150 m/s
- Use the non-differentiable cost function

$$\min_{T,\sigma} \max_{x_l, x_u} \{|h_u(t_{f_u}) - 3048|, |3048 - h_l(t_{f_l})|\}$$

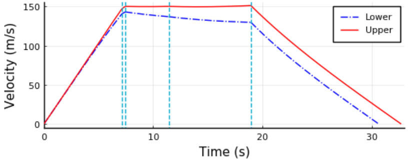
Numerical Example - Results



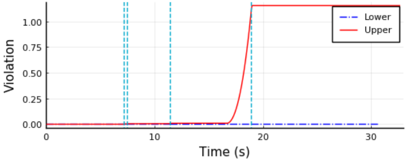
(a) Altitude profile



(c) Input trajectory



(b) Velocity profile



(d) Constraint violation

Discussion

Augmented dynamics formulation

- Easy way to include constraints in the formulation
- No need for separate mesh refinement/cost integration step
- Possibly degrade dynamics simulation performance

Imperial College London

HiPEDS

Conclusions

Key contributions

- Preliminary framework using the Mesh Adaptive Direct Search derivative-free solver
- Definition of augmented dynamics to measure constraint violations

Future directions

- Investigate other input trajectory representations
- Integrate multiple-shooting into the formulation
- Improve warm-starting performance of MADS for closed-loop application
- Investigate parallelizations inside the blackbox function