HARVARD
John A. Paulson
School of Engineering
and Applied Sciences

Constrained Unscented Dynamic Programming

Brian Plancher, Zac Manchester and Scott Kuindersma
Harvard Agile Robotics Lab

The Big Picture:

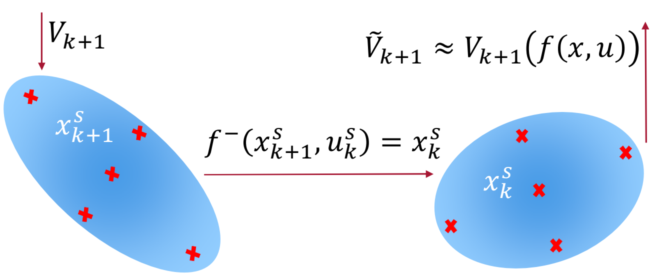
Differential Dynamic Programming (DDP) has become a popular approach to performing trajectory optimization for complex, underactuated robots. However, DDP presents two practical challenges. First, the evaluation of dynamics derivatives during optimization creates a computational bottleneck, particularly in the computation of second derivatives of the system's dynamics. Second, constraints on the states (e.g., boundary conditions, collision constraints, etc.) require additional care since the state trajectory is implicitly defined from the inputs and dynamics. We addresses both problems by building on recent work on Unscented Dynamic Programming (UDP)---which eliminates dynamics derivative computations in DDP---to support general nonlinear state and input constraints using an augmented Lagrangian. The resulting algorithm has the same computational cost as first-order penalty-based DDP variants, but can achieve high-accuracy constraint satisfaction without the numerical ill-conditioning associated with penalty methods.

UDP:

DDP is an iterative algorithm that begins with a guess for an optimal trajectory and improves it by successively minimizing a quadratic approximation of the cost-to-go.

$$V_k^*(x) = \min_u l(x,u) + V_{k+1}\big(f(x,u)\big)$$

UDP takes the classic DDP algorithm and replaces the gradient and Hessian calculations with the **unscented transform**: an approximation computed from a set of sample points.



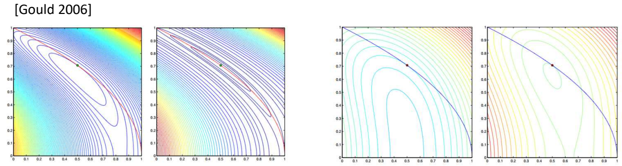
[CDC 2016]

Augmented Lagrangian Constraints:

A natural approach to approximately enforcing constraints is to apply a penalty for constraint violations in the cost function. One popular constraint function is the quadratic penalty method, which suffers from a small gradient near the optimum leading to numerical instability. To overcome the numerical issues associated with penalty methods, augmented Lagrangian solvers add a term that estimates the Lagrange multipliers associated with the constraints.

$$V_k^*(x) = \min_u \begin{matrix} l(x,u) + V_{k+1}\big(f(x,u)\big) + \\ \mu\|\phi(x,u)\|^2 + \lambda^T\phi(x,u) \end{matrix}$$

[Gould 2006]

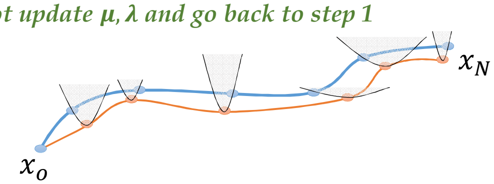


Quadratic Penalty   Augmented Lagrangian

The CUDP Algorithm:

CUDP proceeds in the same manner as DDP but adds in the unscented transform from UDP and augmented Lagrangian constraints (and their associated outer loop update procedure) as follows:

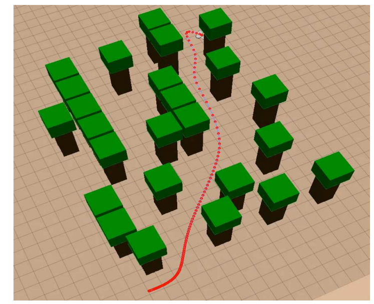
1. Compute the cost-to-go *(including constraint costs)* and the associated optimal feedback control update to the controls backward in time *using the unscented transform*
2. Simulate the system forward in time to create a new nominal trajectory
3. Repeat this process until convergence
4. *At convergence test for constraint satisfaction and if not update μ, λ and go back to step 1*



B. Plancher, Z. Manchester, and S. Kuindersma, "Constrained Unscented Dynamic Programming," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017.

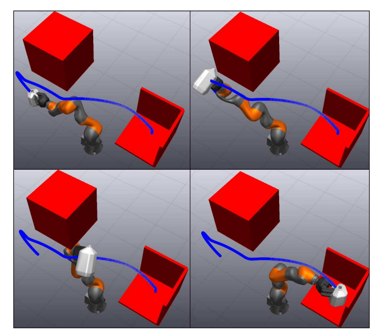
Performance Results:

We compared the use of the unscented transform and augmented Lagrangian against a standard quadratic penalty method and standard first order approximation of DDP known as iLQR in a series of constrained tasks.



Constraints
- Torque Limits on motors
- No-contact constraints with trees
- Final state position and velocity constraint

	Φ < 1e-2	Φ < 1e-4	Φ < 1e-6
Penalty iLQR	✓	✗	✗
Penalty UDP	✓	✓	✗
AL iLQR	✓	✗	✗
AL UDP (CUDP)	✓	✓	✓



Constraints
- Torque Limits on motors
- No-contact constraints with block and shelf
- Final state position and velocity constraint

	Φ < 5e-1	Φ < 1e-2	Φ < 5e-3
Penalty iLQR	✓	✗	✗
Penalty UDP	✓	✗	✗
AL iLQR	✓	✓	✗
AL UDP (CUDP)	✓	✓	✓