High Confidence Reconfigurable Distributed Control

Mani Chandy **John Doyle** **John Hauser** **Richard Murray**

Ali Jadbabie Mark Milam Rick Hindman Bill Dunbar

SEC PI Meeting
12 June 2000

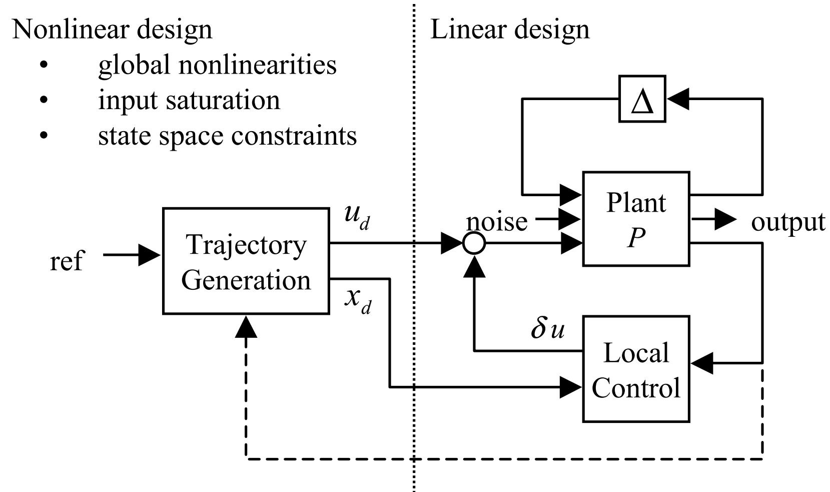
Outline

I. **Brief overview and recent results**
- Theory: optimization based control (MPC + CLF)
- Analyisis: exploration of trajectory space through operability maps
- Computation: real-time, flatness-based trajectory generation

II. **Short term next steps (6-12 months)**

III. **Long term plan (1-3 years)**

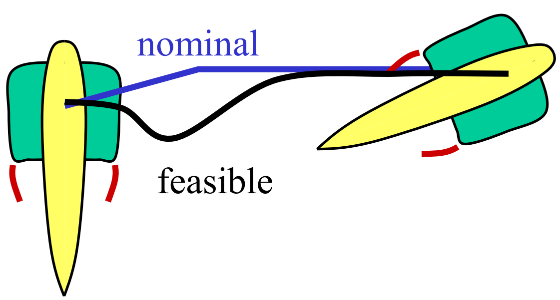
Online Control Customization: Model-Based Trajectory Generation and Tracking



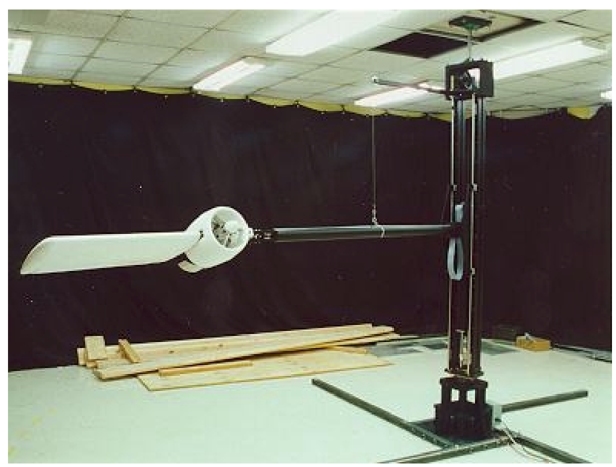
Approach: Two Degree of Freedom Design

- Use online trajectory generation to construct feasible trajectories
- Use model predictive control for local performance
- Exploit dynamics through flatness, operating characteristics, mechanics

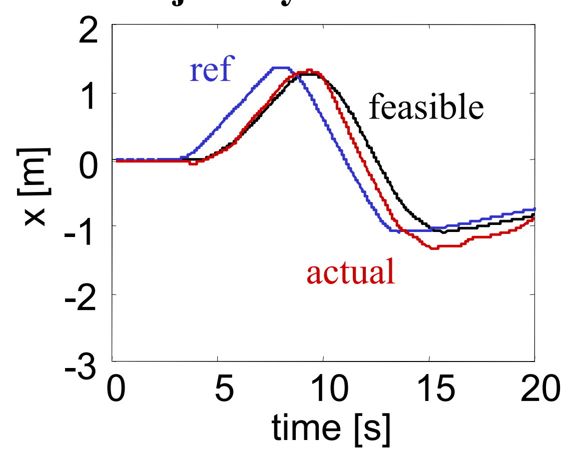
Rapid Transition from Hover to Forward Flight



Caltech Ducted Fan



Real-Time Trajectory Generation



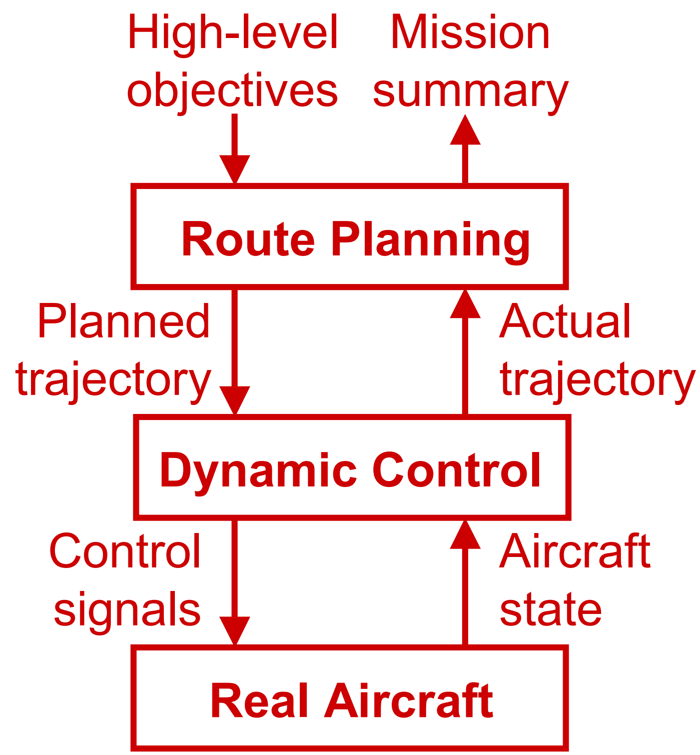
Theory: MPC + CLF Approach

Basic Idea

- Use online models to compute receding horizon optimal control
- Use CLF-based terminal *cost* to give stability proofs rather than terminal *constraints*
- Allows rapid reconfiguration for changing mission objectives, aircraft condition, etc

Recent Results

- Replace optimal terminal cost condition with incremental terminal cost condition
- Allows suboptimal iteration better suited for online use
- Application: simplified ducted fan with lift/drag characteristics (ignores pitch dynamics)



Jadbabie and Hauser, CDC 00

Analysis: Trajectory Exploration for Simplified Dfan

Operability Maps

- State dependent set of achievable accelerations, given limits on thrust and angle of attack
- Example (right):

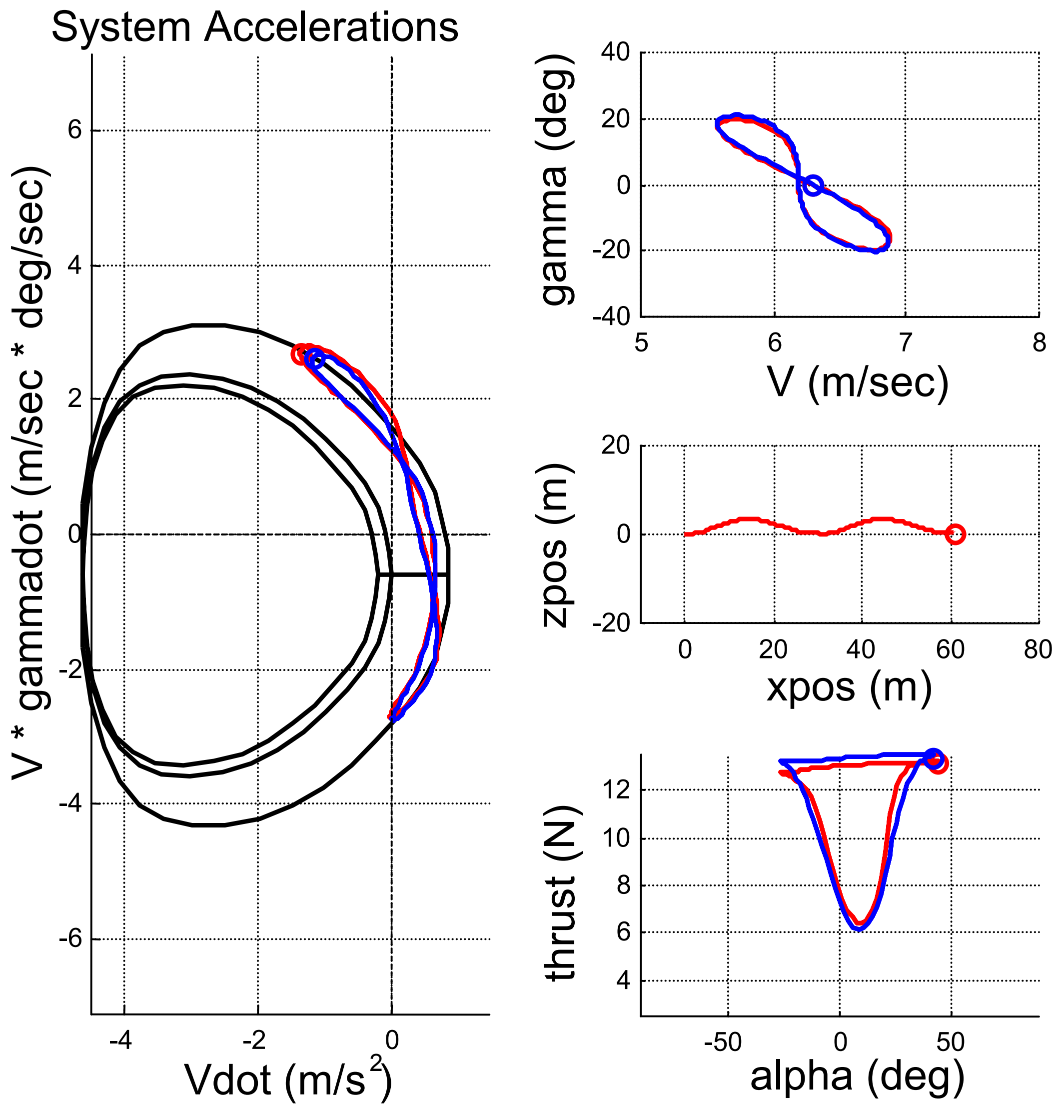
  2.5N < thrust < 13.5N

  “reasonable” alpha

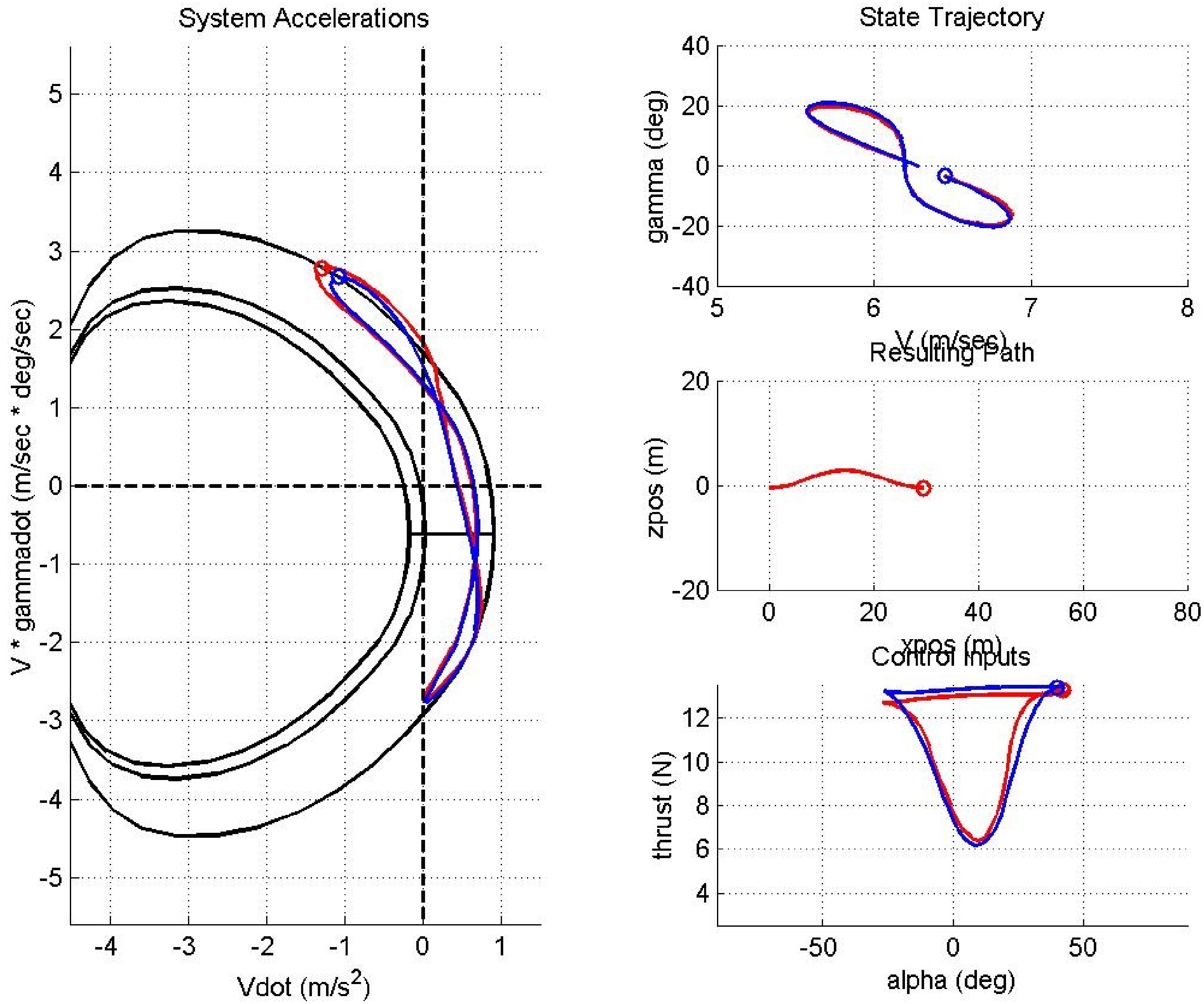
Trajectory Exploration

- Use simplified model to explore trajectory with the help of state-dependent operability maps
- Use results trajectory as input to MPC algorithm

Hindman and Hauser, 2000



Example: Op Maps + MPC on Ducted Fan Model



Computation: Real-Time Trajectory Generation

$$\dot{x} = f(x,u)$$

Collocation

$$(x,u) = \quad \alpha_i \phi^i(t)$$

$$\dot{x}(t_i) = f(x(t_i),u(t_i))$$

Flatness

$$z = z(x,u,\dot{u},\ldots,u^{(p)})$$

$$x = x(z,\dot{z},\ldots,z^{(q)})$$

$$u = u(z,\dot{z},\ldots,z^{(q)})$$

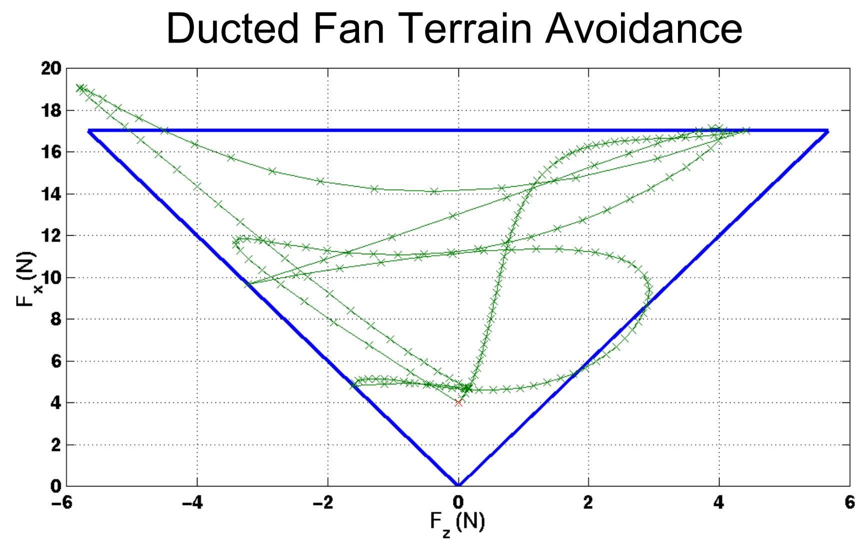
$$z = \quad \alpha_i \phi^i(t)$$

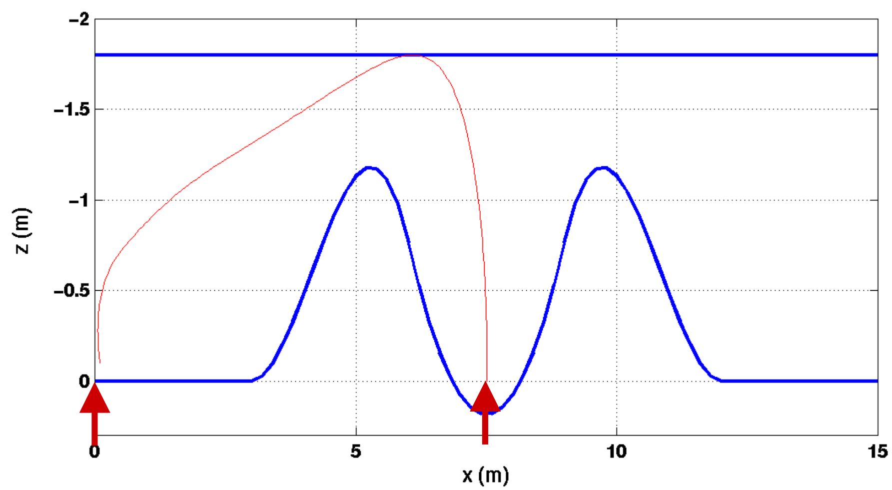
Quasi-collocation

$$y = h(x)$$

$$(x,u) = \Gamma(y,\dot{y},\ldots,y^{(q)})$$

$$0 = \Phi(y,\dot{y},\ldots,y^{(p)})$$





Nonlinear Trajectory Generation (NTG) Package

Advantages

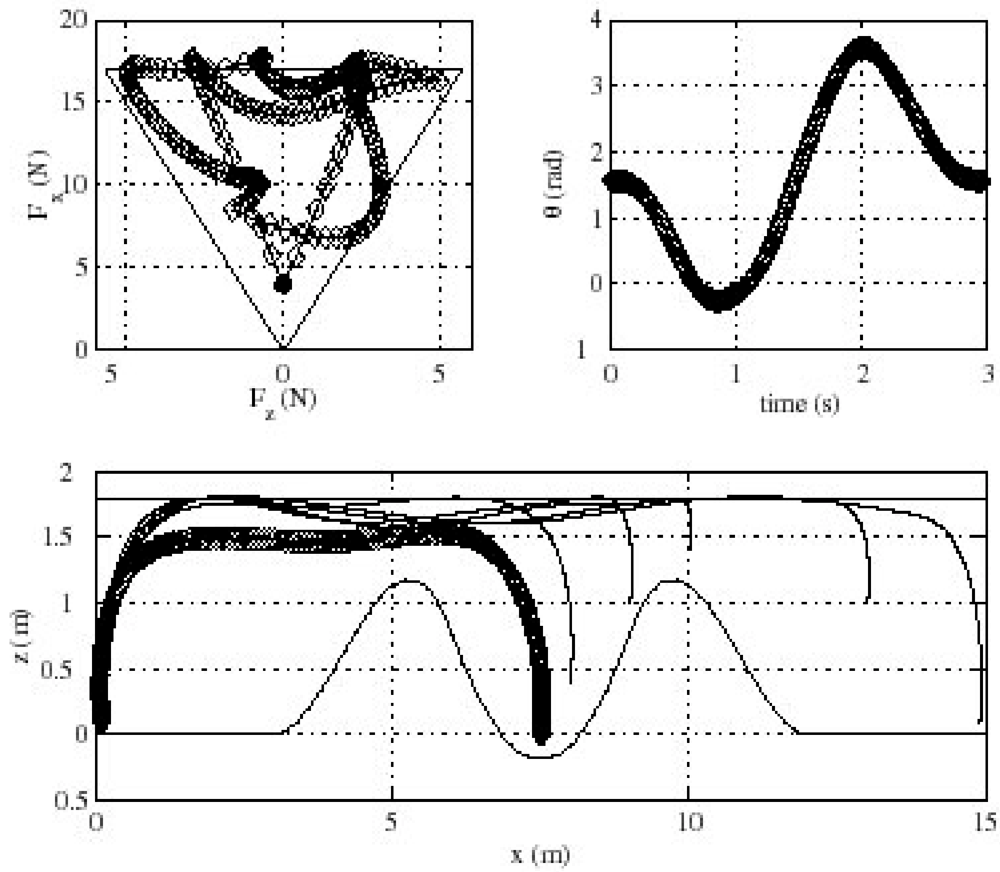
- Handles constraints
- *Very* fast (real-time), especially from warm start
- Good convergence

Disadvantages

- No convergence proofs
- Misses constraints between collocation points
- Doesn't exploit mechanical structure (except through flatness)

Milam, Mushambi & M
CDC 00

Planar Ducted Fan: Warm Starts



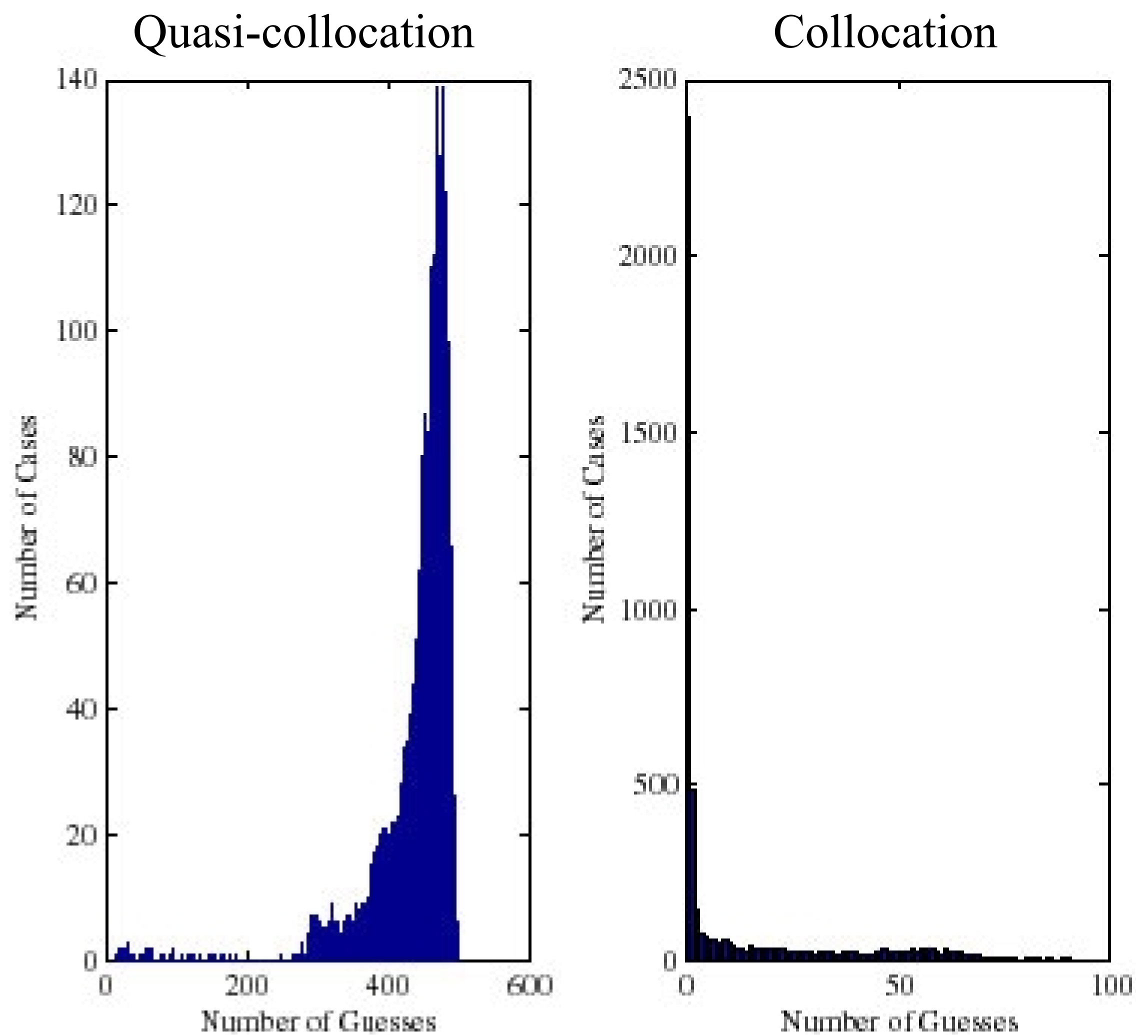
Results: Robust Convergence Properties

Numerical Studies

- 6461 test cases
- 500 initial guess for spline coefficients
- Total of > 3M runs

- Count # of cases that converge for given # of initial guesses

- Comparison between quasi-collocation (x, y, th) and full collocation (states and inputs)



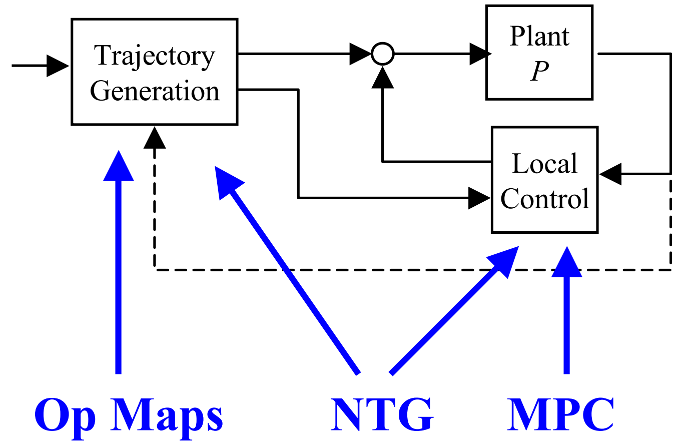
Short Term Tasks: Implementation and Integration

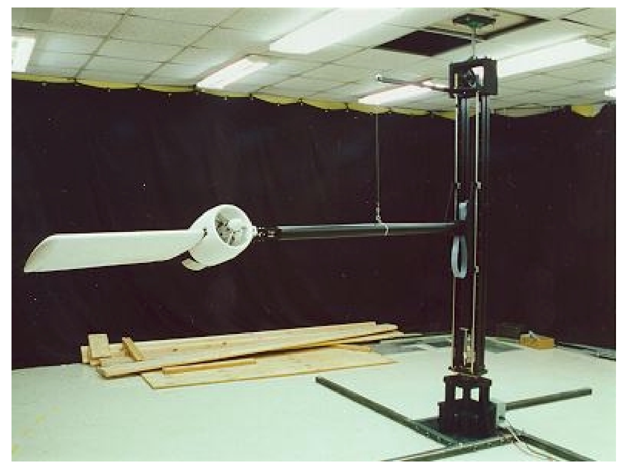
Combine MPC/CLF + NTG + Op Maps

- Switch from RIOTS to NTG for MPC stabilization; implement on dSPACE system
- Use Operability Maps to explore trajectory space; generate maneuver catalog

Status/Issues

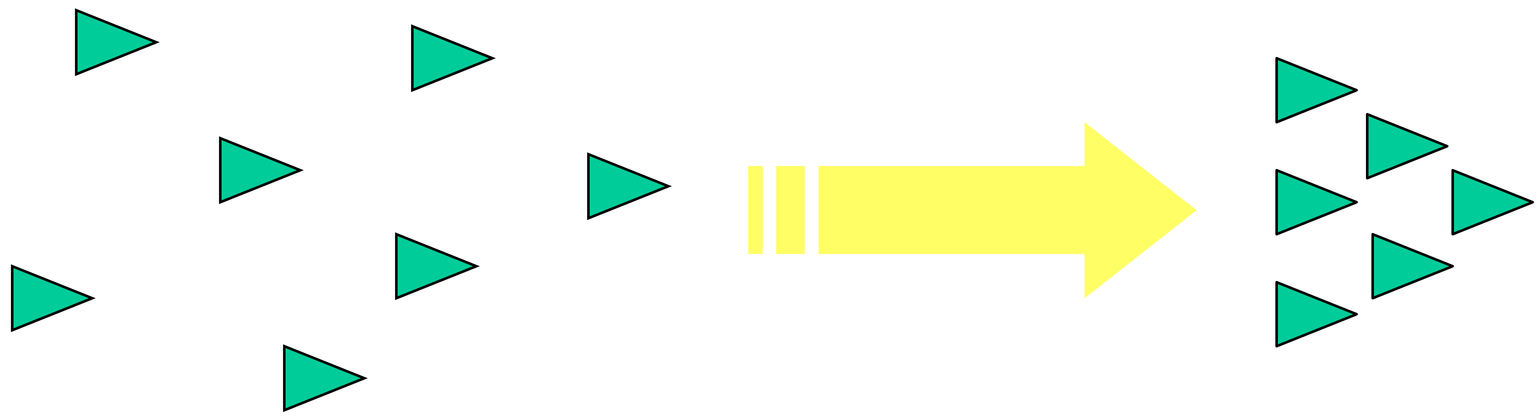
- Improving models (Modelica-based)
- Working on dSPACE implementation of NPSOL (for NTG use)
- Building HW-in-the-loop capability at Colorado for prototyping + multi-vehicle sims
- No plan in place for OCP integration yet
- Jadbabie visit to UTRC in Jun-Jul

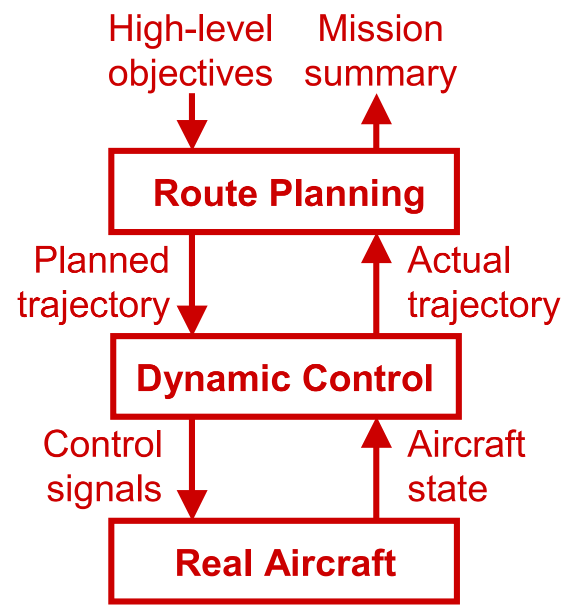




Milestone: single vehicle, acrobatic maneuvering on Caltech ducted fan (Target: 10/1/00)

Long Term Tasks: Multi-Vehicle Rejoin Capability





Understand richness of trajectory space

- Single vehicle → multi-vehicle planning
- Explore operability maps for feasible trajectories

Optimization-based, hierarchical control architecture

- Transition from attitude control to wingtip tracking
- Shift in cost function, model structure

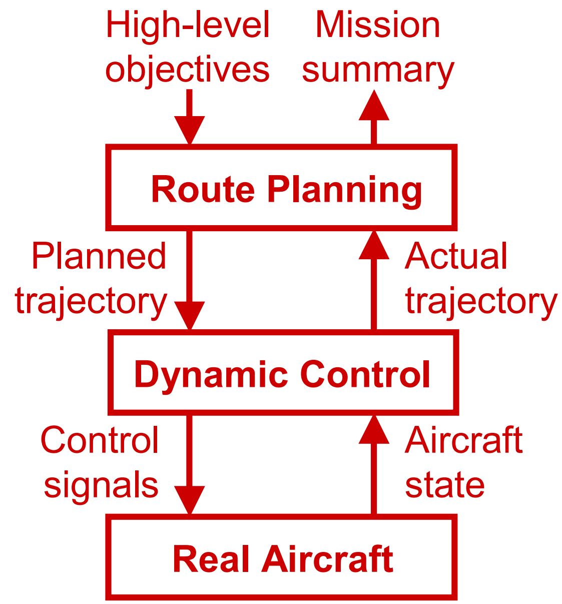
Multi-vehicle coordination/distributed control

- Distribution and coordination of control across flock
- High confidence operation via fault accomodation

Summary

Progress to date

- Theory: optimization based control (MPC + CLF)
- Analysis: exploration of trajectory space thru operability maps
- Computation: real-time, flatness-based trajectory generation



Next steps

- Near term: single vehicle, acrobatic maneuvering on Caltech ducted fan
- Long term: multi-vehicle control strategies for re-join task

Issues

- Need better integration with OCP to insure future compatibility of dSPACE-based tools being developed at Caltech
- Need better integration with active state models to insure compatibility with online optimization algorithms