Adaptive Optimization for System Performance
and
Combined Bernstein Polynomial, Optimal Reciprocal Collision Avoidance, Differential Dynamic Programming for Trajectory Replanning and Collision Avoidance for UAM Vehicles

Matthew Houghton, Michael Acheson, Andrew Patterson, Alex Oshin,
Kasey Ackerman, Irene Gregory

NASA Langley Research Center

AIAA Hampton Roads Section Technical Seminar

16 March 2023

Adaptive Optimization for System Performance: Parameterized Differential Dynamic Programming*

Alex Oshin*#, Matthew D. Houghton*, Michael J. Acheson*, Irene M. Gregory* and Evangelos Theodorou#

*NASA Langley Research Center**

Georgia Institute of Technology#

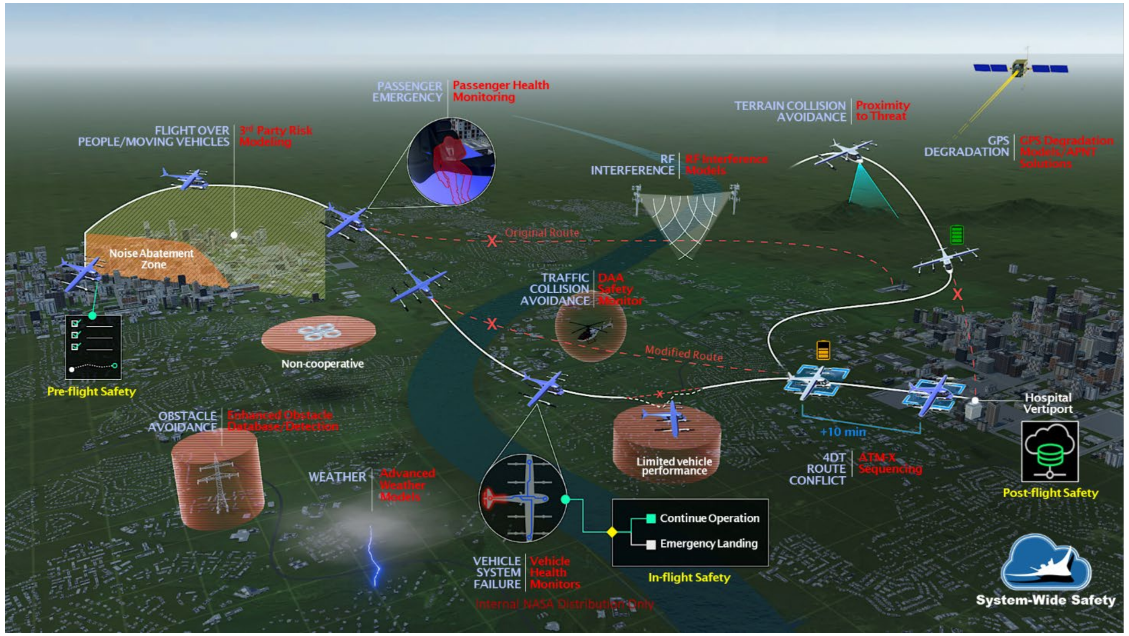
*Oshin, Alexander B., Houghton, Matthew D., Acheson, Michael J., Gregory, Irene M., Theodorou, Evangelos A., “Parameterized Differential Dynamic Programming,” In the Proceedings of Robotics: Science and Systems, June 27-July1, 2022, New York, USA.

Irene.M.Gregory@nasa.gov

Motivation for Adaptive Optimization for System Performance

Emerging aerospace sectors – missions and vehicles

- Autonomous cargo delivery
- Urban Air Mobility (UAM)
- Complexity of the environment
- Unconventional configurations with multi-modal dynamics - rotor-borne vertical takeoff/landing, fixed-wing cruise, transition phase between the two
- Highly nonlinear flight dynamics
- Autonomous flight for scalability

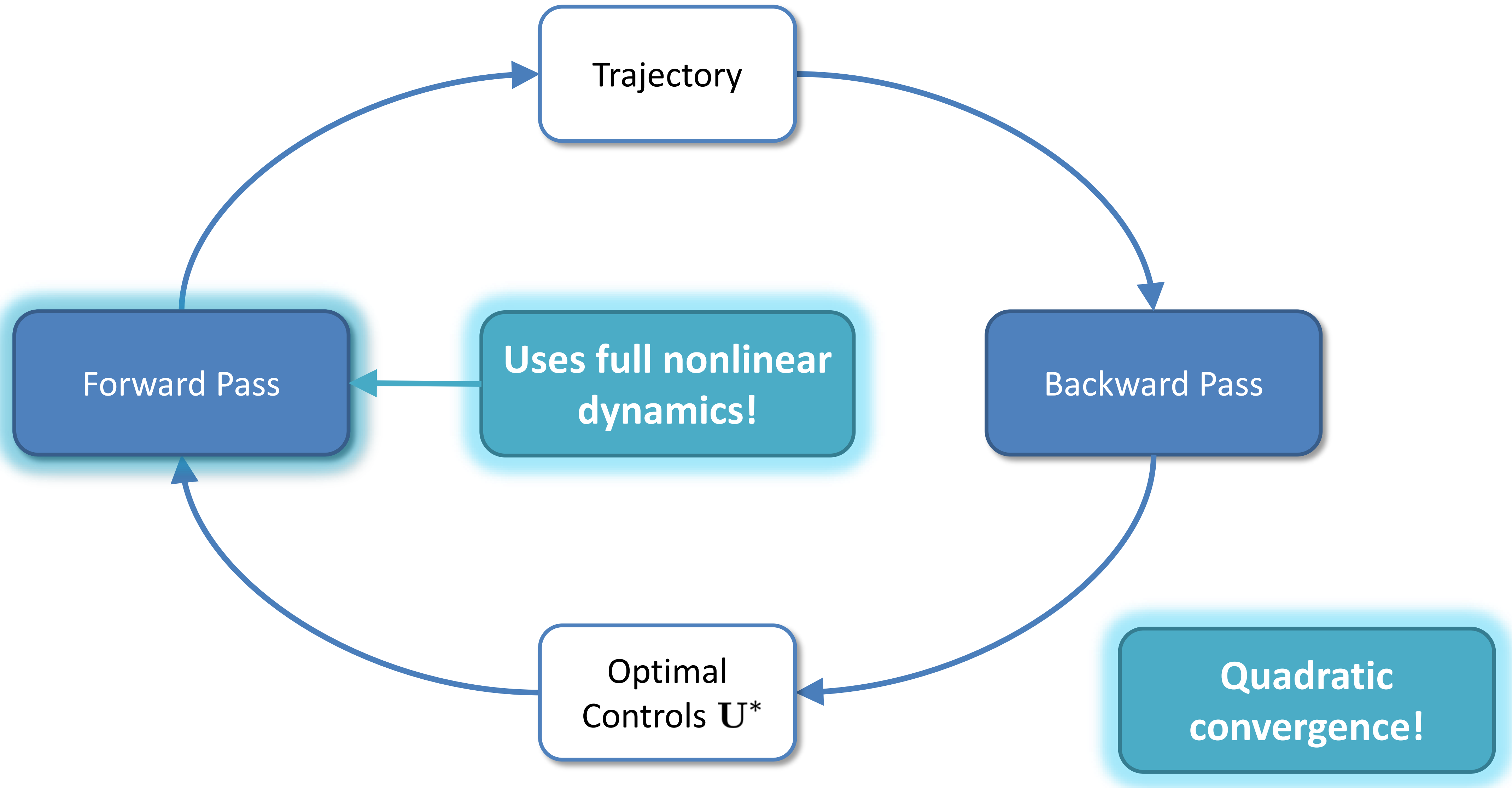


Planner challenges:

- Principled solutions/guarantees
- Accurate trajectory planning & replanning
- Epistemic uncertainty in model
- Multiple operational modes and flight regimes
- Transferability to different vehicles

Irene.M.Gregory@nasa.gov

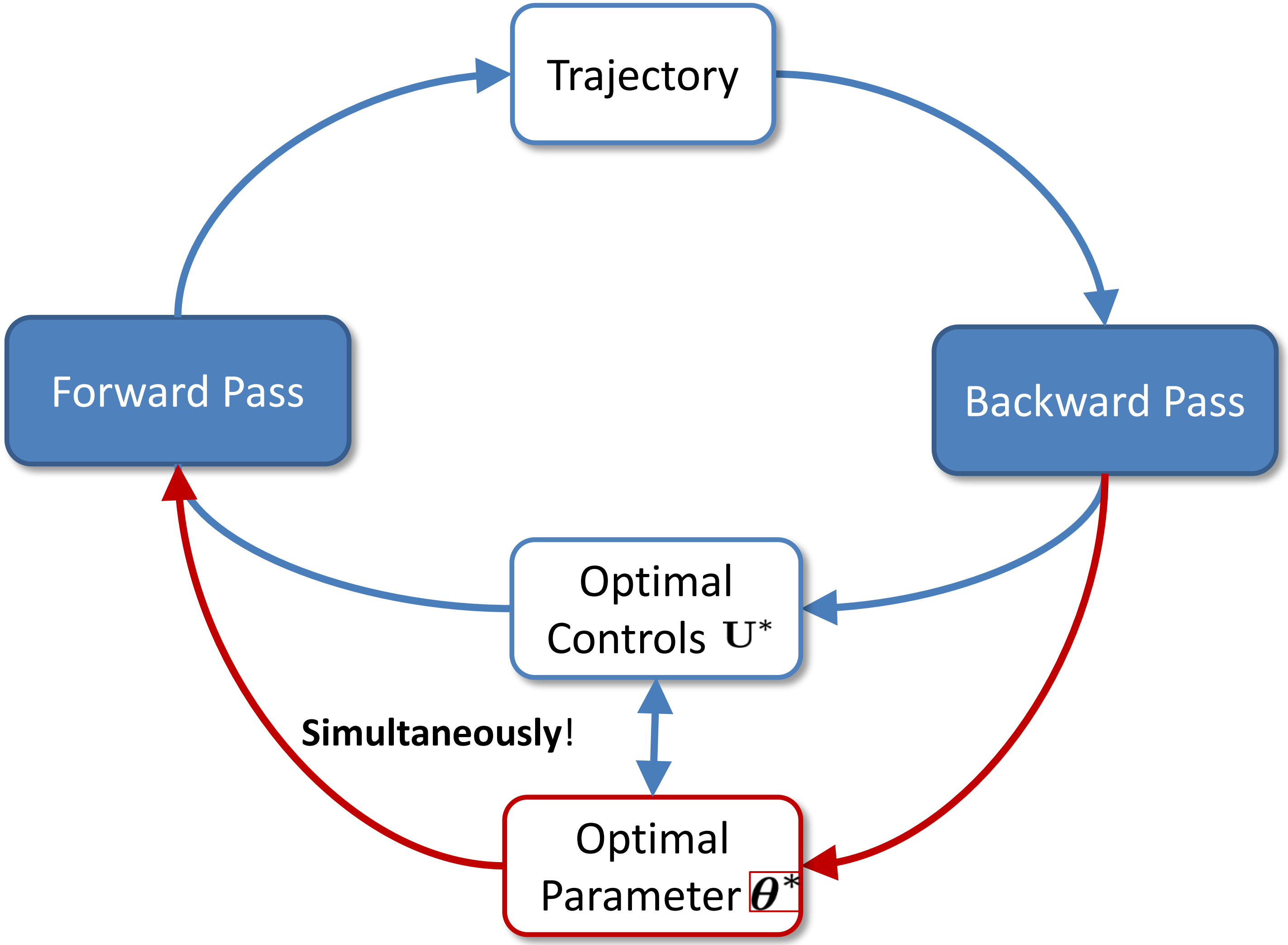
Differential Dynamic Programming (DDP)



Irene.M.Gregory@nasa.gov

Parameterized Differential Dynamic Programming (PDDP)*

- Second-order algorithm derived by extending classical optimal control (DDP)
- **Convergence guarantees** independent of initialization
- **Co-optimizes** for controls and parameters simultaneously
- **Generalizes** to multiple tasks, including adaptive MPC and switching time optimization
- Enables time-optimal trajectory planning for multimodal systems, including **UAM vehicles**



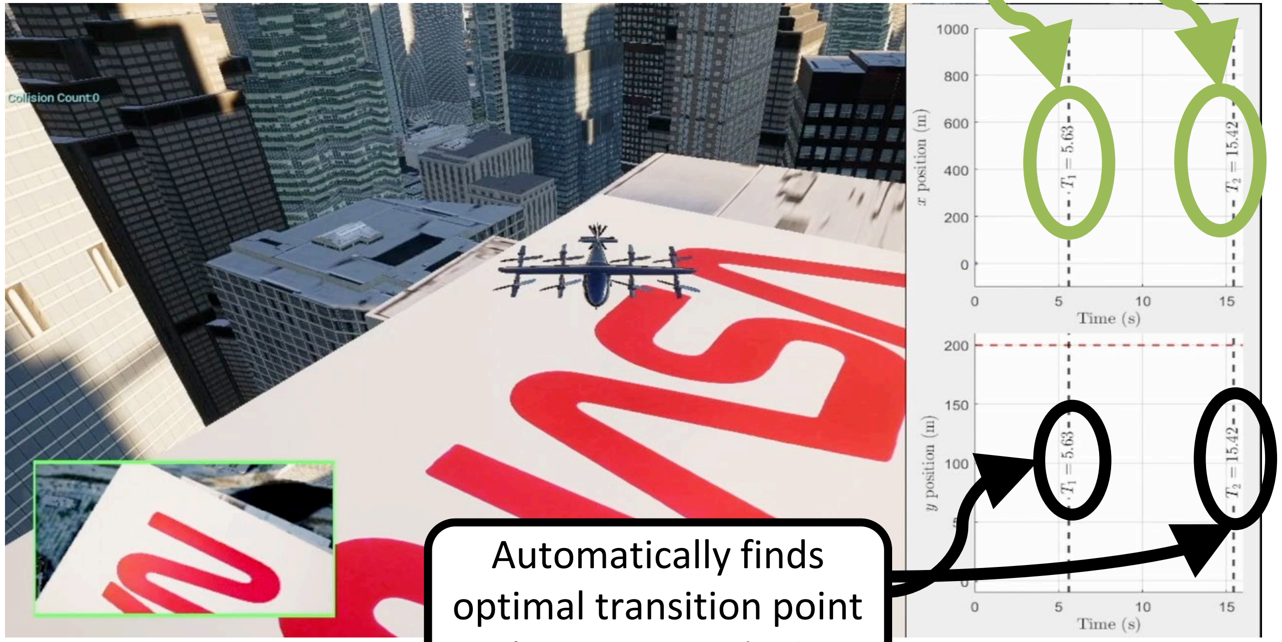
* Oshin, A., Houghton, M., Acheson, M., Gregory, I., and Theodorou, E., "Parameterized Differential Dynamic Programming," *Proceedings of Robotics: Science and Systems*, New York City, NY, USA, 2022. https://doi.org/10.15607/RSS.2022.XVIII.046.

Irene.M.Gregory@nasa.gov

PDDP Applications

Switching Time Optimization

Avoids manual tuning of terminal times!

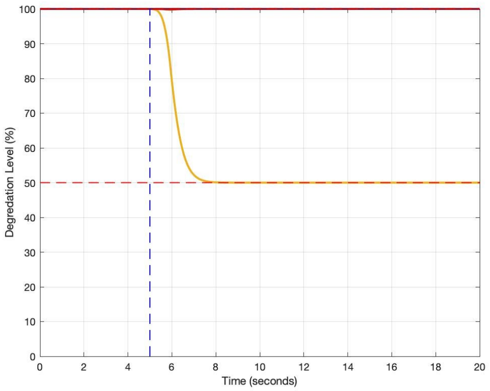


Automatically finds optimal transition point between modes!

Adaptive Model Predictive Control

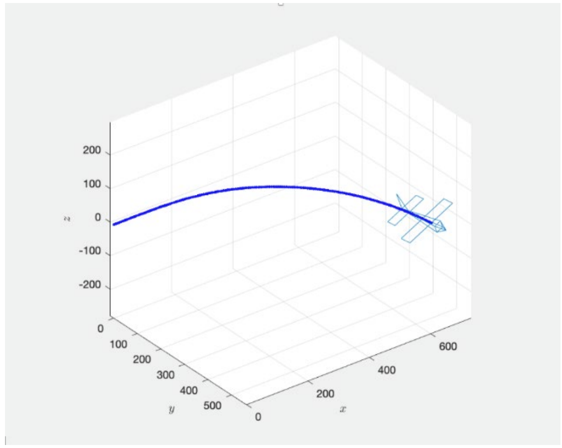
Moving Horizon Estimation

Maximize likelihood of observed states



Model Predictive Control

Plan future trajectory



Irene.M.Gregory@nasa.gov

PDDP Transition Optimization

- Long-term planning requires the L+C vehicle to change operating modes from hover to forward flight. Classical trajectory planning methods struggle with determining how to transition between modes.
- PDDP L+C experiment involves vertical takeoff into cruise transition with multiple target states.
- Switching Time Optimization selects the **optimal transition** times between targets and flight regimes (without direct input from researchers).

Multimodal System

$$\dot{\mathbf{x}}(t) = \mathbf{f}^{(i)}(\mathbf{x}(t), \mathbf{u}(t))$$

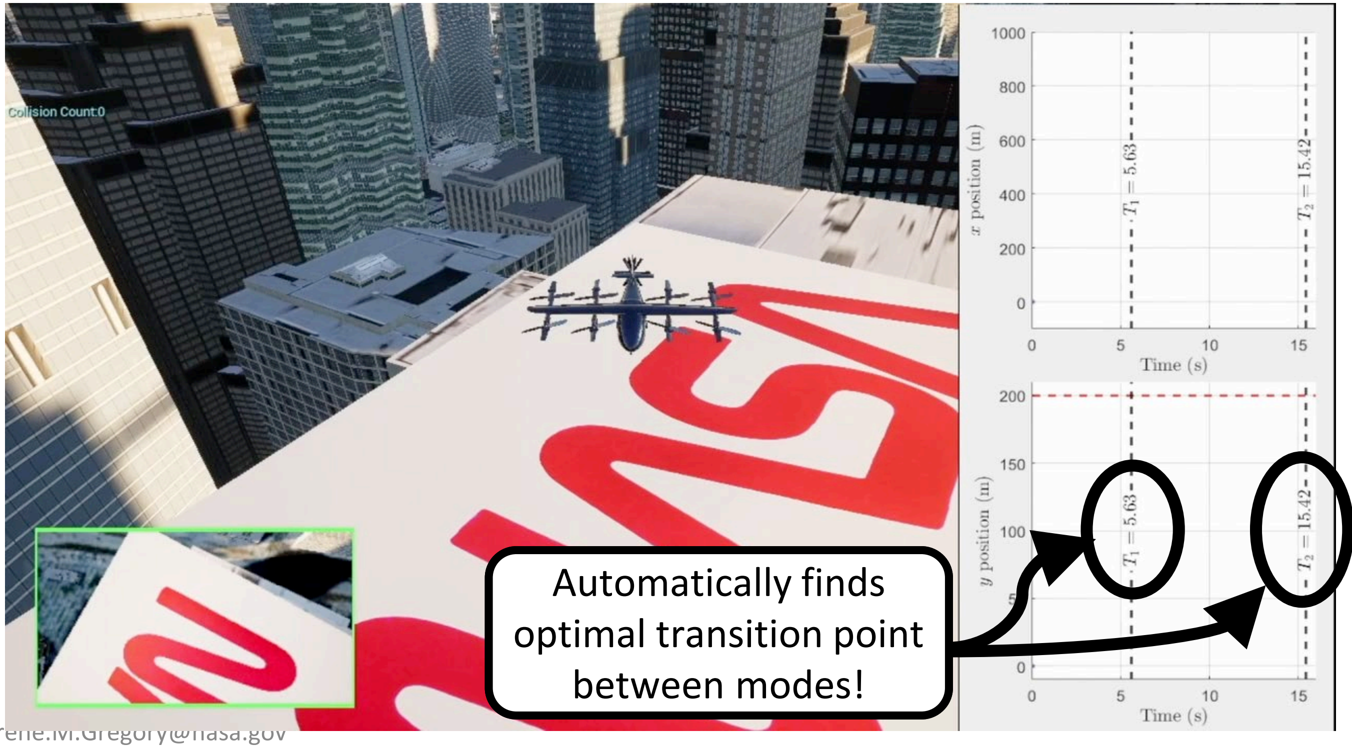
$i = 1, 2, \ldots, N$

Switching Time System

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \theta_i \mathbf{f}^{(i)}(\mathbf{x}_t, \mathbf{u}_t)\Delta t$$

$$\sum_{t=T_{i-1}+1}^{T_i} \theta_i \mathcal{L}^{(i)}(\mathbf{x}_t, \mathbf{u}_t) + \phi^{(i)}(\mathbf{x}_{T_i+1})$$

$i = 1, 2, \ldots, N$

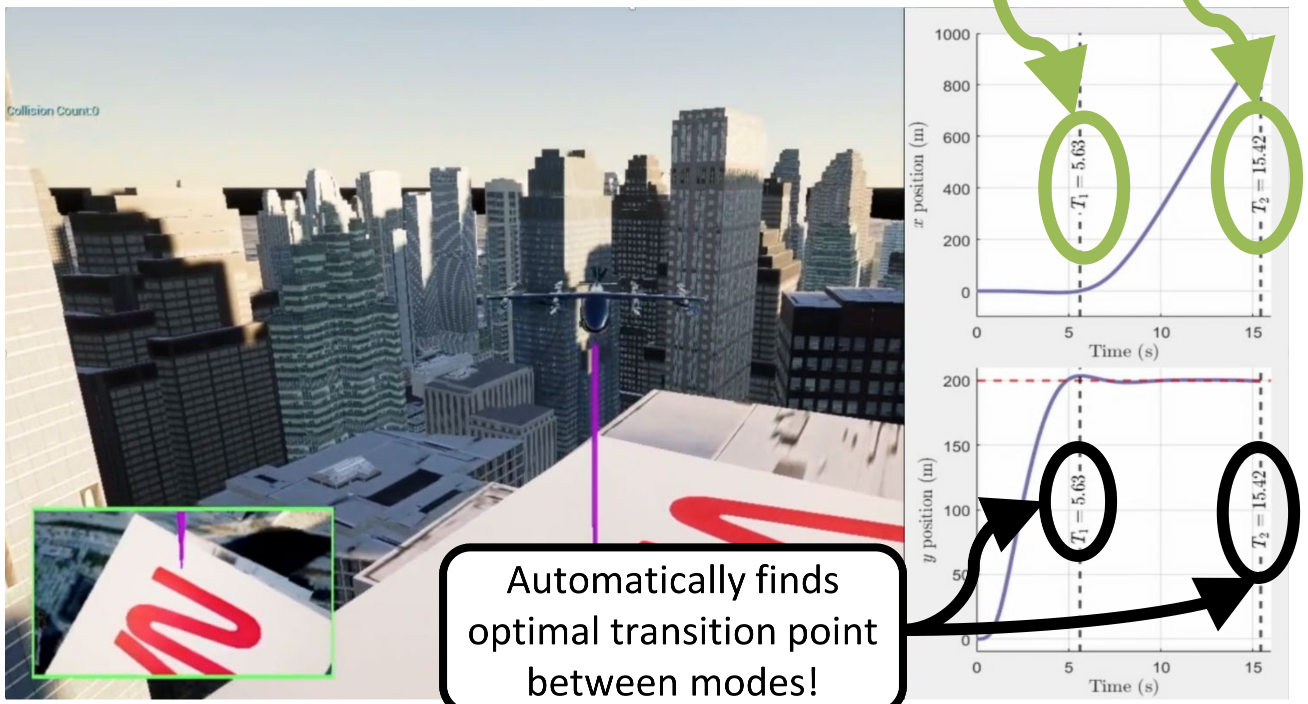


Irene.M.Gregory@nasa.gov

PDDP Applications

Switching Time Optimization

Avoids manual tuning of terminal times!

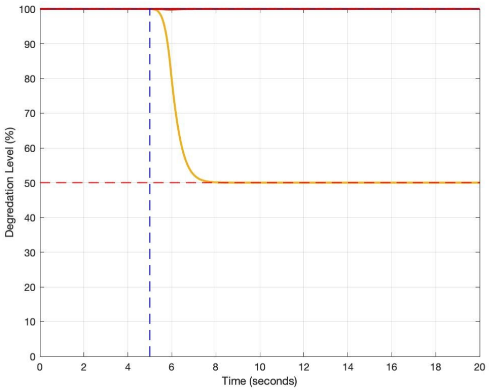


Automatically finds optimal transition point between modes!

Adaptive Model Predictive Control

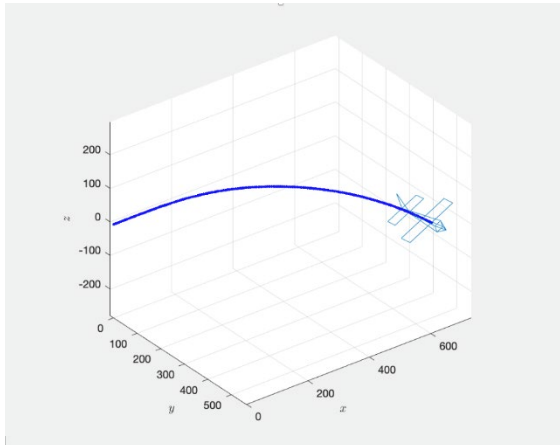
Moving Horizon Estimation

Maximize likelihood of observed states



Model Predictive Control

Plan future trajectory



Irene.M.Gregory@nasa.gov

PDDP Applications

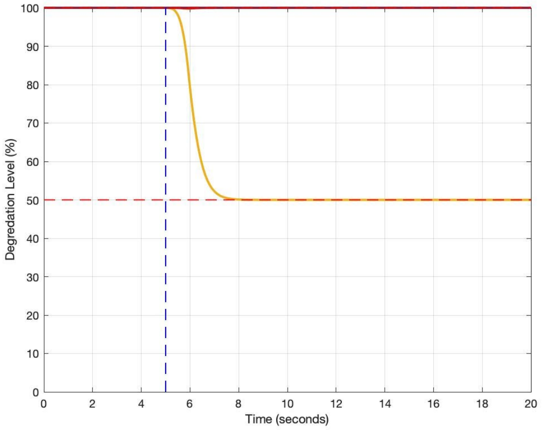
Switching Time Optimization

Avoids manual tuning of terminal times!

Adaptive Model Predictive Control

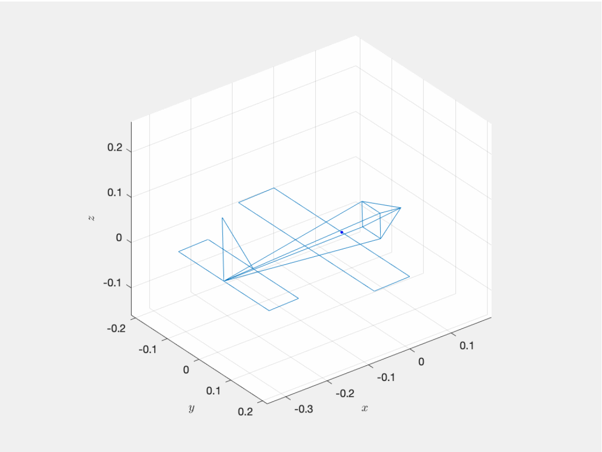
Moving Horizon Estimation

Maximize likelihood of observed states



Model Predictive Control

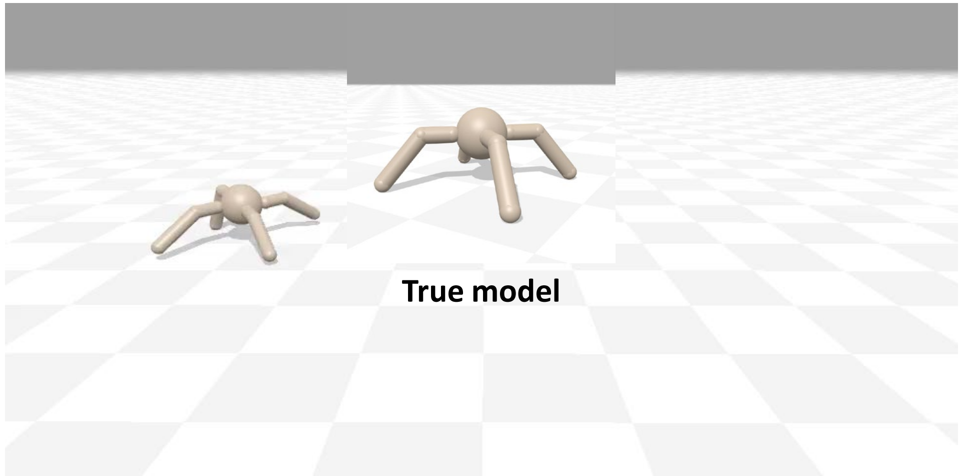
Plan future trajectory



Irene.M.Gregory@nasa.gov

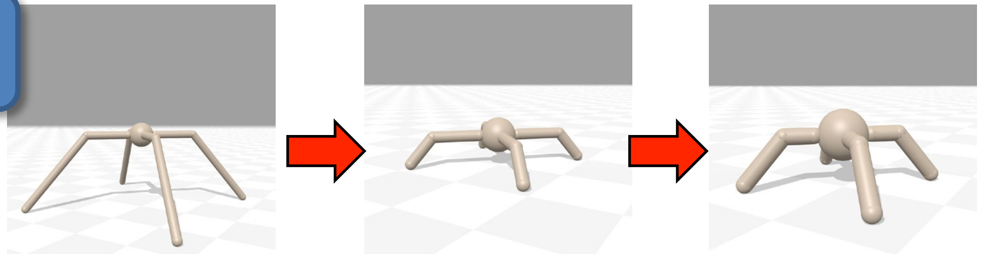
PDDP: Adaptive MPC

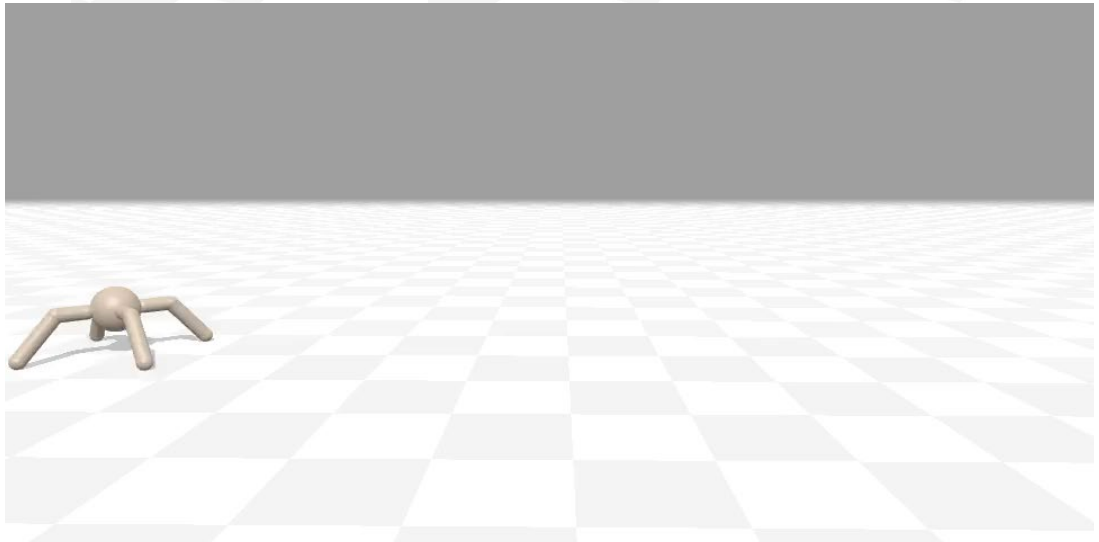
DDP planning on model with incorrect parameters



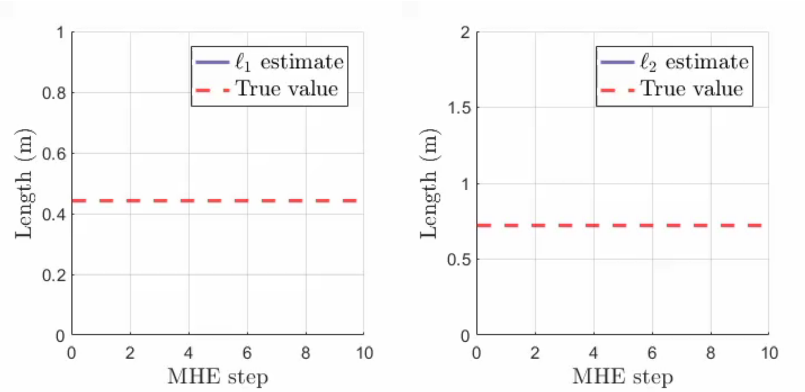
Executing plan on true model: Failure

Ant





PDDP with adaptive control: Success



Irene.M.Gregory@nasa.gov

Parameterized Differential Dynamic Programming (PDDP)

PDDP is a trajectory optimization algorithm that builds upon DDP

- Enables the **co-optimization** of a **trajectory** and time invariant **parameters** in the same process.
- Parameters can be extremely diverse and goal specific
- Experiments tested PDDP's ability to successfully **estimate vehicle dynamic parameters** while implementing **optimal trajectories**, resulting in Adaptive Model Predictive Control

Fault Detection

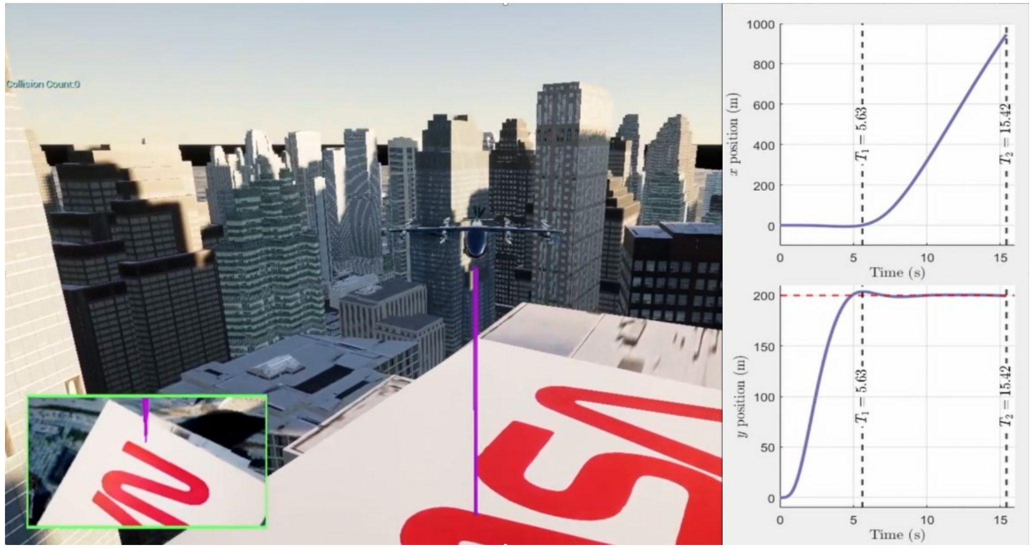
- Online estimation of vehicle **dynamic** parameters
- **Online estimation** of **degradation** level for effectors + rotors
- **Replan trajectory** based on new estimation of vehicle parameters
- Deviations in estimation from norms can alert system ID of vehicle to run further diagnostics of vehicle health

Irene.M.Gregory@nasa.gov

PDDP Applications

Switching Time Optimization

Avoids manual tuning of terminal times!



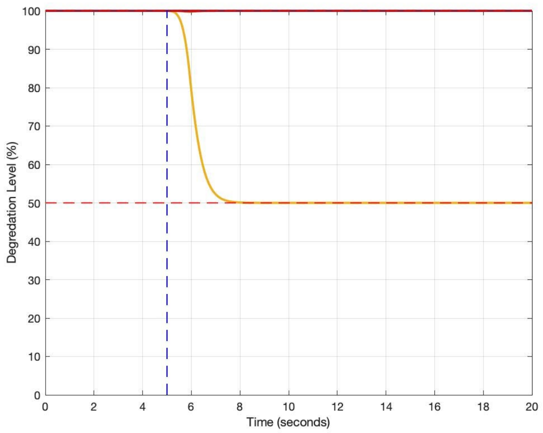
Adaptive Model Predictive Control

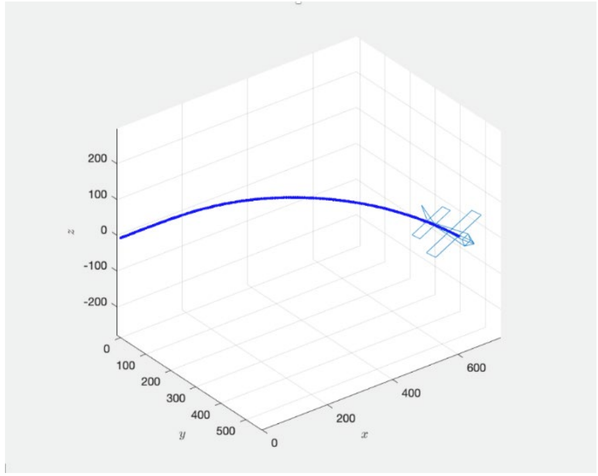
Moving Horizon Estimation

Maximize likelihood of observed states

Model Predictive Control

Plan future trajectory





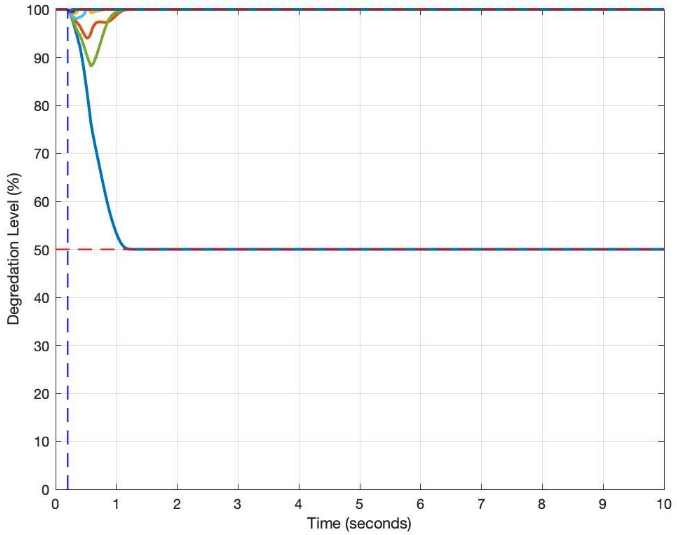
Irene.M.Gregory@nasa.gov

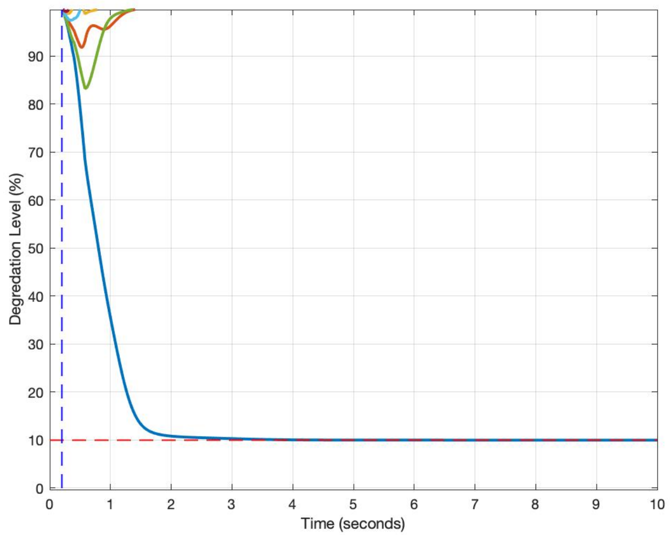
Fault Detection: Rotor Failure

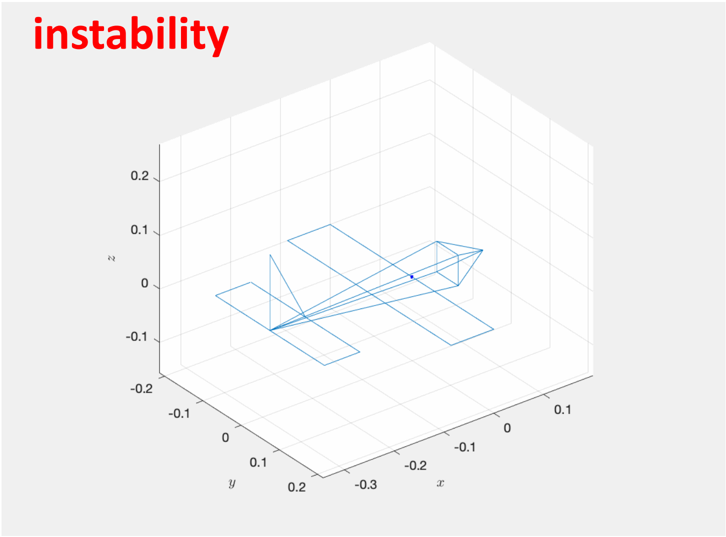
PDDP extends to Fault Detection of vehicle states (rotors and effectors)

<u>Experiment 1: Vertical Takeoff</u>

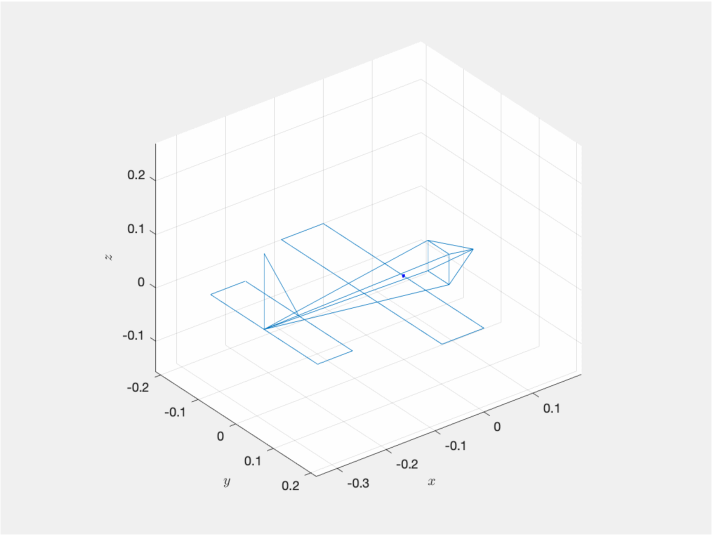
- Begin in hover
- Early Failure/Degradation
- Ascent to 200 ft
- Heavily utilizes rotors in VTOL flight regime



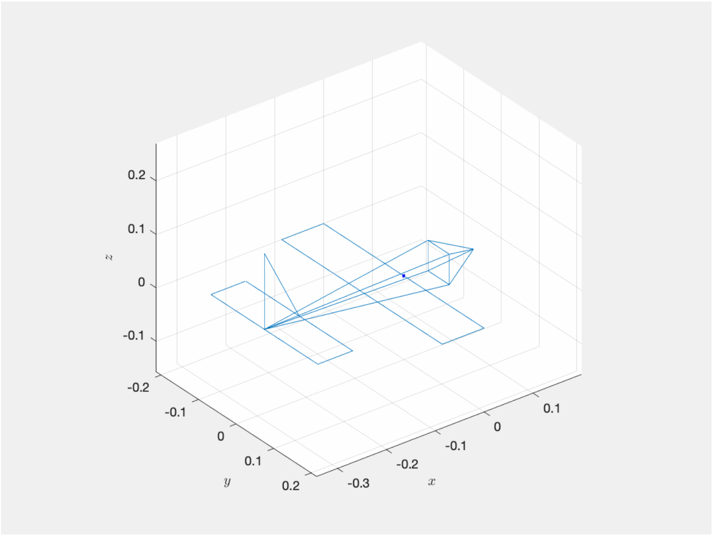




Takeoff Failure Without PDDP



50 % Rotor 1 Degradation



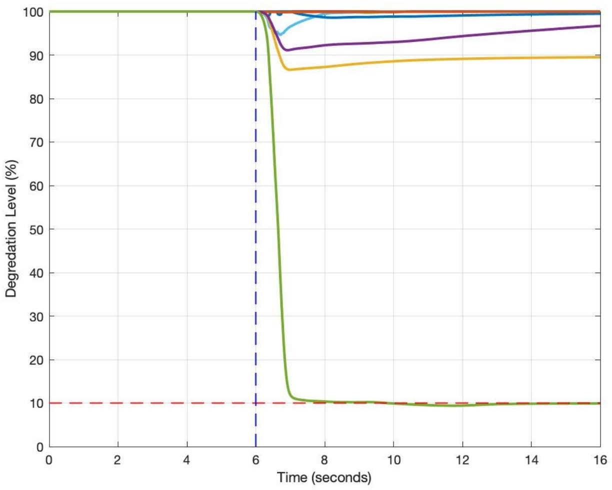
90 % Rotor 1 Degradation

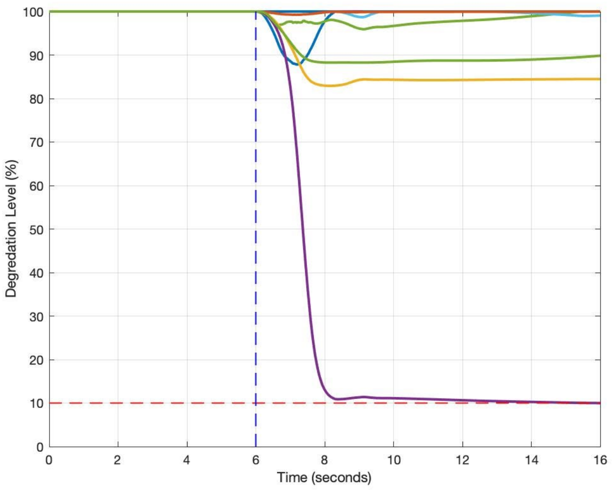
Irene.M.Gregory@nasa.gov

Fault Detection: Effectors

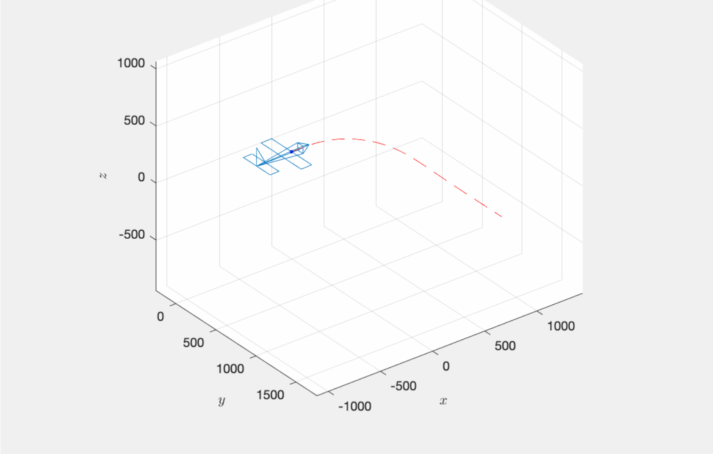
Experiment 2: Bank Right Turn

- Begin in fixed-wing cruise
- Failure/Degradation at 6 seconds
- Perform a right bank turn
- Heavily utilizes effectors in fixed-wing flight regime

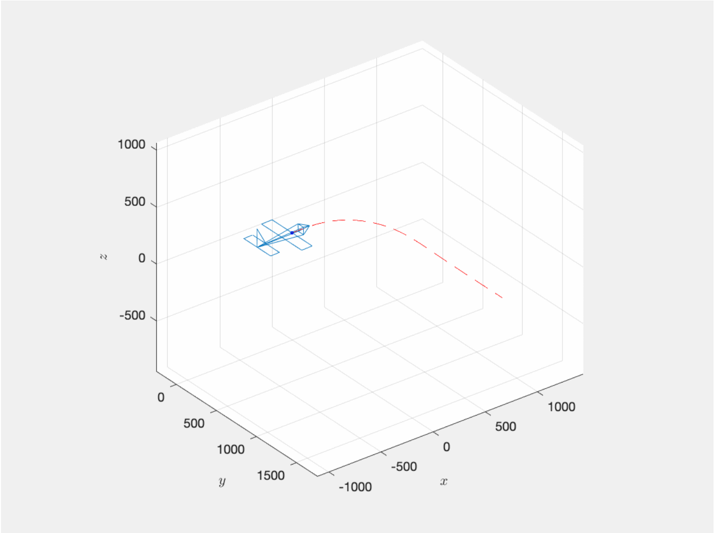




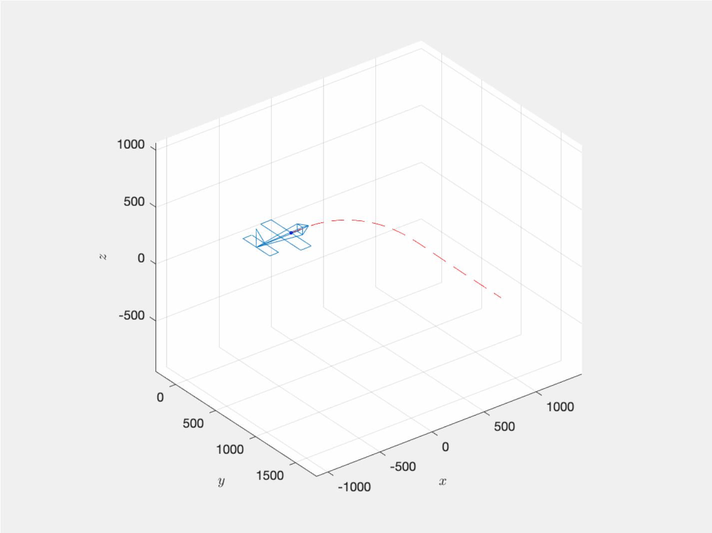
Vehicle discontinues turn after failure



Failure Mid Bank Turn No PDDP



90 % Rudder Degradation



90 % Aileron Degradation

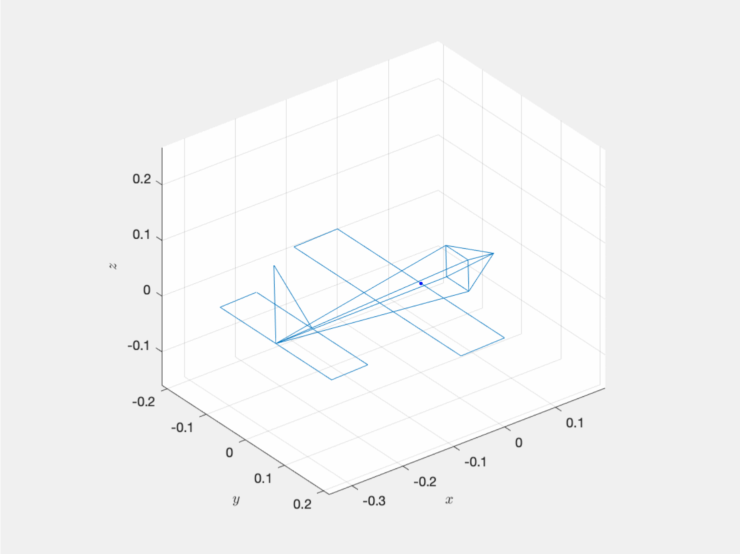
Irene.M.Gregory@nasa.gov

Fault Detection: Split Effector Failure

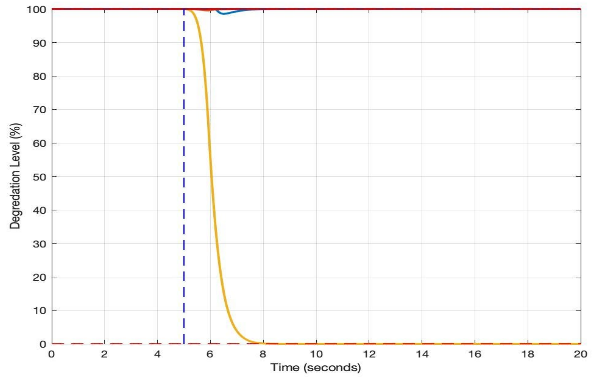
Experiment 3: Split Effector Bank Right

- Previous state configuration for L+C has used ganged effectors
- This experiment added state values of both the LEFT and RIGHT Ailerons
- Added states found to reduce the uncertainty of PDDP's parameter estimation even at small degradation values
- Failure/Degradation of Left Aileron ONLY at 5 seconds for bank right turn experiment
- All experiments capable of replanning a similar trajectory post failure

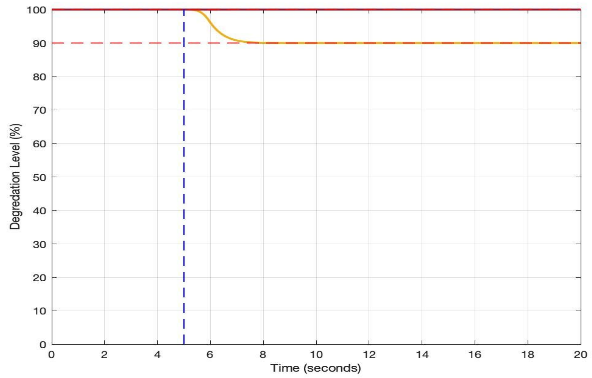
Left Aileron failure Bank Right



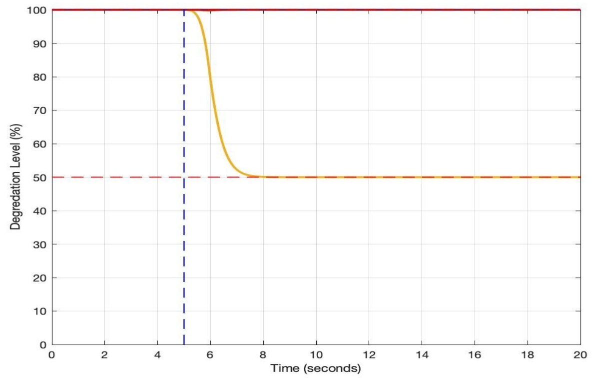
100% Degradation



10% Degradation



50% Degradation

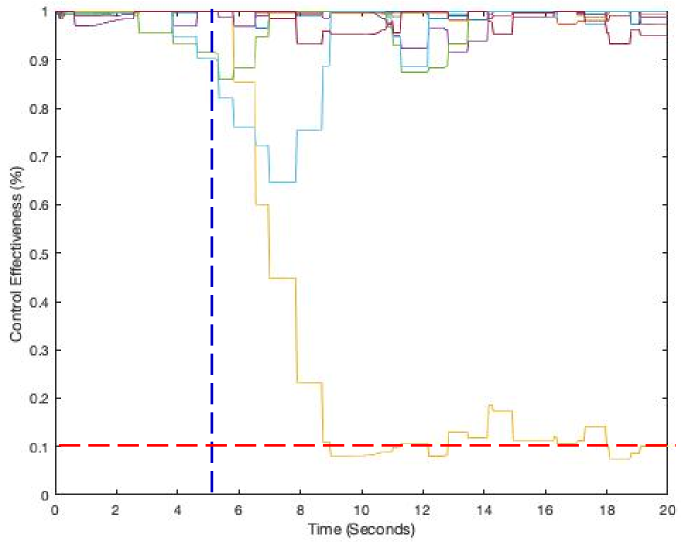


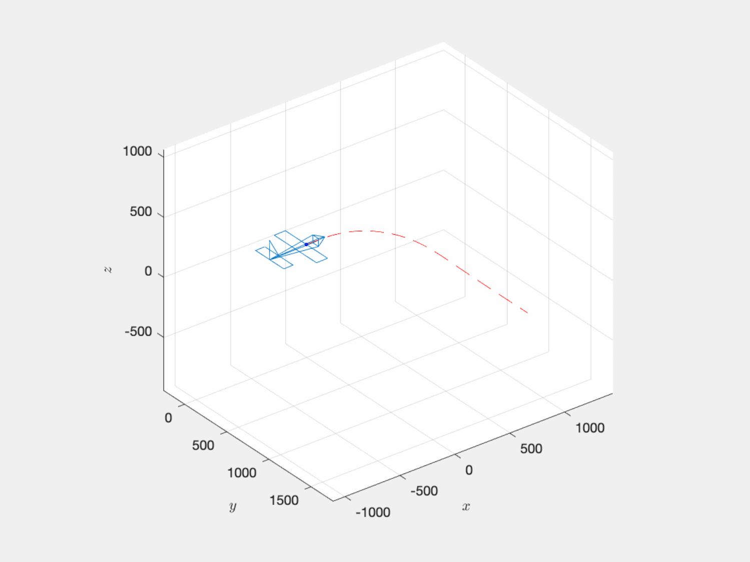
Irene.M.Gregory@nasa.gov

Fault Detection: Estimation with Noise

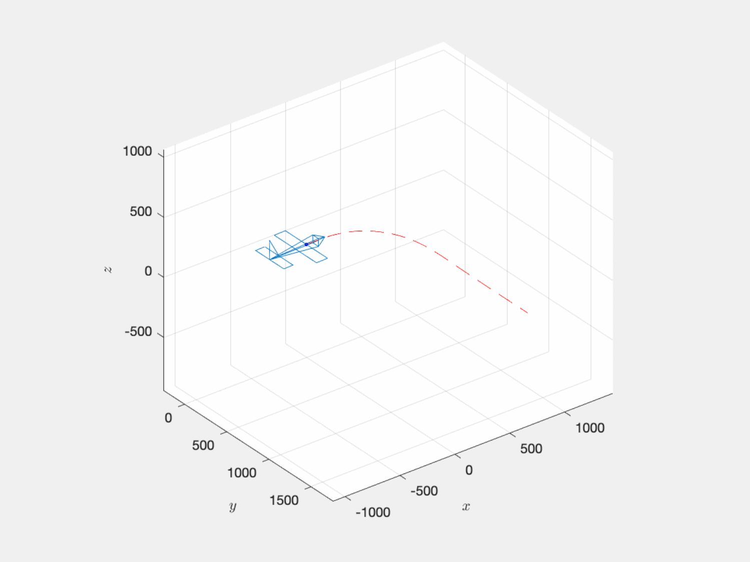
Experiment 3: Bank Right Turn

- Inclusion of process and measurement noise causes non-PDDP informed case to fail
- PDDP successfully maintains vehicle stability and plans trajectory using modified dynamics





90 % Split Aileron Degradation no PDDP



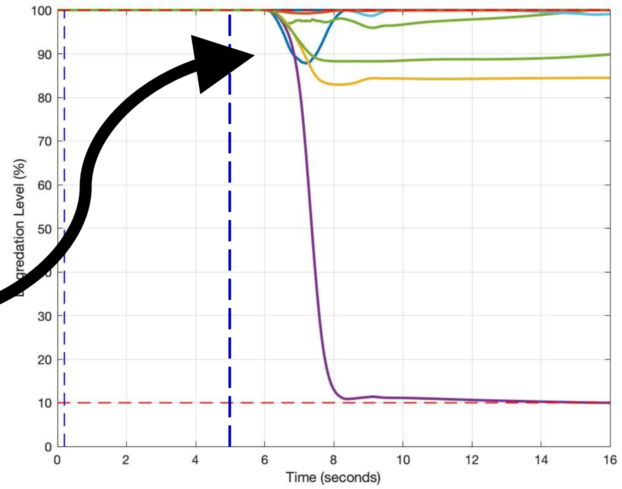
90 % Split Aileron Degradation

Fault Detection: Effect of Split Effector Failure

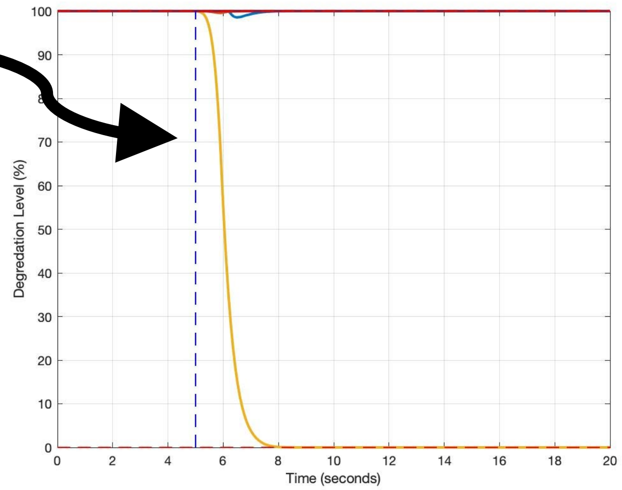
<u>Results</u>

- PDDP effectively utilizes state information to estimate both severe and minimal failures
- PDDP can replan using updated parameters in MPC fashion
- PDDP estimates are improved by utilization of state and the specificity of state information
- PDDP is sensitive enough to inform system ID to minor and major degradation/failures

Note:
Giving PDDP greater access to specific vehicle states improves the distinguishability of fault estimation



90 % Ganged Aileron Degradation



100 % Split Aileron Degradation

Irene.M.Gregory@nasa.gov

Summary - Parameterized Differential Dynamic Programming (PDDP)

- Second-order algorithm derived by extending classical optimal control
- **Convergence guarantees** independent of initialization
- **Co-optimizes** for controls and parameters simultaneously
- **Generalizes** to multiple tasks, including adaptive MPC and switching time optimization
- Enables time-optimal trajectory planning for multimodal systems, including **UAM vehicles**

Application of PDDP – Current experimentation and directions

- **Fault detection** (parameter estimation)
  - Can run both as a full optimal control or strictly in the backward path to identify dynamic degradation
- **Adaptive MPC** - Replanning trajectory to accommodate new identified dynamics
  - Even when vehicle is incapable of following original trajectory new trajectory is planned to attain the original goal as closely as dynamically feasible
- **Switching Time Optimization**
  - **Optimal transition time** between flight regimes (difficult for highly nonlinear vehicles like L+C)
  - **Decreases tuning** work for engineers when planning for common maneuvers that transition between flight regimes
  - Allows for the input and optimization of multiple target states for **long-term planning** and replanning

Irene.M.Gregory@nasa.gov

Combined Bernstein Polynomial, Optimal Reciprocal Collision Avoidance, Differential Dynamic Programming for Trajectory Replanning and Collision Avoidance for UAM Vehicles*

Matthew Houghton, Michael Acheson, Andrew Patterson, Alex Oshin, Irene Gregory

NASA Langley Research Center

*Houghton, Matthew D., Acheson, Michael A., Oshin, Patterson, Andrew P., Alexander B., Gregory, Irene M., "Combined Bernstein Polynomial, Optimal Reciprocal Collision Avoidance, Differential Dynamic Programming for Trajectory Replanning and Collision Avoidance for UAM Vehicles" 2023 AIAA SciTech Forum, National Harbor, MD, January 2023. AIAA-2023-2544

Outline

- Motivation: Trajectory Replanning and Collision Avoidance for VTOL vehicles with highly nonlinear dynamics are slow and computationally costly
- COBRA-DDP proposed
  - Bernstein Polynomials
  - Optimal Reciprocal Collision Avoidance
  - Differential Dynamic Programming
- Experiments:
  - Sample Results demonstrating the algorithm
- Conclusions and Ongoing Work

Lift + Cruise VTOL Vehicle

Irene.M.Gregory@nasa.gov

(CO)mbined (B)ernstein Polynomial, Optimal (R)eciprocal Collision (A)voidance DDP

Trajectory Re-planner Requirements:

- "Real-time" dynamically feasible trajectories for UAM class (transitioning) vehicles with separation assurances
- Dynamic planning for large number of (stationary & moving) cooperative/uncooperative obstacles

Combine to get best of each algorithm!

Piecewise Bernstein Polynomial Curves:

- Advantages: Fast and compact trajectory representation, smooth derivatives (position, velocity & acceleration)
- Disadvantages: one piece-wise segment can't represent all curves exactly (e.g., circular arcs)

Optimal Reciprocal Collision Avoidance (ORCA):

- Advantages: fast computation for large number of cooperative/non-cooperative with separation assurances
- Disadvantages: no assurance of dynamic feasibility

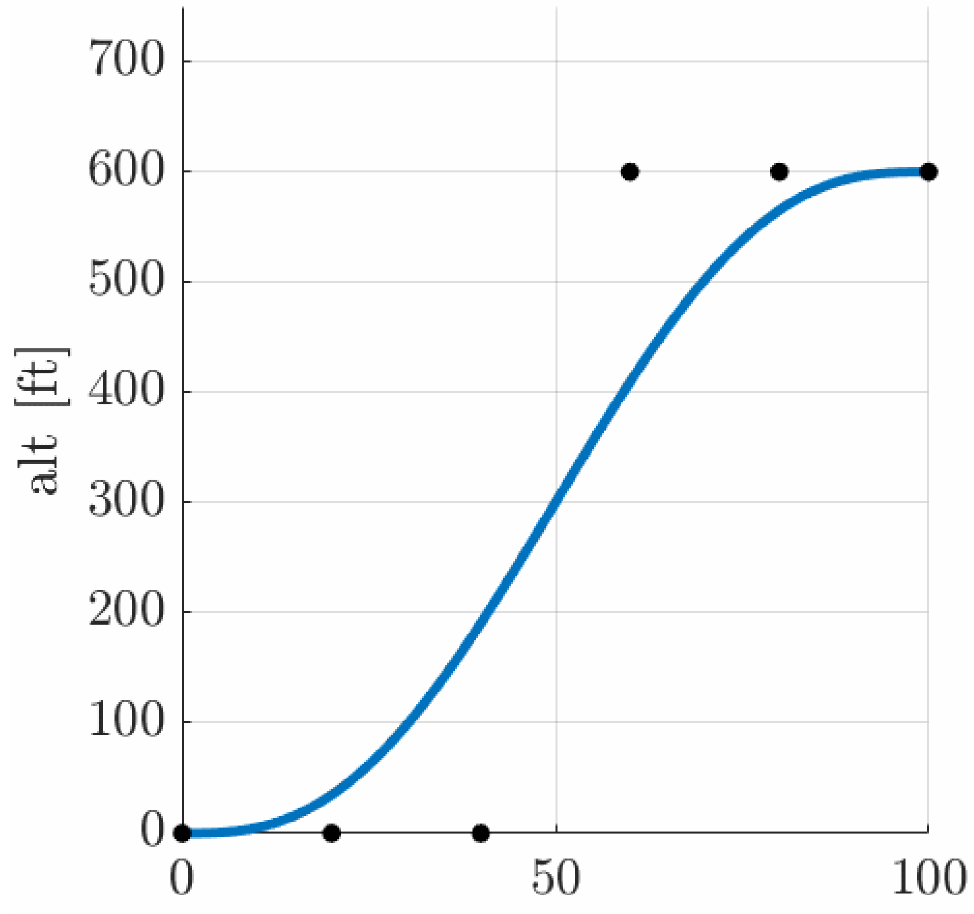
Differential Dynamic Programming (DDP):

- Advantages: fast computation of dynamically feasible optimal trajectories
- Disadvantages: Degraded computation speed for incorporation of state constraints (e.g., obstacles)

Irene.M.Gregory@nasa.gov

Bernstein Polynomials

- Original Emphasis was mathematical



Irene.M.Gregory@nasa.gov

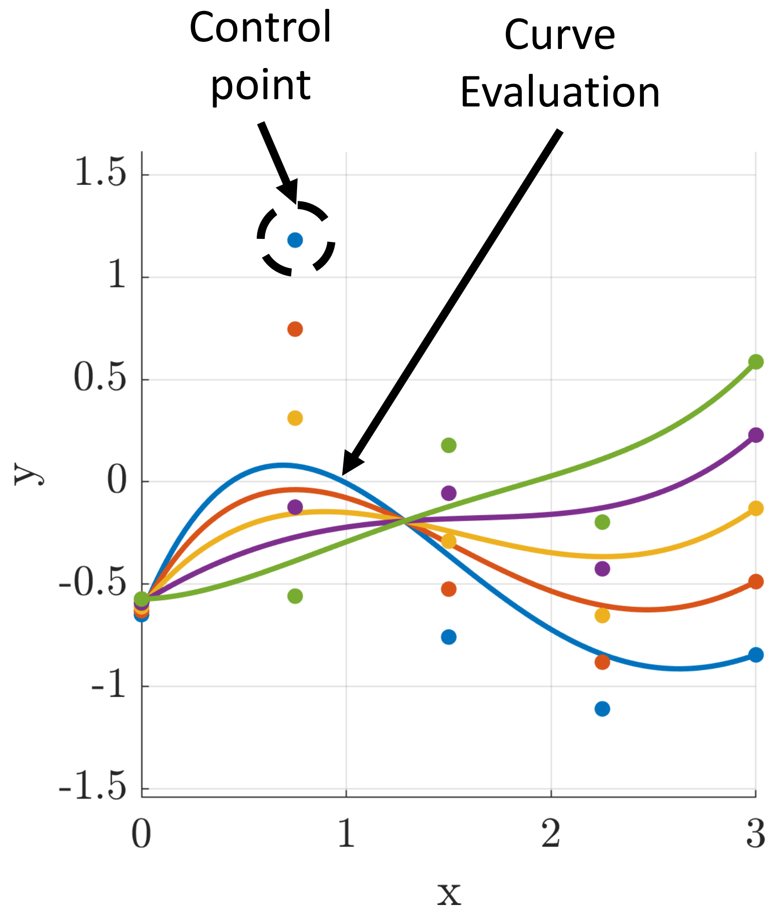
Bernstein Polynomials

Basics

- Polynomial curves using Bernstein basis (rather than monomial basis)
- Polynomial coefficients become control points

Benefits of Bernstein Form

- Control points have physical interpretation
  - Curve connected to end points
  - Curve contained inside control points
- Fast collision and constraint checking algorithms
  - Dynamic feasibility checks
- Differentiation yields Bernstein curve



Bernstein Polynomial control points and evaluations.

Irene.M.Gregory@nasa.gov

Dynamic Waypoints

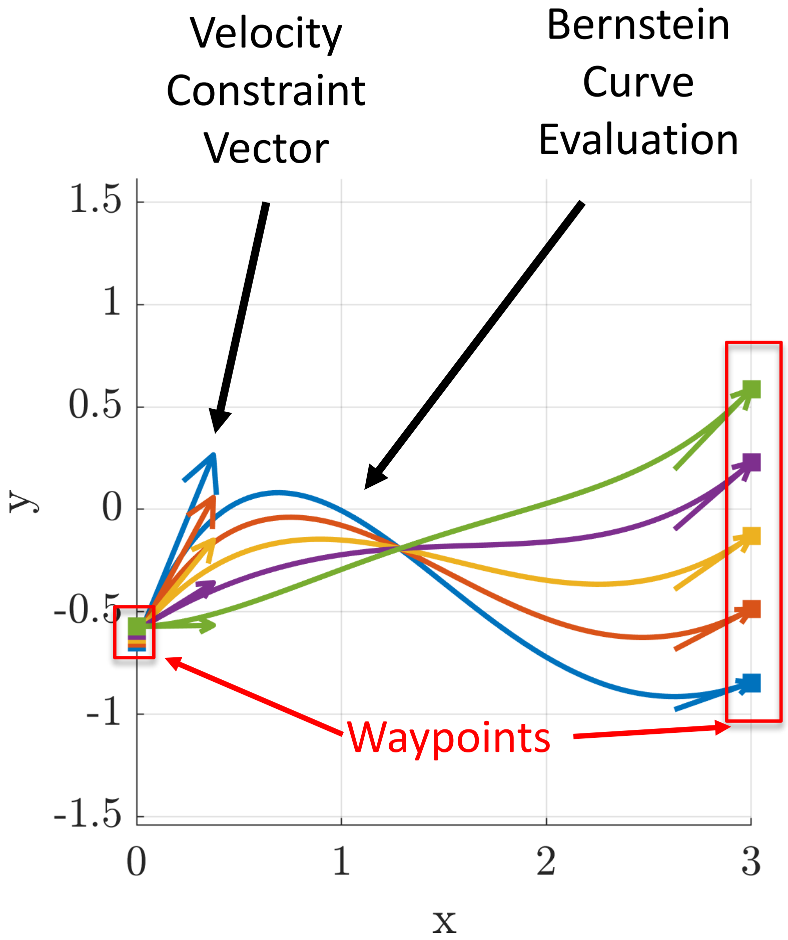
- Waypoints allow specification of spatial/dynamic initial/terminal constraints.
- *e.g. climb from 100 to 200 ft and accelerate from 0 fps to 4 fps.*

Conversion

- Generate Bernstein control points from waypoints (matrix multiplication)
- Bernstein polynomials are the back-end

Use

- ORCA waypoints are expressed as terminal constraints on time, position, velocity
- DDP samples Bernstein polynomial at arbitrary times



Waypoints connected by polynomial curve evaluations (Only position and velocity constraints shown)

Irene.M.Gregory@nasa.gov

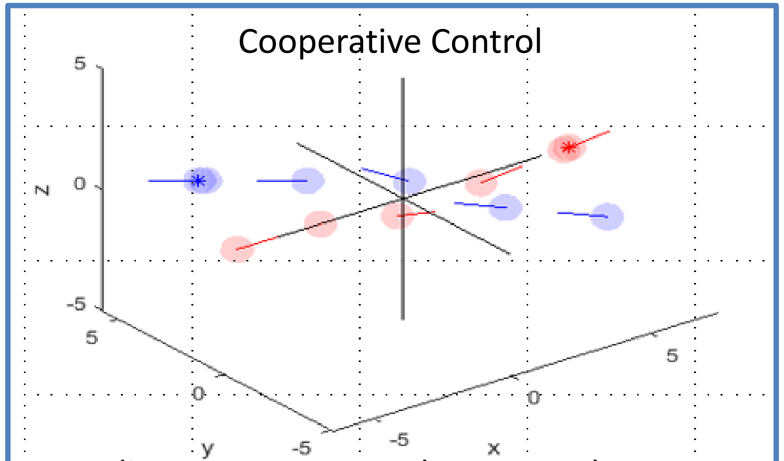
ORCA

Optimal Reciprocal Collision Avoidance Algorithm (robotics community focused):

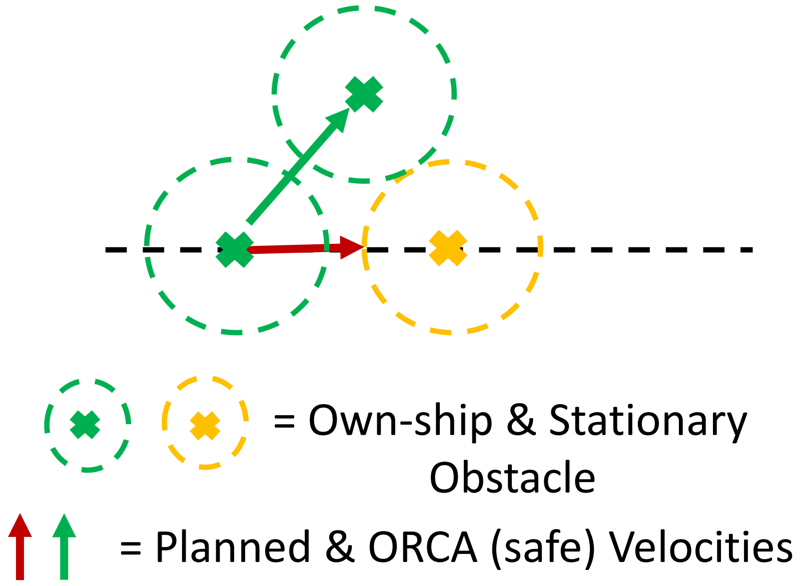
Collision Avoidance: robotics literature defines as autonomous robot navigation with fixed/moving obstacles (other intelligent vehicles)
Recurring cycle: sense/act, repeat

ORCA:

- Input: position and velocity knowledge (own-ship, obstacles/vehicles)
- Output: next own-ship velocity step (magnitude and direction)
- Point modeling (no vehicle dynamics) with safety sphere (keep-out radius)
- "Velocity object" representations, provide mathematical guarantees of collision free for look-ahead time
- Cooperative law: each vehicles applies ½ velocity correction
- Uncooperative law: own-ship takes 100% of velocity correction

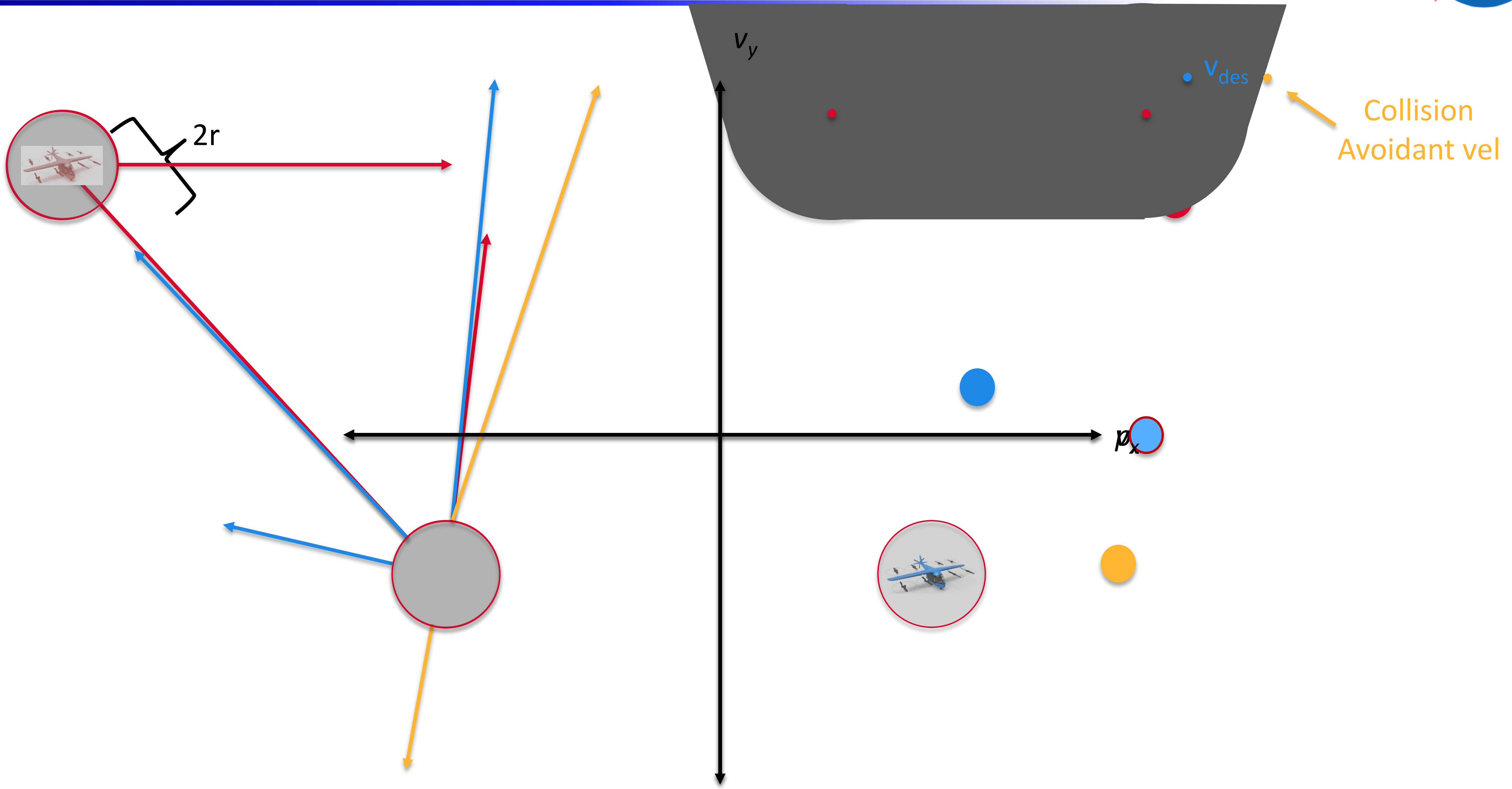


Credit: Bart Bacon, Langley Research Center



Irene.M.Gregory@nasa.gov

ORCA + Velocity Obstacle Explanation



Irene.M.Gregory@nasa.gov

Optimal Trajectory Problem

Problem: Find optimal control (and states) to achieve a desired end state while minimizing cost

Discrete system nonlinear dynamics

$$\mathbf{x}_{t+1} = \mathbf{f}(\mathbf{x}_t, \mathbf{u}_t)$$

State Vector

$$\mathbf{x}_t \in \mathbb{R}^n$$

Control Vector

$$\mathbf{u}_t \in \mathbb{R}^m$$

Cost Function

$$\mathcal{J}(\mathbf{U}) = \sum_{t=1}^{T-1} \mathcal{L}(\mathbf{x}_t, \mathbf{u}_t) + \phi(\mathbf{x}_T)$$

Running Cost

$$\mathcal{L}(\mathbf{x}_t, \mathbf{u}_t)$$

Terminal Cost

$$\phi(\mathbf{x}_T)$$

State Trajectory

$$\mathbf{X} := \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$$

Control Trajectory

$$\mathbf{U} := \{\mathbf{u}_1, \ldots \mathbf{u}_{T-1}\}$$

Finite Time

$$T \in \mathbb{N}^+$$

Irene.M.Gregory@nasa.gov

Differential Dynamic Programming

DDP: Given nominal trajectory, use linear (or quadratic) approx. of system nonlinear dynamics and quadratic approx. of cost to yield updates to optimal controls that quadratically converge

Cost Function

$$\mathcal{J}_i(\mathbf{x}_i, \mathbf{U}_i) := \sum_{t=i}^{T-1} \mathcal{L}(\mathbf{x}_t, \mathbf{u}_t) + \phi(\mathbf{x}_T)$$

Truncated Control Sequence

$$\mathbf{U}_i := \{\mathbf{u}_i, \ldots, \mathbf{u}_{T-1}\}$$

Bellman's Principle of Optimality: find overall optimal control as sequence minimization for each truncated control sequence backwards in time (cost-to-go)

$$V(\mathbf{x}_i) = \min_{\mathbf{u}_i} \Big[\underbrace{\mathcal{L}(\mathbf{x}_i, \mathbf{u}_i) + V(\mathbf{x}_{i+1})}_{Q(\mathbf{x}_i, \mathbf{u}_i)} \Big]$$

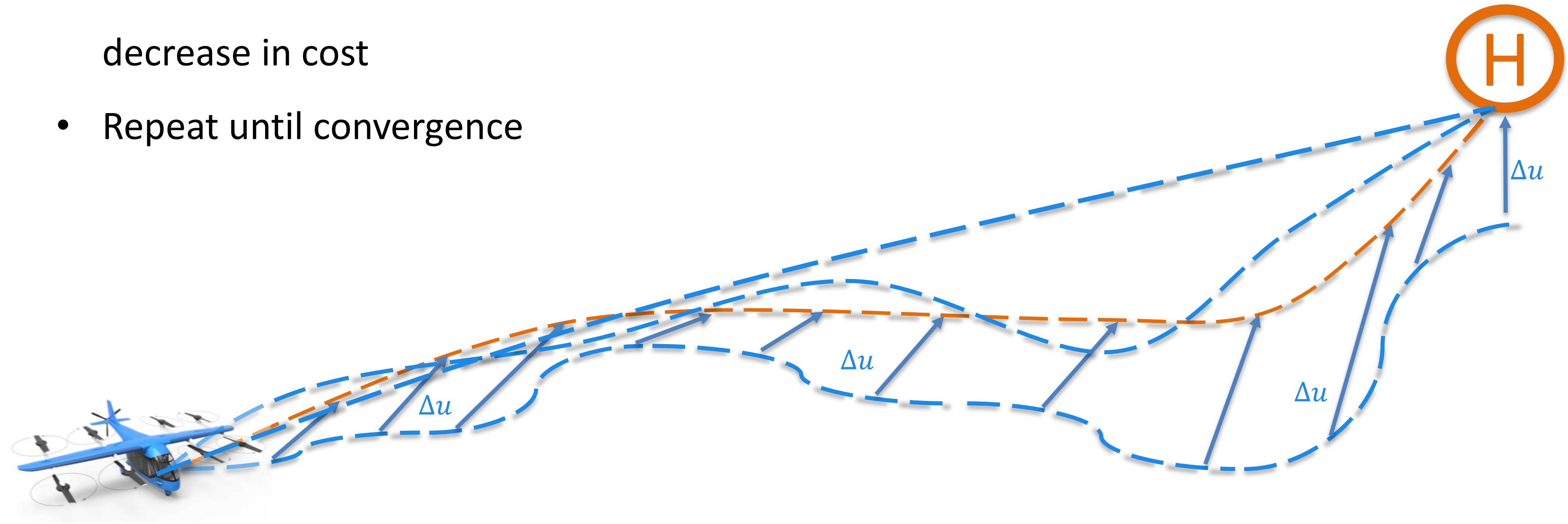
Irene.M.Gregory@nasa.gov

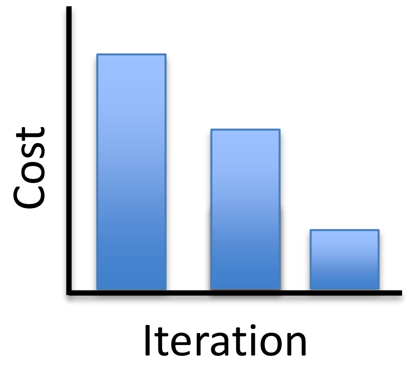
Differential Dynamic Programming

- Given: Initial state x_0, nominal control trajectory $\mathbf{u}_{0:T-1}$
- Repeat until convergence:
  - Forward pass:
    - Forward simulate dynamics from x_0 using $\mathbf{u}_{0:T-1}$ to get state trajectory $\mathbf{x}_{0:T}$
    - Compute derivatives of dynamics $f, f_x, f_u, f_{xx}, f_{xu}, f_{ux}, f_{uu}$ at each time t
    - Compute costs and derivatives $\ell, \ell_x, \ell_u, \ell_{xx}, \ell_{xu}, \ell_{ux}, \ell_{uu}$ at each time t
  - Backward pass:
    - Compute quadratic value function expansion $Q, Q_x, Q_u, Q_{xx}, Q_{xu}, Q_{ux}, Q_{uu}$ at each time t
    - Compute feedforward and feedback gains k and K at each time t
    - Update control $u_t \leftarrow u_t + \alpha k_t + K_t \delta x_t$
      - $\alpha \in [0, 1]$ is a learning rate that is tuned
      - Perform line search on α to heuristically find the best learning rate

Differential Dynamic Programming

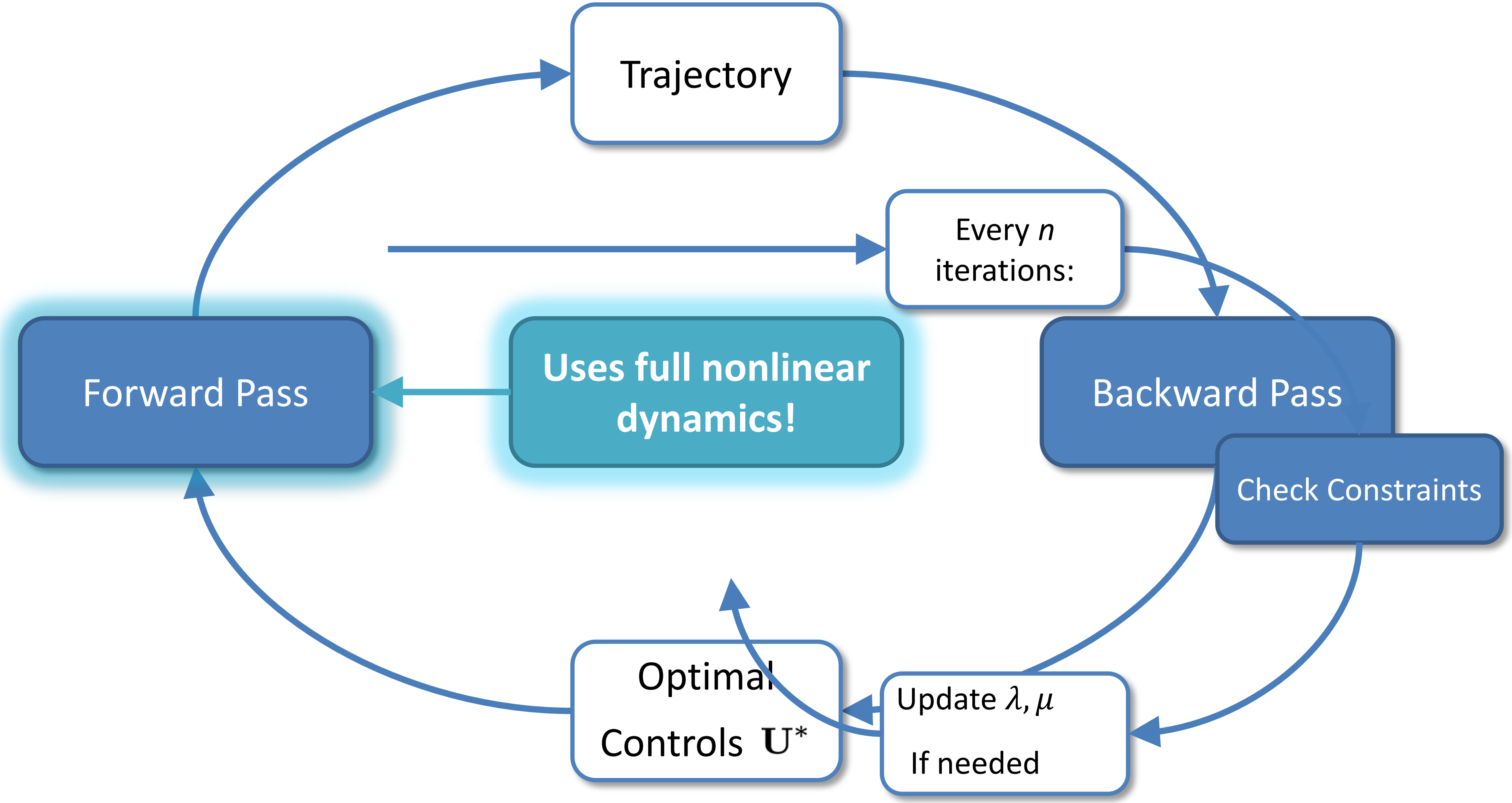
- Apply nonlinear dynamics to initial trajectory, $\boldsymbol{x_0}, \boldsymbol{u}$
- Find controls that minimize expected cost using approx. of cost and dynamics, $\Delta\boldsymbol{u}$
- Apply updated controls to determine if new trajectory leads to a decrease in cost
- Repeat until convergence





Irene.M.Gregory@nasa.gov

AL-DDP Depiction



Irene.M.Gregory@nasa.gov

DDP and AL-DDP

Differential Dynamic Programming (DDP):

- Given nominal trajectory, use linear (or quadratic) approximation of system nonlinear dynamics and quadratic approximation of cost, yields updates that quadratically converge
- Standard DDP does not handle state or control constraints

Augmented Lagrangian DDP (AL-DDP):

- Adds state constraints to the original optimal control problem
- Convert single constrained problem into series of unconstrained problems using penalty functions
- Optimization for state constraints greatly **increases computational complexity**, requires an **inner and outer loop**

$$\min_{\mathbf{U}} \mathcal{J}(\mathbf{U}) = \min_{\mathbf{U}} \sum_{t=0}^{T-1} \mathcal{L}(\mathbf{x}_t, \mathbf{u}_t) + \phi(\mathbf{x}_T),$$

Irene.M.Gregory@nasa.gov

DDP and AL-DDP

Differential Dynamic Programming (DDP):

- Given nominal trajectory, use linear (or quadratic) approximation of system nonlinear dynamics and quadratic approximation of cost, yields updates that quadratically converge
- Standard DDP does not handle state or control constraints

Augmented Lagrangian DDP (AL-DDP):

- Adds state constraints to the original optimal control problem
- Convert single constrained problem into series of unconstrained problems using penalty functions
- Optimization for state constraints greatly **increases computational complexity**, requires an **inner and outer loop**

$$\min_{\mathbf{U}} \tilde{\mathcal{J}}(\mathbf{U}) = \min_{\mathbf{U}} \sum_{t=0}^{T-1} \tilde{\mathcal{L}}(\mathbf{x}_t, \mathbf{u}_t, \lambda_t, \mu_t) + \tilde{\phi}(\mathbf{x}_T, \lambda_T, \mu_T),$$

$$\text{subject to } \mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t),\ \forall t = 0, \ldots, T-1,$$

where

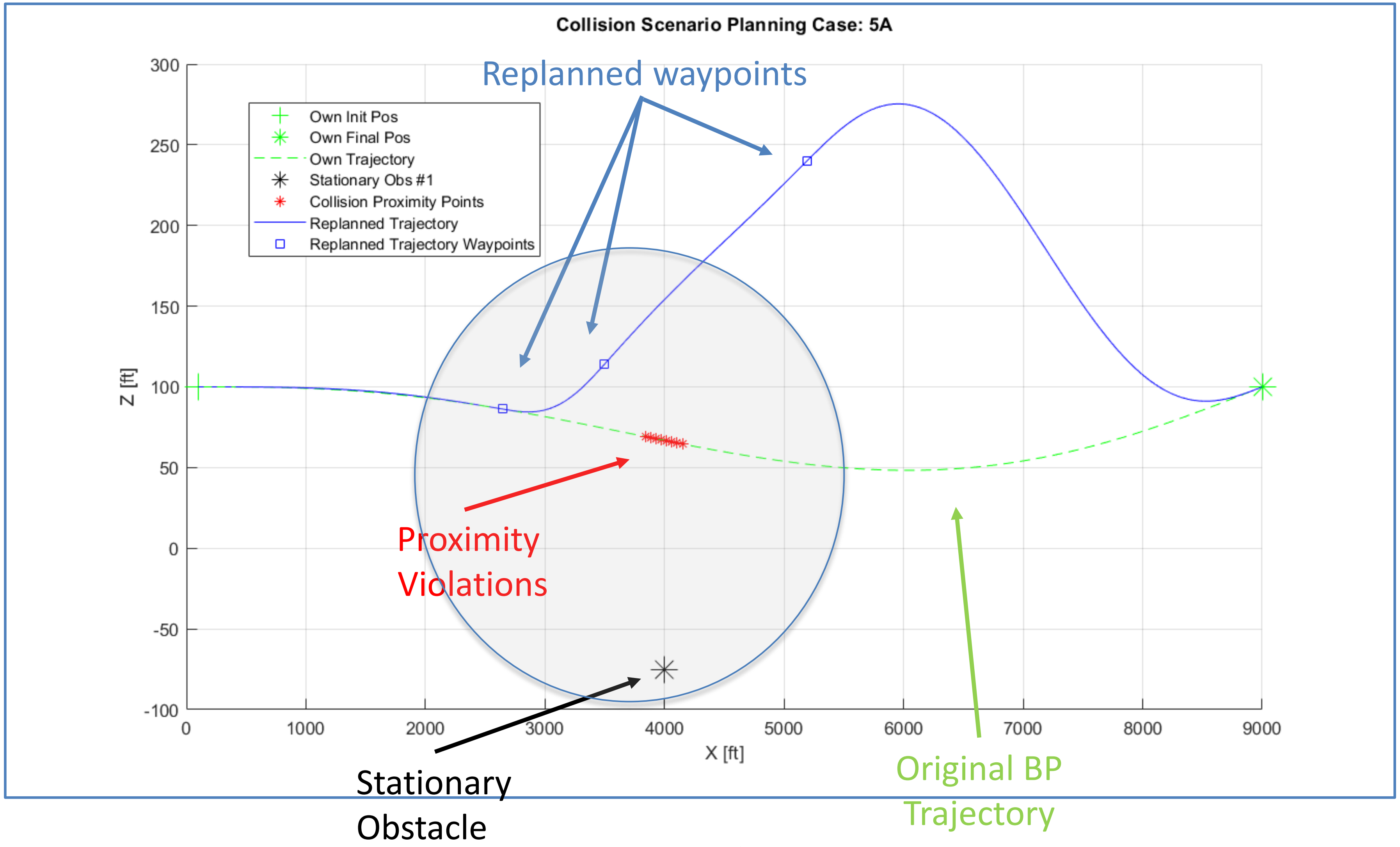
$$\tilde{\mathcal{L}}(\mathbf{x}_t, \mathbf{u}_t, \lambda_t, \mu_t) = \mathcal{L}(\mathbf{x}_t, \mathbf{u}_t) + \sum_{i=1}^{c} \mathcal{P}(g_{t,i}(\mathbf{x}), \lambda_{t,i}, \mu_t),$$

$$\tilde{\phi}(\mathbf{x}_T, \lambda_T, \mu_T) = \phi(\mathbf{x}_T) + \sum_{i=1}^{c} \mathcal{P}(g_{T,i}(\mathbf{x}), \lambda_{T,i}, \mu_T).$$

Irene.M.Gregory@nasa.gov

Integration: ORCA + Bernstein Polynomials

Goal: Modify own-ship BP trajectory to ensure smooth collision avoidance



Irene.M.Gregory@nasa.gov

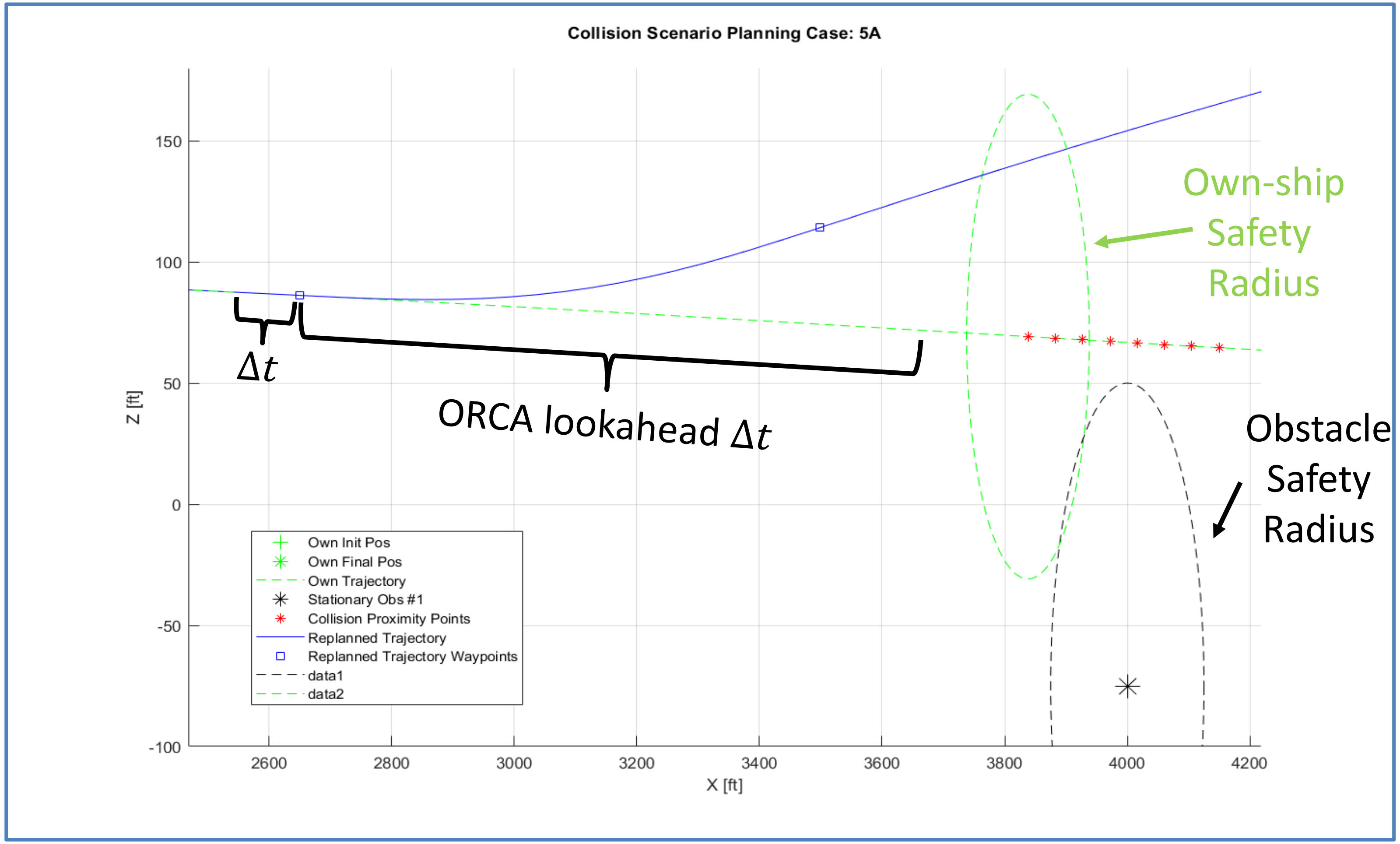
Integration: ORCA + Bernstein Polynomials

Goal: Modify own-ship BP trajectory to ensure smooth collision avoidance

Given:

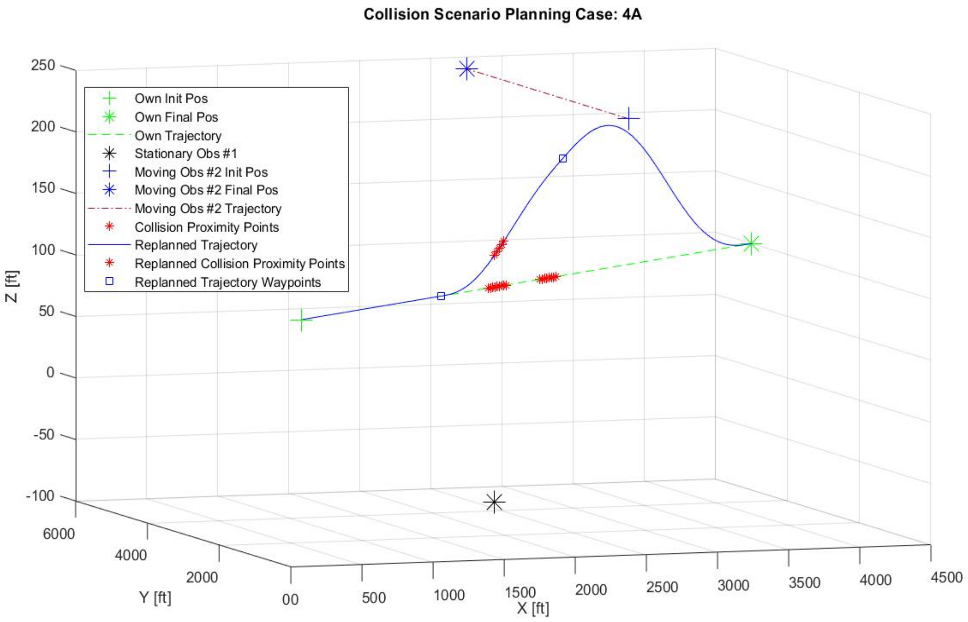
- **Own-ship BP trajectory**
- **Obstacle BP trajectory (estimated)**

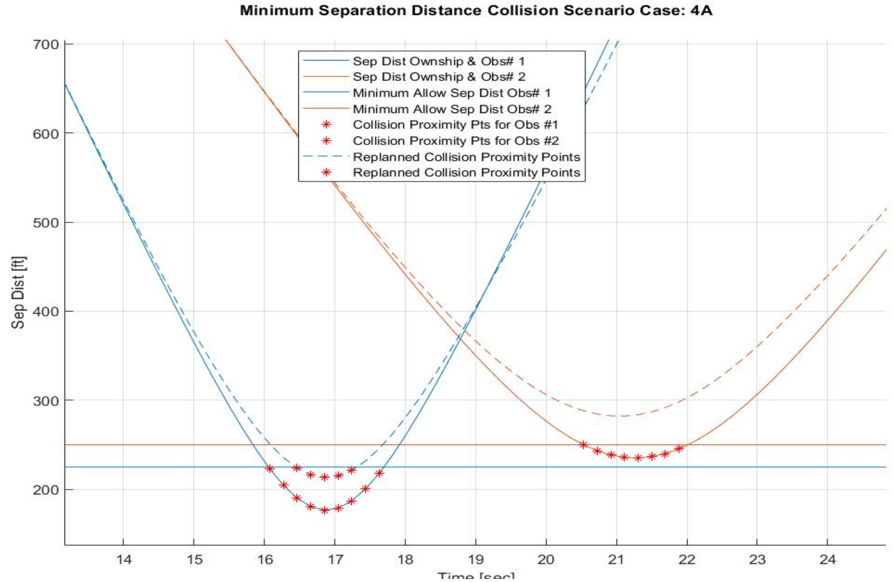
- Perform ORCA collision detection using Δt steps (one-time pass) along own-ship trajectory
- Pending collision detected, use ORCA output velocity to create new waypoint, then continue searching
- ORCA waypoints are used to create a BP curve, smoothing out the trajectory

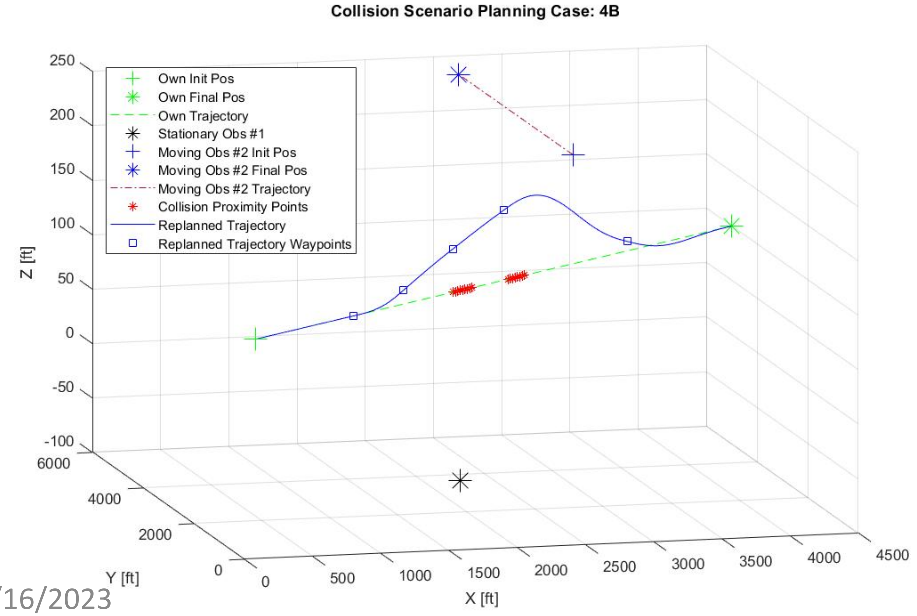


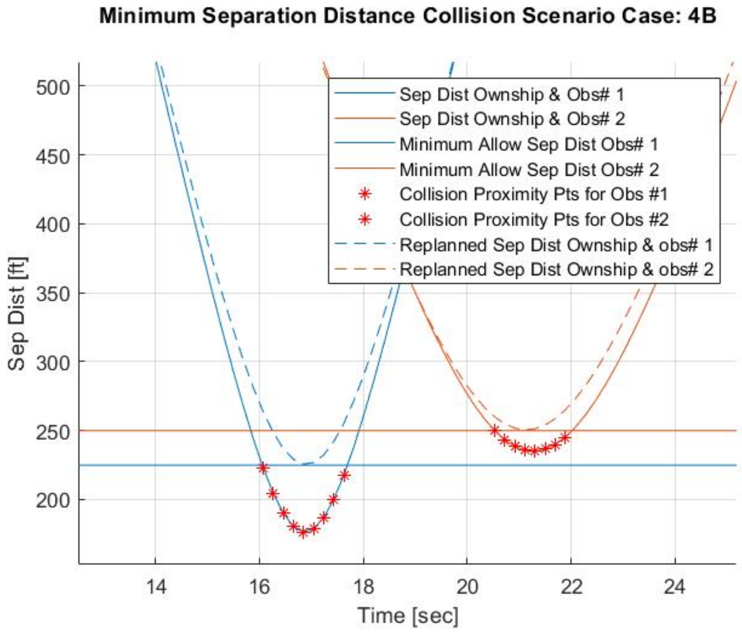
Irene.M.Gregory@nasa.gov

Augmentations to ORCA



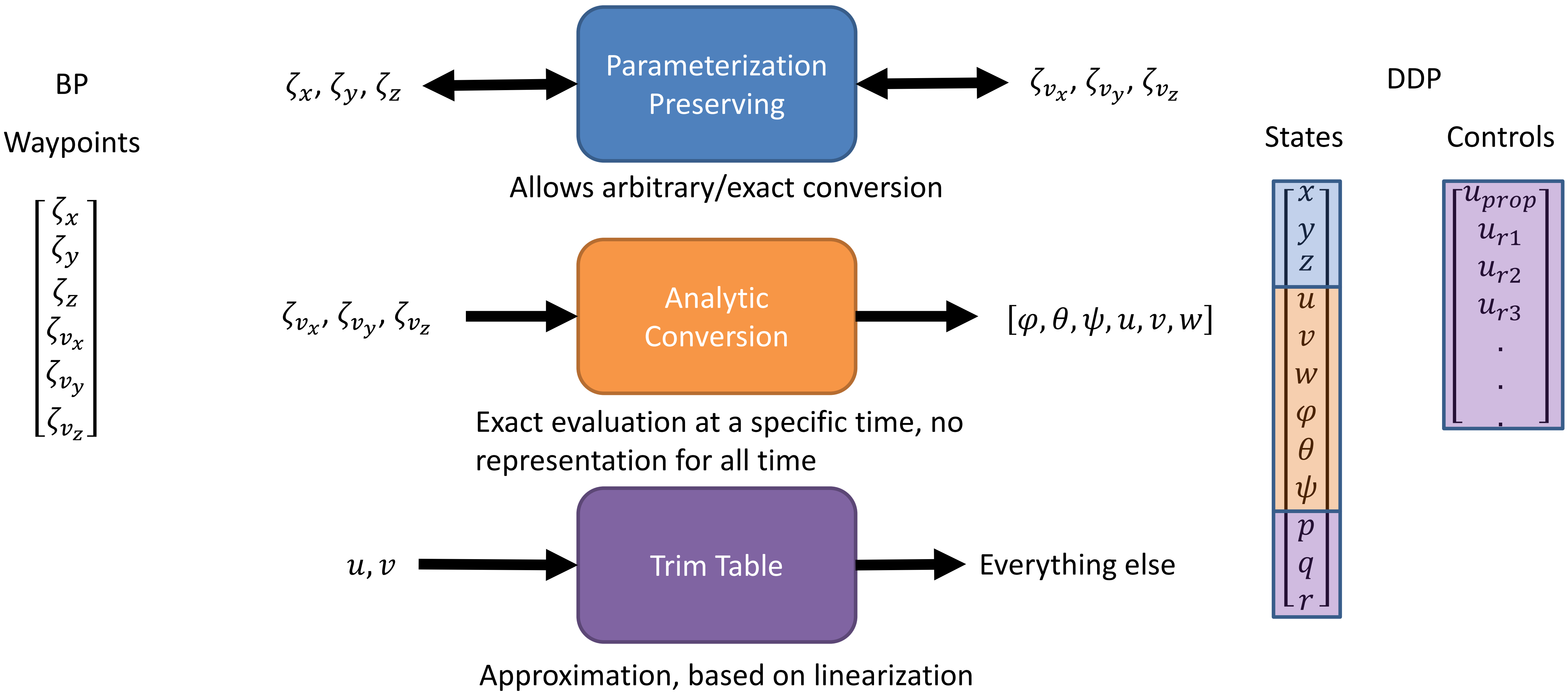






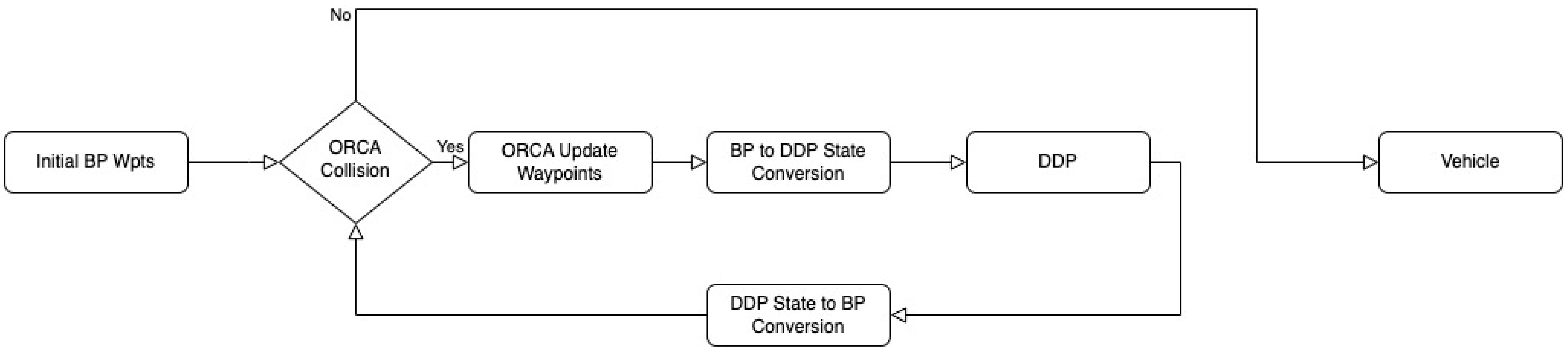
Irene.M.Gregory@nasa.gov

Integration: Bernstein Polynomials + DDP



Irene.M.Gregory@nasa.gov

Algorithm Implementation

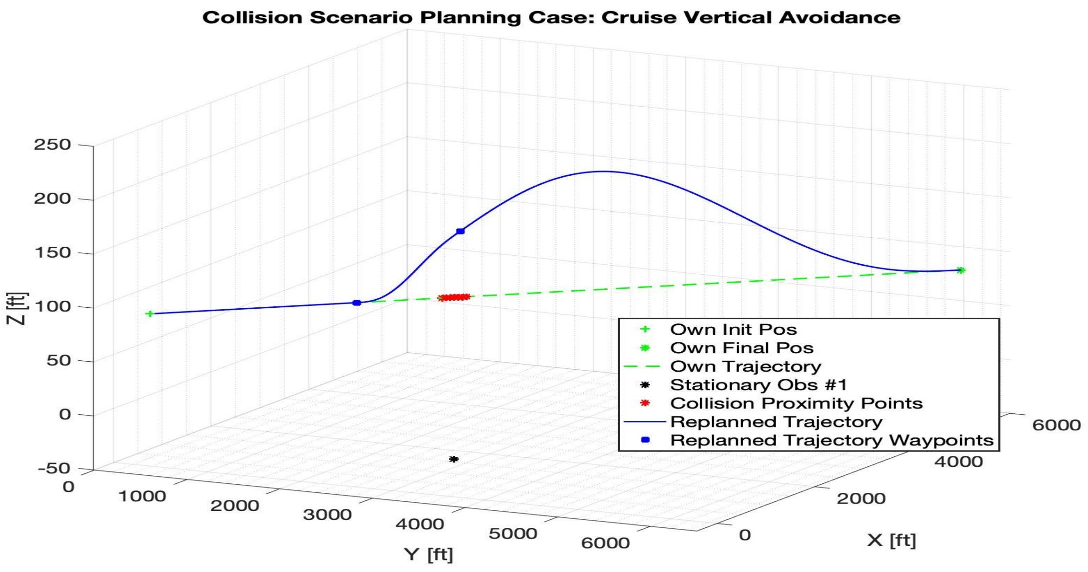


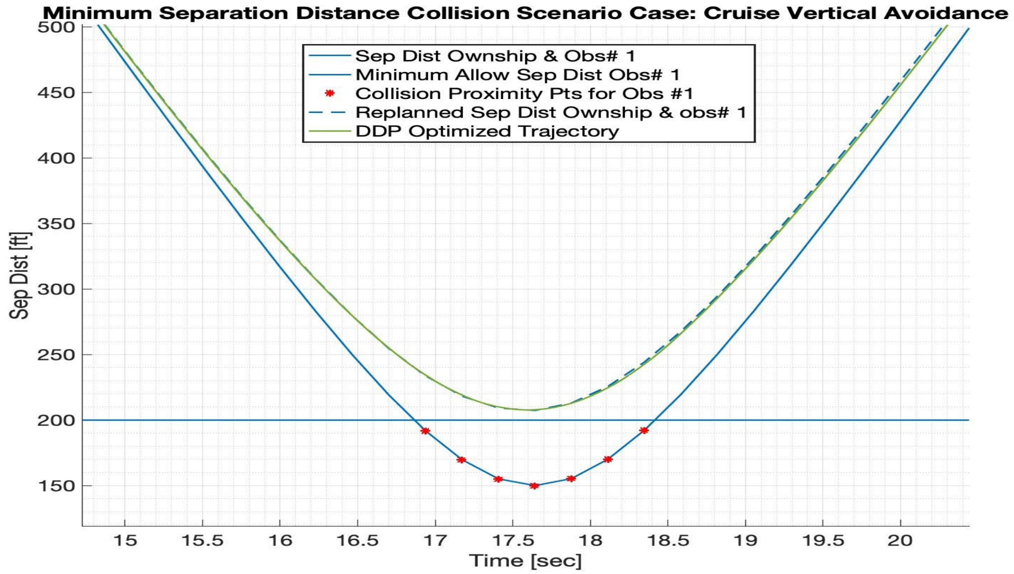
- Leverages ORCA fast collision avoidance checks and preferential avoidance direction selection
- BP's serve as compact trajectory representation between ORCA and DDP that can be quickly evaluated at any time along the curve
- DDP provides dynamically feasible optimal trajectories given simplified ORCA information

Irene.M.Gregory@nasa.gov

Experiments: Cruise to Altitude Change

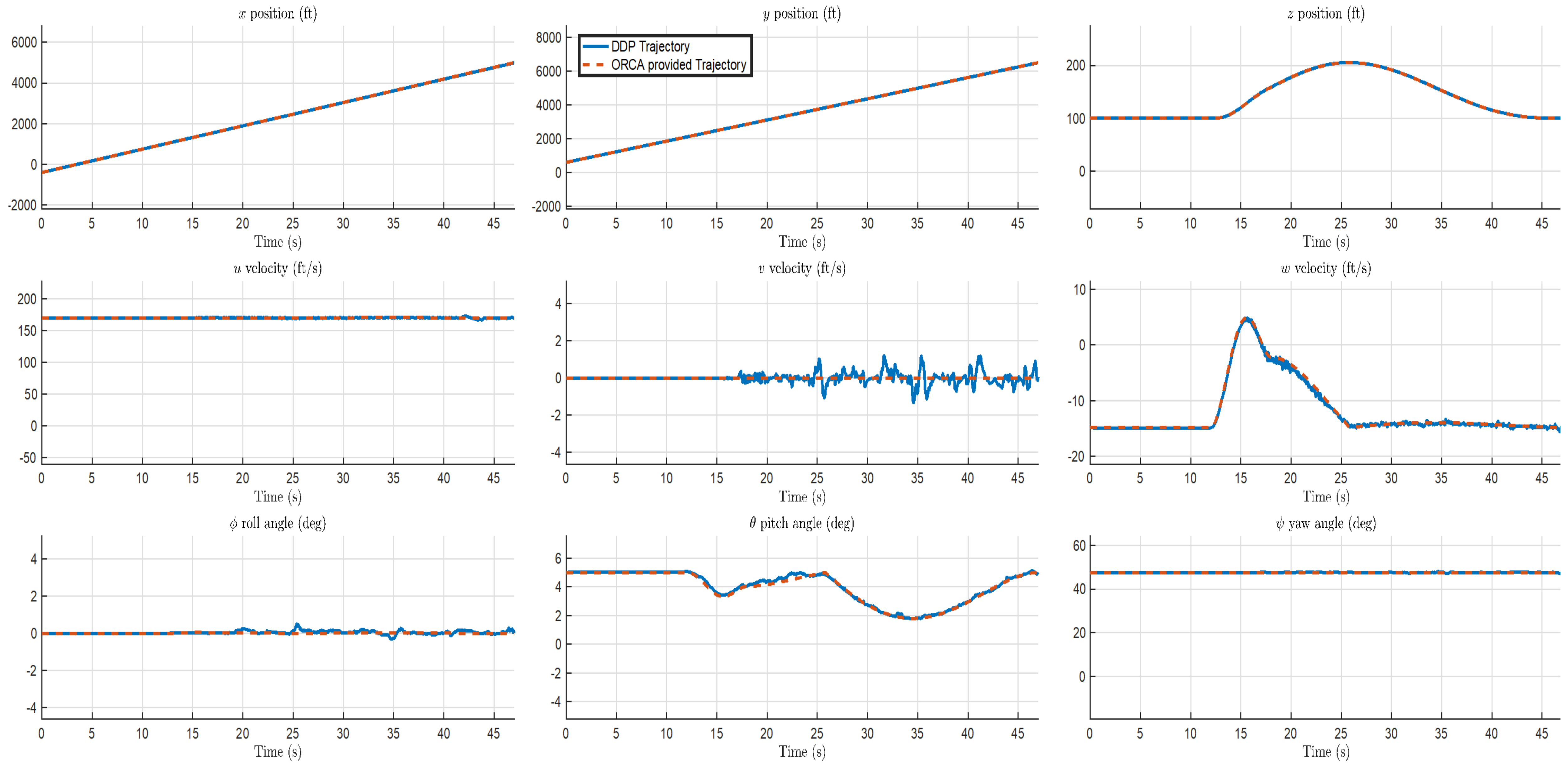
- 170 ft/s cruise with 200 ft safety radius
- Static obstacle 150 ft below
- Recognizes safety breach
- ORCA recalculates, maintains safety radius
- ORCA waypoints converted into BP curves
- BP curves integrated with trim knowledge and passed to DDP
- DDP optimized new trajectory avoids obstacle





Irene.M.Gregory@nasa.gov

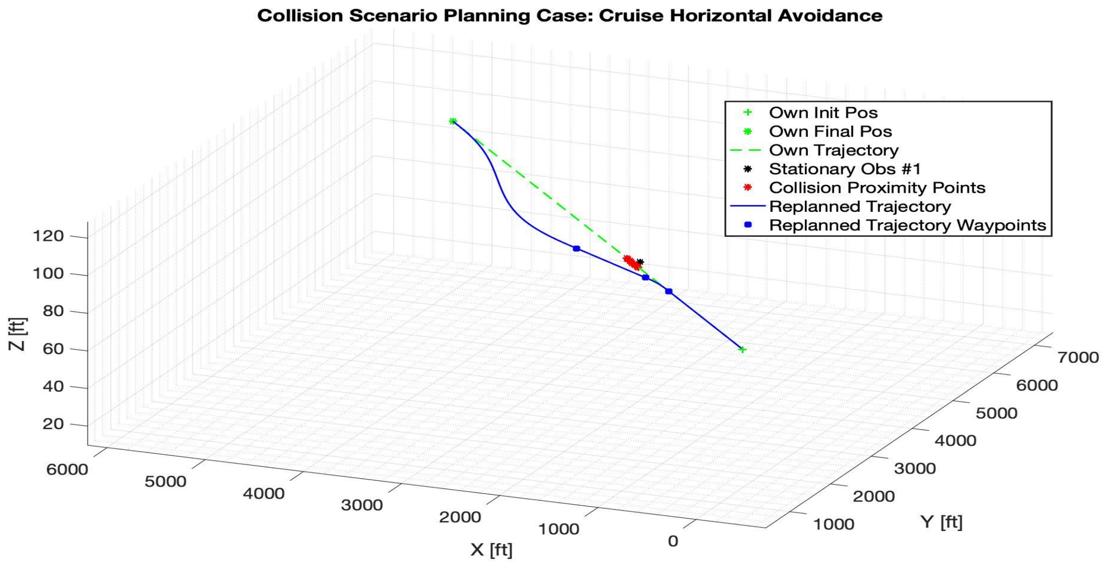
Cruise to Altitude Change

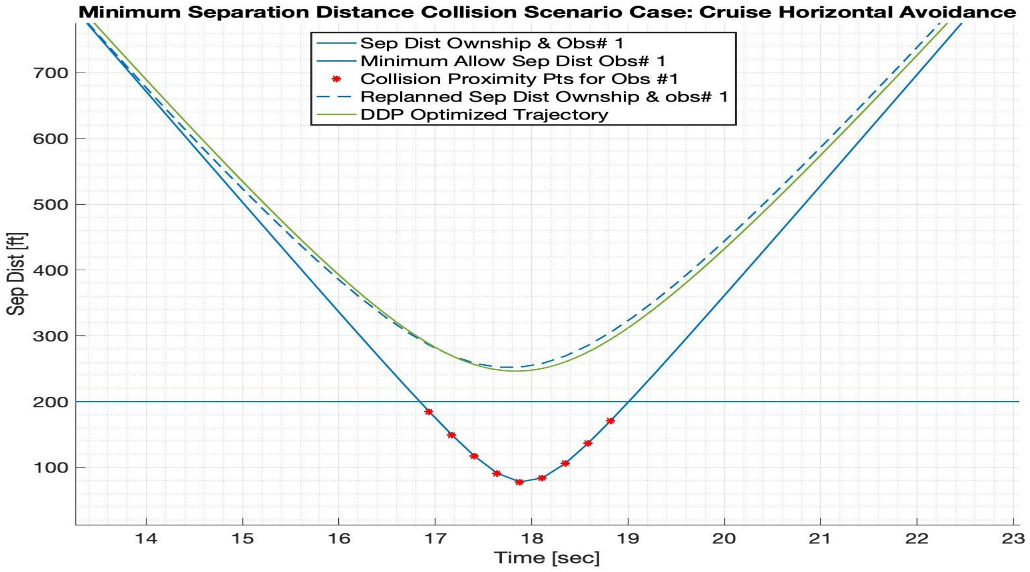


Irene.M.Gregory@nasa.gov

Cruise to Horizontal Maneuver

- 170 ft/s cruise with 200 ft safety radius
- Static obstacle 75 ft to the right
- Recognition of safety radius breach
- ORCA recalculates, maintains safety radius
- ORCA waypoints converted into BP curves
- BP curves integrated with trim knowledge and passed to DDP
- DDP optimized new trajectory avoids obstacle

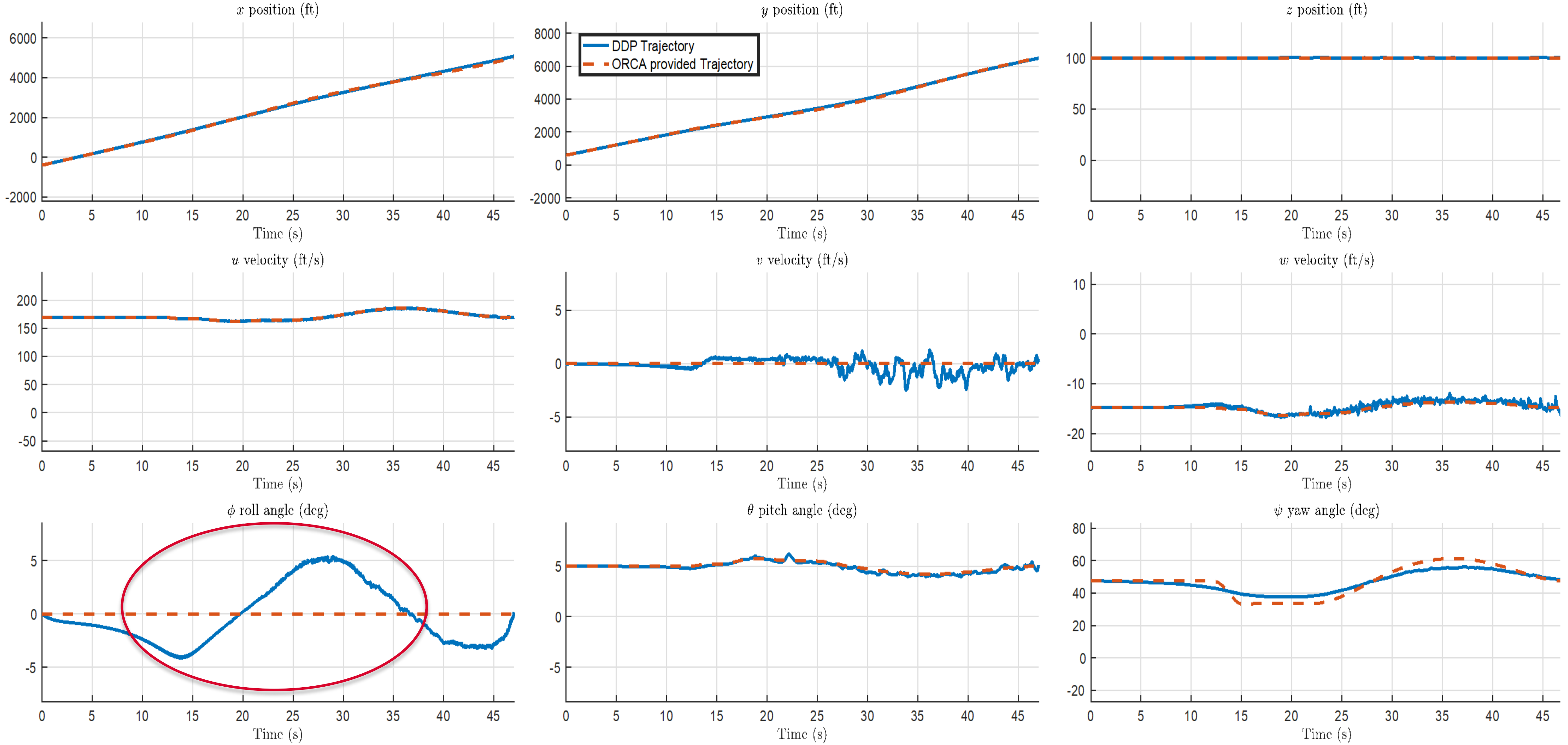




Irene.M.Gregory@nasa.gov

Cruise to Horizontal Maneuver

DDP was provided wing-level cruise trim values for suggested trajectory

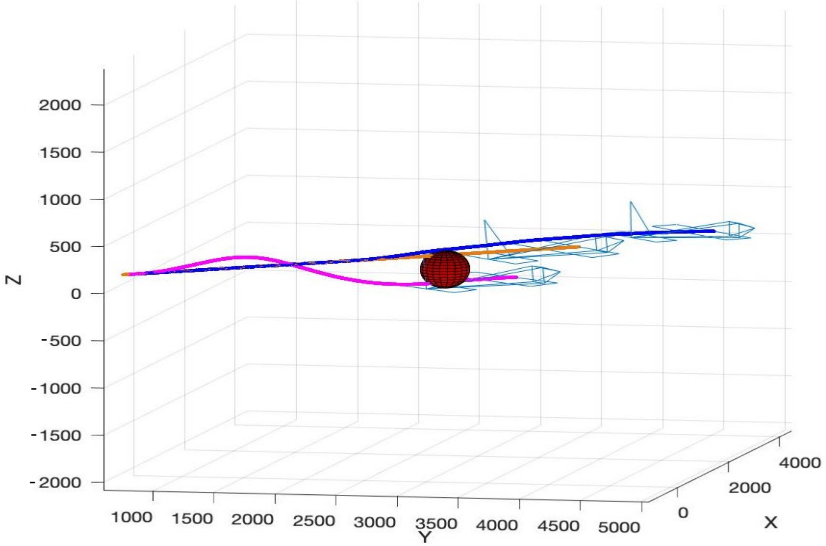


Irene.M.Gregory@nasa.gov

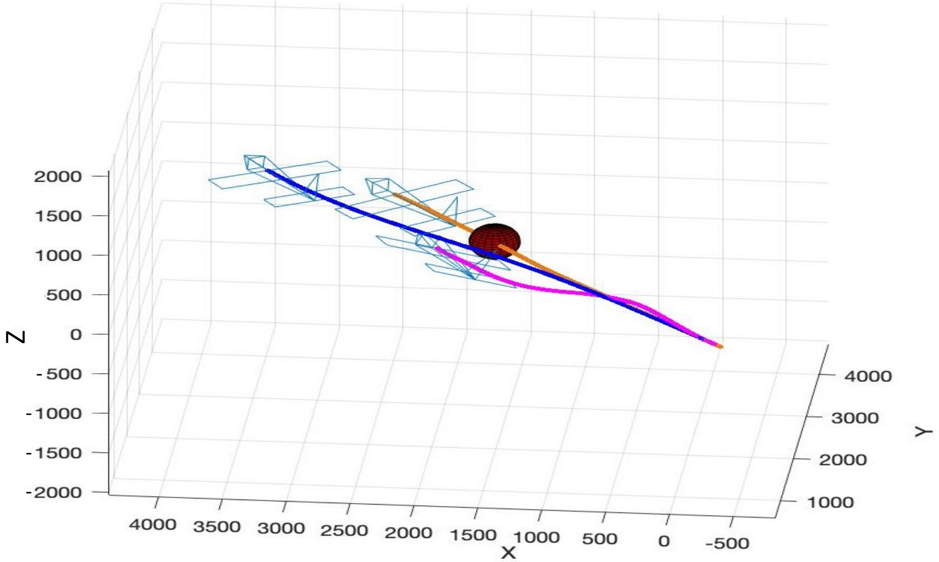
AL-DDP Warmstarting with COBRA-DDP

AL-DDP Warmstart Experiments

- Provided cruise initial trajectory:
  - AL-DDP **Fails** (orange trajectories)
- Provided no warmstart initial trajectory:
  - AL-DDP **always** avoids under obstacle (magenta trajectories)
- Provided COBRA-DDP trajectory:
  - **Successful** replanning and avoids collision (blue trajectories)
- COBRA-DDP's trajectory provides a feasible warmstart trajectory to help minimize computation of AL-DDP
- COBRA-DDP enables selection of desired velocity, allowing for the set up of potential flight rules or expected avoidance patterns (e.g. left to left)



Cruise Altitude Change Experiment



Cruise Horizontal Turn Experiment

Irene.M.Gregory@nasa.gov

Conclusions and Ongoing Work:

Conclusion:

- BP curves serve as an effective transfer of trajectory information between ORCA and DDP
- COBRA-DDP plans dynamically feasible collision avoiding trajectories and can select preferential avoidance direction
- COBRA-DDP can enhance the optimization of other state-constrained optimizers such as AL-DDP

Irene.M.Gregory@nasa.gov

Further Details on Future Work

Ongoing Work:

- MPC implementation of COBRA-DDP
- Demonstration of collision avoidance with multiple moving/stationary, cooperative/uncooperative obstacles
- Further ORCA augmentation to consider complex vehicle maneuvers
- What advantages are available to vehicles that communicate to each other using BP curves?

Irene.M.Gregory@nasa.gov

Comparison of Acoustic Models and Trajectory Generation Methods for an Acoustically-Aware Aircraft*

Kasey A. Ackerman and Irene M. Gregory

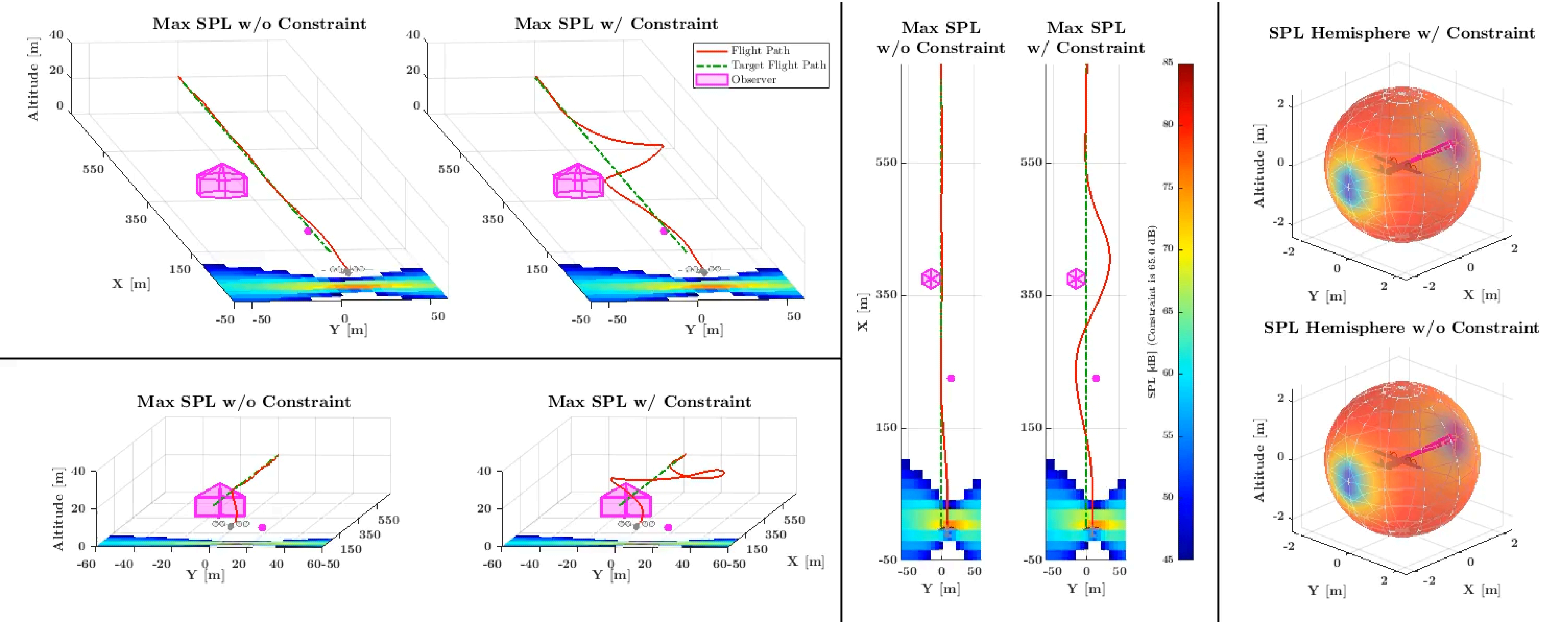
NASA Langley Research Center

Hampton, VA 23681

*Ackerman, Kasey J., Gregory, Irene M., "Comparison of Acoustic Models and Trajectory Generation Methods for an Acoustically-Aware Aircraft," 2023 AIAA SciTech Forum, National Harbor, MD, January 2023. AIAA-2023-2543

Noise Model Comparison

- Hemisphere model



Questions?

Contact Information:

Irene.M.Gregory@nasa.gov

matthew.d.houghton@nasa.gov

kasey.a.ackerman@nasa.gov

Acknowledgements: NASA Aeronautics Research Mission Directorate (ARMD)

- Transformative Tools and Technologies (TTT) project
  Revolutionary Air Mobility / Autonomous Systems / Intelligent Contingency Management (ICM)
- Revolutionary Vertical Lift Technologies (RVLT) project

Parameterized Differential Dynamic Programming (PDDP)

PDDP is a trajectory optimization algorithm that builds upon DDP

- Enables the **co-optimization** of a **trajectory** and time invariant **parameters** in the same process.
- Parameters can be extremely diverse and goal specific
- Experiments tested PDDP's ability to successfully **estimate vehicle dynamic parameters** while implementing **optimal trajectories**, resulting in Adaptive Model Predictive Control

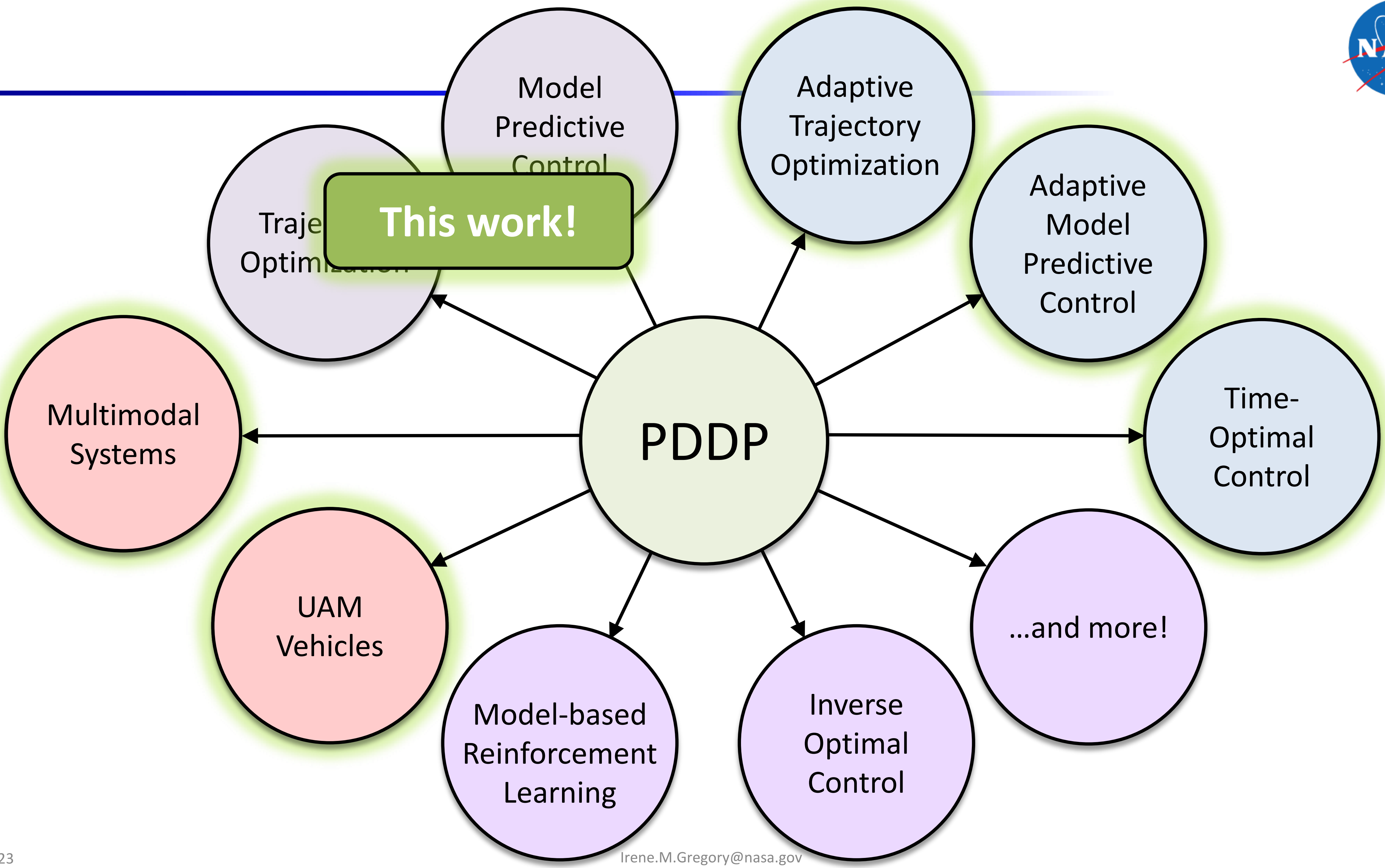
Switching Time Optimization

- Calculation of **optimal transition time** between flight regimes (Difficult for highly nonlinear vehicles like L+C)
- **Decreases tuning** work for engineers when planning for common maneuvers that transition between flight regimes (Vertical takeoff into fixed-wing cruise)
- Allows for the input and optimization of multiple target states for long-term planning and replanning

Fault Detection

- Online estimation of vehicle **dynamic** parameters
- **Online estimation** of **degradation** level for effectors + rotors
- **Replan trajectory** based on new estimation of vehicle parameters
- Deviations in estimation from norms can alert system ID of vehicle to run further diagnostics of vehicle health

Irene.M.Gregory@nasa.gov



Irene.M.Gregory@nasa.gov

Brief Overview of DDP and PDDP

Differential Dynamic Programming:

- Given nominal trajectory, use linear (or quadratic) approx. of system nonlinear dynamics and quadratic approx. of cost to yield updates to optimal controls that quadratically converge

Parametric Differential Dynamic Programming:

- Discrete system with nonlinear dynamics
- θ represents time-invariant system parameter(s)
- Goal is now to minimize the cost function with respect to both the controls, u and the parameters, θ
  - Estimation of unknown parameters and states of a dynamical system through Moving Horizon Estimation (MHE)
  - Initial parameters are set for a dynamical system, θ and for this example do not match the real system
  - The vehicle applies a portion of the trajectory given these initial parameters using a typical MPC cost
  - The resulting trajectory taken is fed into the estimation cost, which tries to find the correct parameters given the difference between the planned trajectory and what occurred on the real system
  - The new parameters are used to update the model of system. A combined cost can be derived over both task simultaneously using PDDP

Irene.M.Gregory@nasa.gov

Acknowledgements & References

Acknowledgements:

- NASA Aeronautics Research Mission Directorate (ARMD), Transformative Tools and Technologies (TTT) project, under the Revolutionary Air Mobility / Autonomous Systems / Intelligent Contingency Management (ICM)

References:

- Farouki, R. T., “The Bernstein polynomial basis: A centennial retrospective,” Computer Aided Geometric Design, Vol. 29, No. 6, 2012, pp. 379–419.
- Jacobson, D. H., and Mayne, D. Q., Differential Dynamic Programming, American Elsevier Pub. Co., 1970.
- Birgin, E. G., Castillo, R. A., and Martinez, J. M., “Numerical comparison of augmented Lagrangian algorithms for nonconvex problems,” Computational Optimization and Applications, Vol. 31, No. 1, 2005, pp. 31–55.
- van den Berg, J., Guy, S. J., Lin, M., and Manocha, D., “Reciprocal n-body Collision Avoidance,” In Proc. of the IEEE International Conference on Robotics and Automation, 2010.
- Silva, C., Johnson, W. R., Solis, E., Patterson, M. D., and Antcliff, K. R., “VTOL urban air mobility concept vehicles for technology development,” Aviation Technology, Integration, and Operations Conference, 2018.

Irene.M.Gregory@nasa.gov