Real-Time Atmospheric Entry Trajectory Computation Using Parametric Sensitivities

6th International Conference on Astrodynamics Tools and Techniques – ID 121

David Seelbinder
German Aerospace Center
Institute of Space Systems
✉ david.seelbinder@dlr.de

Prof. Christof Büskens
University Bremen
Center for Technomathematics
✉ bueskens@math.uni-bremen.de

University Bremen

DLR

Guided Entry Phase

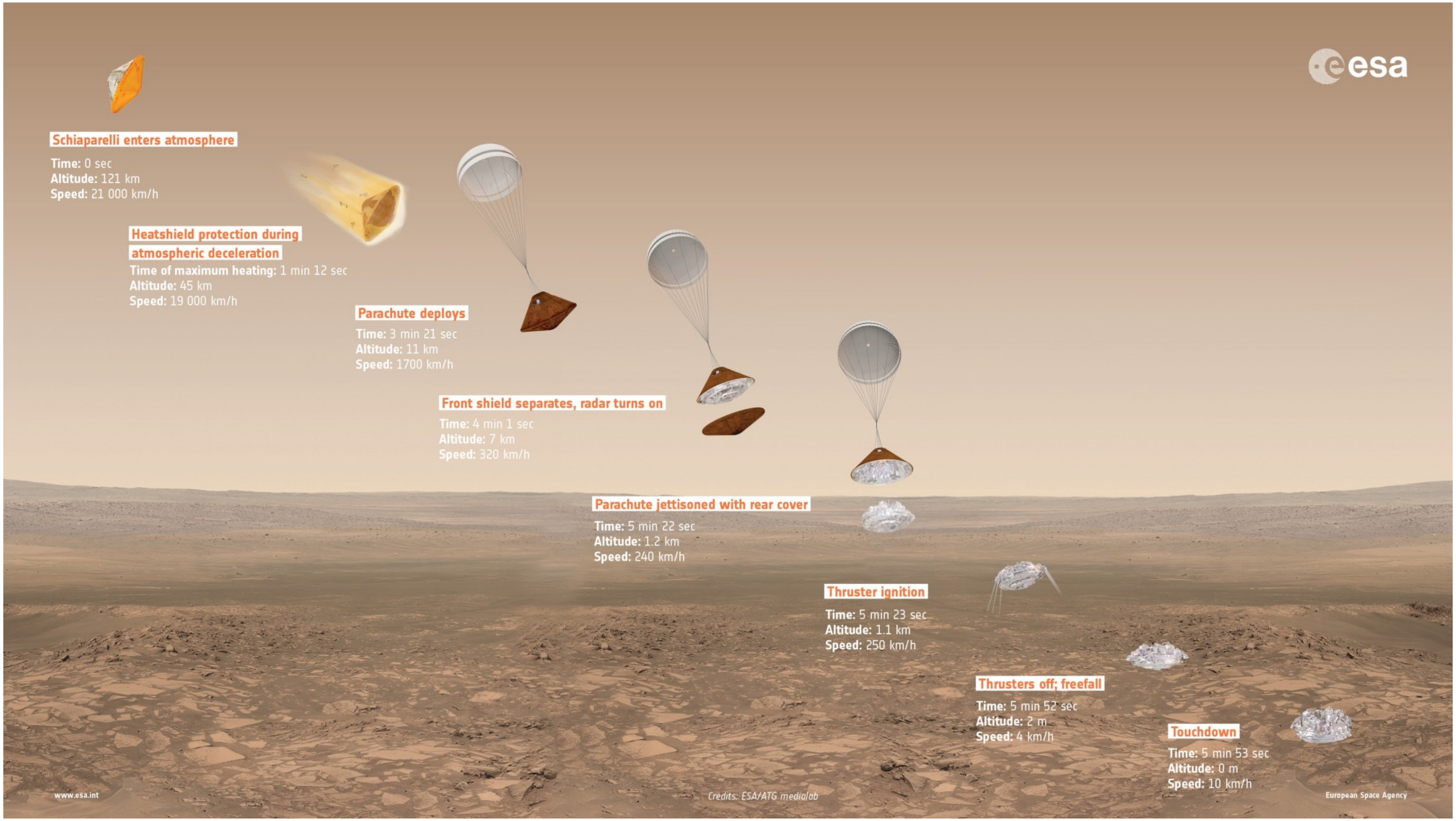


Image credit: ESA

Guided Entry Phase

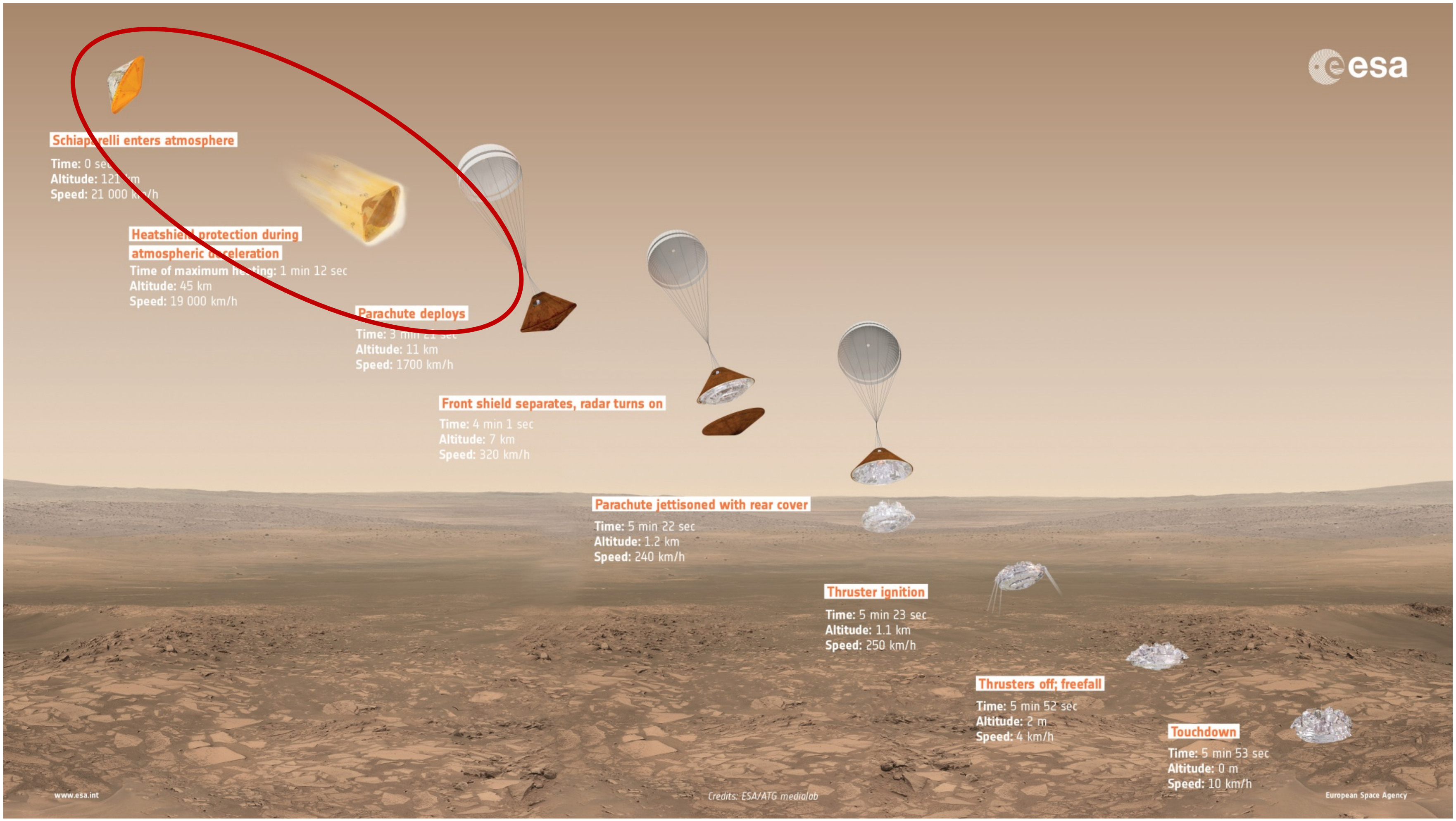


Image credit: ESA

Agenda

- Trajectory computation
  - Offline sensitivity analysis
  - Online solution approximation
- Two-Degree of Freedom Guidance System
- Results of Monte Carlo Campaign
- Results of Processor-in-the-loop test
- Summary

Offline Process: Problem Formulation and Transcription

OCP(p)

Formulation as parametric Optimal Control Problem:

$$\min_z \quad J(x,u,p) = g\big(x(t_0), x(t_f), p\big) + \int_{t_o}^{t_f} l(x(t), u(t), p)\, dt$$

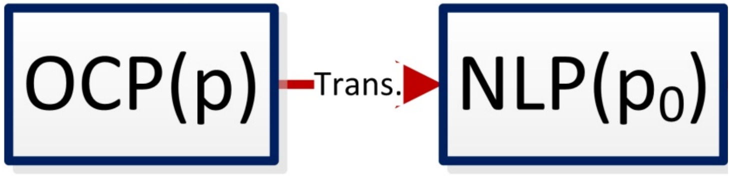
$$\text{w.r.t.} \quad \dot{x}(t) = f(x(t), u(t), p)$$

$$\psi_0(x(t_0), p) = 0$$

$$\psi_f\big(x(t_f), p\big) = 0$$

$$C(x(t), u(t), p) \leq 0 \qquad t \epsilon [t_0, t_f]$$

Offline Process: Problem Formulation and Transcription



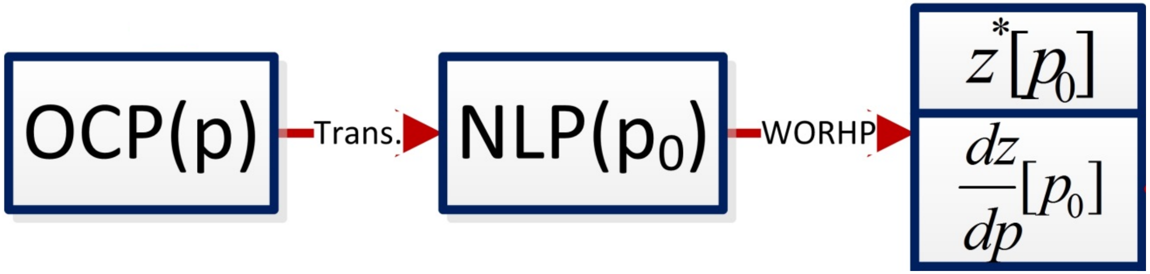
- Direct optimization
  - Discretization (Runge-Kutta-4, Linear control interpolation)
  - Transcription into a parametric Nonlinear Program (shooting technique)

$$\min_z \quad F(z, p_0)$$

$$\text{w.r.t.} \quad g_l \leq G(z, p_0) \leq g_u$$

- Choice of a nominal parameter set p_0

Offline Process: Sensitivity Analysis I



- Obtain nominal optimal solution $z_0 = z^*(p_0)$
- Let z_0 fulfill strong second order sufficient conditions
- Karush-Kuhn-Tucker-matrix is invertible
- Sensitivity differentials (SD) are computable

$$\begin{bmatrix} \frac{\boldsymbol{dz}}{\boldsymbol{dp}}[\boldsymbol{p_0}] \\ \frac{d\eta^a}{dp}[p_0] \end{bmatrix} = - \begin{bmatrix} \nabla_z{}^2 L(z_0, \eta_0^a, p_0) & \nabla_z G^a(z_0, p_0)^T \\ \nabla_z G^a(z_0, p_0) & 0 \end{bmatrix}^{-1} \begin{bmatrix} \nabla_{zp}{}^2 L(z_0, \eta_0^a, p_0) \\ \nabla_p G^a(z_0, p_0) \end{bmatrix}$$

Online Solution Approximation: p-Step

- Parameter vector p
  - Perturbation of the initial condition ψ_0
  - Perturbation of model parameters: mass, aerodynamic coefficients

- Approximation of optimal solution for disturbed parameters p using Taylor expansion

$$z^*(p) \approx z_1 \coloneqq z_0 + \frac{dz}{dp}[p_0] \cdot (p - p_0)$$

Online Solution Approximation: p-Step

- Parameter vector p
  - Perturbation of the initial condition ψ_0
  - Perturbation of model parameters: mass, aerodynamic coefficients

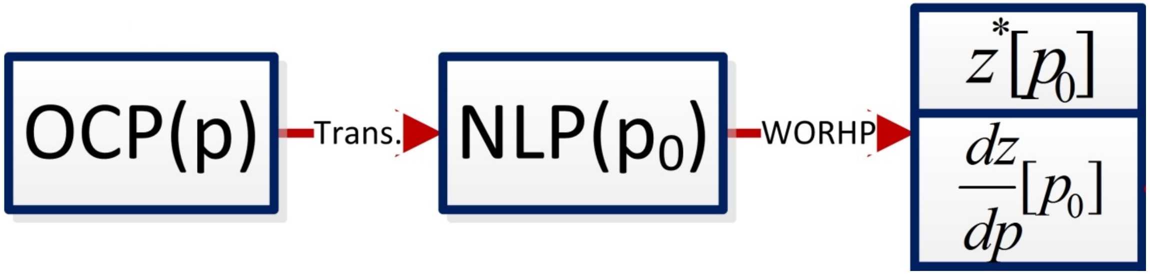
- Approximation of optimal solution for disturbed parameters p using Taylor expansion

$$z^*(p) \approx z_1 \coloneqq z_0 + \frac{dz}{dp}[p_0] \cdot (p - p_0)$$

➡ error in the active constraints

$\|G^a(z_i, p)\| > 0$

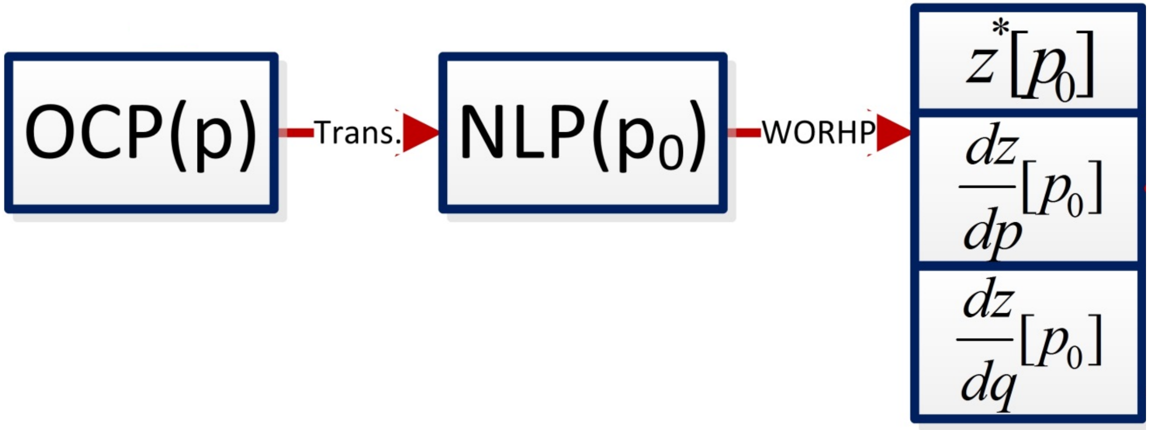
Offline Process: Sensitivity Analysis II



$$\min_z \quad F(z, p_0)$$

$$\text{w.r.t.} \quad g_l \leq G(z, p_0) \leq g_u$$

Offline Process: Sensitivity Analysis II



- Additional parameter vector q with nominal value $q_0 = 0$

$$\min_z \quad F(z, p_0)$$

$$\text{w.r.t.} \quad g_l \leq G(z, p_0) - q_0 \leq g_u$$

- $\frac{dz}{dq}$ can be computed analog to $\frac{dz}{dp}$

Online Solution Approximation: p-Step and q-Step

- Parameter vector p
  - Perturbation of the initial condition ψ_0
  - Perturbation of model parameters: mass, aerodynamic coefficients

- Approximation of optimal solution for disturbed parameters p using Taylor expansion

$$z^*(p) \approx z_1 \coloneqq z_0 + \frac{dz}{dp}[p_0] \cdot (p - p_0)$$

⇒ error in the active constraints $\|G^a(z_i, p)\| > 0$

Online Solution Approximation: p-Step and q-Step

- Parameter vector p
  - Perturbation of the initial condition ψ_0
  - Perturbation of model parameters: mass, aerodynamic coefficients

- Approximation of optimal solution for disturbed parameters p using Taylor expansion

$$z^*(p) \approx z_1 \coloneqq z_0 + \frac{dz}{dp}[p_0] \cdot (p - p_0)$$

→ error in the active constraints $\|G^a(z_i, p)\| > 0$

- $\frac{dz}{dq}$ is used to iteratively correct the constraint error and at the same time improve the order of optimality of the approximation

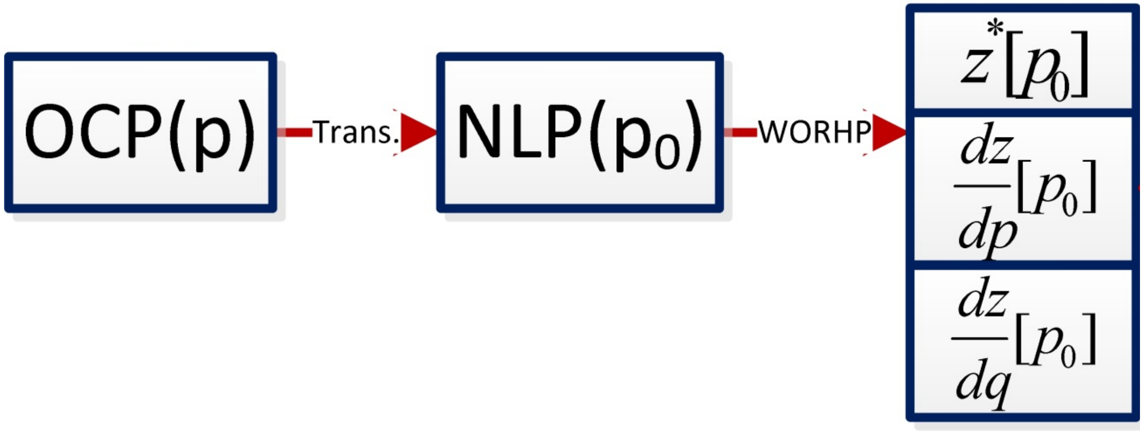
while $\|G^a(z_i, p)\| > \varepsilon$

$$q_i = G^a(z_i, p)$$

$$z_{i+1} \coloneqq z_i + \frac{dz}{dq^a}[p_0] \cdot q_i$$

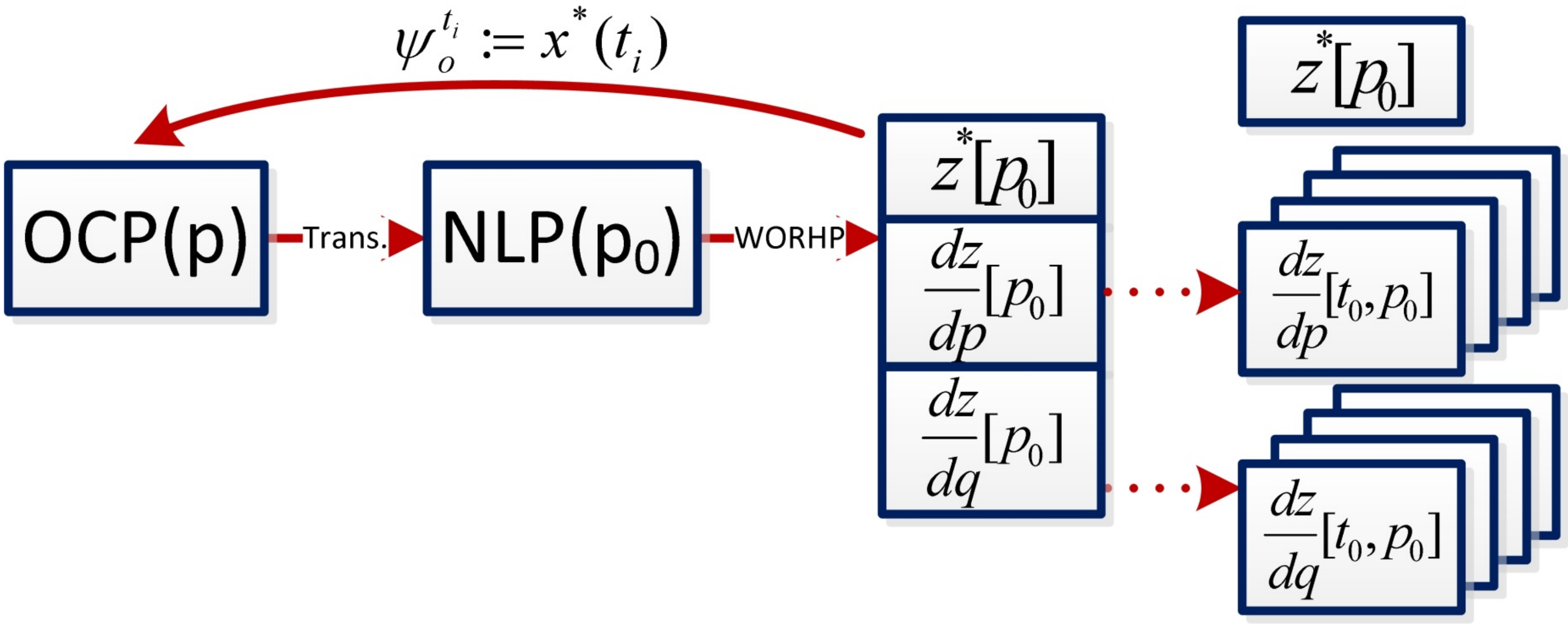
For $\|q - q_0\| < \delta$ iteration converges against a fixpoint z_∞ at which $\|G^a(z_\infty, p)\| = 0$

Offline Process: Build Sensitivity Catalog



- For trajectory computation at time t the sensitivity differentials must be known for the initial condition $\psi_0^t := x^*(t),\ t \in [t_0, t_f)$

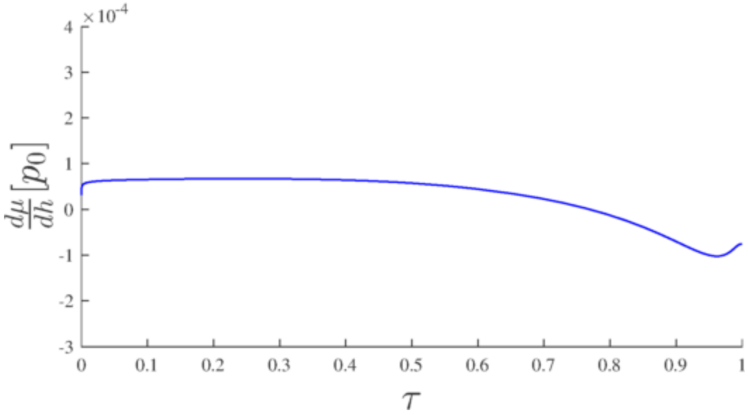
Offline Process: Build Sensitivity Catalog

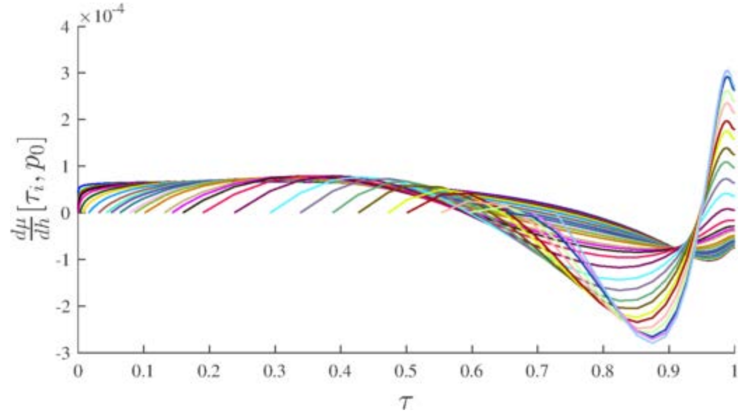


- For trajectory computation at time t the sensitivity differentials must be known for the initial condition $\psi_0^t := x^*(t)$, $t \in [t_0, t_f)$
- Repeat sensitivity analysis at discrete points $t_i \in [t_0, t_f)$, $0 < i \leq l$ of the nominal trajectory $x^*(t)$

Sensitivity on Discrete Points of the Nominal Trajectory

- Example: Sensitivity of μ against perturbations in h at $x^*(t_i)$, $0 \leq i \leq k$





Sensitivity on Discrete Points of the Nominal Trajectory

- Example: Sensitivity of μ against perturbations in h at $x^*(t_i)$, $0 \leq i \leq k$

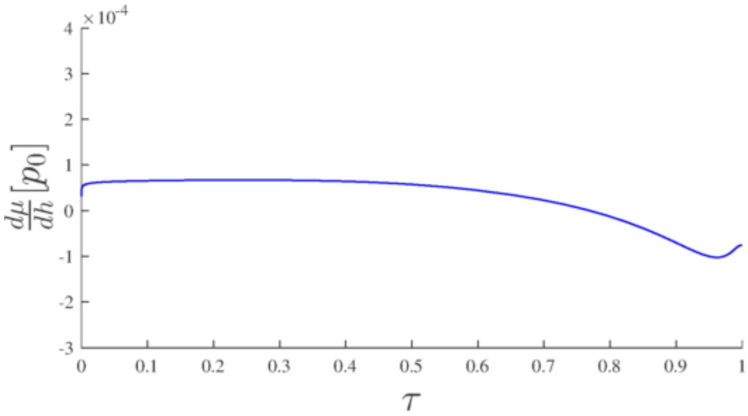
- Interpolation between SD is feasible based on the continuity of the OCP

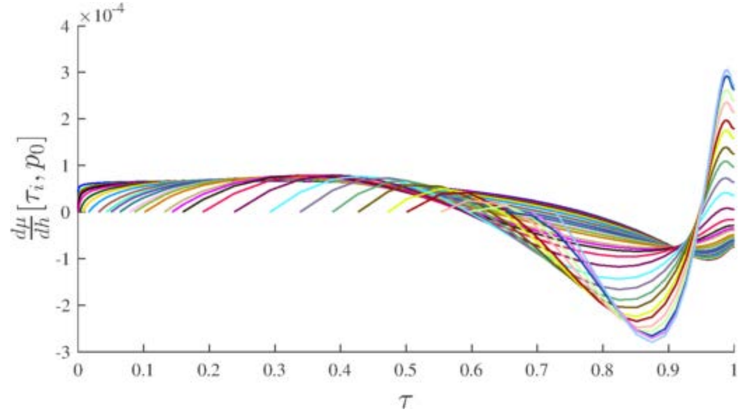
- Sensitivity surfaces

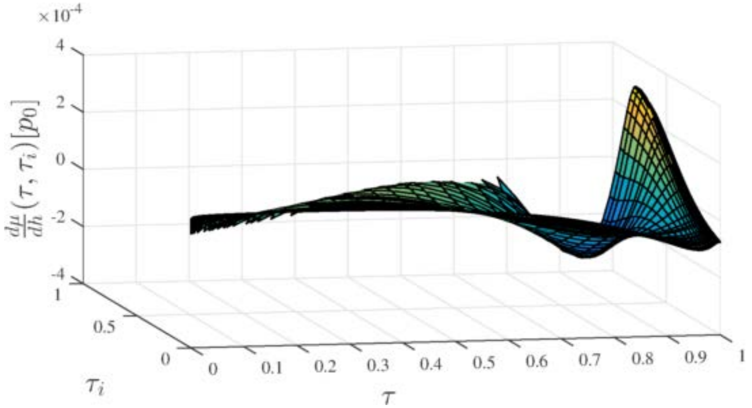
- Online: At time $\bar{t}$ surface is evaluated at $(t, \bar{t})$, $\bar{t} \leq t \leq t_f$

$$\frac{d\mu}{dh}(t, \bar{t}) \approx \frac{d\mu}{dh_{\bar{t}}}(t)$$

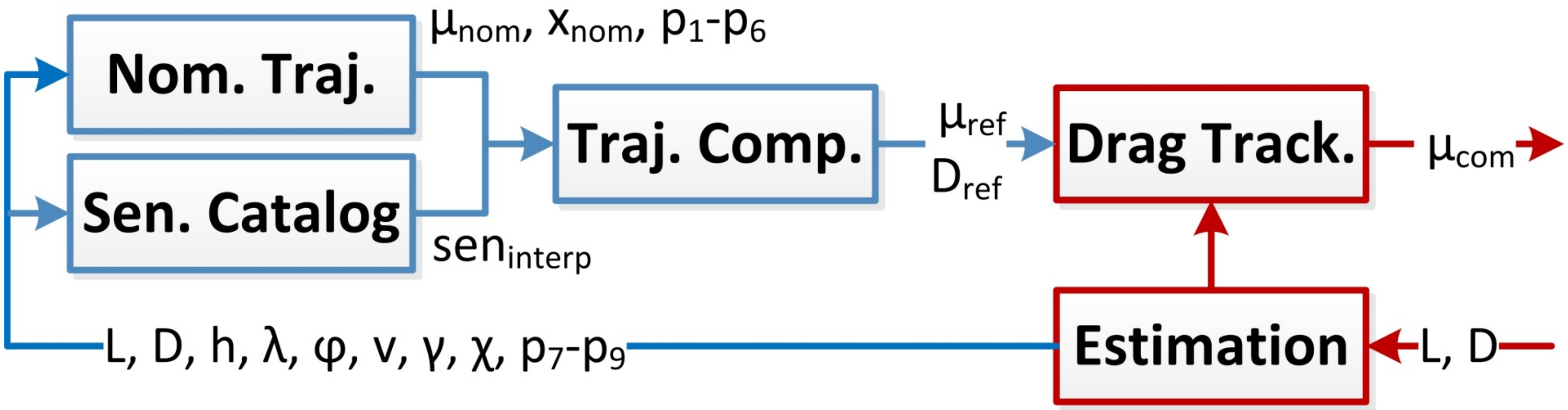
to obtain an approximation of the SD against a perturbation affecting the system at the instant $\bar{t}$







Guidance System Overview



- Two-degree-of-freedom design
  - Fast inner tracking loop (20 Hz)
  - Slow outer trajectory loop (0.05 Hz)

- Trajectory computation outputs
  - near optimal discrete u^*, x^* for the entire remaining process
  - optimal bank angle profile μ_{ref} and drag profile D_{ref} obtained from u^*, x^*

- Drag tracking controller based on Mease et. al.

Monte Carlo: Perturbed Environment

- Comparatively large EIP state errors (using a uniform error distribution)
- Atmospheric perturbations:
  - Random temperature profile between warm and cold conditions
  - Random sinusoidal density perturbations of up to 50% amplitude
- Aerodynamic coefficients perturbed by up to 10%
- Guidance input:
  - true state, lift and drag falsified with white noise
  - Atmospheric perturbation parameters are estimated using an extended kalman filter

State	Pert.
h_0	+- 3 km
λ_0	+- 0.3265°
φ_0	+- 0.1632°
v_0	+- 200 m/s
γ_0	+- 1°
χ_0	+- 1°

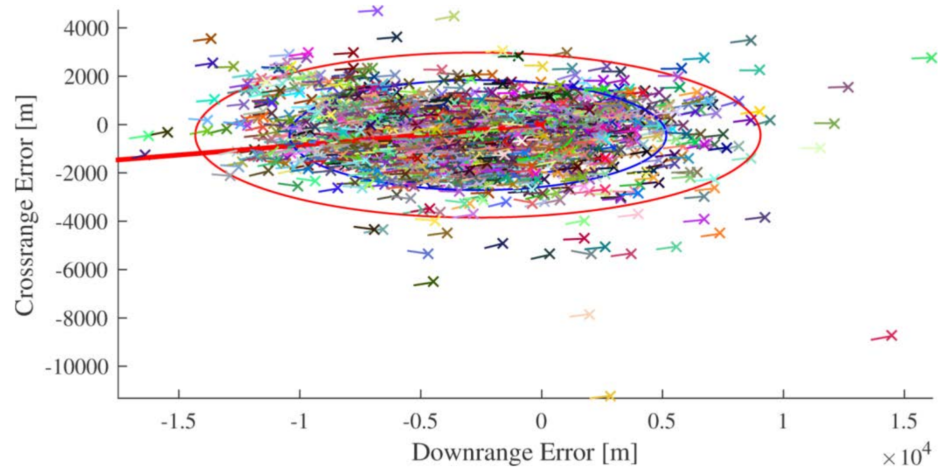
Param	Pert.
ρ	Temp. +- 50%
c_L	+- 10 %
c_D	+- 10 %
wind	+- 200 m/s
mass	+- 20 kg

Monte Carlo: Results

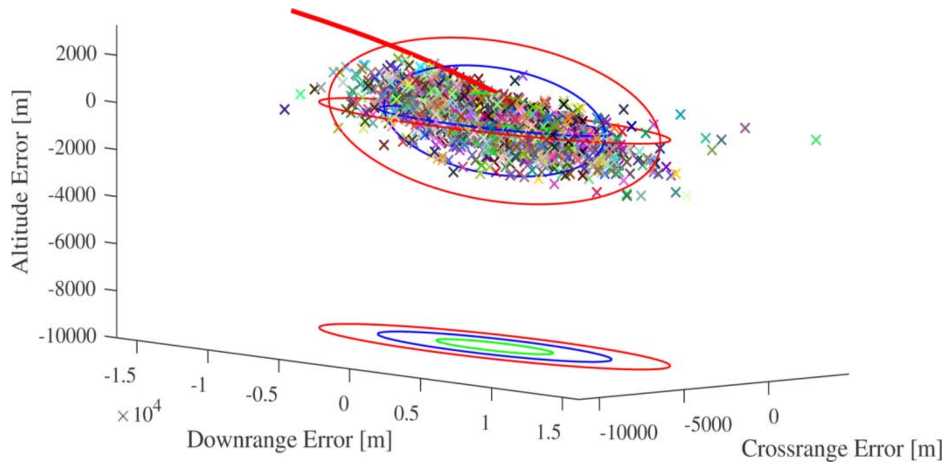
$(|\mu| + 3\sigma)$ hori. dist.: **12.3 km**

$(|\mu| + 3\sigma)$ alt. error: **2.4 km**

(3.5 DOF, 2500 MC cases)



Result	Mean (μ)	Std. Dev. (σ)
Eucl. dist.	4.1 km	2.6 km
Hori. dist.	4 km	2.7 km
DR error	- 2.6 km	3.8 km
CR error	- 0.4 km	1.1 km
Alt. error	- 0.4 km	0.7 km
Vel. error	4 m/s	6 m/s



Processor-in-the-loop

- Test on RASTA-101 with 80 MHz LEON2 processor
- GNC c-code from autocoding from Embedded Matlab
- TASTE toolset: Onboard SW interface definition, communication setup and target compilation
- Used dense NLP formulation
  - grid length 70
  - ~25 MB sensitivity data

Processor-in-the-loop

- Test on RASTA-101 with 80 MHz LEON2 processor
- GNC c-code from autocoding from Embedded Matlab
- TASTE toolset: Onboard SW interface definition, communication setup and target compilation
- Used dense NLP formulation
  - grid length 70
  - ~25 MB sensitivity data

Trajectory computation time: **< 1 sec.**

Summary

- Applied sensitivity analysis to discretized optimal control process for different initial conditions
- Repeated online trajectory computation based on sensitivity interpolation, Taylor expansion and iterative constraint correction
- Two-degree-of-freedom guidance system using drag tracking
- Promising results in 3.5 DOF Monte Carlo campaign
- Real-time capability proven by PIL test on LEON2 processor

Summary

- Applied sensitivity analysis to discretized optimal control process for different initial conditions
- Repeated online trajectory computation based on sensitivity interpolation, Taylor expansion and iterative constraint correction
- Two-degree-of-freedom guidance system using drag tracking
- Promising results in 3.5 DOF Monte Carlo campaign
- Real-time capability proven by PIL test on LEON2 processor

Thank you for your attention!

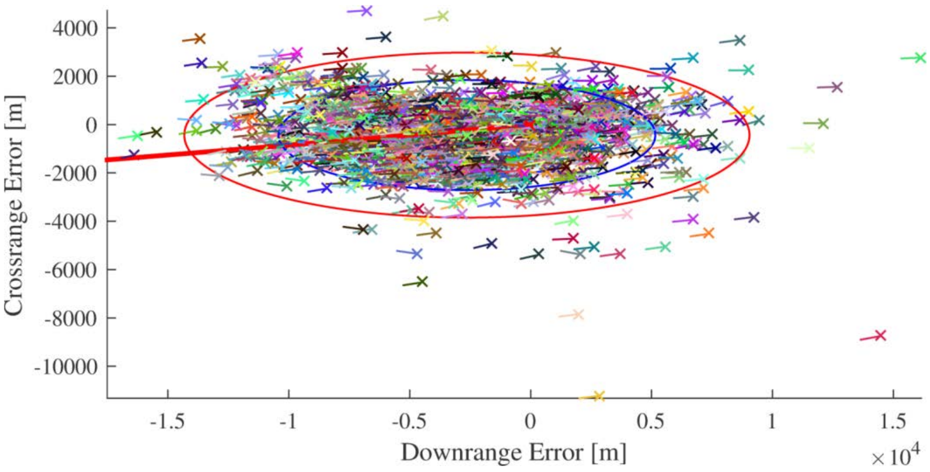
david.seelbinder@dlr.de

This research was conducted under No. 4000107257/12 NL/GLC/al of the ESA Networking and Partnering Initiative.

Monte Carlo and PIL Results

State	Pert.
h_0	+- 3 km
λ_0	+- 0.3265°
φ_0	+- 0.1632°
v_0	+- 200 m/s
γ_0	+- 1°
χ_0	+- 1°

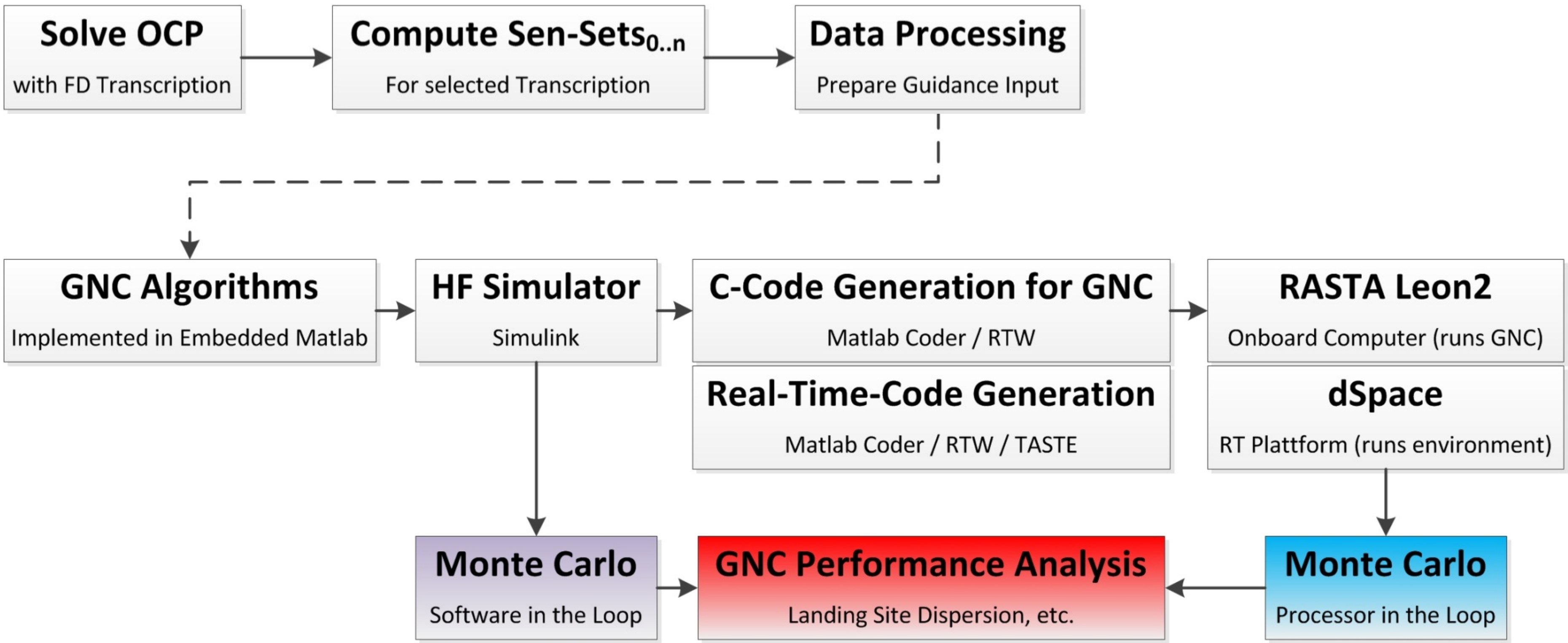
Param	Pert.
ρ	Temp. +- 50%
c_L	+- 10 %
c_D	+- 10 %
wind	+- 200 m/s
mass	+- 20 kg



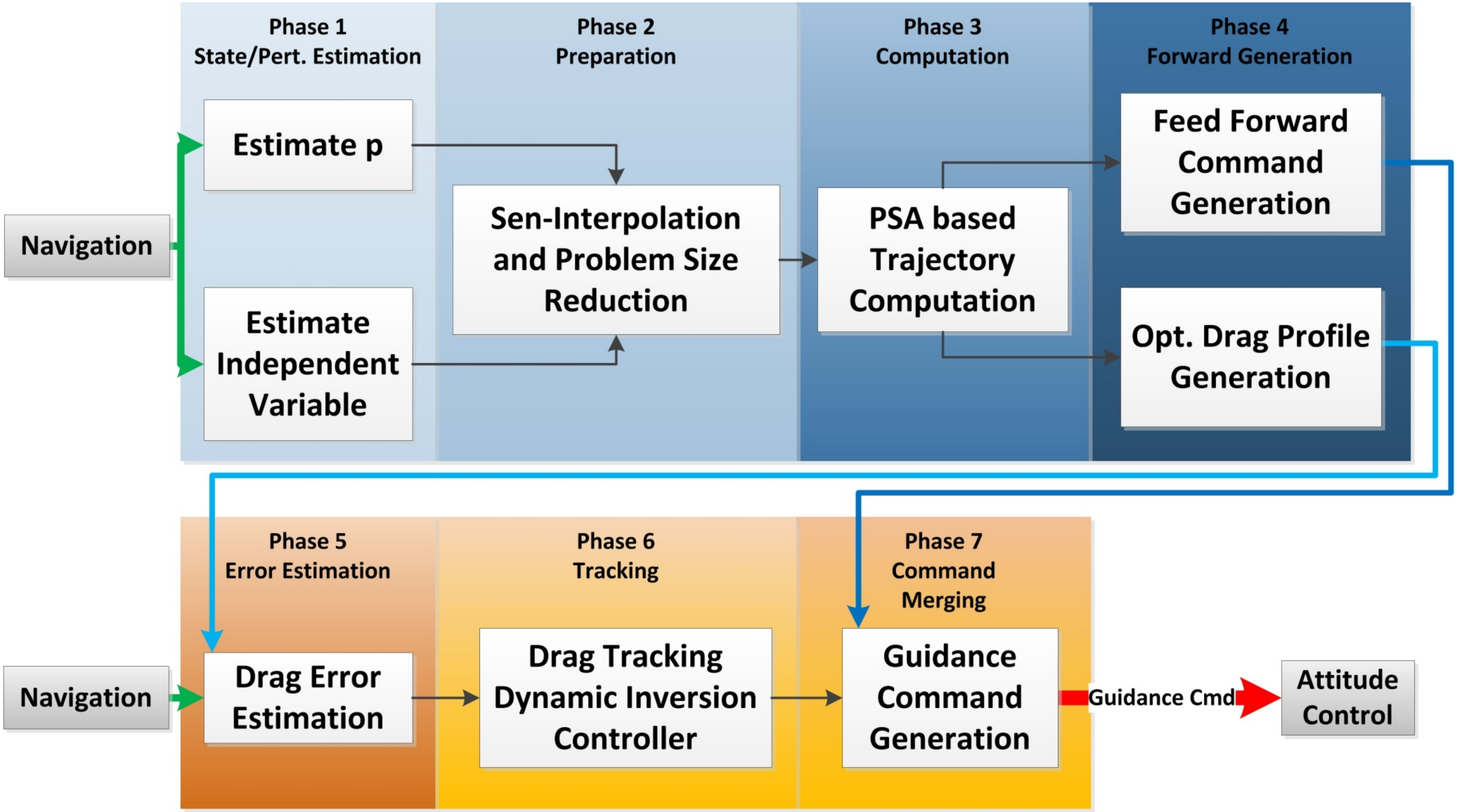
Result	Mean (μ)	Std. Dev. (σ)
Eucl. dist.	4.1 km	2.6 km
Hori. dist.	4 km	2.7 km
DR error	- 2.6 km	3.8 km
CR error	- 0.4 km	1.1 km
Alt. error	- 0.4 km	0.7 km
Vel. error	4 m/s	6 m/s

Task	i7, 2.9 GHz	LEON2, 80 MHz	Factor
Sen. interp.	12 ms	324 ms	~ 27
p-step	0.05 ms	3 ms	~ 60
q-step	0.19 ms	8 ms	~ 42
$G(z,p)$	0.88 ms	32 ms	~ 36

Real-Time Production Code Generation and Testing



Closed Loop Guidance Design

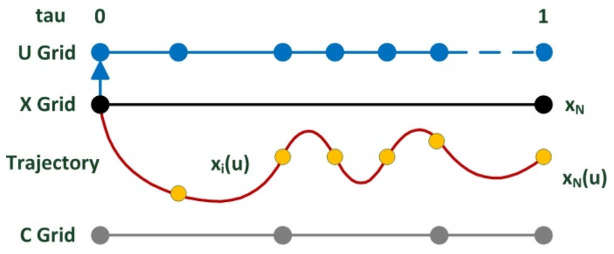


Parameter Dependent NLP Resulting From Control Discretization ("Single Shooting")

Minimize $\Phi(u,p) = g(x_1, x_N(u), p)$

subject to $\psi(x_1, x_N(u), p) - q^1 = 0$

$S(x_i(u), u_i, p) - q^2 \leq 0 \qquad i = 1, \dots, N$

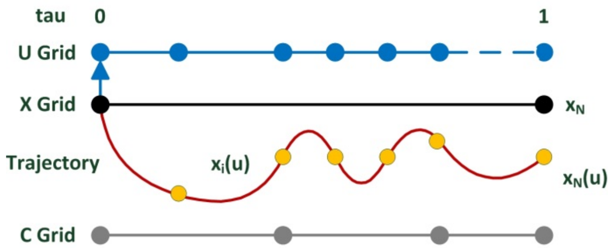


Parameter Dependent NLP Resulting From Control Discretization ("Single Shooting")

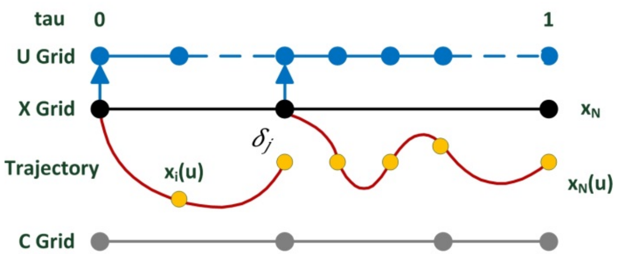
Minimize $\Phi(u,p) = \; g(x_1, x_N(u), p)$

subject to $\psi(x_1, x_N(u), p) - q^1 = \; 0$

$S(x_i(u), u_i, p) - q^2 \leq \; 0 \qquad i = 1, \dots, N$



Parameter Dependent NLP Resulting From Control and State Discretization ("Multiple Shooting")



Minimize $\Phi(u,x,p) = \; g(x_1, x_N, p)$

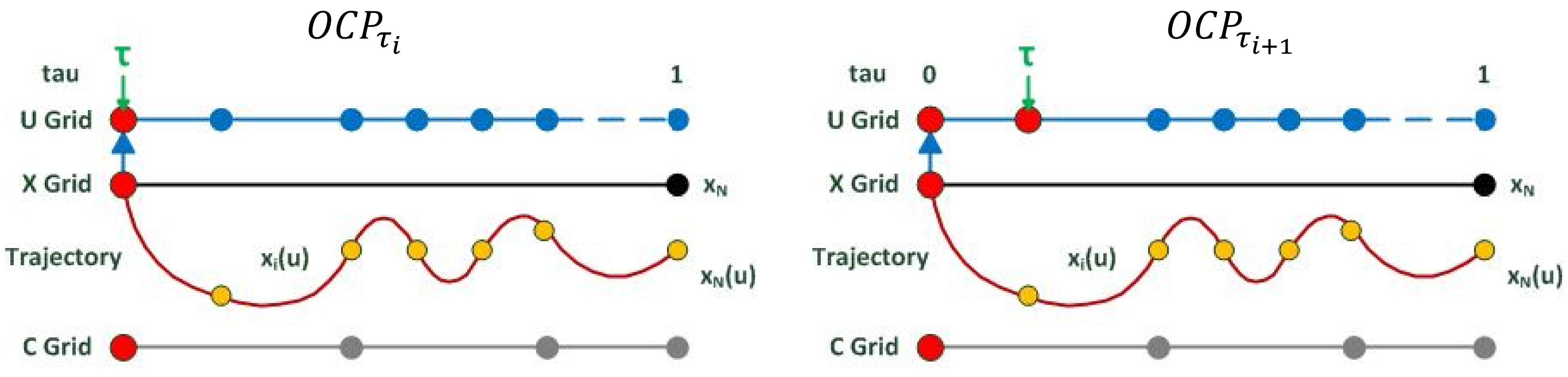
subject to $\psi(x_1, x_N, p) - q^1 = \; 0$

$S(x_i, u_i, p) - q^2 \leq \; 0 \qquad i = 1, \dots, N$

$\delta_j(x_i, u_{i \dots k}, x_{i+1}, p) - q^3 = \; 0 \qquad j = 1, \dots, SI \qquad k = 1, \dots, CpSI$

Mind: Only optimizable ("free") variables should be included in the NLP…

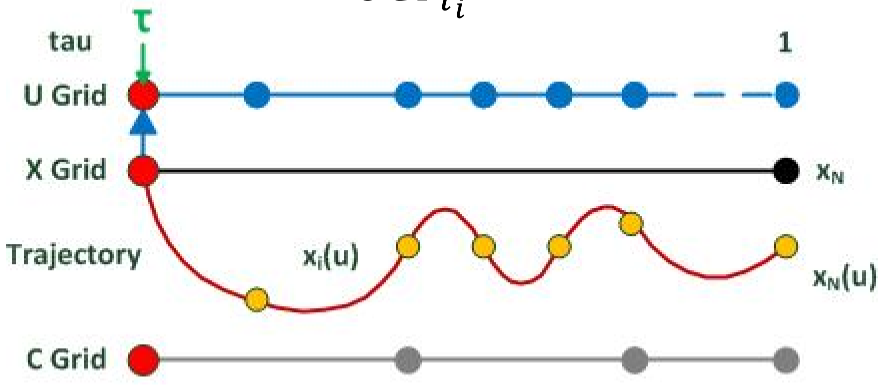
Initial Condition Dependency of Sensitivity Differentials



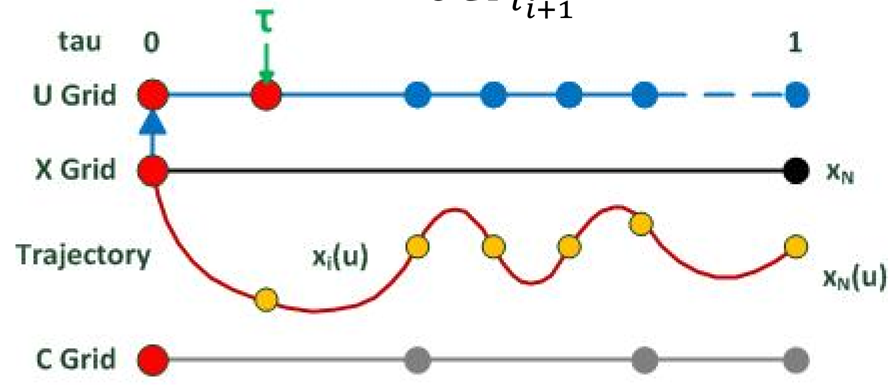
- OCP_i and OCP_{i+1} are closely related by the Bellman Principle

Initial Condition Dependency of Sensitivity Differentials

OCP_{τ_i}



$OCP_{\tau_{i+1}}$



- OCP_i and OCP_{i+1} are closely related by the Bellman Principle
- In the NLP representation of the OCP, gridded optimization variables become fixed up to τ_{cur}!
- Fixed variables can be removed ➔ **K-Matrix is time dependent**

$$K_{\tau_i} = \begin{bmatrix} \nabla_z{}^2 L(z_0, \eta_0^a, p_0) & \nabla_z G^a(z_0, p_0)^T \\ \nabla_z G^a(z_0, p_0) & 0 \end{bmatrix}$$

$$K_{\tau_{i+1}} = \begin{bmatrix} \nabla_z{}^2 L(z_0, \eta_0^a, p_0) & \nabla_z G^a(z_0, p_0)^T \\ \nabla_z G^a(z_0, p_0) & 0 \end{bmatrix}$$

Estimation of Sensitivity Matrix Dimensions for PSA Update

	SS	FD
# NLP variables: N	u*D	(u+x)*D
# Constraints: M	x	(x-1)*D
Size(dz/dp) = N*p	u*D*p	(u+x)*D*p
Size(dz/dq) = N*M	u*x*D	$(u*x+x^2)*(D^2 - D)$

Example: Entry Problem: x=10, u=1, D=50, no path constraints!

SS size(dz/dq) = 500
FD size(dz/dq) = 269500

Lower bound on multiplication operations required per update iteration

➔ SS clearly preferable as it will allow for a much **higher trajectory generation frequency**