Multidisciplinary Modeling and Simulation Framework for Reusable Launch Vehicle System Dynamics and Control

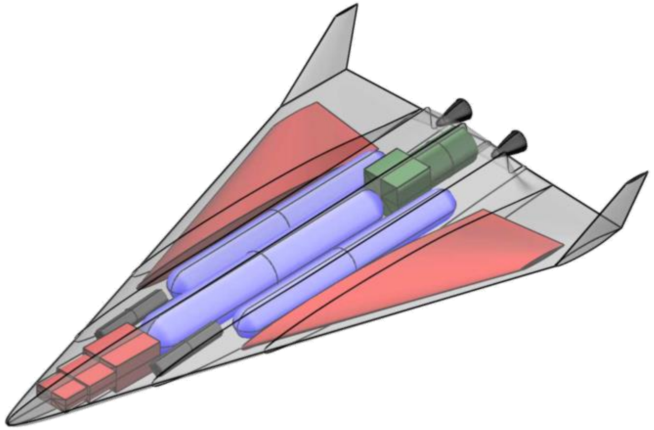
Lâle Evrim Briese, Paul Acquatella B., Klaus Schnepper

DLR, German Aerospace Center
Institute of System Dynamics and Control
Oberpfaffenhofen, Germany

Introduction

- **Background**
  - Several studies on future launch vehicle configurations and technologies for **expendable and reusable launch vehicles** have been extensively conducted in the past at DLR
  - Currently, partly or fully reusable launch vehicles using different return methods are investigated at DLR in the context of the research projects **AKIRA** and **X-TRAS**
- **Goal:**
  - Modeling and simulation of launch vehicle **system dynamics** for the subsequent development of **advanced control systems**

DLR-SART SpaceLiner 7 - Copyright © DLR



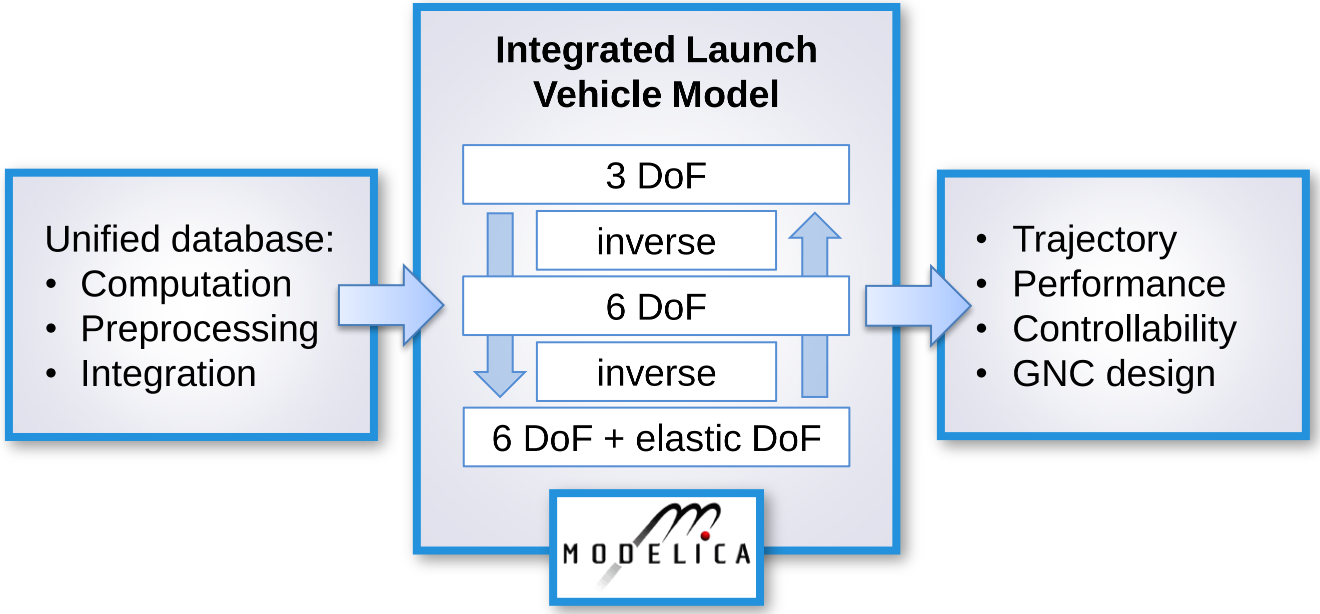
DLR-SART Aurora, Copyright © DLR

DLR-SART LFBB, Copyright © DLR

Introduction

- **Challenges:**
  - Highly interconnected disciplines
  - Changing environmental conditions
  - Changing mission types and requirements
  - Consistent modeling for each level of detail

- **Objective:**
  - Development of a multibody **modeling and simulation framework** for preliminary design studies of reusable launch vehicles
    - Multidisciplinary
    - Object-oriented
    - User-friendly
    - Flexible



Overview

1. **Object-oriented Modeling with Modelica / Dymola**

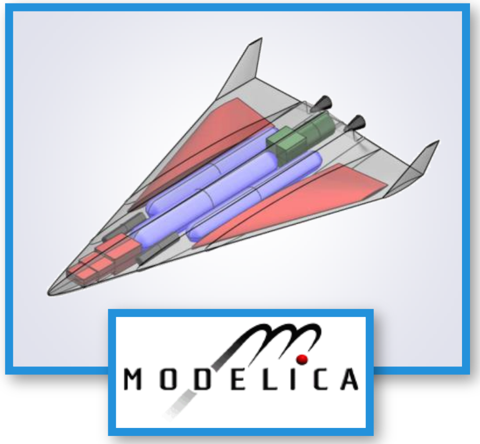
2. Launch Vehicle Modeling, Guidance, and Control Framework

    - Launch Vehicle Modeling
    - Trajectory Optimization
    - Nonlinear Inverse Modeling

3. Simulation Case Study: Aurora Descent Flight

4. Summary & Outlook

Object-oriented Modeling with Modelica / Dymola

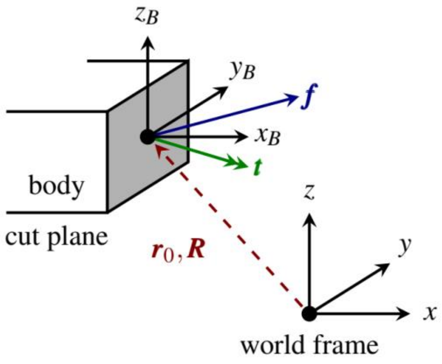


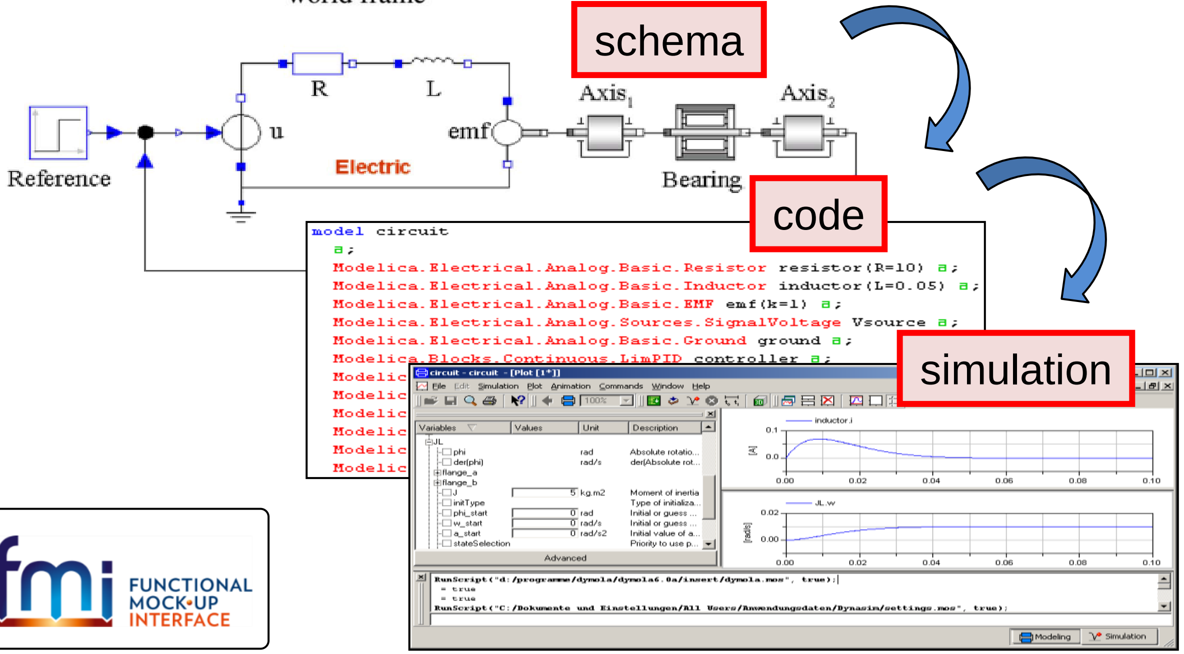
- **Modelica**: object-oriented modeling language
    - Modeling of physical systems
    - Multidisciplinary (e.g. mechanics, electrics)
    - Acausal description (equation-based)
    - Algebraic, discrete and differential equations

$$\boldsymbol{F}(\dot{\boldsymbol{x}}(t), \boldsymbol{x}(t), \boldsymbol{y}(t), t) = 0$$

- **Dymola**: simulation environment based on Modelica
    - Automatic code generation
    - Graphical and textual programming
    - Supports model exchange and co-simulation

- **Applications**: automotive, aerospace, robotics, …



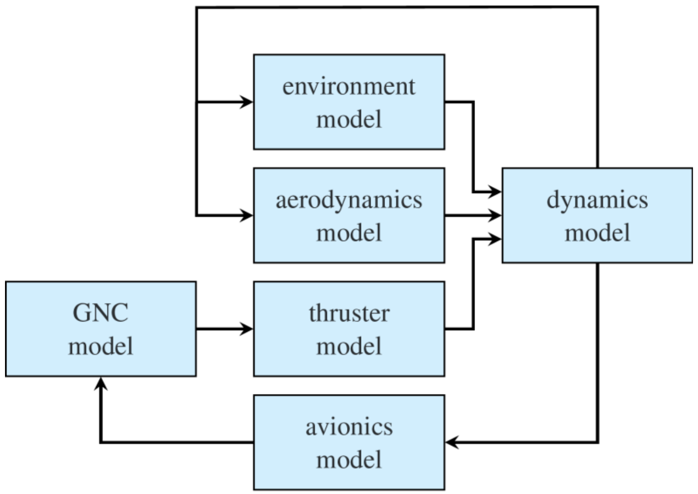


FMI FUNCTIONAL MOCK-UP INTERFACE

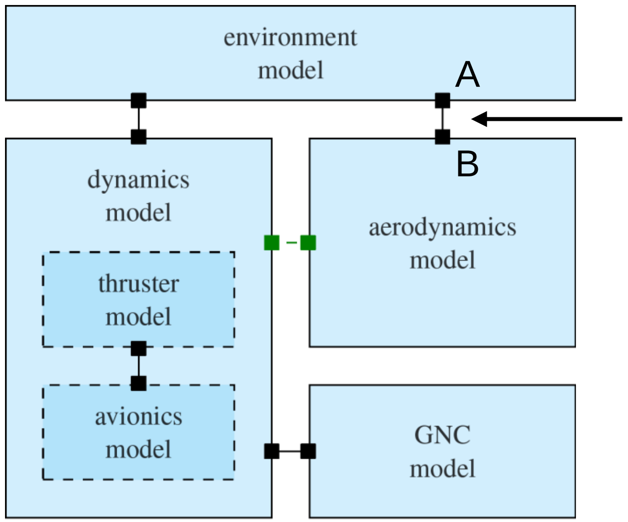
Object-oriented Modeling with Modelica / Dymola

Classic vs. Acausal Modeling Approach

- **Classic Modeling Approach:**
  - Subsystems considered as “signal processors”
  - Flow direction defined by fixed inputs & outputs



- **Acausal Modeling Approach:**
  - Subsystems considered as “energy exchangers”
  - Combination of flow and potential variables



$$\boldsymbol{r}_A = \boldsymbol{r}_B$$
$$\boldsymbol{R}_A = \boldsymbol{R}_B$$
$$\boldsymbol{0} = \boldsymbol{f}_A + \boldsymbol{f}_B$$
$$\boldsymbol{0} = \boldsymbol{t}_A + \boldsymbol{t}_B$$

Combination of Classic and Acausal Modeling in Modelica

Overview

1. Object-oriented Modeling with Modelica / Dymola

2. **Launch Vehicle Modeling, Guidance, and Control Framework**

    - Launch Vehicle Modeling
    - Trajectory Optimization
    - Nonlinear Inverse Modeling

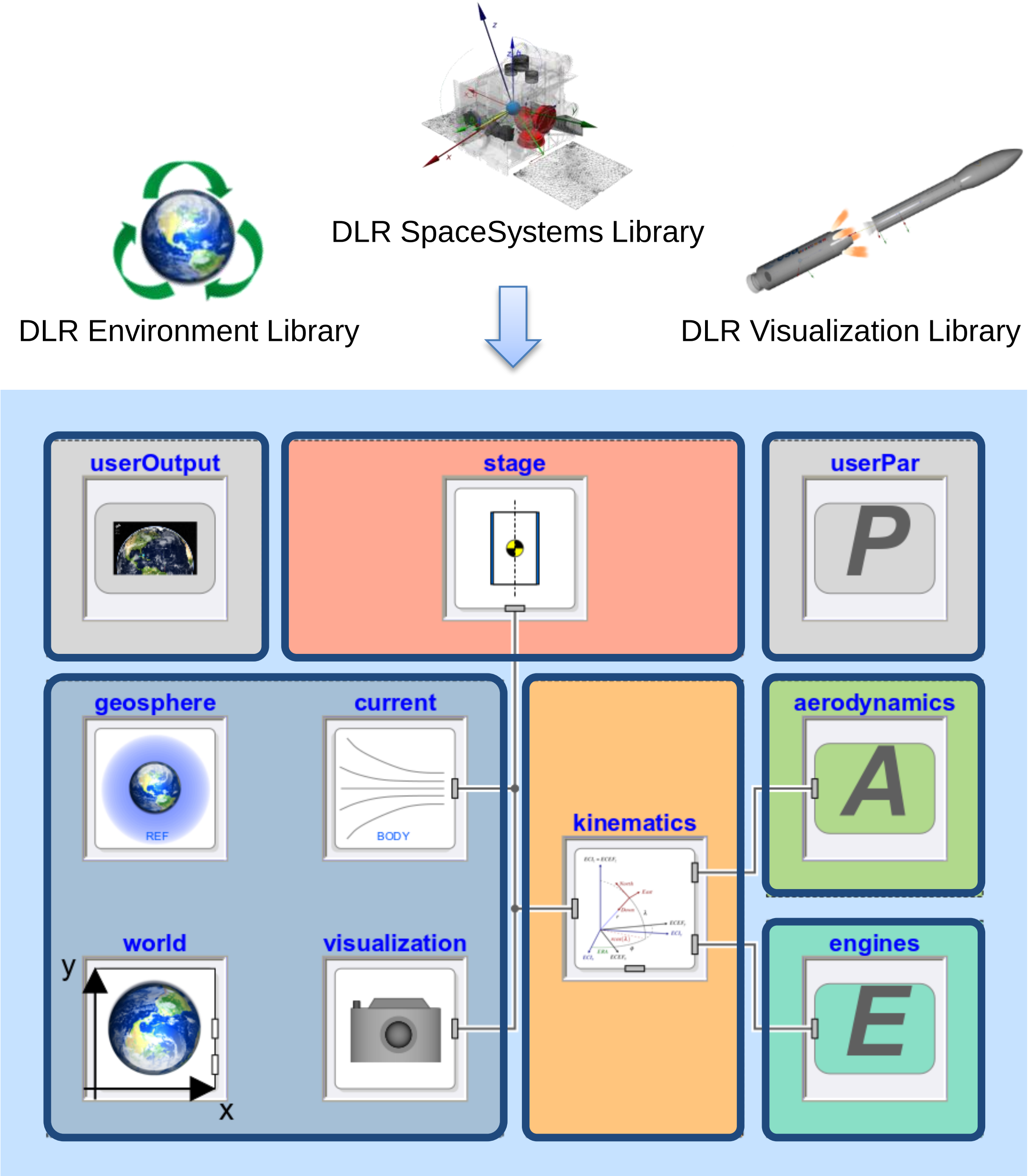
3. Simulation Case Study: Aurora Descent Flight

4. Summary & Outlook

Launch Vehicle Modeling Framework

General Overview

- **Multidisciplinary Modeling Approach**
  - Environment → gravity, atmosphere, …
  - Kinematics → states, transformations, …
  - Dynamics → stage, aerodynamics, engines
  - Unified database → user-defined inputs and outputs
- **Object-oriented Modeling Approach**
  - Modular, user-friendly, adaptive
  - Acausal and equation-based
  - Replaceable components
  - Consistent modeling:
    - 3-DoF (point mass)
    - 6-DoF (point mass, multibody)
- **Supported by Modelica-based Libraries (DLR-SR)**



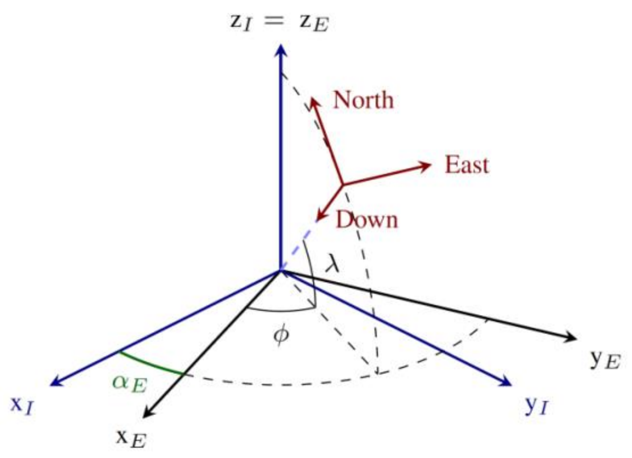
Launch Vehicle Modeling Framework
Environment

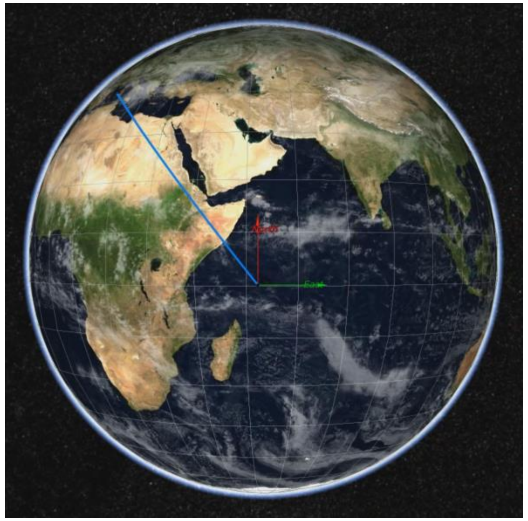
- Based on the DLR Environment Library
- Visualization of Earth, Sun & Moon

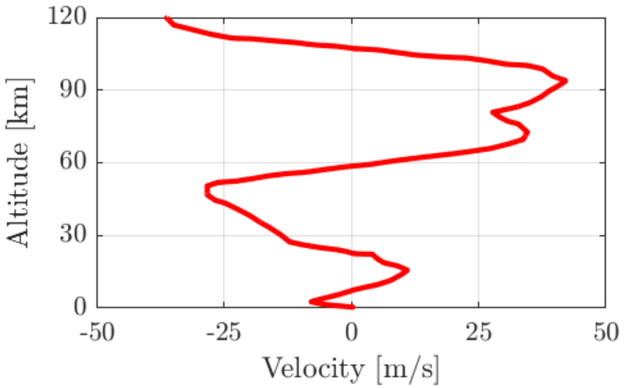
- **world:**
  - Geocentric / geodetic Coordinate Systems based on World Geodetic System WGS'84
  - Gravitational Models, such as EGM96

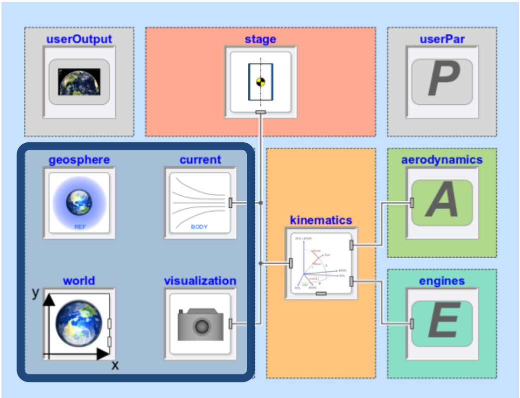
- **geosphere:**
  - Atmosphere Models, such as ISA or NRL-MSISE-00
  - Considers geoid undulation and height above MSL

- **current:**
  - Provides wind profiles (including turbulence models)
  - Applies wind forces individually to each launch vehicle based on current atmospheric conditions (geosphere)





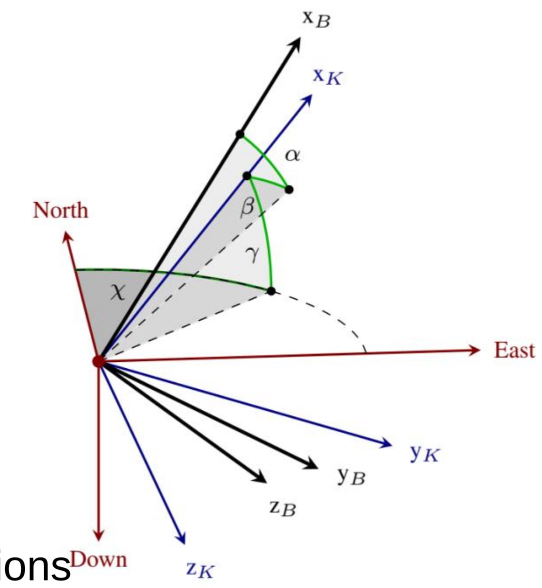


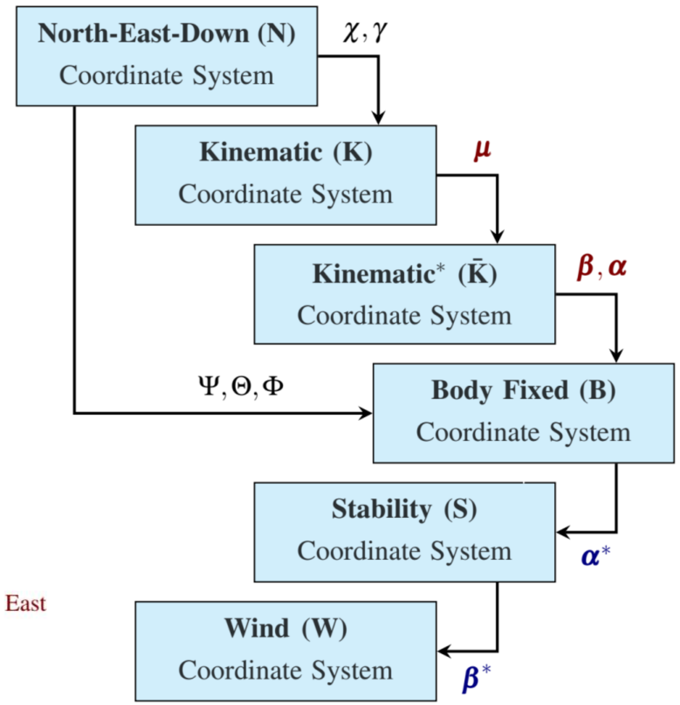


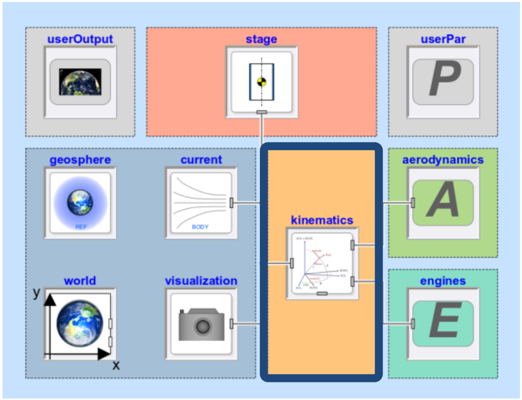
Launch Vehicle Modeling Framework

Kinematics

- Provides flight coordinate systems and physical transformations
- Allows individual state selection based on simulation study
- **Position** defined by latitude, longitude, altitude (considering spherical or ellipsoid planet definition)
- **Velocity** typically defined using flight path parameters $\{V, \gamma, \chi\}$
- Singularity ($\gamma = 90°$) can be avoided by velocity vector w.r.t. NED
- **Orientation** defined by Euler Angles Ψ (yaw), Θ (pitch), Φ (roll)
- Singularity ($\Theta = 90°$) can be avoided by Euler-Rodrigues quaternions
- **Angular Rates** defined w.r.t. body fixed coordinate system







Launch Vehicle Modeling Framework

Dynamics

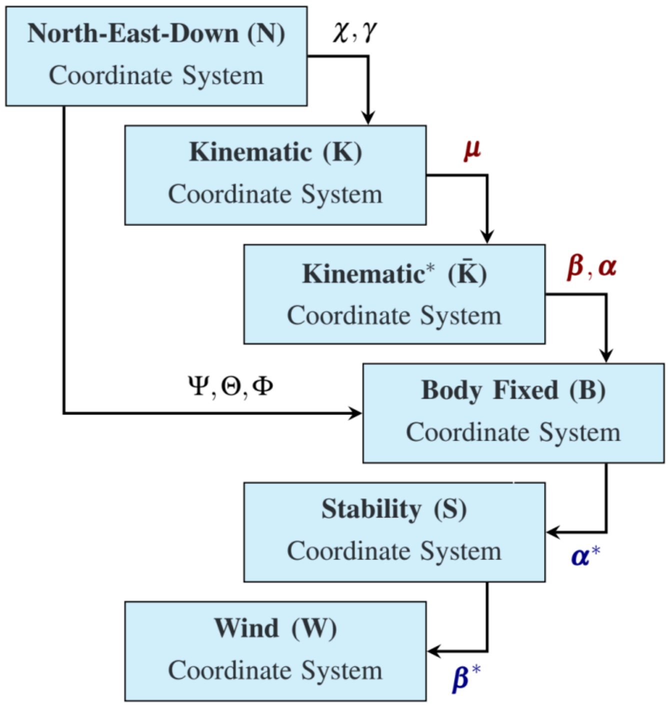
- **Equations of Motion (6-DoF)**
  - Newton-Euler Equations of Motion implemented per default:

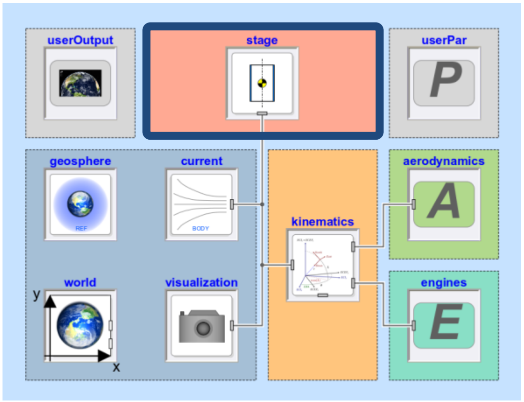
$$\begin{bmatrix} m\boldsymbol{I}_3 & 0 \\ 0 & \boldsymbol{I}_B \end{bmatrix} \begin{bmatrix} \boldsymbol{a}_B \\ \boldsymbol{\alpha} \end{bmatrix} + \begin{bmatrix} 0 \\ \boldsymbol{\omega} \times \boldsymbol{I}_B \boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} \boldsymbol{F} \\ \boldsymbol{M} \end{bmatrix}$$

  - For variable mass dynamics, the Equations of Motion are extended by:

$$\begin{bmatrix} m\boldsymbol{I}_3 & 0 \\ 0 & \boldsymbol{I}_B \end{bmatrix} \begin{bmatrix} \boldsymbol{a}_B \\ \boldsymbol{\alpha} \end{bmatrix} + \begin{bmatrix} 0 \\ \boldsymbol{\omega} \times \boldsymbol{I}_B \boldsymbol{\omega} \end{bmatrix} + \begin{bmatrix} \frac{\mathrm{d}m}{\mathrm{d}t} \boldsymbol{v}_r \\ \frac{\mathrm{d}\boldsymbol{I}_B}{\mathrm{d}t} \boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} \boldsymbol{F} \\ \boldsymbol{M} \end{bmatrix} + \begin{bmatrix} \boldsymbol{F}_{C2} \\ \boldsymbol{M}_{C2} \end{bmatrix}$$

- **Equations of Motion (3-DoF)**
  - Time-scale separation between translational & rotational dynamics
  - Angular velocities & accelerations are automatically set to zero





Launch Vehicle Modeling Framework

Dynamics

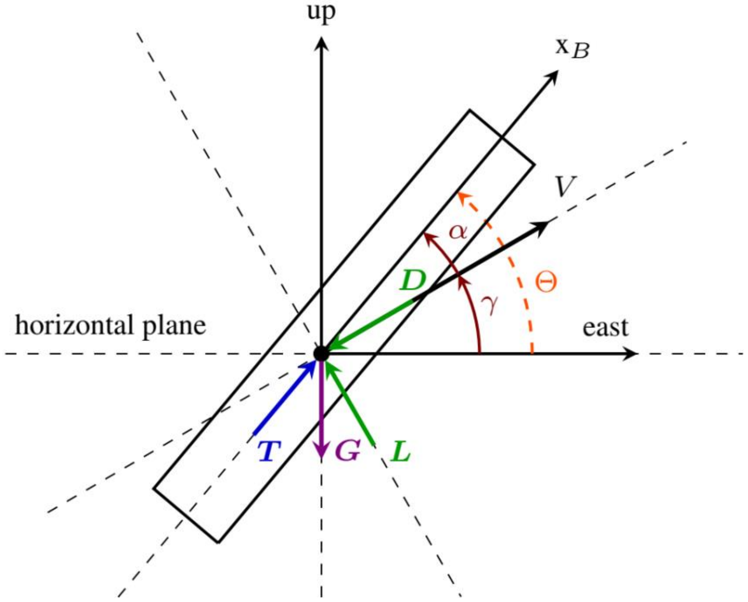
- **Forces and moments** are provided by dedicated replaceable components

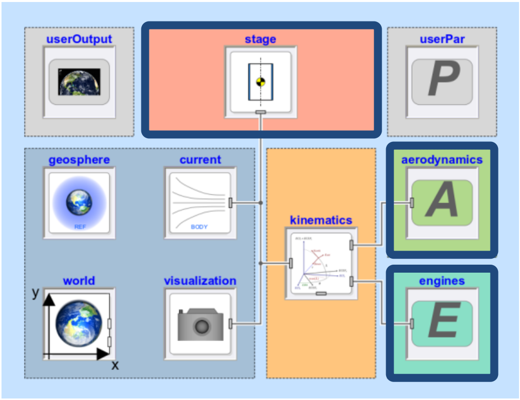
- **Gravity** (stage): $\boldsymbol{F}_G^N = \begin{bmatrix} mg_x \\ mg_y \\ mg_z \end{bmatrix}$

- **Thrust** (engines): $\boldsymbol{F}_T^B = \begin{bmatrix} T \\ 0 \\ 0 \end{bmatrix}$ e.g. with $T = c_s \left[T_{vac} - \frac{p}{p_0}(T_{vac} - T_{sl})\right]$

- **Aerodynamics**: $\boldsymbol{F}_A^S = \begin{bmatrix} -D \\ Y \\ -L \end{bmatrix} = \frac{1}{2}\rho V^2 S_{ref} \begin{bmatrix} -c_D \\ c_Y \\ -c_L \end{bmatrix}$

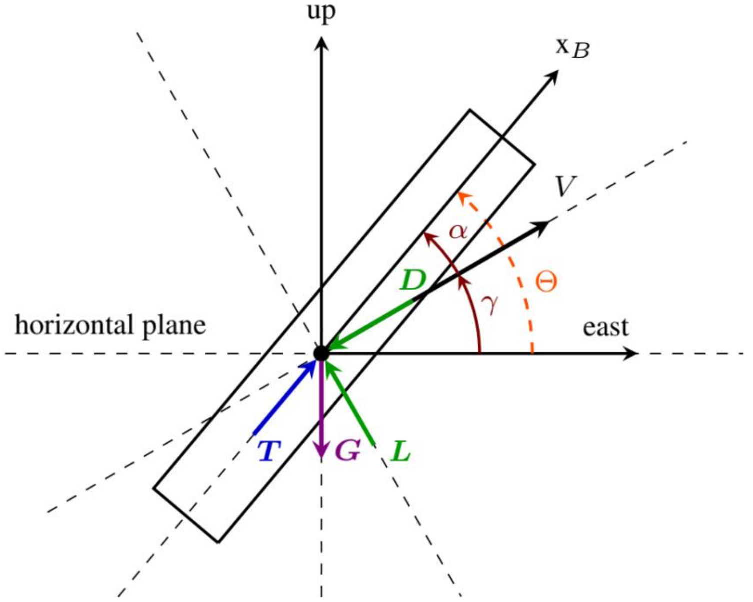
$$\boldsymbol{M}_A^B = \begin{bmatrix} l \\ m \\ n \end{bmatrix} = \frac{1}{2}\rho V^2 S_{ref} \begin{bmatrix} l_{ref} c_l \\ b_{ref} c_m \\ l_{ref} c_n \end{bmatrix}$$





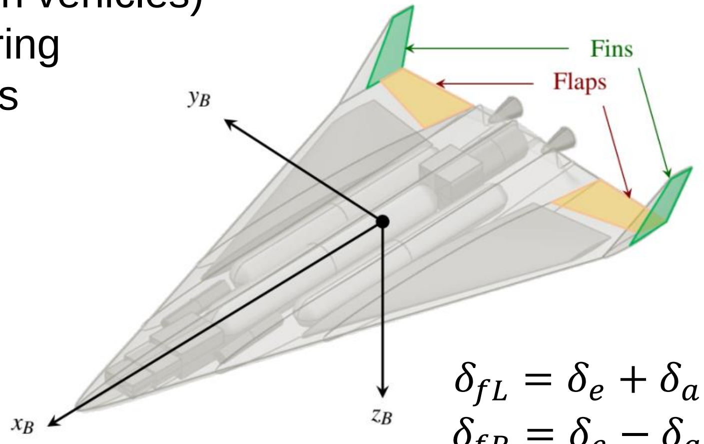
Launch Vehicle Modeling Framework

Aerodynamics

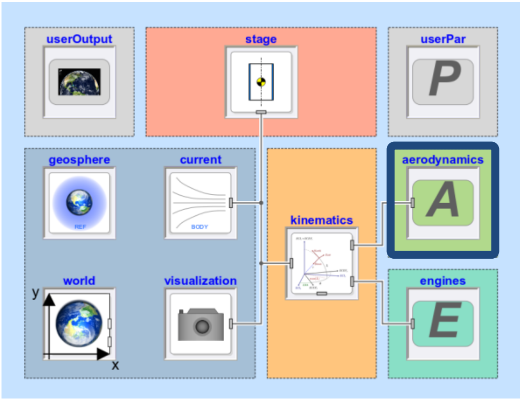


- **Aerodynamic coefficients** provided by DLR-RY (CAC, Hotsose)
- Aerodynamic database mainly dependent on the chosen level of detail
    - **3-DoF** (e.g. trajectory optimization)
        - Only translational lift & drag coefficients with $\boldsymbol{c}_A = f(Ma, \alpha)$
    - **6-DoF** (e.g. preliminary controllability studies)
        - Full aerodynamic coefficient matrix with $\boldsymbol{c}_A = f(Ma, \alpha)$
    - **Extended 6-DoF** (e.g. for winged reusable launch vehicles)
        - Full aerodynamic coefficient matrix considering aerodynamic angles, rates, deflection angles

$$\boldsymbol{c}_A = \begin{bmatrix} c_D \\ c_Y \\ c_L \\ c_l \\ c_m \\ c_n \end{bmatrix} = \begin{bmatrix} c_{D,\alpha} + c_{D,\delta_a} + c_{D,\delta_e} + c_{D,\delta_r} \\ c_{Y,\delta_r} \\ c_{L,\alpha} + c_{L,\delta_a} + c_{L,\delta_e} \\ c_{l,\delta_a} \\ c_{m,\alpha} + c_{m,\delta_a} + c_{m,\delta_e} \\ c_{n,\delta_r} \end{bmatrix}$$



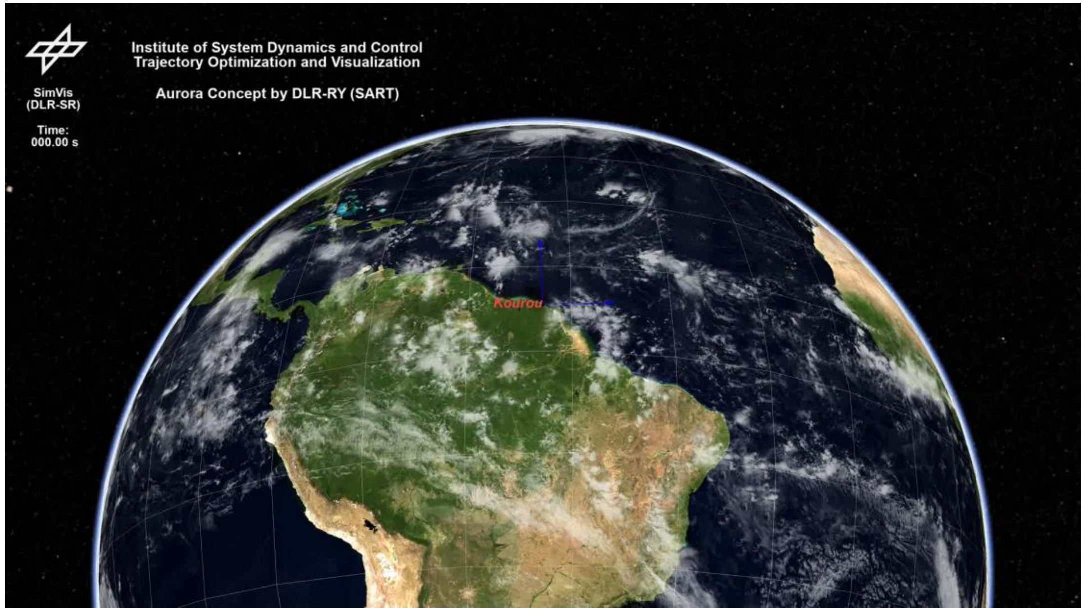
$$\delta_{fL} = \delta_e + \delta_a$$
$$\delta_{fR} = \delta_e - \delta_a$$

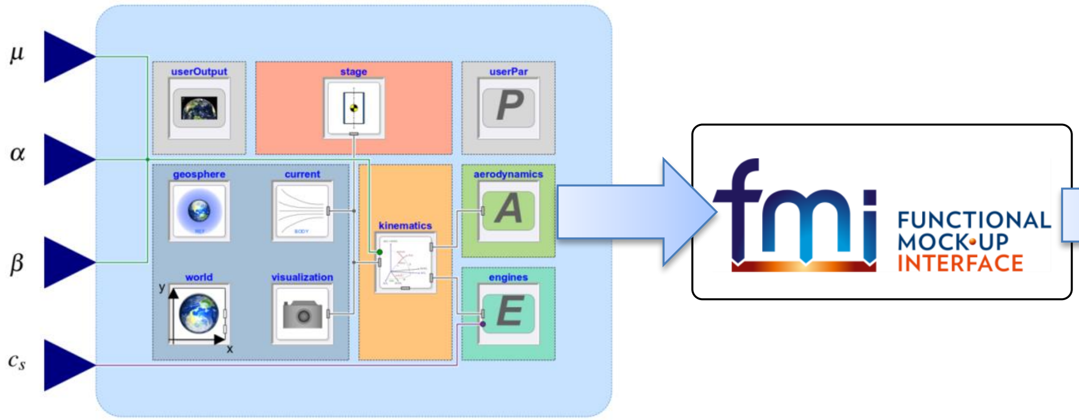


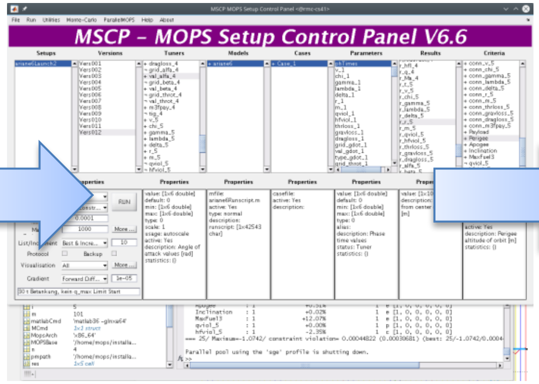
Trajectory Optimization

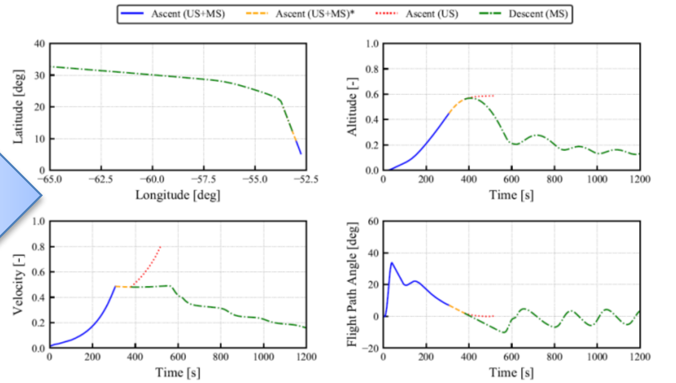
Optimal Guidance

- **Coupling of Modelica models with the Matlab-based trajectory optimization framework MOPS TrajOpt**
  - 3 DoF multibody model of the launch vehicle with aerodynamics angles and thrust throttle factor as inputs
  - Generation of a Functional Mock-up Unit (FMU) of the Modelica model using the FMI 2.0 Standard
  - Multi-objective and multi-phase trajectory optimization with MOPS TrajOpt using the FMU via multiple shooting

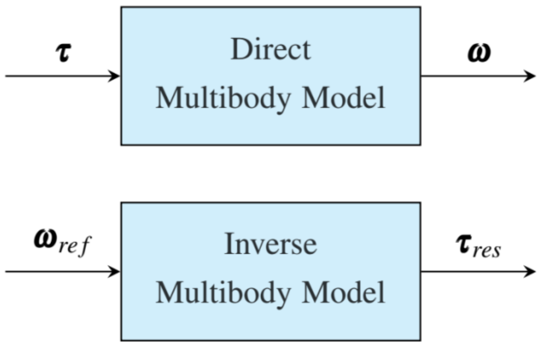




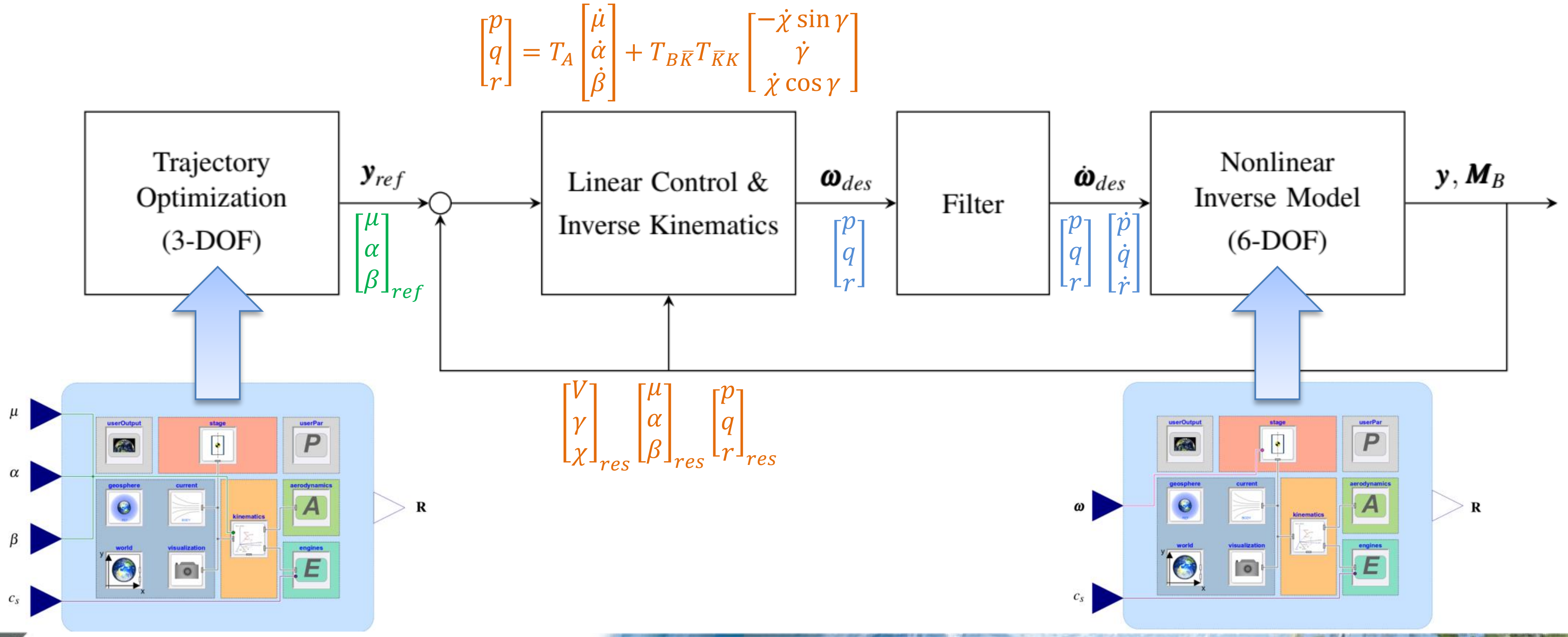




Nonlinear Inverse Modeling



- **Investigation of moment budgeting within preliminary design studies**:
  - Implementation of 6-DoF Nonlinear Direct and Inverse Multibody Models
  - Computation of required moments in order to follow optimal reference trajectory

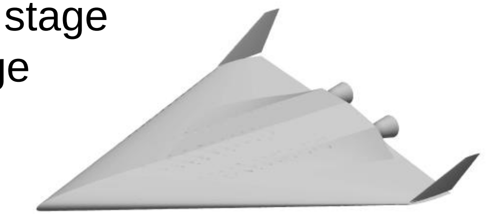


Overview

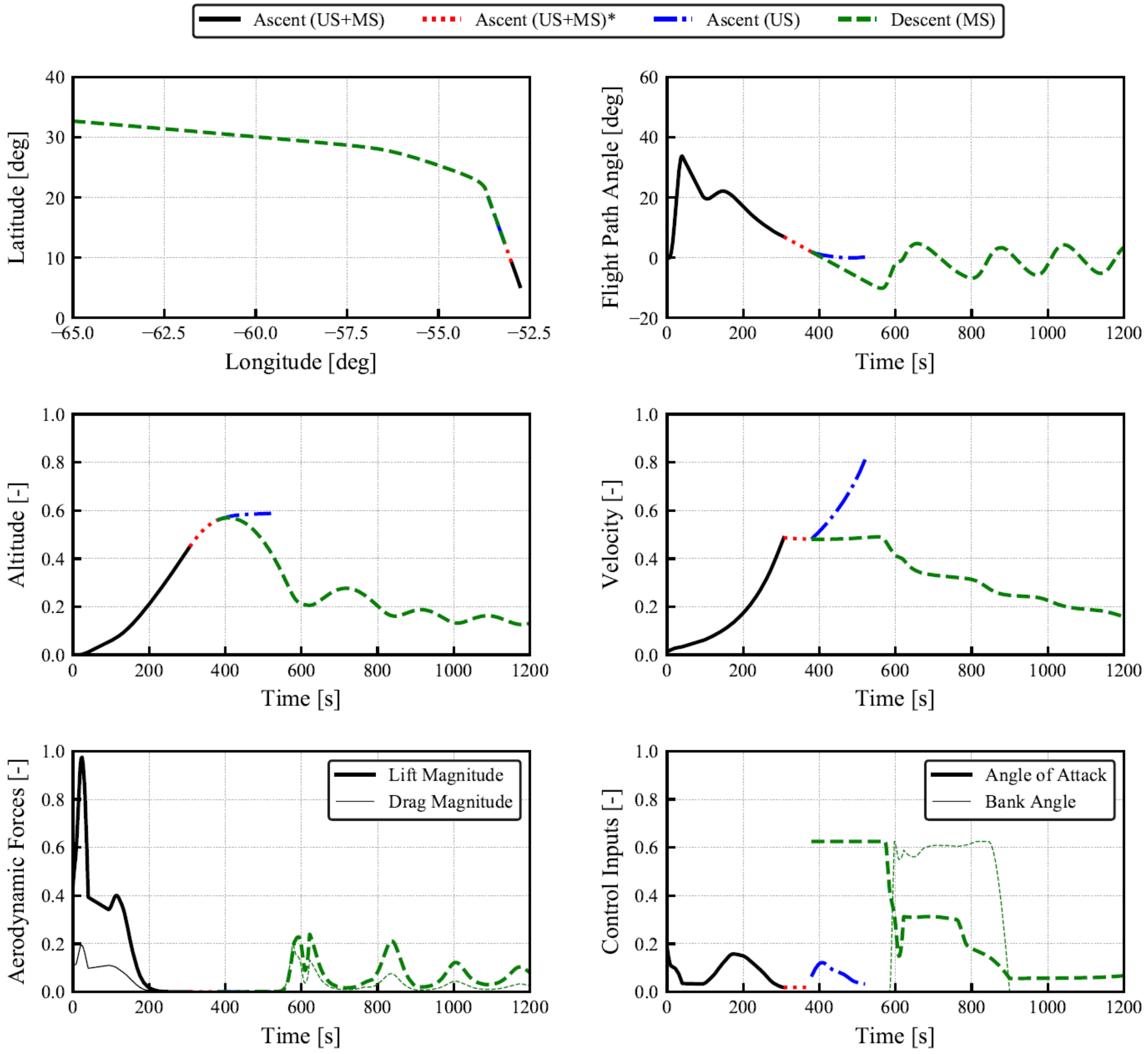
Case Study: Aurora by DLR-RY

Horizontal Liftoff and Horizontal Landing

- **Delta-winged launch vehicle configuration (Aurora):**
  - Two-stage to orbit configuration
  - Winged reusable main stage
  - Expendable upper stage

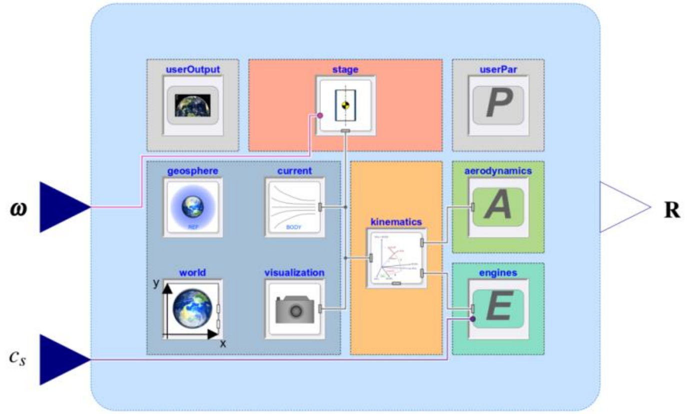


- **Multi-Phase definition:**
  - Ascent (US + MS)
    - Horizontal liftoff
    - Ballistic phase and separation
  - Ascent (US)
    - Fairing release
    - Payload release
  - Descent (MS)
    - Descent flight to landing site

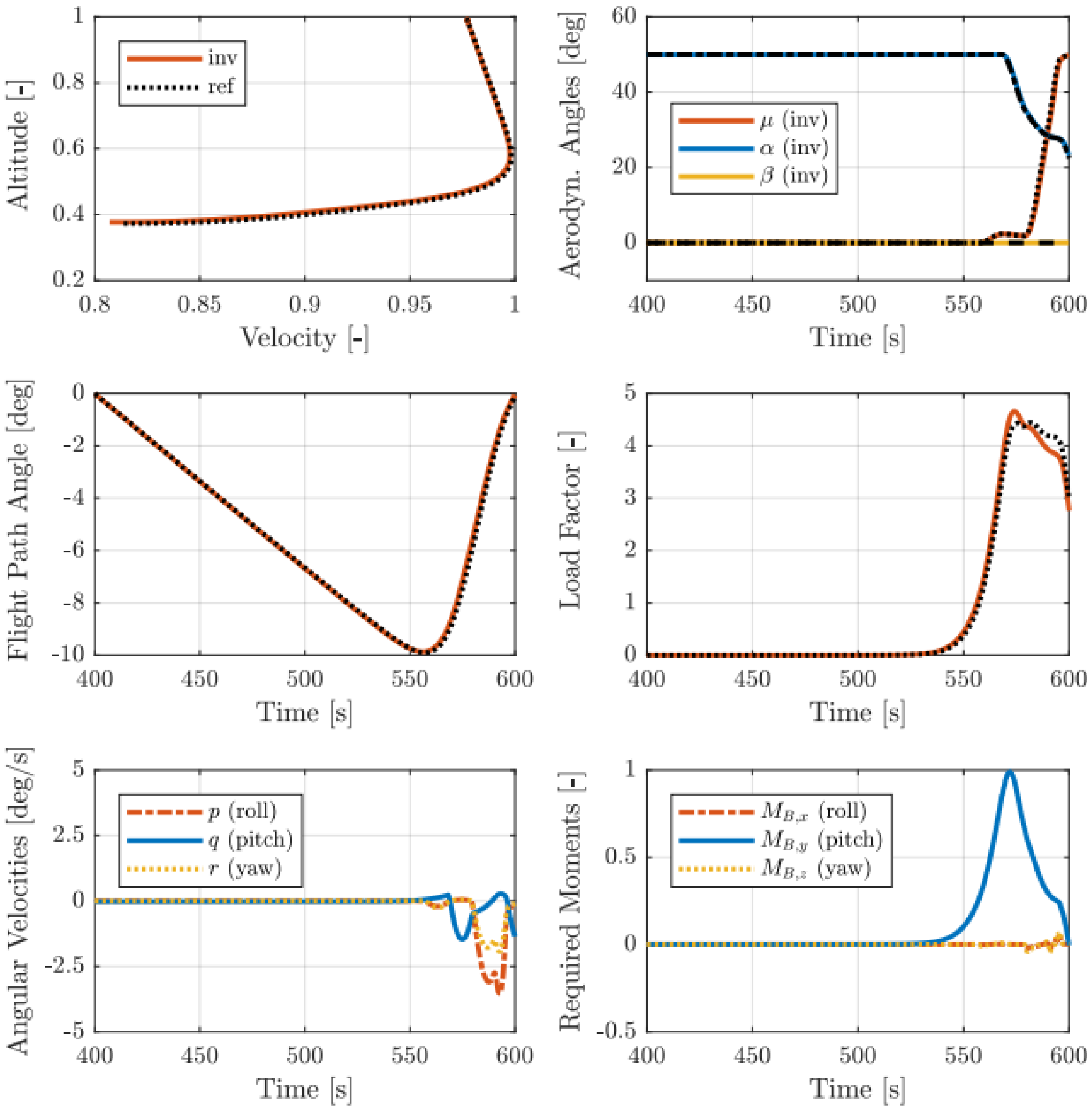


Case Study: Aurora by DLR-RY
Results for Phase P5-a

- **Inputs** of the nonlinear inverse model: **angular rates**
- **Outputs** of the nonlinear inverse model: **required moments**



- Nonlinear inverse model follows optimal guidance commands $\{\mu, \alpha, \beta\}$ provided by the trajectory optimization
- Position control is not considered!



Case Study: Aurora by DLR-RY

Results for Phase P5-c

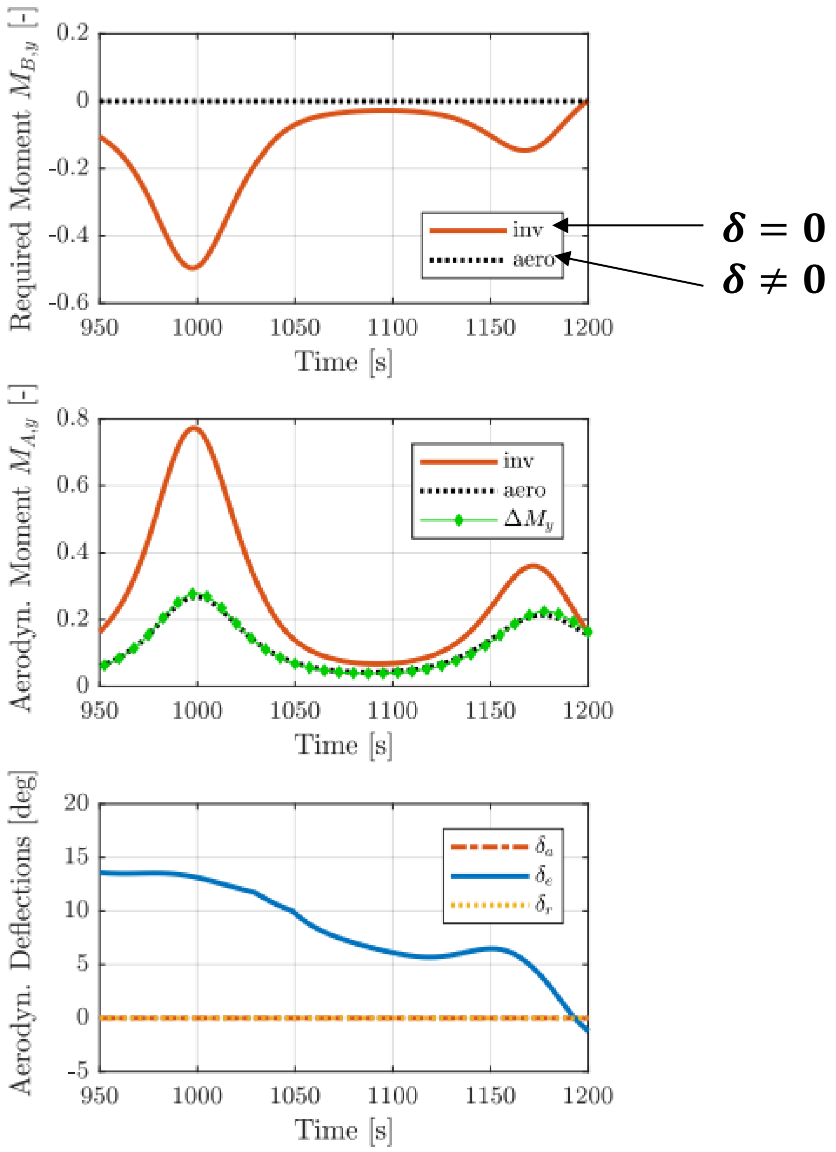
- 6-DoF nonlinear inverse model contains full aerodynamic coefficient matrix (including moments) assuming deflection angles to be zero

$$\delta_a = \delta_e = \delta_r = 0$$

- Comparison with the cascaded **nonlinear inversion control** including aerodynamic surface deflections [Acquatella et al., IAC 2018]:
    - Additional aerodynamic moments to be provided by aerodynamic surface deflection angles are obtained by nonlinear inverse model

$$\Delta M = M_{ext} + M_{aero}$$

    - The required deflection angles to fulfill ΔM can then be obtained by nonlinear inversion of the aerodynamic dataset (if possible)



Case Study: Aurora by DLR-RY

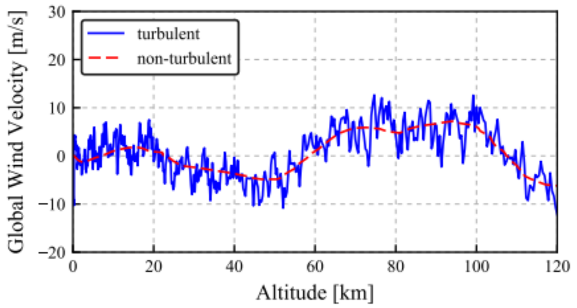
Results for Phase P5-c (wind effects)

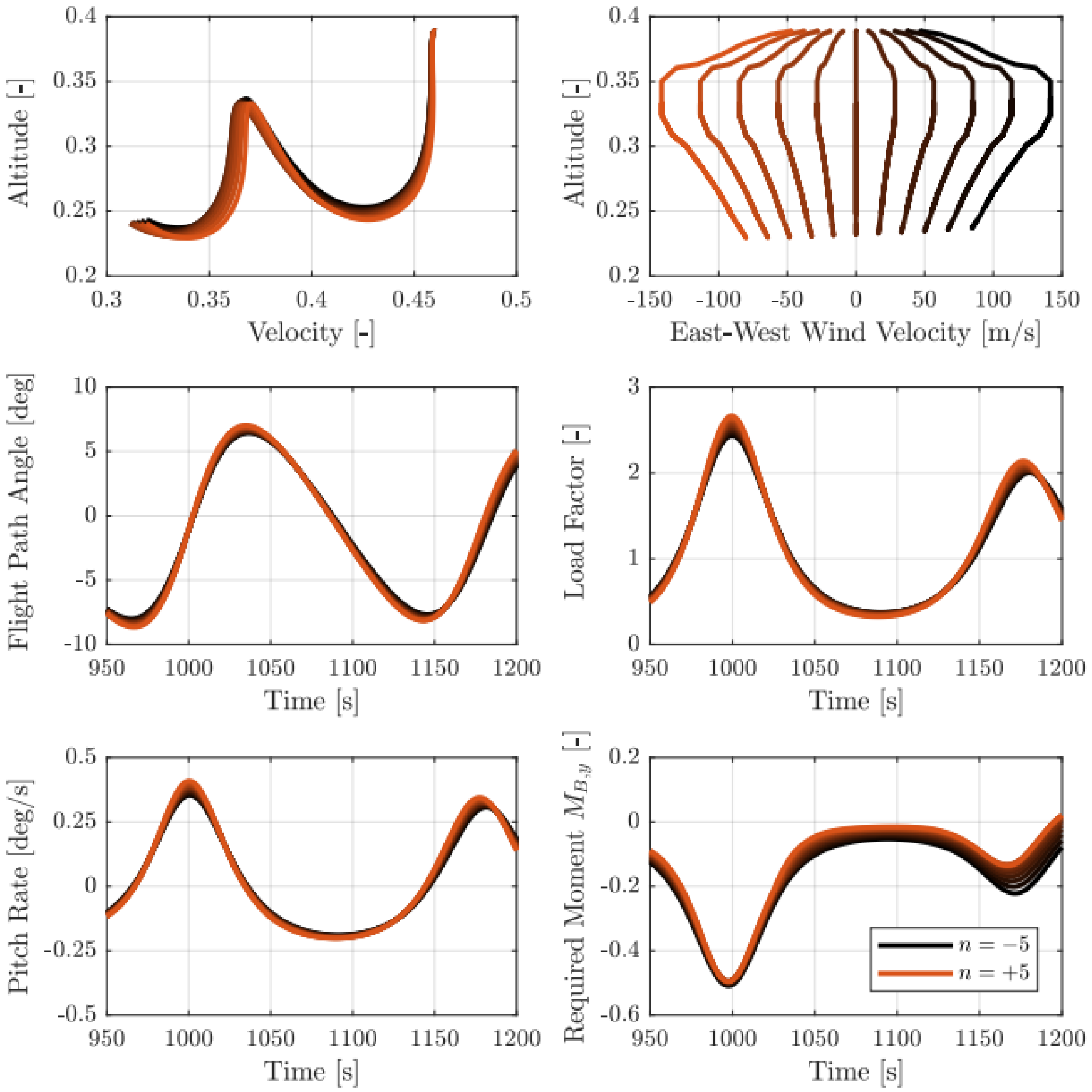
- Wind effects have major impact on the calculation of the effective aerodynamic angles and required rates

$$\boldsymbol{v}_w \to \boldsymbol{v}_{air} \to \begin{bmatrix}\mu\\ \alpha\\ \beta\end{bmatrix}^* \to \begin{bmatrix}p\\ q\\ r\end{bmatrix} \to \begin{bmatrix}M_x\\ M_y\\ M_z\end{bmatrix}$$

- The inverse kinematics compensates the disturbance introduced by additional wind velocities

- Additional turbulence models can be superimposed to the wind profiles, if necessary (not used in this paper)





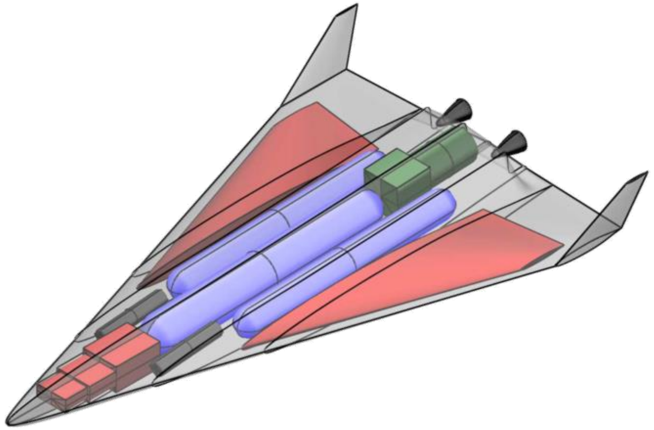
Overview

1. Object-oriented Modeling with Modelica / Dymola
2. Launch Vehicle Modeling, Guidance, and Control Framework
   - Launch Vehicle Modeling
   - Trajectory Optimization
   - Nonlinear Inverse Modeling
3. Simulation Case Study: Aurora Descent Flight
4. **Summary & Outlook**

Summary & Outlook

- **Objective**: Development of a multibody **modeling and simulation framework** for preliminary design studies of reusable launch vehicles
- **Method**: Multidisciplinary, object-oriented modeling using Modelica / Dymola, including (automatic) nonlinear model inversion
- **Results**: Trajectory optimization and moment budgeting of the reusable launch vehicle Aurora
- **Outlook:**
    - Implementation of external perturbations
    - Investigation of parametric uncertainties and structural elastic effects

DLR-SART SpaceLiner 7 - Copyright © DLR



DLR-SART Aurora, Copyright © DLR

DLR-SART LFBB, Copyright © DLR

Thank you very much for your attention!

Lâle Evrim Briese, Paul Acquatella B., Klaus Schnepper

DLR, German Aerospace Center
Institute of System Dynamics and Control
Oberpfaffenhofen, Germany