Planetary Entry System Synthesis Tool (PESST)

Richard E. Otero, Michael J. Grant, Bradley A. Steinfeldt
Advisor: Robert D. Braun

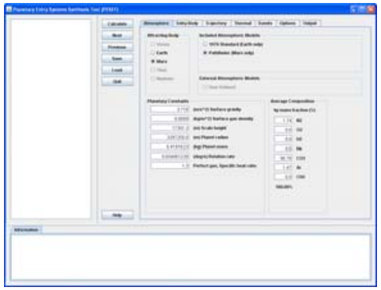
Abstract

The Planetary Entry Systems and Synthesis Tool (PESST) is a rapid conceptual design tool developed by the Space Systems Design Laboratory (SSDL) at the Georgia Institute of Technology. This framework has the capability to estimate the performance and mass of an entry system using user-defined geometry, hypersonic aerodynamics, flight mechanics, thermal response, and mass estimation. Earth and Mars atmospheres are preloaded with the ability to also use either user-defined or GRAM atmospheric models. Trade studies can be performed by parameter sweeps to gain an excellent understanding of the design space for conceptual studies. This framework is broadly applicable to conceptual studies of EDL systems with varying landed precision requirements.

Overview

- Conceptual design tool for entry, descent, and landing system analysis
- Written primarily using Fortran 95
  - Graphical User Interface written in Java
  - One guidance algorithm still using Matlab
- Provides a rapid sizing and synthesis framework suitable for entry system conceptual design
- Analysis includes:
  - Vehicle geometry
  - Hypersonic aerodynamics
  - Aerothermodynamics
  - Flight mechanics
  - Guidance, navigation, and control
  - Deployable aerodynamic decelerator performance
  - Mass sizing
  - Vehicle synthesis
- Directly integrates the relevant disciplinary analyses for entry, descent, and landing
  - First-order engineering models of each of the disciplinary analysis allow for fast computation
  - Appropriate data is shared between modules of the framework to allow for system convergence
- PESST can be used to
  - Determine required entry mass to achieve a desired landed mass
  - Perform parametric studies of driving design parameters to understand the design space

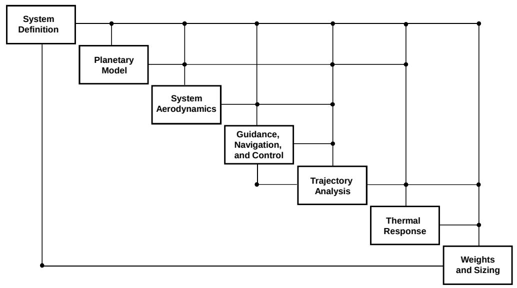
Graphical User Interface



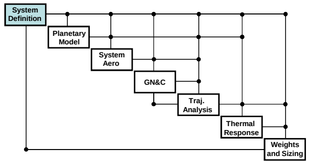
Current Architecture

- PESST has been migrated to a new architecture
  - The desired addition of a Monte Carlo and parametric capability placed a higher priority on the rapid completion of cases
  - In preparation for multi-processor servers, the new architecture utilizes a more flexible language for multithreaded programming
  - Interfaces for third-party modules required a design with strong extensibility
- Users desiring a graphical interface can use a Java GUI that writes the input files required by the PESST framework
  - Greater language flexibility allows for the use of object oriented design principles; promoting extensibility and maintainability
  - The strengths of Java in interface design and networking will be complemented by the speed and libraries available for Fortran 95
- Users can also access the capabilities of PESST directly from the command line
- A Matlab wrapper has been developed and is used to test or utilize modules that have not yet been ported to Fortran 95

Organization of tool



System Definition

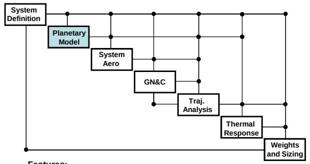


Purpose:
- To define system parameters and options

Inputs:
- User selects geometry
  - Sphere-cone
  - Biconic
  - Capsule
  - Microprobe
  - User defined from CAD file
- Planet
- Initial / target trajectory conditions
- TPS materials
- Events that will occur
- Initial mass estimates that will be used and converged during the iteration

Features:
- Geometry is created by a meshing library that aids in the consideration of arbitrary shapes
  - Internally stock shapes are defined as equations
    - E.g., $r^2 = x^2 + y^2 + z^2$ would mesh a sphere
  - Can accept user defined geometries through gts files which can be created from stereolithography (stl) CAD files
- Parachutes defined by a C_D vs. Mach input file

Planetary Model



Purpose:
- Model used to represent the environment present for the EDL sequence

Inputs:
- Planet to be used
  - Earth and Mars presently defined
  - User can modify planetary constants

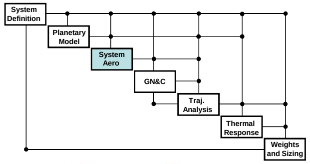
Outputs:
- Atmospheric characteristics
  - Density, pressure, temperature
- Planetary constants consistent with planet input
  - Sutton-Graves heating constant, surface gravity, etc

Assumptions:
- Uses Pathfinder atmospheric data for Mars when not using a GRAM model
- Atmosphere solely a function of altitude
- Spherical, rotating planet
- Inverse-square law gravitational force
- No wind

Features:
- Interfaces for GRAM models
- Arbitrary atmosphere input via tables

What is Planned:
- General heating models for a user defined atmosphere

Aerodynamics



Purpose:
- Provide approximate hypersonic aerodynamic information for the vehicle

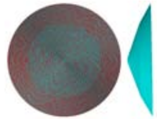
Inputs:
- A geometric mesh for the vehicle
- User selection to use either Newtonian or modified Newtonian aerodynamics

Outputs:
- C_L and C_D for the vehicle over a range of angle of attack

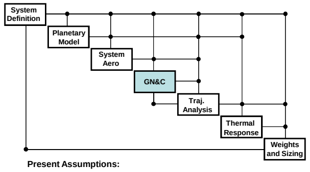
Assumptions:
- Hypersonic aerodynamics over the entire trajectory
- Real-gas effects not significant in conceptual design
- Subsonic and supersonic aerodynamics not significant in conceptual design

Features:
- Use of a high-quality surface mesh to generate hypersonic aerodynamics

What is Planned:
- Rarefied aerodynamics based on Kn number
- Aerodynamic moment calculations
- Additional stock shapes; including Inflatable Aerodynamic Decelerators



Guidance, Navigation, and Control



Purpose:
- Serve as a means to control the atmospheric flight path
  - Hypersonic flight
    - Modified Apollo
  - Propulsive terminal descent
    - Gravity turn
    - Analytic optimal control algorithm

Inputs:
- Vehicle initial state
- Target state
- Available control effectors
  - Bank angle
  - Thrust magnitude and direction
- Guidance mode(s) desired

Outputs:
- Appropriate control vector for guidance mode
  - Prescribed bank angle for hypersonic guidance
  - Thrust direction and magnitude for propulsive descent

Present Assumptions:
- Guidance called at the same frequency as the equations of motion
- Navigation is currently assumed perfect
- No control lag

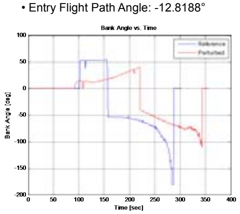
What is Planned:
- Subsonic guided parachute guidance

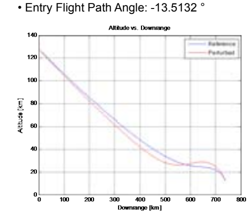
Hypersonic Guidance

- GT Modified Apollo Guidance
  - Terminal point controller that flies vehicle to same final range as that of a reference trajectory by controlling vertical lift via bank angle

Reference Trajectory
- Entry Velocity: 4370 m/s
- Entry Flight Path Angle: -12.8188°

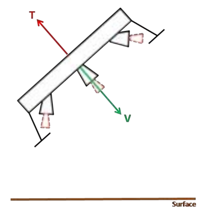
Perturbed Trajectory
- Entry Velocity: 4150 m/s
- Entry Flight Path Angle: -13.5132 °



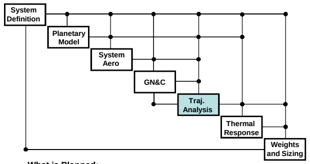


Propulsive Terminal Descent Guidance

- Analytic guidance law based on the work of D'Souza
  - Closed-loop guidance algorithm
  - Fuel-optimal assuming aerodynamic forces are negligible and the planet is flat and non-rotating
  - Targeting of terminal state position and velocity vectors
  - Initiation occurs at user-specified time (or conditions)
- Gravity turn guidance law
  - Targets end altitude and velocity but not a particular landing location
  - Approximately constant thrust guidance law where the thrust opposes velocity vector
  - Start occurs when the available thrust matches the required thrust to meet the specified end condition



Trajectory Analysis



Purpose:
- Numerically integrates the equations of motion to obtain atmospheric flight path

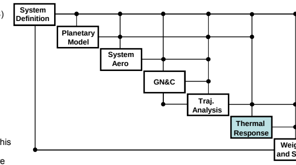
Inputs:
- Vehicle parameters (mass and aerodynamics)
- Initial state (position, velocity, bank angle)
- Events (triggering of vehicle changes, mass drops, etc.)

Outputs:
- Position
- Velocity
- Mass
- Convective stagnation point heating
- Radiative stagnation point heating
- Other standard atmospheric trajectory data

Present Assumptions:
- Spherical planet rotating about single axis
- Bank angle modulation decoupled from other attitude dynamics
- 3DOF + bank angle simulation only
- Single body propagation

What is Planned:
- Static trim calculation
- Parameter targeting and optimization capability
- Trade study parameter sweeps

Thermal Response



Purpose:
- Size the thermal protection system (TPS)

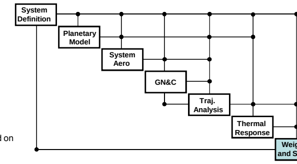
Inputs:
- Trajectory information regarding the convective and radiative heating environment
- TPS material desired
- Bondline temperature requirement

Outputs:
- TPS thickness required to handle the ablation that will be experienced during this EDL while still meeting the insulation requirements of the bondline temperature limit

Present Assumptions:
- Uniform thickness forebody heatshield sized to stagnation-point heating conditions

What is planned:
- Method optimizations to increase performance

Weights and Sizing



Purpose:
- Subsystem mass estimation

Inputs:
- Area covered by TPS
- TPS thicknesses and material
- Parachute deployment conditions
- Current estimate for the entry mass

Outputs:
- Mass estimates for subsystems
  - Structural mass uses MER based on Pathfinder, MER, and MSL projections
  - Propulsion system based on required fuel and engine requirements
  - Parachutes sized on the loads they will experience during deployment
  - Heatshield sizing based on medium fidelity one dimensional thermal ablation model

Present Assumptions:
- Vehicle mass estimating relationships derived largely from trade studies performed in support of the Mars Science Laboratory project

Current Uses for Synthesis Framework

- The new architecture enables batch runs from the command line by any system that can write or modify PESST input files which are also human readable
- Including system level sizing and synthesis capabilities allows the impact of the technologies enabling pinpoint landing to be examined at the mission design level
  - Closing the design about the thermal protection system, propellant mass fraction, and subsystem masses will enable effects to be examined on
    - Entry mass
    - Peak deceleration
    - Peak heating
    - Integrated heat load
    - Other mission level design constraints

Acknowledgements

The PESST framework has been under development within the Space Systems Design Laboratory (SSDL) at Georgia Tech for the past three years. Support for the PESST framework has been provided by NASA Langley Research Center, the Jet Propulsion Laboratory and the Charles Stark Draper Laboratory. The authors extend their gratitude to these three institutions for their insight and feedback throughout the PESST development cycle. Additional thanks are extended to the following SSDL graduate students for their contributions to PESST: Devin Kipp (JPL), John Dec (NASA Langley), Grant Wells and John Theisinger.