The Pointing Error Engineering Tool (PEET): From Prototype to Release Version

Marc Hirth[(1)] , Haifeng Su[(1)], Thomas Ott[(2)], Massimo Casasco[(3)], Sohrab Salehi[(4)]

[(1)] Astos Solutions GmbH

[(2)] Airbus DS – Space Systems

[(3)] ESA, ESTEC

[(4)] Rhea System for ESA

Outline

- Introduction
- PEEH Methodology in the Tool
- Software Overview
- “Advanced Statistical Method”
- Conclusion

Introduction

History

- 2008:Control performance standard ECSS-ST-60-10C
- 2011: ESA Pointing Error Engineering Handbook (PEEH)
- 2012: PEET prototype released
- 2013: Prototype update for formation-flying missions
- 2014-2016: Development of extended framework

Motivation

More and more stringent performance requirements in ESA projects (e.g. scientific or laser-communication missions)

Why PEEH?

- Necessity for clear and accurate pointing error engineering methodology

Why PEET?

- Systematic and user-friendly application of methodology
- Automated performance management process
- Replacement of “manual” computations
- Support dissemination of the methodology

PEEH Methodology in the Tool

PEEH Methodology

PEEH provides explicit guideline for requirement definition and evaluation of error budgets using dedicated analysis steps:

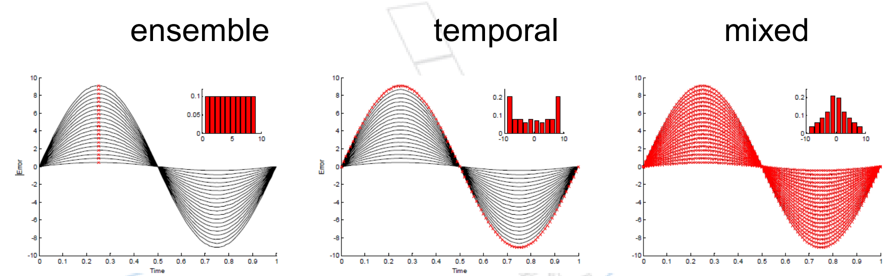
- "AST-0" : Requirement specification
- AST-1: Error source characterization
- AST-2: Transfer analysis
- AST-3: Error index contribution analysis
- AST-4: Error evaluation

The PEET workflow is fully compatible with the methodology and provides all related parameters and setup options

AST-0: Requirement specification

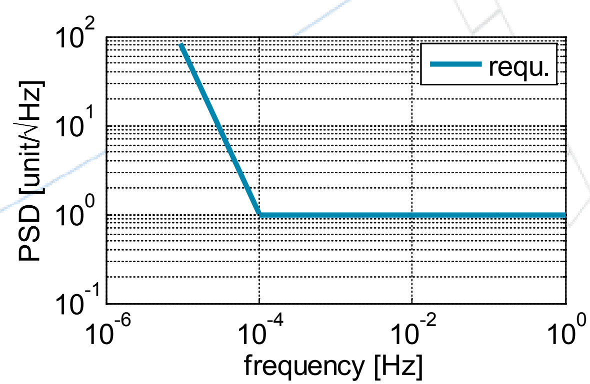
Unambiguous definition of requirements by parameters dependent on requirement type

- Statistical requirements
  - Max. error value (per axis or half-cone)
  - Related level of confidence
  - Metric for time-windowed errors
  - Statistical interpretation



Statistical interpretation (ECCS-ST-60-10C)

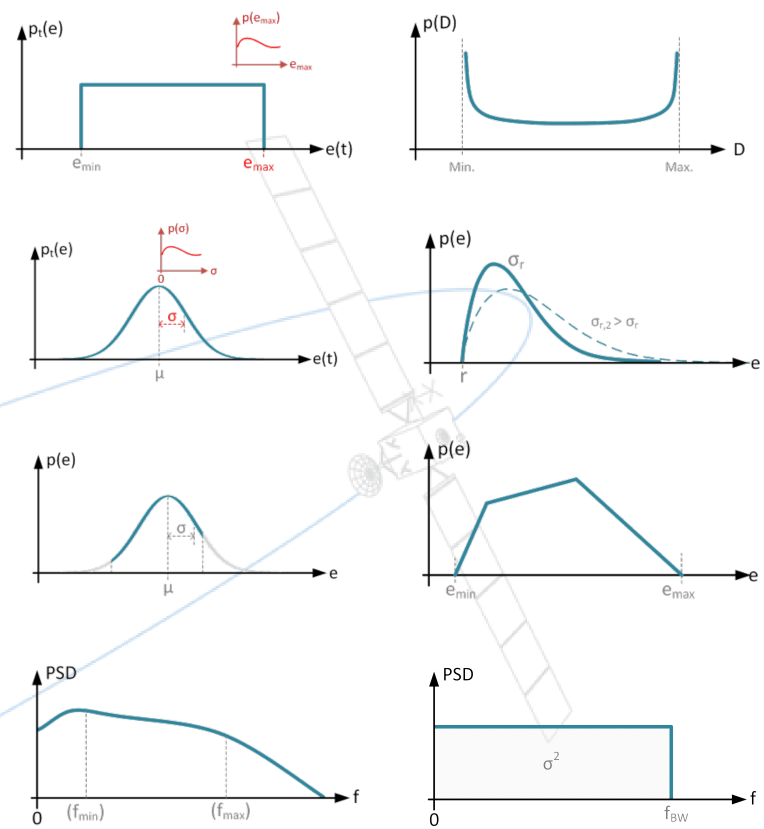
- Spectral requirements
  - Spectral requirement function (PSD upper bound)
  - (Metric for time-windowed errors)



AST-1: Error Source Characterization

Identification of potential error sources and classification based on characteristic properties:

- Random variables
  - Time-constant & Time-random
- Periodic errors
- Drift errors
- Random processes
  - BLWN or PSD

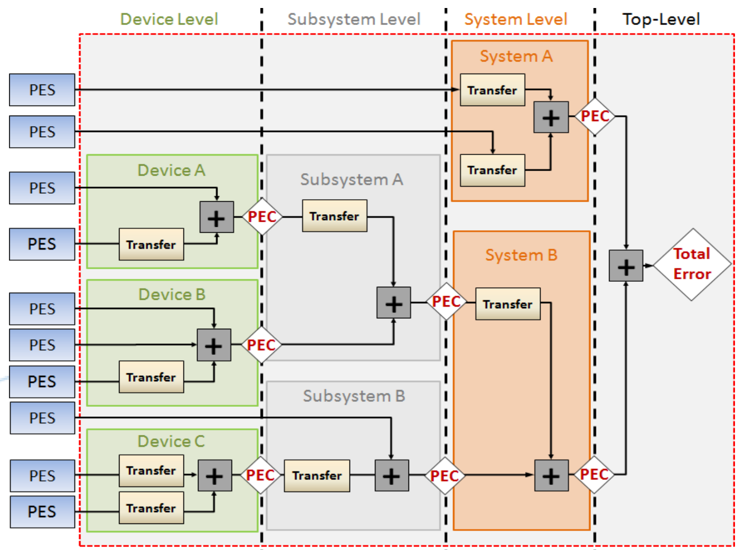


Statistical distributions describe temporal behavior and/or ensemble distribution of parameters

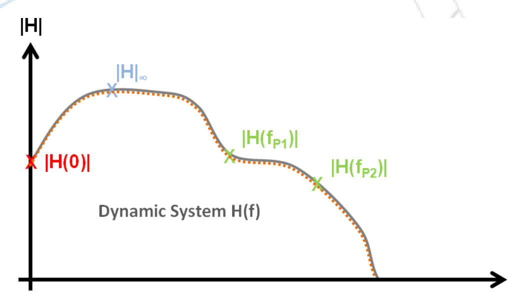
AST-2: Transfer analysis

Determines how initial error sources affect the figure of merit, i.e. the “route” to the final error contribution

- Static systems
  - Generic models
  - Coordinate Transformations, etc.
- Dynamic (LTI) systems
  - Generic models
  - Flexible plant, gyro-stellar estimator
- Feedback systems



Transfer rules dependent on error signal class



AST-3: Error Index Contribution Analysis

Astos Solutions

Impact of time-windowed errors for all metrics present in ECSS standard (e.g. APE, MPE):

- Frequency domain evaluation
  - Metric expressed as rational transfer functions
  - Applied to periodic and random process errors
- Evaluation according to ECSS rules
  - Random variables

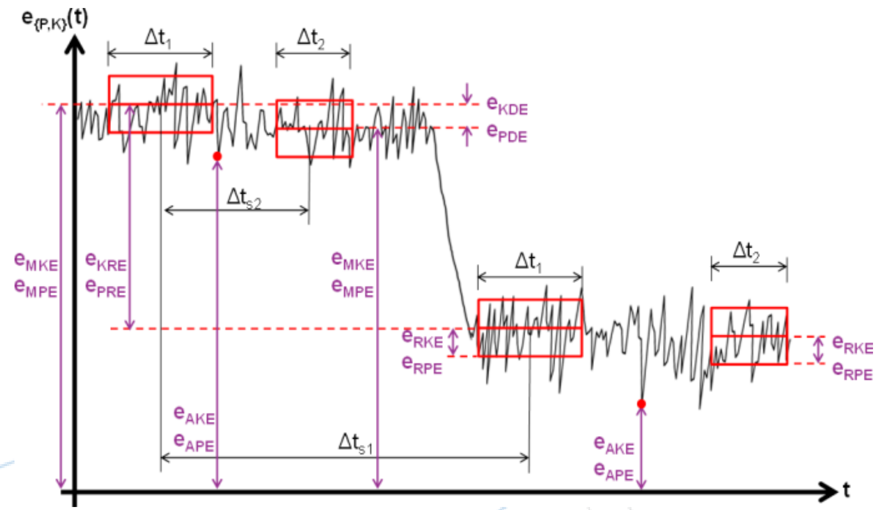
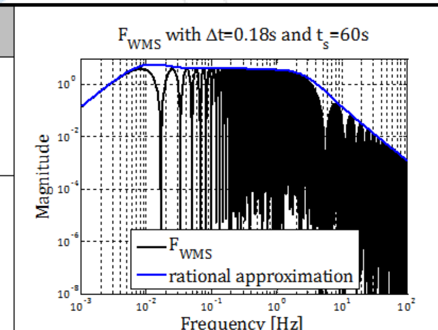


Table B-3: Budget contributions from zero mean Gaussian random errors

Index	S.I.	Distribution			Notes
		P(e)	μ(e)	σ(e)	
APE	E	$P(3s)=\frac{1}{3}P(s)$	3 μ	3σ	See text. For P(s), μ and σs see B.6
	T	$G(0, swc^2)$	0	swc	G(m,V) = Gaussian with specified mean & variance, swc = worst case s
	M	$\int G(0,s^2)P(s)ds$	0	⟨s⟩	For P(s), μs and <s> see B.6. For derivation see B.7

Windowed Mean Stability (WMS)
$F_{WMS}(\omega,\Delta t,\Delta t_s) = F_{STA}\times F_{WME} = \frac{4(1-\cos(\omega\Delta t_s))^2}{(\omega\Delta t)^2}$
rational approximation: $\tilde{F}_{WMS}(\omega,\Delta t,\Delta t_s) = \tilde{F}_{STA}\times\tilde{F}_{WM}$



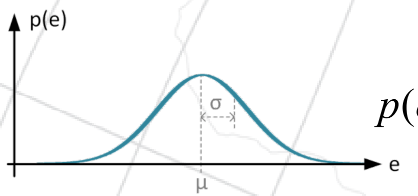
AST-4: Error evaluation

Summation of final error contributions according to the selected statistical method:

- Simplified statistical method (used in prototype)
  - Evaluation based on stat. moments (central limit theorem) and a confidence factor
  - Simplified correlation (full/no correlation)

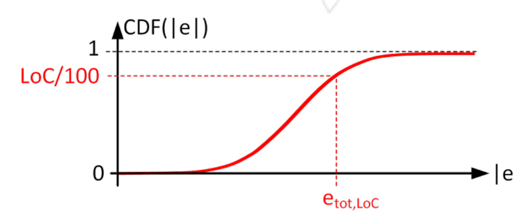
$$e(68\% / 95\% / 99.7\%) = \text{“}\Sigma|\mu_i| + n_p \cdot \Sigma\sigma_i\text{“} \quad , n_p = 1,2,3$$

- Advanced statistical method (used in release version)
  - Evaluation based on probability density functions & explicit level of confidence values
  - Specific correlation between contributions



$$p(e) = \left(\left| \sum_{i=1}^{N_D} e_i \right| \right)$$

$$e_{tot,LoC} = \int_0^{LoC/100} p(e)\, de$$



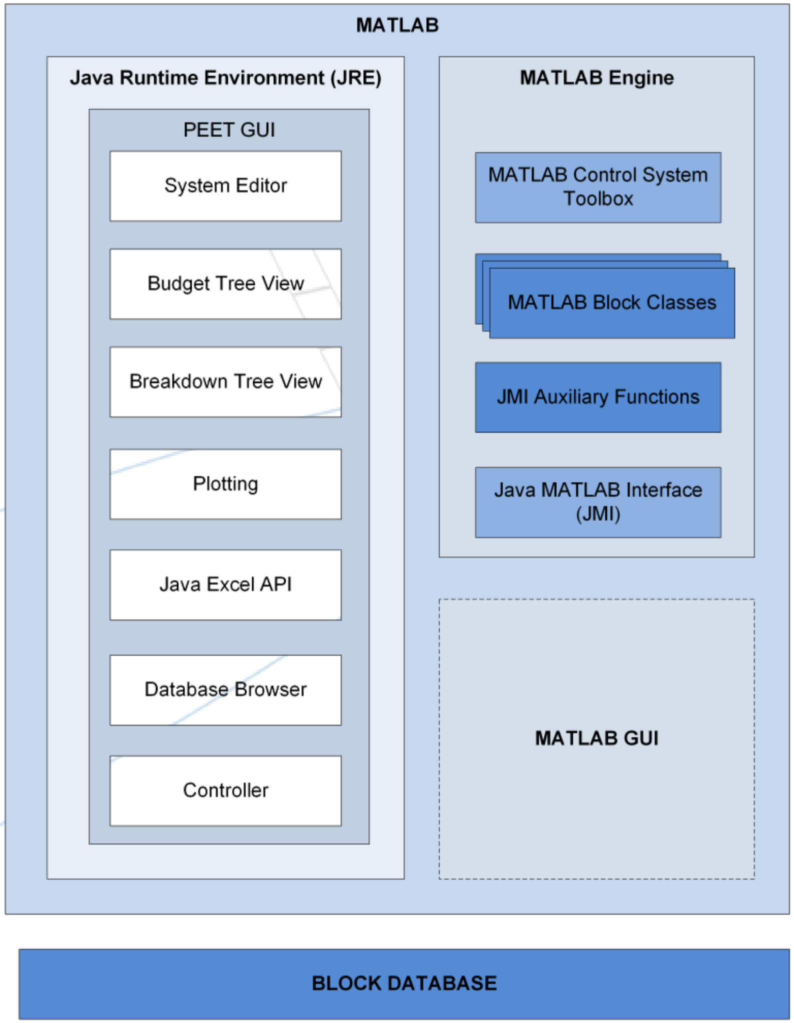
Software Overview

Platforms and Requirements

- Designed as extension to Matlab with following requirements:
  - Matlab 2011b and later
  - Control System Toolbox
- Multiple platform support
  - Windows
  - Linux
  - Macintosh
- No “critical” hardware requirements
  - Runs on standard desktop PCs and laptops

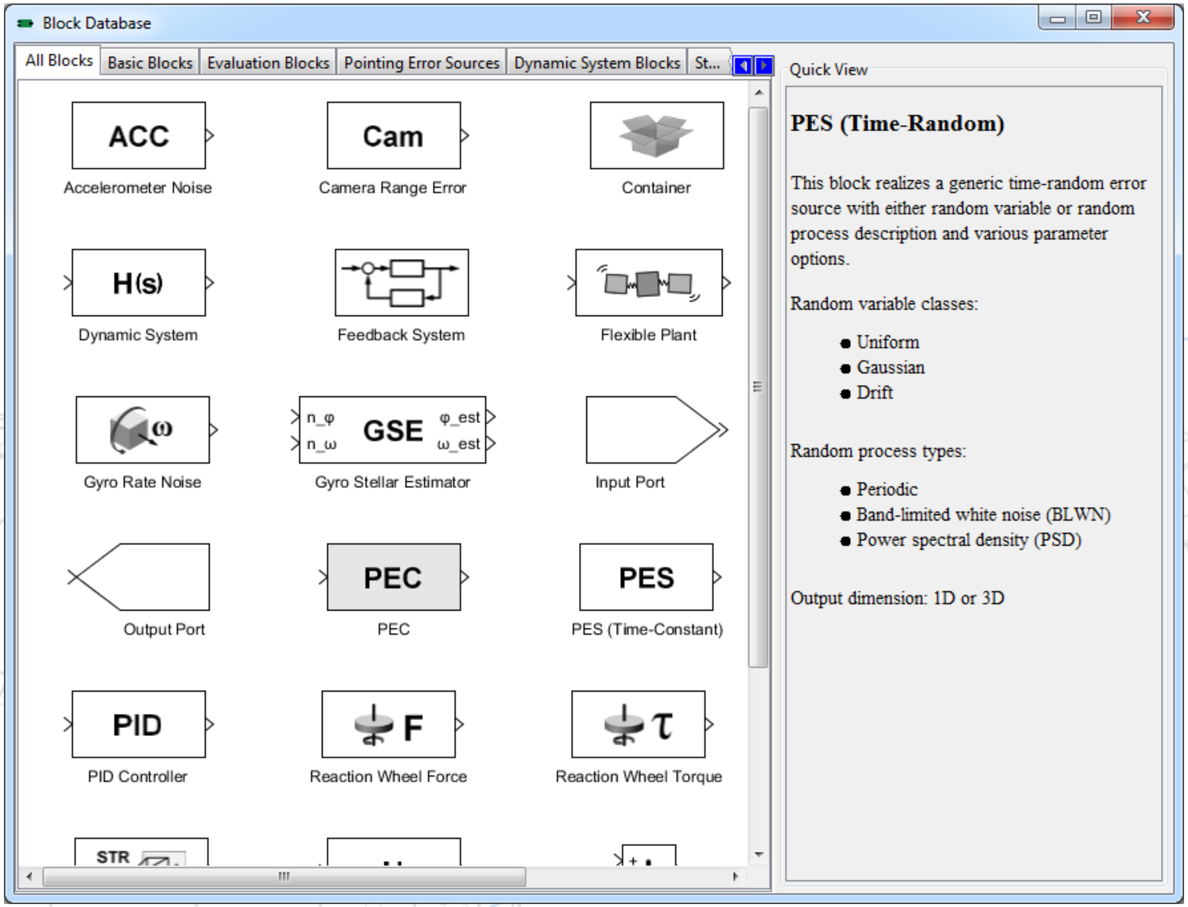
Architecture and External Interfaces

- Main components:
  - Java GUI
  - Matlab core algorithms
  - Communication via Java-Matlab Interface (JMI) and XML scenario definition files
- Interfaces:
  - Import from MS Excel spreadsheets
  - Links to Matlab workspace variables
  - Configurable report to MS Excel
- Operation modes:
  - GUI mode
  - Script-base execution controlled by (user-defined) Matlab scripts



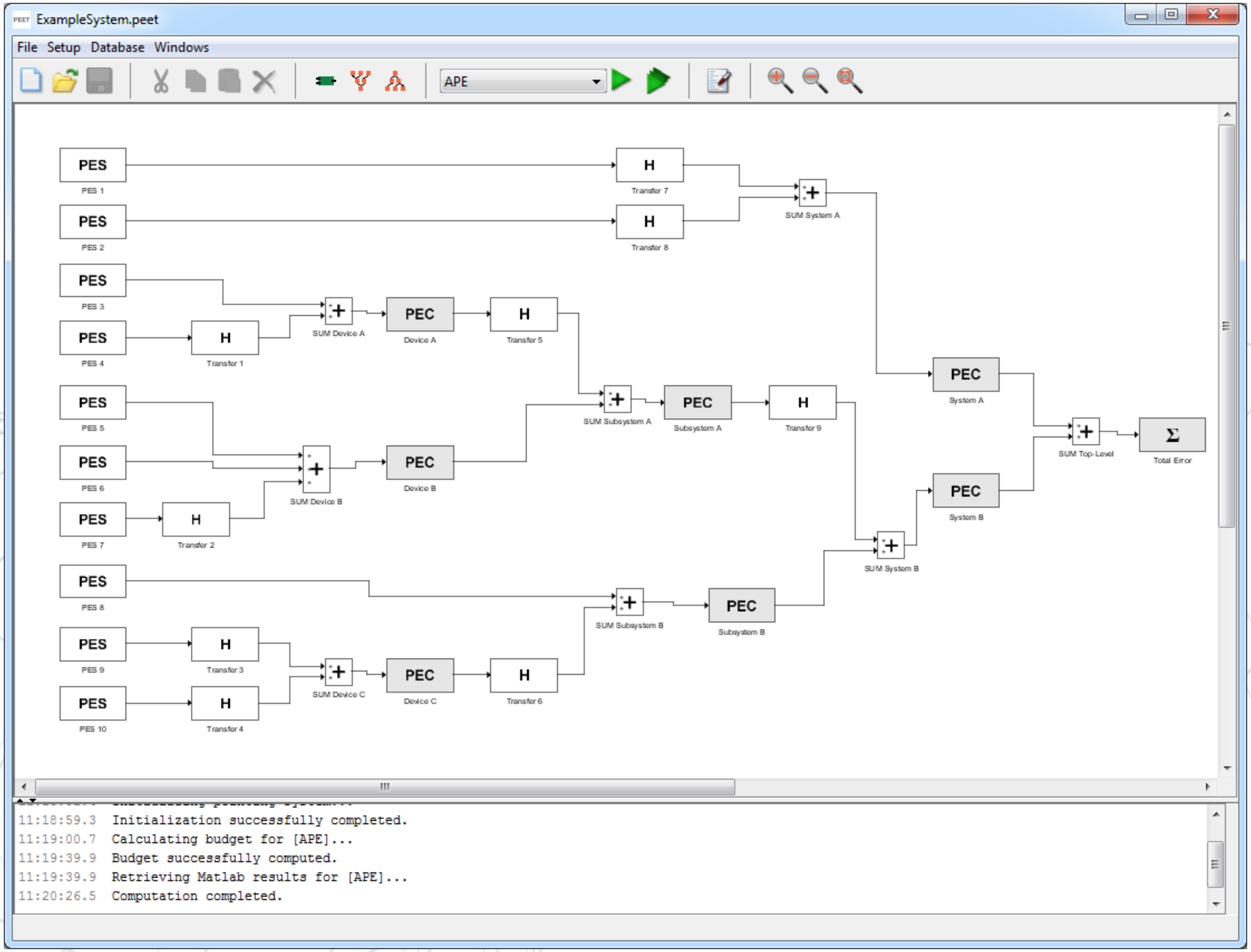
Graphical User Interface

Block Database: Select identified error source and system models



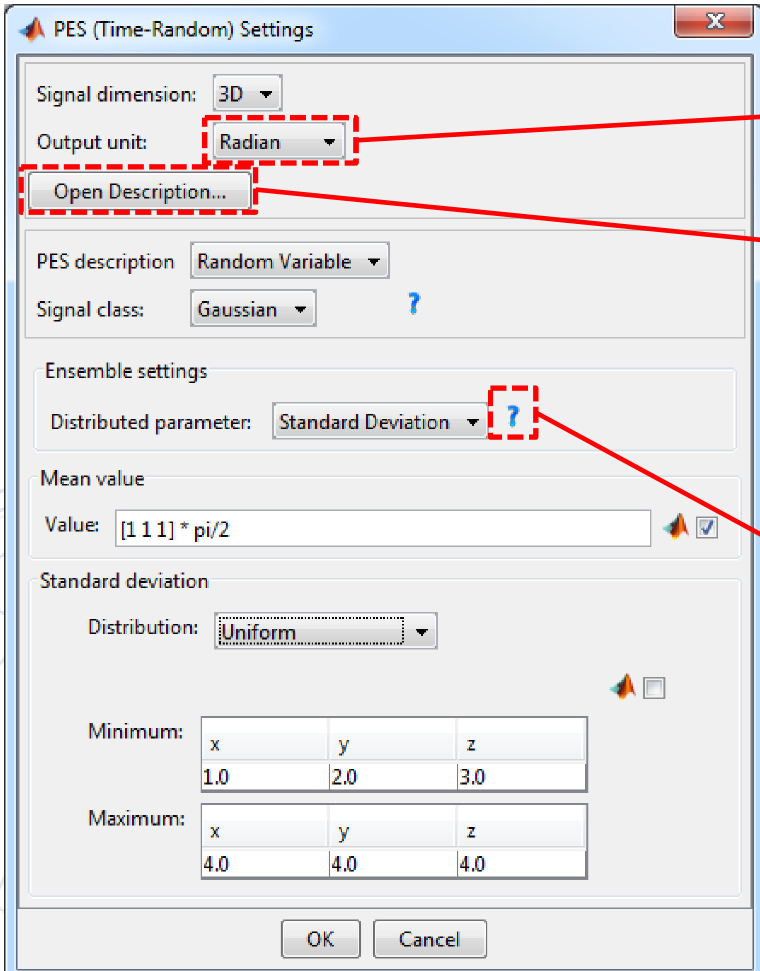
Graphical User Interface (cont'd)

System Editor: Define requirements & system interconnections



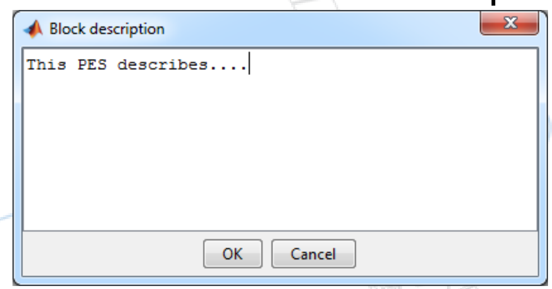
Graphical User Interface (cont'd)

Block Dialogs: Specify source/system characteristics

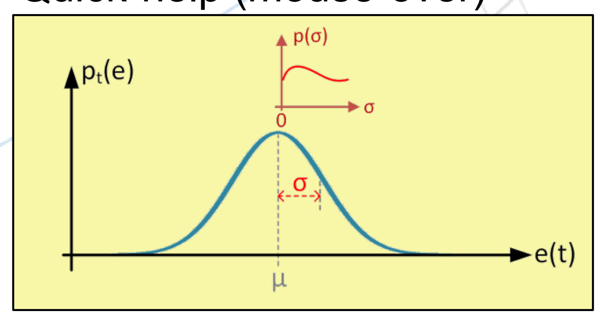


Custom or default units

User-defined block description

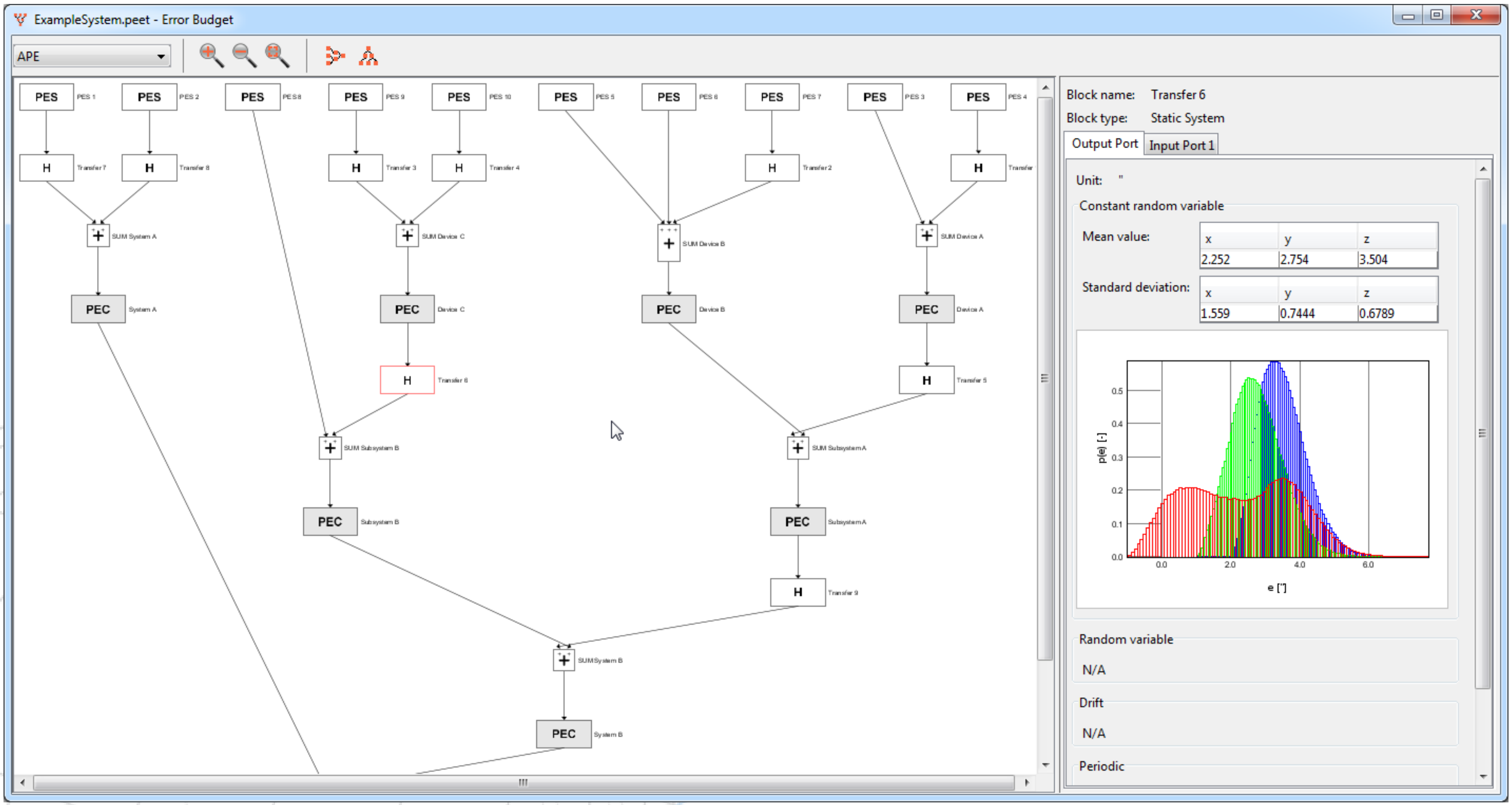


Quick-help (mouse-over)



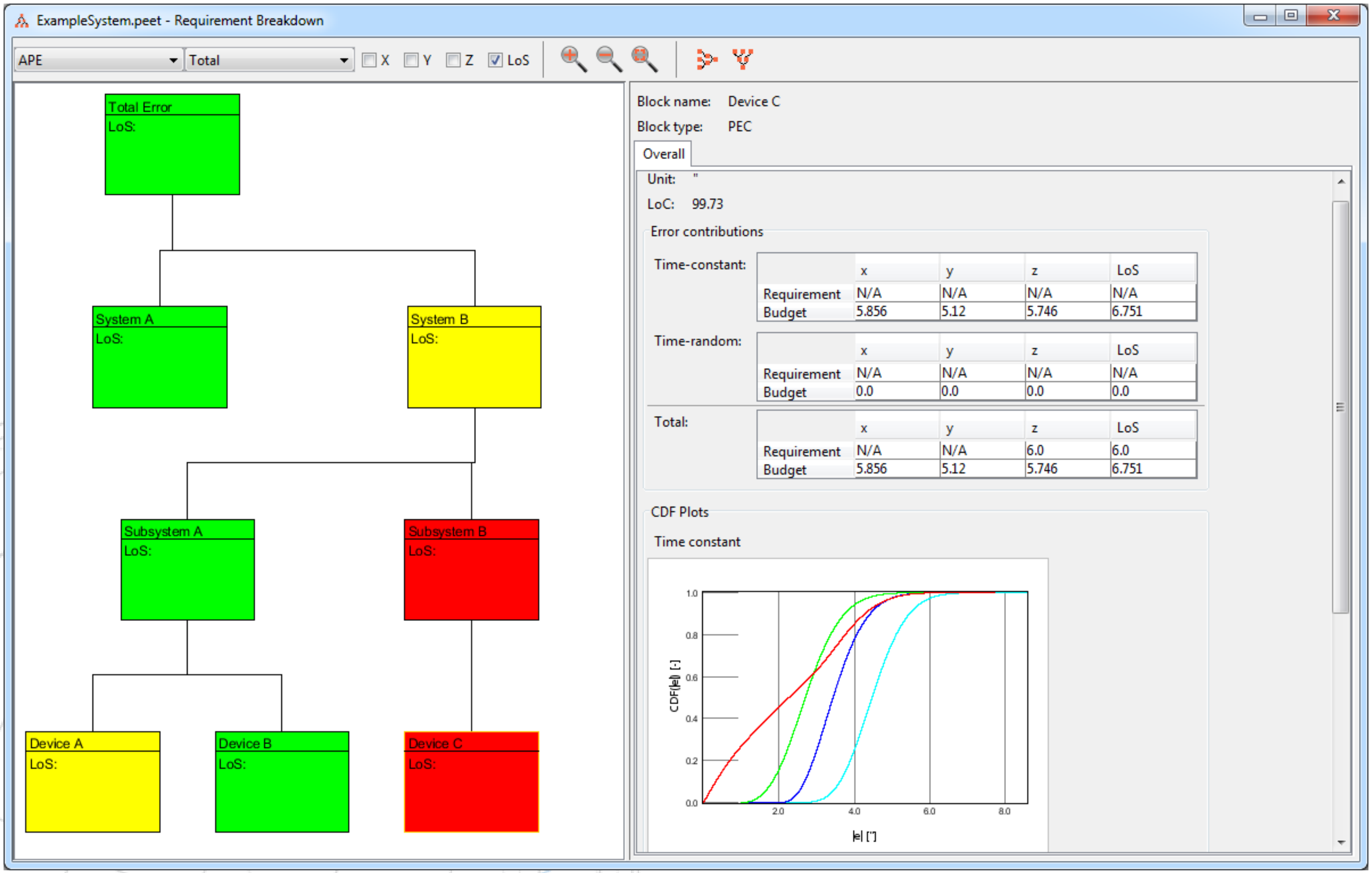
Graphical User Interface (cont'd)

Budget Tree View: Analyze statistics of error signal components



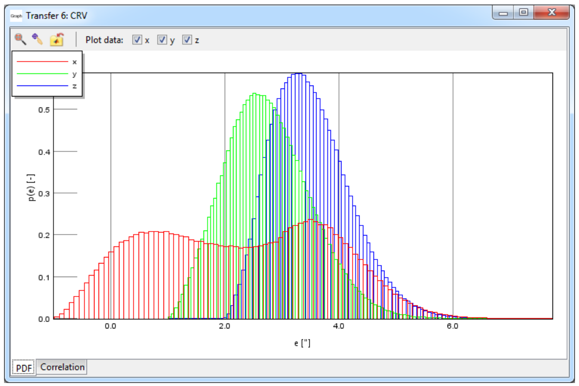
Graphical User Interface (cont'd)

Breakdown Tree View: Analyze compliance with requirements

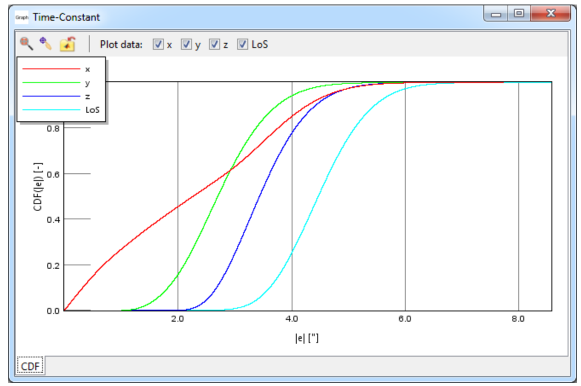


Graphical User Interface (cont‘d)

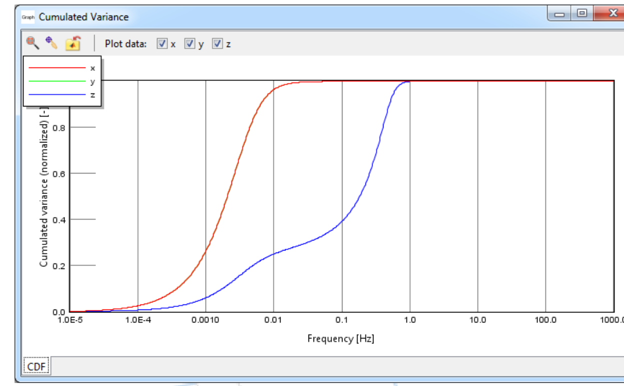
Plot Viewer: Detailed result inspection with various plot types



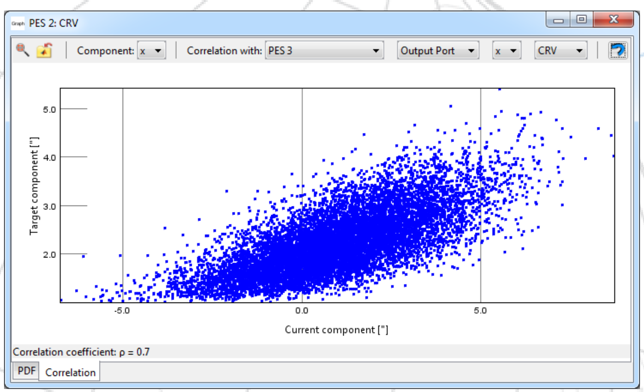
PDF plots



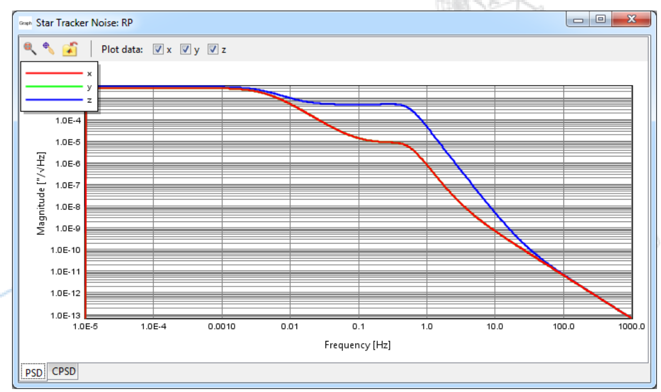
CDF plots



Cumulated variance (of random process)



Correlation „scatter“ plots



PSD plots (auto- and cross-spectra)

“Advanced Statistical Method”

Background

Simplified statistical method used in software prototype:

- Based on statistical moments (μ, σ) and confidence factors only
- Relies on applicability of central limit theorem

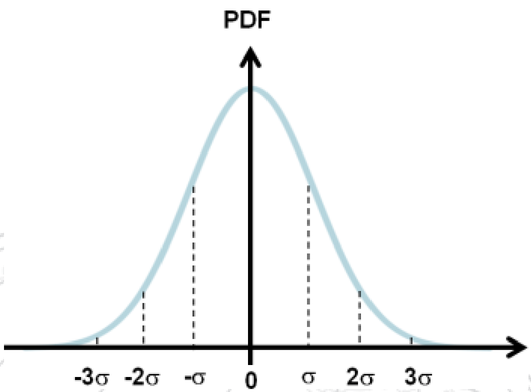
➡ For a valid level of confidence evaluation, total error needs to follow (at least nearly) a Gaussian distribution

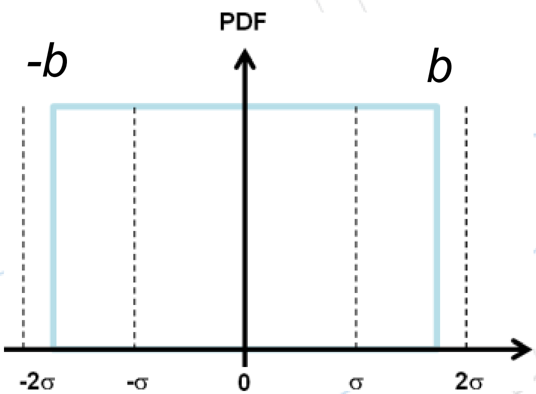
➡ If dominant non-Gaussian errors are present, this leads to significant systematic errors

Limitations of the Simplified Method (1)

Example:

„The error shall be smaller than X with 99.73% (3σ) probability“
(applied to a Gaussian and a uniform distribution)





$e = 3\sigma = 3\sigma_G$

➡ correct "by definition"

$e = 3\sigma = 3\sigma_U = \sqrt{3}\, b$

➡ Correct value: 0.9973 b

➡ ≈ 80% systematic error

➡ Even exceeds given bounds

Limitations of the Simplified Method (2)

Calculation of line-of-sight (LoS) errors:

- PEEH provides derived expression for instantaneous (or deterministic) LoS errors

$$e_{LOS} = \sqrt{e_x^2 + e_y^2}$$

- ECSS provides approximate solution solution for „statistical“ errors (valid for zero-mean Gaussian with closely equal σ)

$$e_{LOS} = \max(\sigma_x, \sigma_y)\sqrt{-2\log(1-p_c)}$$

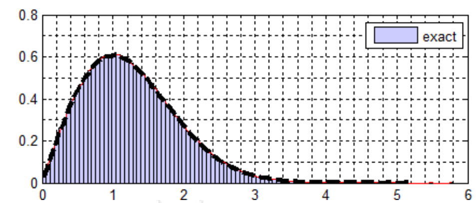
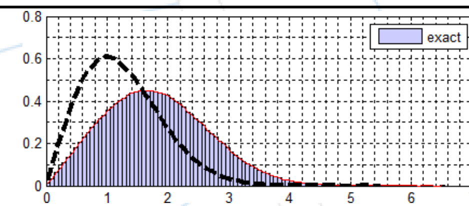
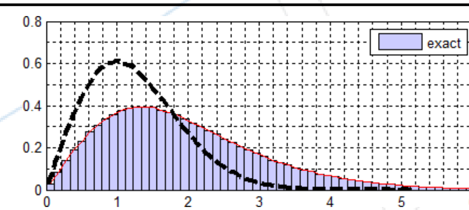
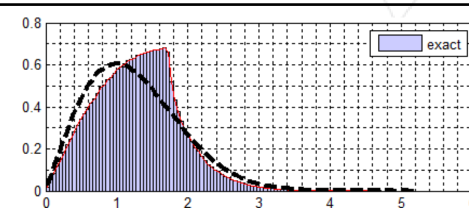
- Exact description via PDF

$$e_{LOS} = \int_0^{LoC} p\left(\sqrt{e_x^2 + e_y^2}\right) de$$

➡ “Careless” application of the first expressions to non-matching conditions again leads to systematic errors

Limitations of the Simplified Method (3)

Calculation of line-of-sight (LoS) errors: 68,3% LoC

Case	e_{LoS}			PDF
	“Deterministic”	ECSS approx.	“Exact” (num.)	
X: G(0,1) Y: G(0,1)	-7%	0%	1.5158	
X: G(1,1) Y: G(1,1)	30%	-30%	2.190	
X: G(0,1) Y: G(0,2)	-22%	+30%	2.305	
X: U(-√3, √3) Y: G(0,1)	-10%	-3%	1.57	

Implementation (1)

Analytical treatment (PDF convolution and CDF determination) in conflict with several constraints:

- SW requirement: No further MATLAB toolboxes shall be used (e.g. the Symbolic Toolbox)
- Even with symbolic computation, closed-form solutions could not be guaranteed (for arbitrarily complex systems)
- With a numerical description of the PDF convolution, the joint PDF of all error sources is required
  - Usually not known by the user
  - In best case, knowledge of (marginal) PDFs and correlation in terms of correlations coefficients expected

Implementation (2)

Therefore, the chosen approach is entirely numerical:

- Sample-based (around 1e6 samples per source, 10000 PDF bins)
- Dedicated inverse transform sampling method allows both „imprinting“ correlation and PDF information

→ see paper for more information

- Intrinsic drawback: loss of accuracy with respect to the analytical computation
  - Error expected to < 1% in entire computation chain
  - Thus safely negligible compared to potentially large systematic errors of simplified method

Conclusion

Conclusion

- PEET is a tool to accurately compute statistical and spectral error budgets (release: mid 2016).
- PEET is not restricted to pointing applications, but can also be used in other engineering fields.
- PEET is well-suited for integration in analysis tool chains with the available Excel & Matlab interfaces.
- PEET files provide much better transparency (model assumptions) and flexibility (model adaption) than purely "tabular" budgets.
- Limitation: PEET cannot explicitly account for non-linearities and transient system behavior, i.e. it shall not be understood as a replacement for E2E simulators.