GNC 2017 : 10th International ESA Conference on Guidance, Navigation & Control Systems
29 May - 2 June 2017, Salzburg, Austria
European Space Agency

PointingSat – High Precision Pointing Error Analysis with ESA PEET v1.0

Ott T.[(1)], Hirth M.[(2)], Casasco M.[(3)], Goerries. S.[(1)], Gedon D.[(1)], Ponche A.[(1)]

[(1)]*Airbus, Friedrichshafen, Germany*
Thomas.Ott@airbus.com
[(2)]*Astos Solutions, Stuttgart, Germany*
[(3)] *ESA/ESTEC, Noordwijk, The Netherlands*

ABSTRACT

In 2011 ESA published the ESA Pointing Error Engineering Handbook as applicable document. The Handbook complements the ECSS control performance standard. It provides guidelines for a step-by-step engineering process from pointing error requirement specification, to systematic pointing error analysis, and the compilation of pointing error budgets. As pointing error engineering is relevant to any space mission, ESA developed the Pointing Error Engineering Tool PEET software to support the user in applying the elements in the Handbook and in the ECSS control performance standard. The prototypes of PEET (v0.X) have been in use by several projects in Airbus. Among others, this is the case for the MetOp-SG project, where PEET is the tool used for pointing error analysis of 144 instrument and platform budgets. Based on that experience Airbus supported Astos Solutions to further develop PEET from a prototype to a release v1.0 in an ESA GSTP study. This included the cross-validation of the software.

This paper demonstrates the capabilities of PEET v1.0 and its cross-validation by means of the PointingSat case study. PointingSat is a fictive but realistic mission that covers almost all use cases for the ESA PEET software v1.0. The mission has demanding pointing error requirements in all possible ECSS error indices. The corresponding PEET model of PointingSat has a high level of detail in the modelling of pointing error sources and system transfers of the pointing errors from their origin to the axis of interest. The paper will show step-by-step how to model and analyse PointingSat in PEET. This includes the pointing error source characterisation, the modelling of the system transfer and the pointing error evaluation w.r.t. the requirements.

Finally, the results of the PEET v1.0 cross-validation w.r.t. an equivalent state-of-the-art Monte Carlo simulation in Matlab Simulink will be presented. The cross-validation showed that the PEET v1.0 software produces the same results as the equivalent Monte Carlo simulation for the “advanced statistical method”, as it is called in the ESA pointing error engineering handbook. PEET v1.0 also produces the same results as the latest PEET prototype v0.6, but for the so called “simplified statistical method” as the prototype is limited to this method. In the benchmarking the achieved computational accuracy is comparable, but the computational speed of PEET v1.0 is >10 times faster than the equivalent Monte Carlo simulation. In addition the gain of accuracy by going from the “simplified statistical method” in PEET v0.6 to the “advanced statistical method” in PEET v1.0 is demonstrated.

1 ESA POINTING ERROR ENGINEERING

The ESA Pointing Error Engineering framework used as reference in Europe is described hereafter based on the summary and discussions in [9]. The framework is defined in the ESA Pointing Error Engineering (EPEE) Handbook and the ECSS standards and handbooks in the E-60 discipline of control engineering, which are available at [4]. An overview of current ECSS and ESA documents in the E-60 discipline is given in Figure 1. The EPEE Handbook is based on the ECSS standards and handbooks and complements those by providing practical guidelines and a step-by-step process. The ECSS-E-ST-60-10C [3] and ECSS-E-HB-60-10A [5] are the most relevant ones for pointing error engineering. The E-ST-60-20C [6] and the E-ST-60-21C [7] are relevant for describing Pointing Error Sources (PES) inherent in a star sensor or gyro.

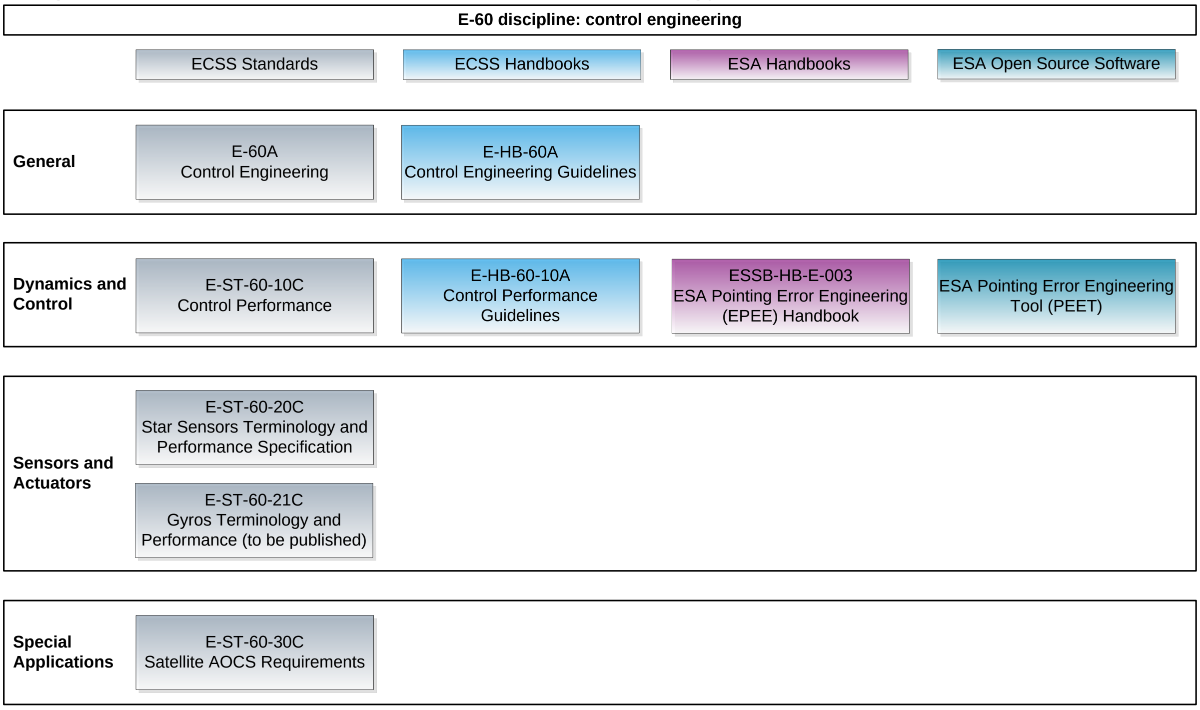


Figure 1: ECSS and ESA documents relevant for pointing error engineering

The ECSS documents provide an approximate pointing error engineering approach covering the analysis steps AST-1, 3 and 4 in [1]. But they do not provide an approach with proper level of accuracy for high accuracy pointing missions. In this case a more accurate approach is needed that also covers AST-2 and AST-1 in more detail. The EPEE Handbook addresses this need and provides accurate modelling techniques for describing PES with their frequency domain properties in AST-1. By modelling and analysing the frequency domain properties an exact error index contribution can be determined. As introduced in [1] these techniques are based on various publications that trace back to the initial paper of [8]. The ESA pointing error engineering tool (PEET) in [10] has been developed to support the application of these techniques and goes even beyond in the implementation of high accuracy computational methods as described in chapter 2. However, the main purpose of PEET is to guide and support the pointing budgeting and analysis in general by being conform to all ECSS and ESA documents. Before the release of the EPEE Handbook in the year 2011, the ECSS-E-ST-60-10C and ECSS-E-HB-60-10 were generally required in projects at Airbus. After the year 2011 the EPEE Handbook together with the PEET software have been applied in new projects and studies, like MetOp-SG, Euclid, EDRS, LOFT, XIPE, ATHENA.

2 HIGH ACCURACY FEATURES OF ESA PEET V1.0

2.1 Overview

In chapter 2.2 to 2.5 a summary of the main PEET v1.0 software features is given that enable a high precision pointing error analysis. Those features are in line with the ECSS standard [5] and the EPEE handbook [1] on pointing error engineering.

2.2 Simplified versus Advanced Statistical Method

The EPEE Handbook [1] provides two analysis methods, the Simplified Statistical Method (SSM) and the Advanced Statistical Method (ASM). The simplified method is based on the applicability of the central limit theorem for the combination of pointing error source statistics. The central limit theorem states that the sum of a large number of independent distributed random variables converges to a Gaussian distribution. This is illustrated in Figure 2, which shows an example of the summation of uniform distributions (for n={1;2;3;4} from the left to the right).

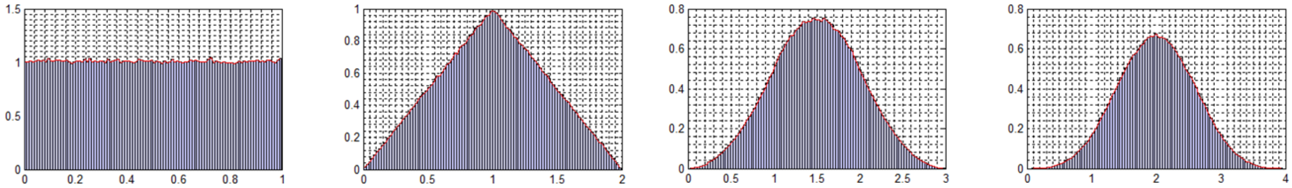


Figure 2: Simplified Statistical Method - Example: Sum of uniform distributions converge to Gaussian distribution

If the central limit theorem applies, all PES can be entirely described only via their basic statistical moments (mean and variance) neglecting their real underlying probability density function (PDF). These moments are exact statistical quantities, even after summation of different error sources with arbitrary PDF during the systems transfer (AST-2 in [1]).
However, the level of confidence evaluation related to AST-4 of [1] is only accurate, if the final error contribution has at least a close-to-Gaussian distribution. Then the equivalence of 1σ (2σ, 3σ,...) values with a confidence level of 68.3% (95.5%, 99.7%,...) is applicable. In all other cases where a dominant non-Gaussian contribution is present, proper evaluation of the level of confidence requires the knowledge of the underlying PDF. For instance, for a single uniform error contribution the 2σ value computed with the SSM already exceeds the possible bounds of the real error signal as illustrated in Figure 3.

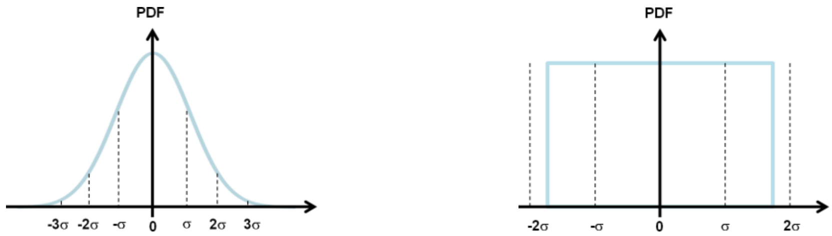


Figure 3: Error evaluation with confidence coefficients - correct results for a Gaussian distribution (left) and significant deviation for non-Gaussian distribution (right)

If the central limit theorem does not apply, then the ASM is the choice for high accuracy pointing error analysis. The ASM uses exact PDF (and f-domain) information in the analysis. It maintains and propagates the information of the underlying PDF from each PES (and their combinations during the system transfer) to the final error contribution. The summation of the PES is done via convolution and leads to a joint distribution. The following equation describes a joint distribution for two PES:

$$p_{e_1+e_2}(e) = \int p_{e_1}(e_1)\, p_{e_2}(e-e_1)\, de_1 = \int p_{e_2}(e_2)\, p_{e_1}(e-e_2)\, de_2$$

In addition, frequency domain information of the PES is propagated analytical from their origin in a system to the final error. The implementation of the ASM in PEET v1.0 is done in a numerical approach because analytical solutions for the integrals are hard to obtain or do not even exist, cf. [10].

The final pointing error value with a specified level of confidence is then determined by integration of the final error PDF. Obviously a certain numerical error is introduced in the evaluation when deriving the PDF from the histogram of random samples. For a sufficiently large sample size (around 1e6 samples), this computational error is <1% with respect to an exact analytical solution. Compared to the gain in accuracy by using the ASM, this computational error is considered negligible and completely tolerable. This is shown in the following example:

Example: Uniform distribution *p(e)* = *U*(-1,1) and a 99.7% level of confidence

Analytical result with ASM:

$$e_{tot,LoC} = \int_0^{0.997} \left|U(-1,1)\right| de = \int_0^{0.997} U(0,1)\, de = \int_0^{0.997} \frac{1}{1-0}\, de = 0.997$$

Simplified method:

$$e_{tot,n_p} = \left|\mu_{tot}\right| + n_p \cdot \sigma_{tot} = \left|\frac{1+(-1)}{2}\right| + 3 \cdot \sqrt{\frac{\left(1-(-1)\right)^2}{12}} = 1.7321$$

i.e. analytical result + 73.73% systematic error.

2.3 Line-of-Sight Probability Density Function

Having the pointing error PDF of each axis, PEET v1.0 can compute the LoS pointing error PDF based on it. This numerical but otherwise exact computational approach is more accurate than taking any other approaches that are only valid under certain restrictions; cf. [1] and [5]. Depending on the nature of the PES, the restricted approaches lead to either conservative or optimistic results, see Figure 4. The restricted approaches thus do not serve as upper bound estimates.

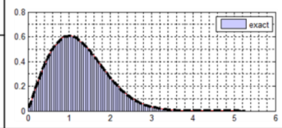
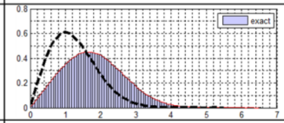
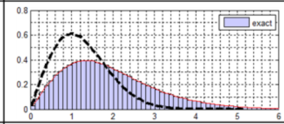
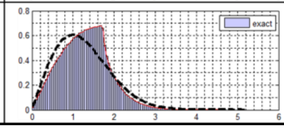
Case	e_{LOS}			PDF
	$\lvert e_{LOS}\rvert = \sqrt{e_{x\theta}^2 + e_{y\phi}^2}$	ECSS approx.	"Exact" (num.)	
X: G(0,1) Y: G(0,1)	1.4142	1.5158	1.5158	
X: G(1,1) Y: G(1,1)	2.8284	1.5158	2.190	
X: G(0,1) Y: G(0,2)	2.2361	3.0316	2.305	
X: U(-√3, √3) Y: G(0,1)	1.4142	1.5168	1.57	

Figure 4: Different approaches for computing the LoS PDF

2.4 Frequency-Domain Analysis

The frequency domain analysis for pointing error propagation and evaluation in PEET v1.0 complements the PDF computations and thus provides a complete characterization of the pointing. It has already been introduced in detail and in the context of ECSS in [2]. Hereafter a summary is given.

The main advantage of the frequency domain analysis is the exact correspondence of the time-windowed statistics of a time-series and the integration of a corresponding weighted power spectral density (PSD) as shown in Figure 5. Instead of generating time-series in simulations, one can simply work with analytical expressions. This enables responsive analysis, cf. [9].

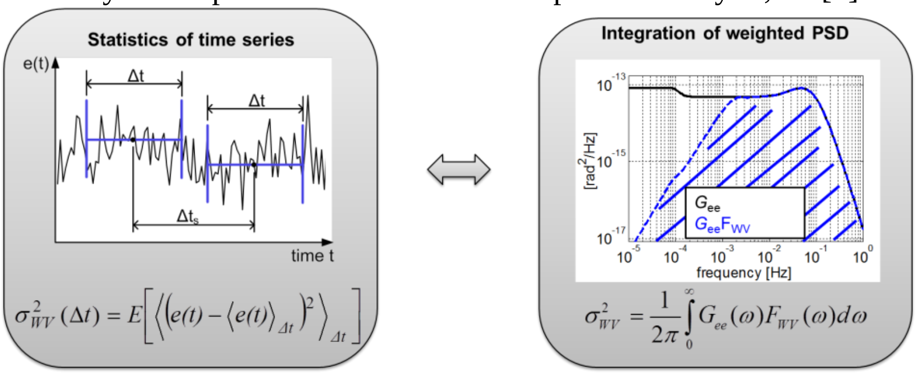


Figure 5: Exact correspondence of the time-windowed statistics of a time-series and the integration of a corresponding weighted PSD

This analytical approach can also be used for propagating a PES through the pointing system of interest via the following simple equation:

$$\mathbf{G}_{yy}(\omega) = \mathbf{H}(s)\mathbf{G}_{uu}(\omega)\mathbf{H}^*(s)$$

with: $\mathbf{H}(s)$ MIMO transfer function of physical system
$\mathbf{G}_{uu}(f)$ PSD matrix of input signals $\mathbf{u}(t)$
$\mathbf{G}_{yy}(f)$ PSD matrix of output signals $\mathbf{y}(t)$

2.5 Concept of Statistical Domains

The concept of statistical domains in PEET v1.0 is necessary for the accurate assessment of physically and probabilistically meaningful correlation options between different types of error sources and a more flexible definition and evaluation of pointing error requirements.

Ref. [1] clearly distinguishes between time-constant and time-random error sources and according to the summation rules. This implicitly splits the contributions to the total error already in two domains, "Time" and "Ensemble", which are separately evaluated. Between these domains, no correlation can be specified as they have physically nothing in common (e.g. the distribution of a misalignment and the distribution of the temporal noise of a sensor).

The temporal domain is common (“global”) for the error evaluation, however different ensemble domains could exist. For instance, ensemble random contributions could be assigned to domains such as “Manufacturing” (misalignments, displacements, multiple satellites, etc.) or “Observations” (error contributions that do not vary in time over a single observation, but due to varying conditions between different observations).

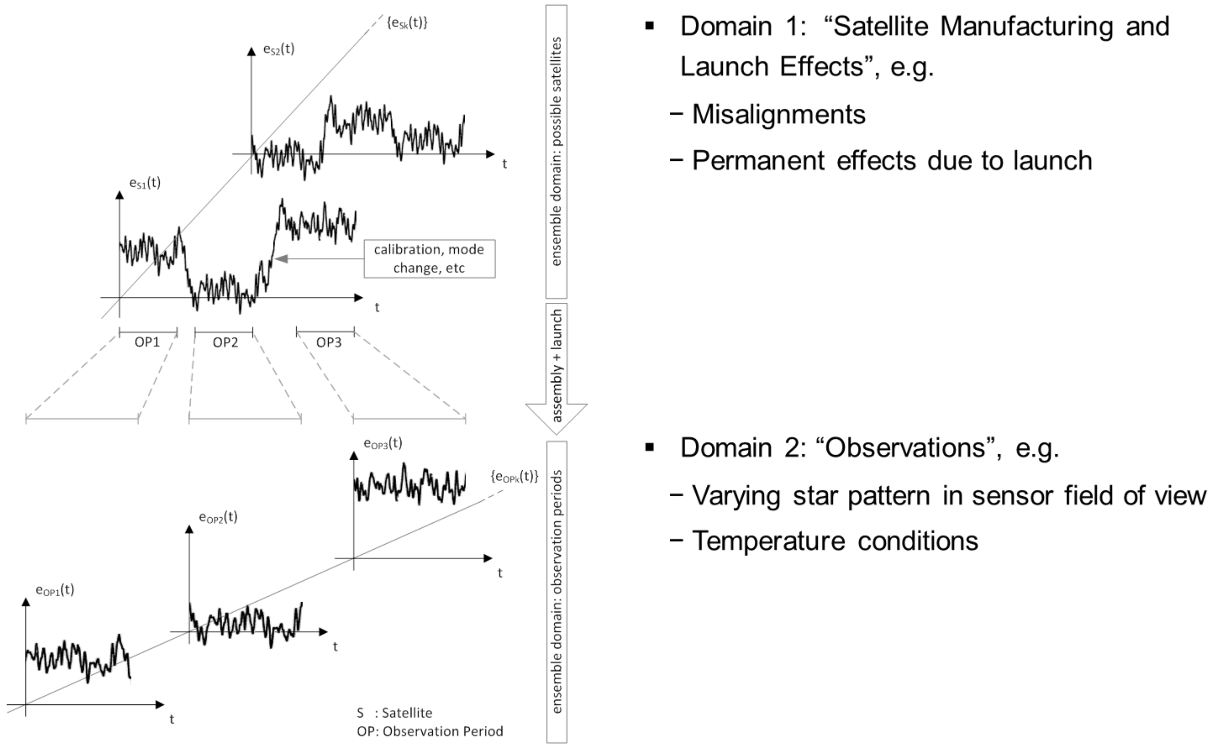


Figure 6: Different Ensemble Domains in PEET v1.0

These domains are independent by definition, and consequently no correlation is meaningful between them. Furthermore, a tailored treatment for these domains is possible in terms of requirement specification which is – most importantly - compliant with the rules and methods in [1] and [5].

In Figure 7 an example is given for a general PES (e.g. noise in electronics) that has a Gaussian PDF with a variance that is uniformly distributed e.g. due to different operational temperatures.

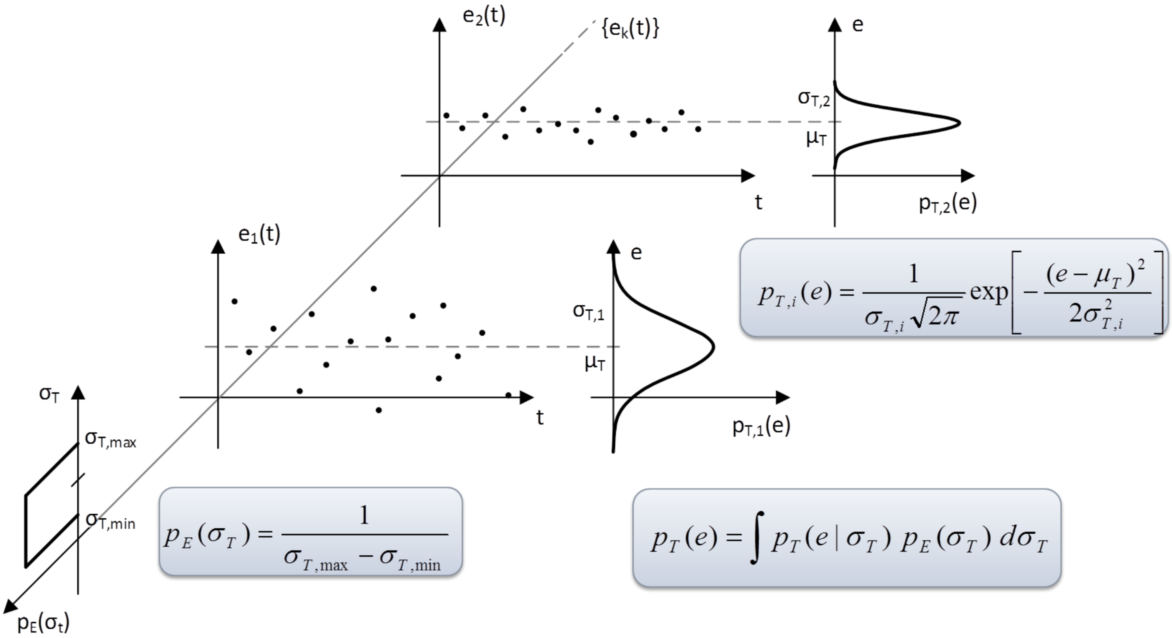


Figure 7: PES modelled as time- and ensemble-random variable (t, k)

A detailed description of the statistical domain concept is given in the user manual of the PEET v1.0 software.

3 POINTING-SAT

3.1 Overview

The PointingSat mission is a fictive but realistic mission scenario, which covers most functionalities of PEET v1.0. It was setup in that way to provide a suitable case study for cross-validation of the PEET v1.0 software to qualify it for release. Also the PES and system transfer modelling is done in high level of detail for that purpose. A complete PointingSat analysis document is available with the PEET v1.0 software.

3.2 Mission Scenario

PointingSat is a geostationary mission supporting the disaster assessment and monitoring for the European continent. The primary payload is a telescope for multi-spectral imaging (VIS, NIR, TIR, and MW) which allows detection and tracking of different ecological, economical and humanitarian incident follow-ups such as fires, algal bloom spread, oil slick or infrastructural damages after earthquakes, floods or windstorms. The main payload of PointingSat is a high-resolution telescope which is mounted on a stable optical bench. The IR focal planes are housed in cryostats and cooled. The mission scenario and the S/C are schematically illustrated in Figure 8.
As (dependent on the incident to be observed) the areas to be monitored are much larger than the payload FOV, highly accurate pointing and pointing stability of the satellite is required to allow single raster scanning of the relevant area on the one hand and repeated scanning of the same area in different spectral ranges. Above mentioned image acquisition strategy and multi-channel usage leads to requirements on different kinds of pointing errors (error indices) whose general definitions are illustrated in Figure 9.

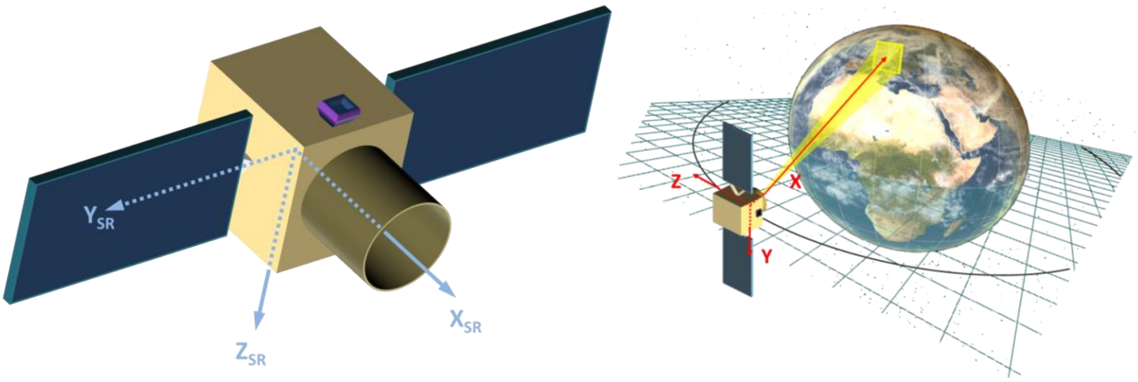


Figure 8: PointingSat Mission Scenario

The PointingSat AOCS uses a star-tracker (2 camera heads in cold redundancy) and fibre-optical gyros (3+3 cold-redundant) for attitude and rate determination. A set of 10 cold-gas thrusters (thrust range from 1 μN to 0.5 mN) is used for the precision pointing attitude maneuvers.

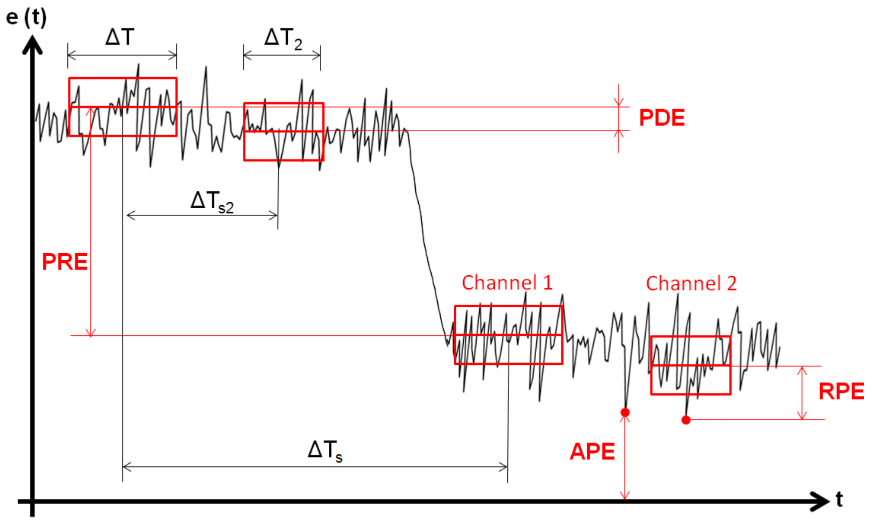


Figure 9: PointingSat Error Indices

3.3 Requirement Specification

In this paper the pointing relative performance error (RPE) requirement is taken as an example to show the analysis process with the PEET software. The requirement specification is defined in Table 1. Its specification in PEET v1.0 is given in Figure 10.

The purpose of the requirement is the need of a stable orientation throughout the integration time of the respective spectral channel (the window time Δt is the maximum integration time out of the individual channels). The image quality is determined by the aberration of the point spread function during the integration time of a single observation. Pointing variations during exposure lead to a broadening of the point spread function and thus to aberration.

Table 1: RPE requirement specification

Pointing Error Rqmt		**RPE**			
Evaluation Period		Nominal Observation			
Error Index		RPE			
Window-Time Δt [s]		0.5			
Stability-Time Δts [s]		-			
Unit		arcsec			
Required Error Value		x	y	z	LoS
		-	-	-	3
Ensemble Domains		Pc			
'Assembly+Launch' Ensemble (AED)		95.5%			
'Equipment Noise' Ensemble (ENED)		68.2%			
'External Environment' Ensemble (EEED)		68.2%			
Domain Treatment		Temporal Domain			
		Statistical		Worst-case	
Ensemble Domain	Statistical	-		-	
	Worst-case	AED,ENED,EEED		-	
Reference frame		LoS (x-axis) of the PointingSat-SAT-SR frame			
Applicable PES		All			

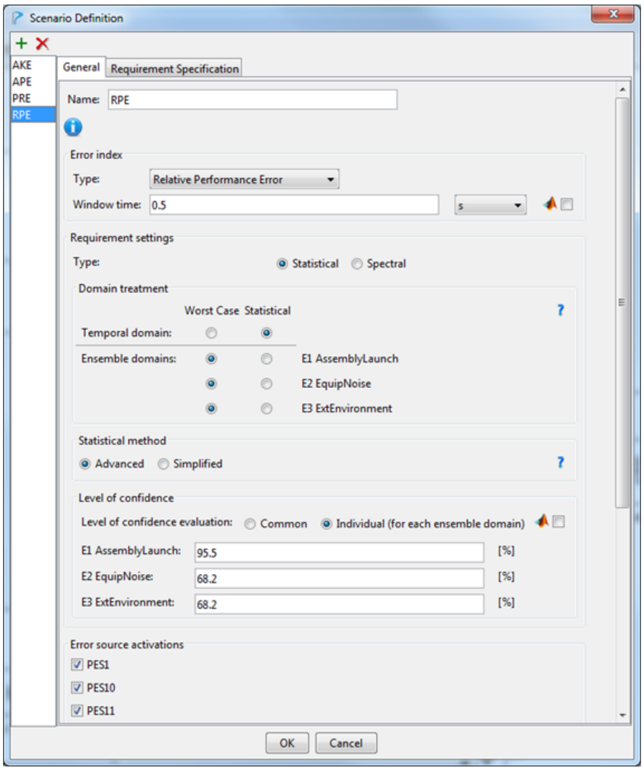


Figure 10: RPE rqmt specification in PEET

3.4 Modelling Pointing Error Sources & System Transfer

The PointingSat can be schematically broken-down in PES, system transfers and summations for modelling purposes. This is shown in Figure 11.

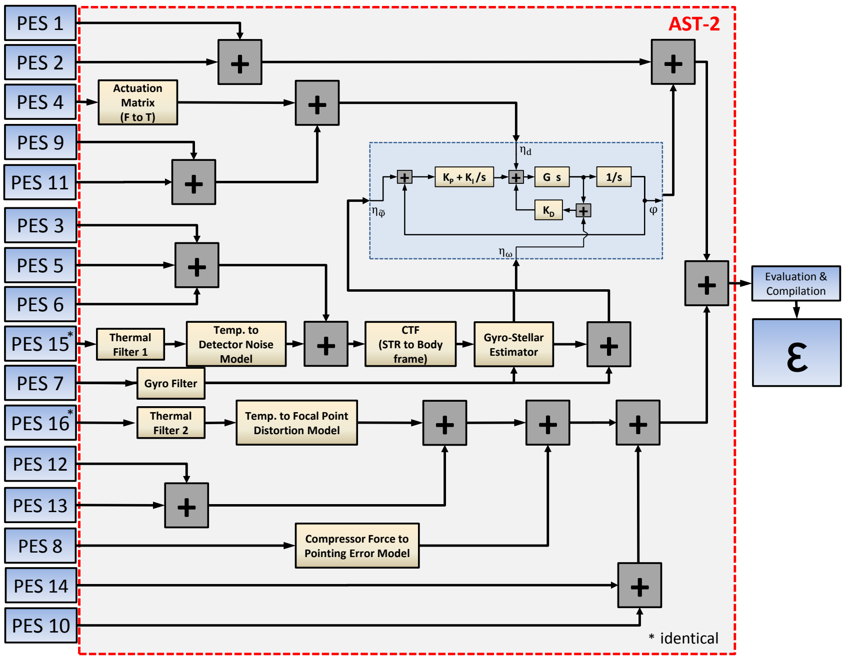


Figure 11: PointingSat schematic break-down in PES, system transfers and summations

The PEET v1.0 software supports the modelling by providing a block database and a system editor with all necessary elementary blocks to perform that task. The block database and the system editor are shown in Figure 12.

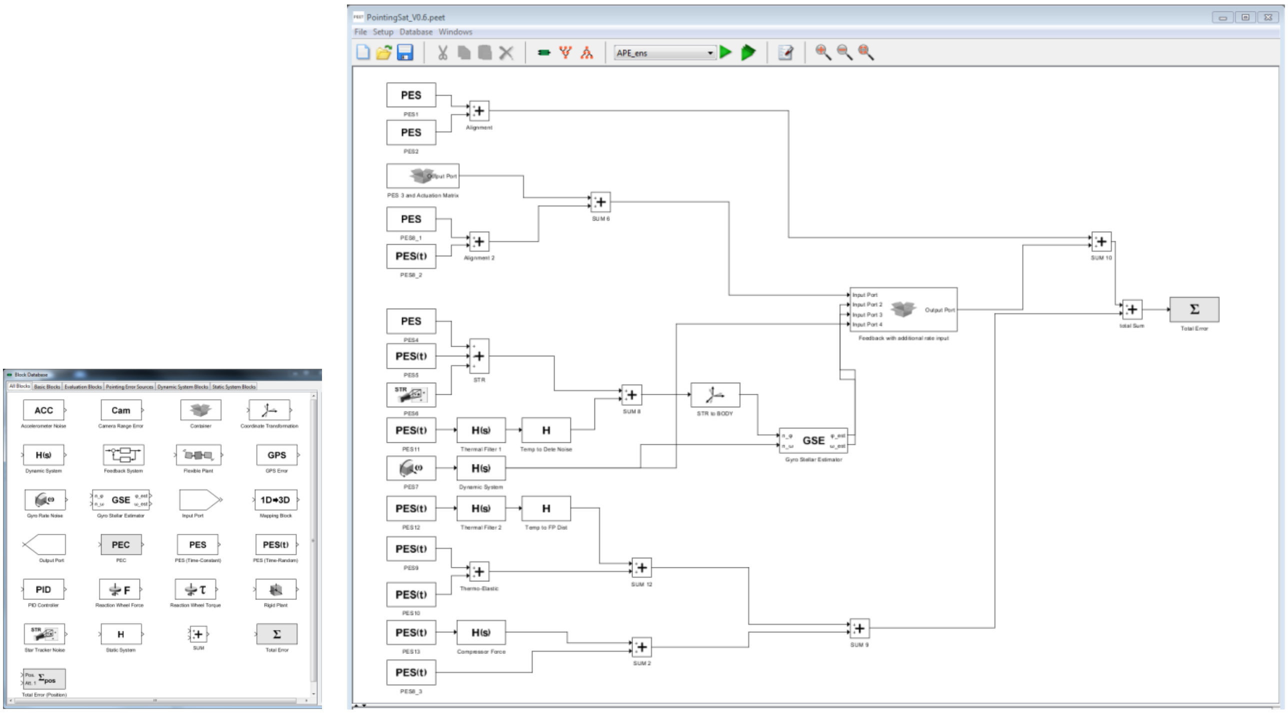


Figure 12: PEET v1.0 block database (left) and system editor (right)

The block database includes all necessary elementary blocks, but also specific blocks that are already a combination of elementary blocks. An example of such a block is the Gyro model block. It is based on the IEEE standard [11] as shown in Figure 13. PointingSat includes such a block in its pointing model.

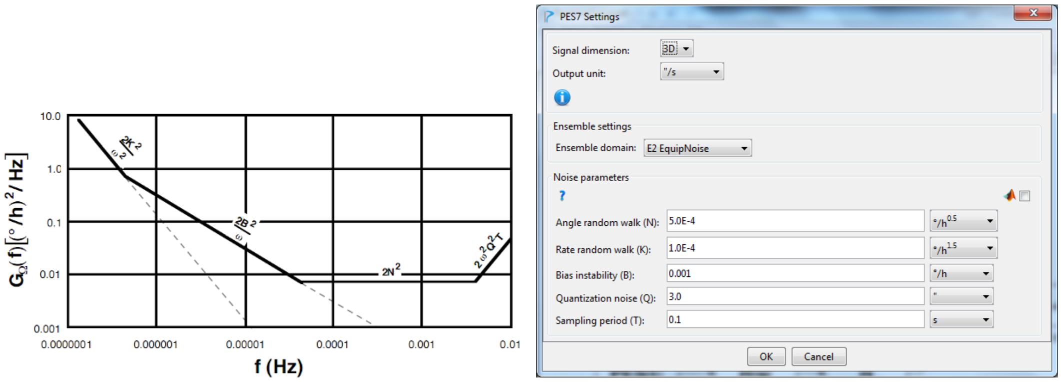


Figure 13: Gyro IEEE model (left) and the corresponding interface of the model block in PEET v1.0 (right)

Another specific block is the closed-loop transfer editor. When opening that block an editor opens to support the modelling of closed-loop systems. In PointingSat a closed-loop attitude control system is modelled with this editor.
A complete description of the block database can be found in the user manual of PEET v1.0.

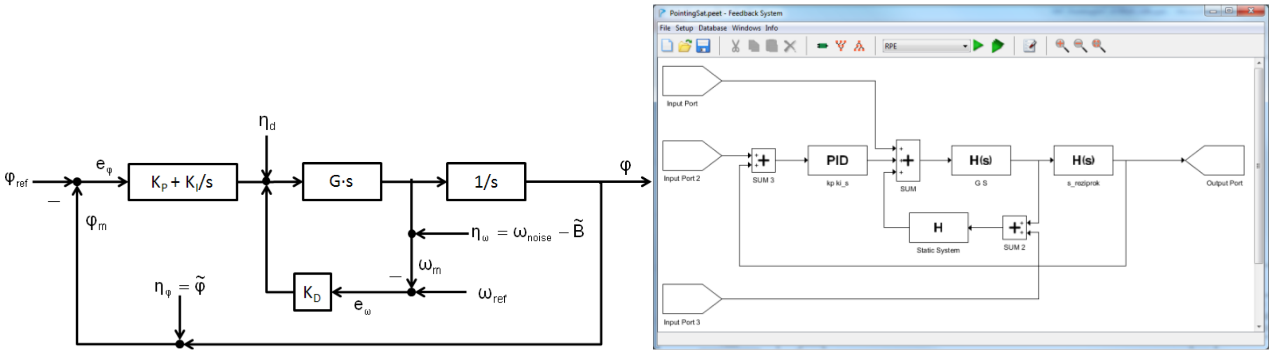


Figure 14: Schematic closed-loop control transfer (left) and the corresponding closed-loop editor in PEET v1.0 (right)

3.5 Pointing Budget

Once the pointing requirements are specified and the pointing system is modelled, the PEET software computes the pointing budget. The budget can then be analysed in the so-called tree-view, which is shown in Figure 15. In the tree-view the budget can be analysed at any nodal point with the error budget value and characteristics (e.g. PSD, PDF, correlation) at that point.

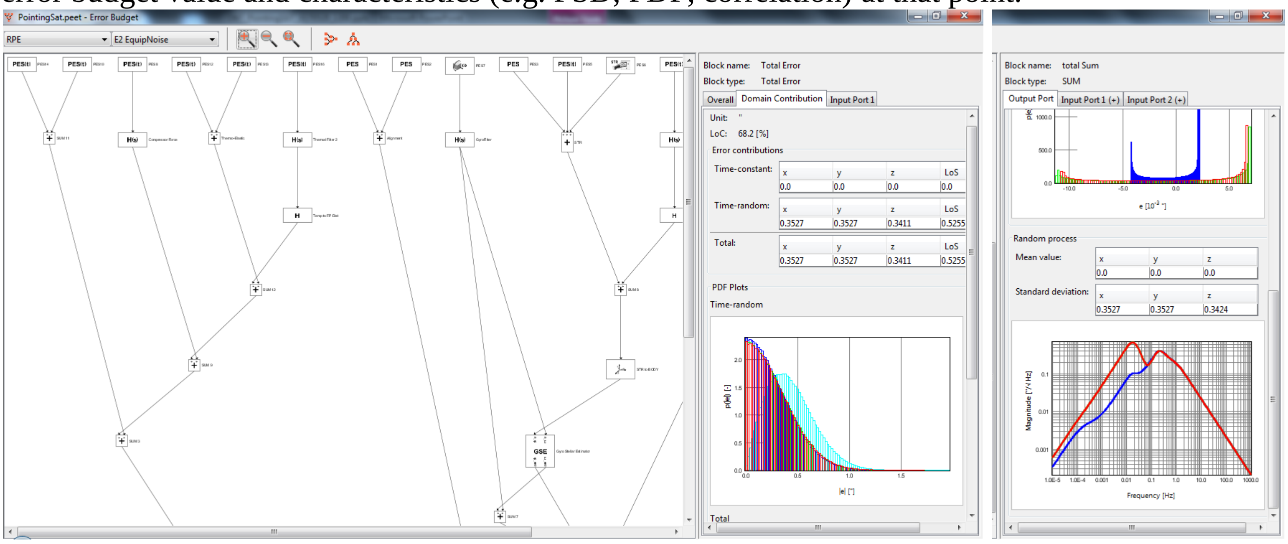
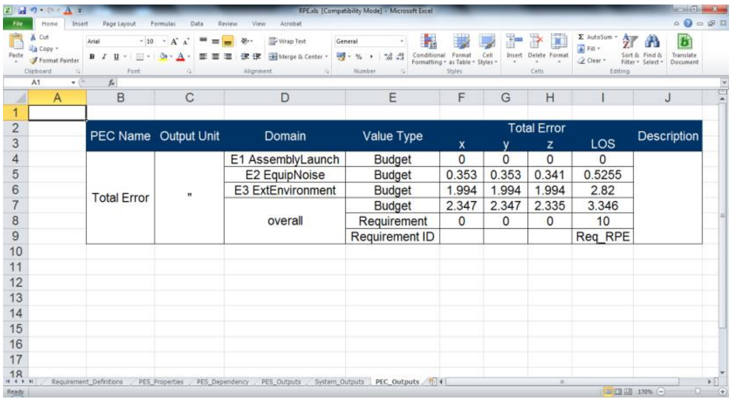


Figure 15: Tree-view of the PointingSat pointing error budget with PDF (left) and PSD (right) characteristics.

All information in PEET, i.e. final budget and error values at a nodal point can be automatically exported into an Excel file that can be used for reporting. One sheet of that Excel file is shown in Figure 16.



PEC Name	Output Unit	Domain	Value Type	Total Error x	y	z	LOS	Description
Total Error	"	E1 AssemblyLaunch	Budget	0	0	0	0	
		E2 EquipNoise	Budget	0.353	0.353	0.341	0.5255	
		E3 ExtEnvironment	Budget	1.994	1.994	1.994	2.82	
		overall	Budget	2.347	2.347	2.335	3.346	
			Requirement	0	0	0	10	
			Requirement ID				Req_RPE	

Figure 16: PEET v1.0 Excel Report for the RPE PointingSat budget

3.6 Cross-Validation with Matlab Monte Carlo Simulations

The PEET v1.0 software was cross-validated with two case studies before release. The cross-validation was performed by Airbus based on the experience of pointing error analysis for several flying space missions. In the cross-validation the case study was modelled in PEET v1.0 and in an equivalent state-of-the-art Monte Carlo simulation in Matlab Simulink. The findings during the cross-validation were fed back into the PEET v1.0 software development to consolidate it and provide a solid release version.

As case study the fictive PointingSat mission was chosen to cover most of the functionality, especially the high accuracy models and computations. The other case study was the relative pointing of the Laser Communication Terminals (LCT) on EDRS-C and Sentinel-2. This mission was chosen to cover the case of relative pointing and to have a real mission. The cross-validation process is shown in Figure 17.

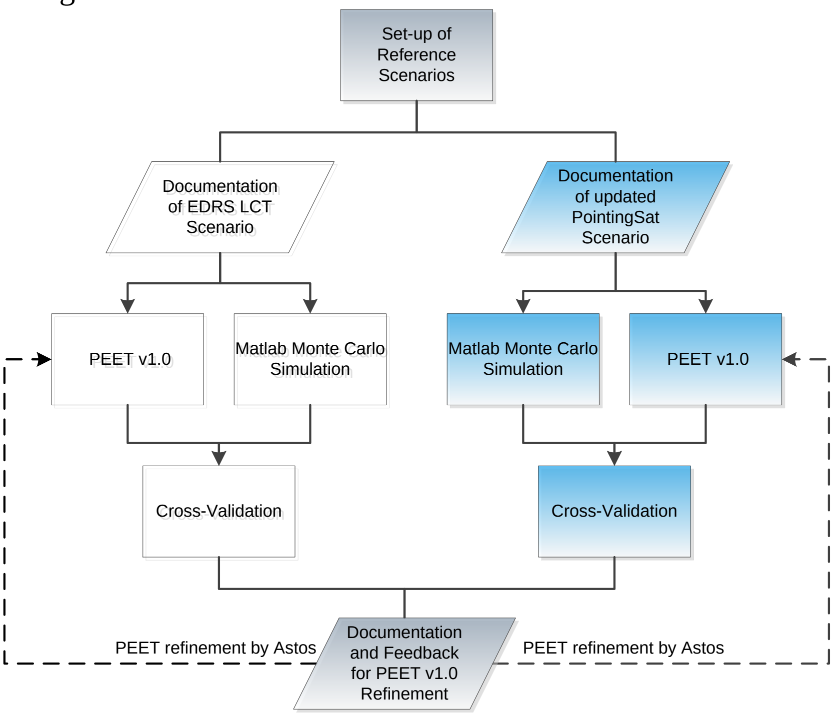


Figure 17: Cross-validation process for PEET v1.0

The cross-validation showed that the PEET v1.0 software produces the same results as the

equivalent Monte Carlo simulation for the advanced statistical method. PEET v1.0 also produces the same results as the latest PEET prototype v0.6, but for the so called simplified statistical method as the prototype is limited to this approach. In the benchmarking the achieved computational accuracy of PEET v1.0 versus an equivalent Monte Carlo Simulation is comparable. The deviation of results for both case studies is < 1% on atomic level and < 8% on system level. The deviation of 8% is a result of the limited computational capabilities. The MCS that are equivalent to PEET v1.0 computations were performed in Matlab Simulink. Thereby Simulink generated a too large amount of data for the used computers. For that reason the number of runs was reduced from 1 million to 0.5 million. That reduces the accuracy by a factor of 10. Hence it is expected that the system level MCS would achieve the same results as PEET v1.0 if one could run 1 million MCS. This already clearly states the advantage of PEET, which can handle 1 milion samples for the performed case studies in computational times that are faster than equivalent MCS in Simulink.
The computational speed of PEET v1.0, t_{PEET}, versus an equivalent Monte Carlo Simulation (MCS), t_{MCS}, is shown in Table 1 for the different pointing error requirements of the case studies. Based on the table one can state that:

$$t_{PEET} < 0.004\ to\ 0.2\ \cdot t_{MCS}$$

In addition the expected gain in accuracy has been shown by going from the SSM in PEET v0.6 to the ASM in PEET v1.0.

Table 2: Comparison of computational times between PEET v1.0 and MCS

Case Study	Requirement	Runtime		PEET deviation of x.x % compared to MCS
		PEET [min]	Matlab MCS [min]	
PointingSat	APE	11	1380	**0.8**
	RPE	6	30	**20.0**
	PRE	10	1380	**0.7**
	AKE	1	12	**8.3**
EDRS-S2-LCT	TUC (AKE)	4	960	**0.4**
	AKES (KDE)	1	30	**3.3**

4 METOP-SG EXPERIENCE

The ESA project MetOp-SG is the first project to use ESA PEET v1.0 in phase B/C/D. The main motivations to take the effort of introducing a new tool and with it a new analysis process are:

- The simplification of the pointing error engineering due to a standardized process with a clear computational approach and interfaces.
- High accuracy computation of the Performance Drift Errors (PDE), which budgets are to be modelled and evaluated in the frequency domain.

In MetOp-SG there are in average three different contractors involved in the assessment of the pointing budgets for one instrument. The mission has 10 instruments on two platforms and each instrument has one to five different pointing requirements. That leads to a total of 144 budgets to be analysed among ~30 different sub-contractors and the mission prime. A standardized pointing error engineering process is thus the key success factor for cost-efficient and high-quality engineering.

The process for MetOp-SG is illustrated in Figure 18. The exchange of data is purely based on PEET models and Excel reports automatically generated from PEET. In addition MS Excel input sheets are used that provide background information on the PES characteristics. The final results are then included in the instrument and system pointing dossier docuements.

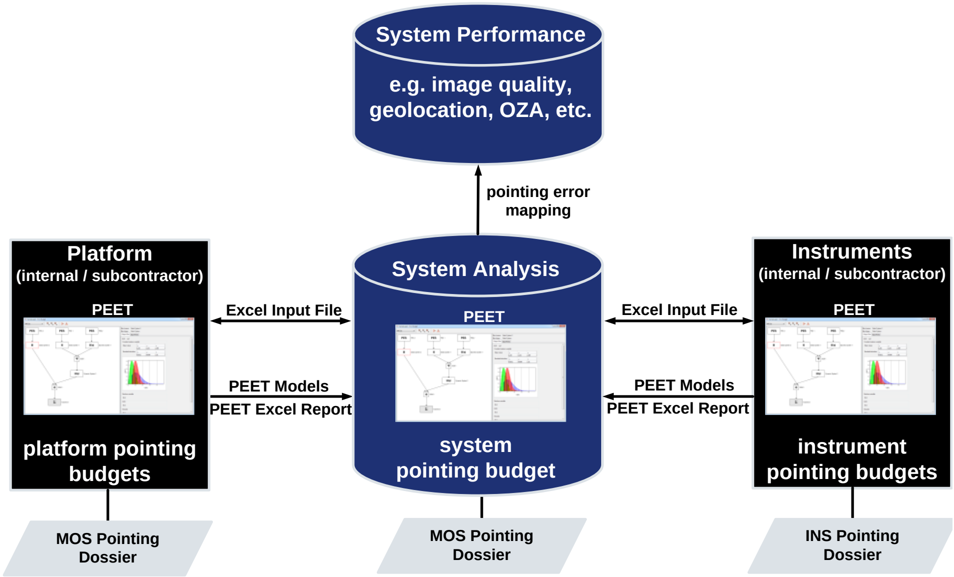


Figure 18: MetOp-SG pointing error engineering process

As can be seen in Figure 19, the number of models could be decreased by using the ESA PEET software. The number of models could be decreased from 144 models in MS Excel to 65 with PEET v0.6 to 37 with PEET v0.6 in combination with Matlab (for parameter initialization) and finally to 18 in PEET v1.0 in combination with Matlab. That reduces the model maintenance effort considerably and thus lowers the risk for having inconsistencies among the different involved parties.

Number of Models				
	Excel	PEET v0.6	PEET v0.6 improved using Matlab files	PEET v1.0 using Matlab files
Instrument	46	17	17	**8**
Platform	49	24	3	**2**
System	49	24	17	**8**
Total	144	65	37	**18**

Figure 19: Number of pointing models necessary to evaluate 144 budgets in MetOp-SG

The pointing analysis with the PEET software had to undergo several verification and test runs before it was introduced in the MetOp-SG project. The test results of PEET v0.6 w.r.t. classical approaches supported by MS Excel were published in [9]. An excerpt of the test results of PEET v1.0 w.r.t. PEET v0.6 is given in Figure 20. All platform budgets deviate by < 1%, which is as expected due to the different computational methods in PEET v0.6 and v1.0. The results are based on the application of the SSM that is baseline for MetOp-SG.

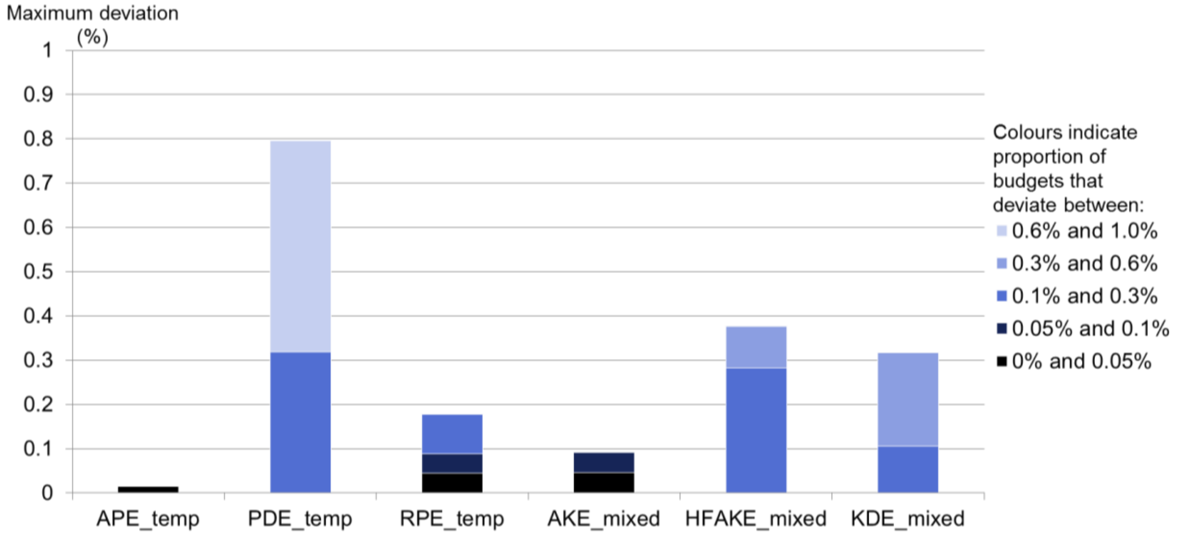


Figure 20: Deviation of results obtained with PEET v1.0 and PEET v0.6 for the MetOp-SG platform pointing budgets

In MetOp-SG the pointing error requirements (e.g. PDE ≈ 350 arcsec) are relaxed compared to high precision pointing missions (e.g. PDE ≈ 10e-3 arcsec). However, the PES are order of magnitudes higher in MetOp-SG because the satellites have to host several instruments with scanners and other mechanisms. That has the consequence that a pointing error engineering approach with high precision is necessary to achieve meaningful results. This is especially necessary for the pointing PDE requirements, which require accurate frequency domain analysis to determine the contribution of the PES to the final pointing error, cf. [1]. Some of the main PES drivers are shown in Figure 21 with the corresponding frequency domain metric. The driving PES include the Solar Array Drive Mechanism (SADM) vibrations, the Antenna Pointing Mechanism (APM) torques and µVibrations, the instrument speed variations of the scan mechanisms and the system µVibrations.

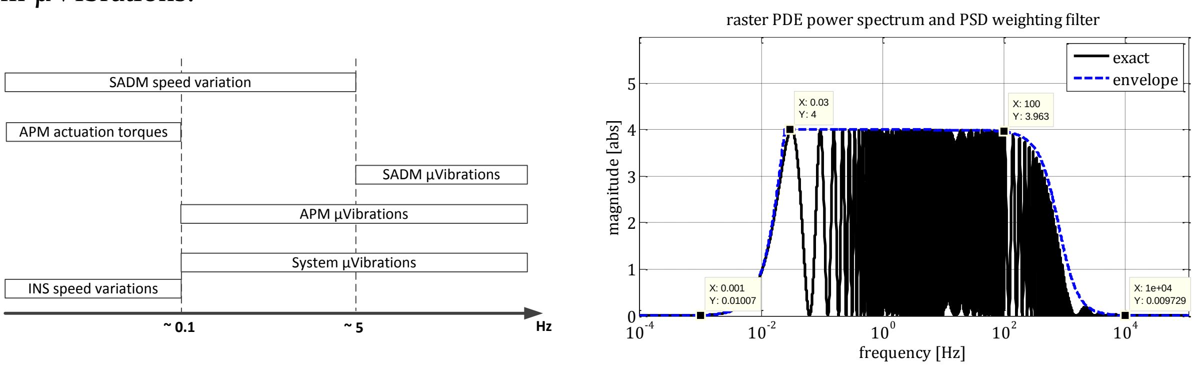


Figure 21: Interdisciplinary MetOp-SG PES in the frequency domain (left) and the corresponding frequency domain metric for evaluation (right), cf. [1].

5 CONCLUSIONS

In sum, the conclusion drawn is that the PEET v1.0 software is considered to be an important tool for any future space mission for correctly and efficiently analysing pointing performance and knowledge such that responsive and accurate feedback for system design can be given. This conclusion is based on the cross-validation of the PEET v1.0 software by Airbus and the experience gained in the MetOp-SG project. The migration from PEET v0.6 to v1.0 is suggested to profit from the higher accuracy in the "advanced method" and the user friendliness of v1.0 that will save time and thus cost. This is also suggested for running projects already using PEET v0.6 because V1.0 produces the same results in the "simplified approach", which is currently the standard approach for missions like MetOp-SG.

6 ACKNOWLEDGMENTS

The results presented here have been partially achieved under funding of the ESA GSTP contract No. 4000111774/14/NL/MH. The MetOp-SG specific results have been obtained under the respective project funding. Herewith we thank the colleagues in the MetOp-SG project and ESA for supporting the publication of these results.

7 REFERENCES

[1] ESA Engineering Standardisation Board, "Pointing Error Engineering Handbook ESSB-HB-E-003", *ESA-ESTEC Requirements & Standards Division*, 2011.

[2] Ott T., Benoit A., P. Van den Braembussche, W. Fichter, "ESA Pointing Error Engineering Handbook", *8th International ESA Conference on Guidance, Navigation & Control Systems*, Karlovy Vary CZ, June 2011.

[3] ECSS - European Cooperation for Space Standardization, "Control Performance Standard", ECSS-E-ST-60-10C, 2008.

[4] ECSS - European Cooperation for Space Standardization, www.ecss.nl (30 June 2014)

[5] ECSS - European Cooperation for Space Standardization, "Control Performance Guidelines", ECSS-E-HB-60-10A, 2010

[6] ECSS - European Cooperation for Space Standardization, "Stars sensors terminology and performance specification Standard", ECSS-E-ST-60-20C, 2008

[7] ECSS, "Gyros terminology and performance Standard", ECSS-E-ST-60-21C, 2008

[8] Lucke R.L., Sirlin S.W., San Martin A.M., "New Definition of Pointing Stability: AC and DC Effects", *The Journal of the Astronautical Sciences*, Vol. 40, No. 4, p. 557-576, 1992.

[9] Ott T., Wiedermann G., Crombez V., Damilano P., "Pointing Budgeting Using the ESA Pointing Error Engineering Handbook and Tool: Benefits and Limitations", *9th International ESA Conference on GNC Systems*, Oporto, Portugal, 2014

[10] Hirth M., Su H., Ott T., Casasco M., Ortega G., "PEET V1.0: The State-of-the-Art Pointing and Performance Error Engineering Tool for Space Missions", *10th International ESA Conference on Guidance, Navigation & Control Systems*, Salzburg, Austria, June 2017.

[11] IEEE Standard Specification Format Guide and Test Procedure for Single-Axis Interferometric Fiber Optic Gyros; *IEEE Std 952-1997*; IEEE: New York, NY, USA, 1998