IDENTIFICATION OF ASCENT PHASE LAUNCHER DYNAMICS USING MACHINE LEARNING

Diego Navarro-Tapia[1]**, Andrés Marcos**[1,2]**, Joris Belhadj**[3]

[1] *Technology for Aerospace Control (TASC) Ltd, TA12 6BN Martock, United Kingdom; (e-mail: diego.navarro-tapia/andres.marcos@tasc-group.com)*
[2] *Universidad Carlos III de Madrid (UC3M), Madrid, Spain; (e-mail: anmarcos@ing.uc3m.es)*
[3] *ESA-ESTEC, 2201AZ Noordwijk, The Netherlands; (e-mail: joris.belhadj@esa.int)*

ABSTRACT

This article presents the application of an on-board parameter identification approach using machine learning techniques to a launch vehicle system during its ascent-flight phase. The identification framework is based upon sparse regression, compressed sensing and robust control modelling, and allows for the identification of linear or non-linear equations from measurement data alone. The results show that the proposed approach is able to successfully identify the time-varying rotational dynamics of a launch vehicle for nominal and dispersed scenarios.

1 INTRODUCTION

The consolidation of the artificial intelligence (AI) field has resulted in a paradigm shift towards data-driven machine learning (ML) tools for modelling, design, analysis, verification and validation. There is much interest in the Space community in using these AI/ML methods with the aim to improve the performance, robustness and/or capabilities of the current (traditional and advanced) modeling and design approaches, but there are not yet many feasibility studies, and much less applications to systems of sufficient fidelity, to guarantee the successful transfer of the AI/ML methods to industrial operability.

Addressing the aforementioned lack of studies, ESA released in 2020 a call for proposals to study the use of AI techniques for GNC design, implementation, and verification. This article presents results from one of the three selected projects referred to in this article as *AI4GNC* and participated by Deimos Engenharia (DME, as coordinators), Deimos Space (DMS), INESC-ID, Lund University, and TASC. Specifically, the article shows the results used to demonstrate the feasibility of a data-driven ML approach used to identify the most relevant parameters of a launch vehicle during its atmospheric ascent-flight phase.

The ML technique used is based on sparse regression techniques [1] in conjunction with compressed sensing, and it allows for the identification of linear or non-linear systems from measurement data alone. The algorithm exploits sparsity-promoting techniques and machine learning with a library of possible candidate functions to identify the governing equations of systems characterized by relatively few non-zero terms.

In a first phase of the *AI4GNC* project, the one presented in this article, the algorithm is applied to a simplified model of a launcher during its atmospheric ascent flight (a well-known 2^{nd} order non-linear transfer function) in order to demonstrate its feasibility, performance, robustness, and shortcomings. This phase is critical to assess whether the algorithm is capable of being used subsequently for the non-linear benchmark as well as to gather experience and knowledge on its tuning and on-board implementation capabilities.

The aim of this feasibility application is to identify the two most relevant rigid-body rotational launcher parameters, commonly known as a_6 and k_1. These parameters are of particular interest to flight mechanics and control, as they are directly linked to the controllability and stability of the vehicle. The analysis of the proposed compressed sparse identification approach is presented in incremental steps of complexity in order to build confidence and gain insight on the process: starting with the case of constant dynamics, and gradually building up the complexity of the identification approach by considering a windowing compressed estimation, and finally a real ascent-flight, time-varying profile for the launcher dynamics.

The layout of the article is as follows. Section 2 presents the ML-based identification approach. Section 3 describes the simplified launch vehicle simulator used to perform the assessment of the proposed approach. The verification results are presented in Section 4, and finally, conclusions are given in Section 5.

2 COMPRESSED SPARSE REGRESSION APPROACH

As aforementioned, the proposed approach combines data-driven sparse regression with compressed sensing to provide an on-board parameter identification approach. This is done in two main steps: one for compressing the collected data, and the second where sparse regression is used to perform the estimation of the chosen parameters. A third step, connected to LFT robust modeling theory, is also used to provide confidence levels on the estimation as well as augment the capability of the approach towards on-board uncertainty range estimation.

Consider a dynamical system as given in Eq. (1), where $x(t) \in \Re$ represents the measured variables of the system at time t (e.g., states and inputs), and the function f is sparse (i.e. it consists of only a few non-zero terms -which is applicable to many physical systems):

$$\dot{x}(t) = f(x(t)) \tag{1}$$

The main goal of the proposed approach is to identify the dynamics of the function f by only using time-series data, and be able to achieve this on-board and in real-time.

2.1 Step I - Compressed data collection

The first step consists of collecting time-series data using a series of m discrete snapshots in time t. Note that it can be obtained directly by measurement from the real system or, for verification and validation (V&V) purposes, can be emulated to come from real-time simulation of a high-fidelity, nonlinear model of the system.

In preparation for the subsequent identification step, the time-series data is arranged in matrix form $\mathcal{X} \in \Re^{m\times n}$ as shown in Eq. (2). Similarly, a matrix of derivatives is formed $\dot{\mathcal{X}} \in \Re^{m\times n}$. This matrix can be either numerically computed from $\mathcal{X}$ or also measured if appropriate sensors are available.

$$\mathcal{X} = \begin{bmatrix} \mathbf{x^T(t_1)} \\ \mathbf{x^T(t_2)} \\ \vdots \\ \mathbf{x^T(t_m)} \end{bmatrix} = \begin{bmatrix} x_1(t_1) & x_2(t_1) & \cdots & x_n(t_1) \\ x_1(t_2) & x_2(t_2) & \cdots & x_n(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(t_m) & x_2(t_m) & \cdots & x_n(t_m) \end{bmatrix} \tag{2}$$

The compressed, or sub-sampled, data $\mathcal{Y} \in \Re^{p\times n}$ can be obtained from the full dataset $\mathcal{X}$ as follows:

$$\mathcal{Y} = \mathcal{C}\mathcal{X} \tag{3}$$

where $\mathcal{C}$ is an operator that performs the sub-sampling step, with $p < m$. The choice of $\mathcal{C}$ has a direct effect on the performance of the identification step. In this work, random sub-sampling was chosen as it has been shown to be effective for compressive sampling of spectrally sparse signals [2].

2.2 Step II - Compressed sparse regression identification

In this second step, the function f is approximated as a linear model of possible candidate functions $\Theta(\mathcal{Y}) \in \Re^{p\times q}$ and a matrix $\Xi \in \Re^{q\times n}$, see Eq. (4), and a sparse regression optimization is used to identify the best candidate functions that fit the compressed-data from the previous step.

$$f(\mathcal{Y}) \approx \sum_{k=1}^{q} \theta_k(\mathbf{y}^{\mathbf{T}})\xi_k = \Theta(\mathcal{Y})\Xi \tag{4}$$

The library $\Theta(\mathcal{Y})$ contains q column vectors, each representing a possible term in the governing equations to be identified. These candidates can be constant, polynomial and/or trigonometric functions of the compressed-data matrix $\mathcal{Y}$ from Eq. (3). There is no systematic approach to define the best library of candidate functions, but it is recommended that they are selected based on the physical dependency of the system. With respect to the matrix Ξ, each of its columns represents a sparse vector $\xi_k \in \Re^n$ formed by those coefficients in the right-hand side of Eq. (1) which are active.

Then, the dynamical system may be represented as an overdetermined linear regression problem for learning the governing equations, see Eq. (5):

$$\dot{\mathcal{Y}} = \Theta(\mathcal{Y})\Xi \tag{5}$$

The unknown coefficients in matrix Ξ can be obtained by penalizing the number of non-zero terms in the dynamics using an optimization problem solved by convex ℓ_1-regularized sparse regression. This is shown in Eq. (6), where λ is a sparsity-promoting regularization weight chosen by simple hyper-parameter tuning:

$$\xi_k = \underset{\xi_k'}{\operatorname{argmin}} \|\dot{\mathcal{Y}}_k - \Theta(\mathcal{Y}_k)\xi_k'\|_2 + \lambda\|\xi_k'\|_1 \tag{6}$$

Possible sparse regression algorithms are: lasso [3], sparse relaxed regularized regression [4], and step-wise sparse regression [5]. In this work, the *SINDy* method is used, which achieves sparsity by using a sequential thresholding least squares (STLS) algorithm [1]. The latter is an iterative algorithm that performs least-square regression while simultaneously thresholding those parameters smaller than a defined cut-off value (given by the parameter λ). This process of regression and thresholding is repeated until convergence.

2.3 Step III - Connection with LFT theory

A third step is included in the proposed approach with the two-fold aim of: (1) providing a confidence range for the parameter estimation, and (2) extend the identification capability of the approach.

The first aim is for practical reasons since in non-purely academic examples, perfect identification of a parameter is hampered by many sources of error, uncertainty and/or issues such as delay, saturation, and signal quantification. Thus, for systems with wide dynamical variation, it is required to have a quantitative measure on the confidence of the estimation. The second aim follows the first in that once confidence bands have been defined, then the approach can be used to identify the associated level of uncertainty –which can be helpful to modify the robustness of the closed-loop system. In both aims, it is proposed to use an (offline) LFT model of the system.

LFT theory [6] is a well-established and suitable approach to model system uncertainties (e.g., parametric system variations, non-modelled dynamics, uncertain time delays, actuator and sensor nonlinearities). The LFT representation naturally fits inside the wider robust control framework which, among other things, can provide analytical guarantees over the stability and performance of a closed-loop system in the presence of uncertainties.

In the proposed approach, an LFT model of the system is developed offline using a grey-box approach [7], which combines first-principle analytical equations with simulation-based data-fitting. This LFT model serves to obtain the confidence range of the first aim (by perturbing its uncertain parameters randomly and/or to extreme combinations), but it also can be uploaded on-board to then generate a residual signal with respect to the onboard estimated parameter value, and thus provide a quantitative way to identify the level of uncertainty.

A general uncertainty model for a parameter a is given in Eq. (7), where a_0 represents its nominal value, σ_a the known level of uncertainty, and δ_a is an unknown but norm-bounded constrained uncertainty flag ($\|\delta_a\| \leq 1$).

$$a = a_0(1 + \sigma_a \delta_a) \tag{7}$$

If the proposed identification approach is applied to the system in (1), an estimate $\hat{a}$ of the parameter is obtained. If this value lies within the known range σ_a of a_0, then there is confidence that the onboard estimate is adequate (and if it is beyond the range, that possibly another factor is at play such as faults or strong wind effects not accounted in the LFT modelling). Further, if the range σ_a is sufficiently large, then the δ_a value can be identified and used to provide a rough estimate on the uncertainty region at that instance (e.g. within $[0-30]\%$, $[30-60]\%$, $[60-100]\%$ of $|\sigma_a|$).

3 SIMPLIFIED ASCENT-FLIGHT LAUNCH VEHICLE SIMULATOR

This section describes the simplified single-axis simulator developed by TASC to support the assessment of the identification approach described in Section 2. The simulator is implemented in Matlab/Simulink using generic Simulink blocks and specific blocks from the control system toolbox. Figure 1 shows a diagram of the simulator that contains guidance, navigation and control functions as well as TVC actuator and launch vehicle dynamics.

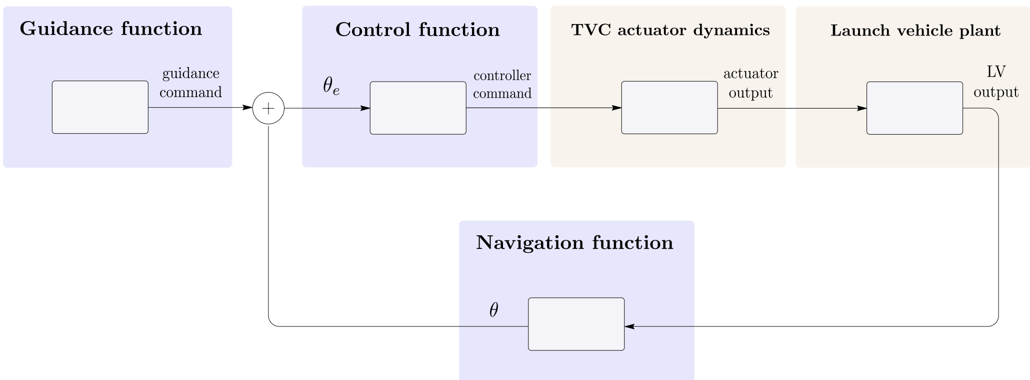


Figure 1: Simulink diagram of simplified launch vehicle simulator

The control function is a standard proportional-derivative controller designed using the guidelines from reference [8]. It is noted that a discrete pseudo-derivative filter is employed to compute the attitude rate error signal $\dot{\theta}_e$. The navigation function adds white noise to the attitude measurements, whereas the guidance function is implemented as an open-loop look-up table that provides attitude references for an ascent-flight trajectory with initial vertical-flight phase and subsequent pitch-over manoeuvre.

The TVC actuator is characterized as a second-order transfer function with additive white noise. The launch vehicle model is described with the simplified single-axis rotational dynamics given in Eq. 8. This model consists of two parameters: the aerodynamic instability coefficient a_6 and the control efficiency parameter k_1. These two parameters determine the main rotational rigid-body motion dynamics of the vehicle between the attitude angle θ and the nozzle deflection angle β_θ. These two parameters are generally used in a linear fashion to represent the rotational dynamics of the vehicle for design and modelling as they are linked to the controllability and stability of the vehicle as shown in [9]. In addition, it is standard in industrial launcher mission preparation to start with this rotational relation for attitude control design purposes [10]. Nonetheless, it is important to remark that they capture the nonlinear dynamics due to key system variables (such as aerodynamics, thrust profile and vehicle's moment of inertia) captured within the equation's parameters.

$$G_{LV}(s) = \frac{k_1}{s^2 - a_6} \tag{8}$$

4 RESULTS

The analysis of the compressed sparse identification approach was carried out through three incremental steps. First, a preliminary sparse identification is performed using constant launch vehicle dynamics (Section 4.1), that is, constant a_6 and k_1. This is followed by two windowed compressed runs, one with constant dynamics (Section 4.2), and another one with time-varying dynamics extracted from a real launch vehicle ascent-flight profile (Section 4.3).

4.1 Compressed sparse regression identification for constant dynamics

This section applies the identification approach described in Section 2 to a launch vehicle system with fixed rotational dynamics throughout its ascent flight with the following system parameters: $a_6 = 2$ and $k_1 = -7$. The reference trajectory consists of a standard attitude ascent-flight profile with initial vertical-flight phase and subsequent pitch-over manoeuvre.

The compressed sparse identification procedure is shown schematically in Fig. 2. The first step consists of collecting the full-data set $\mathcal{CX}$ as described in Section 2.1. To this end, the time-series data of the main system variables (nozzle deflection angle β_θ, attitude θ and its derivative $\dot{\theta}$) are collected using the simulator described in Section 3. It is noted that the measurements are compressed by means of random sub-sampling. In particular, a compression factor (i.e. decimation) of 3 is used, which means keeping only one third of all the samples in the data set. The compressed data is then arranged in a large data matrix $\mathcal{CX} = \mathcal{C}[\theta \; \dot{\theta} \; \beta_\theta]^T$ and in the corresponding derivative matrix $\mathcal{C}\dot{\mathcal{X}}$, where each row is a discrete snapshot of the data in time.

The second step (see Section 2.2) involves providing a library Θ of candidate functions in $\mathcal{CX}$, each representing a possible term in the governing equations to be identified. It is recommended to select the library $\Theta(\mathcal{CX})$ based on the knowledge and physical insight of the system, but in this case for the sake of demonstration it is assumed that the dynamics are unknown. Thus, a space of polynomials in $\mathcal{CX}$ is selected up to second order, that is, $\Theta(\mathcal{CX}) = \mathcal{C}[1 \; \theta \; \dot{\theta} \; \beta_\theta \; \theta^2 \; \theta\dot{\theta} \; \theta\beta_\theta \; \dot{\theta}^2 \; \dot{\theta}\beta_\theta \; \beta_\theta^2]$. Then, the compressed sparse identification problem can be formulated as in Eq. 5, see Fig. 2. Subsequently, the unknown coefficients in matrix Ξ are solved via sparse regression using the STLS algorithm [1].

The results of the identification approach are illustrated in Fig. 2, see matrix Ξ. The sparse identification algorithm correctly identifies the dynamics of the system (i.e. $a_6 = 2$ and $k_1 = -7$). It is interesting to observe that the matrix Ξ is sparse in the space of possible functions (i.e. there are only a few non-zero terms governing the physical equations). Also, it is noted that the same good results were obtained for different actuator and sensor noise seeds, showing that for this case, the identification process is insensitive to the stochastic nature of the system.

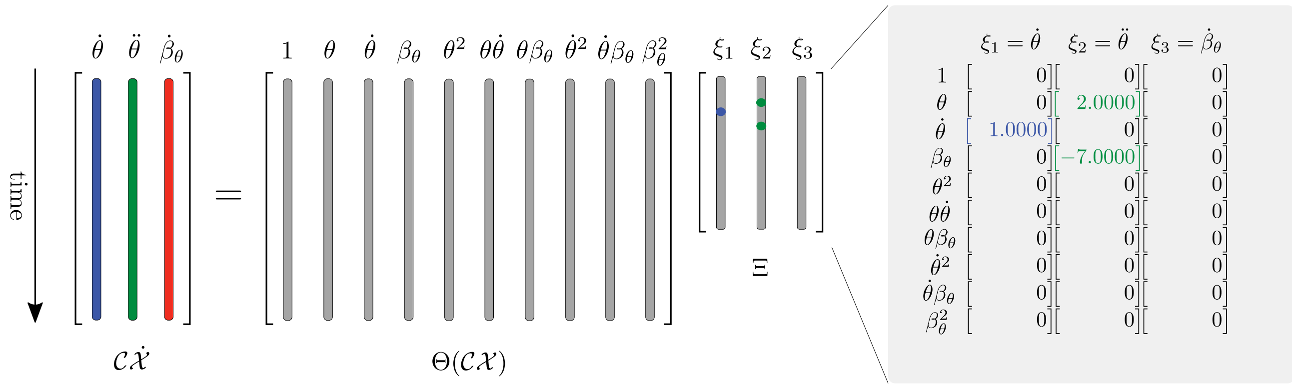


Figure 2: Schematic of compressed sparse regression for a launch vehicle with constant dynamics

4.2 Windowed compressed sparse regression identification for constant dynamics

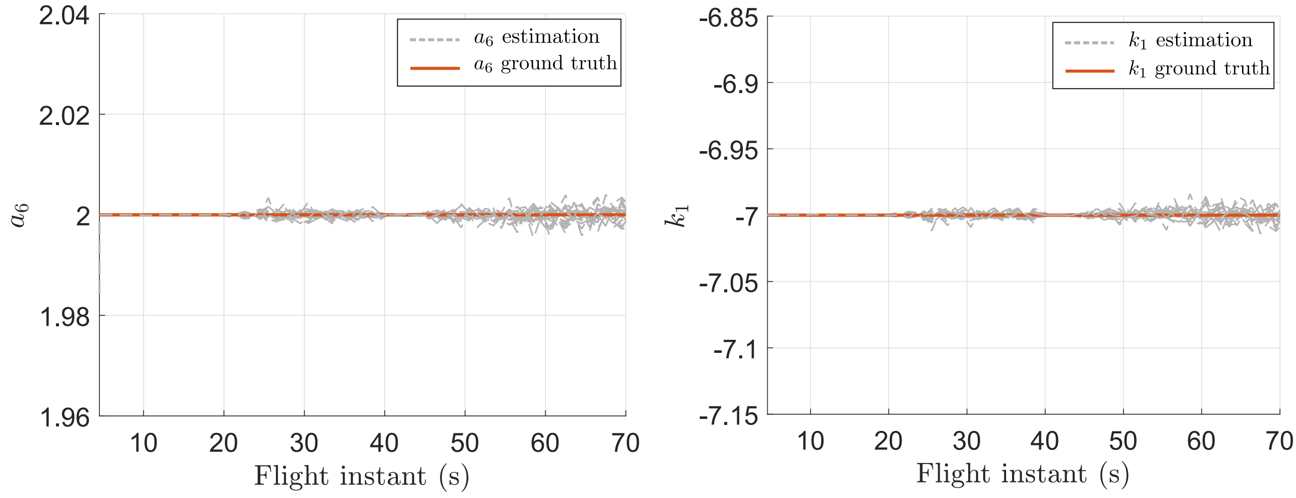
The above system identification case showed that the compressed sparse algorithm works properly when the governing equations can be formulated in terms of polynomial candidate functions and the system has time-invariant constant coefficients. This is in good agreement with the algorithm performance presented in [1, 11].

However, most dynamical systems operate for a wide variation of their parameters, e.g. launch vehicles mass-center of gravity-inertia (MCI) changes during ascent and descent phase due to the fast and large propellant consumption. Thus, in order to deal with large and/or rapid parameters' variation for non-constant (i.e. time-varying) systems, the proposed estimation must be performed using time-series windowing as in [12]. That is, the data collection step structures the data in windows of N_w samples so the identification step can be applied to each window and sequentially cover the full operational domain.

The selection of the window length arises in practice as a trade-off between the number of samples N_w and the estimation accuracy. On the one hand, the window length shall be selected to ensure that the variability of the system parameters is not very high, otherwise the sparse identification would fail to find a suitable function f to match the dynamics of the windowed data. On the other hand, the sparse algorithms may fail if the window length is too small due to the lack of enough samples to solve the compressed sparse regression. This is a critical issue when data is compressed, since a data window that is too small will mean that an insufficient number of samples are used for the subsequent sparse regression identification step.

In order to build confidence on the approach, the windowed compressed sparse identification algorithm is applied in this subsection to the previous constant dynamics case, but dividing the ascent-flight phase data into 70 windows of $1\,\mathrm{s}$ length.

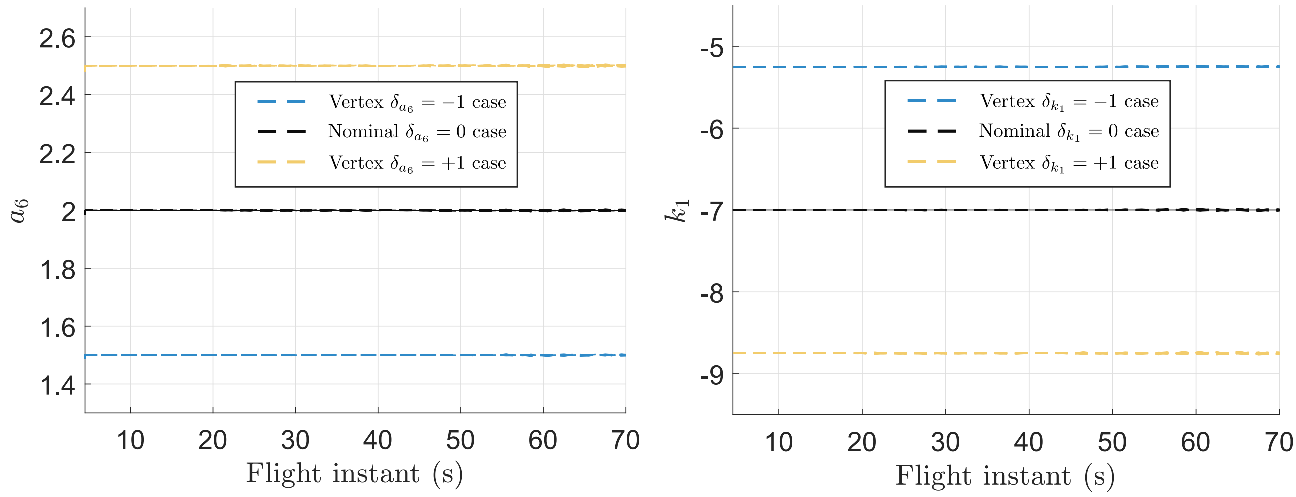
Figure 3 shows the results of the windowed compressed sparse regression approach for 20 random seed simulations (this is done to analyse the stochastic effect of the noise generators). It is noted that the randomization here affects the actuator and sensor noise generators as well as the random sub-sampling process. Nonetheless, the results show that the windowed identification algorithm is also able to correctly identify the constant rotational dynamics. Specifically, the flight parameters obtained present less than 1% of error with respect to the ground truth.



(a) a_6 estimation (b) k_1 estimation

Figure 3: Windowed compressed sparse regression results for constant dynamics: nominal case

The same approach was also applied to a constant dynamics case but with uncertain parameters (e.g. $\sigma_{a_6} = \sigma_{k_1} = 0.25$). Figure 4 shows the results of the identification algorithm using three different uncertain configurations: nominal with all the defined uncertain parameters ($\# = a_6, k_1$) set to their nominal values (i.e. $\delta_\# = 0$) and two vertex cases with all the parameters set to their maximum and minimum values (i.e. $\delta_\# = 1$ and $\delta_\# = -1$, respectively). As before, the windowed identification is repeated 20 times for each random seed simulation. It is clear from the results that the proposed identification approach allows identifying the uncertainty range of the parameters with less than 1% of error with respect to the ground truth.



(a) a_6 estimation (b) k_1 estimation

Figure 4: Windowed compressed sparse regression results for constant dynamics: dispersed cases

4.3 Windowed compressed sparse regression identification for nominal time-varying dynamics

This section presents the evaluation of the proposed windowed compressed sparse identification approach using a real time-varying rotational profile extracted from the atmospheric ascent-flight of the VEGA launcher VV05 mission [13]. The rapid time variation of the rotational parameters during the atmospheric flight is illustrated in Fig. 6.

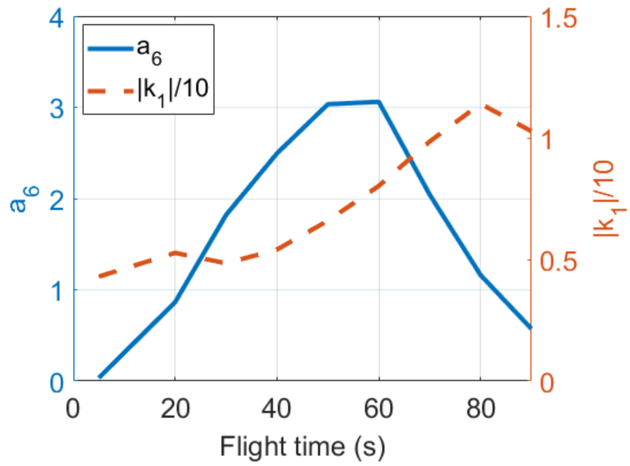
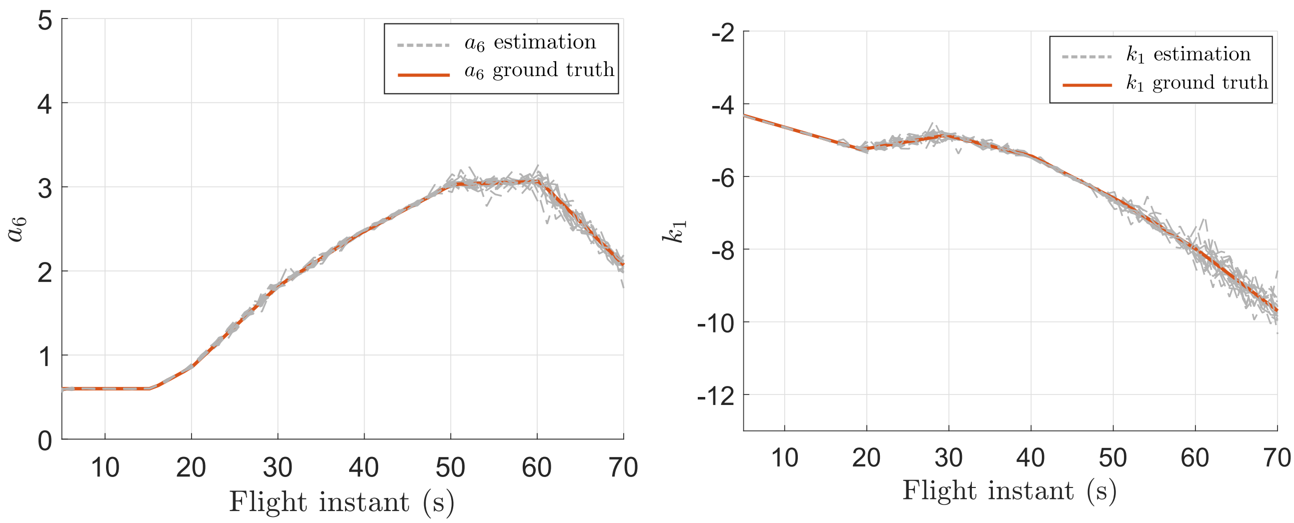


Figure 5: VEGA VV05 mission rigid-body rotational parameters [13]

The launch vehicle plant is implemented as a linear parameter-varying (LPV) model in order to encapsulate the time-varying nature of the system. Due to the dynamic change of the flight parameters, the estimation is performed using a window of $0.3\,\mathrm{s}$.

Similarly to the case presented in Section 4.2, the windowed compressed sparse regression results are applied to 20 random seed simulations, see Fig. 6. It can be seen that in all the cases the windowed estimation algorithm correctly identifies the time-varying dynamics of the system.

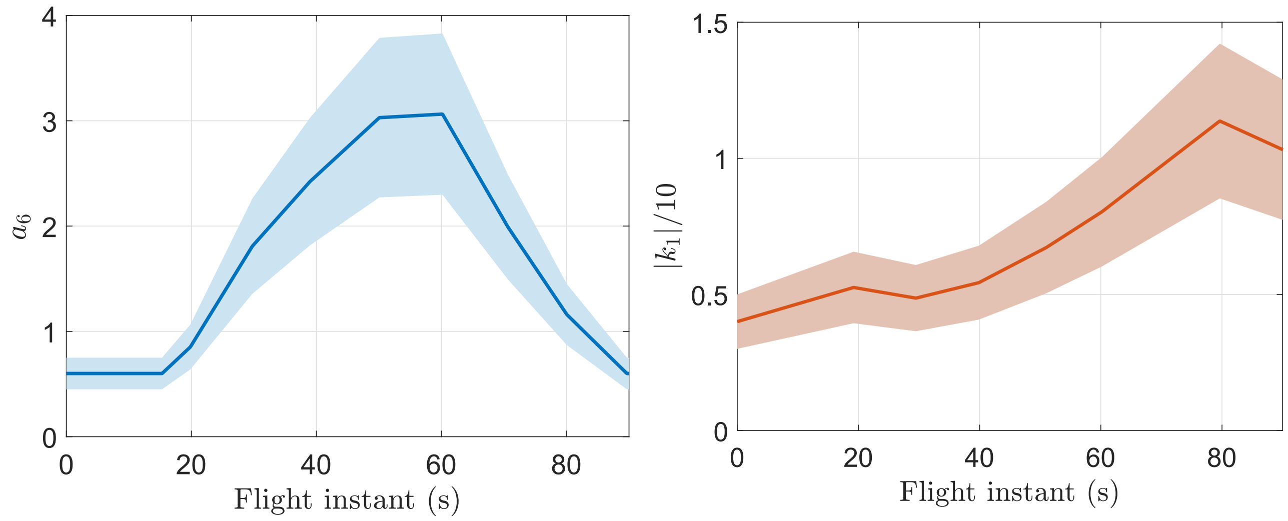


(a) a_6 estimation

(b) k_1 estimation

Figure 6: Windowed compressed sparse regression results for time-varying dynamics: nominal case

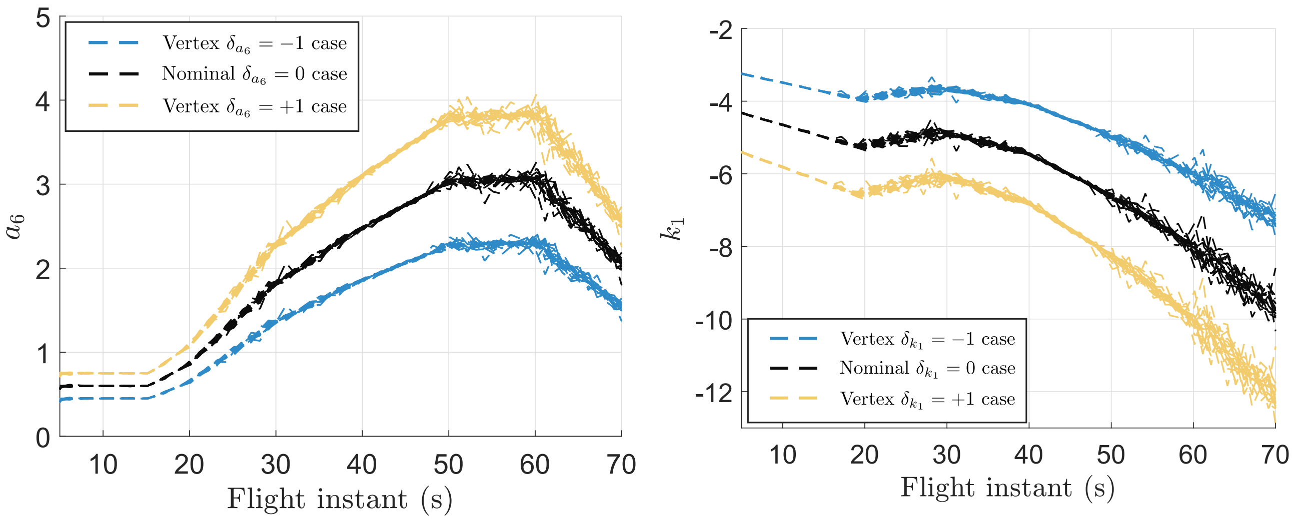
In a second test, the uncertainty behaviour shown in Fig. 7 is used to perturb the rotational dynamics. The results are shown in Fig. 8 for the three uncertain configurations used before, nominal and two vertex cases (i.e. $\delta_{\#} = 1$ and $\delta_{\#} = -1$), and the 20 random-seed campaign. As before, the identification is quite accurate for each of the uncertainty cases. It is worth noting that the results in Fig. 8 can be connected to the offline LFT model via the uncertainty norm-bounded parameters δ_{a_6} and δ_{k_1}, which can be used to develop an onboard uncertainty identification approach.



(a) a_6 uncertain profile

(b) k_1 uncertain profile

Figure 7: VEGA VV05 mission: uncertain rigid-body rotational parameters profile



(a) a_6 estimation

(b) k_1 estimation

Figure 8: Windowed compressed sparse regression results for time-varying dynamics: uncertain cases

5 CONCLUSIONS

This article demonstrates the application and feasibility of on-board system parameter identification approach developed based on machine learning techniques (compressed sparse regression) and robust control modelling (linear fractional transformation). The approach was applied to a simplified ascent-flight launch vehicle simulator for the identification of two rigid-body rotational parameters: a_6 and k_1. The identification process was conducted in an incremental manner to gain insight into the proposed process, starting with a constant dynamics case and gradually increasing the launch vehicle complexity as well as that of the identification approach.

The results show that the proposed windowed compressed sparse identification approach can correctly identify the time-varying linear dynamics of a launch vehicle in both nominal and dispersed scenarios. These results were connected to robust control via an off-line LFT modelling, which allows to perform uncertainty level identification. This information could be very valuable on-board, since it allows to determine whether a system parameter is within the modelled uncertainty range and if not, then possibly trigger control reconfiguration functionalities to ameliorate the larger uncertainty impact.

ACKNOWLEDGMENT

This work was part of the project "Artificial intelligence techniques for GNC design, implementation, and verification" (*AI4GNC*) funded by the ESA-TDE programme with contract No. 4000134108/21/NL/CRS. The view expressed in this paper can in no way be taken to reflect the official opinion of the European Space Agency.

The authors acknowledge the contribution of Manuel Martínez Pérez as TASC engineer during the *AI4GNC* project. Special thanks also to Dr. Paulo Rosa (coordinator of the *AI4GNC* consortium), Dr. Samir Bennani (ESA control expert), Dr. David Sánchez de la Llana, and Dr. Valentin Preda (ESA-ESTEC) for their valuable support feedback in the project.

Dr. Marcos gladly acknowledges funding as Beatriz Galindo Distinguished Senior Researcher by the Spanish Government, and by the Madrid Government (Comunidad de Madrid-Spain) under the Multiannual Agreement with UC3M in the line of "Research Funds for Beatriz Galindo Fellowships" (SPACEROBCON-CM-UC3M), and in the context of the V PRICIT (Regional Programme of Research and Technological Innovation).

REFERENCES

[1] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *Proceedings of the National Academy of Sciences*, vol. 113, no. 15, pp. 3932–3937, 2016. [Online]. Available: https://www.pnas.org/doi/abs/10.1073/pnas.1517384113

[2] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006. [Online]. Available: https://doi.org/10.1109/tit.2005.862083

[3] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996. [Online]. Available: http://www.jstor.org/stable/2346178

[4] K. Champion, P. Zheng, A. Y. Aravkin, S. L. Brunton, and J. N. Kutz, "A unified sparse optimization framework to learn parsimonious physics-informed models from data," *IEEE Access*, vol. 8, pp. 169 259–169 271, 2020.

[5] L. Boninsegna, F. Nüske, and C. Clementi, "Sparse learning of stochastic dynamical equations," *The Journal of Chemical Physics*, vol. 148, no. 24, p. 241723, 2018. [Online]. Available: https://doi.org/10.1063/1.5018409

[6] J. C. Doyle, A. Packard, and K. Zhou, "Review of LFTs, LMIs, and μ," in *Proceedings of the* 30^{th} *IEEE Conference on Decision and Control*, December 1991, pp. 1227–1232.

[7] A. Marcos, S. Bennani, C. Roux, and M. Valli, "LPV modeling and LFT uncertainty identification for robust analysis: application to the VEGA launcher during atmospheric phase," in *Proceedings of the* 1^{st} *IFAC Workshop on Linear Parameter Varying Systems*, October 2015.

[8] E. Gadea, "Design of a robust controller for the VEGA TVC using the μ-synthesis technique," Master's thesis, Universitat Politècnica de Catalunya, Barcelona, Spain, 2011.

[9] D. Navarro-Tapia, A. Marcos, S. Bennani, and C. Roux, "Structured $\mathcal{H}_\infty$ control based on classical control parameters for the VEGA launch vehicle," in *Proceedings of the IEEE Multi-Conference on Systems and Control*, September 2016.

[10] C. Roux and I. Cruciani, "Scheduling schemes and control law robustness in atmospheric flight of VEGA launcher," in *Proceedings of the* 7^{th} *ESA International Conference on Spacecraft Guidance, Navigation and Control Systems*, June 2008.

[11] S. Rudy, A. Alla, S. L. Brunton, and J. N. Kutz, "Data-driven identification of parametric partial differential equations," *SIAM Journal on Applied Dynamical Systems*, vol. 18, no. 2, pp. 643–660, 2019. [Online]. Available: https://doi.org/10.1137/18M1191944

[12] S. Li, E. Kaiser, S. Laima, H. Li, S. L. Brunton, and J. N. Kutz, "Discovering time-varying aerodynamics of a prototype bridge by sparse identification of nonlinear dynamical systems," *Phys. Rev. E*, vol. 100, p. 022220, Aug 2019. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.100.022220

[13] D. Navarro-Tapia, "Robust and adaptive TVC control design approaches for the VEGA launcher," Ph.D. dissertation, University of Bristol, 2019.