Development of a Digital Twin for Electrified Aircraft Powertrain Health Management

Thomas W. Kennings
Intelligent Control and Autonomy Branch
NASA Glenn Research Center
Cleveland, Ohio, USA
thomas.w.kennings@nasa.gov

Donald L. Simon
Intelligent Control and Autonomy Branch
NASA Glenn Research Center
Cleveland, Ohio, USA
donald.l.simon@nasa.gov

***Abstract*— The augmentation of aircraft powertrains with electrical power systems is a promising path to reducing aircraft fuel consumption, emissions, and noise. Like conventional propulsion systems, electrified aircraft propulsion (EAP) systems will be subject to wear and tear throughout their lifecycles. System health management for EAP will enable efficient flight and maintenance scheduling, realizing economic, safety, and reliability benefits. A digital twin is, broadly, a dynamically updated virtual representation of an individual physical asset. This paper presents a Kalman filter-based approach for the development of a digital twin for an electrified powertrain and applies the approach to an EAP controls testbed. Measurements from nominal testbed operations are used to update a nonlinear model of the testbed. A Kalman filter is then created and used to identify and isolate anomalous testbed behavior based on measurements from off-nominal operations. Results show that the Kalman filter-based digital twin can monitor individual powertrain components' health for degradation or other changes in performance. The applicability of the presented digital twin approach to any hybrid- or fully-electrified powertrain is emphasized.**

Keywords—Electrified Aircraft Propulsion, Digital Twin, System Health Monitoring, Kalman Filter

I. INTRODUCTION

Reducing fuel consumption is of major interest to the commercial aviation industry because fuel represents a significant portion of airline operating costs. Emissions and noise reduction are also of interest to the commercial aviation industry for compliance with increasingly stringent industry requirements. One promising path to reducing fuel consumption, emissions, and noise is electrified aircraft propulsion (EAP), which is the augmentation or replacement of aircraft propulsion systems with electric powertrains. EAP systems present several unique engineering challenges that have become a focus of the National Aeronautics and Space Administration's (NASA) Aeronautics Research Mission Directorate.

Like conventional propulsion systems, hybrid- or fully-electric aircraft propulsion systems will be subject to wear and tear through their normal use or environmental effects. Electrification introduces new components to aircraft propulsion systems, necessitating appropriate monitoring of those components for degradation and faults. System health management (SHM) is an engineering and operations strategy that seeks to reduce lifetime operating costs and often realizes economic benefit by assessing and tracking system health, helping airlines to efficiently schedule flights and maintenance. Sudden changes in system performance due to internal faults are typically covered under a sub-field of SHM called fault management. Health and fault management for electrified aircraft propulsion systems will be necessary for their adoption from an economic as well as a safety and reliability perspective. In this paper, a digital twin approach is applied to a NASA EAP controls testbed with the intent of developing a SHM digital twin methodology applicable to electrified powertrains for EAP or other applications.

The remainder of this paper is organized as follows: Section II provides an overview of the EAP controls testbed. Section III reviews digital twins and the underlying theory applied in this work. Section IV discusses system modeling, characterization testing, and model parameter updates. Section V details the design of the digital twin, including the inputs, outputs, and matrices of the underlying Kalman filter. Results are presented and discussed in section VI. Finally, the paper is concluded in section VII.

II. HYBRID PROPULSION EMULATION RIG

The Hybrid Propulsion Emulation Rig (HyPER, Figure 1) is a testbed at the NASA Glenn Research Center used for hardware-in-the-loop evaluation of EAP control algorithms [1]. The HyPER testbed enables rapid, low-cost evaluation of novel EAP control methods compared to using actual turbomachinery.

Fig. 1. The HyPER testbed at the NASA Glenn Research Center

The testbed features two rotating shafts, each connected to two 66 kW electric motors, with a bi-directional 100 kW inverter per motor. The testbed architecture and naming convention is illustrated in Figure 2. Inverters 1 and 2 are connected to one DC bus, and inverters 3 and 4 are connected to another. Each DC bus is also connected to a 100 kW bi-directional power supply. An algorithm is used to scale the speed and power level of a simulated engine to the testbed hardware (and vice versa) while maintaining accurate system dynamics [2].

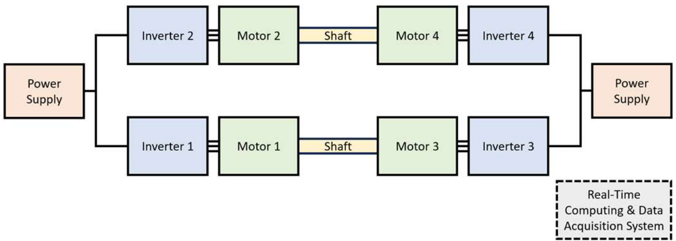

Fig. 2. Architecture and naming convention for the HyPER testbed

The present work considered the two shafts (along with their respective motors and inverters) as separate, independent systems. That is, a digital twin of similar architecture was constructed for each shaft system. While pairs of inverters share a power supply, their DC buses were modeled as being separate because DC voltage measurements varied slightly from inverter to inverter.

III. Digital Twin

The American Institute of Aeronautics and Astronautics (AIAA) defines a digital twin as "a set of virtual information constructs that mimics the structure, context and behavior of an individual / unique physical asset or a group of physical assets, is dynamically updated with data from its physical twin throughout its life cycle, and informs decisions that realize value" [3]. The virtual representation may take many forms, ranging from spreadsheets to detailed simulation models. The defining feature of a digital twin is that the virtual representation is connected to its physical counterpart by the transfer of data or information. Digital twin implementations can vary in scale and complexity, depending on the needs and resources of the system owner. The digital twin paradigm can be applied to each unit in a fleet as a logical way to organize the vast amounts of data returned fleetwide.

By continuously collecting and self-updating based on data from their physical counterparts, digital twins can reduce uncertainty about the status of their associated physical asset. By integrating information like part numbers, service history and usage data, digital twins can give operators a detailed understanding of how their system or fleet of systems is evolving in the field. From a SHM perspective, this translates to improved estimation of key system condition indicators, which enables efficient scheduling of operations and maintenance, ultimately realizing cost savings and efficient system or fleet utilization.

Another major benefit of a digital twin is that it provides an up-to-date model of its physical counterpart, which enables accurate model-based system analysis and engineering using powerful computational tools. The digital twin doesn't replace the need for physical tests and measurements, but it can help focus the scope of physical testing, resulting in time and cost savings. This is especially true for systems that require extensive and costly testing, such as for aircraft certification processes.

A. Kalman Filter

Inspired by prior development of aircraft engine on-board self-tuning models, the digital twin methodology presented in this paper applies an optimal linear state estimator known as a Kalman filter to tune model parameters based on system measurements and control commands [4, 5]. The Kalman filter calculates residuals between measurements and model predictions and, based on these residuals, estimates the system state. Included in the system state vector are health parameters reflective of the model's deviation from the physical system.

Kalman filter design requires a linear state-space model reflecting system dynamics around a given operating condition. The state, input, and output vectors (x, u, and y, respectively) should be chosen carefully for the sake of system observability. These vectors will imply the structure of the linear matrices A, B, C, and D. The Kalman filter also needs nominal steady state values of the state, inputs, and outputs (x_0, u_0, and y_0, respectively) at the designed operating condition.

The Kalman filter also requires two covariance matrices: The measurement covariance matrix, R, and the process covariance matrix, Q. The measurement covariance matrix, R, describes the covariance between measurements in the output vector, y, and affects the state estimate's susceptibility to measurement noise. The process covariance matrix, Q, represents uncertainty in the system dynamic model and affects how quickly or cautiously the Kalman filter adjusts its estimates in response to new measurements. The design of the Kalman filter's vectors and matrices is discussed further in section V.

In this study, steady-state Kalman filtering is applied. This means that while the Kalman filter is a dynamic system, the Kalman gain matrix, K, is pre-computed for the designed operating condition. To create a Kalman filter that spans the system's operating envelope, linear state-space models, covariance matrices, and the Kalman gain matrix are pre-computed at multiple operating conditions and their values are stored for later use. During execution, this structure is interpolated based on operating condition so that the Kalman filter is valid at any point within the system's operating envelope. The density of the pre-computed structure will affect the amount of linearization error caused by the interpolation.

To compute the Kalman gain matrix, the state estimation error covariance matrix, P, is first calculated by solving the following Riccati equation [6]:

$$P = APA^T - APC^T(CPC^T + R)^{-1}CPA^T + Q \tag{1}$$

The steady-state Kalman gain matrix, K, can then be calculated:

$$K = PC^T(CPC^T + R)^{-1} \tag{2}$$

During execution, at the beginning of each timestep, the residuals between the current input and output vectors and their respective nominal steady-state values are computed:

$$\Delta u_k = u_k - u_0 \tag{3a}$$

$$\Delta y_k = y_k - y_0 \tag{3b}$$

where u_k and y_k are composed of measurements from timestep k, and Δu_k and Δy_k are the residual vectors at timestep k. The Kalman filter is then provided Δu_k and Δy_k and produces an estimate of the state vector, $\hat{x}_k$, in a two-step process. First, an a-priori state residual estimate $\Delta\hat{x}_k^-$ is calculated using the a-posteriori state residual estimate from the previous timestep and the current input residual:

$$\Delta\hat{x}_k^- = A\Delta\hat{x}_{k-1}^+ + B\Delta u_k \tag{4}$$

Then, an a-posteriori state residual estimate $\Delta\hat{x}_k^+$ is calculated for the current timestep using the current a-priori state residual estimate and the current input and output residuals:

$$\Delta\hat{x}_k^+ = \Delta\hat{x}_k^- + K(\Delta y_k - C\Delta\hat{x}_k^- - D\Delta u_k) \tag{5}$$

The state estimate for the current timestep is then computed:

$$\hat{x}_k = x_0 + \Delta\hat{x}_k^+ \tag{6}$$

Since (4) depends on the result of (5) from the previous timestep, $\Delta\hat{x}_{k-1}^+$ must be initialized before the first timestep. In this study, it was initialized to zero.

IV. System Modeling and Characterization

A. Nonlinear HyPER Model

A nonlinear MATLAB/Simulink model of the HyPER testbed had previously been developed using NASA's Electrical Modeling and Thermal Analysis Toolbox (EMTAT) [7]. The model consists of connected blocks, each representing various testbed components including motors, inverters, shafts, cables, and power supplies. Model parameters, such as motor efficiencies and cable resistances, were previously populated using manufacturer-provided specifications or assumed values.

While the model was mostly sufficient for pre-test simulation and verification, the data used to populate model parameters may have been inaccurate due to incorrect modeling assumptions, manufacturing tolerances, and/or gaps in manufacturer-provided specifications. Additionally, the model did not have mechanisms to account for changes in system parameters over time.

B. Characterization Testing

Tests were conducted and measurements collected to characterize the performance of the HyPER testbed across its operating envelope. The operating envelope of each shaft's system (shaft, motors, inverters, and DC buses) was considered in terms of the speed of the rotating shaft and the torque produced by one of the attached motors (motor 3 or 4, for this study). Figure 3 illustrates the operating envelope considered in this study. Each shaft's operating envelope was identical. Like in a typical aircraft engine, the shafts of the HyPER testbed were only rotated in one direction.

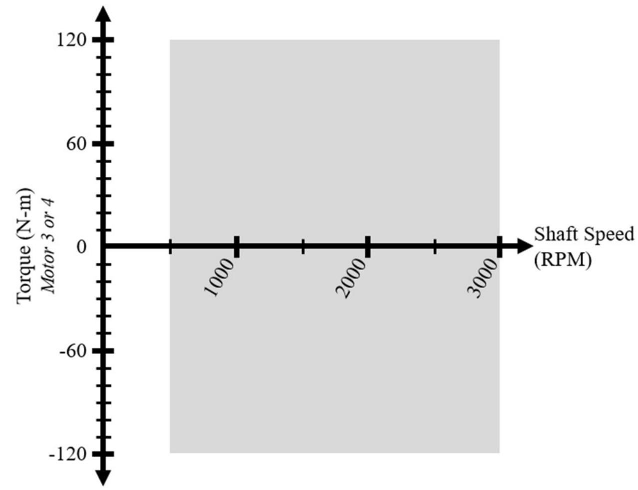

Fig. 3. Operating envelope (shaded) exercised during characterization testing

In all tests, one electric motor on each shaft applied a prescribed torque, stepping from +120 to -120 N·m in 10 N·m increments. Torque levels of ±5 N·m were also included to provide higher-fidelity characterization at low-torque conditions. At each torque level, a 30 s dwell time was observed for transient effects to diminish in order to collect steady-state data. Figure 4 shows the torque profile applied by the 'torque mode' motor for each test. The other motor maintained the shaft's rotational speed at the appropriate setpoint of 500, 1000, 1500, 2000, 2500, or 3000 rpm. Speeds lower than 500 rpm were not characterized due to changes in motor behavior and a lack of interest in operating the testbed at low speeds. Tests were repeated so that each motor had a turn controlling shaft speed. In total, twelve tests totaling 186 min were conducted to collect characterization data.

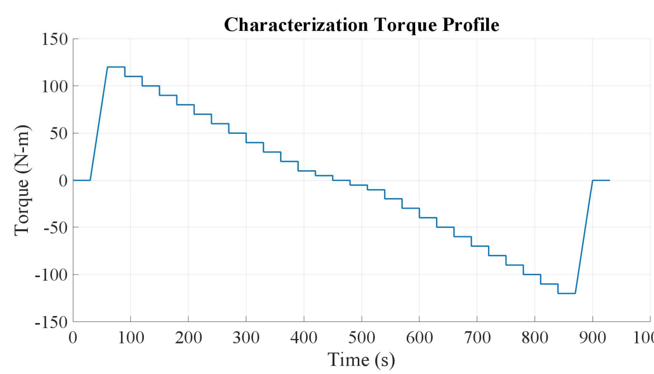

Fig. 4. Torque command profile for the 'torque mode' motor during characterization tests.

Various measurements taken internally by inverters (e.g., DC bus voltage and current, AC power) were collected with a sample period of 3 ms, which corresponds to a sample rate of 333.33 Hz. 20 kHz shaft speed measurements were also collected using optical encoders, and were later down sampled to 333.33 Hz before being supplied as inputs to the Kalman filter. All measurements were received and processed by a central real-time computing and data acquisition system.

C. Baselining Nonlinear HyPER Model Parameters

Data collected from characterization tests were used to perform an initial update to several parameters of the nonlinear HyPER model to align its performance with the real system. Table 1 summarizes the parameters updated during this step and the approaches taken to their calculation. The first-order equation used for fitting shaft damping coefficients and inertias was:

$$\tau_{net} = J\dot{N} + BN \tag{7}$$

where τ_{net} is the sum of the torques applied by the shaft's attached motors, J is the shaft inertia, B is the damping coefficient, and N and $\dot{N}$ are the shaft speed and acceleration, respectively.

TABLE I. MODEL PARAMETER CALCULATIONS

	Calculation Approach
Motor and Inverter Efficiency Maps	Comparing input and output powers (DC, AC, and/or mechanical) for each component
Shaft Damping Coefficient Maps	Fitting first-order shaft equation to steady-state data
Shaft Inertias	Fitting first-order shaft equation to transient data
Power Supply Voltages and Cable Resistances	Fitting Ohm's Law to voltage and current data

After calculating the updated parameters, they were applied to the nonlinear HyPER model's components. Figure 5 illustrates the improvements to model accuracy due to these updates when using real test data as model inputs. The updated model's outputs more closely matched the real-world measurements, demonstrating that the fitted parameters more accurately captured the steady-state behavior of the real testbed. The updated nonlinear model will be discussed later as an important piece of the digital twin.

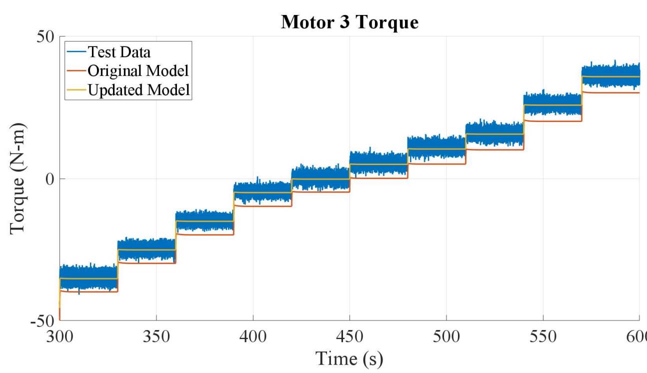

Fig. 5. Comparison of previous and updated HyPER models when real test data were used as model inputs.

V. STATE-SPACE MODEL AND KALMAN FILTER DESIGN

The two HyPER testbed shafts were treated as two fully independent systems, each with its own identically-structured Kalman filter. This approach was chosen to simplify the Kalman filter's structure and because prior experience with the testbed did not suggest any coupling between the shaft systems.

A. Selection of State Variables, Inputs, and Outputs

State, input, and output vectors were chosen carefully so that the performance of each individual powertrain component was observable. The state vector, x, was designed to include the shaft speed as well as a series of "health parameters." For example, the state vector for the shaft system containing motors 1 and 3 was:

$$x = [N \quad \delta V_{S,1} \quad \delta\eta_{I,1} \quad \delta\eta_{M,1} \quad \delta V_{S,3} \quad \delta\eta_{I,3} \quad \delta\eta_{M,3} \quad \delta\tau_3]^T \tag{8}$$

where N is the shaft's speed, $\delta V_{S,1}$ and $\delta V_{S,3}$ are the biases to the power supplies connected to motors 1 and 3, $\delta\eta_{I,1}$ and $\delta\eta_{I,3}$ are the biases to the efficiencies of inverters 1 and 3, and $\delta\eta_{M,1}$ and $\delta\eta_{M,3}$ are the biases to the efficiencies of motors 1 and 3, respectively. $\delta\tau_3$ is the bias to the torque produced by motor 3. Torque bias was applied to only one motor per shaft because it accounted for the net torque produced by both motors together. Motors 3 and 4 were chosen for torque biasing arbitrarily.

The input vector, u, was composed of the torque supplied by each motor. For the shaft system containing motors 1 and 3:

$$u = [\tau_1 \quad \tau_3]^T \tag{9}$$

The output vector, y, was composed of the shaft speed, DC bus voltages and currents, and AC powers. For the shaft system containing motors 1 and 3:

$$y = [N \quad V_{DC,1} \quad I_{DC,1} \quad P_{AC,1} \quad V_{DC,3} \quad I_{DC,3} \quad P_{AC,3}]^T \tag{10}$$

where $V_{DC,1}$ and $V_{DC,3}$ are the DC bus voltages, $I_{DC,1}$ and $I_{DC,3}$ are the DC bus currents, and $P_{AC,1}$ and $P_{AC,3}$ are the AC powers measured by inverters 1 and 3, respectively.

B. Linear State Space Model Matrices

The A, B, C, and D matrix structures followed the state, input, and output vector structures. Only matrix elements with known physical correlation were populated, making some matrices sparse. Since these matrices contained partial derivatives, their elements were computed by applying small perturbations to the nonlinear HyPER model and measuring the effects on relevant quantities. Updated model parameters were used in this process.

C. Kalman Filter Covariance Matrices

The measurement covariance matrix, R, was designed as a diagonal matrix where each diagonal element was the corresponding measurement's variance. This structure reflected the assumption that measurement noise was independent across sensors. The process covariance matrix, Q, was designed as a scaled identity matrix, reflecting the assumptions that process noise was independent for all state variables and that the state variables were of similar magnitudes.

The scale factor applied to the process covariance matrix provided a simple tuning parameter for adjusting the adaptation rate of the Kalman filter. For example, mechanical play in shaft couplings caused the torque produced by the motors to be noisy at low-torque operating conditions. To prevent the health parameter estimates from varying wildly in response, the process covariance matrix was scaled to significantly reduce the filter's adaptation rate only at those operating conditions.

D. Piecewise-Linear Construction

A set of Kalman filter matrices and nominal state, input, and output vectors were computed every 25 rpm and every 1 N·m spanning the testbed's operating envelope, forming a piecewise-linear Kalman filter. During execution, shaft speed and motor torque data were used to linearly interpolate these piecewise structures, and the interpolated matrices and vectors were used by the Kalman filter to compute (3a)-(6). The health parameter estimates were then added to their respective quantities in the nonlinear model, causing the corrected model's outputs to closely match the real data. Close matching between the corrected model and real measurements suggested that the estimated health parameters were accurate. An alternative design was attempted in which health parameters were instead implemented as *multipliers* for their respective quantities, but this approach suffered from sensitivity issues, particularly at low-torque or low-speed operation. Figure 6 illustrates the flow of information throughout the digital twin.

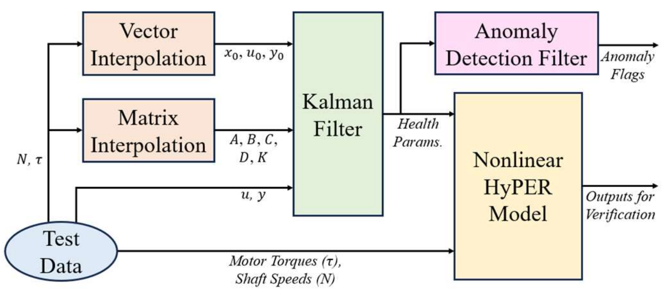

Fig. 6. Digital twin block diagram showing the flow of information throughout the Kalman filter, the nonlinear HyPER model, and the anomaly detection filter.

E. Anomaly Detection Filter

System anomalies or degradation were expected to cause an increase in the magnitude of health parameter estimates. Accordingly, an anomaly detection filter was designed to identify such changes. In this study, the sum of squared health parameters was compared to a fixed threshold to detect anomalies. Equal weights were applied to each health parameter in the summation. The threshold value was chosen based on observation of the sum of squares signal during nominal and off-nominal tests.

VI. Results and Discussion

The ability of the digital twin to self-tune and detect anomalies was evaluated using several sets of data from the HyPER testbed. These evaluations were performed post-test; the Kalman filter was not executing in real-time. The data used for these evaluations were separate from the data used to initially update the nonlinear model. Measurements were collected with motor 1 maintaining a shaft speed of 1000 rpm while motor 3 swept torque from +120 to -120 N·m. For the other shaft system, motor 4 maintained shaft speed while motor 2 presented the sweeping torque load. This configuration meant that power was circulated throughout the testbed, which minimized the strain on the power supplies. By sweeping the torque load, the piecewise-linear Kalman filter was evaluated at operating conditions between those at which it had been pre-computed.

Figures 7 and 8 show the health parameter estimates and the anomaly detection signal for the shaft containing motors 1 and 3 in response to both nominal and anomalous test data. Their content will be further discussed in the following subsections. In all cases, the outputs of the corrected nonlinear HyPER model closely matched the real test measurements, demonstrating the Kalman filter's effectiveness in tuning the model.

A. Nominal Operation

State estimates and the anomaly detection signal produced in response to nominal test data are shown in blue (blue traces are the same in Figure 7 and Figure 8, although the y-axis scales are different). As expected, the Kalman filter estimated very small health parameters because the testbed's condition hadn't significantly changed since updating the model's parameters. Some estimates experienced startup and shutdown transients due to the shaft's speed traversing between 0 and 500 rpm (t=0 to 15 s and t=1005 to 1015 s), which was outside the digital twin's operating envelope. At low-torque operating conditions (approximately t=400 to 600 s), state estimates changed very little because the Kalman filter's process covariance matrix was scaled to reduce the filter's adaptation rate in response to increased process noise, as discussed in section V.

B. Induced Voltage Anomaly

To test the Kalman filter's ability to isolate anomalous conditions, motor 1's DC bus was purposely reduced from the nominal 350 V down to 330 V before conducting the test. Other than the off-nominal bus voltage, this test was exactly the same as the nominal test. Speed and torque measurements confirmed that the modified bus voltage did not impact the operating conditions reached throughout the test. Results for this case are depicted by the orange traces shown in Figure 7. Here, the Kalman filter correctly identified a -20 V health parameter for motor 1's DC bus voltage. Unexpectedly, the Kalman filter also identified nonzero health parameters for the connected motor and inverter efficiencies. It is suspected that the DC bus voltage had an unmodeled effect on the inverter's efficiency, though AC power measurements were not detailed enough to fully understand this effect. Future work may more closely investigate the effects of sensor selection, positioning, and accuracy on health estimate ambiguity. The "anomaly detection" threshold (denoted by the dashed black line in the lower left subplot) was exceeded during this test, demonstrating the anomaly detection filter's ability to detect system misconfiguration.

C. Swapping Shaft Data

To evaluate the digital twin's sensitivity, data collected from each of the testbed's shaft systems were swapped before being sent to the Kalman filter. The data were interchanged between motor/inverters 1 and 4, and between motor/inverters 2 and 3, so that the data were still representative of the components acting in torque mode vs. speed control mode. Although the two shaft systems used hardware of identical make and model, there were subtle differences in their components' performances due to manufacturing tolerances or uneven degradation. The orange traces in Figure 8 show the Kalman filter's response to this situation. Health parameter estimates changed significantly and the anomaly detection threshold was reached, demonstrating that the digital twin uniquely represents its specific physical asset and is sensitive to changes in system health.

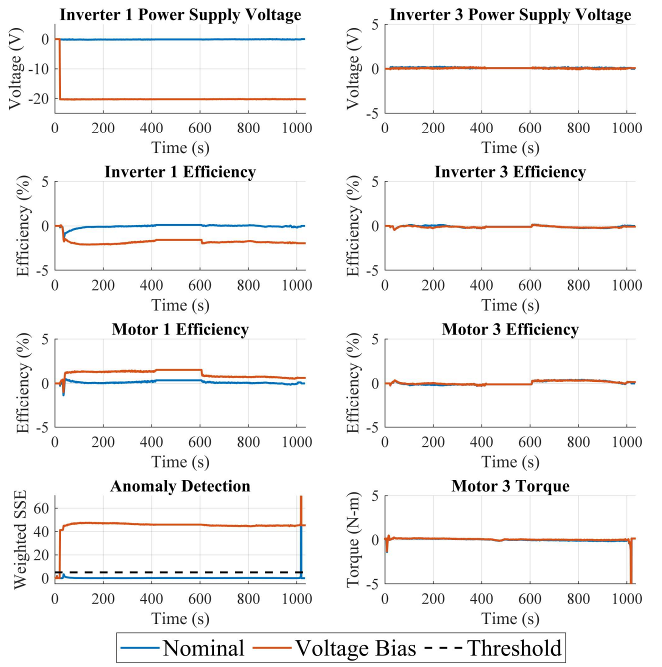

Fig. 7. Health parameter estimates and calculated anomaly detection signals for nominal and induced voltage anomaly tests.

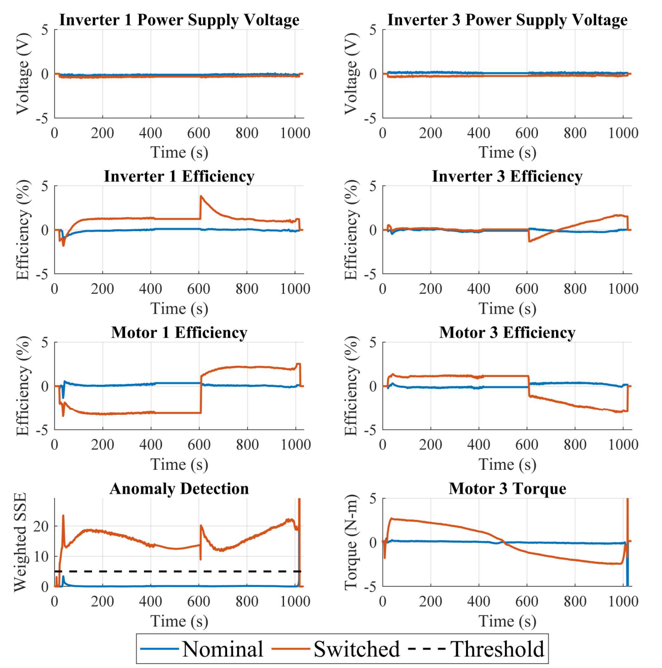

Fig. 8. Health parameter estimates when test data were organized correctly vs. when switched across shaft systems before being sent to the Kalman filter.

While the anomaly detection filter demonstrated in this study used equal weights and an ad hoc threshold, weights and thresholds could be selected to optimize for sensitivity to certain components based on the system owner's maintenance and reliability posture, the consequences of degradation in specific components, and the susceptibility of measurements to noise. More advanced methods could be used to evaluate health parameters to diagnose internal faults, including monitoring for sudden changes, comparison against other units in a fleet, and the use of fault tree logic or machine learning.

To apply the digital twin paradigm to a fleet of systems, data could be collected and centrally stored for each unit in the fleet representing its condition, usage, and maintenance history. The Kalman filter could then update each unit's health parameters based on incoming data related to that unit's recent usage. Such an approach could enable health trend monitoring, detailed diagnostics, and efficient scheduling of flights and fleet maintenance, ultimately realizing economic value to the system owner. Information technology infrastructure for collecting and processing data from the fleet's many units would be necessary to accomplish this. Most aircraft operators have existing infrastructure for the collection and analysis of fleetwide flight data, though EAP and the digital twin approach discussed here may necessitate the expansion of such systems.

VII. Conclusions

This paper presents a Kalman filter-based approach for constructing a digital twin of an electrified powertrain. A digital twin in this context is, broadly, a dynamically updated virtual representation of a specific physical asset. Measurement data from the physical testbed were used to demonstrate the digital twin's ability to identify and isolate off-nominal system performance. The digital twin identified minor, expected differences between a nonlinear testbed model and nominal test data acquired from the physical system. The digital twin was then demonstrated against several off-nominal testbed conditions, proving that it can detect system misconfiguration and is sensitive to changes in system health. Future work could focus on the effects of sensor selection and positioning, real-time execution of the digital twin for rapid fault detection and isolation, and enhancement of the anomaly detection filter.

Acknowledgment

The authors would like to acknowledge the Advanced Air Transport Technology and the Subsonic Vehicles Technologies and Tools projects for funding this work. Thank you to Marcus Horning for aiding in operating the HyPER testbed.

References

[1] H. Buescher, D. E. Culley, S. J. Bianco, et al., "Hybrid-electric aero-propulsion controls laboratory: Overview and capability," presented at the AIAA SciTech Forum, Jan. 2023.

[2] Bianco, S. J., Simon, D. L., and Hill, E. D., "Adaptive control and scaling approach for the emulation of dynamic subscale torque loads," AIAA Paper No. 2024-2356, Jan. 2024.

[3] "Digital twin: Definition & value, an AIAA and AIA position paper," American Institute of Aeronautics and Astronautics, Inc., Dec. 2020.

[4] Luppold, R. H., Roman, J. R., Gallops, G. W., and Kerr, L. J., "Estimating in-flight engine performance variations using kalman filter concepts," AIAA 25th Joint Propulsion Conference, Paper No. AIAA-89-258, 1989

[5] T. Brotherton, A. Volponi, R. Luppold, and D. Simon, "eSTORM: Enhanced self tuning on-board real-time engine model," in Proceedings of the 2003 IEEE Aerospace Conference, Mar. 2003.

[6] D. Simon, Optimal State Estimation: Kalman, H∞, and Nonlinear Approaches. Hoboken, NJ: John Wiley & Sons, Inc., 2006.

[7] M. Bell and J. Litt, "An electrical modeling and thermal analysis toolbox for electrified aircraft propulsion simulation," presented at the AIAA Propulsion and Energy Forum 2020, Aug. 2020.