Design of an Application System for Typical Aero-Engine Digital Flight-Test Simulation Based on Systems Engineering

GUO Zhengbo^{1,2}

2026-06-22

ChinaRxiv

Abstract

With the application objective of digital flight testing for typical aero-engine subjects, and based on a systems engineering mindset, the design of a simulation application system for typical digital flight testing of aero-engines is carried out from the aspects of requirements analysis, objective confirmation, architecture design, and system design. The system includes a structured definition and analysis module for simulation objectives, a simulation business architecture analysis and design module, a mission simulation design module, a test parameter and additional measurement point analysis and determination module, a mission simulation comparison and intelligent monitoring module, and a comprehensive result evaluation module. It supports model-based engine test design, test optimization, safety monitoring, and result evaluation, realizes digital flight test simulation and application verification for aero-engines, can meet the needs arising from the transformation toward a new model of digital aero-engine development, and accelerates the advancement of digital engineering construction and application for aero-engines.

Full Text

Design of an Application System for Typical Aero-Engine Digital Flight-Test Simulation Based on Systems Engineering

GUO Zhengbo1,2

(1. Engine Department, Chinese Flight Test Establishment, Xi' an 710089, China;
2. School of Power and Energy, Northwestern Polytechnical University, Xi' an 710129, China)

Abstract: With typical aero-engine digital flight-test subjects as the application target, an application system for typical aero-engine digital flight-test simulation was designed based on a systems-engineering mode of thinking. The design was carried out in terms of requirements analysis, objective confirmation, architecture design, and system design. The system includes modules for structured definition and analysis of simulation objectives, simulation business architecture analysis and design, mission simulation design, analysis and determination of test parameters and additionally installed measurement points, mission simulation comparison and intelligent monitoring, and comprehensive result evaluation. It supports model-based engine test design, test optimization, safety monitoring, and result evaluation, realizing aero-engine digital flight-test simulation and application verification. It can meet the needs arising from the transformation toward new digital development models for aero-engines and accelerate the construction and application of digital engineering for aero-engines.

Keywords: systems engineering; digital flight test; aero-engine; requirements analysis; architecture design; simulation application system

Chinese Library Classification No.: V231.3
Document code: A
DOI: 10.11776/j.issn.1000-4939.2026.03.009
Article No.: 1000-4939(2026)03-0589-09
CSTR: 32414.14.cjam.issn.1000-4939.2026.03.009

Design of a typical digital flight test simulation application system for aero-engines based on systems engineering

GUO Zhengbo1,2

(1. Engine Department of Chinese Flight Test Establishment, 710089 Xi' an, China;
2. School of Power and Energy, Northwestern Polytechnical University, 710129 Xi' an, China)

Abstract: With the goal of digital flight testing for typical aero-engine subjects, a simulation application system for aero-engine digital flight testing was designed based on a systems engineering mindset, covering demand analysis, objective confirmation, architecture design, and system design. This includes modules for structured definition and analysis of simulation objectives, simulation business architecture analysis and design, task simulation design, determination of test parameters and installation of monitoring points, task simulation comparison and intelligent monitoring, and comprehensive result evaluation. It supports model-based engine testing design, test optimization, safety monitoring, and result evaluation, enabling digital flight simulation and application verification for aero-engines. This system can adapt to the evolving needs of new digital development models for aero-engines, accelerating the construction and application of digital aero-engine projects.

Date received: 2025-08-26
Funding project: Special Scientific Research Project for Civil Aircraft of the Ministry of Industry and Information Technology (No. MJZ3-2N22)
Corresponding author: GUO Zhengbo, Senior Engineer. E-mail: 429547513@qq.com
Citation format: GUO Zhengbo. Design of an application system for typical aero-engine digital flight-test simulation based on systems engineering[J]. *Chinese Journal of Applied Mechanics*, 2026, 43(3): 589–597.
GUO Zhengbo. Design of a typical digital flight test simulation application system for aero-engines based on systems engineering[J]. *Chinese Journal of*

Applied Mechanics, 2026, 43(3): 589–597.

Key words: systems engineering; digital flight testing; aero-engine; requirement analysis; architecture design; simulation application system

Systems engineering is an interdisciplinary and systematic methodology. Regarding the connotation of systems engineering, Wang Yingluo[1] proposed that systems engineering is a structured development process from concept to operation, formed by scientifically defining user requirements, simultaneously considering operation, performance, testing, and manufacturing, and carrying out integrated design. Liang Di et al.[2] proposed that, based on systems engineering, the key nodes for the efficient design and development of complex industrial systems in interdisciplinary and multi-domain application scenarios can be identified from different perspectives. Guo Fang et al.[3] proposed that the systems engineering V model is a highly structured life-cycle model in systems engineering development. It describes the entire process from requirements analysis to system delivery, with particular emphasis on the parallel planning and correspondence between development and testing, as well as the traceability of requirements. A typical systems engineering V model is shown in Fig. 1.

Visible labels in Fig. 1: requirement definition; requirements analysis; functional analysis; architecture design; design implementation; verification and validation plan; component integration and verification; subsystem integration and verification; system integration and verification; comprehensive verification and validation.

Fig. 1 Typical block diagram of the systems engineering V model

Driven by the current demand for accelerated development of digital engineering for aviation equipment, Wu Yunhao et al.[4] proposed that the use of systems engineering methods can effectively integrate big data analysis, advanced computing technology, cyber-physical integration technology, and other technologies, thereby realizing full-life-cycle management of aviation equipment in requirements analysis, design, development, verification, validation, operation, and support. Zhao Min et al.[5] proposed adopting digital engineering to build a digital business model oriented toward the product development process, so as to realize the digitalization of aerospace equipment.

As the "heart" of an aircraft, the aero-engine has long been a "serious concern" constraining the development of China' s aviation industry. Promoting and applying digital technologies in this field as soon as possible is of great significance for enhancing national core competitiveness. Huang Weina et al.[6] analyzed the current implementation status and development trends of digital engineering abroad and, in combination with the foundation of digital construction in China' s aero-engine industry, proposed an overall technical framework for aero-engine digital engineering and provided development suggestions at the implementation level. Luo Tingting[7] studied a requirements management method for commercial aero-engines based on systems engineering, constructed a requirements management system, and improved the requirements management capability for commercial aero-engines. Xi Weiwei[8] proposed applying DOORS software to requirements management in the development of aero-engine FADEC systems, thereby improving the efficiency of R&D requirements confirmation. Shi Yanyan et al.[9], in order to realize modularization of aero-engine requirements management, studied an aero-engine requirements management model based on product breakdown structure, improving the requirements management capability throughout the aero-engine life cycle. Lu Chuanchuan et al.[10] proposed the overall framework, organizational model, and implementation path for requirements management in military aero-engine development. Ji Yan et al.[11] studied the requirements management process and method for forward design of civil aero-engines based on systems engineering thinking, laying a foundation for requirements-driven forward R&D of aero-engines. Gao Yuanyuan et al.[12], by analyzing the current status of digitization of aero-engine test benches, summarized the requirements and technical application characteristics of such links as automated bench testing, online analysis of test data, test fault diagnosis, and test process management, thereby guiding the digital transformation of test benches.

As a key link in product R&D, flight testing is the most important verification test in the full life cycle of aero-engine products. Jiao Yang[13] pointed out that realizing the digital transformation of flight testing plays a vital role in improving the development efficiency and reliability of engine products and reducing the development cycle and development cost. Analysis of relevant domestic and international literature shows that there are few reports related to digital flight testing of aero-engines. How to effectively adopt systems engineering design thinking to develop a simulation application platform for whole-engine-level digital flight-test verification of aero-engines, and to realize efficient and safe digital-intelligent flight testing of aero-engines, is an urgent problem that must be solved in the digital transformation of China' s current engine flight-test industry.

The main problems currently faced in the transformation stage of digital flight testing for aero-engines in China are as follows: In the flight-test design stage, Liu Siyu et al.[14] preliminarily carried out research on flight-test point design based on discrete optimization and other methods for engine performance-characteristic flight-test subjects. At present, however, forward-design software or systems for flight testing at the tool level have not yet been formed, and application in model flight testing is limited. In the flight-test implementation and flight-test evaluation stages, Pan Pengfei[15] proposed research on aero-engine condition monitoring and fault diagnosis methods based on flight-test data, but software tools or systems for engine intelligent monitoring and result analysis and evaluation oriented toward model applications have not yet been formed, and application in model flight testing is limited.

In view of the main problems currently faced by digital flight testing of aero-engines, this study, guided by the thinking mode of the systems engineering V model, proposes a systems-engineering-based design method for an integrated simulation application system for digital flight testing of typical aero-engine subjects. It preliminarily realizes key functions such as scientific flight-test design, intelligent flight-test monitoring, and accurate quantitative evaluation of flight-test results, thereby promoting the accelerated digital transformation of the flight-test mode.

1 Requirements Analysis

According to user requirements and the expected objectives, the simulation application system should have the following main functions: structured definition and analysis of digital flight-test simulation objectives; analysis and design of the digital flight-test simulation business architecture; simulation design for digital flight-test missions; analysis and determination of digital flight-test parameters and additional instrumentation measurement points; simulation comparison and intelligent monitoring of digital flight-test missions; and comprehensive evaluation of digital flight-test results. The requirement analysis for each function is as follows.

1) **Structured definition and analysis of digital flight-test simulation objectives:** This function mainly performs simulation-objective analysis for key processes such as mission design, test modification, flight-test comparison and safety monitoring, and result evaluation for performance subjects. It simulates performance flight-test missions through simulation objectives and analyzes the results, providing reference results before the test.

2) **Analysis and design of the digital flight-test simulation business architecture:** This function defines and generates simulation architecture framework diagrams through visual drag-and-drop operations, defines the elements of each link in the full performance flight-test process, and builds a functional model of the full-process simulation architecture system.

3) **Simulation design for digital flight-test missions:** This function generates an optimal test-point matrix by carrying out the overall activity process of "preliminary design of the flight-test matrix → mission simulation deduction → iterative optimization of test design → determination of the mission profile."

4) **Analysis and determination of digital flight-test parameters and additional instrumentation measurement points:** This function includes two parts: optimization and determination of test parameters, and optimization and determination of measurement points for test parameters. Its core objective is to determine a reasonable and feasible optimal layout scheme for additional parameter measurement points.

5) **Simulation comparison and intelligent monitoring of digital flight-test missions:** Flight-mission comparison is mainly performed

by invoking models and parameters from the foundational database and comparing them with actual flight-test results, so as to evaluate the correctness and accuracy of the parameters for the objectives of a single sortie. Flight-safety monitoring mainly invokes health-assessment and condition-monitoring models to perform safety-monitoring functions such as full-state parameter evaluation, performance monitoring, and fault warning during the flight process.

6) **Comprehensive evaluation of digital flight-test results:** This function mainly includes "flight conclusion evaluation" and "test conclusion extension." For engine-characteristic flight-test subjects, relevant models, evaluation criteria, airworthiness clauses, and other items in the foundational database are invoked to comprehensively evaluate the combined digital-and-real flight results at the summary stage.

2 System Design

2.1 System Organizational Architecture Design

The system composition mainly includes the basic support layer, resource element layer, general framework layer, algorithm model layer, business algorithm layer, and capability generation layer.

1) **Basic support layer**

The basic support layer mainly consists of underlying supporting infrastructure such as hardware servers and modeling and simulation terminals. It clarifies guiding theories, basic concepts, engineering methods, operating modes, and evolution rules, providing fundamental compliance requirements and constraint rules for the construction of business information systems.

Diagram text: high-performance data servers; modeling and simulation terminals; domestic operating system; theoretical system and standards/specifications; basic support layer.

Fig. 2 Architecture diagram of the basic support layer

2) **Resource element layer**

The resource element layer is the cornerstone of the digital information system, covering basic equipment data resources, document data resources, model data resources, simulation and test data resources, and so on. The data involved mainly include: measured data, including engine-installed ground-test and flight-test data; knowledge information, including relevant GJB and other standards, test matrices, technical documents, technical reports, etc.; simulation data, including three-dimensional geometric model data and simulation data, etc.

Diagram text: equipment digital models; GJB standards and regulations database;

document database; simulation database; flight-test database; test matrix; resource element layer.

Fig. 3 Diagram of resource element layer architecture

3) **General framework layer**

The basic general framework includes the desktop program development framework, the Web program development framework BootSpring, the logging system framework, the three-dimensional visualization display framework, the general basic function framework, the unified data access framework, and so on. The general framework layer provides foundational reusable modules and components for building the entire system, improving the efficiency and stability of system development.

Diagram text: desktop program framework; logging system framework; 3D visualization engine VTK; general mathematical function library; Web program framework; artificial intelligence algorithm library; 3D scene visualization engine OSG; unified data access library; general framework layer.

Fig. 4 Diagram of general framework layer architecture

4) Algorithm Model Layer This is the core layer of the simulation application system, including two parts: a general algorithm model library and a simulation software library. The general algorithm models include DOE experimental design models, surrogate models, test-data processing algorithms, multi-source heterogeneous simulation integration framework FMI libraries, process templates, and process automation components. The simulation calculation programs and software include performance simulation calculation, flow-field simulation calculation, structural simulation calculation, FMI multidisciplinary collaborative simulation frameworks, etc., which can provide fundamental finite-element-based simulation computing capabilities and services of various types.

Diagram text:

- Performance simulation calculation; Flow-field simulation calculation
- Structural simulation calculation; FMI/FMU
- Geometric modeling algorithm model; DOE experimental design model
- Test-data processing algorithm; Process templates and process automation
- Optimization design algorithm model; Surrogate model (approximation model)
- Multi-source heterogeneous simulation integration framework; Role-based permission management
- Algorithm model layer

Figure 5. Diagram of algorithm model layer architecture

Fig. 5 Diagram of algorithm model layer architecture

5) Business Algorithm Layer This layer provides specific algorithms and models related to the business of the simulation application system. It is mainly used to support the business logic of the entire simulation application system and, upward, to support the formation of various capabilities and services. The business algorithm layer mainly includes one-dimensional engine performance evaluation methods, flight-test mission design methods, flight-test mission optimization algorithms, data processing methods, flight-test mission comparison and monitoring methods, and comprehensive evaluation methods for flight-test results.

Diagram text:

- One-dimensional engine performance evaluation method; Test mission design method
- Flight data preprocessing method; Flight-test mission comparison method
- Health monitoring model; Test mission optimization method
- Comprehensive evaluation method for flight-test results; Multidisciplinary simulation of engine flight test
- Business algorithm layer

Figure 6. Diagram of business algorithm layer architecture

Fig. 6 Diagram of business algorithm layer architecture

6) Capability Generation Layer On the basis of the resource layer and the model layer, and centered on flight-test business processes and data-processing business, this layer carries out capability-layer integration, including the flight-test simulation objective analysis module, flight-test simulation architecture design module, flight-test mission design and optimization module, test modification optimization module, digital flight-test mission simulation comparison and intelligent monitoring module, digital flight-test result comprehensive evaluation module, etc. It realizes full-process automated simulation and evaluation for flight-test subjects, effectively improving the digital testing capability for powerplants and fuel systems.

Diagram text:

- Digital flight-test simulation objective structured definition and analysis module; Digital flight-test simulation business architecture analysis and design module
- Digital flight-test mission simulation design module; Digital flight-test test-parameter and added measurement-point analysis and determination module
- Digital flight-test mission simulation comparison and intelligent monitoring module; Digital flight-test result comprehensive evaluation module
- Capability generation layer

Figure 7. Diagram of capability generation layer architecture

Fig. 7 Diagram of capability generation layer architecture

2.2 Business Architecture Design

From the business perspective, the simulation application system is generally based on actual flight-test business processes, and founded on flight-test requirements, specifications, experiential knowledge, etc. Through digital technologies and software development technologies, it constructs systems and tools that conform to the processes and requirements. The core of the business architecture of the simulation application system includes two parts: management and maintenance of flight-test-related resources, including standards and specifications, technical reports, algorithm programs, model data, etc.; implementation of tools and services related to the flight-test mission process, forming unified, standardized, and easy-to-use software. The two communicate and interact through data service interfaces.

The core business of the simulation application system takes flight-test requirements, mission documents, etc., as inputs. Through a series of processes –flight-test mission design and optimization → test parameters and added measurement-point optimization → flight-test mission comparison and monitoring → comprehensive evaluation of flight-test results, etc.–it generates flight-test reports and corresponding data, and stores them in the model and database of the simulation application system for subsequent use, thereby forming, as a whole, a standardized, interactive, and mutually supportive business system.

2.3 Development of Core Functional Modules

2.3.1 Digital Flight-Test Simulation Objective Structured Definition and Analysis Module

This module focuses on the intelligent generation and dynamic optimization of flight-test outlines for performance-subject flight tests. It supports manual input or automatic parsing of performance assessment requirement documents. Based on natural language processing or rule-engine technologies, it accurately extracts key indicators, boundary conditions, and verification logic, and constructs a reusable and extensible digital flight-test assessment objective library. Relying on multi-source simulation business models, the module maps abstract objectives into input constraints for submodules such as mission simulation design, test parameters and measurement-point layout, simulation comparison monitoring, and result evaluation. It realizes the automated orchestration of test processes, resource allocation, and evaluation criteria, improves the scientific validity and traceability of flight-test schemes, and provides standardized and intelligent decision support for efficient verification of complex performance subjects.

2.3.2 Flight-Test Simulation Business Architecture Analysis and Design Module

This module is based on model-based systems engineering (model-based sys-

With the model-based systems engineering (MBSE) philosophy as its core, the SysML language is used to perform multidimensional, multi-granularity architectural modeling of the entire process of engine-performance flight-test experiments. It supports drag-and-drop construction of package diagrams, block diagrams, parametric diagrams, sequence diagrams, activity diagrams, and so forth, enabling graphical traceability along the requirement–function–logic–physical chain. Through SysML module definitions, it automatically parses the hierarchical relationships and interaction protocols among physical elements such as the engine body, test sensors, data collectors, excitation controllers, and ground stations. Based on activity diagrams and state machines, behavioral modeling is carried out for key stages in performance-subject flight-test processes, such as starting, steady state, acceleration and deceleration, and in-flight relight, ensuring complete constraints on timing, state, and data dependencies. Finally, a reusable and extensible architecture library is generated, as shown in Fig. 8.

Fig. 8 SysML model template library for flight-test simulation business

The SysML model output by the simulation architecture design module can be integrated with subsequent simulation execution, test-script generation, and test-result evaluation, thereby realizing full-lifecycle digital management of engine-performance flight-test experiments from top-level design to verification closed loop.

2.3.3 Flight-Test Mission Simulation Design Module

This module uses existing basic-library resources to conduct automated design, simulation deduction, and iterative optimization of flight-test missions. Oriented toward the requirements of engine-performance verification, the module constructs an automated design framework centered on the test matrix. First, it interfaces with the basic library to complete the unified encapsulation of historical flight-test data, providing a highly reliable data source for subsequent modeling. Then, response-surface methods and other approaches are used to train proxy models for key performance indicators such as thrust and fuel-consumption rate. In this way, the complex coupling relationship between design variables and response quantities is rapidly captured through nonlinear mapping, significantly reducing the number of calls to high-fidelity simulations. The proxy-model generation interface is shown in Fig. 9.

During the construction of the test matrix, an initial candidate set is generated and combined with single-objective or multi-objective optimization algorithms such as particle swarm optimization and genetic algorithms, as shown in Fig. 10. Under the premise of satisfying test requirements, a solution set with the minimum number of flight-test sorties is obtained, thereby realizing joint optimization of the test configuration, altitude–velocity profile, throttle schedule, and measurement-point configuration.

Fig. 9 The generation settings of the proxy model

Fig. 10 Interface screenshot of optimization algorithm settings

2.3.4 Flight-Test Parameter and Installation Measurement-Point Analysis Module

By integrating basic-library resources, this module enables the software to support parameter feature analysis, sensitivity analysis, measurement-point layout optimization, and the generation of optimization reports, so as to assist users in making data-driven decisions. Oriented toward the accuracy requirements of engine-performance verification, it constructs a measurement-point optimization framework with a closed loop of "data–model–decision." First, the module reads historical flight-test data and performs time-domain statistical analysis and frequency-domain power spectral analysis on key test parameters such as thrust, temperature, and pressure. Subsequently, sensitivity algorithms are used to quantify the contribution of each parameter to the performance indicators; highly sensitive variables are screened and redundant variables are eliminated, thereby reducing the complexity of subsequent measurements. The sensitivity-algorithm settings interface is shown in Fig. 11.

Fig. 11 Sensitivity analysis settings interface screenshot

For the measurement rake structure, finite-element modal analysis is used to obtain the first six natural frequencies and mode shapes, ensuring that the dynamic stiffness and vibration-resistance margin under the engine's broadband excitation meet the requirements. Furthermore, through CFD

The flow-field simulation evaluates the disturbance effects on the inlet duct, compressor inlet, and combustor-exit flow fields after the measurement rake is installed; calculates the total-pressure distortion index, the increase in turbulence intensity, and possible induced separation regions. Finally, the module automatically generates a parameter-sensitivity ranking, modal safety-margin contour maps, and flow-field reports, thereby realizing model-based, data-driven, and traceable decision-making for measurement-point layout.

2.3.5 Digital Flight-Test Mission Simulation Comparison and Intelligent Monitoring Module By integrating high-performance computational models and real-time monitoring algorithms, this module enables the software to predict, evaluate, and warn of issues that may affect flight safety. At the same time, after flight, it updates the relevant models and databases to ensure continuous performance improvement and risk reduction.

With high-fidelity model fusion and real-time computation as its core, the module constructs a "prediction–comparison–correction" closed-loop system oriented toward engine performance verification. The module uses a model bus to call the engine performance model, parameter prediction model, and fault diagnosis model in the basic library, forming an integrated simulation engine. Users can revise model hyperparameters and fault thresholds according to the results, ensuring continuous improvement of model accuracy throughout the full life cycle and providing risk-reduction decision support for subsequent sorties.

2.3.6 Comprehensive Evaluation Module for Flight-Test Results This module is used to analyze and evaluate flight-test results. By integrating a variety of evaluation tools and models, the software can comprehensively assess engine performance under different flight conditions and generate detailed test reports.

The module performs time alignment and cleaning of measured data over the entire flight-test process. Based on interpolation and reconstruction algorithms, it generates a continuous four-dimensional response surface of "altitude–speed–thrust–fuel consumption rate," enabling immediate visualization and interactive comparison of any operating point within the envelope. It automatically extracts the mean, variance, confidence interval, and exceedance frequency of key performance indicators, and completes multidimensional difference analysis against the baseline simulation results.

To enable data analysis at points outside the available data, this module adds a parameter prediction function. Through methods such as linear interpolation, linear regression, and neural-network models, as shown in Fig. 12, users can analyze arbitrary points inside and outside the entire flight envelope, thereby strengthening validation of the results.

Visible interface text in Fig. 12: "Linear interpolation algorithm" ; "Simple linear regression prediction" ; "Neural network algorithm" ; "Select matrix: simulation test data" ; "Interval time: 2" ; "Number generated: 10" ; "Start analysis (for data at future time points)."

Fig. 12 Interface diagram of the numerical prediction function
Fig. 12 Interface screenshot of numerical prediction function

After result evaluation, the user needs to specify the data required for the report. The software automatically reads the required content from the data, including but not limited to evaluation-related parameters in the flight-test project, simulation results, atmospheric environment, flight attitude, engine operating state, principal unmeasurable component characteristic parameters, major system health-status and operating-status functional and performance parameters, and the underlying engine boundary-performance data outside the aircraft-platform envelope. The software automatically generates a test report conforming to the template and automatically adds the parameters to the relevant locations in the document.

3 Typical Function Verification

Based on the designed digital flight-test simulation application system for aeroengines, typical function verification was carried out for aeroengine performance-characteristic flight-test subjects. The results are as follows.

1) The structured target definition and analysis module was used to complete the writing of performance flight-test content and the generation of the test-point matrix, and to export the flight-test outline, as shown in Fig. 13.

Visible interface text in Fig. 13: "Engine flight-test outline preview"; "Download (DOCX)"; "Export PDF"; "Test items, methods, and requirements"; "Engine performance characteristic inspection test."

Fig. 13 Export function for the engine performance flight-test outline
Fig. 13 Export of engine performance flight-test outline

2) The architecture analysis and design module was used to draw system architecture diagrams for the engine performance-characteristic flight-test subject, including test use-case diagrams, sequence diagrams, activity diagrams, and state diagrams. Part of the architecture is shown in Figs. 14–16.

Visible text in Fig. 14: "Digital flight-test platform"; actors "flight-test engineer" and "simulation engineer"; use cases "test design—technical design, parameter definition, digital prototype creation"; "formulate test plan—measurement-point layout, requirements analysis, DOE design"; "execute virtual test—mission rehearsal, fault prediction, safety assessment"; external inputs "receive upstream system input" and "bench-test data input"; relationship label "<>."

Fig. 14 Test use-case diagram (flight-test engineer use-case portion)
Fig. 14 Test use-case diagram (flight-test engineer use cases)

Visible text in Fig. 15: "Take-off phase"; "pre-take-off inspection (system status/parameter verification)"; "engine warm-up (N1 > 65%, EGT stable)"; "take-off power setting (TOGA/maximum continuous power)"; "data acquisition system startup (1 kHz sampling rate)"; "climb phase."

Fig. 15 Test activity diagram (take-off phase portion)
Fig. 15 Test activity diagram (take-off phase section)

Fig. 16 Test state diagram (engine start and run section)

Text in the diagram: system power-on; stopped state; startup request [$safetycheckpassed$]; startup preparation; fuel supply normal, ignition system ready; starting; startup failure, $N1 < 10\%$ timeout; $N1 > 15\%$, $EGT < 800\,^{\circ}\mathrm{C}$, ignition successful; warming up; $N1 > 45\%$, EGT stable, oil pressure > 40 psi; idle operation; "composite state of power operation"; acceleration request, $N1 > 65\%$; normal operation; shutdown request, $N1$ decreasing; cooling; $N1 < 5\%$, $EGT < 100\,^{\circ}\mathrm{C}$.

3) Based on the test matrix, a surrogate model was generated using a neural network, and the surrogate model was effectively validated. The model calculation results and data errors are shown in Figs. 17 and 18.

Fig. 17 Comparison plot of sample data vs. surrogate-model-generated data

Plot labels: y-axis, bleed-air flow / (kg·s^{-1}); x-axis, test data index; legend, bleed-air flow—measured/simulated value; bleed-air flow—surrogate-model calculated value.

Measured/simulated value	Surrogate-model calculated value	Error value
0.283 4	0.283 4	0
0.247 379	0.247 379	0
0.264 927	0.264 927	0
0.293 236	0.293 236	0
0.306 624	0.306 624	0
0.316 345	0.316 345	0
0.329 729	0.329 729	0
0.343 841	0.343 841	0
0.356 603	0.356 603	0
0.368 789	0.368 789	0
0.380 994	0.380 994	0
0.393 026	0.393 026	0
0.404 724	0.404 724	0

Fig. 18 Verification of the validity of the surrogate-model calculation results

The module plots the optimized matrix as a flight-test mission profile, as shown in Fig. 19.

Fig. 19 Test profile diagram

Plot labels: fuel flow rate / (kg · s^{-1}), time / s; flight altitude / km, time / s; flight speed / (m · s^{-1}), time / s.

4) The test modification optimization module was used to perform finite-element modal analysis on the measurement rake for performance flight tests, obtaining the first six natural frequencies and mode shapes. Through CFD flow-field simulation, the disturbance effects on the flow fields at the inlet duct, compressor inlet, and combustion-chamber outlet after installation of the measurement rake were evaluated, and the total-pressure distortion index, the increase in turbulence intensity, and the range of any potentially induced separation regions were calculated. The results are shown in Fig. 20.

Fig. 20 Inlet-duct exit-plane pressure distribution plot

Figure labels: pressure / Pa; 766.71, 708.86, 651.02, 593.17, 535.32, 477.47, 419.63, 361.78, 303.93, 246.08.

5) In the task simulation comparison and intelligent monitoring module, the engine-performance flight-test model is read. Using the model bus, a co-simulation framework as shown in Fig. 21 is constructed, including the engine control state diagram, engine control model, engine performance model, and engine physical model. An engine-control UI is drawn according to the state diagram for simulation control. The optimized test-matrix data are then read and the model scheme is executed to obtain the simulation results, as shown in Fig. 22.

Fig. 21 Construction of a joint simulation model for engine performance

Fig. 22 Simulation results curve plot

6) Validation of the comprehensive evaluation module for flight-test results. The module performs time alignment and cleaning of the measured data over the entire flight-test process. Based on interpolation and reconstruction algorithms, it generates a continuous four-dimensional response surface of "altitude–speed–thrust–fuel-consumption rate," enabling real-time visualization and interactive comparison at arbitrary operating points within the envelope. It automatically extracts the mean, variance, confidence interval, and out-of-limit frequency of key performance indicators, and completes multidimensional difference analysis against the baseline simulation results. The comparison between the simulation and measured results is shown in Fig. 23.

Simulation fuel flow (k⋯)	Simulation thrust (N)	Simulation bleed-air flow (kg⋯)	Test fuel flow (k⋯)	Test thrust (N)	Test bleed-air flow (kg⋯)
0.2	0	0	0.2	0	0
0.672327	269.733	0.0842215	0.672327	269.733	0.0842215
0.672986	906.744	0.283122	0.672986	906.744	0.283122
0.587449	907.633	0.2834	0.587449	907.633	0.283123
0.629118	792.272	0.247379	0.629118	792.272	0.283124
0.696345	848.47	0.264927	0.696345	848.47	0.283125
0.728136	939.137	0.293236	0.728136	939.137	0.283126
0.75122	982.012	0.306624	0.75122	982.012	0.283127
0.783005	1013.14	0.316345	0.783005	1013.14	0.283128
0.816515	1056.01	0.329729	0.816515	1056.01	0.283129
0.846822	1101.21	0.343841	0.846822	1101.21	0.283130
0.875759	1142.08	0.356603	0.875759	1142.08	0.283129
0.904743	1181.11	0.368789	0.904743	1181.11	0.283132
0.933315	1220.19	0.380994	0.933315	1220.19	0.283133

Fig. 23 Simulation data (left) vs. test data (right) comparison plot

In the comprehensive results-evaluation module, the simulation results and the real test-data results are read and compared. The data comparison and error plot are shown in Fig. 24, from which it can be seen that the error lies within the acceptable range. The calculated results for other points within the flight envelope are shown in Fig. 25, ultimately confirming that the real test data meet the requirements.

Fig. 24 Comparison and error plot of simulation vs. actual data

Time	Fuel flow	Flight altitude	Flight speed	Thrust	Bleed-air flow
1001.8	1.41381202122[illegible]	[illegible]0.60882356[illegible]	[illegible]5731622588[illegible]	[illegible]8.35220873[illegible]	[illegible]97014579763402
1003.8	1.41355654426[illegible]	[illegible]9.95563376[illegible]	[illegible].7329687638[illegible]	[illegible]8.01480632[illegible]	[illegible]76916113823822
1005.8	1.41330106730[illegible]	[illegible]9.30244396[illegible]	[illegible]5734314938[illegible]	[illegible]7.67740390[illegible]	[illegible]6817647884243
1007.8	1.41304559034[illegible]	[illegible]8.64925416[illegible]	[illegible].7356611137[illegible]	[illegible]7.34001494[illegible]	[illegible]96719181944663
1009.8	1.41279011339[illegible]	[illegible]7.99606436[illegible]	[illegible]5737007288[illegible]	[illegible]7.00259908[illegible]	[illegible]76620716005083
1011.8	1.41253463643[illegible]	[illegible]7.34287456[illegible]	[illegible].7383534635[illegible]	[illegible]6.66519666[illegible]	[illegible]6522250065503
1013.8	1.41227915947[illegible]	[illegible]6.68968476[illegible]	[illegible]573969963[illegible]	[illegible]6.32779425[illegible]	[illegible]96423784125923
1015.8	1.41202368251[illegible]	[illegible]6.03649496[illegible]	[illegible].7410458134[illegible]	[illegible]5.99039184[illegible]	[illegible]6325318186343
1017.8	1.41176820556[illegible]	[illegible]5.38330516[illegible]	[illegible]5742391988[illegible]	[illegible]5.65298942[illegible]	[illegible]6226852246763
1019.8	1.41151272860[illegible]	[illegible]4.73011535[illegible]	[illegible].7437381632[illegible]	[illegible]5.31558701[illegible]	[illegible]6128386307183
1021.8	1.41125725164[illegible]	[illegible]4.07692555[illegible]	[illegible]5745084338[illegible]	[illegible]4.97818460[illegible]	[illegible]6029920367603

Fig. 25 Linear-regression-based prediction for other time points

4 Conclusion

This study proposes a systems-engineering-based design method for a typical digital flight-test simulation application system for aircraft engines. Taking the digital flight test of aircraft-engine performance-characteristic subjects as the typical application objective, it realizes a closed loop of “requirements–architecture–model–data–evaluation” and establishes a set of aircraft-engine intelligent simulation application for full-process engine digital flight testing

system, supports model-based forward engine test design, iterative optimization of test requirements, intelligent safety monitoring, and precise quantitative evaluation, and can preliminarily meet the operational needs of digital flight testing for aero-engines.

In the future, capabilities will be further expanded toward multi-disciplinary knowledge graphs, real-time digital-twin flight testing, cross-platform FMI/SysML interoperability, and digital airworthiness evidence packages, so as to establish an integrated simulation application platform for digital flight testing of aero-engines applicable to “all subjects and all aircraft types,” and to form comprehensive digital forward-design capability, integrated analysis and evaluation capability, and full-process digital flight-test capability.

References:

[1] WANG Yingluo. *Systems Engineering* [M]. 6th ed. Beijing: China Machine Press, 2024.

[2] LIANG Di, SHAN Linting. *Fundamentals and Applications of Systems Engineering* [M]. Beijing: Tsinghua University Press, 2018.

[3] GUO Fang, LIU Hui, YANG Huiyun, et al. Analysis of differences in airworthiness management between military and civil aircraft based on systems engineering theory [J]. *Science Technology and Engineering*, 2024, 24(13): 5620–5625.

GUO Fang, LIU Hui, YANG Huiyun, et al. Analysis of the distinction in military and civil aero-engine airworthiness management based on the systems engineering [J]. *Science Technology and Engineering*, 2024, 24(13): 5620–5625 (in Chinese).

[4] WU Yunhao, DONG Yu, LU Yuanjie. Research on application methods of digital engineering strategy in aviation equipment development [J]. *Aircraft Design*, 2025, 45(2): 7–11.

WU Yunhao, DONG Yu, LU Yuanjie. Research on application method of digital engineering strategy in aviation equipment development [J]. *Aircraft Design*, 2025, 45(2): 7–11 (in Chinese).

[5] ZHAO Min, ZHOU Fanli, CHEN Liping, et al. Develop digital engineering and realize the digitalization of aerospace equipment [J]. *Missiles and Space Vehicles*, 2021(6): 106–113.

ZHAO Min, ZHOU Fanli, CHEN Liping, et al. Digitalization of spacecraft by digitization engineering [J]. *Missiles and Space Vehicles*, 2021(6): 106–113 (in Chinese).

[6] HUANG Weina, LI Fangjuan, QI Hongbin. Preliminary research and development considerations on aero-engine digital engineering [J]. *Acta Aeronautica et Astronautica Sinica*, 2024, 45(5): 128–145.

HUANG Weina, LI Fangjuan, QI Hongbin. Preliminary investigation and thoughts on aero-engine digital engineering development [J]. *Acta Aeronautica et Astronautica Sinica*, 2024, 45(5): 128–145 (in Chinese).

[7] LUO Tingting. Research on systems-engineering-based requirements management methods for commercial aero-engine development [J]. *Aeronautical Manufacturing Technology*, 2015(3): 109–111.

LUO Tingting. System engineering-based requirement management method for commercial aeroengine [J]. *Aeronautical Manufacturing Technology*, 2015(3): 109–111 (in Chinese).

[8] XI Weidi. Research on the application of DOORS in requirements confirmation for FADEC systems [J]. *Aeronautical Computing Technique*, 2018, 48(2): 86-89.

XI Weidi. Research on application of DOORS in FADEC system requirements confirmation [J]. *Aeronautical Computing Technique*, 2018, 48(2): 86-89 (in Chinese).

[9] SHI Yanyan, WANG Guihua, LIU Qingdong, et al. Research on aero-engine requirements management methods [J]. *Aeroengine*, 2017, 43(1): 91-94.

SHI Yanyan, WANG Guihua, LIU Qingdong, et al. Research on aircraft requirements management method [J]. *Aeroengine*, 2017, 43(1): 91-94 (in Chinese).

[10] LU Chuanchuan, LI Lunwei, SUN Wenbin, et al. Exploration of requirements management for aero-engine development [J]. *Gas Turbine Experiment and Research*, 2019, 32(4): 58-62.

LU Chuanchuan, LI Lunwei, SUN Wenbin, et al. Exploration of aero-engine development requirements management [J]. *Gas Turbine Experiment and Research*, 2019, 32(4): 58-62 (in Chinese).

[11] JI Yan, LI Zhimin. Application of requirements management in civil aero-engine development [J]. *Aerospace Power*, 2020(2): 36-39.

JI Yan, LI Zhimin. The application of requirement management in civil aero engine [J]. *Aerospace Power*, 2020(2): 36-39 (in Chinese).

[12] GAO Yuanyuan, QIN Xiao. Discussion on the development of aero-engine test beds based on digital technology [J]. *Internal Combustion Engine & Parts*, 2025(9): 96-98.

GAO Yuanyuan, QIN Xiao. Development of aero engine test bed based on digital technology [J]. *Internal Combustion Engine & Parts*, 2025(9): 96-98 (in Chinese).

[13] JIAO Yang. Flight testing is a systems-engineering endeavor that challenges risks and explores limits [N]. *Science and Technology Daily*, 2023-05-05(2).

[14] LIU Siyu, LIANG Yan, MA Mingming. Optimized design method for flight-test points based on discrete optimization [J]. *Acta Aerodynamica Sinica*, 2023, 41(1): 86-92.

LIU Siyu, LIANG Yan, MA Mingming. Optimal flight test design method based on discrete optimization [J]. *Acta Aerodynamica Sinica*, 2023, 41(1): 86-92 (in Chinese).

[15] PAN Pengfei. Aero-engine condition monitoring and fault diagnosis based on flight-test data [J]. *Journal of Propulsion Technology*, 2021, 42(12): 2826-2837.

PAN Pengfei. Flight data based condition monitoring and fault diagnosis of aero-engine [J]. *Journal of Propulsion Technology*, 2021, 42(12): 2826-2837 (in

Chinese).

(Editor: Li Kunlu)

Note: Figure translations are in progress. See original paper for figures.