Information Technology

Distributed Engine Control Empirical/Analytical Verification Tools

Key factors such as control system performance, reliability, weight, and bandwidth utilization can be systematically assessed.

John H. Glenn Research Center, Cleveland, Ohio

NASA's vision for an intelligent engine will be realized with the development of a truly distributed control system featuring highly reliable, modular, and dependable components capable of both surviving the harsh engine operating environment and decentralized functionality. A set of control system verification tools was developed and applied to a C-MAPSS40K engine model, and metrics were established to assess the stability and performance of these control systems on the same platform. A software tool was developed that allows designers to assemble easily a distributed control system in software and immediately assess the overall impacts of the system on the target (simulated) platform, allowing control system designers to converge rapidly on acceptable architectures with consideration to all required hardware elements. The software developed in this program will be installed on a distributed hardware-in-the-loop (DHIL) simulation tool to assist NASA and the Distributed Engine Control Working Group (DECWG) in integrating DCS (distributed engine control systems) components onto existing and next-generation engines.

The distributed engine control simulator blockset for MATLAB/Simulink and hardware simulator provides the capability to simulate "virtual" subcomponents, as well as swap actual subcomponents for hardware-in-the-loop (HIL) analysis. Subcomponents can be the communication network, smart sensor or actuator nodes, or a centralized control system. The distributed engine control blockset for MATLAB/Simulink is a software development tool. The software includes an engine simulation, a communication network simulation, control algorithms, and analysis algorithms set up in a modular environment for rapid simulation of different network architectures; the hardware consists of an embedded device running parts of the C-MAPSS engine simulator and controlled through Simulink.

The distributed engine control simulation, evaluation, and analysis technology provides unique capabilities to study the effects of a given change to the control system in the context of the distributed paradigm. The simulation tool can support treatment of all components within the control system, both "virtual" and real; these include communication data network, smart sensor and actuator nodes, centralized control system (FADEC — full authority digital engine control), and the aircraft engine itself. The DECsim tool can allow simulation-based prototyping of control laws, control architectures, and decentralization strategies before hardware is integrated into the system. With the configuration specified, the simulator allows a variety of key factors to be systematically assessed. Such factors include control system performance, reliability, weight, and bandwidth utilization. The ability to provide a configurable, high-fidelity distributed engine control simulation, control system analysis, and HIL evaluation is a unique capability of the technology.

This work was done by Jonathan DeCastro and Eric Hettler of Impact Technologies, LLC; Rama Yedavalli of the Ohio State University; and Sayan Mitra of the University of Illinois at Urbana for Glenn Research Center. Further information is contained in a TSP (see page 1).

Inquiries concerning rights for the commercial use of this invention should be addressed to NASA Glenn Research Center, Innovative Partnerships Office, Attn: Steven Fedor, Mail Stop 4–8, 21000 Brookpark Road, Cleveland, Ohio 44135. Refer to LEW-18829-1.

Dynamic Server-Based KML Code Generator Method for Level-of-Detail Traversal of Geospatial Data

Innovation uses a C- or PHP-code-like grammar that provides a high degree of processing flexibility.

Stennis Space Center, Mississippi

Geospatial data servers that support Web-based geospatial client applications such as Google Earth and NASA World Wind must listen to data requests, access appropriate stored data, and compile a data response to the requesting client application. This process occurs repeatedly to support multiple client requests and application instances. Newer Web-based geospatial clients also provide user-interactive functionality that is dependent on fast and efficient server responses. With massively large datasets, server-client interaction can become severely impeded because the server must determine the best way to assemble data to meet the client applications request. In client applications such as Google Earth, the user interactively wanders through the data using visually guided panning and zooming actions. With these actions, the client application is continually issuing data requests to the server without knowledge of the server's data structure or extraction/assembly paradigm.

A method for efficiently controlling the networked access of a Web-based