Design System for Managing Complexity in Aerospace Systems

Sandor Becz[1]
Lenze SE, 630 Douglas Street, Uxbridge MA 01569

Alessandro Pinto[2], Lawrence E. Zeidner[3], Ritesh Khire[4], Andrzej Banaszuk[5], Hayden M. Reeve[6]
United Technologies Research Center, East Hartford, CT, 06108

A new generation of aerospace systems has introduced more-electric technologies and highly coupled architectures, leading to a substantial increase in design complexity. The inability of current design systems to manage this complexity has lead to substantial cost and schedule overruns during the development of new military and commercial platforms. This paper presents the framework for a new design process to manage system complexity and highlights four key elements: 1) Abstraction-based design tools to allow complex interactions to be assessed at a high level of abstraction early in the design cycle; 2) Quantitative complexity metrics that can guide early design decisions and identify sources of complexity within candidate designs; 3) Advanced architecture synthesis methods that enable the formal and automated architecture synthesis, enumeration, and evaluation of feasible architecture options; and 4) Robust uncertainty management for identifying strong coupling between systems with high uncertainty, thereby allowing the identification and management of key risks.

I. Introduction

The past several decades have seen the introduction of significant technological and architectural changes in aerospace systems in an effort to improve the performance and capability of new platforms. For example, the 787 and F-35 have incorporated more-electric systems for functions such as cabin pressurization and flight control actuation, respectively. The push to lower maintenance cost and increased dispatch reliability has lead to the adoption of prognostics and health management (PHM) systems. The implementation of fly-by-wire for flight control has reduced weight and pilot work load on many platforms. However, these and other driving forces have lead to an exponential growth in the complexity of modern aerospace platforms and accompanying design and development challenges.

The fifth generation F-35 tactical fighter offers a good example of the increased capability these new systems provide and also the resulting challenges. The F-35 is the only fighter capable of transitioning from vertical flight to supersonic cruise. It offers superior survivability through the incorporation of features such as internal stores and a composite airframe. In short, it offers 3-8 times the operational capability of fourth generation aircraft such as the F-16 or F-18 while providing superior range. This capability, however, has come at the cost of increased technical complexity. For example, the F-35 has 130 subsystems, order 10^5 interfaces, and 90 percent of its functions managed by software[1]. This is a substantial growth from the F-16 that has 15 subsystems, order 10^3 interfaces, and less than 40 percent of its functions managed by software[1]. Greater electrical loads from avionics, power electronics, and airframe actuation have increased power management requirements and more closely coupled systems together. These increased avionics and electrical loads, when combined with increased engine and

Distribution Statement "A" (Approved for Public Release, Distribution Unlimited)
[1] Director of Engineering, AIAA Member.
[2] Senior Engineer, Embedded Systems & Networks, AIAA Member.
[3] Staff Engineer, Thermal Management, AIAA Member.
[4] Senior Engineer, System Design & Integration, AIAA Member.
[5] Fellow, Systems Department, AIAA Member.
[6] Staff Engineer, Thermal Management, AIAA Senior Member.

actuation heat loads have increased the demands on the aircraft thermal management system while the thermal constraints of a composite airframe and limited ram cooling make it harder to get heat off the aircraft.

These increases in technical complexity have been accompanied by increases in complexity of system requirements and organizational partnerships. Complexity in requirements stems from the need to meet multiple present and future mission objectives. The F-35 design meets multi-role objectives through three different variants: Short Take Off and Vertical Landing (STOVL), Carrier Variant (CV), and Conventional Take Off and Landing (CTOL) operations (Figure 1). This requires that the centerbody of the F-35 be designed to meet a wide range of requirements. There is also increasing complexity in the development team. The F-35 is being funded by nine partner nations and being developed by a broad multi-national team. Increasing complexity in requirements and development teams is not solely a challenge for government-funded development programs. The 787 is a good example of a multi-national development team and supply chain and the challenges associated with this.

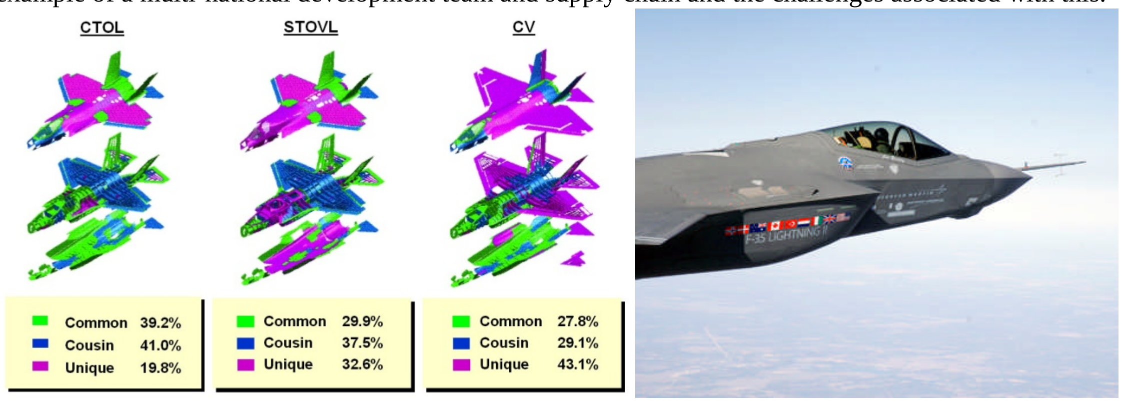

Figure 1: Illustration of the three F-35 variants (left) and F-35 in flight

These tightly coupled aircraft subsystems often have very different length and time scales and the dynamic nature of their interactions, coupled with their inherent uncertainty, drive system behavior that is often difficult to predict during conceptual and preliminary design stages. This results in "emergent behavior" or functionality that is not intended but realized during validation and verification of the system. For example, the F-35 has required redesign of its 270 VDC electrical system following issues identified during flight testing[2] and its thermal management system, which meets requirements, may need to be redesigned to provide more thermal margin[3]. The A400M provides further example of complexity issues, in this case stemming from development and integration of the Full Authority Digital Engine Control (FADEC)[4]. The complexity of new aerospace systems has contributed to unexpected development costs and delays on both military programs (F-35 delayed 1-3 years, A400M delayed 3 years) and commercial programs (787 delayed 28 months, A380 delayed 18 months).

Analysis conducted by the US Government Accountability Office (GAO) of major defense acquisition programs found that research and development costs are on average 42% higher than originally estimated and that the average delay in delivering initial capability to the war-fighter is 22 months[5]. Analysis by the RAND Corporation found that the largest component in the growth in the cost of fixed-wing aircraft has come from increased complexity[1]. As systems have become more complex they have become not only more expensive to develop, but the ability to predict that development cost is poor. Managing and minimizing the complexity of new system development offers the ability to reduce both the magnitude and unpredictability of development cost. The GAO found that development programs that had more knowledge earlier in the development cycle incurred reduced cost overruns. Specifically: Programs that start development with fully mature critical technologies experienced 30% less R&D cost growth; Programs that held system engineering reviews (requirements review, functional review, or preliminary design review) prior to development start experienced 20% less cost growth; Finally, programs that had no changes in key performance parameter requirements had one third the cost growth of other development programs.

To meet the challenge of developing future complex systems in a cost-effective and predictable manner a new generation of design processes and tools are required that manage complexity at multiple levels. In many aerospace subsystems the design architecture and technologies have not changed for decades. The arrival of new architectures and technologies has not been accompanied by a commensurate advancement and adoption of the design processes and tools needed to develop these very complex systems. The next section presents a new design paradigm and discusses key needs and potential tools that must be created to address these challenges. The unifying theme of these tools is to attain knowledge earlier in the design stage in order to better identify the scale, impact, and behavior of system interactions early in the design process in order to ensure complexity is understood before key design decisions and development investments are made.

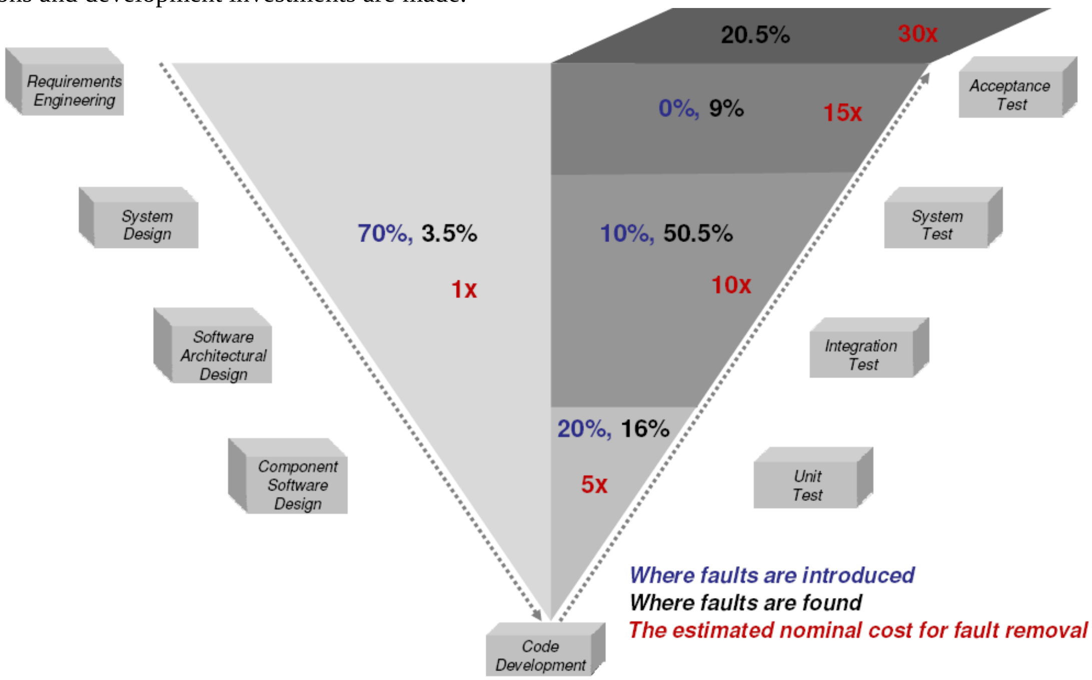

Figure 2: The traditional design 'V' showing an estimation of the introduction, detection, and cost of removal of faults during software development

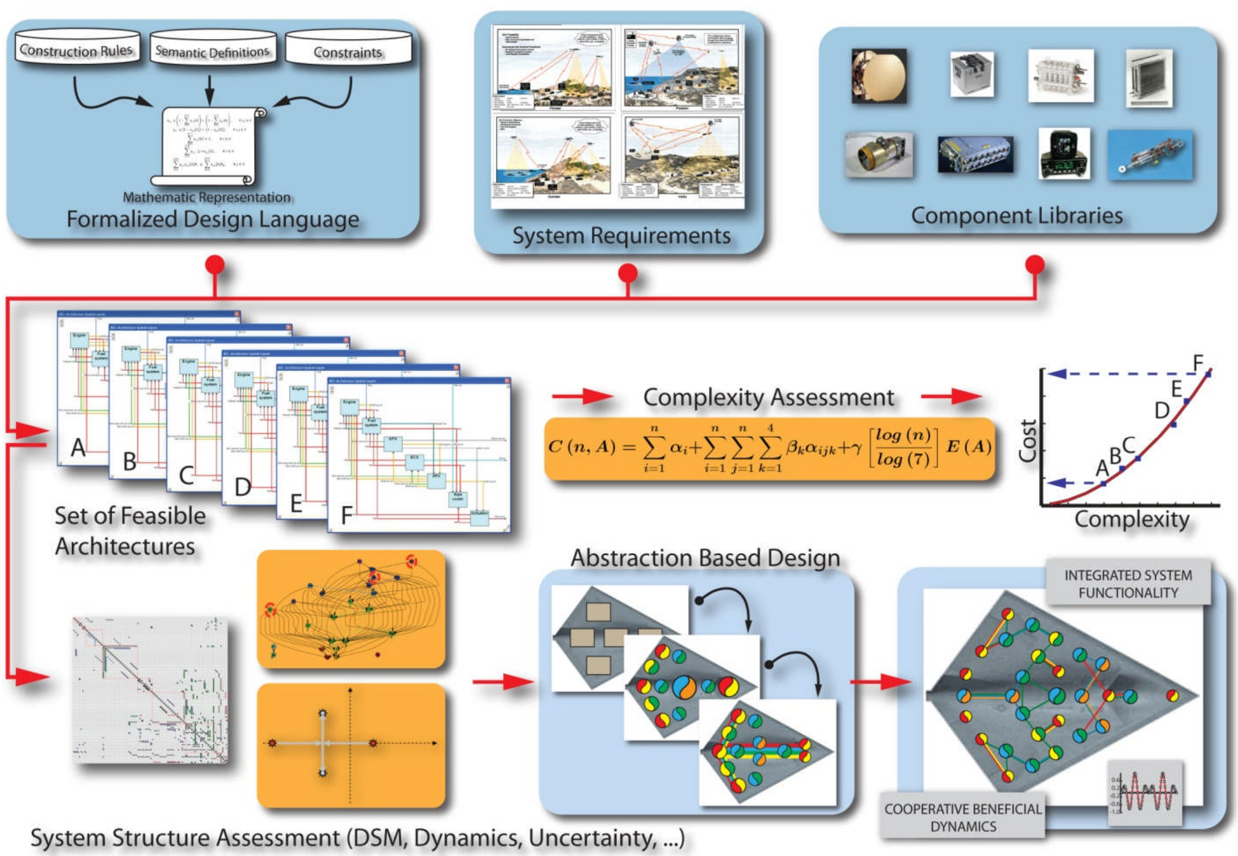

Figure 3: Proposed design process for complex systems

II. Overview of Design Process for Complex Systems

The design approach employed within the defense industry today is one of sequential flows of information separated along functional lines. For example, power system design activity flows from top level requirements which exist separately from hydraulic system requirements, and so on with each major subsystem. Within each system domain, the architectural layout of components is typically performed first with the overall functionality and performance determined through analysis of the selected architecture. Once the primary components have been assembled the supporting systems such as controls and communication are then developed around this layout. This approach of subdivision and refinement into more detailed design representations results in a 'design V'. This paradigm has been applied widely in many industries to organize the development process and in many applications the sequential development and refinement from concept to preliminary design to detailed design is functional. For complex systems with a great deal of interaction this compartmentalized design process leads to significant amounts of rework late in the design phase due to issues related to performance shortfalls as well as unpredicted emergent behavior resulting from various system interactions. The cost of the late detection of faults during software development has been characterized by Bruce Lewis of Army ARMDEC using data from a NIST analysis of the economic impact of inadequate infrastructure for software testing[6] (Figure 2). This shows a characteristic common to many development programs: most errors are introduced into the design early but not caught until late in the program resulting in far greater cost to fix. Therefore, the ability of subsystem suppliers to pass requirements and static boundary conditions "over the wall" to other subsystem suppliers is no longer acceptable. Decisions made in isolation respective to one system (power distribution, for example) have repercussions on all other systems due to emergent behavior.

Given these limitations inherent in today's design processes, a new design approach is required that incorporates not only individual subsystem functional performance, but also all of the dynamic interactions between these systems. In addition, control and communication must be considered early in conceptual design as primary aspects of equal importance to meeting platform performance. To maintain tractability, abstraction must be introduced into the design flow to expose only those elements relevant to various stages of design, but without sacrificing the link between design choices and system performance requirements. Synthesis techniques that automatically explore the design space in search of architectures with improved performance must also be included due to the limitations of humans to manually search the large number of configurations possible.

A candidate design system is shown in Figure 3. System requirements and component libraries (representing the existing knowledge of constitutive parts of potential solutions) are formally represented in a formal design language such as AADL, UML, and SysML. This information feeds a design process that has four key elements:

1. **Abstraction Based Design Tools:** Provides a design and evaluation framework that can model complex heterogeneous systems. Enables abstraction of system models to allow complex interactions (e.g., controls and communications) to be assessed at a high level of abstraction early in the design cycle. Incorporates verification and 'correct by construction' elements.
2. **Quantitative Complexity Metrics:** Quantification of complexity to provide a metric that can guide early design decisions and identify sources of complexity within candidate architectures. Complexity quantification needs to be abstracted to enable its use from early conceptual design through product development.
3. **Advanced Architecture Synthesis Methods:** An advanced set of tools that enables the formal and automated architecture synthesis, enumeration, and evaluation of all feasible architecture options, and decomposition and clustering of architectural elements in order to minimize complexity propagation in the system.
4. **Robust Uncertainty Management:** Advanced tools to access the interplay between complexity and uncertainty. Enabler in identifying strong coupling between systems with high uncertainty, thereby allowing the identification and management of key risks.

The following sections address each area in more detail, discussing the key requirements of these approaches and novel elements.

A. Abstraction Based Design

To meet the needs of future platform developers, abstraction of the design space is necessary to allow early conceptual efforts to progress quickly without sacrificing the ability to determine if performance requirements are satisfied. For example, as architectures become increasingly distributed and heterogeneous, reliability and fault tolerance become primary design factors that can no longer be relegated to a "post-assessment" once systems design is complete. Therefore, a means of quickly decomposing systems and analyzing their interconnections and the effect of cascading failures must be developed to facilitate the exploration of extremely large design spaces.

The methodology known as Platform Based Design (PBD)[7] is a key intellectual framework for product development in situations where there is a great heterogeneity of subsystems – different physical, computational and communication subsystems and choices for architectural implementation. PBD also addresses situations where there is large scale difference both in state and physical distribution as well as the situation where the subsystems are integrated such that there is significant interaction that must be recognized and exploited in order to meet performance and cost targets.

There are two key principles behind PBD that address issues in the development of complex systems for defense applications. PBD addresses complexity by introducing into the design process layers of abstraction that provide appropriate fidelity for effective design decision making while bringing forward implementation constraints into the design early in the overall process thus enabling early verification. PBD also separates out the specification of the functionality and the architecture and at each layer of abstraction maps the required functionality to a chosen architecture and then refines the choices at each layer. This separation enables the effective trade study and capture of requirements versus cost and performance and the reduction of overall complexity.

PBD methodology was successfully adopted by several automotive manufacturers in Europe for managing development time and cost as this design process enables management of increased complexity, enables software reuse, and reduces the verification and validation effort. However, there is significant investment needed to extend Platform Based Design to design of military aerospace systems because of the additional system complexity, performance and safety requirements well beyond and above those of the commercial car industry.

Effective verification of system level performance is key to delivering high performance and cost effective solutions to the defense contractor base. PBD is a framework for the effective deployment of tools for this verification to be done either through a "correct by construction" approach or through the deployment of verification tools throughout the design process. The key enabling technology is twofold: first to insert appropriate verification in the abstraction layers early in the design process and second to deploy robust design tools to quantify and mitigate the effects of uncertainty.

The creation of abstraction layers as defined in the PBD methodology is a key to controlling complexity and producing a truly scalable design methodology. At each layer of the process design space exploration is used to create a rich set of alternative architectures to meet required functionality, moreover, it is critical to verify the design with constraints that will be seen at lower levels so that "large loop" re-designs are avoided. PBD thus offers the framework for effective design. An example of how PBD might be applied to the architecture synthesis of an aerospace electrical system is provided in Reference [8].

B. Characterization and Quantification of Complexity

A key element of the proposed design system is the ability to quantify the complexity of advanced systems throughout the design process. This requires both abstract (i.e., low fidelity) complexity metrics to serve as a leading indicator of complexity for use early in the process during configuration selection, cost estimation, and bidding and proposing, all the way to detailed complexity metrics for use during detailed design. There has been considerable work defining complexity but less effort on constructing a quantitative metric that can guide design decisions. Kim and Wilecon[9] provide a review of complexity definitions that have been developed. These definitions cover the range of project, product, R&D/innovation, integration, and market areas and the definitions include the following core elements:

- *Numbers:* Number of different disciplines or departments involved. Number of parts, technologies, or functions required in a product.
- *Degree of Interdependency:* Level of interdependency among the domains, functions, or disciplines involved.
- *Intricacy or difficulty:* Novelty of project (minor modifications and growth and derivative versions versus clean sheet designs with untested technologies).
- *Limitations:* A compounding factor that can increase the complexity in the areas above. Examples include: limited time to market, tight performance requirements (weight, thrust), stringent constraints (thermodynamic limitations).

Other definitions have been used in the aerospace field. The Air Force Research Laboratory's (AFRL) INtegrated Vehicle ENergy Technology (INVENT) program[10] views complexity as equivalent to the inflexibility of a design to meet future growth requirements. That is, how tightly integrated the design space is with respect to change. Arena et al.[1] used the term complexity loosely to refer to the increased capability of aircraft. Jones et al.[11] developed a quantitative metric to estimate the cost of large scale systems by developing a metric based on the number of nodes and links within a system. To effectively manage complexity in the future, domain specific standard measures of complexity are needed that would allow competing offerings to be ranked, similar to the capability-to-cost index (CCI) used to quantify the goals for future gas turbine engine performance. More importantly, identifying the key attributes and contributing factors that create or amplify complexity (and therefore development cost and risk) is a key requirement to being able to manage and minimize complexity. United Technologies Research Center (UTRC) has started to explore different methods and constructs to quantify complexity for aerospace systems[12]. This work investigates candidate system attributes that could be used quantify complexity.

C. Advanced Architecture Synthesis Methods

Advanced architecture synthesis methods are required to meet three emerging challenges. The first challenge stems from next generation systems becoming more complex and multi-disciplinary, resulting in the design process no longer being able to rely on the intuitive expertise of a small number of designers to make the initial down-selections to create a design space that is tractable for current methods. A formal process is required to synthesize architectures to ensure all known requirements and interactions are acceptable. Furthermore, the rate of technological advancement and complexity of these systems has increased the design configuration trade space. UTRC has developed the Architectural Enumeration and Evaluation (AEE) framework to enable the efficient and traceable decision making that rapidly reduces the entire design space to regions that warrant further investigation[13,14]. AEE provides a rigorous, efficient, and exhaustive means to explore a very large number of technology and architecture configurations for new application areas. AEE can tackle trade spaces that have millions or billions of design configuration possibilities and efficiently consider all possibilities to identify the set of feasible configurations (typically numbering in the thousands) and the set of promising concepts that are worthy of higher fidelity investigation (typically 10-100).

The second challenge involves superior evaluation of architecture options early in the design cycle. This will be achieved by the evaluation of both PBD domain models and a complexity metric (at appropriate levels of abstraction) during architecture evaluation and selection.

The third challenge is to understand how the architectures can be partitioned into sub-domains (so organizational entities can design and develop them) in a way that minimizes the propagation of complexity between the sub-domains and therefore minimizes likely development risk and cost. The complexity of each architecture depends not only upon its constituent technologies and its interconnections, but also on how and to what degree the architecture is organized hierarchically into modules. Each architecture can be decomposed into many different hierarchical configurations of modules, each with its own degree of complexity. UTRC has explored a method that uses a spectral graph partitioning algorithm recursively to determine the hierarchy of modules based on the limited information available at this early stage in the design process[15].

D. Methods to assess the impact of uncertainty throughout the design process

Introduction of immature or new technologies introduces significant uncertainties during the development of complex systems. This uncertainty is manifested as the lack of accurate characterization of the subsystems during early design and selection phase. This is in addition to other well known sources of uncertainties such as environmental conditions and evolving system requirements. With the variability and uncertainty associated with parameters in complex systems, a formal treatment of their impact on emergent behavior must be included in any new design paradigm. This component is virtually non-existent in current design systems, especially in the early phases of design.

UTRC has assessed the impact of uncertainties on complex system selection[16]. This work demonstrated, through numerical simulation, that selection of good system architecture is critical to minimize the vulnerability of complex system to above mentioned uncertainties. In other words, the selection of apt complex system architecture will allow it to fulfill all the functional requirements. At the same time, the system will be robust against uncertainties, potentially resulting in minimum development time and resource investment.

III. Concluding Remarks

Some of the design system elements presented above, such as Platform Based Design, have been used extensively in other industries to great effect. Others, such as complexity quantification, have not yet been integrated into a design system. Regardless, substantial effort and investment is required to further adapt, pilot, and integrate these elements into the aerospace sector design chain. This will require substantial continuing commitment, contribution, and collaboration from academia, industry, and the government.

Acknowledgments & Disclaimers

The development of the methods and results contained in this work [8, 12, 14-16] was sponsored under the DARPA contract: “Abstraction Based Complexity Management” #FA9550-07-C-0024.

The views, opinions, and/or findings contained in this article/presentation are those of the author/presenter and should not be interpreted as representing the official views or policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the Department of Defense.

References

[1]Arena, M. V., Younossi, O., Brancato, K., Blickstein, I., Grammich, C. A., “Why Has the Cost of Fixed-Wing Aircraft Risen?” RAND Corporation, 2008, http://www.rc.rand.org/pubs/monographs/2008/RAND_MG696.pdf

[2]Boeder, J., “F-35 JSF Hit by Serious Design Problems”, Daily Industry Daily, December 2007, http://www.defenseindustrydaily.com/f-35-jsf-hit-by-serious-design-problems-04311/ [cited 27 January 2010]

[3]Perrett, B., “F-35 May Need Thermal Management Changes”, Aviation Week, March 12 2009, http://www.aviationweek.com/aw/generic/story.jsp?id=news/F35-031209.xml&headline=F-35%20May%20Need%20Thermal%20Management%20Changes&channel=defense [cited 27 January 2010]

[4]http://www.independent.co.uk/news/business/analysis-and-features/the-euro20bn-plane-that-may-not-fly-1866040.html [cited 27 January 2010]

[5]http://www.gao.gov/new.items/d09326sp.pdf [cited 27 January 2010]

[6]NIST Planning Report 02-3, “The Economic Impacts of Inadequate Infrastructure for Software Testing”, May 2002. http://www.nist.gov/director/prog-ofc/report02-3.pdf

[7]Sangiovanni-Vincentelli, A., “Quo Vadis SLD: Reasoning about Trends and Challenges of System-Level Design”. *Proceedings of the IEEE*, Vol. 95, No. 3, pp. 467-506, 2007.

[8]Pinto, A., Becz, S., Reeve, H. M., “Correct-by-Construction Design of Aircraft Electric Power Systems”, To be presented at 10th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, Fort Worth Texas, 2010.

[9]Kim, J. and Wilemon, D. “An empirical investigation of complexity and its management in new product development”, Technology Analysis & Strategic Management, Vol. 21, Issue 4, 2009, pp. 547-564.

[10]Walters, E. A., and Iden, S., “INVENT Modeling, Simulation, Analysis and Optimization”, AIAA Aerospace Sciences Meeting, Orlando, FL, 2010-287.

[11]Jones, R., Hardin, P., and Irvine, A., “Simple Parametric Model for Estimating Development (RDT&E) Cost”, 2009 ISPA/SCEA Joint Conference.

[12]Zeidner, L. E., Becz, S., Khire, R., Reeve, H. M., “Design Issues for a Bottom-Up Complexity Metric Applied to Hierarchical Systems “,To be presented at 10th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, Fort Worth Texas, 2010.

[13]Zeidner, L.E., St. Rock, B.E., Desai, N.A., Reeve, H.M., and Strauss, M.P., “Application of a Technology Screening Methodology for Rotorcraft Alternative Power Systems,” *AIAA Aerospace Sciences Meeting*, AIAA-2010-1505, Orlando, FL 2010.

[14]Zeidner, L. E., Becz, S., Reeve, H. M., and Khire, R., “Architectural Enumeration & Evaluation for Identification of Low-Complexity Systems”, To be presented at 10th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, Fort Worth Texas, 2010.

[15]Zeidner, L. E., Becz, S., and Banaszuk, A., “System Complexity Reduction via Spectral Graph Partitioning to Identify Hierarchical Modular Clusters”, To be presented at 10th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, Fort Worth Texas, 2010.

[16]Khire, R., Becz, S., Reeve, H. M., Zeidner, L. E., “Assessing performance uncertainty in complex hybrid systems”, To be presented at 10th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, Fort Worth Texas, 2010.