

Cost-Benefit Analysis of Error Reduction for Complex Systems

Robert A. McDonald*

California Polytechnic State University, San Luis Obispo, CA, 93401

The cost-benefit tradeoff of analysis fidelity in complex systems analysis has been posed as an optimization problem providing quantitative guidance for investment in improving analysis. A nonlinear constrained optimizer to solve this problem has been integrated into an error tradeoff environment providing an intuitive tool for the decision maker to consider investment in fidelity at the same time as design decisions are considered. An example demonstrates the efficacy of the improved interface to enable fidelity improvement investment decisions.

I. Introduction & Motivation

IN the early stages of conceptual design, the decision maker is faced with myriad decisions. Foremost in any designer's mind are the design decisions – which concept, how big, what shape, what material, etc. The designer must also consider requirements decisions – how fast, how far, how capable, how much, how often, etc. In addition, the designer must choose design methods to approach the problem – trade studies, intuition, optimization, synthesis, etc. Finally, the designer must make decisions about the models and assumptions used in the analysis which informs the other decisions – empirical, linearized, inviscid, high fidelity, etc.

This last group may be categorized as fidelity decisions. One may initially desire to use the best methods possible to minimize error. However, the designer is faced with deadlines and must use iteration in the design process. Limited resources and the need for repeated analysis mean that the designer must always be satisfied with “good enough” tools.

Designers have long relied on simplified methods in conceptual design; choice in these methods has typically been guided by experience. The complex and unconventional systems being considered today force us to reconsider those choices. Much interest and effort has been focused on bringing higher fidelity and physics-based tools earlier into the design process. Although generally agreed upon, this necessary effort has been largely unguided by any quantitative metrics.

The need for higher fidelity, physics-based analysis tools in conceptual design is not just about tomorrow's advanced systems. The challenges faced by the aerospace industry today – and the solutions being pursued to meet those challenges – demand new and improved design oriented analyses. For example, the extensive use of composite materials on an otherwise conventional aircraft requires new approaches to structural layout and corresponding new weight estimating relationships. Unmanned aircraft with missions measured in months instead of hours require a fundamentally new approach to aircraft mission analysis and sizing. The packaging and layout of today's high performance aircraft are now driven by consideration of power, heat, and electromagnetic interference.

The early stages of design are also a time when the decision maker must be extremely concerned with risk. The decision maker should understand the certainty that the candidate design will meet the customer's requirements. One aspect of this risk is created by the error in the analysis tools used in conceptual design. The designer should know which analyses introduce the most significant error in the model and they should understand the impact and cost of reducing these errors.

The reduction of error must be considered strategically as well as tactically. The aerospace industry has historically pursued bold, long-term projects which required significant investment in improved analysis capability. At times, these projects have been the primary focus of large portions of major companies, government research centers, and university researchers. The investment of research and development effort should be guided both by the potential performance gains of technologies, but also by the risk reduction potential of improved analysis.

In previous work,¹⁻³ the cost-benefit tradeoff of error in complex systems was implied. A fidelity trade environment was developed through which the decision maker could intuitively investigate the role of error in a particular complex system. Through what-if games, decisions about the management and direction of a project could be made.

*Lockheed Martin Assistant Professor, Aerospace Engineering, One Grand Avenue, AIAA Member.

The present research is directed to make the cost-benefit tradeoff of error in complex systems design explicit and to quantify it directly. Furthermore, the intuitive decision making environment will be enhanced with tools to provide specific guidance to the optimal programmatic approach to meet a project's goals with respect to error.

II. Cost-Benefit of Error Reduction

It is impossible to fully validate a systems analysis or a design process. Many systems are too large, complex, and expensive to build test and validation articles. Furthermore, the operating conditions throughout the life cycle of a system are impossible to predict and control for a validation experiment. Sometimes designers are interested in revolutionary systems for which there is no historical counterpart which can be used for validation.

Error is introduced at every point in a complex systems design process. Every error source propagates through a complex system in the same way information propagates. If a system has feedforward, the errors feed forward. If a system has feedback, then errors feed back. If a system has coupled loops, then errors are coupled.

As with error propagation through a single analysis, error sources grow and decay when propagated through a complex system. These behaviors are made more complex by the complex interactions of a complete system. This complication and the loss of intuition that accompanies it makes proper error propagation calculations even more important to aid the decision maker.

Error can not be eliminated. However, it is possible to reduce many sources of error present in a complex systems analysis: better physical assumptions, fewer simplifications, better numerical implementations, and improved meta-models are all among the ways to reduce error at its source. Each error reduction has an associated cost while the complex system is varyingly sensitive to every error source. These relationships enforce a complex tradeoff between error sources and the cost-benefit of minimizing each error source.

III. Error Propagation for Complex Systems

In addition to error inherent to a calculation, in any calculation the error in any input quantity contributes to an erroneous output. The magnitude of error in an output may be diminished or amplified relative to the error of the inputs. In a series of calculations representing a complex system, error can build up and interact in non-intuitive ways. The behavior of error becomes even more obscure when the series of calculations are coupled.

Error propagation is a set of techniques used to quantify and understand the error in an output quantity based on some knowledge of the error of the input quantities and error introduced during the process. Error propagation was originally developed for the experimental sciences.^{4,5} In an experiment, the source of all error is measurement, as Mother Nature makes no mistakes. Conversely, computer analysis codes have many sources of error. They are subject to error in their inputs and their process but there is no measurement error in their outputs.^{6,7}

A sensitivity approach was used to rapidly approximate the propagation of error through complex systems. The error propagation formulation is that derived and presented by the author in previous works.^{1,2}

This approach relies on system sensitivity analysis (SSA)⁸⁻¹⁰ to model the behavior of a complex system as a whole. The system sensitivity matrix proposed by Sobieski contains the partial derivatives which govern the interaction of tasks within a complex system. The notation used in the development of SSA has evolved; this work most closely matches the later style.¹⁰ The form of the system sensitivity matrix (\mathbf{M}) used in this work is given below.

$$\mathbf{M} = \begin{bmatrix} \mathbf{I}_1 & -\frac{\partial \mathbf{g}_1}{\partial \mathbf{g}_2} & \cdots & -\frac{\partial \mathbf{g}_1}{\partial \mathbf{g}_l} \\ -\frac{\partial \mathbf{g}_2}{\partial \mathbf{g}_1} & \mathbf{I}_2 & & \vdots \\ \vdots & & \ddots & -\frac{\partial \mathbf{g}_{l-1}}{\partial \mathbf{g}_l} \\ -\frac{\partial \mathbf{g}_l}{\partial \mathbf{g}_1} & \cdots & -\frac{\partial \mathbf{g}_l}{\partial \mathbf{g}_{l-1}} & \mathbf{I}_l \end{bmatrix} \quad (1)$$

The propagation of random error sources through a complex system is governed by Equation 2 given below. This equation first combines the input error sources (σ_x) and the task local error sources (ζ_f). It then uses the inverse system sensitivity matrix (\mathbf{M}^{-1}) to map these errors into the propagated global error (σ_f). This equation is approximate only in that it represents a linearization for the case of finite error sources. Note that the square operations in Equation 2 are performed on a term by term basis. Although the notation and approach to error propagation is different from that of Du and Chen,¹¹ the resulting equations are equivalent.

$$\sigma_{\mathbf{f}}^2 \approx (\mathbf{M}^{-1})^2 \left[\zeta_{\mathbf{f}}^2 + \left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right)^2 \sigma_{\mathbf{x}}^2 \right] \quad (2)$$

IV. Modeling Cost-Benefit of Error Reduction for Complex Systems

The cost-benefit tradeoff of error reduction in a complex systems analysis may be stated as an optimization problem: Choose a combination of error reductions which meet a set of system error goals at minimum cost.

In this optimization problem, the solution space is the set of error reductions. Error sources may be inherent to a calculation ($\zeta_{\mathbf{f}}$) or there may be error in any input quantity ($\sigma_{\mathbf{x}}$). Reductions in either type of error source may be pursued, thus providing the problem variables $\Delta\zeta_{\mathbf{f}} \equiv \zeta_{\mathbf{f}}^* - \zeta_{\mathbf{f}}$ and $\Delta\sigma_{\mathbf{x}} \equiv \sigma_{\mathbf{x}}^* - \sigma_{\mathbf{x}}$. Where $\zeta_{\mathbf{f}}^*$ and $\sigma_{\mathbf{x}}^*$ represent the current state of the art error levels.

The total cost is a function of the desired error reductions $\$ (\Delta\zeta_{\mathbf{f}}, \Delta\sigma_{\mathbf{x}})$. The form of the objective function will be discussed in Section V.

The system error goals provide inequality constraints for the optimization problem. They are governed by the propagation of error through a complex system i.e. Equation 2. Of course, error can not be completely eliminated and there may be some maximum reduction which is possible for each error source. Likewise, we do not wish to allow an increase in any source of error which may lead to an unrealizable cost savings (or even profit). These conditions provide bounds constraints for the problem.

The system level error goals may be stated as $\bar{\sigma}_{\mathbf{f}}$. And the caps limiting the improvement of any individual error source may be stated as $\bar{\Delta\zeta}_{\mathbf{f}}$ and $\bar{\Delta\sigma}_{\mathbf{x}}$. Using these definitions, the optimal cost-benefit problem governing error improvement in a complex system may be stated in standard optimization form as in Equation 3.

$$\begin{aligned} \text{Choose: } \mathbf{A} &= (\Delta\zeta_{\mathbf{f}}, \Delta\sigma_{\mathbf{x}}) \\ \text{To minimize: } \$_{\mathbf{A}} &= \$_{\mathbf{A}}(\mathbf{A}) \\ \text{Subject to: } \mathbf{c}_{\mathbf{A}1} &= \sigma_{\mathbf{f}} - \bar{\sigma}_{\mathbf{f}} \leq 0 \\ \mathbf{c}_{\mathbf{A}2} &= -\Delta\zeta_{\mathbf{f}} \leq 0 \\ \mathbf{c}_{\mathbf{A}3} &= \Delta\zeta_{\mathbf{f}} - \bar{\Delta\zeta}_{\mathbf{f}} \leq 0 \\ \mathbf{c}_{\mathbf{A}4} &= -\Delta\sigma_{\mathbf{x}} \leq 0 \\ \mathbf{c}_{\mathbf{A}5} &= \Delta\sigma_{\mathbf{x}} - \bar{\Delta\sigma}_{\mathbf{x}} \leq 0 \end{aligned} \quad (3)$$

Figure 1(a) depicts a notional cost-benefit tradeoff solution space for error improvements. Two error sources ($\sigma_{\mathbf{x}1}$ and $\sigma_{\mathbf{x}2}$) are being considered in this example. The current state of the art is marked with an '*'. The black and green constraints represent the maximum attainable improvement possible in $\sigma_{\mathbf{x}1}$ and $\sigma_{\mathbf{x}2}$, respectively. The red and blue constraints represent two propagated system level error goals.

Although the objective function of Equation 3 is arbitrary, the constraints are all linear in $\zeta_{\mathbf{f}}^2$ and $\sigma_{\mathbf{x}}^2$. Consequently, it may be more convenient to state the problem in an alternative B form as in Equation 4.

$$\begin{aligned} \text{Choose: } \mathbf{B} &= (\zeta_{\mathbf{f}}^2, \sigma_{\mathbf{x}}^2) \\ \text{To minimize: } \$_{\mathbf{B}} &= \$_{\mathbf{B}}(\mathbf{B}) \\ \text{Subject to: } \mathbf{c}_{\mathbf{B}1} &= \sigma_{\mathbf{f}}^2 - \bar{\sigma}_{\mathbf{f}}^2 \leq 0 \\ \mathbf{c}_{\mathbf{B}2} &= \zeta_{\mathbf{f}}^2 - (\zeta_{\mathbf{f}}^*)^2 \leq 0 \\ \mathbf{c}_{\mathbf{B}3} &= \bar{\zeta}_{\mathbf{f}}^2 - \zeta_{\mathbf{f}}^2 \leq 0 \\ \mathbf{c}_{\mathbf{B}4} &= \sigma_{\mathbf{x}}^2 - (\sigma_{\mathbf{x}}^*)^2 \leq 0 \\ \mathbf{c}_{\mathbf{B}5} &= \bar{\sigma}_{\mathbf{x}}^2 - \sigma_{\mathbf{x}}^2 \leq 0 \end{aligned} \quad (4)$$

Here the minimum attainable error levels are $\bar{\zeta}_{\mathbf{f}}$ and $\bar{\sigma}_{\mathbf{x}}$ which can be derived from the maximum error improvement and state of the art error levels $\bar{\zeta}_{\mathbf{f}} = \zeta_{\mathbf{f}}^* - \bar{\Delta\zeta}_{\mathbf{f}}$ and $\bar{\sigma}_{\mathbf{x}} = \sigma_{\mathbf{x}}^* - \bar{\Delta\sigma}_{\mathbf{x}}$.

Figure 1(b) represents the same notional cost-benefit tradeoff solution space as transformed by alternative solution variables B as for Equation 4. As expected, all of the constraints in Figure 1(b) are now straight lines.

Independent of the choice or construction of the objective cost function, the shape of the solution space as defined by the inequality constraints in Equation 3 or 4 or Figure 1 provides a great deal of insight. For example, the corners of the solution space defined by the intersection of the constraints provides a set of input error targets which *must* be

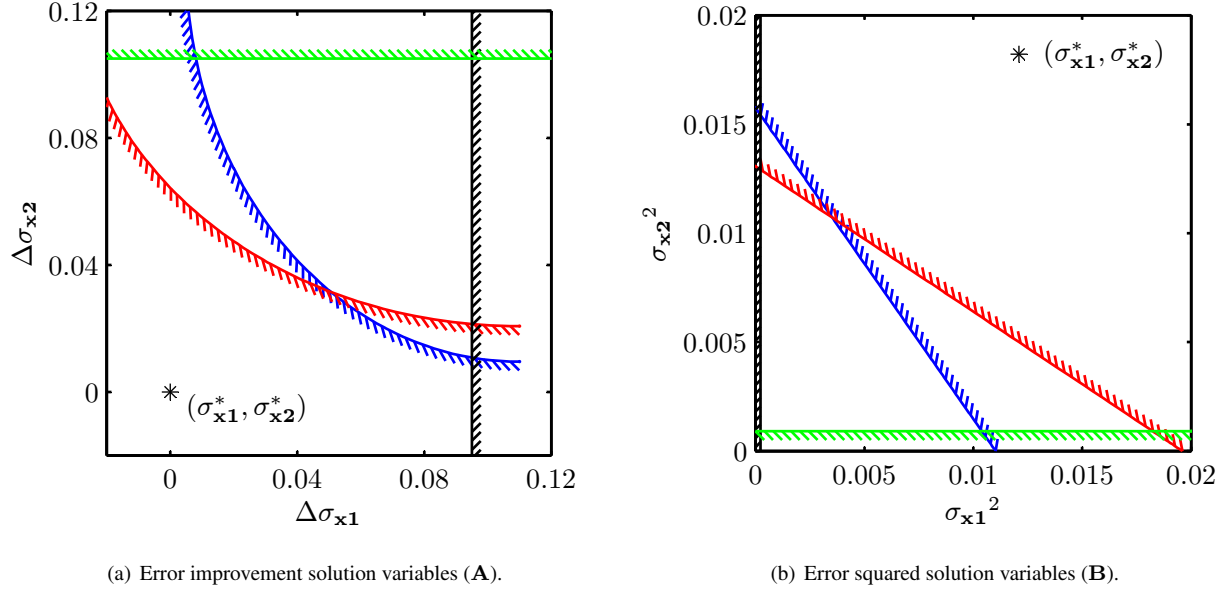


Figure 1. Depictions of the cost-benefit solution space with alternate choices of the solution variables.

attained in order to meet the system level error goals. This ‘do or die’ fidelity level is an important metric for project planning.

Fortunately, because of the linearity of Equation 2, the corners of the solution space may be calculated directly and efficiently. Unfortunately, because of the high dimensionality of the solution space common to complex systems problems, visualization of the solution space as in Figure 1 may not be possible. Section VI will investigate techniques for intuitively providing the most important information about the solution space to the decision maker.

Error improvements are not expected to be independent. Work to reduce one source of error will likely pay benefit for another. For example, investment intended to reduce error in the calculation of aerodynamic drag may also reduce error in the calculation of lift and moment. Likewise, investment in stress analysis may reduce error in mass properties and aeroelasticity calculations. It may be beneficial to consider the optimization problem in terms of sets of related error reductions which are grouped into error reduction programs.

From this program-centric point of view, the solution space of the optimization problem is transformed to the set of investment levels in each program. It is convenient to handle these investment levels as fractions of the fully-funded program cost f_P . The total cost and the error reductions due to the programs will need to be expressed as functions of these program fractions. This results in an alternative problem statement C given below as Equation 5.

$$\begin{aligned}
 \text{Choose: } & \mathbf{C} = (f_P) \\
 \text{To minimize: } & \$C = \$C(\mathbf{C}) \\
 \text{Subject to: } & c_{C1} = \sigma_f - \bar{\sigma}_f \leq 0 \\
 & c_{A2} = -f_P \leq 0 \\
 & c_{A3} = f_P - 1 \leq 0
 \end{aligned} \tag{5}$$

In this formulation, bounds constraints are added to ensure that the program fractions remain between zero and one. The system error goals are expressed as inequality constraints as before.

Stating the optimization problem as in Equation 5 results in a problem with some complication to the error goals constraint c_{C1} especially when compared to form B. The variable transformation may be viewed as a new set of basis vectors. This complication reduces the transparency of the equations, but does not significantly increase the complexity of the optimization problem.

V. Error Reduction Cost Modeling

The author is unaware of any previous attempts to quantitatively model the cost required to improve the fidelity of an engineering model. Although there is no standard error reduction cost model (ERCM) in the literature, some characteristics which a candidate ERCM should possess may be stated by observation.

The cost to do nothing should be zero. The cost to reduce error should increase monotonically with the error reduction. Error can not be completely eliminated, so the cost to completely eliminate error should be infinite.

When considering possible functional relationships to construct an ERCM, it is important to keep the end user in mind. Any ERCM will need to be tuned to the problem at hand. This includes a representation of the current state of the art and some sense of the difficulty of the problem. An intuitive ERCM which requires less calibration information and which specifies that information in a straightforward manner will be preferred.

An equal cost ERCM as suggested by Equation 6 provides an obviously naïve approach; there is no reason to believe that diverse error reductions will come at equal cost. An equal cost model satisfies the zero intercept and monotonic requirements, but does not have the asymptotic behavior desired in the preceding paragraphs. Of course, the bounds constraints of Equation 3 or 4 may be used to prevent obviously incorrect results. An equal cost model does not provide for grouping errors into programs.

$$\$ = \sum \Delta c_f + \sum \Delta \sigma_x \quad (6)$$

In Equation 6, \$ is the total cost and Δc_f is a vector of improvements to the vector of error sources c_f . Similarly, $\Delta \sigma_x$ is a vector of improvements to the vector of input error sources σ_x .

The principle attraction to Equation 6 lies in its utter simplicity. The user may apply an equal cost ERCM without specifying any information about the absolute or relative cost of reducing a set of errors. Much of the complexity of the cost-benefit tradeoff of error reduction is due to the relative importance of each error source and the sensitivity of the system level errors as governed by the propagation of error through the complex system (Equation 2). This complexity is expressed in the cost-benefit tradeoff through the system error goal constraints of Equation 3 or 4. For problems where the required error improvements are small, or are dominated by the importance of a particular error source (rather than the relative cost of improvement), an equal cost ERCM may suffice.

A linear ERCM provides a logical progression from the very simple equal cost model. A linear model will allow the user to specify the relative cost of improving each error source. This information will be provided through an error cost slope - cost per improvement. A linear ERCM with cost slopes m_f and m_x is included as Equation 7.

$$\$ = \sum m_f \Delta c_f + \sum m_x \Delta \sigma_x \quad (7)$$

Again, the primary value of a linear ERCM lies in simplicity. This approach incorporates the relative cost of an error improvement as well as the relative importance of each error source to meeting the system error goals. The addition of relative cost information should allow informed decision making in the presence of more subtle tradeoff. Although a linear ERCM is still simple and does not exhibit the desired asymptotic behavior, it is expected to be quite effective when the fidelity improvements required to meet the goals aren't great.

When dramatic improvements in fidelity are required, thereby pushing the limits of what is possible, a more sophisticated ERCM will likely be required. This situation will likely push the required error improvements into the asymptotic area of investment representing the limits of what is possible. Technology forecasting 'S-curves' present one potential approach to such an ERCM.

The ideas represented by these candidate ERCM's may be simply extended to allow related error reductions to be placed into programs. An equal cost, linear, or nonlinear model may be adopted to model progress within a program. Such a program-based model would of course need to be paired with form C (Equation 5) of the optimization problem.

When combining programs, there are two important aspects to consider. First, the combination of cost; typically simple addition will suffice. Second, the combination of error; simple addition is an obvious model, but some situations may call for more sophisticated models. For example, there may be two competing programs directed at improving a drag estimate; a wind tunnel test and a computational study. If each study aims to reduce the error in the estimate by 5%, do we expect 10% improvement if both programs are fully funded?

Equation 8 represents a simple ERCM capable of representing error improvement programs. Each program has a fully funded cost and a corresponding set of error reductions. Progress within a program is represented by a fractional level of funding for each program; error improvement is reduced by the same fraction. The cost of multiple programs combine through simple addition as does the benefit of multiple programs.

$$\begin{aligned}
 \$ &= \sum \$_{\mathbf{P}} \mathbf{f}_{\mathbf{P}} \\
 \Delta\zeta_{\mathbf{f}} &= \sum (\Delta\zeta_{\mathbf{f}})_{\mathbf{P}} \mathbf{f}_{\mathbf{P}} \\
 \Delta\sigma_{\mathbf{x}} &= \sum (\Delta\sigma_{\mathbf{x}})_{\mathbf{P}} \mathbf{f}_{\mathbf{P}}
 \end{aligned}
 \tag{8}$$

In Equation 8, $\$_{\mathbf{P}}$ is a vector of the cost to fully fund a series of programs and $\mathbf{f}_{\mathbf{P}}$ is the vector of fractions of completion for the programs. Similarly, $(\Delta\zeta_{\mathbf{f}})_{\mathbf{P}}$ and $(\Delta\sigma_{\mathbf{x}})_{\mathbf{P}}$ are the vectors of improvement for each error source for a fully funded program.

VI. Intuitive Decision Making Tools

Although the cost-benefit tradeoff of error reduction programs has been stated as an optimization problem, there is no compelling reason to expose the decision maker to the optimizer used to solve the problem. Instead, intuitive interfaces can be developed for the specification of the problem elements – system error goals, base error values, and error reduction programs. Furthermore, intuitive displays such as an investment portfolio pie chart can be developed to report the solution to the problem. In this manner, the decision maker may focus on the decision at hand; not on the mechanism used to provide guidance.

The decision support user interface developed in the author’s previous work¹⁻³ was extended in this way to support the cost-benefit tradeoff problem. A snapshot of the user interface during a typical use case is included as Figure 2.

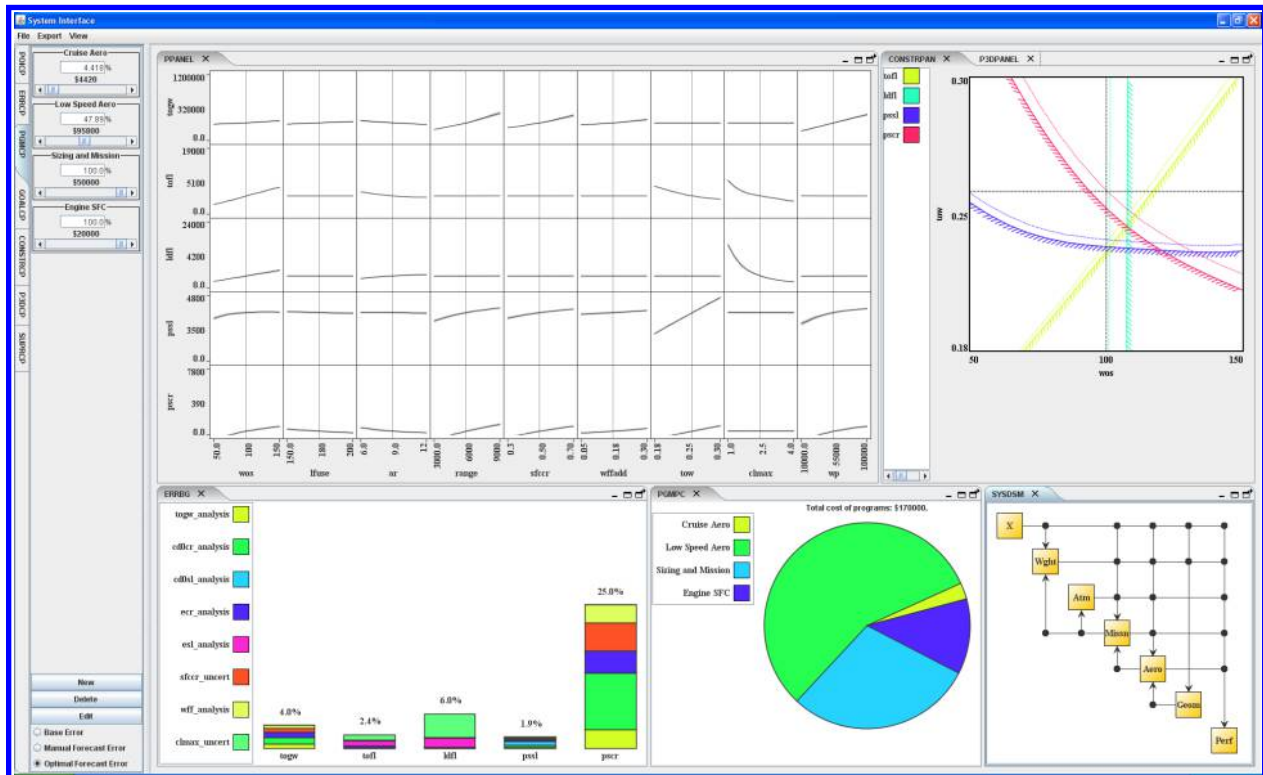


Figure 2. Decision support user interface.

Fortunately, the optimization problem posed in this work is very inexpensive to solve in the environment provided by the decision support interface. Consequently, the optimization problem is solved automatically ‘behind the scenes’; thereby the shape of the solution space and the optimal investment portfolio will instantly reflect any change in the user inputs.

Although this problem is well suited to a direct constrained optimization technique such as Sequential Quadratic Programming (SQP), the author is unaware of any off-the-shelf SQP routines written in Java. Faced with the task of developing a nonlinear programming solver, a more simple approach was taken.

The optimization problem is solved by a Sequential Unconstrained Minimization Technique (SUMT) which modifies an unconstrained optimizer with a penalty function. After each SUMT iteration converges, the penalty is made more severe and the problem is re-optimized.

The unconstrained optimizer used in this work is a Polak-Ribiere conjugate gradient method developed in previous related work.^{12,13} It uses a cubic line search which employs the Wolfe-Powell conditions to test for convergence. In this work, the unconstrained optimizer was extended to a simple SUMT approach using a quadratic exterior penalty function.¹⁴ Due to the relatively simple mathematical form of the problem, analytical derivatives were used for the objective function and all of the constraint functions.

VII. Example Complex System

To demonstrate and test the cost-benefit analysis method and user interface presented in this research, the example developed in previous work¹⁻³ was extended. In that work, this example was used to propagate diverse error sources through a representative complex system. The user interface enabled what-if error reduction scenarios to be explored at the same time as design decisions. A Monte Carlo analysis was also performed to validate the sensitivity technique for error propagation.

The complex system model representing an aircraft was assembled from simple tasks representing the primary disciplines contributing to the design of a transport aircraft. Textbook methods were used to implement each of the tasks. The resulting system is shown in Figure 3.

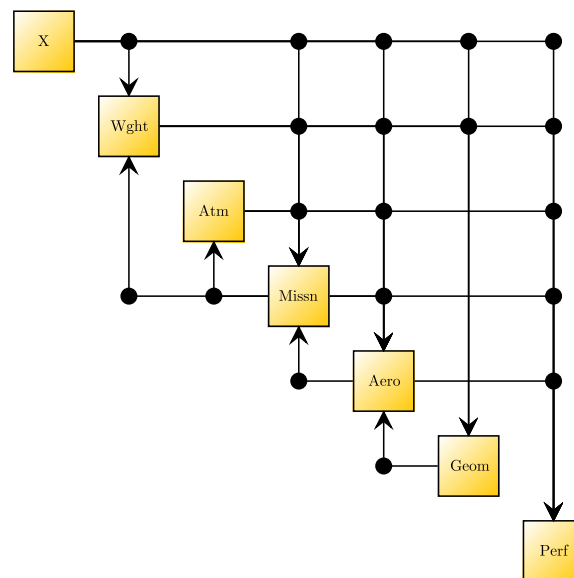


Figure 3. Example system DSM

An intuitive metamodeling environment developed in parallel to this effort^{2,12,13} was used to fit Gaussian process metamodels to the analyses over a range of input variables appropriate for the example problem. The visualization environment relies on the metamodels to provide an inexpensive approximation of the tasks.

VIII. Cost-Benefit Example

The ERCM model specified by Equation 8 requires the specification of base error sources and error improvement programs. The base error sources are set to the same baseline values presented in previous work. The base error values are listed in Table 1.

Four representative error improvement programs were specified for this example. Programs are defined by a fully-funded cost and a set of one or more improvements to error sources. In this example, the programs do not overlap – no two programs improve the same error source; this is not a requirement, but it simplifies the example. The error improvement programs are described in Table 2.

Table 1. Base error sources

Quantity	Value
$\sigma C_{L,max}$	5%
ςe_{cr}	5%
ςW	5%
$\varsigma C_{D,0 cr}$	5%
σSFC	5%
$\varsigma C_{D,0 sl}$	5%
ςe_{sl}	5%
$\varsigma W_f/W$	5%

Table 2. Error improvement programs

(a) Cruise aerodynamics		(b) Sizing and mission	
Program Cost	\$100,000	Program Cost	\$50,000
Quantity	Improvement	Quantity	Improvement
ςe_{cr}	-2.5%	ςW	-2.5%
$\varsigma C_{D,0 cr}$	-3%	$\varsigma W_f/W$	-4%

(c) Low speed aerodynamics		(d) Engine SFC	
Program Cost	\$200,000	Program Cost	\$20,000
Quantity	Improvement	Quantity	Improvement
$\sigma C_{L,max}$	-3%	σSFC	-3%
$\varsigma C_{D,0 sl}$	-2.5%		
ςe_{sl}	-2%		

The system propagated error due to these sources is depicted in Figure 4. For these results, the system input variables were set to the baseline settings. Each bar in the graph depicts the total propagated error in each system response. Each component of the bar represents how much of that error was due to each error source. Figure 4(a) corresponds to the baseline error settings; this represents the current state of the art and the maximum value possible for each error source. Alternatively, Figure 4(b) corresponds to the propagated error if all four programs from Table 2 were fully funded; this represents the minimum value possible for each error source. Note that the vertical scale between these figures is not the same.

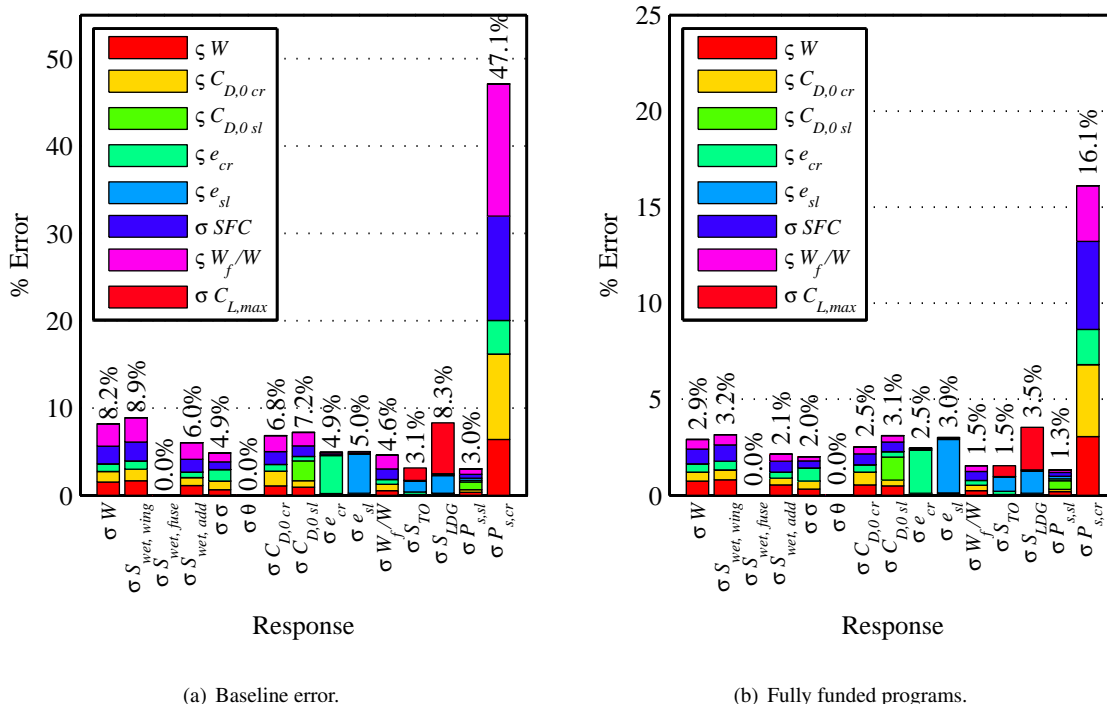


Figure 4. System propagated error.

It would cost \$370,000 to fully fund all of the proposed programs. The proportions of the investment for a fully funded portfolio are depicted in the pie chart in Figure 5.

The optimization problem statement of the cost-benefit error reduction requires the decision maker to specify some system level propagated error goals. These goals are used as inequality constraints in the optimization problem. The maximum and minimum propagated error values depicted in Figure 4 can provide guidance for selecting reasonable goals. The goals used in this example problem are listed in Table 3.

Table 3. System error goals.

Quantity	Target
σW	4%
σS_{TO}	3%
σS_{LDG}	6%
$\sigma P_{s,cr}$	30%

When these goals are applied to the optimization problem (with a minimum cost objective), a solution in terms of program fractions and the corresponding investment portfolio is generated. This optimal solution for this baseline problem is presented in Figure 6.

In this sample problem solution, two programs (Engine SFC and Sizing & Mission) were fully funded, Low Speed Aero was partially funded, and Cruise Aero received minimal funding. The total investment required to meet the system

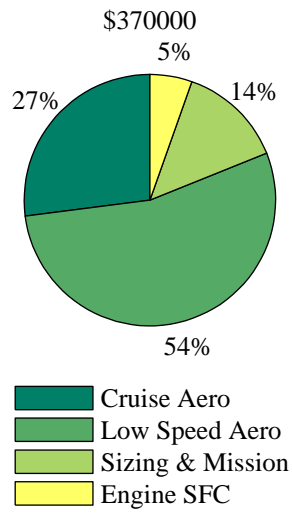


Figure 5. Investment portfolio to fully fund all programs.

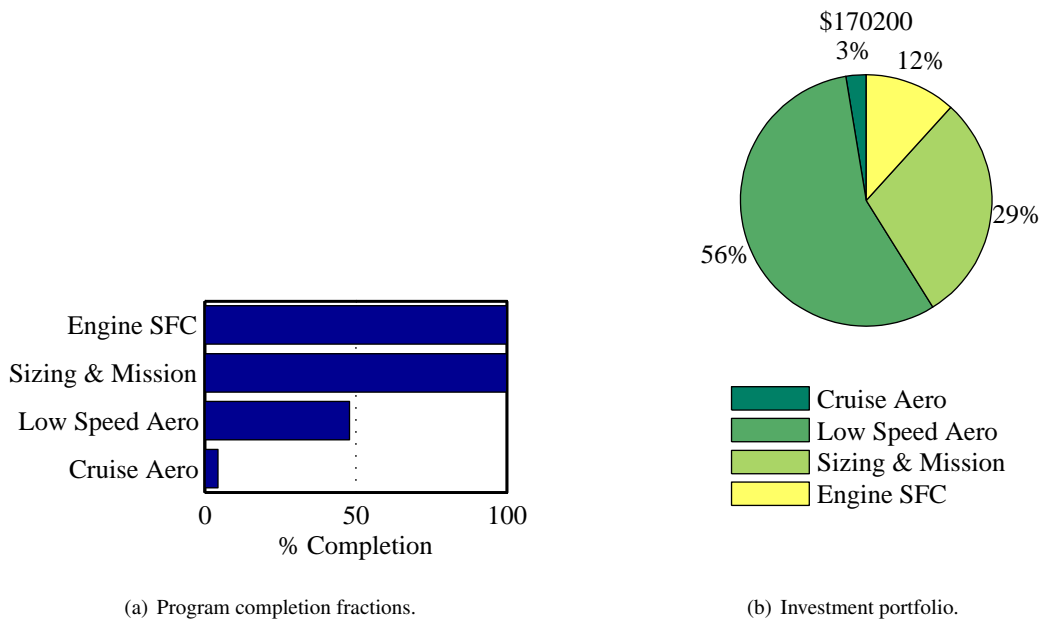


Figure 6. Optimal program plan for baseline goals.

fidelity goals was \$170,200, considerably less than the fully funded cost of \$370,000. Although Low Speed Aero was funded less than 50%, the program's high cost caused it to represent 56% of the total investment portfolio. Part of the reason that both Sizing & Mission and Engine SFC were fully funded is their relatively low program cost.

The system propagated error corresponding to the optimal program solution is depicted in Figure 7. Inspection of the errors for all of the responses reveals that two of the goals were met exactly (the constraints were active) and the other two were surpassed by a wide margin.

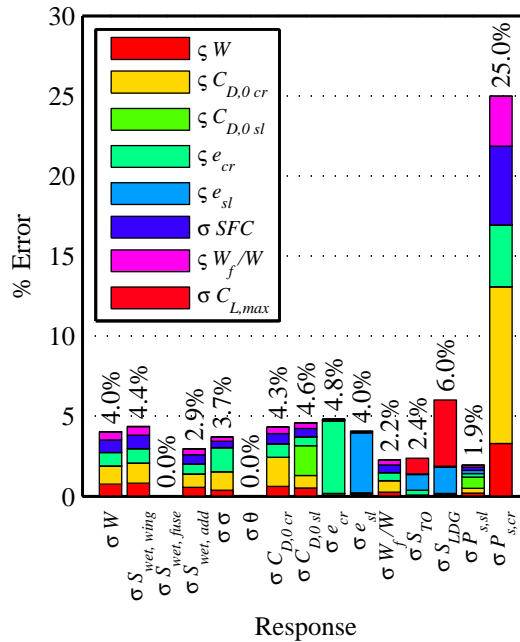


Figure 7. System propagated error for optimal program plan.

This capability to specify target error levels – and to solve for the investment program required to meet those levels – allows the error propagation problem to be *inverted*. Specific quantitative guidance can be provided to the decision maker faced with planning a research and development program for a large, complex system.

This guidance is recalculated and provided interactively by the decision making interface. The user may quickly explore the changes induced by changing error goals, program costs, program benefits, base error levels, and system level inputs – design variables, mission parameters, technology measures, etc.

IX. Conclusions

The cost-benefit tradeoff of error reduction for a complex system was made explicit and quantitative. A cost-benefit model for error reduction programs was proposed which is capable of representing the combined benefit of a program on multiple error sources. The tradeoff problem was stated as an optimization problem using system propagated error targets as inequality constraints.

A nonlinear constrained optimizer was integrated into an existing computer program for assisting decision making. User interface elements were also added which allowed for the intuitive specification of error reduction programs and system level error goals. This integration was done seamlessly and invisibly; the uninformed user will have no idea that an optimizer is involved in producing the presented solutions.

The results of the optimization problem represent quantitative and impartial guidance for the distribution of limited resources in the development program for a complex system. These results are presented interactively to the user in an intuitive format which is well matched to the problem at hand.

This research has developed a tool which provides an intuitive way for the user to pose a question – and receive a rigorous answer – to a question which had not previously been posed. This question of investment planning for fidelity improvement programs for complex systems has historically been of central importance for the success or failure of

major aerospace projects.

X. Acknowledgements

This work has been supported by NASA Research Announcement Grant NNX07AO14A under the supervision of technical monitor Craig Nickol.

References

- ¹McDonald, R. A., "Error Allocation in Complex Systems Design," No. AIAA-2006-7130, 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Portsmouth, VA, September 6-8 2006.
- ²McDonald, R. A., *Error Propagation and Metamodeling for a Fidelity Tradeoff Capability in Complex Systems Design*, Ph.D. thesis, Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA, August 2006.
- ³McDonald, R. A., "The Role of Error in the Conceptual Design of a Transport Aircraft," No. AIAA 2007-654, 45th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, 9-11 January 2007.
- ⁴Baird, D. C., *Experimentation: An Introduction to Measurement Theory and Experiment Design*, Prentice-Hall Inc., Englewood Cliffs, NJ, 1962.
- ⁵Taylor, J. R., *An Introduction to Error Analysis The Study of Uncertainties in Physical Measurements*, University Science Books, Sausalito, California, 2nd ed., 1997.
- ⁶Oberkampf, W. L., Diegert, K. V., Alvin, K. F., and Rutherford, B. M., "Variability, Uncertainty, and Error in Computational Simulation," *HTD-Vol. 375-2, ASME Proceedings of the 7th AIAA/ASME Joint Thermophysics and Heat Transfer Conference*, edited by R. S. Amano, B. F. Armaly, T. S. Chen, A. Emery, J. A. Liburdy, D. P. Shatto, N. K. Anand, B. Blackwell, T. Y. Chu, J. Lage, O. Oosthuizen, and K. A. Woodbury, No. H1137B, 1998.
- ⁷Oberkampf, W. L., DeLand, S. M., Rutherford, B. M., Diegert, K. V., and Alvin, K. F., "Error and Uncertainty in Modeling and Simulation," *Reliability Engineering & System Safety*, Vol. 75, 2002, pp. 333-357.
- ⁸Sobieszcanski-Sobieski, J., "Sensitivity of Complex, Internally Coupled Systems," *AIAA Journal*, Vol. 28, No. 1, January 1990, pp. 153-160.
- ⁹Sobieszcanski-Sobieski, J., "Higher Order Sensitivity Analysis of Complex, Coupled Systems," *AIAA Journal*, Vol. 28, No. 4, April 1990, pp. 756-758.
- ¹⁰Sobieszcanski-Sobieski, J., "Sensitivity Analysis and Multidisciplinary Optimization for Aircraft Design: Recent Advances and Results," *Journal of Aircraft*, Vol. 27, No. 12, December 1990, pp. 993-1001.
- ¹¹Du, X. and Chen, W., "Efficient Uncertainty Analysis Methods for Multidisciplinary Robust Design," *AIAA Journal*, Vol. 40, No. 3, 2002, pp. 545-552.
- ¹²Baukol, C., *Development of an Integrated Gaussian Process Metamodeling Application for Engineering Design*, Master's thesis, California Polytechnic State University, 2009.
- ¹³Baukol, C., McDonald, R. A., and Delmas, N., "A User Friendly Interface for Gaussian Process Metamodeling," No. AIAA 2009-802, 47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition, Orlando, FL, 2009.
- ¹⁴Rao, S. S., *Engineering Optimization: Theory and Practice*, John Wiley & Sons, 3rd ed., 1996.