13661
P.17

NASA Technical Memorandum 104074

A SYSTEM APPROACH TO AIRCRAFT OPTIMIZATION

Jaroslaw Sobieszczanski-Sobieski

March 1991

National Aeronautics and Space Administration

Langley Research Center
Hampton, Virginia 23665-5225

A SYSTEM APPROACH TO AIRCRAFT OPTIMIZATION

by

Jaroslaw Sobieszczanski-Sobieski
NASA Langley Research Center, MS 246
Hampton Virginia 23665 U.S.A.

SUMMARY

Mutual couplings among the mathematical models of physical phenomena and parts of a system such as an aircraft complicate the design process because each contemplated design change may have a far reaching consequences throughout the system. This paper outlines techniques for computing these influences as system design derivatives useful for both judgmental and formal optimization purposes. The techniques facilitate decomposition of the design process into smaller, more manageable tasks and they form a methodology that can easily fit into existing engineering organizations and incorporate their design tools.

1. INTRODUCTION

The engineering design process is a two-sided activity as illustrated in Fig. 1. It has a qualitative side dominated by human inventiveness, creativity, and intuition. The other side is quantitative, concerned with generating numerical answers to the questions that arise on the qualitative side. The process goes forward by a continual question-answer iteration between the two sides. To support that process one needs a computational infrastructure capable of answering the above questions expeditiously and accurately. For development of such an infrastructure, the idea of "push button design" ought to be discarded in favor of a realistic recognition of the role of human mind as the leading force in the design process and of the role of mathematics and computers as the indispensable tools. It is clear that while conceiving different design concepts is a function of human mind, the evaluation and choice among competing, discretely different concepts, e.g., classical configuration vs. a forward swept wing and a canard configuration, requires that each concept be optimized to reveal its full potential. This approach is consistent with the creative characteristics of the human mind and the efficiency, precision, and infallible memory of the computer.

The computational infrastructure for support of the design process entails data management, graphics, and numerics. The first two embodied in CAD/CAM systems are well-known and are taken for granted as a framework for the numerics. The purpose of this paper is to introduce some new techniques which may be regarded as a subset of the latter. Included in the discussion are the system behavior derivatives with respect to design variables, their use for both judgmental and mathematical optimization purposes, formal decomposition of a system into its components, and ramifications of that decomposition for system sensitivity analysis and optimization, all illustrated by aircraft application examples. The impact on the design process of a methodology formed by these techniques is also examined.

2. EFFECT OF DESIGN VARIABLE CHANGE IN A COMPLEX SYSTEM

An aircraft is a complex system of interacting parts and physical phenomena whose behavior may be influenced by assigning values to the design variables. Since the design process is, generally, concerned with an aircraft that does not yet exist, one works with its surrogate—a system of mathematical models that correspond, roughly, to the engineering disciplines, and to physical parts of the vehicle. These mathematical models send data to each other as depicted in the center of Fig. 2, and they also accept design variable values as inputs from the designers. To know how to change these design variables, designers must know the answers to "what if" questions, such as "what will be the effect on the system behavior if the design variables X, Y, Z will be changed to $X+\Delta X$, $Y+\Delta Y$, $Z+\Delta Z$?", implied by the loop in Fig. 2.

An example of a hypersonic aircraft in Fig. 3 illustrates how difficult it may be to answer an "what if" question for even a single variable change in a complex system in which everything influences everything else. Consider a structural cross-sectional thickness t in the forebody of a hypersonic aircraft shown in the upper half of Fig. 3 as a design variable that is to be changed. The lower half of the figure depicts a complex chain of influences triggered by the change of t and, ultimately, affecting the vehicle performance. The change of t influences the position of the bow shock wave relative to the inlet in two ways: through the nose deflection, and through the weight and the center of gravity position both of which affect the trimmed angle of attack. The shock wave position relative to the inlet is a strong factor in the propulsive efficiency of the engine that, in turn, combines with the weight to influence the aircraft performance. Additional influence on performance is through the angle of attack whose change alters the vehicle aerodynamic lift and drag. The resultant modifications of the performance may require resizing of the vehicle which, of course, may be a sufficient reason to change t again, and so on, until the iteration represented by the feedback loop in Fig. 3 converges.

The above iteration engages a number of mathematical models such as structures, aerodynamics, propulsion, and vehicle performance. For the purposes of this discussion, each such model may be regarded as a black box converting input to output and, consistent with the black box concept, the inner workings of the model will be left outside of the scope of the discussion. While it may not be too difficult to evaluate the input-on-output effect for each single black box taken separately, evaluation of the resultant change for the entire system of such black boxes may be exceedingly difficult, especially when iterations are involved. In general, the resultant may be a small difference of large numbers, so even its sign may be impossible to predict without a precise reanalysis of the entire system.

To generalize from the above example, let X and Y denote the system input and output, respectively, e.g., the structural cover thickness t and a measure of performance such as the aircraft range. Then, the derivative dY/dX is a measure of the influence of X on Y and its value answers quantitatively the associated "what if" question. More precisely, the derivative value informs only about the rate of change of Y at the value of X for which the derivative was obtained. Determination of the increment of Y for a given finite increment of X, if $Y(X)$ is nonlinear, can be done approximately by a linear extrapolation

$$Y_{\text{new}} = Y_{\text{old}} + \frac{dY}{dX}\Delta X \tag{1}$$

Capability to extrapolate as above for many different X and Y variables, enables one to decide, either judgmentally or by means of an optimization program, which variables X to change and by how much, in order to improve the design in some way. However, that capability is predicated on availability of the derivatives dY/dX termed the system design derivatives (SDD). For large system analysis, especially if the analysis is iterative, its is advantageous to avoid the brute force method of finite differencing on the entire system analysis in computation of these derivatives.

2.1 System Design Derivatives

Remembering that the mathematical model of an engineering system may be an assemblage of a large number of mathematical models representing its components and the governing physical phenomena, it is convenient to limit the discussion to three such black box models since that number is small enough to foster comprehension and, yet, large enough to develop a general solution pattern. Ascribing a vector function representation to each black box, the set of equations representing the system of the black boxes α, β, γ exchanging data as illustrated in Fig. 4 is

$$\begin{aligned} Y_\alpha &= Y_\alpha(X, Y_\beta, Y_\gamma) \\ Y_\beta &= Y_\beta(X, Y_\alpha, Y_\gamma) \\ Y_\gamma &= Y_\gamma(X, Y_\alpha, Y_\beta) \end{aligned} \tag{2}$$

The Y and X variables in the above are vectors entered in the black boxes selectively, e.g., some, but not necessarily all, elements of the vectors X and Y_α enter the black box β as inputs. Regarding $Y_\beta(X, Y_\alpha, Y_\gamma)$ as an example of a black box, the arguments, X, Y_α, Y_γ, are the inputs and Y_β is an output. The functions in eq. 2 are coupled by their outputs appearing as inputs, hence they form a set of simultaneous equations that can be solved for Y for given X. The act of obtaining such a solution is referred to as the system analysis (SA). In the presence of nonlinearities, SA is usually iterative.

For each function in eq. 2, one can calculate derivatives of output with respect to any particular input variable, assuming that other variables are fixed. From the entire system perspective, these derivatives are partial derivatives since they measure only the local input-on-output effect, as opposed to SDD which are total derivatives because they include the effect of the couplings. To prepare for further discussion, the partial derivatives corresponding to the Y-inputs are collected in the Jacobian matrices designated by a pair of subscripts identifying the origins of the output and input, respectively. For example,

$$J_{\gamma\alpha} = [\partial Y_\gamma / \partial Y_\alpha] \tag{3}$$

is a matrix whose j-th column is made of the partial derivatives $\partial Y_{\gamma i}/\partial Y_{\alpha j}$. Assuming the length of Y_γ as N_γ and the length of Y_α as N_α, the dimensions of matrix $J_{\gamma\alpha}$ are $N_\gamma \times N_\alpha$. It will be mnemonic to refer to the partial derivatives in the Jacobian matrices as the cross-derivatives.

The remaining partial derivatives corresponding to the X-inputs are collected in vectors, one vector per each of the NX elements of the vector of design variables X, e.g.

$$\{\partial Y_\alpha / \partial X_k\}' = [\partial Y_\alpha / \partial X_k], \quad k = 1, \ldots, NX; \tag{4}$$

is a vector of the length N_α ($'$ denotes transposition).

Calculation of the above partial derivatives may be accomplished by any means available for a particular black box at hand, and may range from finite differencing to quasi-analytical methods (ref. 1, and 2).

It was shown in ref. 3 that differentiation of the functions in eq. 2 as composite functions and application of the implicit function theorem leads to a set of simultaneous, linear, algebraic equations, referred to as the Global Sensitivity Equations (GSE), in which the above partial derivatives appear as coefficients and the SDD are the unknowns. For the system of eq. 2, the GSE are

$$\begin{bmatrix} I & -J_{\alpha\beta} & -J_{\alpha\gamma} \\ -J_{\beta\alpha} & I & -J_{\beta\gamma} \\ -J_{\gamma\alpha} & -J_{\gamma\beta} & I \end{bmatrix} \begin{Bmatrix} dY_\alpha/dX_k \\ dY_\beta/dX_k \\ dY_\gamma/dX_k \end{Bmatrix} = \begin{Bmatrix} \partial Y_\alpha/\partial X_k \\ \partial Y_\beta/\partial X_k \\ \partial Y_\gamma/\partial X_k \end{Bmatrix} \tag{5}$$

These equations may be formed only after the SA was performed for a particular X, a particular point in the design space because the computation of the partial derivatives requires that all the X and Y values be known. For a given X, the matrix of coefficients depends only on the system couplings and is not affected by the choice of X for the right hand side. Hence that matrix may be factored once and reused in a backsubstitution operation to compute as many sets of SDD's as many different X_k variables are represented in the set of multiple right-hand-side vectors.

As recommended in ref. 3, numerical solution of eq. 5 and interpretation of the SDD values will be facilitated by normalization of the coefficients in the matrix and in the right hand sides by the values of Y_o and X_o of the Y and X variables for which the partial derivatives were calculated. The normalized coefficients take on the following form, illustrated by a few examples from i-th row in the β partition in eq. 5

$$-\frac{\partial Y_{\beta i}}{\partial Y_{\alpha j}} q_{\beta\alpha ij}; \quad -\frac{\partial Y_{\beta i}}{\partial Y_{\gamma j}} q_{\beta\gamma ij}; \quad \frac{dY_{\beta i}}{dX_k} q_{\beta X ik} \tag{6}$$

where the normalization coefficients q are

$$q_{\beta\alpha ij} = \frac{Y_{\alpha jo}}{Y_{\beta io}}; \quad q_{\beta\gamma ij} = \frac{Y_{\gamma jo}}{Y_{\beta io}}; \quad \bar{q}_{\beta Xik} = \frac{X_{ko}}{Y_{\beta io}}$$

Solution of the normalized eq. 5 yields normalized values of the SDD's from which the unnormalized values may always be recovered given the above definitions.

Formation of the GSE and their solution for a set of SDD's will be referred to as the System Sensitivity Analysis (SSA).

2.2 Utility of the System Design Derivatives

The SDD carry the trend information that under a conventional approach would be sought by resorting to statistical data or to the parametric studies. The former have the merit of capturing a vast precedent knowledge but may turn out to be ineffective if the vehicle at hand is advanced far beyond the existing experience. The latter provide an insight into the entire interval of interest but only for a few variables at a time, and that insight tends to be quickly lost if there are many design variables, in which case the computational cost of the parametric studies also may become an impediment.

In contrast, the SDD information is strictly local but it reflects the influences of all the design variables on all aspects of the system behavior. Therefore, the SSA should not be regarded as a replacement of the above two approaches but as their logical complement whose results are useful in at least two ways.

2.2.1 Ranking design variables for effectiveness

A full set of SDD for a system with NY variables in Y and NX variables in X is a matrix $NY \times NX$. The j-th column of the matrix describes the degree of influence of variable X_j on the behavior variables Y. Conversely, the i-th row shows the strength of influence of all the design variables X on the i-th behavior variable Y_i. For normalized SDD's, comparison of these strengths of influence becomes meaningful and may be used to rank the design variables by the degree of their influence on the particular behavior variable. This ranking may be used as a basis for judgmentally changing the design variable values and for deciding which design variables to use in a formal optimization.

An example of such ranking is illustrated for the wing of a general aviation aircraft shown in Fig. 5. The design variables are thicknesses t of the panels in the upper cover of the wing box and the behavior variable is the aircraft range R. The chain of influences leading from a panel thickness to the range calculated by means of the Breguet formula is depicted on the left side in Fig. 6. In the Breguet formula, W_e denotes the zero-fuel weight and W_p stands for the fuel weight. Increasing t in one of the panels increases the weight W_e and, in general, reduces the drag of a flexible wing by stiffening its structure. Consequently, the range is influenced in conflicting ways that would make prediction by judgment difficult. However, the corresponding SSA yields the SDD's for the upper row of the wing cover panels illustrated by the heights of the vertical bars over the upper wing cover panels in Fig. 6. The bars show that among all the wing cover panels, increasing t in the extreme outboard panel would increase range the most.

2.2.2 Gradient-guided formal optimization

Most of the formal optimization methods applicable in large engineering problems use the first derivative information to guide the search for a better design. Since the SDD values provide such information for all the Y and X variables of interest, the SSA may be incorporated, together with SA, in a system optimization procedure (SOP) based on the well-known piecewise approximate analysis approach (e.g., ref. 4). The SOP flowchart is depicted in Fig. 7. An important benefit of the SOP organization is the opportunity for parallel processing seen in the flowchart operation immediately following the SA. In that operation, one computes concurrently the partial derivatives of input with respect to output for all the system black boxes, in order to form the Jacobian matrices (eq. 3) and the right-hand-side vectors (eq. 4) needed to form the GSE (eq. 5) whose solution yields the SDD's. In a conventional approach, these SDD's would be computed by finite differencing on SA. The SDD values are subsequently used in Approximate Analysis (extrapolation formulas) that supplies the optimizer (a design space search algorithm) with information on the system behavior for every change of the design variables generated by that optimizer, and does it at a cost negligible in comparison with the cost of SA.

A generic hypersonic aircraft similar to the one that was discussed in Fig. 3 was used as a test for the above optimization. The geometrical design variables for the case are shown in Fig. 8. Additional design variables were the deflections of the control surfaces, and the cross-sectional structural dimensions of the forebody. The propulsive efficiency measured by the I_{sp} index, defined as the thrust minus drag divided by the fuel mass flow rate, was chosen as the objective function to be maximized. The aircraft take-off gross weight (TOGW) for a given mission is very sensitive to that index, thus maximization of the index effectively minimizes TOGW. For the reasons discussed in conjunction with Fig. 3, the problem requires consideration of a system composed of aerodynamics, propulsion, performance analysis, and structures. The optimization included constraints on the aircraft as a whole and on behavior in the above disciplines. Results are shown in Table 1 in terms of the initial and final values of the design variables (cross-sectional dimensions omitted) and of the objective function, all normalized by the initial values. Considering that the initial values resulted from an extensive design effort using a conventional approach, the nearly 13% improvement in the propulsive efficiency was regarded as very significant indeed.

Another example of the SOP application is the case of a hypersonic interceptor (Fig. 9a) reported in ref. 5. The optimization objective was the minimum of TOGW for the mission profile illustrated in Fig. 9b. The system comprised the modules of the configuration geometry, configuration mass properties, mission performance analysis, aerodynamics, and propulsion as depicted in Fig. 10, and the design variables were the wing area, scale factor for the turbojet engine, scale factor for the ramjet engine, and the fuselage length. The constraint list included a limit on the time needed to reach the combat zone, the take-off velocity, and the fuel available mass being at least equal to the one required (the fuel balance constraint). It should be noted that in a conventional approach to aircraft design, satisfaction of the latter constraint is one of the principal goals in development of a baseline configuration whose improvement is subsequently sought by parametric studies in which the design variables are varied while always striving to hold the fuel balance constraint satisfied. In contrast to that practice, the optimization reported in ref. 5 allowed the fuel balance constraint to be violated in the baseline configuration and achieved satisfaction of that constraint in the course of the optimization process. This demonstrated that an optimization procedure may do more than just improve on an initial, feasible configuration; it can actually synthesize an optimal configuration starting with one that is not even capable of performing a required mission.

The optimization results are illustrated by a vertical bar chart in Fig. 11 that shows the changes of the design variables and of a significant (13%) improvement of the objective function. The figure shows also that the initially violated constraints of time to intercept and take-off velocity were brought to satisfaction in the optimal configuration. The SOP converged in only 4 to 5 repetitions of SA and SSA.

3. MERITS AND DEMERITS

Before discussion of the ramifications of the above sensitivity-based optimization in a system design process, it may be useful to examine briefly the merits and demerits of the proposed approach relative to the conventional technique of generating SSD by finite differencing on the entire SA.

3.1 Accuracy and Concurrent Computing

The SSA based on eq. 5 has two unique advantages. First, the accuracy of SDD is intrinsically superior to that obtainable from finite differencing whose precision depends on the step length in a manner that is difficult to predict. As pointed out in ref. 6 it is particularly true in the case of an iterative SA whose result often depends on an arbitrary, "practical" convergence criterion. Second, there is an opportunity for concurrent computing in the generation of the partial derivatives which exploits the technology of parallel processing offered by multiprocessor computers and computer networks. Concurrent computing also enables the engineering workload to be distributed among the specialty groups in an engineering organization to compress the project execution time.

3.2 Computational Cost

Experience indicates that in large engineering applications, most of the optimization computational cost is generated by the finite difference operations. Therefore, relative reduction of the cost of these operations translates into nearly the same relative reduction of the cost of the entire optimization procedure.

The computational cost of the SSA based on eq. 5, designated C_1, may be reduced, in most cases very decisively, below that of finite differencing on the entire SA, denoted by C_2, but to achieve that reduction the analyst should be aware of the principal factors involved. To define these factors, let the computational cost of one SA be denoted by CSA while CBA_i will stand for the computational cost of one analysis of the i-th black box in the system composed of NB black boxes. The i-th black box receives an input of NX_i design variables X, and NY_i variables Y from the other black boxes in the system. Assuming for both alternatives the simplest one step finite difference algorithm that requires one reference analysis and one perturbed analysis for each input variable, the costs C_1 and C_2 may be estimated as

$$C_1 = \sum_i (1 + NX_i + NY_i) CBA_i; \qquad (7)$$

$$C_2 = (1 + NX) CSA$$

Even though one may expect $CBA_i < CSA$, a sufficiently large NY_i may generate $C_1 > C_2$ and render SSA based on eq. 5 unattractive compared to finite differencing on the entire SA. This points to NY_i, termed the interaction bandwidth, as the critical factor whose magnitude should be reduced as much as possible. Reducing the interaction bandwidth requires judgment as illustrated by an example of an elastic, high aspect ratio wing treated as a system whose aeroelastic behavior is modeled by interaction of aerodynamics and structures, represented by an CFD analysis and Finite Element analysis codes, respectively. If one let the full output from each of these black boxes be transmitted to the other, there might be hundreds of pressure coefficients entering the structural analysis and thousands of deformations sent to the aerodynamic analysis. With the NY_i values in the hundreds and thousands, respectively, it would be quite likely that $C_1 > C_2$. However, one may condense the information flowing between the two black boxes by taking advantage of the high aspect ratio wing slenderness. For a slender wing it is reasonable to represent the entire aerodynamic load by, say, a set of 5 concentrated forces at each of 10 separate chords, and to reduce the elastic deformation data to, say, elastic twist angles at 7 separate chords. This condensation reduces the NY_i values to 50 for structures and 7 for aerodynamics. In the finite element code, that implies 50 additional loading cases all of which can be computed very efficiently by the multiple loading case option—a standard feature in finite element codes. The CFD code would have to be executed only 7 additional times. Thus, the advantage of the interaction bandwidth condensation is evident. In general, a condensation such as the one described above for a particular example may be accomplished by the reduced basis methods, among which the Ritz functions approach is, perhaps, the best known one.

3.3 Potential Singularity

One should be aware when using SSA based on eq. 5 that, in some cases, the matrix of coefficients in these equations may be singular. In geometrical terms, a solution in SA may be interpreted geometrically as a vertex of hyperplanes on which the residuals of the governing equations for the black boxes involved are zero. As pointed in ref. 3, eq. 5 are well-conditioned if these hyperplanes intersect at large angles, ideally when they are mutually orthogonal. For two functions of two variables the zero-residual hyperplanes reduce to the zero-residual contours, and an example of a nearly-orthogonal solution intersection is shown in Fig. 12a. In some cases, the intersection angles may tend to be very acute, in the limit they may be zero in which case a solution exist by virtue of tangency of two curved contours as illustrated in Fig. 12b. It is shown in ref. 3 that eq. 5 imply local linearization of these contours in the vicinity of the intersection point so that the solution point is interpreted as an intersection of the tangents. Consequently, in the situation depicted in Fig. 12b the tangents coincide and the matrix of eq. 5 becomes singular. In such a case, eq. 5 should be replaced by an alternative formulation of the system sensitivity equations in ref. 3 based on residuals.

There were no cases of singularity reported so far in any applications probably because the system solutions of the type illustrated in Fig. 12b characterize an ill-posed system analysis usually avoided in practice.

3.4 Discrete Variables

Neither the reference technique nor the SSA based on eq. 5 can accommodate truly discrete design variables. Truly discrete design variables are defined for the purposes of this discussion as those with respect to which SA is not differentiable. These are distinct from quasi-discrete variables with respect to which SA is differentiable but which may only be physically realizable in a set of discrete values. An example of the former is an engine location on the aircraft: either under the wing or at the aft end of the fuselage. An example of the latter is sheet metal thickness available in a set of commercial gages.

In the case of truly discrete design variables, different combinations of such variables define different design concepts (alternatives) and each concept may be optimized in its own design space of the remaining continuous variables, in order to bring it up to its true potential. Then, one may choose from among the optimal alternatives. Occasionally, a continuous transformation might be possible between two concepts that seem to be discretely different. For example, a baseline aircraft with a canard, a wing, and a conventional tail may be reshaped into any configuration featuring all, or only some of these three lifting surfaces. This is so because a sensitivity-guided SOP may eliminate a particular feature, if a design variable is reserved for that feature and if the feature is present in the initial design (however, a feature initially absent cannot, in general, be created).

3.5 Non-utilization of Disciplinary Optimization

Organization of the SOP discussed above may be described as "decomposition for sensitivity analysis followed by optimization of the entire, undecomposed system". It may be regarded as a shortcoming that the procedure leaves no clear place for the use of the vast expertize of optimization available in the individual black boxes representing engineering disciplines. Examples of such local, disciplinary optimization techniques are the optimality criteria for minimum weight in structures, and shaping for minimum drag for a constant lift in aerodynamics. It appears that combining these local, disciplinary optimization techniques with the overall system optimization should benefit the latter. Indeed, one way in which these techniques may be used without changing anything in the SOP organization described above is in the SOP initialization. Obviously, starting SOP from a baseline system composed of the black boxes already preoptimized for minimum weight, minimum drag, maximum propulsive efficiency, etc. should accelerate the SOP convergence and improve the end result. Such local optimizations could be accomplished separately for each black box, assuming X and guessing at the Y inputs.

Beyond that, the issue of incorporating the local, disciplinary optimization in SOP remains to be a challenge for further development. Some solutions were proposed in ref. 7 and 8 but their effectiveness is yet to be proven in practice. This issue will be taken up again in the later discussion in conjunction with the special case of a hierarchic system decomposition which does accommodate the local optimizations.

4. FORMAL DECOMPOSITION

When the system at hand contains a large number of black boxes and, especially, if there is little or no experience with its solution, it is useful to apply a formal technique to determine the data flow among the black boxes. The data flow information is useful because it characterizes the system as non-hierarchic, hierarchic, or hybrid, and this, in turn, helps to choose an optimization approach and to establish an efficient organization of computing. Such formal techniques are available in Operations Research and some of them were adapted for the system analysis and optimization purposes, e.g., ref. 9.

4.1 N-square Matrix

A brief introduction to one such technique begins with a formalization of a black box (a module) in the system as one that receives inputs through the top and bottom horizontal sides and sends the output through the left and right vertical sides as as shown in Fig. 13. Using that formalism, one can represent a four-module system example depicted by the diagram (known as the graph-theoretic format) in Fig. 14a in a different format shown in Fig. 14b. That format is known as the N-square Matrix format because N modules placed along the diagonal form an N^2 table. The N-square Matrix format assumes that the modules are executed in order from upper left to lower right (although, if possible, concurrent executions are allowed). If the execution order is not yet known, the order along the diagonal may be arbitrary. Referring to Fig. 13, each module may, potentially, send data horizontally, left and right, and receive vertically from above and from below. The actual data transmissions from and to i-th module are determined by comparing the module input list to the predecessor module output lists while moving upward in column i. Wherever a needed input item is found on the output list from module j, a dot is placed at the intersection of the i-th column and j-th row as a data junction indicating transmission of output from module j to input of module i. After the predecessor module search gets to the first module, it switches to module $i+1$ and continues downward through all the successor modules to module N. If more than one source is found for a particular input item, a unique, single source must be judgmentally selected. However, an output item may be used by several receiver modules and may also be sent to the outside. The input items that could not be found in the vertical search are designated primary inputs to be obtained from the outside of the system. The above search is readily implementable on a computer.

When the above search procedure is completed for all the modules, the result is an N-square Matrix as in Fig. 14b that conveys the same information as the diagram in Fig. 14a but is amenable to computerized manipulation. To see what such manipulation may achieve, observe that each dot in the upper triangle of the N-square Matrix denotes an instance of the data feedforward, and each dot in the lower triangle notes an instance of the data feedback. Of course, every instance of a feedback implies an iteration loop required by the assumed diagonal order of the modules. However, that order may be changed at will by a code that may be instructed to switch the modules around, with the associated permutations of the rows and columns to preserve the data junction information, in order to eliminate as many instances of feedback as possible. If all of them are eliminated the system admits a sequential module execution, and may offer opportunities for concurrent executions of some modules. If a complete elimination of the feedbacks is not possible, they are reduced in number and clustered. An example of a fairly large N-square Matrix in the initial, arbitrary order is shown in Fig. 15a while its clustered state is shown in Fig. 15b. In the clustered state the system is hybrid—partially hierarchic and partially non-hierarchic. A software tool that is available to make the above transformation is described in ref. 9. All the modules in one of the clusters in Fig. 15b may be regarded as a new supermodule, and the system diagram may be drawn in terms of these supermodules as shown in Fig. 16. This diagram defines a hierarchic decomposition of a system because the data flow from the top of the pyramidal hierarchy to the bottom, without reversing the flow and without lateral flow, while inside of each cluster there is a system whose modules define a non-hierarchic decomposition.

The N-square Matrix structure has a reflection in the structure of the matrix of coefficients in eq. 5: each feedforward instance in the former gives rise to a Jacobian matrix located below the diagonal in the latter and each feedback is reflected in a Jacobian above the diagonal. Hence, a sequential system without feedbacks has a matrix of coefficients populated only below the diagonal so that eq. 5 may be solved by backsubstitution of the right hand sides without factoring of the matrix of coefficients.

4.2 SOP Adapted to Hierarchic System

When a decomposed system has a hierarchic structure, its SOP may be reorganized to include separate optimizations in each black box. This SOP version was introduced in ref. 10 and called an optimization by linear decomposition. It has found a number of applications, for example, it was the basis for an algorithm for multilevel structural optimization by substructuring in ref. 11, and its use in multidisciplinary applications was reported in ref. 12 for control-structure interaction and in ref. 13 for optimization of a transport aircraft.

Multilevel optimization of a hierarchic system by a linear decomposition exploits the top-down flow of the analysis information. At the bottom level, the inputs obtained from analysis at the next higher level and the appropriate design variables are regarded as constants in optimization of each, bottom-level black box. Derivatives of each such optimization are computed with respect to these input constants by means of an algorithm described in ref. 14 and are used in linear extrapolations (hence the name of the technique) to approximate the effect of the input constants on the optimization results. Optimizations in the black boxes at the next higher level approximate their influence on the lower level optimization by means of these extrapolations. Thus, the top black box optimization is performed taking an approximate account of the effect of its variables (the system level variables) on all the black boxes in the hierarchic pyramid. As mentioned in the foregoing, the advantages of the SOP exploiting the hierarchic structure of the system is a separation of the bottom level detailed optimizations from the top level system optimization, and breaking the large system optimization problem into a number of smaller optimization problems, in contrast to the non-hierarchic system SOP (Fig. 7) in which optimization is performed for the system as a whole. However, if any of these black boxes in a hierarchic system contains a cluster (see discussion of Fig. 16) of black boxes forming a non-hierarchic system, the non-hierachic system SOP (Fig. 7) may be used to optimize it locally. Hence, both methods for system optimization described above, the one based on the linear decomposition (ref. 10) as well as the SOP based on Fig. 7 flowchart have their place in optimization of a general case of a hybrid engineering system that exhibits both the hierarchic and non-hierarchic structures depicted in Fig. 16.

As reported in ref. 13, the linear decomposition method was used to optimize the variables of configuration geometry and cross-sectional structural dimensions of a transport aircraft illustrated in Fig. 17a for minimum fuel burned in a prescribed mission, under constraints drawn from the disciplines of aerodynamics, performance and structures. The analysis was relatively deep, e.g., a CFD code in aerodynamics, and a finite element model of the built-up structure of the airframe structures. The number of design variables was over 1300, and the number of constraints was also in thousands. Optimization was conducted decomposing the problem into a three-level hierarchic system shown in Fig. 17b. A sample of results is depicted in Fig. 18 showing a smooth convergence of the fuel mass and the structural weight in only 4 to 6 cycles (one cycle comprised the top-down analysis and the bottom-up optimizations), for both feasible and infeasible initial design.

5. GENERALIZATION TO ENTIRE VEHICLE DESIGN PROCESS

The approach to the system sensitivity and optimization discussed in the foregoing may be generalized to serve the entire design process as shown in ref. 15 using as an example a definition of that process given in ref. 16. The process defined in ref. 16 is a conventional, sequential process illustrated in Fig. 19. As suggested in the upper right corner of the flowchart, any change in a major design variable such as the wing or engine size requires reentry into the sequence and repetition of all the operations in the chain. However, the black boxes forming the sequence are also forming a coupled system whose diagram is depicted in Fig. 20. The arrows in the diagram represent the data flow among the black boxes, examples of the data being defined in Table 2. Application of the SSA based on eq. 5 to the system in Fig. 20 leads to GSE in the format shown in Fig. 21. In the abbreviated notation used in that figure, Y_{ij} stands for a Jacobian matrix J_{ij} defined in eq. 3. Solution of the equations shown in Fig. 21 yields the SDD values that answer the "what if" questions implied in the upper right corner of the flowchart in Fig. 19, and does it for all the variables of interest simultaneously and without repeating the entire chain for every question. The SDD values may then be used to support judgmental design decisions and/or to guide a formal optimization according to the SOP in Fig. 7.

6. CONCLUDING REMARKS

Design of an engineering system, such as an aircraft, is a formidable task involving a myriad of cross-influences among the engineering disciplines and parts of the system. The time-honored approach to that task is to decompose it into smaller, more manageable tasks. The paper outlines some recently developed techniques that support such an approach by building an engineering system optimization on a modular basis, that comprises engineering specialty groups and their black box tools and allows engineers to retain responsibility for their domains while working concurrently on manageable tasks and communicating with each other by means of sensitivity data. The modularity and concurrence of operations map onto the familiar structure of the engineering organizations and are compatible with the emerging computer technology of multiprocessor computers and distributed computing. The only major new requirement is the generation of derivatives of output with respect to input in each specialty domain.

The use of sensitivity data as the communication medium is the distinguishing feature of the proposed approach and represent a major improvement over the present practice because it adds the trend information to the function value information. Both types of information enhance the human judgment and intuition while being readily usable in guiding the formal optimization procedures.

Acknowledgment: Contribution of the NASP configuration optimization example (Fig. 8 and Table 1) by Dr. F. Abdi and Mr. J. Tulinius of Rockwell International—North American is gratefully acknowledged.

7. REFERENCES

1. Proceedings of the Symposium on Sensitivity Analysis in Engineering, NASA Langley Research Center, Hampton Va, Sept. 1986; Adelman, H. M.; and Haftka, R. T.—editors. NASA CP-2457, 1987.

2. Adelman, H. A.; and Haftka, R. T.: Sensitivity Analysis of Discrete Structural Systems, AIAA J., Vol. 24, No. 5, May 1986, pp. 823–832.

3. Sobieszczanski-Sobieski, J.; On the Sensitivity of Complex, Internally Coupled Systems; AIAA/ASME/ASCE/AHS 29th Structures, Structural Dynamics and Materials Conference, Williamsburg, Va, April 1988; AIAA Paper No. CP-88-2378, and AIAA J., Vol. 28, No. 1, Jan. 1990, also published as NASA TM 100537, January 1988.

4. Sobieszczanski-Sobieski, J.: From a Black-Box to a Programing System, Ch.11 in Foundations for Structural Optimization—A Unified Approach; Morris, A. J., ed.J.; Wiley & Sons, 1982.

5. Consoli, R. D.; and Sobieszczanski-Sobieski, J.: Application of Advanced Multidisciplinary Analysis and Optimization Methods to Vehicle Design Synthesis; 17th Congress of the International Council of the Aeronautical Sciences (ICAS), Stockholm, September 1990, Proceedings of.

6. Thareja, R.; and Haftka, R. T.: Numerical Difficulties Associated with Using Equality Constraints to Achieve Multi-level Decomposition in Structural Optimization, AIAA Paper No. 86-0854, AIAA/ASME/ASCE/AHS 27th Structures, Structural Dynamics, and Materials Conference, San Antonio, Texas, May 1986.

7. Sobieszczanski-Sobieski, J.: Optimization by Decomposition: A Step from Hierarchic to Non-Hierarchic Systems; Second NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization; Hampton, Virginia, September 28–30 1988; Proceedings published as NASA CP - No. 3031; editor: Barthelemy, J. F.

8. Bloebaum, C. L.: Non-Hierachic System Decomposition in Structural Optimization Formal and Heuristic System Decomposition Methods in Multidisciplinary Synthesis; Ph.D. Dissertation, School of Engineering, Department of Aerospace Engineering, University of Florida, Gainsville, FL., 1991.

9. Rogers, J. L.: A Knowledge-Based Tool for Multilevel Decomposition of a Complex Design Problem; NASA TP 2903, 1989.

10. Sobieszczanski-Sobieski, J.: A Linear Decomposition Method for Large Optimization Problems—Blueprint for Development; NASA TM 83248, February 1982.

11. Sobieszczanski-Sobieski, J., James, B. B.; and Riley, M. F.: Structural Sizing by Generalized, Multilevel Optimization, AIAA J. Vol. 25, No. 1, January 1987, p. 139.

12. Zeiler, T. A.; and Gilbert, M. G.: Integrated Control/Structure Optimization by Multilevel Decomposition, NASA TM 102619, March 1990.

13. Wrenn, G. A.; and Dovi, A. R.: Multilevel Decomposition Approach to the Preliminary Sizing of a Transport Aircraft Wing; AIAA Jornal of Aircraft, Vol. 25, No. 7, July 1988, pp. 632–638.

14. Sobieszczanski-Sobieski, J.; Barthelemy, J. F.; and Riley, K. M.: Sensitivity of Optimum Solutions to Problem Parameters; AIAA J, Vol. 21, Sept. 1982, pp. 1291–1299.

15. Sobieszczanski-Sobieski, J.: Sensitivity Analysis and Multidisciplinary Optimization for Aircraft Design: Recent Advances and Results; Int'l Council for Aeronautical Sc., 16th Congress, Jerusalem, Aug.–Sept., 1988; Proceedings of, Vol. 2, pp. 953–964.

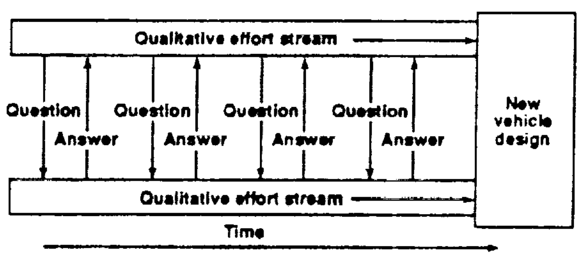
16. Torenbeek, E.: Synthesis of Subsonic Airplane Design; Delft University Press; 1982.

Table 1
Hypersonic aircraft optimization results

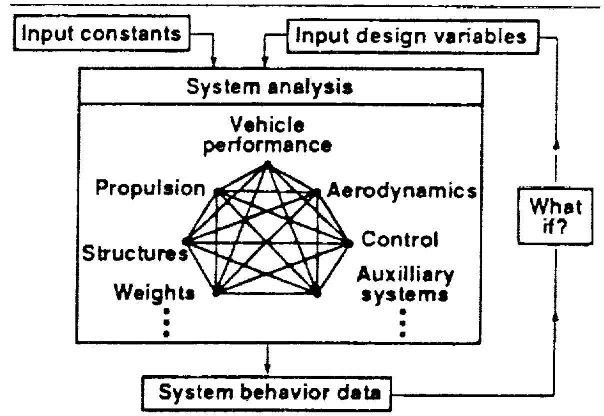
Optimization parameter	Baseline value	Optimization results
Design variable		
1. Forebody length	1.000	1.0209
2. Cone angle	1.000	0.9693
3. Upper surface height	1.000	1.0029
4. Geometric transition length	1.000	1.0760
5. Elevon deflection	1.000	0.8620
6. Bodyflap deflection	1.000	1.0320
Objective		
Effective trimmed Isp	1.000	1.1259

Table 2
Coupling data in aircraft system

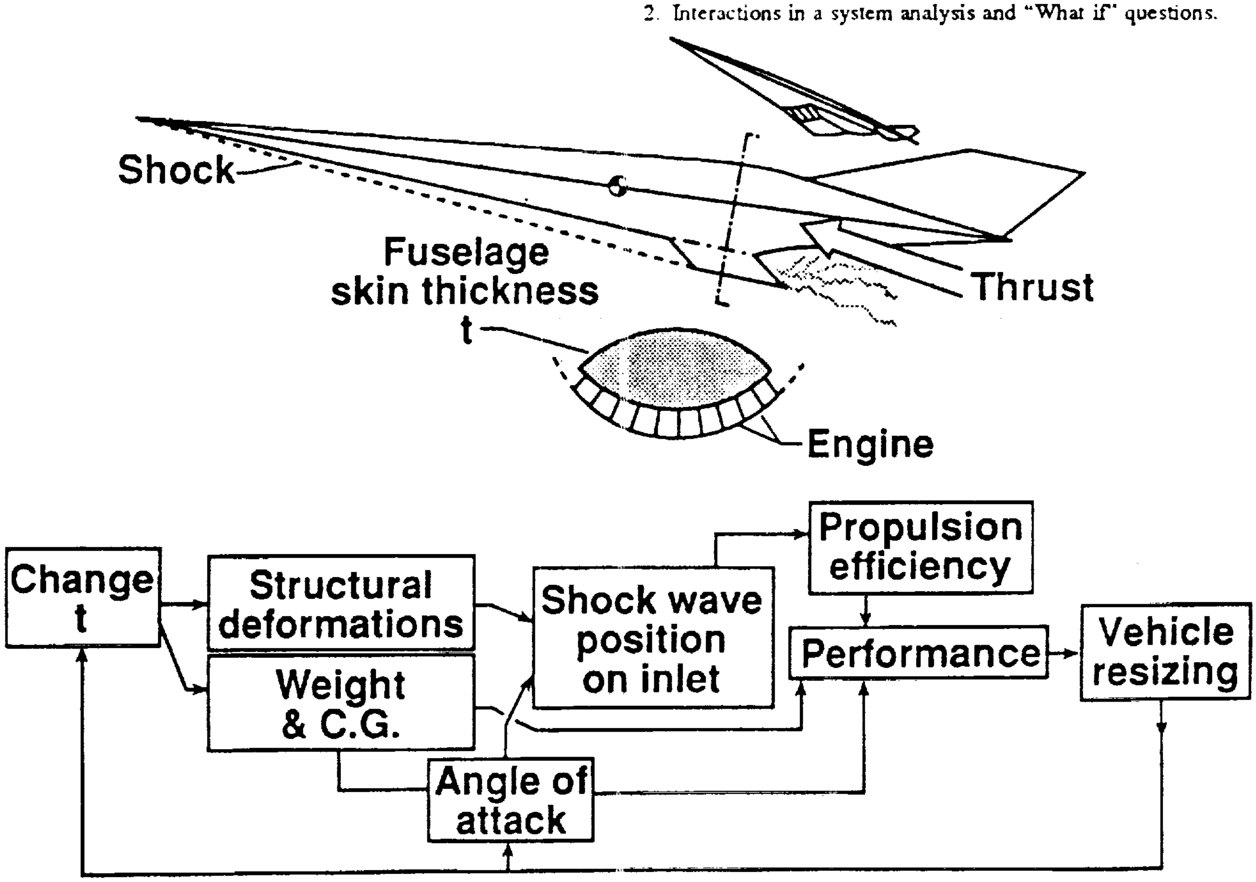
Vector Y	Content examples
1	See the box labeled INPUT.
2	Wing area, aspect ratio, taper, sweep angle, airfoil geometry data. Engine thrust.
3	Fuel tank locations and assumed volumes.
4	Wing structural weight and internal volume.
5	Take-off Gross Weight.
6	See box 6.
7	Landing gear weight and location, in stowed and extended position. Take-off field length.



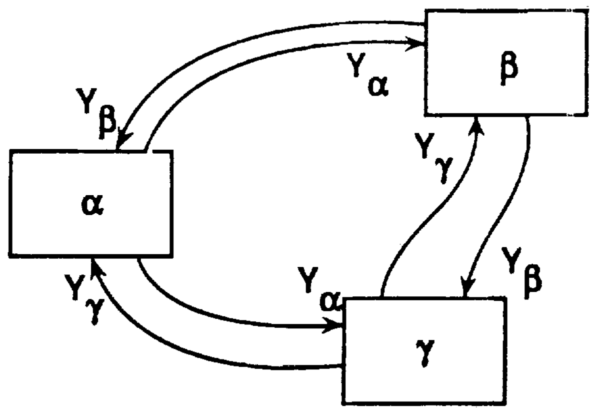
1. Qualitative and quantitative sides of a design process.



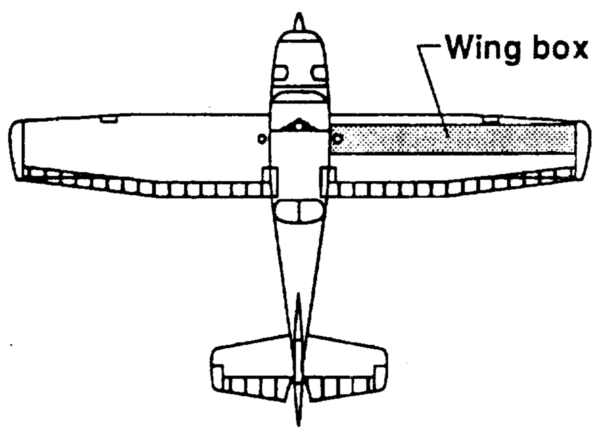
2. Interactions in a system analysis and "What if" questions.



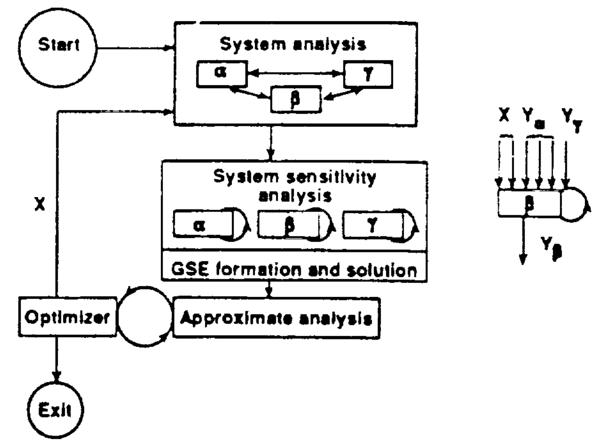
3. A design change triggering a complex chain of effects.



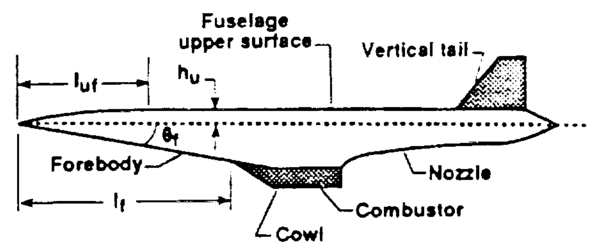
4. Example of a three component system.



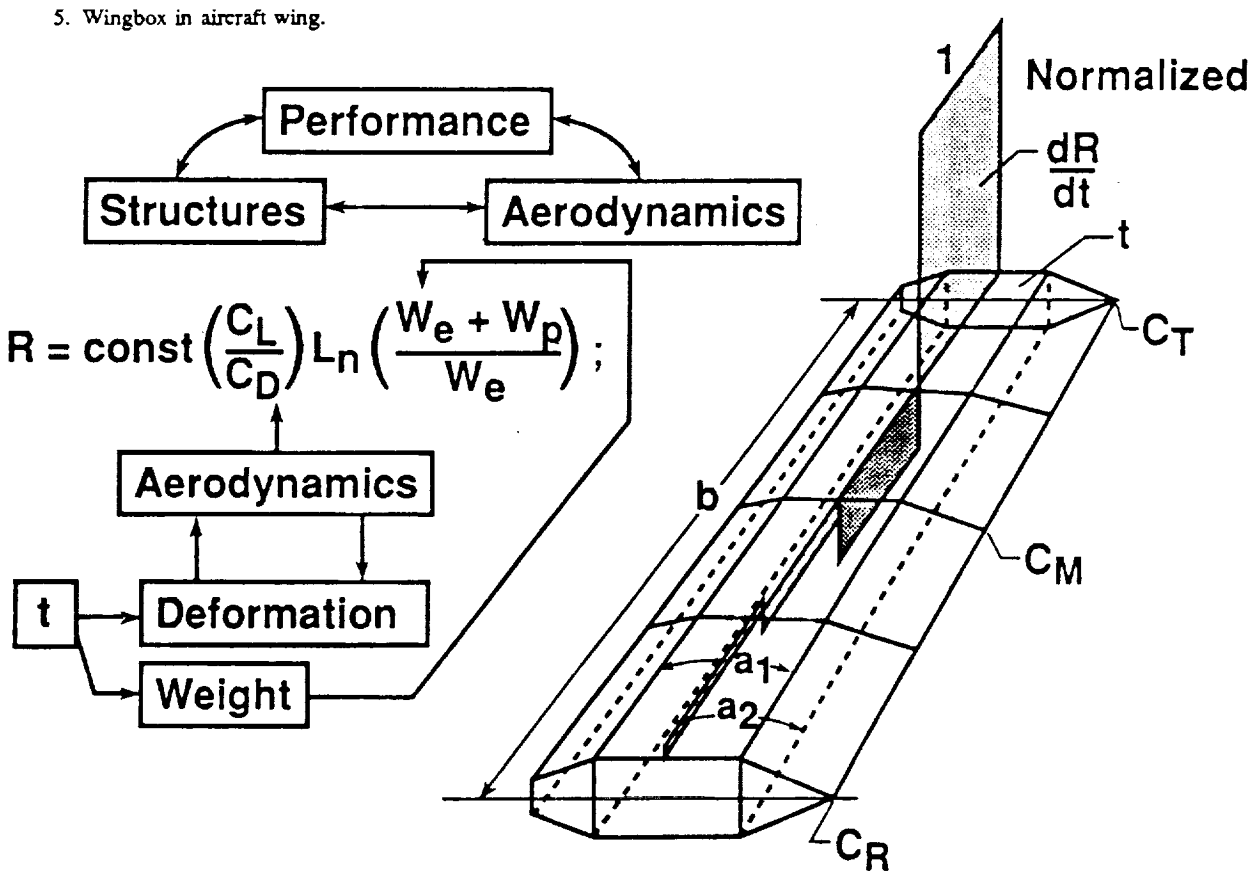
5. Wingbox in aircraft wing.



7. Flowchart of the System Optimization procedure (SOP).

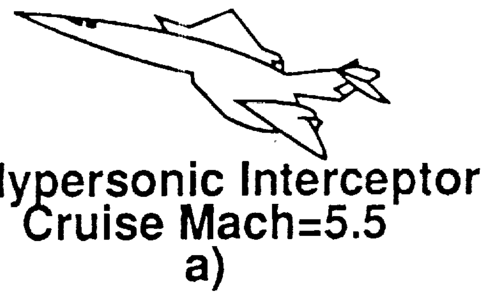


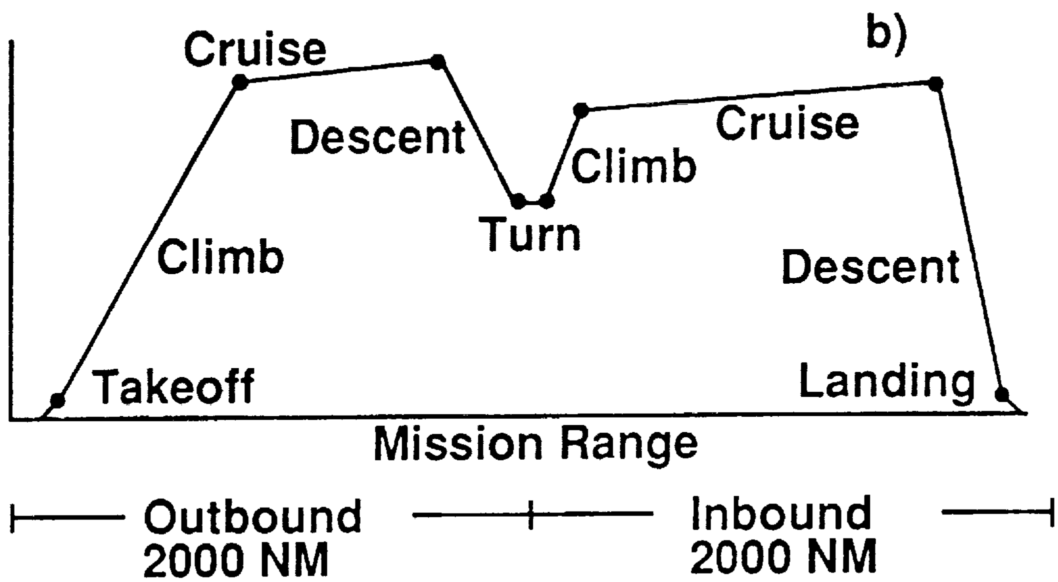
8. Hypersonic aircraft; some of the configuration design variables.



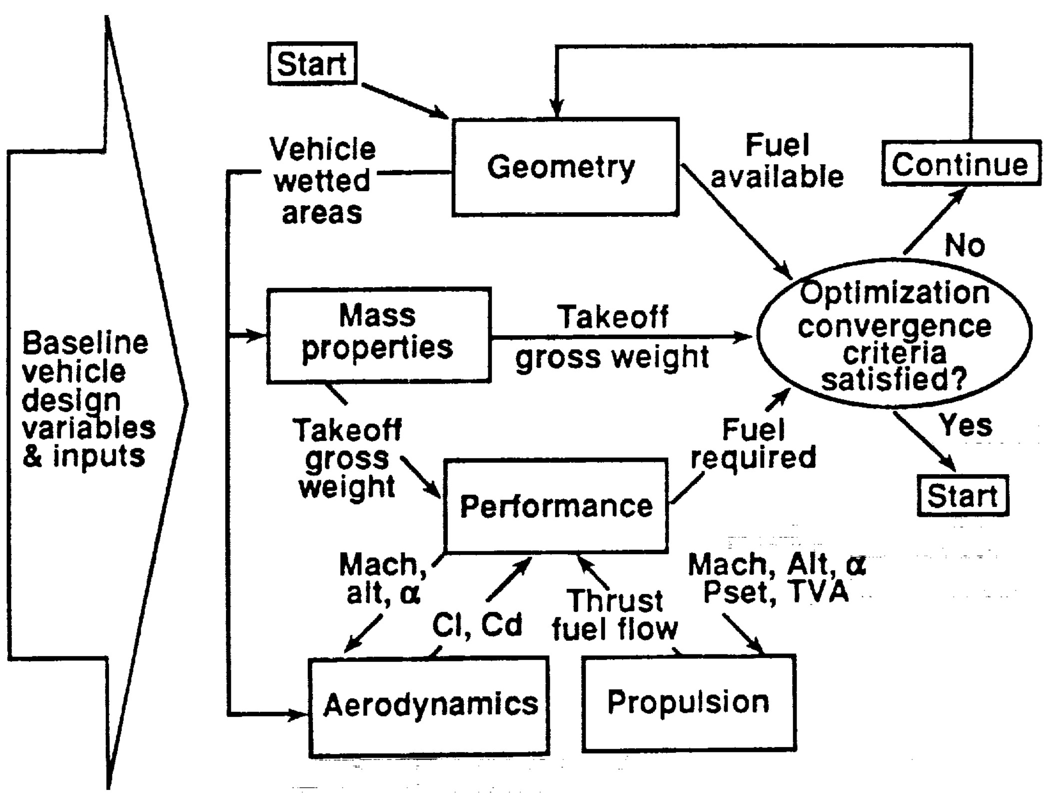
6. a) System of mathematical models, the Breguet formula, and the channels of influence for the wing cover thickness;

b) Vertical bars illustrate magnitude of derivatives of range with respect to thickness.

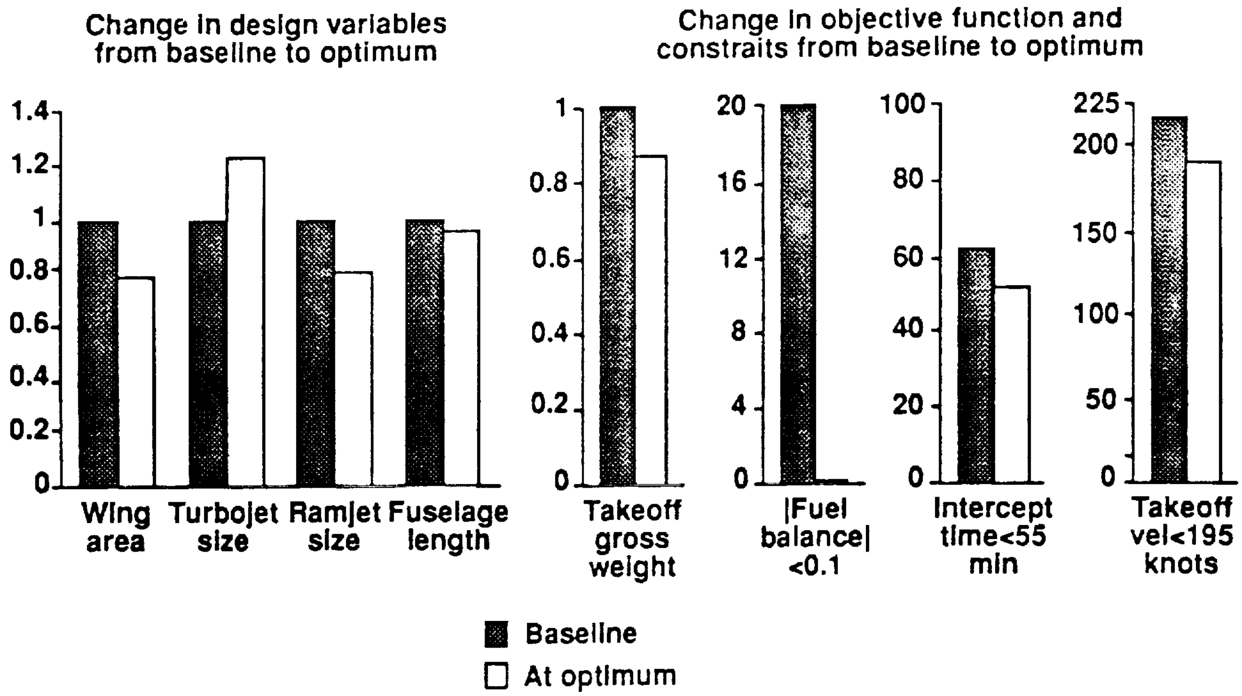




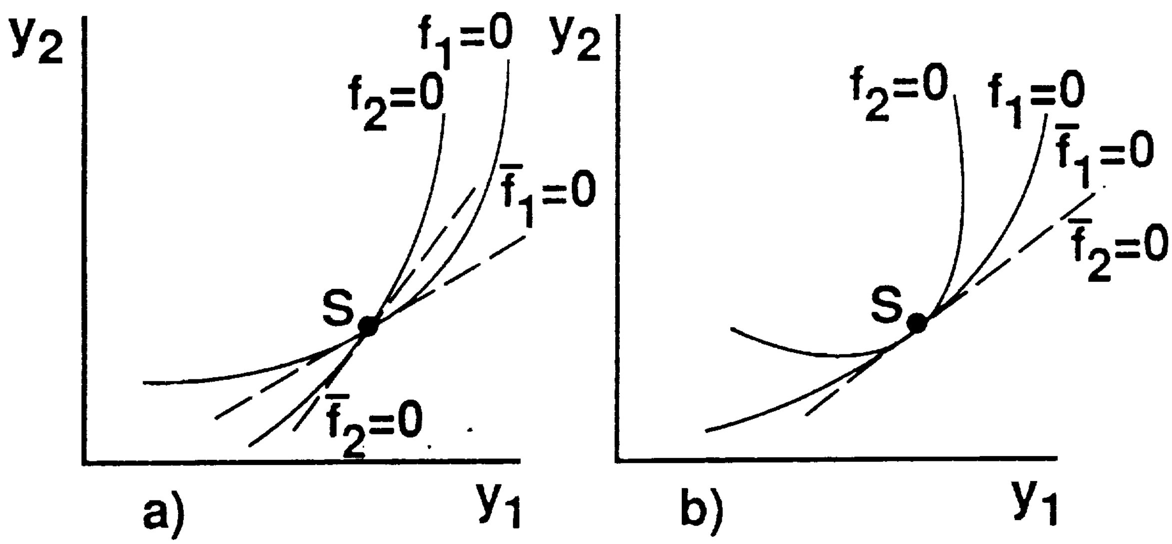
9. a) Hypersonic interceptor, b) Mission profile.



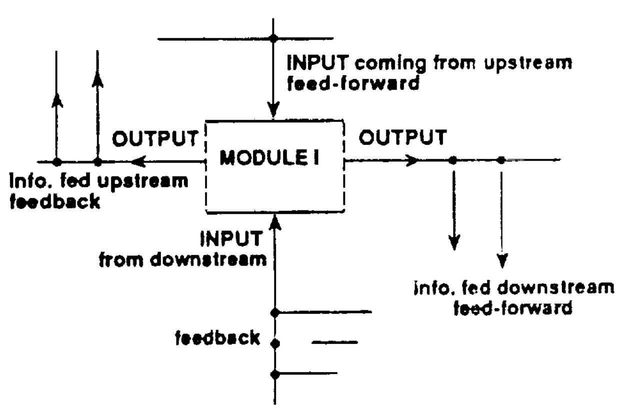
10. System of mathematical models for hypersonic interceptor optimization.



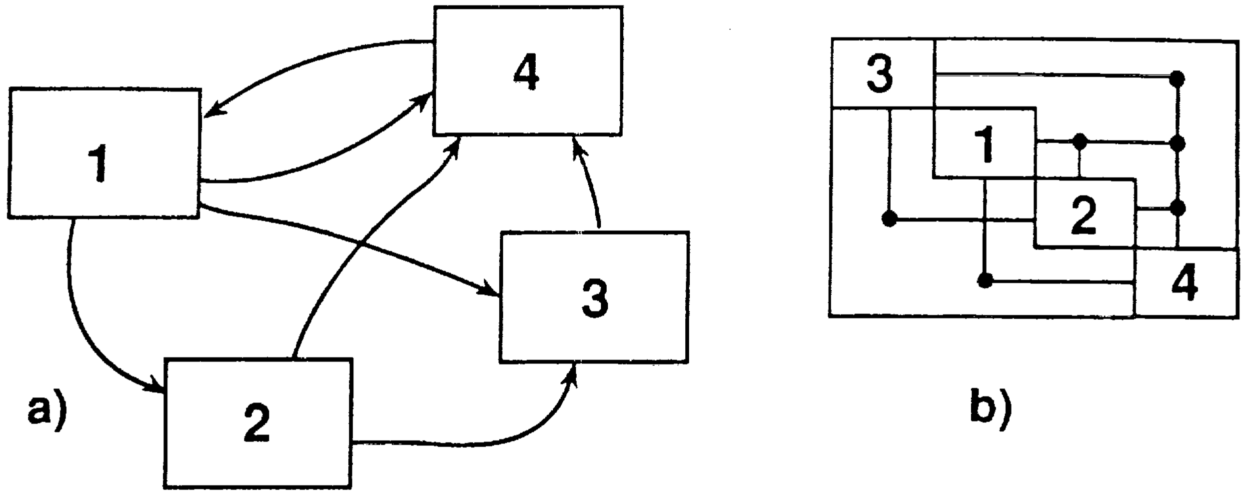
11. Sample results from hypersonic interceptor optimization.



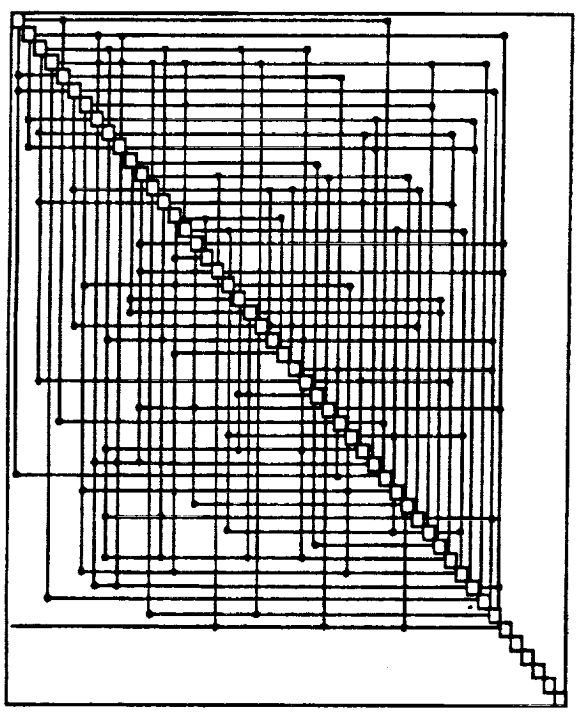
12. System solution: a) Intersection point; b) Tangency point.

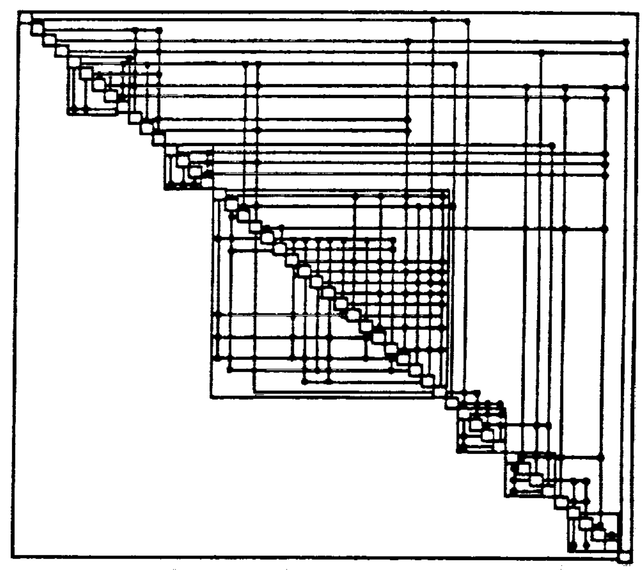


13. Schematic definition of a module.



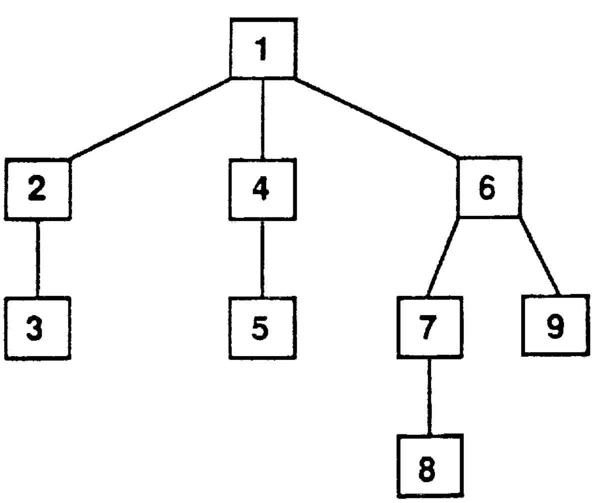
14. Example of a system: a) Graph format; b) N-square Matrix format.



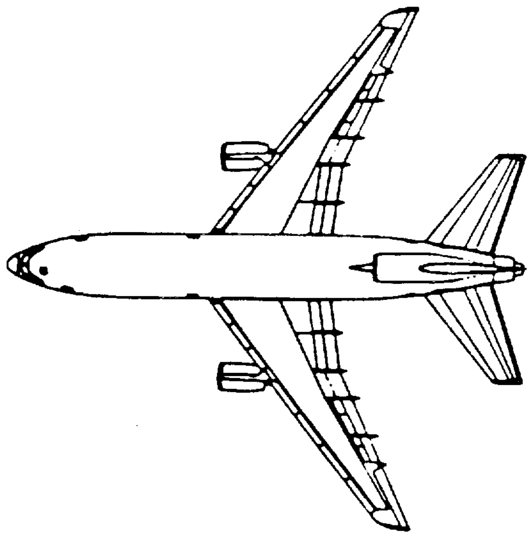


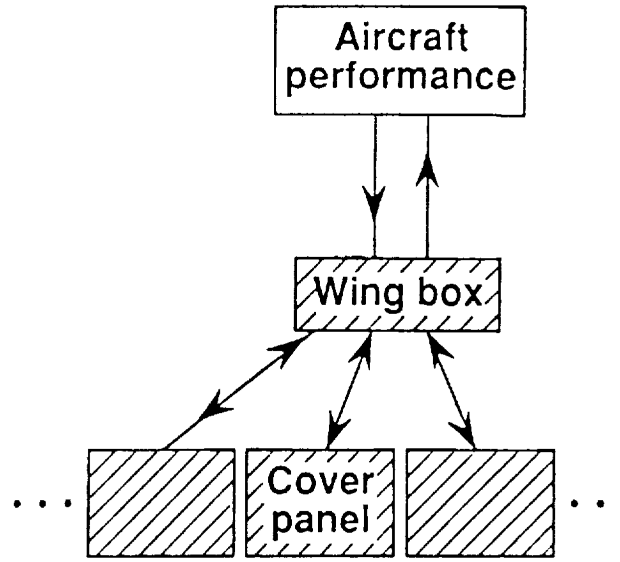
b) Execution order rearranged to reduce and cluster the feedbacks.

15. System N-square Matrix: a) Random execution order;

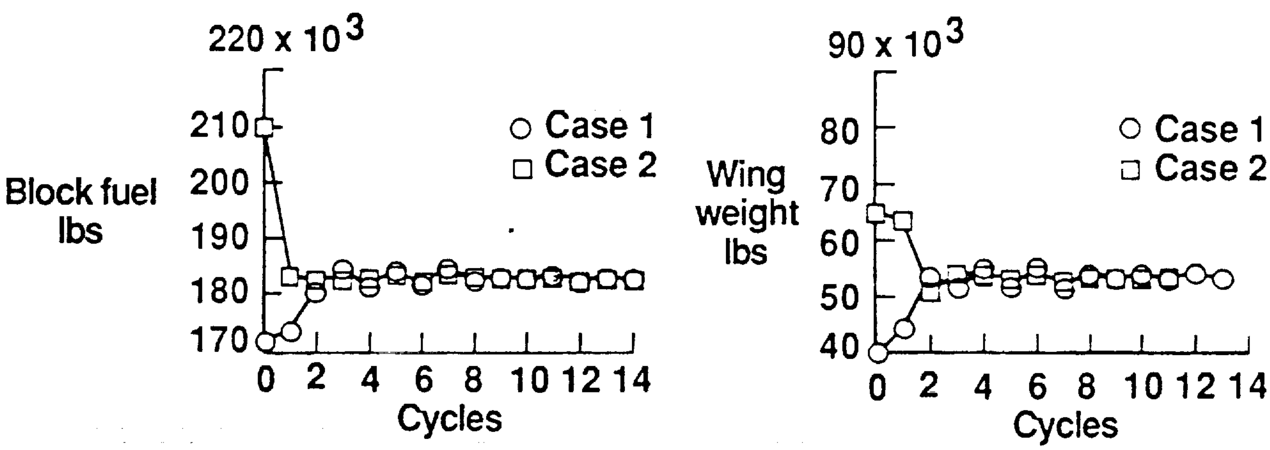


16. Hierarchic structure of clusters in a system.

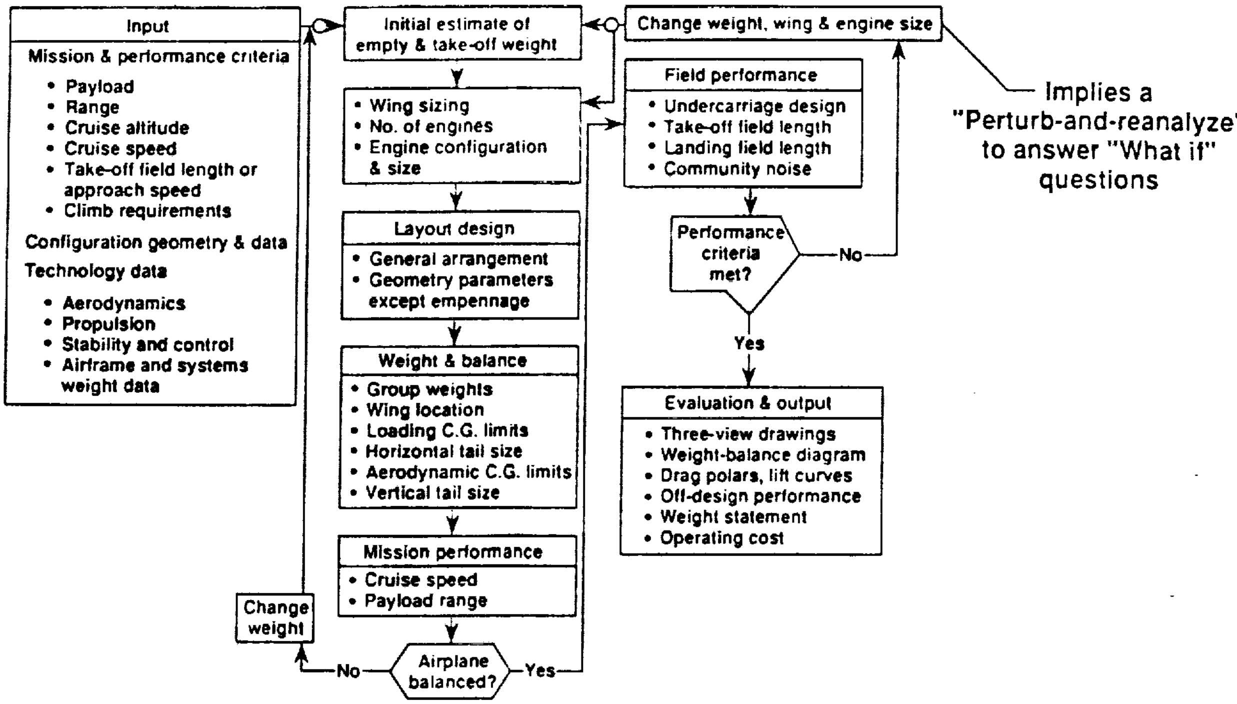




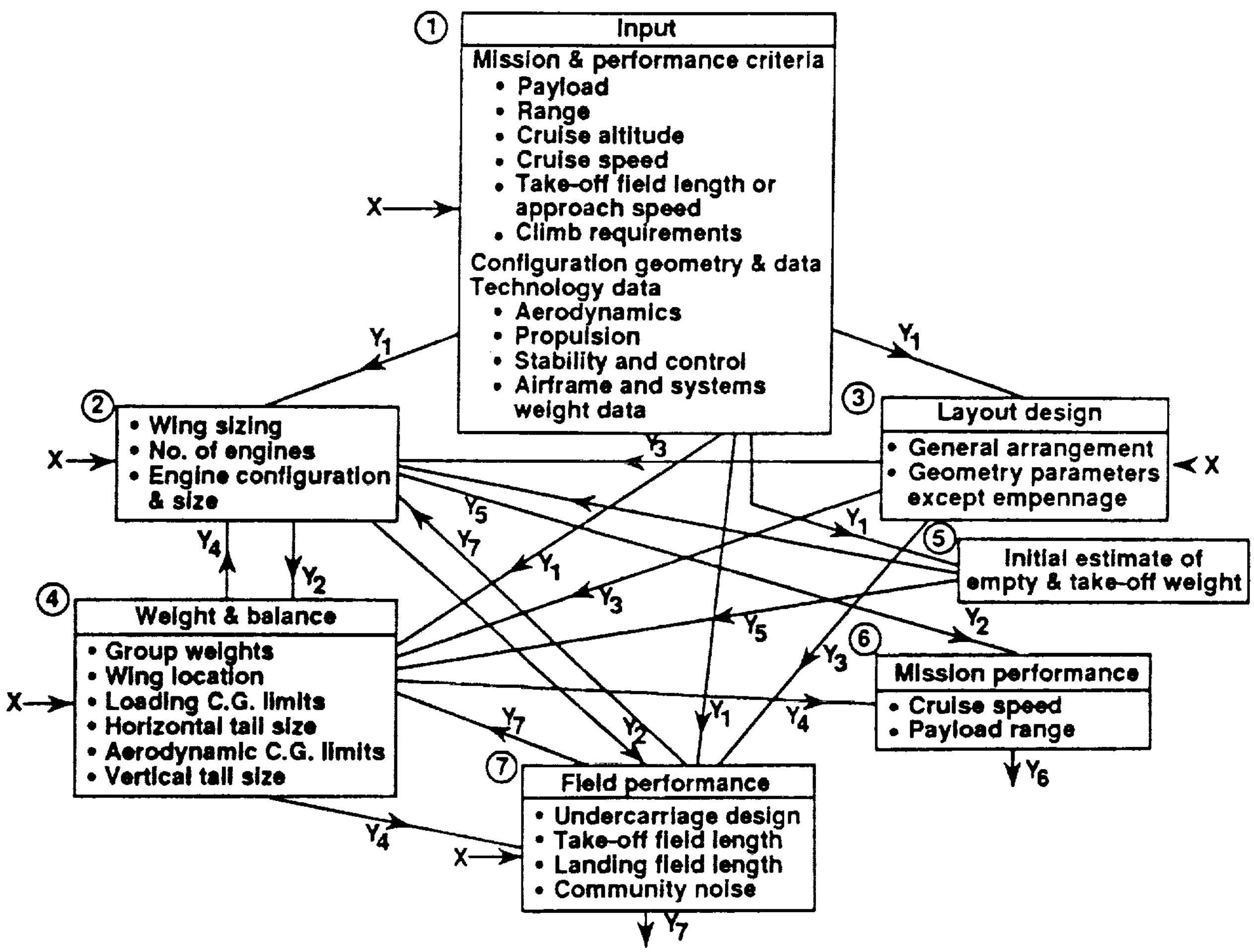
17. Optimization of a transport aircraft: a) Configuration; b) Hierarchic system of modules.



18. Sample of results from transport aircraft optimization.



19. A conventional, sequential design process for aircraft.



20. Black boxes from Fig. 19 forming a system.

- System sensitivity equations of design represented as coupled system

$$\begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 & 0 \\ -Y_{21} & I & -Y_{23} & -Y_{24} & -Y_{25} & 0 & -Y_{27} \\ -Y_{31} & 0 & I & 0 & 0 & 0 & 0 \\ -Y_{41} & -Y_{42} & -Y_{43} & I & -Y_{45} & 0 & -Y_{47} \\ -Y_{51} & 0 & 0 & 0 & I & 0 & 0 \\ 0 & -Y_{62} & 0 & 0 & -Y_{65} & I & 0 \\ -Y_{71} & -Y_{72} & -Y_{73} & -Y_{74} & 0 & 0 & I \end{bmatrix} \begin{Bmatrix} \frac{dY_1}{dX_k} \\ \frac{dY_2}{dX_k} \\ \frac{dY_3}{dX_k} \\ \frac{dY_4}{dX_k} \\ \frac{dY_5}{dX_k} \\ \frac{dY_6}{dX_k} \\ \frac{dY_7}{dX_k} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial Y_1}{\partial X_k} \\ \vdots \\ \vdots \\ \frac{\partial Y_i}{\partial X_k} \\ \vdots \\ \vdots \\ \vdots \end{Bmatrix} \cdots \begin{Bmatrix} \frac{\partial Y_1}{\partial X_L} \\ \vdots \\ \vdots \\ \frac{\partial Y_i}{\partial X_L} \\ \vdots \\ \vdots \\ \vdots \end{Bmatrix} \cdots$$

- These system derivatives answer "What if" questions regarding these variables without reanalyzing the system

21. GSE matrix for the system of Fig. 20.

Report Documentation Page

NASA
National Aeronautics and Space Administration

1. Report No. NASA TM-104074	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle A System Approach to Aircraft Optimization		5. Report Date March 1991
		6. Performing Organization Code
7. Author(s) Jaroslaw Sobieszczanski-Sobieski		8. Performing Organization Report No.
		10. Work Unit No. 505-63-50
9. Performing Organization Name and Address NASA Langley Research Center Hampton, VA 23665-5225		11. Contract or Grant No.
		13. Type of Report and Period Covered Technical Memorandum
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546-0001		14. Sponsoring Agency Code

15. Supplementary Notes

To be presented at the NATO-AGARD Structures and Materials Panel Meeting, in Bath, United Kingdom, on April 28 - May 5, 1991.

16. Abstract

Mutual couplings among the mathematical models of physical phenomena and parts of a system such as an aircraft complicate the design process because each contemplated design change may have a far reaching consequences throughout the system. This paper outlines techniques for computing these influences as system design derivatives useful for both judgmental and formal optimization purposes. The techniques facilitate decomposition of the design process into smaller, more manageable tasks and they form a methodology that can easily fit into existing engineering organizations and incorporate their design tools

17. Key Words (Suggested by Author(s)) computational infrastructure, optimization, methodology	18. Distribution Statement Unclassified - Unlimited. Subject Category 05

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassified	Unclassified	16	A03

NASA FORM 1626 OCT 86