EMDO: An Engineering Approach to Multidiscipline Design Optimization

G. N. Vanderplaats[1]
Vanderplaats Research & Development, Inc., Colorado Springs CO 80906

Design optimization technology has advanced a great deal over the last four decades so that, today, we can apply optimization to a very wide range of design tasks. Formal methods for considering multiple disciplines in the optimization process has not progressed as well due to the complexity of the task as well as the complexity of the methods proposed. The purpose here is to define a general design optimization approach that closely models traditional design methods while making maximum use of optimization. While it cannot always be demonstrated that a true optimum will result using the EMDO method, it is clear that this method will enhance the design process and lead to improved designs without dramatically changing the way engineers work.

I. Nomenclature

$F(X)$	=	System objective function
$F_i(X,Y^i)$	=	Objective function for sub-system i
$g_j(X)$	=	j-th system level inequality constraint
$g_j(X,Y^i)$	=	j-th inequality constraint for sub-system i
m	=	Number of system level constraints
m_i	=	Number of constraints for sub-system i
n	=	Number of system level design variables
n_i	=	Number of design variables for sub-system i
X	=	Vector of system level design variables
X_i^L	=	Lower bound on system level design variable i
X_i^U	=	Upper bound on system level design variable i
Y^i	=	Vector of local design variable for sub-system i
Y_i^{iL}	=	Lower bound on design variable i of sub-system i
Y_i^{iU}	=	Upper bound on design variable i of sub-system i

I. Introduction

Multidiscipline design is as old as engineering itself. Indeed, given enough time and money, engineers create excellent, if not optimum, multidiscipline systems. Design methods have evolved as they have because they work. Therefore, in an optimization environment, it makes sense to learn from these traditional methods and apply optimization as a tool to improve design quality and increase productivity.

Research in engineering design optimization has progressed greatly since its inception over 45 years ago with the classic paper by Schmit[1]. This has led to a mature technology for use in a significant percentage of design tasks, and numerous commercial optimization products are available today. However, the majority of applications are for single disciplines such as structures or airfoils.

Multidiscipline design optimization (MDO), where more than one discipline are included in the optimization process has been a goal almost from the beginning. Indeed, in 1964, Schmit and Thornton

---

[1] President, Fellow AIAA.

optimized a wing considering both aerodynamics and structures[2]. In the 1970s conceptual aircraft design was performed by including all key disciplines in the optimization problem (ref. 3 plus numerous references listed there). Because only basic aircraft parameters were designed and the analysis was based on simple physics, all design variables were considered simultaneously.

As optimization tasks grew in size and complexity, it became apparent that the direct approach of simply coupling the analyses with an optimizer was not tractable. This was due to the large number of variables involved as well as the coupling among the disciplines.

An important step in MDO was offered by Sobieszczanski-Sobieski[4] in 1982, when he recognized and formally dealt with the interactions among disciplines where more than just a few system and sub-system variables were considered.

While this method was mathematically elegant, several difficulties were encountered in it's implementation. First, it effectively required extensive chain rule differentiation which could generate poor answers due to numerical noise, particularly if functions were evaluated iteratively (functions themselves were noisy). Second, it required sensitivities of the optimum sub-systems[5]. These can be shown to be directional derivatives, especially if sensitivities with respect to design variables are needed[6]. This is easily demonstrated by a two variable function space where the optimum has two active constraints. Increasing a variable will follow one constraint while decreasing it will follow the other. Because the sensitivity is the vector product of either of these two distinct search directions with the gradient of the objective function, the optimum sensitivity is clearly not unique. Finally, and most important, the method was viewed as complicated, requiring special expertise to use. This is a key issue in gaining widespread use of optimization because today's engineers demand "turn key" solutions. Regardless, and perhaps because of these issues, reference 4 generated an extraordinary amount of research in MDO over the past twenty five years with a multitude of complimentary and competing ideas[7, 8]. This has led to the most recent method called Bi-Level Integrated System Synthesis, or "BLISS"[9].

The BLISS algorithm is described in some detail in reference 9 and need not be reproduced here. However, the basic philosophy can be simply described.

In the BLISS algorithm, the overall system is comprised of system variables and sub-system variables. For example, system variables may include overall geometry of a wing, such as aspect ratio, thickness-to-chord ratio, sweep, etc. The structures sub-system will typically include member sizing variables, while the aerodynamic sub-system may include empennage sizing for controllability, and aerodynamic shaping of the leading edge (since that surface is often not included directly in the structural optimization). The sub-systems clearly interact since the deformed shape defines input to the aerodynamics module and the resulting air loads provide input to the structures module. In past MDO algorithms, these interactions were handled via communication between sub-modules. In the BLISS algorithm, all coupling is dealt with by the system control. Various formulations are used to deal with this coupling. Important features of this method are that it separates design variables into system and sub-system variables, and it allows sub-system groups to work with considerable independence. In these respects, it more closely parallels how design is traditionally performed than do previous MDO methods.

Notwithstanding the positive aspects of the various MDO methods offered in the past 25 years, they remain rather complex in their implementation. This raises questions of how robust such methods can be and also how well they will be received by the general design community.

The goal here is to take a fresh look at Multidiscipline Design Optimization. The source of the present method is the observation that most progress in such areas as structural optimization over the past 30 years has not come from algorithmic complexity but by observing and copying what good designers have always done. Basis vectors[10] and formal approximations[11-15] are examples where substantial progress has come through careful problem formulation utilizing techniques that have long been known and understood by practicing engineers.

The method presented here can be summarized very simply: "Do what we've always done but take advantage of the immense computer power available for speed and use optimization at every step to reduce design time and improve quality." Having said that, there will be some techniques that we can use in an optimization environment to simplify the overall process and improve the probability of achieving the true optimum. Notably, we can often eliminate iterative processes in favor of adding constraints, thus using the optimizer to reduce computational time.

1) Aircraft Design

Aircraft design is perhaps the most commonly identified multidiscipline design task. Here, aerodynamics, structures, propulsion, trajectory, controls and other disciplines interact within the overall system. In the late 1960s and 1970s, numerous computer programs were created for conceptual aircraft design and many of these utilized optimization[3]. Though these efforts are seldom referenced and almost never referred to as MDO, they were indeed early MDO efforts. Some of these efforts continue even today and others are being created by various aircraft companies. The key to most of these efforts was that only a few system design variables were considered and (in most cases) the disciplines did not perform optimization at the local level.

This early work is important, not so much for the results it produced as much as for some important lessons learned. First of all, the underlying analysis must properly model the true physics or optimization will generate unrealistic designs. Second, the choice of design variables is important and third, time consuming design iterations can often be replaced by constraints.

As an example of the importance of modeling physics, Figure 1 shows the effect of aspect ratio on wing weight. Each equation plotted here was created based on historical data by a different weights engineer and each was shown to correlate well with the data used. Note that case one shows that, historically, as aspect ratio has increased, wing weights actually decreased while case two shows almost no change and the other cases show that wing weight increases with aspect ratio as expected. When the wing weight was estimated using case one, the optimizer increased the wing aspect ratio on the F-5 aircraft test case to its upper bound (set arbitrarily to 100!), clearly contrary to what is expected based on simple physics.

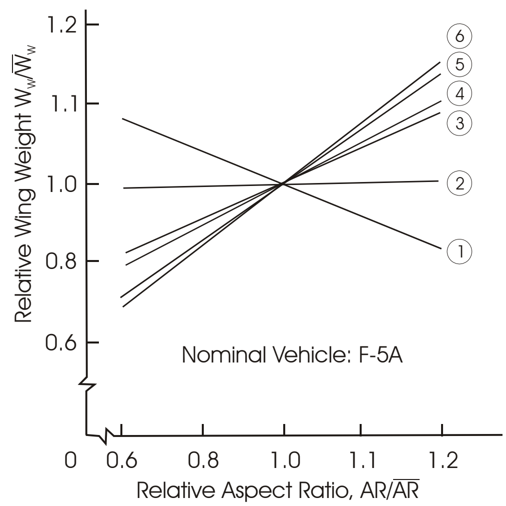

Figure 1. Wing weight Versus Aspect

Figures 2 through 4 demonstrate the importance of choosing the proper design variables. Figure 2 is an oblique flying wing where a key constraint is that the wing has sufficient volume to contain fuel and equipment. Traditionally, wing loading (gross take-off weight divided by wing area) is considered to be a basic design variable. Figure 3 shows the design space in terms of the wing thickness-to-chord ratio and wing loading. Note that the optimum is poorly defined and the design space is actually non-convex (suggesting the possibility of relative minima). Now consider

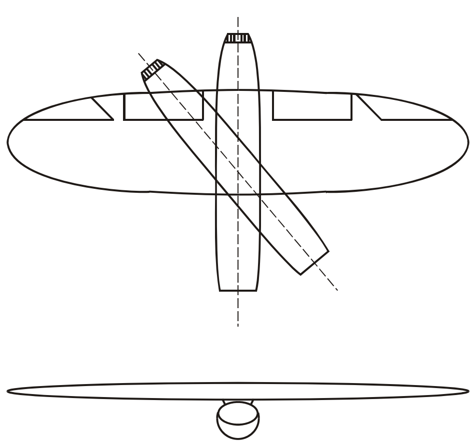

Figure 2. All Wing Remotely Piloted Vehicle

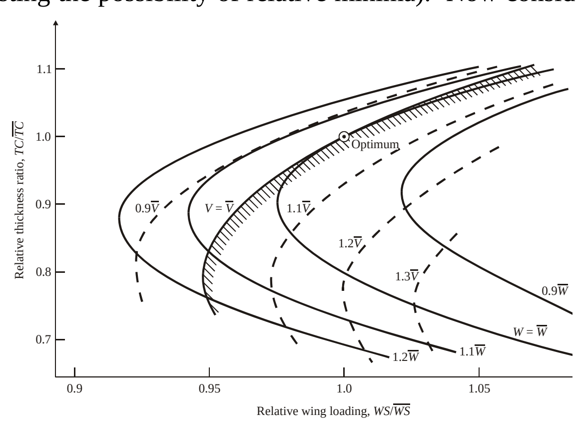

Figure 3. Design Space Using Wing Loading as a Design Variable

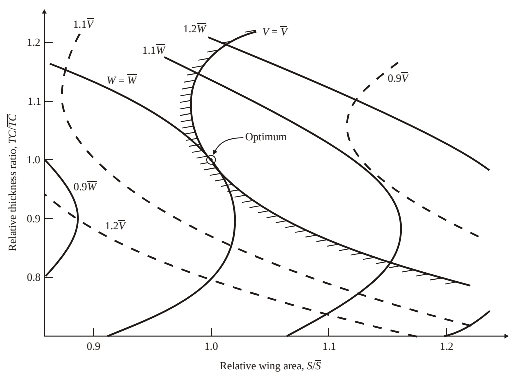

Figure 4. Design Space Using Wing Area as a Design Variable

a simple change of variables where we let the actual wing area be the design variable instead of wing loading. Figure 4 shows the design space for this case where now the design space is convex and the optimum is well defined. In each case, the optimum design is the same but in the second case it is much more easily found computationally.

Finally, consider the iterative process of calculating the gross take-off weight of an aircraft to fly a specified mission. The general design approach is;

For a specified fuselage length and diameter, maximum thrust, wing aspect ratio, wing area, wing thickness-to-chord ratio, etc.

1. Estimate the gross take-off weight.
2. Call the geometry module to determine volume and space requirements.
3. Call the trajectory module to "fly the mission" and calculate fuel used in each leg.
    a. Call aerodynamics module.
    b. Call propulsion module.
4. Add up all fuel usage for the mission.
5. Call the mass properties module to estimate all component masses.
6. Add all component masses plus fuel to provide a calculated gross take-off weight.
7. Update the estimated gross take-off weight and iterate until the calculated weight equals the estimated weight.

Of course this is a very simplistic outline of the design process but it is sufficient to consider some key points. First, the geometry module should not actually perform sizing of the fuselage to contain passengers, etc. and/or wing to contain fuel. It should calculate the needed dimensions and volumes to be compared with the dimensions and calculated volumes specified by the system design variables. Then the optimizer will adjust the specified sizes to provide the needed volumes. Secondly, if the gross weight is taken as an independent design variable, the difference between the estimated gross weight and the calculated gross weight can simply be taken as an equality constraint, eliminating the iteration of step 7 above[3] Thus, at the end of the optimization process, these two values will agree without all of the intermediate iterations. In fact, in most cases, this can be treated as an inequality constraint with the calculated weight required to be less than or equal to the gross weight and the constraint will naturally be satisfied with equality at the optimum.

From this simple discussion, it is clear that the design process can be significantly improved with the use of optimization. Here, the individual modules normally do not perform any sub-optimization tasks though that would certainly be allowed.

2) Ship Design

The ship synthesis process is very similar to that of aircraft synthesis. Here the "design spiral" shown in Fig. 5 represents the traditional synthesis process. Typical design variables are length (length between perpendiculars), width (beam), height, and prismatic and midship section coefficients. The prismatic coefficient relates the volume of the hull to the rectangular volume enclosed by the outer dimensions, and the midship section coefficient relates the cross section to that of a rectangle containing that cross section.

The design spiral is similar to the aircraft sizing process, so that at each step the appropriate information is calculated and there may be internal iterations to satisfy volume, performance, and stability requirements. The subsequent iterations through the spiral represent iterative refinements, just as in the aircraft

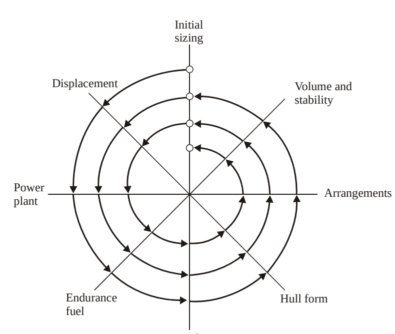

Figure 5. The Ship Design Spiral

synthesis process.

When using optimization for ship design, we remove the sizing functions from the analysis and create a series of constraints to ensure that volume, power, and stability requirements are met. Once again, in doing this we greatly simplify the analysis while transferring the design decisions to the optimizer. Typical design objectives are now to minimize ship displacement, maximize cruise speed, or minimize fuel consumption.

In reference 16 an established ship synthesis model[18] was modified to create the desired optimization capability. Table 1 gives the results of the optimization of a modern naval ship for maximum displacement.

Table 1 Ship Synthesis Results

Parameter	Initial Value	Optimum Value
Length Between Perpendiculars, m	91.40	120.20
Length/Beam Ratio	9.07	8.76
Beam/Height Ratio	3.14	3.32
Prismatic Coefficient	0.59	0.50
Midship Section Coefficient	0.75	0.77
Displacement (Tons)	2865	3512

For the initial design, several constraints were badly violated and the optimization overcame this to achieve the design shown. Reference 16 also presents examples of trade-off studies to indicate the versatility of the capability. Here again we are reminded that the analysis is based on first principles and that the results must be carefully checked by competent specialists. An interesting outcome of this study was that it produced an optimum very close to that obtained in the traditional way. The difference was that the traditional approach required multiple years and thousands of combinations of the design variables while the optimization task required under one day. Of course several months were required to convert the original software to use with optimization but after that, the combined capability could be used many times.

3) General Observations

The above cases of aircraft and ship synthesis are but two examples where multidiscipline systems are routinely designed, normally without optimization. It should be noted that optimization could be used at any point in these design processes, whether for a single discipline, a group of disciplines or for the entire system.

It is often argued that applying optimization to the individual disciplines will not produce an optimum system. That is, the optimum system is not the sum of optimum components. This has been stated so often that, today, it is almost accepted as gospel.

The standard example to prove this is that if an aerodynamicist and a structural engineer design the same wing, it will be totally different. The optimum aerodynamic design will be thin, highly swept and high aspect ratio while the optimum structural design will be thick, unswept and short. However, if the thickness, sweep and aspect ratio are treated as system design variables and the range (for example) of the aircraft is maximized, the optimum trade-off between aerodynamic efficiency and structural mass will automatically be accounted for. Therefore, the key issue is that of choosing system level design variables and determining what subsystem variables the disciplines are allowed to change. In this case, the aerodynamic shape would be defined by system variables while internal structural dimensions of wing skin, spar and rib dimensions may be determined as a sub-optimization task. Of course, there remains interaction between aerodynamic loads and structural deformations but these may be dealt with in the traditional iterative fashion.

Indeed, if all variables for all disciplines could be designed simultaneously as system variables, we should find the true optimum. Therefore, the issue becomes that of separating system variables that generate a true optimum from sub-system variables that may be designed independently but are implicit functions of the system variables.

Even within a given discipline, we can argue that if structural optimization is applied to a single rib at a time, a wing made up of these ribs will not be optimum[18]. This is because force redistribution alters the loads on the ribs. Again, if the entire wing is designed, or at least the entire structural box is designed, this issue becomes irrelevant. Because we can efficiently deal with thousands of design variables (millions of design variables in the case of topology optimization) and constraints in modern structural optimization, such an approach is technically straightforward. Alternatively, even if a single rib is considered at a time, optimization may be iteratively performed if the design changes in a single iteration are limited to perhaps 20 percent. This approach may not be the most elegant but it represents good engineering practice and makes good use of the power of optimization.

Now the issue becomes, "How can we create a general design environment for systems made up of multiple disciplines and making maximum use of optimization?" We will accept the fact that the result may not be a globally optimum system if we can efficiently design a better system than before. In other words, the goal is not to be perfect, just to be better than the competition.

II. General Design Synthesis

The key to the present algorithm is to use optimization at every opportunity. However, if some disciplines do not provide "optimum" solutions for their part of the system, information traditionally provided by the discipline is still used in the overall optimization process.

A fundamental requirement is that we must define a set of system variables, **X**, that are common across all disciplines. Also, we must define the system objective function and system constraints. Multiple objectives are allowed. Thus, we have the standard optimization task defined as;

$$\text{Minimize } F(X) \tag{1}$$

Subject to;

$$g_j(X) \le 0 \quad j = 1, m \tag{2}$$

$$X_i^L \le X_i \le X_i^U \quad i = 1, n \tag{3}$$

Equality constraints may also be included but are omitted here for brevity.

Now, whenever the optimizer requires the values of the objective and constraint functions, one or many discipline programs can be called to evaluate these. The input to the disciplines is the system design variables, ***X***, and the output is the discipline's optimum design variables, objective and constraint values along with the discipline's contribution to the system objective and constraints (e.g. structural component mass or fuel mass). Thus for discipline *i*, we have the same form of the optimization task;

$$\text{Minimize } F_i(X, Y^i) \tag{4}$$

Subject to;

$$g_j(X, Y^i) \le 0 \quad j = 1, m_i \tag{5}$$

$$Y_i^{iL} \le Y_i \le Y_i^{iU} \quad i = 1, n_i \tag{6}$$

The local objective function, F_i, is a function of the discipline's choosing, Y^i is the set of local variables and $g_j(X,Y^i)$ are local constraints. Figure 6 shows the interactions between the system and sub-systems.

For example, the propulsion discipline may maximize efficiency at various operating conditions with limits on noise and pollutants. The engine mass may be a result of this sub-optimization and it is returned to the system where the overall objective may include engine mass as well as other masses together with fuel to fly a mission.

The discipline may call further subsystems without loss of generality. For example, an engine sub-system may call modules for pumps, engine controls, fan design, etc. Figure 7 shows the repetitive nature of this process.

Also, the discipline may not actually perform optimization. If the discipline or sub-discipline involves pump design, the result returned to the system may come from a previously designed set of pumps.

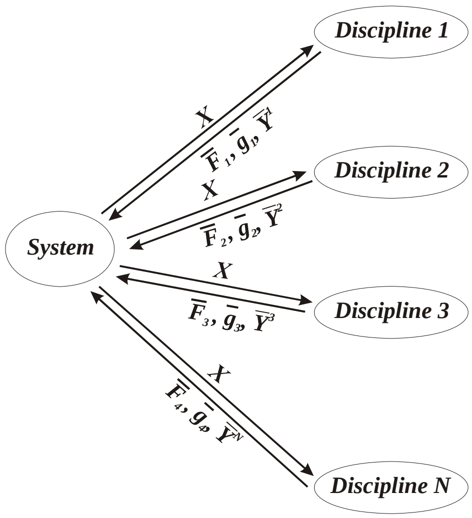

Figure 6. System and Discipline

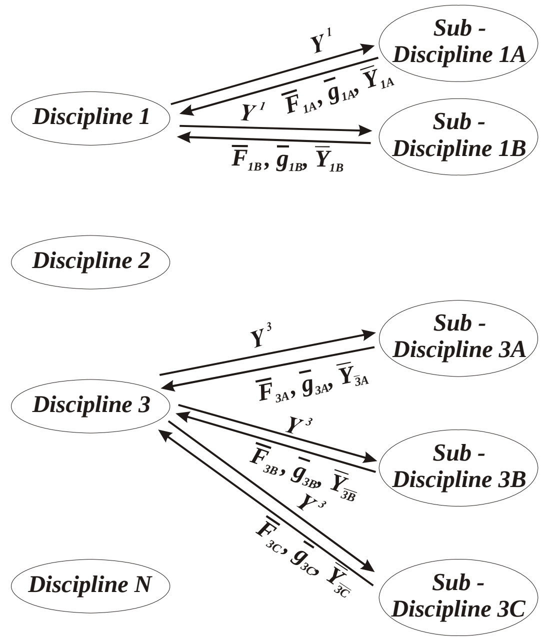

Figure 7. Discipline to Sub-Discipline Interaction

Alternatively, it may consist of response surface approximations to previously optimized pumps, where a numerous pumps were optimized for different combinations of system design variables. This may all be done in advance or in real time using a single or many computers or processors. In other words, the discipline is free to solve its problem however it chooses. It is only important that the results be consistent. That is, if the same set of system design variables are provided to the discipline twice, the discipline must return the same responses.

Implementation of this algorithm is very straightforward using modern optimization software. Figure 8 shows the script for conceptual aircraft design using the VisualDOC/VisualScript software[20]. The internal looping is to perform the trajectory portion where both aerodynamics and propulsion modules are called. Here, it is assumed that an existing engine is used and the aerodynamics is calculated for the geometry defined by the system variables. In the mass properties module, simple analytical data can be used or structural optimization software may be called to optimize the structure for the proposed configuration. If it is desired to design the propulsion system as part of the overall process, the propulsion design module would be called before calling the trajectory module to design the engines for the maximum thrust specified by the system variables.

The process outlined here is straightforward and imposes minimal disruption to existing design processes while making maximum use of optimization. While it may be argued that this will not guarantee a true system optimum, examples of such cases are difficult to create. Indeed, such cases usually indicate that coupling variables need to be moved from sub-systems to the system level.

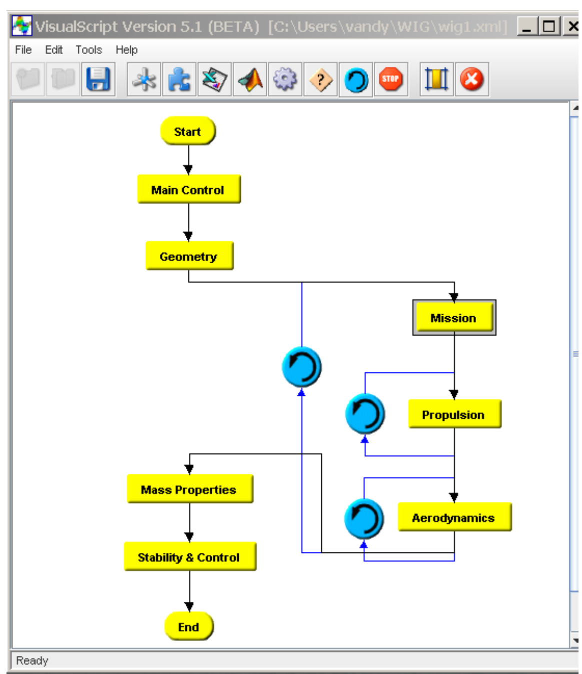

Figure 8. Aircraft Synthesis

III. Examples

A. Aircraft Wing

Figure 9 shows a simplified aircraft wing. The objective is to maximize the aircraft range (Breguet range factor) by changing the thickness to chord ratio and aspect ratio of the wing and to minimize the wing structural mass[20]. The thickness to chord ratio and aspect ratio were treated as system variables in VisualDOC and the skins, spars and ribs were optimized as a sub-problem using the GENESIS structural optimization program[21]. For each proposed set of system variables, the structural optimization module was called. Here, there was an iterative loop between aerodynamics (using simplified strip theory) and structural analysis to generate a set of loads and consistent displacement pattern as shown in Figure 10. The EMDO process increased the range by 25 percent while changing the thickness to chord ratio from 0.12 to 0.14 and the aspect ratio from 6.86 to 5.88.

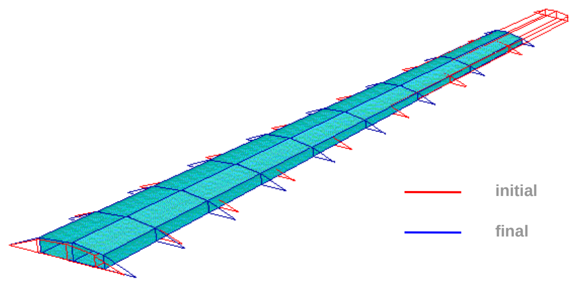

Figure 9. Simplified Aircraft Wing

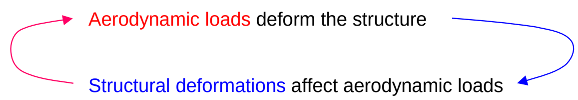

Figure 10. Aerodynamics/Structures Interactions

4) Thermal-Structural Beam

Consider the simple steel cantilevered beam shown in Figure 11 which is 10m in length. The beam is subject to thermal loads with a base temperature of 1400°K and the air temperature of 400°K. The three independent static loads are $P_1=P_2=P_3=1000$N. The objective is to minimize mass with a flux greater than 5 kw, a maximum axial tip displacement of 25 mm and a maximum stress under each of the static load cases of 100 Mpa. This problem was solved using the modified method of feasible directions in VisualDOC. [20]

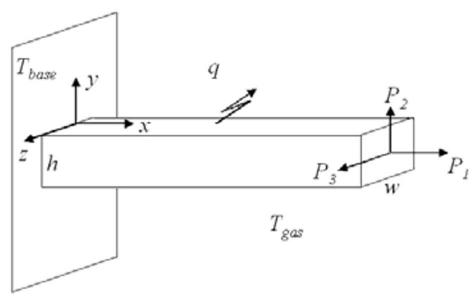

Figure 11. Thermal-Structural Beam

Figure 12 shows the problem setup where the thermal analysis is an iterative process that is required to converge to a tolerance of less than one percent between cycles with a maximum of 100 cycles. The purpose of this example is to demonstrate that such iterative loops can often be eliminated using optimization.

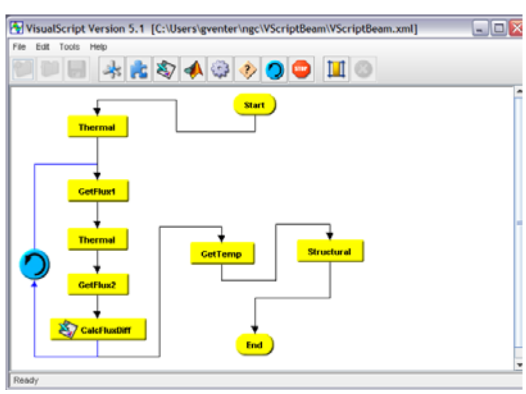

Figure 12. Program Flow

Table 2 gives the optimization results for two cases. In the first, the thermal problem was completely converged at each analysis as shown by the loop in Figure 12. In the second case, the maximum number of iterations in this loop was set to one (no convergence). Each time the thermal analysis was called, the results from the last call were used as input. Without the loop, a slightly better design was achieved using about the same number of function evaluations and optimization iterations. The two approaches gave a quite different design due to the "flatness" of the design space.

Table 2 Cantilevered beam results

Parameter	Initial Design	Results	
		Full Convergence	No Convergence
Width	1.000 m	0.0656 m	0.0325 m
Height	1.000 m	1.631 m	3.2290 m
Unconverged Flux	13.82 kw		5.03 kw
Converged Flux	29.36 kw	5.01 kw	5.03 kw
Mass	78350 kg	8386 kg	8299 kg

The reason this works is that optimization itself is iterative making smaller and smaller chances in the design variables as convergence is approached. Therefore, the thermal analysis converges naturally. In practice, the inner loop may be continued to convergence for the initial design in order to provide a higher level of confidence in the process. This same approach may be applied to the aircraft wing example above. It may be noted that if this does not work, the design may be ill-conditioned and, therefore, less reliable.

5) Upper Stage Rocket Engine

Figure 13 shows a typical rocket engine. Design of such systems include meanline pump design, meanline turbine design, combustion models, thermal/structural interactions, system power balance and others. While design of such systems is confidential due to proprietary information or classification, it can be said that the EMDO approach is being applied to this class of designs. This resulted in the following quote.[22]

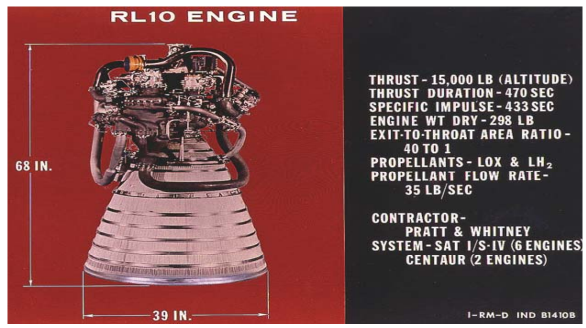

Figure 13. Rocket Engine

"Lately our program for doing optimization of this upper stage rocket engine has been showing much success. We now can routinely run overnight optimizations that show a much higher degree of fidelity than anything we have seen before. The system optimizations are pretty interesting too. For example the results are showing that overall engine weight can be minimized by de-tuning some components (and making them heavier) so that other components can be made smaller and lighter. This was something that was not previously done in this field where everything is made to push performance."

IV. Conclusions

A engineering approach to multidiscipline design optimization (EMDO) has been presented and demonstrated. The main concept is to perform design much as we always have but use optimization at every opportunity. A key issue in effective application of the method is separating design variables between system and sub-system level variables. By keeping any "coupling" variables at the system level, the likelihood of finding the true optimum is improved. With the EMDO method, the design groups maintain autonomy and, in fact, may not even use optimization. Those that do use optimization provide improved information to the system level decision maker. The EMDO method offers considerable flexibility in performing design in a parallel or distributed environment. Finally, sub-system optimization may be performed separately and provided to the system level design as a set of response surfaces that provide optimum sub-system information in terms of system level variables.

The goal of this method is to encourage more widespread use of optimization without the need for theoretical training or complex programming.

References

[1]Schmit, L.A.: Structural Design by Systematic Synthesis, Proc. 2nd Conference on Electronic Computation, ASCE, New York, 1960, pp. 105-122.

[2]Schmit, L. A., and W. A. Thornton: "Synthesis of an Airfoil at Supersonic Mach Number," NASA CR-144, January 1965.

[3]Vanderplaats, G. N., "Automated Optimization Techniques for Aircraft Synthesis," Proc. AIAA Aircraft Systems and Technology Meeting, Dallas, Texas, Sept. 27-29, 1976. AIAA Paper No. 76-909.

[4]Sobieszczanski-Sobieski, J., "A linear Decomposition Method for Large Optimization Problems - Blueprint for Development," NASA Technical Memorandum 83,248, February 1982.

[5]Sobieszczanski-Sobieski, J., Barthelemy, J. F. and Riley, K. M., "Sensitivity of Optimum Solutions to Problem Parameters," AIAA Journal, Vol. 20, Sept. 1982, pp. 1291-1299.

[6]Vanderplaats, G. N. and Yoshida, N., "Efficient Calculation of Optimum Design Sensitivity," AIAA Journal, Vol. 23, No. 11, Nov. 1985, pp 1798-1803.

[7]Sobieszczanski-Sobieski, J.: Multidisciplinary Design Optimization; an Emerging New Engineering Discipline; in Advances in Structural Optimization, Jose Herskovits (Ed.); Kluwer Academic Publ. 1995; pp.483-496.

[8]Sobieszczanski-Sobieski, J., and Haftka, R.T.: Multidisciplinary Design Optimization: Survey of Recent Developments; Structural Optimization, Vol.14, No.1, Jan. 1997, pp.1-23.

[9]Sobieszczanski-Sobieski, J., Altus, T. D., Phillips, M. and Sandusky, R., "Bi-Level Integrated System Synthesis (BLISS) for Concurrent and Distributed Processing," Proc. 9th AIAA/ ISSMO Symposium on Multidisciplinary Analysis and Optimization, Atlanta, GA, Sept. 4-6, 2002, AIAA Paper 2002-5409.

[10]Pickett, R. M., Jr., M. F. Rubinstein and R. B. Nelson: Automated Structural Synthesis Using a Reduced Number of Design Coordinates, AIAA Journal, Vol. 11, No. 4, 1973, pp. 489-494.

[11]Schmit, L.A. and Farshi, B., "Some Approximation Concepts for Efficient Structural Synthesis," AIAA Journal, Vol. 12, May 1974, pp. 692-699.

[12]Schmit, L.A., and H. Miura: Approximation Concepts for Efficient Structural Synthesis, NASA CR-2552, l976.

[13]Vanderplaats, G. N. and Salajegheh, E., "An Efficient Approximation Technique for Frequency Constraints in Frame Optimization," International Journal for Numerical Methods, Vol. 26, 1988, pp 1057-1069.

[14]Vanderplaats, G.N. and Salajegheh, E., "A New Approximation Method for Stress Constraints in Structural Synthesis," AIAA Journal, Vol. 27, March 1989, pp. 352-358.

[15]Canfield, R.A., "An Approximation Function for Frequency Constrained Structural Optimization," Proceedings of the Second NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, AIAA, Hampton, VA, Sept. 28-30, 1988, pp. 937-953.

[16]Jenkins, J. L.: Application of Optimization Techniques to Naval Surface Combatant Ship Synthesis, M.S. thesis, Naval Postgraduate School, Monterey, Cal., October 1982.

[17]Reed, M. R.: Ship Synthesis Model for Naval Surface Ships, M.S. thesis, Massachusetts Institute of Technology, May 1976.

[18]Topology Optimisation of Aircraft Wingbox Ribs, L. Krog and A. Tucker, Airbus, Bristol, Great Britain; M. Kemp, B. Richard and W. John, Altair Engineering, Coventry, Great Britain, AIAA-2004-4481

[19]VisualDOC Users Manual, Version 5.0, Vanderplaats Research & Development, Inc., Colorado Springs, CO, 2005.

[20]Garcelon, J., Balabanov, V. and Sobieski, J., "Multidisciplinary Optimization of a Transport Aircraft Wing using VisualDOC", presented at the 40th AIAA/ASME/ASCE/AHS/ ASC Structures, Structural Dynamics and Materials Conference, April 1999, St. Louis, MO.

[21]GENESIS Structural Analysis and Optimization Users Manual, Version 8.0, Vanderplaats Research & Development, Inc., Colorado Springs, CO, 2005.

[22]Curtis Johnson, Sierra Engineering, April 5, 2006, Private communication.