MDO TEST SUITE AT NASA LANGLEY RESEARCH CENTER

Sharon L. Padula, Natalia Alexandrov and Lawrence L. Green
NASA Langley Research Center
Hampton, VA

Sixth AIAA/NASA/ISSMO Symposium on
Multidisciplinary Analysis and Optimization

AIAA Paper No. 96-4028
Bellevue, Washington
September 4-6, 1996

MDO TEST SUITE AT NASA LANGLEY RESEARCH CENTER

S. L. Padula*, N. Alexandrov†, L. L. Green*
NASA Langley Research Center
Hampton, VA

Abstract

The NASA Langley Research Center supports a wide variety of multidisciplinary design optimization (MDO) research and requires a set of standard MDO test problems for evaluating and comparing the products of this research. This paper proposes a World-Wide-Web-based test suite for collecting, distributing, and maintaining the standard test problems. A prototype suite of 10 test problems, including written problem descriptions, benchmark solution methods, sample input and output files, and source code is available from the NASA Langley Research Center internet site. Here, design of the MDO test suite is discussed; typical test problems are described; and sample web pages are illustrated.

Introduction

Multidisciplinary design optimization (MDO) research has blossomed in the 12 years since the first Multidisciplinary Analysis and Optimization symposium was held in April 1984 at NASA Langley Research Center. However, the current usefulness and future development of MDO methodology is hampered by the lack of a commonly accepted and readily available suite of test problems. This fundamental hindrance to research has been noted many times (e.g., refs. 1-3).

In current practice, each researcher must develop an individual set of test problems. This practice is unsatisfactory for three reasons. First, the development of test problems by each researcher wastes valuable research time that could ultimately result in new MDO techniques. Often, researchers use trivial or degenerate test problems rather than investing the time necessary to construct representative problems. Second, test problems developed by one researcher cannot generally be reused by other researchers because of the complexity of MDO problems, which makes them difficult to describe unambiguously in published reports. Moreover, test problems often rely on several disciplinary analysis codes that may not be readily available to other researchers because the codes are lost, changed, undocumented, proprietary or prohibitively expensive. Third, when each researcher formulates a unique set of test problems, then no common ground exists for communicating results or for comparing one method with another. In fact, two authors have considered the same set of MDO methods and have drawn opposite conclusions regarding their utility (e.g., refs. 4 and 5).

Ideally, a standard set of test problems should be made available to all researchers. This test suite should include a variety of problems that range from simple to challenging. Each entry in the test suite should include an accurate description of the problem and provide a benchmark solution method. Ideally, the benchmark solution should include the sample input and output and all computer codes necessary to recreate the sample output. This benchmark solution need not be efficient or elegant; its function is to provide a common

---

* Research Engineer, Multidisciplinary Optimization Branch, Senior Member AIAA
† Research Scientist, Multidisciplinary Optimization Branch, Member AIAA

reference point to which all other methods can be compared. In the same way that wind-tunnel data for lift generated by an NACA 0012 airfoil at various wind speeds and angles of attack is used to judge the merit of computational fluid dynamics codes, these benchmark solutions would provide data regarding the cost of convergence for a variety of computer architectures, convergence tolerances, and problem parameters. Because the source code is available, the researcher can execute both the benchmark and the modified code on the same computer and can collect the appropriate measures of effectiveness (e.g., processing time or number of iterations). Moreover, other researchers can verify the results with their own computers and metrics.

Researchers in the Multidisciplinary Optimization Branch at NASA Langley Research Center have constructed a prototype MDO test suite for evaluation by the MDO community. The test suite is available to anyone with access the World Wide Web (WWW) using a browser such as NCSA Mosaic™ or Netscape™.‡ This paper defines and gives examples of MDO test problems, shows sample web pages, provides an overview of the current set of test problems in the test suite, and discusses plans for its further development.

MDO Test Suite

The purpose of the test suite is to support MDO research by providing a large variety of MDO test problems to the research community. These problems can be obtained or adapted from problems that appear in the literature or can be submitted by researchers in industry, academia, and government. Each problem includes computer code and documentation in sufficient detail so that researchers can recreate the benchmark solution and modify the code to test new MDO methods. The test suite contains some simplified MDO problems that are suitable for debugging new methods or for teaching MDO short courses. On the other hand, the suite contains difficult or even intractable problems for challenging the experts and for motivating the development of new techniques.

The contents of each entry in the test suite are similar regardless of the difficulty of the problem. Thus, a potential user can scan the available problems and select those that fit specific needs. A typical test suite entry has the following format.

I. Description
  - A. Background information
  - B. References
  - C. Problem statement
  - D. Benchmark solution strategy
  - E. Computer code overview

II. Source codes and scripts (if available)
  - A. Conventional solution
  - B. Benchmark MDO solution
  - C. Auxiliary subroutines

III. Feedback
  - A. Computational experience
  - B. Best known solutions
  - C. Request forms

‡ The use of trademarks or names of manufacturers in this report is for accurate reporting and does not constitute an official endorsement, either expressed or implied, of such products or manufacturers by the National Aeronautics and Space Administration.

The description section is brief. The creators assume that the researchers who use a test problem will examine the references for technical details and will examine the code for implementation details.

A clear distinction is made between the problem statement and the benchmark solution. The problem statement explains the goal of the optimization procedure and specifies the quantities that can be changed; any constraints on the range of those changes are also provided. The benchmark solution describes one possible method for solving the problem. This approach may not be the most straightforward, the most elegant or the most efficient method. It is simply one possible solution method, which becomes the standard to which all competing solution methods can be compared.

The source codes and sample input and output files demonstrate one implementation of the benchmark solution method. To the extent possible, each code used is well documented, modular, and machine independent which is necessary because the code is a part of the test suite problem documentation.

The intent is for the test suite to be a collection of test problems that includes computer source code and sample results. If auxiliary subroutines (e.g., optimization or mathematical library routines) are required, then a link to other web sites is provided. If the disciplinary source code has access restrictions, then a form is provided to request the code. Other forms are provided for comments, corrections, and experiences. The hope is that users of the test suite will report their computational experience so that several solution methods can be compared in a meaningful way. Feedback from users will be collected and posted in a timely manner.

Sample Test Problems

This section describes the classification of test problems and provides an example from each of the three available classes. The class II example will be familiar to many in the MDO community. This example is described first so that the other two examples can be contrasted to it.

The WWW home page for the test suite points to the three classes of test problems. Class I test problems have simple analyses, analytic derivatives, and known solutions. In many cases, these problems have conventional solution methods that are more efficient than the MDO solution methods. The problems have been formulated to simulate the characteristics observed in practical engineering design problems. Class II test problems are simplified examples of engineering test problems. They include analysis codes for more than one engineering discipline and have solutions with a physical interpretation. In many cases, these problems have multiple solutions that are essentially equivalent (e.g., manufacturing tolerances and measurement accuracy make the differences undetectable). Minor differences in compilers or machine operating systems can affect the performance and the convergence history of these test problems. Class III test problems are examples of challenging engineering design problems. They include analysis codes for more than one discipline and have solutions with both practical and physical interpretations. In some cases, these problems require analysis codes or input data files which are large and/or protected. NASA will release the codes by request on a case-by-case basis. Class III test problems have many design variables and constraints and are computationally expensive which distinguishes them from class II problems.

Electronic Packaging Problem

The first example of a test suite problem is the electronic packaging problem (e.g., ref. 6). The goal of this problem is to minimize the size of a circuit board within certain performance constraints. It has two subsystems: thermal design and electronic circuit design. The eight design variables control the geometry of the heat dissipating package and the resistance characteristics of the circuit.

The benchmark solution method for this problem assumes that the thermal and electronic analyses are separate codes and that the optimizer will be tied directly to this coupled system. Figure 1 shows one way to diagram this solution method. The analysis step involves iteration between thermal and electronic analysis blocks until the outputs y_{TD} and y_{CD} converge to an equilibrium state. The optimization step involves of evaluation of the analysis for many different values of the design variables x until the objective function f is minimized and the constraint functions g are satisfied.

The electronic packaging problem is considered a class II problem. It has only eight continuous design variables and two subsystems, and the objective and constraint functions are computationally inexpensive. However, a realistic thermal and electronic circuit design could clearly be considered a challenging MDO problem.

Heart Dipole Problem

The second sample test suite problem is the heart dipole problem (e.g., ref. 7). As explained in the appendix, this problem has eight design variables, two subsystems, and a benchmark solution method which is very similar to the electronic packaging problem. (See fig. 2.) Furthermore, the minimum value of the objective function is known and the difficulty in finding the solution can be varied from easy to difficult by changing the values of the fixed parameters.

The heart dipole is considered a class I problem. This problem has no natural subsystems and no engineering content and can be stated as a set of eight nonlinear equations in eight unknowns that is solved to interpret the measurements of a heart-monitoring device. The benchmark solution method was created by arbitrarily combining four equations into an objective function f and dividing the remaining four equations into two coupled subsystems.

High-Speed Civil Transport Problem

The third sample test suite problem is the High-Speed Civil Transport (HSCT) design (e.g., ref. 8). This problem has 44 design variables and 300 constraints. The goal is to minimize aircraft weight by changing the wing internal structure. The design problem combines two disciplines: structures and aerodynamics. Dummy routines are included as placeholders for additional disciplines such as performance, flutter, and controls. The existing disciplines are coupled because the aerodynamic analysis depends on structural deformation and the structural analysis depends on aerodynamic loads. The dummy routine that represents performance depends on the structural weight and on the lift and drag of the deformed wing.

The benchmark solution to the HSCT design problem is described in reference 8. The problem is large and complex; as a result, a database manager is required to preserve the convergence history and to service the numerous analysis programs. The problem is computationally expensive; thus, a set of executive script files is required to allow the analysis to run in parallel on a network of different workstations. Finally, the optimization problem is challenging, so that advanced MDO techniques such as generalized sensitivity equations (GSE) or simultaneous analysis and design (e.g., ref. 9) are recommended. The benchmark solution method employs the GSE technique.

This sample problem is considered a class III problem. An HSCT design approaches the size and complexity of the MDO problems that are required by industry. Although the benchmark solution method uses many unjustified assumptions to simplify the process and therefore is not a practical design tool, this solution method illustrates the data flow and computational techniques that are required in practical tools.

Sample Web Pages

This section discusses the reasons for implementing the test suite as a series of WWW pages. An overview of the organization of the website is given and several sample pages are shown. Finally, instructions for accessing the test suite are provided.

The advantages of using the WWW to distribute the MDO test suite are many. The WWW is accessible to most agencies, companies, and universities on a 24-hours basis. The material can be used equally well to provide a quick summary of MDO concerns or a detailed study of a given test problem. Unlike other publishing options, errors that are identified can be corrected quickly and new material can be added as soon as it becomes available. Furthermore, the documentation, code, and sample results can be archived in a single location. Finally, feedback from the users can be encouraged and easily collected.

The MDO test suite website contains a top-level list of all test problems that is organized by class. This list contains hyperlinks to the top-level page for each problem. For example, figure 3 illustrates the top-level page for the heart dipole problem. This page is linked to other pages that contain more detailed descriptions, results, and even FORTRAN source code. Any figures, text or code that might be included in a user's manual can also be included on the web pages. The primary advantage of hypertext is that the user can determine the level of detail necessary and can proceed accordingly.

Access to the test suite is already available through the Internet. A web browser can be used to locate the Multidisciplinary Optimization Branch home page at the universal resource locator (URL) http://fmad-www.larc.nasa.gov/mdob/. Alternatively, an Internet search engine can be used to search for "MDO." Both the MDO Branch and the AIAA MDO technical committee intend to maintain links to the test suite.

Descriptions of each problem can be read, printed, or saved as a file on the user's computer. Similarly, the source code can be accessed and downloaded to a file (fig. 4). Users can communicate with the test suite developers via e-mail or a WWW form.

Submission of new test problems is relatively easy. The hypertext markup language (HTML) source code from any of the current test problems can be downloaded and used as a template for preparing a description of the new problem. After the modified pages have been verified on the user's local machine, the test problem can be submitted via e-mail to the test suite developers. Alternatively, the modified pages can be installed temporarily on the user's own web server and the address (i.e., URL) can be communicated to the test suite group. Unique test problems that are adequately documented, including source code and supporting references, will be assigned a class and given a problem number. Test problems that are incompletely documented or that demonstrate a new approach to an existing test problem can be included in one or more of the computational experience sections of the current test suite.

Status and Future Plans

The test suite has been operational since July 1, 1996 and currently includes 10 complete test problems. Additional potential test problems have been identified (see tables below) and are scheduled for completion in the near future. These problems range in difficulty from simple to quite challenging and complicated. The benchmark solutions illustrate the use of many types of MDO techniques, such as multilevel decomposition, global sensitivity analysis, knowledge-based design, integer programming, and various forms of approximate analysis. Eventually, the test suite will include challenging MDO problems for which all existing techniques are inadequate.

The developers expect this prototype test suite to prove quite useful and to grow. The MDO Branch will encourage this growth by requiring a test suite entry as a deliverable on future contracts as well as an item on staff-member performance plans. Both the AIAA MDO technical committee and ISSMO benchmarking subcommittee have offered to advertise the test suite and to solicit input. Eventually, we believe that researchers will see the test suite as a publication option and will submit test problems at the same time that they submit journal articles. Conceivably, journal editors and reviewers could insist on the examination of a test suite entry before an article on MDO methods could be published.

Several test suite issues have not yet been adequately addressed. These issues include submission standards, feedback mechanisms, and maintenance of the test suite. A proper balance needs to be established between the need for high quality and the need for timely dissemination of information.

Obviously, standards are necessary so that test problems can be successfully run and modified. The existing web pages illustrate standards concerning the amount of information required and the format for its presentation. The test suite developers have produced a set of coding guidelines that address issues such as readability and transportability of the source code. Finally, the developers will use an HTML checker such as Weblint (see http://www.unipress.com/weblint/) to verify that web pages use correct syntax and avoid common mistakes.

Similarly, feedback mechanisms are essential. The current feedback mechanism is based on e-mail and manual changes to the test suite. As the prototype test suite is more heavily utilized, a more automated approach to user feedback will be necessary. The developers are currently experimenting with a guest-book approach. With this approach, anyone can post a message, which becomes immediately available to all users. For example, a message could be used to suggest improvements to an existing test problem or to report a particular computational experience. Rather than write an extended message, the user can provide a URL for another web site for more information. The current plan is to install a guest-book as part of each test problem and another guest-book for general comments. This plan will be reevaluated after the test suite has been operational for 1 year.

Finally, maintenance issues are critical. Benchmark solution methods require that a code be the same for all researchers. However, if errors are detected in a code, then corrective action is required. If a code is modified, then some method of configuration control must be implemented. Maintenance issues can become more complicated when web pages are distributed over a large number of servers rather than collected at a single site; on the other hand, corrections made by a maintaining organization rather than by the originator also can be problematic. The current plan is to maintain all official test problems on NASA Langley Research Center computers. The test suite developers will verify that a source code can be downloaded and that it can be compiled and run on NASA workstations. Any shortcomings in a code or modifications that are required for use on other computers will be addressed in the computational experience section of each test problem. As with the feedback mechanism, this maintenance plan will be reevaluated after 1 year.

Concluding Remarks

The multidisciplinary optimization test suite is a mechanism for improving communication within the MDO research community. The test suite provides a set of standard test problems for use in comparing, evaluating, and categorizing codes and algorithms that are developed via MDO research. A prototype test suite, developed by NASA Langley Research Center, has been constructed and is available through the World Wide Web. Currently, the test suite provides a range of test problems that can be used to instruct students, test new methods, and challenge the experts. However, this suite of problems could potentially become a historical archive of test problems, solution methods, and lessons learned.

The MDO test suite is available to anyone with web access. New test problems are strongly encouraged and will be accepted if they are submitted in a format that is consistent with existing entries. Issues that relate to the maintenance of the test suite, the collection of feedback, and the enforcement of standards require further investigation.

References

1. Venkayya, V. B., Tischler, V. A., and Pitrof, S. M., "Benchmarking in Structural Optimization," *Fourth AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization* -*Collection of Technical Papers,* Sept. 1992.
2. Balling, R. J., and Sobieszczanski-Sobieski, J., "Optimization of Coupled Systems: A Critical Overview of Approaches," *Fifth AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization* -*Collection of Technical Papers,* Sept. 1994, pp. 753–773.
3. Eason, E. D., Nystrom, G. A., Burlingham, A., and Nelson, E. E., "Robustness Testing of Non-hierarchic System Optimization Software," *Fifth AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization* -*Collection of Technical Papers,* Sept. 1994, pp. 817–824.
4. Cramer, E. J., Frank, P. D., Shubin, G. R., Dennis, J. E., and Lewis, R. M., "On Alternate Problem Formulation for Multidisciplinary Design Optimization," *Fourth AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization* – *Collection of Technical Papers,* Sept. 1992, pp. 518–530.
5. Padula, S. L., James, B. B., Graves, P. C., and Woodard, S. E., Multidisciplinary Optimization of Controlled Space Structures with Global Sensitivity Equations, NASA TP-3130, Nov. 1991.
6. Renaud, J. E., "An Optimization Strategy for Multidisciplinary Systems Design," *International Conference on Engineering Design*, August 1993.
7. Dennis, J. E., Gay, D. M., and Vu, P. A., "A New Nonlinear Equations Test Problem," Report 83-16, Rice University, Houston, Texas, 1983. Revised January 1986.
8. Dovi, A. R., Wrenn, G. A., Barthelemy, J-F. M., Coen, P. G., and Hall, L. E., "Multidisciplinary Design Integration Methodology for a Supersonic Transport Aircraft," *Journal of Aircraft,* Vol. 32, No. 2, 1995, pp. 290–296.

9. Haftka, R. T., Sobieszczanski-Sobieski, J., and Padula, S. L., " On Options for Interdisciplinary Analysis and Design Optimization," *Structural Optimization*, Vol. 4, No. 2, 1992, pp. 65–74.

Table 1. Three classes of test problems in version 1.0 of the MDO Test Suite

No.	**Name**	**Discipline 1**	**Discipline 2**	**Discipline 3**
1.1	Heart dipole			
1.2	Propane combustion			
2.1	Aircraft sizing	Aerodynamics	Performance	
2.2	Hub frame	Structures		
2.3	Electronic Package	Thermal	Circuit design	
2.4	Speed Reducer	Structures	Mechanics	
2.5	Power Converter	Electronics		
2.7	Rule-based design	Performance		
3.1	HSCT design	Structures	Aerodynamics	Approx. perf.
3.2	Space Platform	Structures	Controls	
3.4	Aerospike nozzle	CFD	Thermal	Approx. structures
3.6	Aerospike nozzle	CFD	Thermal	Structures
3.7	FIDO 2	Aerodynamics	Structures	Performance
3.8	Damper Placement	Structures		

Table 2. Characteristics of Test Problems in MDO Test Suite

No.	**Name**	**# of design variables (DV)**	**# of constraints**	**Notes**	**Status**
1.1	Heart	8	8	algebraic eqs.	Done
1.2	Propane	10	10	algebraic eqs.	Done
2.1	Aircraft	10	2	empirical curve fits	Planned
2.2	Hub	many	many	parallel processing	Done
2.3	Electronic	8	3		Done
2.4	Speed	7	11	multilevel	Testing
2.5	Power	6	4		Done
2.7	Rule-based	5	5	discrete DV	Done
3.1	HSCT	44	300	GSE and database	Done
3.2	Space	163	41	Needs EAL	Done
3.4	Aerospike	15	5		Planned
3.6	Aerospike	15	5	Needs NASTRAN	Planned
3.7	FIDO 2	many	many		Planned
3.8	Damper	1507	11	integer DV	Done

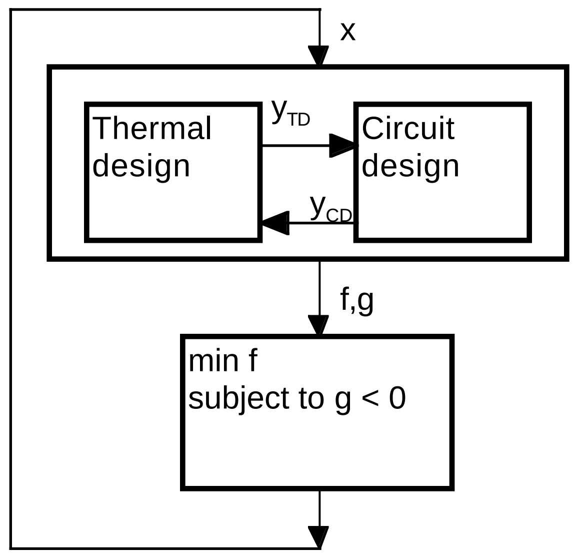


Figure 1. Diagram of benchmark solution method for electronic packaging problem.

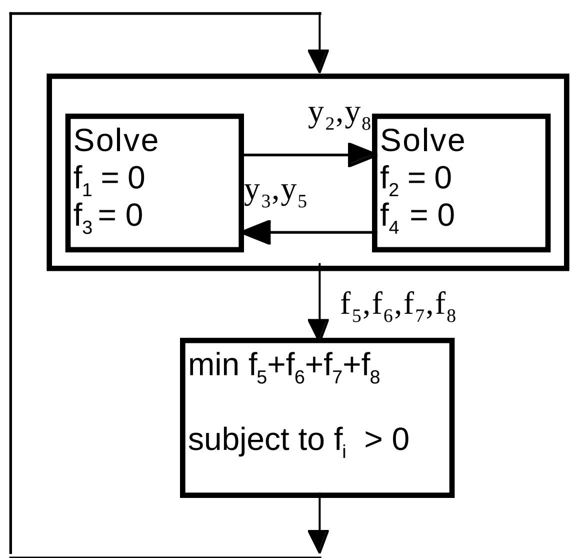
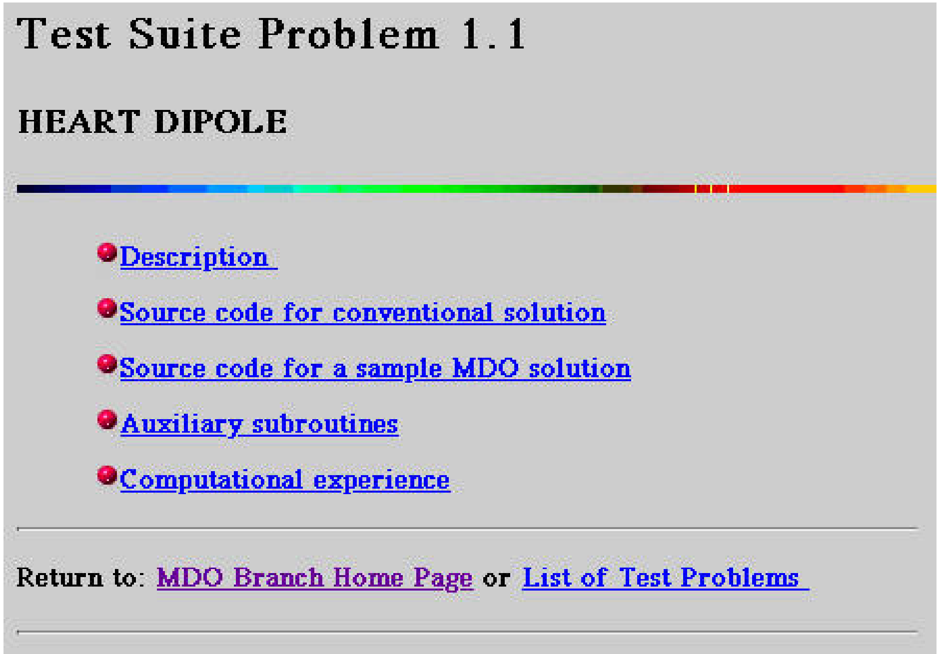


Figure 2. Diagram of benchmark solution method for heart dipole problem.



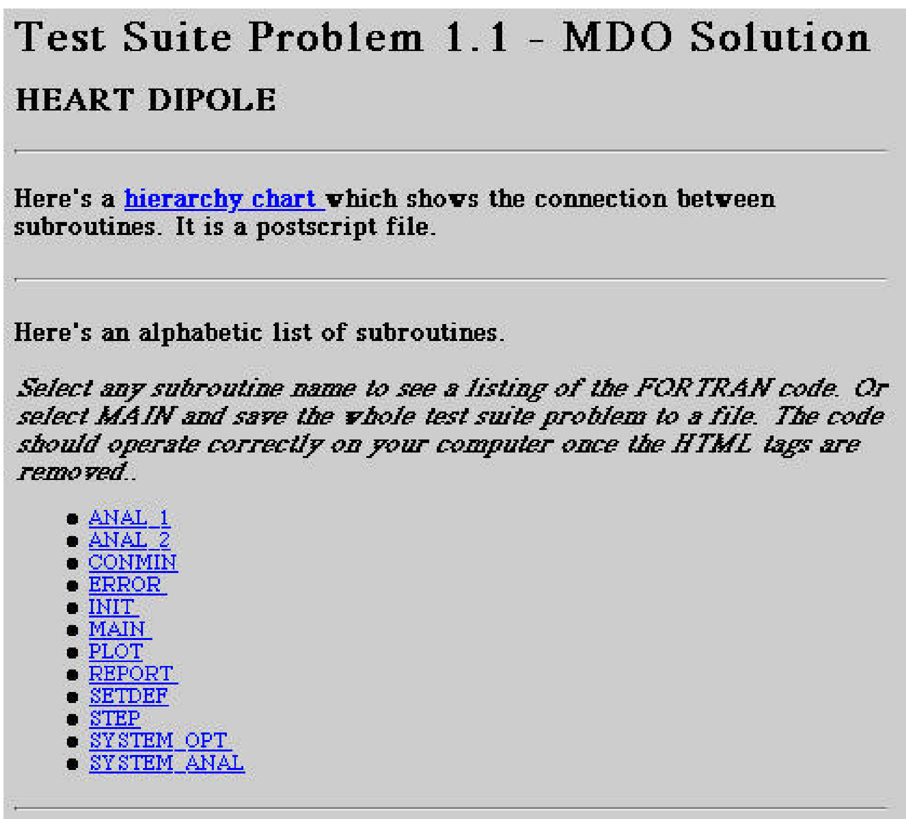
Test Suite Problem 1.1

HEART DIPOLE

- Description
- Source code for conventional solution
- Source code for a sample MDO solution
- Auxiliary subroutines
- Computational experience

Return to: MDO Branch Home Page or List of Test Problems

Figure 3. Top-level WWW page for heart dipole problem.



Test Suite Problem 1.1 - MDO Solution

HEART DIPOLE

Here's a hierarchy chart which shows the connection between subroutines. It is a postscript file.

Here's an alphabetic list of subroutines.

Select any subroutine name to see a listing of the FORTRAN code. Or select MAIN and save the whole test suite problem to a file. The code should operate correctly on your computer once the HTML tags are removed.

- ANAL_1
- ANAL_2
- CONMIN
- ERROR
- INIT
- MAIN
- PLOT
- REPORT
- SETDEF
- STEP
- SYSTEM_OPT
- SYSTEM_ANAL

Figure 4. Sample WWW page for heart dipole problem.

Appendix: Heart dipole problem

The human heart dipole problem arises from the experimental electrolytic determination of the resultant dipole moment in the heart. Given eight pieces of measured data, the problem is to solve a set of eight nonlinear equations in eight unknowns:

Given data $d_{mx}, d_{my}, d_A, d_B, d_C, d_D, d_E, d_F$

Find values of x_i such that

$$f_1 = x_1 + x_2 - d_{mx} = 0$$

$$f_2 = x_3 + x_4 - d_{my} = 0$$

$$f_3 = x_5 x_1 + x_6 x_2 - x_7 x_3 - x_8 x_4 - d_A = 0$$

$$f_4 = x_7 x_1 + x_8 x_2 + x_5 x_3 + x_6 x_4 - d_B = 0$$

$$f_5 = x_1(x_5^2 - x_7^2) - 2x_3 x_5 x_7 + x_2(x_6^2 - x_8^2) - 2x_4 x_6 x_8 - d_C = 0$$

$$f_6 = x_3(x_5^2 - x_7^2) + 2x_1 x_5 x_7 + x_4(x_6^2 - x_8^2) + 2x_2 x_6 x_8 - d_D = 0$$

$$f_7 = x_1 x_5(x_5^2 - 3x_7^2) + x_3 x_7(x_7^2 - 3x_5^2) + x_2 x_6(x_6^2 - 3x_8^2) + x_4 x_8(x_8^2 - 3x_6^2) - d_E = 0$$

$$f_8 = x_3 x_5(x_5^2 - 3x_7^2) - x_1 x_7(x_7^2 - 3x_5^2) + x_4 x_6(x_6^2 - 3x_8^2) - x_2 x_8(x_8^2 - 3x_6^2) - d_F = 0$$

The human heart dipole can be converted into a test problem for MDO methods by defining a system-level problem, and two subsystem-level analyses:

system level optimization

minimize : $f_5 + f_6 + f_7 + f_8$

subject to :

$f_5 \geq 0$

$f_6 \geq 0$

$f_7 \geq 0$

$f_8 \geq 0$

design variables : x_1, x_4, x_6, x_7

subsystem analysis 1

given : x_1, x_4, x_6, x_7

given : outputs from subsystem 2

solve :

$f_1 = 0$

$f_3 = 0$

to find : x_2, x_8

subsystem analysis 2

given: x_1, x_4, x_6, x_7

given: outputs from subsystem 1

solve:

$f_2 = 0$

$f_4 = 0$

to find: x_3, x_5

The coupled analysis system (i.e., subsystem 1 plus subsystem 2) can be solved by using fixed-point iteration. This technique involves guessing values for x_3, x_5 and iterating between the subsystems to find equilibrium values of x_2, x_3, x_5, x_8 for fixed values of the design variables. The system-level optimization problem is solved by using any nonlinear programming algorithm. Notice that the quality of the solution is easily evaluated by testing whether $f_i = 0$ for $i = 1, 2, ..., 8$.