Structural Sizing by Generalized, Multilevel Optimization

Jaroslaw Sobieszczanski-Sobieski*
NASA Langley Research Center, Hampton, Virginia
and
Benjamin B. James† and Michael F. Riley‡
Kentron International, Inc., Hampton, Virginia

Developments of a general multilevel optimization capability and results for a three-level structural optimization are described. The latter is considered a major stage in the method development because the addition of more levels beyond three does not introduce any new qualitative elements. Therefore, a three-level implementation is, qualitatively, the equivalent to a multilevel implementation. The method partitions a structure into a number of substructuring levels where each substructure corresponds to a subsystem. The method is illustrated by a portal framework that decomposes into individual beams. Each beam is a box that can be further decomposed into stiffened plates. Consequently, substructuring for this example spans three levels: the bottom level of finite elements representing the plates, an intermediate level of beams treated as substructures, and the top level for the assembled structure. This example is an extension of a previously presented case that was limited to two levels.

Nomenclature

A = cross-sectional area
C = cumulative constraint, Eq. (22)
c = capacity, a limitation on the ability to meet a particular demand d (e.g., allowable stress)
d = demand, a physical quantity the structure is required to have, to support, or to be subjected to in order to perform its function (e.g., stress)
F = objective function
$f(\ )$ = functional relation
g^{ij} = vector of constraint functions, g_w ($w=1\rightarrow W^{ij}$)
h^{ij} = vector of partitions h_K^{ij}, h_M^{ij}, h_P^{ij}, Eq. (25b)
$h_K^{ij}, h_M^{ij}, h_P^{ij}$ = vectors of the equality constraints defined by Eqs. (13–15), respectively; the vector elements are, respectively: $h_{KS_1}^{ij}$ at 3, $h_{MS_2}^{ij}$ at 4, and $h_{PS_3}^{ij}$ at 5, where $S_1=1\rightarrow S_1^{ij}$, $S_2=1\rightarrow S_2^{ij}$, and $S_3=1\rightarrow S_3^{ij}$
I = cross-sectional moment of inertia
K^{ij} = stiffness matrix of $SSijk$
K^{bij} = boundary stiffness matrix for $SSijk$
L^{ij} = lower bound on X^{ij}, including move limits
M^{ij} = mass of $SSijk$ (a scalar)
P^{ij} = vector of the external loads applied to interior and/or boundary of $SSijk$
P^{bij} = vector of P^{ij} transferred to the boundary of $SSijk$
Q^{ij} = vector of the forces Q_r^{ij}, $r=1\rightarrow R^{ij}$, acting on the boundary of $SSijk$
S_K = summation of stiffnesses contributed by substructures $SSijk$ assembled in a parent substructure $SSmkl$
S_P = summation of the boundary loads contributed by substructures $SSijk$, assembled in a parent substructure $SSmkl$
Q^{ik} = boundary forces Q_r^{ij} of $SSijk$, $r=1\rightarrow R^{ij}$
$SSijk$ = substructure (including the extremes of the assembled structure and a single structural element)
$SSmkl$ = substructure parent of $SSijk$, $m=i-1$, see Fig. 2
$SSSnlp$ = substructure parent of $SSmkl$, $n=m-1$, see Fig. 2
STOC = subject to constraints
SUMT = sequential unconstrained minimization technique
U^{ij} = upper bound on X_t^{ik}
X^{ij} = vector of design variables X_t in $SSijk$, $t=1\rightarrow T^{ij}$
Y^{ij} = vector of the entries in K^{bij}, M^{ij}, and the entries in P^{bij} that are held constant as parameters in optimization of $SSijk$; vector Y^{ij} contains V^{ij} elements Y_v^{ij}
Z^{ij} = vector of cross-sectional dimensions Z_b^{ij}, $b=1\rightarrow B^{ij}$, used as design variables in $SSijk$ that corresponds to a single structural element
π = vector defined by Eq. (26)
ρ = user-controlled constant in the KS function, see Eq. (22)
Δ = increment of a variable (see definition of subscript 0)

Indices, Subscripts, Superscripts

$(\bar{\ })$ = optimal quantity
b = association with the SS boundary
e = identification of an extrapolated value
0 = original (references) value from which an increment is measured

Presented as Paper 85-0697 at the AIAA/ASME/ASCE/AHS 26th Structures, Structural Dynamics and Materials Conference, Orlando, FL, April 15–17, 1985; received Aug. 8, 1985; revision received June 2, 1986.

*Deputy Head, Interdisciplinary Research Office, Structures Directorate. Associate Fellow, AIAA.

†Structures Project Engineer, Aerospace Technologies Division, Advanced Computer Methods Unit. Member AIAA.

‡Scientific Programmer, Aerospace Technologies Division, Advanced Computer Methods Unit.

Introduction

FORMAL optimization methods applied to realistic structures having many components and carrying a large number of loading cases are hindered by an excessive number of design variables and constraints. They frequently become too costly and unmanageable and can easily saturate even the largest computers available today or in the foreseeable future. An obvious remedy is to break large optimization problems into several smaller subproblems and a coordination problem formulated to preserve the couplings among these subproblems. A very important benefit of such an approach, in addition to making the whole problem more tractable, is preservation of the customary organization of the design office in which many engineers work concurrently on different parts of the problem. Therefore, research has been directed toward multilevel optimization methods that decompose large problems into a hierarchically related set of smaller subproblems while preserving their coupling. This approach meshes well with the recent trend in computer technology toward computing distributed over a network of computers whose characteristics may be matched to individual subproblems for more efficiency and convenience. Moreover, the decomposition approach is natural in an engineering organization, since engineers tend to work in teams, concentrating on parts of a project in order to develop a broad work front and, thus, to shorten the development time.

A number of procedures for implementation of the foregoing approach have been proposed for structural applications (e.g., Refs. 1-3). A multilevel optimization with decomposition has also been formulated in a general manner for use in engineering system design[4] concerned with the "tradeoffs" among various physical subsystems that may be governed by different engineering disciplines. The unique feature of the algorithm proposed in Ref. 4 is the use of the optimum sensitivity derivatives introduced in Refs. 5 and 6 as a means of approximate the coupling among the subsystems.

When the system optimization formulation established in Ref. 4 is applied to structural optimization, its analysis part coincides with a general, multilevel substructuring (see, e.g., Refs. 7-9). In the simplest case, the system acquires the meaning of a complete structure and each subsystem corresponds to a single structural component that may be represented by a finite element. This is a two-level, structural optimization whose algorithm was illustrated by an example of a framework reported in Ref. 10. It served as a verification of the general-purpose algorithm laid out in Ref. 4.

Since the general algorithm presented in Ref. 4 allows a theoretically unlimited number of hierarchical subsystem levels in the decomposition, its continuing development requires verification by applications of more than two levels. One such particular application is structural optimization with a three-level decomposition. In this decomposition, the highest level corresponds to the assembled structure, the level below corresponds to the substructures, and the third, bottom level represents the structural components that make up the substructures.

Extension of the scheme beyond the three levels would require more levels of nested substructures sandwiched between the top and bottom levels. Thus, such an extension would not add any qualitatively new subsystems to the scheme. Therefore, one may regard the three-level optimization as the simplest case of the most general multilevel optimization.

The purpose of this paper is to present a general, multilevel algorithm for structural optimization and to outline its validation by an example of a three-level optimization of a framework structure (described in detail in Ref. 11).

One-Level Optimization

An optimization formulation without decomposition serves as a reference from which the multilevel optimization algorithm is derived. The optimization is defined in terms of: the design variables Z_b (which are the cross-sectional dimensions of the structural components), the objective function $F(Z)$ that can be any computable function of these variables (structural mass is the frequent choice), and the constraints $g_w(Z)$ imposed on the behavior variables to account for the potential failure modes. Writing the constraint functions as

$$g = d/c - 1 \leqq 0 \tag{1}$$

the optimization problem in a standard formulation is

$$\min_{Z} F(Z);\ \text{STOC}\ g_w(Z) \leqq 0 \tag{2}$$

and requires a search of the design hyperspace considering all of the design variables and constraints concurrently. In contrast, an algorithm presented in the next section breaks the problem into a number of search and analysis operations, each concerned with a smaller number of design variables and constraints.

Multilevel Optimization

Preliminary Definitions

The diagram in Fig. 1 shows a structure decomposed into several levels of substructures. The term "substructure" will refer to any entity in this decomposition scheme, including the extremes of the full, assembled structure represented by the box on the top of the pyramid and single structural components representing the ultimate geometrical details appropriate to the problem at hand. The substructure levels are numbered from 1 on the top to i_{max} at the bottom. The hierarchical nature of the scheme instigates the use of the term "parent" to the structure at level i that, in turn, is decomposed into a number of "daughter" substructures at level $i+1$. A daughter may have only one parent and that parent must be at the level immediately above. Thus, it will be convenient to label each substructure $SSijk$, where i denotes the level, j defines the position at the level i counting from the left, and k identifies the parent's position at the level $i-1$. The substructure occupying the last position in a particular parent-daughter succession represents the ultimate level of detail at which the decomposition stops. There is no requirement that all such substructures must be at the same bottom level i_{max}. In discussions involving more than one substructure, the triplets nlp, mkl, and ijk are used to

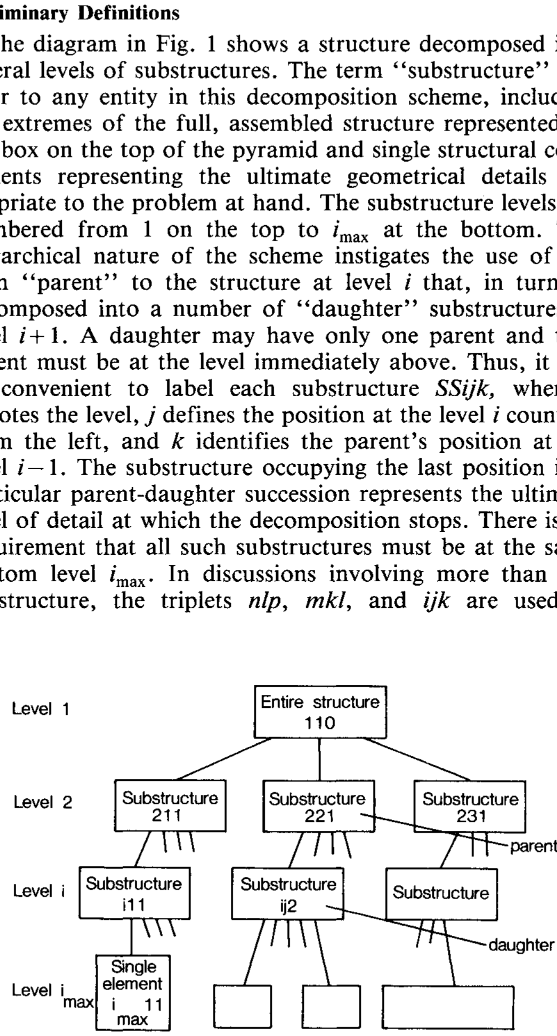

Fig. 1 Multilevel substructuring.

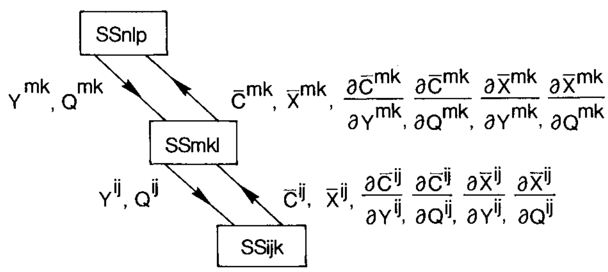

Fig. 2 Flow of information.

distinguished among the substructures forming the hierarchy shown in Fig. 2.

Substructuring analysis (e.g., Refs. 6-8) establishes the following functional relations

$$Q^{ij}=f(K^{bij},P^{bij}) \tag{3}$$

$$K^{bij}=f(K^{ij}) \tag{4}$$

$$K^{ij}=f(K^{bi+1,j})=S_K(K^{bi+1,j}) \tag{5}$$

$$M^{ij}=f(M^{i+1,j})=\sum_j M^{i+1,j} \tag{6}$$

$$P^{bij}=f(P^{ij}) \tag{7}$$

$$P^{ij}=S_P(P^{bi+1,j}) \tag{8}$$

These relations are computable in a manner prescribed by the particular substructuring algorithm chosen. For example, Eqs. (4) and (7) take the form of matrix equations given in Ref. 8, Chap. 9, Sec. 1, as Eqs. 9.13 and 9.14, respectively.

For *SSijk* at the ultimate level of detail, the distinctions between K^{bij}, P^{bij}, and K^{ij}, P^{ij} vanish, and K^{ij}, M^{ij} derive directly from Z^{ij}. Consequently:

$$K^{bij}=K^{ij} \tag{9}$$

$$P^{bij}=P^{ij} \tag{10}$$

$$K^{ij}=f(Z^{ij}) \tag{11}$$

$$M^{ij}=f(Z^{ij}) \tag{12}$$

The local constraints that arise in *SSijk* at the ultimate level of detail involve calculation of the stresses, strains, local buckling, etc., from Q^{ij} and Z^{ij}. In addition, constraints may be imposed on the internal forces, critical forces, and displacements of *SSijk* to account fully for all the constraints that would have been included in the one-level optimization problem represented by Eq. (2).

Although the foregoing definition of substructuring analysis is based on the finite-element stiffness method, the use of a finite-element analysis is not manatory for the multilevel optimization algorithm presented here. As far as that algorithm is concerned, the analysis is a "black box" where only the inputs and outputs are important.

Multilevel Optimization Algorithm

With the substructuring scheme and analysis established in the foregoing, this subsection describes the optimization algorithm itself. The essentials of the computer implementation are also given.

Basic Concept

The basic idea for the proposed multilevel optimization by substructuring stems from the elementary observation, based on Eqs. (3-8), that the effect of a daughter *SSijk* on its parent $SSi-1$, *kl* is felt only through K^{bij}, M^{ij}, and P^{bij}, which depend on $K^{b,i+1,j}$, $M^{i+1,j}$, and $P^{b,i+1,j}$, respectively. Consequently, the stiffness properties, the mass distribution, and the manner in which the interior loads are transferred to the boundary may be controlled in *SSijk* without disturbing the results of the $SSi-1$, *kl* analysis by manipulating the entries of $K^{b,i+1,j}$, $M^{i+1,j}$, and $P^{b,i+1,j}$ as generalized design variables in a manner somewhat restricted so that the entries of K^{bij}, M^{ij}, and P^{bij} are held constant. If the entries are held constant, then the boundary forces Q^{ij} acting on every *SSijk* in $SSi-1$, *kl* remain constant and the effect of manipulating the generalized design variables in a particular *SSijk* is limited to that of *SSijk* itself and its daughters. As it will be explained later, the purpose of the above manipulation of the matrix entries is not to minimize the substructure mass M^{ij}, which, as stated above, remains constant. Instead, the purpose is to improve satisfaction of the constraints in the *SSijk* and its daughters, while performing the task of the total mass optimization at the assembled structure level.

Invariance of the entries K^{bij}, M^{ij}, and P^{bij} can be enforced by rewriting Eqs. (4-8) as equality constraints,

$$h_K^{ij}=K^{bij}-f(K^{bi+1,j})=0 \tag{13}$$

$$h_M^{ij}=M^{ij}-f(M^{i+1,j})=0 \tag{14}$$

$$h_P^{ij}=P^{bij}-f(P^{bi+1})=0 \tag{15}$$

Equations (13-15) establish the entries of K^{bij}, M^{ij}, and P^{bij} as parameters in optimization of *SSijk*. Simple replacement of indices renders these equations valid for $SSi-1$, *kl* and redefines the optimization parameters of the daughter *SSijk* as generalized design variables in the optimization of its parent $SSi-1$, *kl*, so that

$$\{X^{i-1,k}\}=\{Y^{ij}\} \tag{16}$$

$$\{Y^{ij}\}^t=\{K^{bij}\,|M^{ij}\,|P^{bij}\}^t \tag{17}$$

These equations define a recursive relation of the variables and parameters that extends from the top of the substructuring scheme to the bottom. Of course, the number of design variables T^{ij} must exceed the number of parameters V^{ij} [which is equal to the number of individual equations in the vector Eqs. (13-15)],

$$T^{ij}>V^{ij} \tag{18}$$

for a design freedom to exist, allowing for the symmetry of the stiffness matrices. Otherwise, if

$$T^{ij}\leqq V^{ij} \tag{19}$$

then the equality constraints of Eqs. (13-15) either define the *SSijk* design variables uniquely or overdetermine them.

The basic concept outlined above translates into an algorithm to be introduced now in detail.

Optimization at the Most Detailed Level

Introduction of the optimization algorithm begins at the level of the most detailed substructures. Consequently, Eqs. (9-12) apply and the design variables are the cross-sectional dimensions so that

$$X^{ij}=Z^{ij} \tag{20}$$

and the parameters are

$$\{Y^{ij}\}^t=\{K^{ij}\,|M^{ij}\,|P^{ij}\}^t \tag{21}$$

It is assumed that a complete, top-down substructuring analysis for an initialized structure has been carried out so that for an *SSijk* one has computed its Q^{ij}, while its M^{ij}, Z^{ij}, K^{ij}, and P^{ij} are given.

Optimization for improvement of the constraint satisfaction is achieved by minimizing a single measure representing all of the constraints—called the cumulative constraint, a concept similar to the use of a penalty term in the SUMT. A differentiable cumulative constraint function can be obtained (as it was in Ref. 10) by means of the Kresselmeier-Steinhauser function (KS) defined in Ref. 12,

$$C^{ij}=\mathrm{KS}(g_w^{ij})=1/\rho\ln[\Sigma_w\exp(\rho g_w^{ij})] \tag{22}$$

that has the property of approximating the maximum constraint so that

$$\max(g_w^{ij}) < \mathrm{KS}(g_w^{ij}) < \max(g_w^{ij}) + 1/\rho \ell n (W^{ij}) \tag{23}$$

with the factor ρ controlled by the user. Thus, the KS function serves as a convenient single measure of the degree of constraint violation (or satisfaction).

Analysis of *SSijk* yields the local constraints as

$$g^{ij} = f(X^{ij}, Y^{ij}, Q^{ij}) \tag{24}$$

Based on the above definitions, the optimization problem is formulated.

$$\min_{X^{ij}} C^{ij}(X^{ij}, Y^{ij}, Q^{ij}) \mathrm{STOC} \tag{25a}$$

$$h_K^{ij} = 0, \quad h_M^{ij} = 0, \quad h_P^{ij} = 0 \tag{25b}$$

$$L^{ij} \leqq X^{ij} \leqq U^{ij} \tag{25c}$$

Solution of this optimization problem (by any technique available) yields a constrained optimum described by a vector π^{ij} composed of the minimum value of the cumulative constraint $\bar{C}^{(ij)}$ and the optimal vector of the design variables $\bar{X}^{(ij)}$,

$$\pi^{ijt} = \{\bar{C}^{ij} \mid \bar{X}^{ij}\}^t \tag{26}$$

This solution is sensitive to the parameters Y^{ij} and to Q^{kij} so that derivatives $\mathrm{d}\pi^{ij}/\mathrm{d}Y_z^{ij}$ and $\mathrm{d}\pi^{ij}/\mathrm{d}Q_r^{kij}$ exist and may be expressed by a chain differentiation to account for Eqs. (3) and (21) that tie Q^{ij} to Y^{ij},

$$\frac{\mathrm{d}\bar{C}^{ij}}{\mathrm{d}Y^{ij}} = \frac{\partial \bar{C}^{ij}}{\partial Y_z^{ij}} + \sum_r \left(\frac{\partial \bar{C}^{ij}}{\partial Q_r^{ij}}\right)\left(\frac{\partial Q_r^{ij}}{\partial Y_z^{ij}}\right) \tag{27}$$

$$\frac{\mathrm{d}\bar{X}^{ij}}{\mathrm{d}Y_z^{ij}} = \frac{\partial \bar{X}^{ij}}{\partial Y_z^{ij}} + \sum_r \left(\frac{\partial \bar{X}^{ij}}{\partial Q_r^{ij}}\right)\left(\frac{\partial Q_r^{ij}}{\partial Y_z^{ij}}\right) \tag{28}$$

In Eqs. (27) and (28), the partials of $\bar{C}^{ij}$ with respect to Y_z^{ij} and with respect to Q_r^{ij} are obtained from the algorithm described in Ref. 5 and the partial Q_r^{ij} with respect to the Y_z^{ij} by conventional structural sensitivity analysis. Parenthetically, one may add that the algorithm of Ref. 5 uses second derivatives of constraints that may be expensive to calculate. However, a modified version of the algorithm is available in Ref. 6 that avoids the cost of second derivatives and calculates the sensitivity derivatives for $\bar{C}^{ij}$ but not for $\bar{X}^{ij}$.

Optimization of a Parent Substructure

As shown in Fig. 2, the parent substructure *SSmkl*, $m = i-1$, receives from its daughters *SSijk* the minimized values of their cumulative constraints $\bar{C}^{ij}$, optimal values of their design variables $\bar{X}^{ij}$, and the optimum sensitivity derivatives of these quantities with respect to parameters Q^{ij} and Y^{ij}, according to Eqs. (26–28).

Preparing for the formulation of the optimization problem for the parent substructure, we consider the recursive relation between the design variables and parameters according to Eqs. (16) and (17) and recognize that Eqs. (9–12) do not apply. When optimizing the parent substructure, we want to improve the satisfaction of the assembled substructure constraints, such as its elastic deformations and stability that depend on the substructure stiffness, mass, and boundary forces, as

$$g^{i-1,k} = f(X^{i-1,k}, Y^{i-1,k}, Q^{i-1,k}) \tag{29}$$

At the same time, we want to improve that constraint satisfaction in all the substructure daughters. These can be approximated (as in Ref. 10) by linear extrapolation of their cumulative constraints using the derivatives from Eq. 27,

$$\bar{C}_e^{ij} = \bar{C}_0^{ij} + \sum_t \left(\frac{\mathrm{d}\bar{C}^{ij}}{\mathrm{d}X_t^{mk}}\right)\Delta X^{mk} \tag{30}$$

This extrapolation plays a key role in the algorithm because it approximates the daughter-parent coupling without incurring the expense of reoptimizing the daughters [repeating Eq. (25)] for every change of the parent design variables.

Including the $\bar{C}_e^{ij}$ values together with $g^{i-1,k}$ in a cumulative constraint formed by the KS function we have

$$C^{mk} = 1/\rho \ell n \left[\sum_w \exp(\rho g_w^{mk}) + \sum_j \exp(\rho \bar{C}_e^{ij})\right] \tag{31}$$

and the optimization problem to be solved for the parent *SSi*−1, *kl* is

$$\min_{X^{mk}} C^{mk}(X^{mk}, Y^{mk}, Q^{mk}) \mathrm{STOC} \tag{32a}$$

$$h^{mkt} = \{h_K^{mk} \mid h_M^{mk} \mid h_P^{mk}\}^t = 0 \tag{32b}$$

$$L^{mk} \leqq X^{mk} \leqq U^{mk} \tag{32c}$$

$$L^{ij} \leqq X_e^{ij} \leqq U^{ij} \tag{32d}$$

where

$$X_e^{ij} = X_0^{ij} + \sum_t \left(\frac{\mathrm{d}\bar{X}^{ij}}{\mathrm{d}X_t^{mk}}\right)\Delta X_t^{mk} \tag{33}$$

The increment X^{mk} is defined as

$$\Delta X^{mk} = X^{mk} - X_0^m \tag{34}$$

The constraints of Eq. (32b) are analogous to Eqs. (25b) written in a compact format. The constraints of Eq. (32c) incorporate the side constraints to prevent the design variables from attaining physically impossible values (e.g., negative diagonal entries in a stiffness matrix) and include the move limits to control the extrapolation errors introduced by Eq. 30. The constraints of Eq. (32d) are introduced to keep the design variables in the daughters from exceeding their side constraints. These constraints are not essential because their function may be performed directly by the daughter side constraints. In fact, omitting the constraints of Eq. (32d) eliminates the need for the derivatives of $\bar{X}^{ij}$ and allows replacing the algorithm of Ref. 5 with the much less costly algorithm of Ref. 6. However, these constraints are included in this description for completeness.

Solution of the problem of Eqs. (32) generates the result vector and its derivatives that are analogous to those of Eqs. (26–28) with the indices *ij* replaced by $m = i-1$, and *k*.

Optimization of the Next Parent Structure

Moving on to the substructure *SSnlp*, everything that was stated in the preceding subsection on the optimization of *SSmkl* applies to *SSnlp* directly, provided that: the indexes *n*, *l*, and *p* are replaced by another triplet, say, α, β, γ, that identifies the parent of *SSnlp* at the level $\alpha = n-1$ and the indexes *m*, *k*, *l* are replaced by *n*, *l*, *p*. For consistency, Eq. (32d), if it is used, should be replicated to encompass fully each line of succession emanating downward from *SSnlp*. Beyond these changes, no new conceptual elements are introduced and no additional definitions or discussion are needed at the junctions between the levels until one arrives at the top level. Hence, any number of intermediate levels of substructuring can be inserted, if physically justified, into a line of succession extending downward from the assembled structure on the top; i.e., the algorithm is recursive.

Optimization of the Assembled Structure

The assembled structure is designated *SS*110. Its optimization problem is similar to the one described for a parent substructure *SSmkl* with the following differences:

1) No parameters are defined solely for the decomposition purposes; therefore, there is no need for the equality constraints to enforce constancy of the mass and the boundary stiffnesses.

2) The objective function is the mass of the assembled structure.

3) There is no need for a single cumulative constraint (unless one needs it to reduce the number of constraints to be processed at that level).

4) The boundary forces are the external loads on the assembled structure.

Accounting for these differences, the optimization problem for the top level is

$$\min_{X^{11}} M^{11}(X^{11})\,\mathrm{STOC} \tag{35a}$$

$$g^{11} \leqq 0 \tag{35b}$$

$$\bar{C}_e^{2j} \leqq 0 \tag{35c}$$

$$L^{11} \leqq X^{11} \leqq U^{11} \tag{35d}$$

$$L^{2j} \leqq \bar{X}_e^{2j} \leqq U^{2j} \tag{35e}$$

where Eq. (35e) is analogous to Eq. (32d) with the limits L^{2j}, U^{2j} applied in conjunction with extrapolations of the type expressed by Eq. (33) extended recursively to encompass all of the levels below as mentioned in the subsection on *SSnlp*. Unlike in the daughters *SSijk*, the optimization of *SS*110 does not have to be analyzed for the optimum sensitivity. Information transmitted to the top level optimization problem is indicated in Fig. 2.

Iterative Procedure

When the *SS*110 optimization is completed, the entire structure has acquired a new distribution of stiffness and mass within the move limits. Hence, the analysis must be repeated and followed by a new round and substructure optimizations in an iterative manner until convergence. Accordingly, the procedure follows these steps:

1) Initialize all cross-sectional dimensions.

2) Perform a substructuring analysis, including for each substructure at each level, the transformation of the stiffness matrix into the boundary stiffness matrix and the transformation of the forces applied to the interior degrees of freedom to the forces coinciding with the boundary degrees of freedom. Calculations of the behavior derivatives needed for the ensuing optimizations and for the optimum sensitivity analyses are included in the substructuring analysis.

3) Perform the operations of optimization and optimum sensitivity analysis as defined by Eqs. (25-34).

4) Optimize the assembled structure as defined by Eqs. (35).

5) Repeat from step 2 and terminate only when all constraints g^{ij} are satisfied at all levels *and* $M11$ has entered a phase of diminishing returns.

This procedure is illustrated in Fig. 3 by a flow chart in the Chapin chart format.[13]

Salient Features of the Algorithm

In perspective, the multilevel algorithm differs from a conventional one in a number of salient features outlined in this subsection.

A multitude of smaller problems, which may be processed concurrently replace a single large problem. Although the subproblems are isolated, their coupling is preserved because the influence of the changes in the parent on the daughters is represented by linear extrapolation based on the optimum sensitivity and behavior sensitivity derivatives. With the exception of the most detailed level, the stiffness and mass distributions are controlled directly by generalized design variables. Mass is the objective at the top level, while improvement in the constraint satisfaction is the objective at all levels below.

Selection of the generalized design variables is judgmental. In the extreme, one may choose to control all entries of the boundary stiffness matrix, boundary forces vector, and mass of each daughter, although, intuitively, this would seem impractical. Experience will probably show that a limited control, e.g., over the diagonal entries of the stiffness matrix only, will suffice in most cases.

The overall procedure building blocks, i.e., the operations of substructure analysis, constraint calculations, optimization, and behavior and optimum sensitivity analyses are "black boxes" whose algorithmic contents may be freely replaced, provided that the input/output definitions remain unchanged. For example, different types of structural analysis may be used at each level and even for each substructure, as it will be shown in the numerical example.

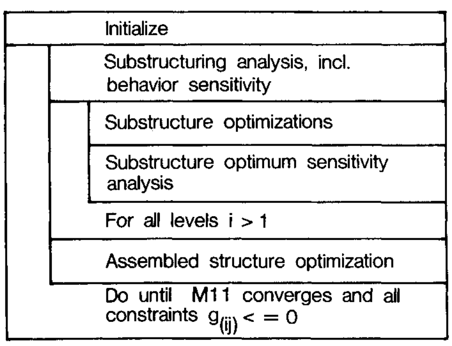

Fig. 3 Multilevel optimization procedure flow chart.

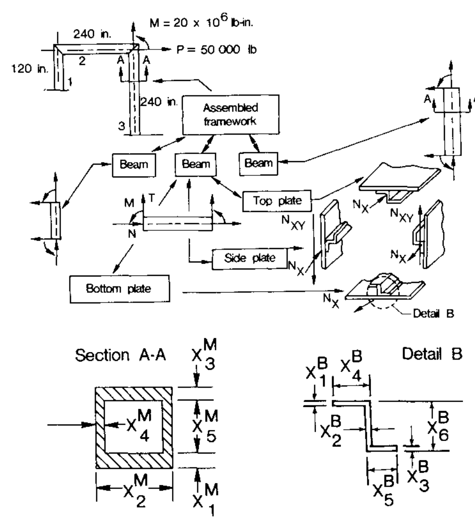

Fig. 4 Portal framework.

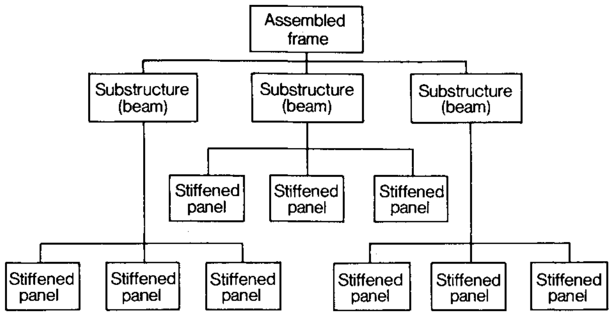

Fig. 5 Hierarchical decomposition of the framework structure shown in Fig. 4.

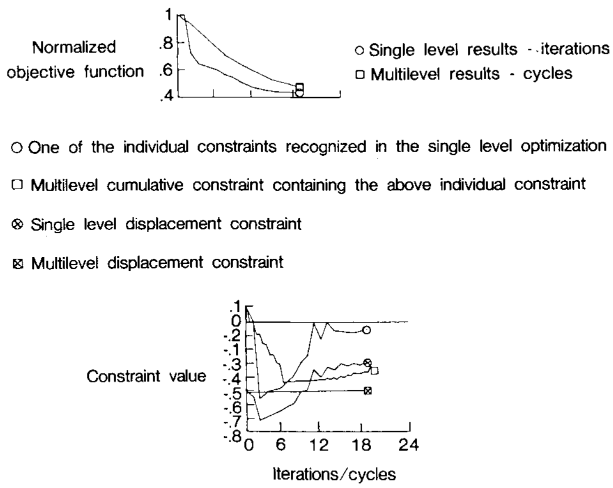

Fig. 6 Representative results.

Numerical Example

Test Case

The subject algorithm was tested by optimizing, with and without decomposition, a framework structure similar to the one used in Refs. 10, 14, 15. As shown in Figs. 4 and 5, the framework assembled at level 1 decomposes into three box beams, each beam being a substructure at level 2. Finally, each beam decomposes into three walls (the fourth wall is symmetric), each wall being the most detailed substructure at level 3. The external loads were applied at one corner of the framework as shown in Fig. 4. There were no interior loads on the substructures.

The objective was to minimize the structural material volume subject to constraints on the displacements of the loaded point, and in- and out-of-plane elastic stability of each beam treated as a column, and the stresses and local buckling of the wall panels treated as stringer-reinforced plates. There were also minimum gage constraints and the physical realizability constraints on the cross-sectional dimensions.

The objective functions, design variables, parameters, and constraints are defined for the multilevel optimization in Table 1. A comprehensive description of all the physical and computational details of the test problem is given in Ref. 11.

Tools for Analysis and Design Space Search

A finite-element analysis was used to calculate the framework's displacements and beam end forces. Stresses in the beams loaded with the end forces were computed by ordinary engineering beam theory. The beams were treated as columns for stability analysis and the local buckling of the walls was based on closed form "designer handbook" formulas provided in Refs. 16 and 17 and implemented as described in Ref. 18.

Table 1 Quantities defined for the multilevel test case optimization

	Top level
Objective	Framework material volume
Design variables	A and I of the beams
Constraints	Displacements of the loaded corner and $\bar{C}_e$ for the beams
	Middle level
Objective	Cumulative constraint C representing the column buckling and $\bar{C}_e$ for the walls
Parameters	Beam cross-sectional area and moment of inertia
Design variables	Wall membrane stiffness contributing to the beam axial and bending stiffnesses controlled through the dimensions shown in Fig. 3 (Sec. A-A)
Constraints	Equality-beam cross-sectional area and moment of inertia
	Bottom level
Objective	Cumulative constraint C representing a set of stress and local buckling constraints of the wall
Parameters	Membrane stiffness of the wall
Design variables	Cross-sectional dimensions shown in Fig. 3 (Detail B)
Constraints	Inequality-minimum gages, geometrical proportions, and geometrical realizability
	Equality-membrane stiffnesses for tension-compression and bending of the wall in its own plane

At each level, the design space search was conducted by the same general-purpose nonlinear mathematical programming code based on the usable-feasible directions technique and documented in Ref. 19.

Three-Level Optimization

The framework was first optimized without decomposition to establish the reference results. Then, the multilevel optimization algorithm was applied to the structure decomposed as shown in Figs. 4 and 5. In the decomposition the stiffened panels are daughters clustered in triplets under a parent box beam. The beams, in turn, are daughters of the assembled structure.

As shown in Table 1, the top level optimization manipulates the beam extensional and bending stiffnesses through the cross-sectional areas and bending moments of inertia. In this particular case, the cross-sectional area plays a dual role as it also controls the beam volume that contributes directly to the objective function.

At the middle level, the stiffnesses expressed by the area and moment of inertia become fixed parameters and the variables are the wall membrane stiffnesses controlled by the geometrical dimension variables. These variables and, consequently, the membrane stiffnesses become fixed parameters at the bottom level at which the ultimate detail dimensions are engaged as variables. The equality constraints arise between the parameters and variables. Owing to relative simplicity of the expressions involved, (see the Appendix of Ref. 11), these constraints were solved explicitly.

Examination of Table 1 in conjunction with the previous description of the analysis tools illustrates the point that dissimilar analysis may be used as needed at different places in a decomposition scheme.

The sensitivity analysis of behavior has been carried out by a single-step forward finite-difference technique. The optimum sensitivity analysis was based on the algorithm given in Ref. 5.

Results and Remarks on the Method Performance

Figure 6 shows a sample of results obtained with and without decomposition. The starting points for both methods are the same. The normalized plots illustrate the objective function, a selected individual constraint, and a cumulative constraint containing the above individual constraint as they varied over the iterations. An iteration is defined in the optimization without decomposition as a usable-feasible directions iteration. In the three-level optimization, it is defined as one execution of the series of steps listed in the procedure definition in the previous section.

The results verified that the multilevel algorithm was capable of finding a feasible design having an objective function close to and, in some cases lower than, the reference optimization without decompositon. Similarly as in Ref. 10, differences up to 72.1% were observed among the detailed design variables obtained by the two methods. However, these differences were no larger than those observed by comparing the designs obtained without decomposition starting from different initial design points. Therefore, these differences can be attributed to the problem nonconvexity. The jagged appearance of the graphs in Fig. 6 is a characteristic of the usable-feasible directions search algorithm, amplified in the multilevel optimization by the extrapolation errors. However, these errors never become excessive and the daughter substructure reactions to the changes in the parent design were effectively predicted by the optimum sensitivity derivatives. It was observed in at least one case that these predictions enabled the optimization at the middle level to remove the constraint violations at a bottom level substructure without any change to the bottom level substructure's sizing. A detailed comparison of the results from both methods is given in Ref. 11.

Regarding computational efficiency, the main intrinsic advantage of the multilevel algorithm is in its capability to process the subproblems concurrently. Demonstration of this advantage would require a large application, distributed computing, and division of work among many people. Consequently, computational efficiency was not one of the goals in execution of the relatively small numerical example on a conventional serial computer; however, the example showed that the amount of computational labor per one iteration was less in the multilevel algorithm than in the single-level, conventional one and that both algorithms required about the same number of iterations for convergence. The example also showed that the multilevel algorithm was data-handling intensive; the operations of data moving and bookkeeping dominated the programming effort.

Conclusions

An algorithm has been developed for solving the structural optimization problem as a set of smaller subproblems corresponding to levels of nested substructures. The intersubstructure couplings are preserved by the behavior and optimum sensitivity derivatives. The algorithm is inherently compatible with distributed computing because the subproblems can be processed concurrently.

Numerical tests performed on a conventional, serial computer for a framework decomposed into a three-level hierarchy of 13 subproblems demonstrated satisfactory convergence to the close vicinity of the reference solutions obtained by an optimization without decomposition. A significant part of the effort to program the algorithm went into the data handling, indicating that a systematic data management system (e.g., see Ref. 20) would be required in extending the applications to more levels. Since the addition of more levels beyond three would introduce no new qualitative elements into the algorithm, the three-level test case has been sufficient as a proof of the concept.

References

[1]Kirsch, U., Reiss, M., and Shamir, U., "Optimum Design by Partitioning into Substructures," *Journal of Structural Division ASCE*, Jan. 1972, p. 249.

[2]Sobieszczanski, J., "Sizing of Complex Structure by the Integration of Several Different Optimal Design Algorithms," *Structural Optimization*, AGARD-LS-70, Sept. 1974.

[3]Schmit, L. A. and Ramanathan, R. K., "Multilevel Approach to Minimum Weight Design Including Buckling Constraints," *AIAA Journal*, Vol. 16, Feb. 1973, pp. 97–104.

[4]Sobieszczanski-Sobieski, J., "A Linear Decomposition Method for Large Optimization Problems—Blueprint for Development," NASA TM 83248, Feb. 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, p. 1291.

[6]Barthelemy, J.-F. M. and Sobieszczanski-Sobieski, J., "Optimum Sensitivity Derivatives of Objective Function in Nonlinear Programming," *AIAA Journal*, Vol. 21, June 1983, pp. 913–915.

[7]Noor, A. K., Kamel, H. A., and Fulton, R. E., "Substructuring Techniques—Status and Projections," *Computers and Structures*, Vol. 8, May 1978, pp. 621–632.

[8]Przemieniecki, J. S., *Theory of Matrix Structural Analysis*, McGraw-Hill, New York, 1968, Chap. 9.

[9]Aaraldsen, P. O., "The Application of the Superelement Method in Analysis and Design of Ship Structures and Machinery Components," Paper presented at National Symposium on Computerized Structures Analysis and Design, George Washington University, Washington, DC, March 1972.

[10]Sobieszczanski-Sobieski, J., James, B., and Dovi, A., "Structural Optimization by Multi-Level Decomposition," *AIAA Journal*, Vol. 23, Nov. 1983, pp. 1775–1782.

[11]Sobieszczanski-Sobieski, J., James, B. B., and Riley, M. F., "Structural Optimization by Generalized Multilevel Optimization," AIAA Paper 85-0698, April 1985 (also NASA TM 87605, Oct. 1985).

[12]Kreisselmeier, G. and Steinhauser, R., "Application of Vector Performance Optimization to a Robust Control Loop Design for a Fighter Aircraft," *International Journal of Control*, Vol. 37, No. 2, 1983, pp. 251–284.

[13]Chapin, N., "New Format for Flowcharts," *Software—Practice and Experience*, Vol. 4, 1974, pp. 341–357.

[14]Lust, R. V. and Schmit, L. A., "Alternative Approximation Concepts for Space Frame Synthesis," AIAA Paper 85-0696-CP, April 1985.

[15]Haftka, R. T., "An Improved Computational Approach for Multilevel Optimum Design," *Journal of Structural Mechanics*, Vol. 12, No. 2, 1984, pp. 245–261.

[16]Angermayer, K., "Structural Aluminum Design," Reynolds Metals Co., Richmond, VA, 1965.

[17]Timoshenko, S. P. and Gere, J. M., *Theory of Elastic Stability*, McGraw-Hill, New York, 1961.

[18]Sobieszczanski-Sobieski, J., "An Integrated Computer Procedure for Sizing Composite Air Frame Structures," NASA TP-1300, Feb. 1979.

[19]Vanderplaats, G. N., "CONMIN—A Fortran Program for Constrained Function Minimization: User's Manual," NASA TM X-62282, Aug. 1973.

[20]*IPAD 'Integrated Program for Aerospace Vehicle Design'*, NASA CP 2143, 1980.