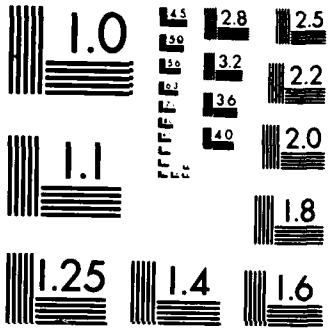
AD-A174 161 INTRODUCTION TO GRASP - GENERAL ROTORCRAFT AEROMECHANICAL STABILITY PROGR (U) ARMY AVIATION RESEARCH AND TECHNOLOGY ACTIVITY MOFFETT FIELD C 1/1

UNCLASSIFIED D H HODGES ET AL 1986 F/G 1/3 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

AD-A174 161

Introduction to GRASP — General Rotorcraft Aeromechanical Stability Program — A Modern Approach to Rotorcraft Modeling

Dewey H. Hodges
Research Scientist and Theoretical Group Leader

A. Stewart Hopkins, Donald L. Kunz, and Howard E. Hinnant
Research Scientists

Rotorcraft Dynamics Division
Aeroflightdynamics Directorate
U. S. Army Aviation Research and Technology Activity (AVSCOM)
Ames Research Center
Moffett Field, California

NOV 1 8 1986
A

Abstract

The General Rotorcraft Aeromechanical Stability Program (GRASP) is described in terms of its capabilities and development philosophy. The program is capable of treating the nonlinear static and linearized dynamic behavior of structures represented by arbitrary collections of rigid-body and beam elements that may be connected in an arbitrary fashion and are permitted to have large relative motions. The main limitation is that periodic coefficient effects are not treated, restricting the solutions to rotorcraft in axial flight and ground contact conditions. Rather than following in the footsteps of other rotorcraft programs, GRASP is more of a hybrid between finite element programs and spacecraft-oriented multibody programs. GRASP differs from standard finite-element programs by allowing multiple levels of substructures in which the substructures can move and/or rotate relative to others with no small-angle approximations. This capability facilitates the modeling of rotorcraft structures, including the rotating/nonrotating interface and details of the blade/root kinematics for various rotor types. GRASP differs from standard multibody programs by considering aeroelastic effects, including inflow dynamics (simple unsteady aerodynamics) and nonlinear aerodynamic coefficients. The main structural element is the *aeroelastic beam* element which may possess arbitrarily more than the 12 degrees of freedom common in beam elements. Although it is assumed in the analysis that the strain components in the *aeroelastic beam* element remain small compared to unity, no kinematical limitations are imposed on the magnitudes of the displacements and rotations. Numerical results from GRASP are presented and compared with results from an existing, special-purpose coupled rotor/body aeromechanical stability program and with experimental data for large deflections of an end-loaded cantilevered beam. The agreement is excellent in both cases.

Presented at the 42nd Annual Forum of the American Helicopter Society, Washington, D. C., June 2–4, 1986.

Introduction

Background

Early work in calculating the aeroelastic stability of hingeless helicopter rotor blades commonly made use of simple physical models such as spring-restrained, centrally-hinged, rigid blades.[1] Later work treated configurations that were somewhat more complex, including some with elastic blades,[2] body degrees of freedom, and inflow dynamics.[3] These approaches are very valuable for gaining physical insight into complicated phenomena such as coupled rotor-fuselage stability. Since they are based on only one physical model, however, they are of limited value when attempting to accurately analyze realistic rotorcraft configurations.

Especially important in the context of aeroelastic stability is the analysis of bearingless rotor systems. For these systems, the blade/root kinematics require a great deal of modeling flexibility, as various configurations may be quite different. The FLAIR program[4] attempted to model bearingless rotor helicopters to calculate aeromechanical stability. In FLAIR the user is limited to a rigid blade, a uniform flexbeam, only a few types of blade/root kinematics, linear aerodynamics, and static induced inflow. While FLAIR has found use in the rotorcraft community, it lacks the flexibility needed to become a serious design tool.

For analysis of problems involving complete rotorcraft, there exist large helicopter simulation programs such as C-81, described in Reference 5, and G400, described in Reference 6. These programs were designed primarily for time-history analysis of rotorcraft behavior in forward flight rather than for aeromechanical stability. Despite their generality and complexity, these programs are still limited. Johnson,[7] in his discussion of these and other large rotorcraft programs, points out some of their limitations, which are primarily related to aerodynamics. Although his CAMRAD program overcomes many of these limitations, these programs (including CAMRAD) are restricted to a fixed number of physical models and lack the modeling flexibility needed to deal with a wide variety of blade/root geometries. They must rely on results, such as a set of modes, from other programs which can present an assortment of

DTIC FILE COPY

This document has been approved for public release and sale; its distribution is unlimited.

86 9 22 096

difficulties for bearingless rotor blades. The mathematical and physical consistency of these combined approaches are seldom examined, and they are likely to *not* be consistent. Furthermore, in stability analyses, a nonlinear static equilibrium solution is needed about which to linearize — an important consideration which most of the earlier simulation programs do not address. One code should exist in which blade structural dynamics, isolated blade stability, and isolated rotor stability, as well as coupled rotor/airframe stability, can all be treated.

Dynamic coupling programs, such as DYSCO,[8] which have a high degree of generality allow coupling of discrete component models and/or modal representations of flexible structures. While DYSCO is a very powerful executive-driven system, it cannot at present treat the aeroelastic behavior of bearingless rotor systems undergoing geometrically nonlinear deformation because it lacks a sufficiently general element in its element library.

Recent implementations applying the finite-element method to rotorcraft problems[9,10] do not overcome these limitations because the physical models are limited to one configuration. If one takes a rotating beam and breaks it up into finite elements, this yields nothing more than a discretized rotating beam. The need to couple it with an airframe or to model blade/root kinematics of an arbitrary configuration is not met by such an approach. The classical finite-element method is based on the breaking up of a single structure (*i.e.*, a beam, plate, or shell) into an arbitrary number of elements and expanding the appropriate field variables into polynomial shape functions. This approach, by itself, lacks the flexibility to deal with truly arbitrary rotorcraft configurations. The reason for this is that a helicopter is a system of structural components, some of which may be rotating and/or translating relative to one another. It is more akin to the so-called "multibody" systems.[11,12] Unfortunately, few multibody programs possess the capability to deal with flexible components, and none have the capability to deal with aeroelastic phenomena as these programs were developed primarily for spacecraft applications.

All previous attempts at modeling rotorcraft problems have embodied certain inherent restrictions that are unacceptable in a truly general-purpose approach. A general-purpose code is needed that would overcome the major shortcomings of the previous philosophies of aeroelastic analysis. Consider, for example, the following typical restrictions: 1) *Restriction to linear, small-displacement approximations of beam elastic deformation.* This restriction is unacceptable in a general-purpose rotorcraft program because the rotor blade aeroelastic problem, especially for hingeless rotor blades, has been conclusively shown to be a nonlinear problem. A consistent approach based on nonlinear kinematics is required. 2) *Restriction to elastic blade models with ordering schemes, second-degree nonlinearity, or "moderate" rotations.* These approximations are unacceptable because the governing equations may have to be augmented with certain higher-order terms if the values of certain structural properties are not within some nominal range (Rosen and Friedmann[13]). Evidently, these higher-order terms need to be present in a general-purpose analysis. It then seems that the ordering scheme, although it can be a valuable tool in the context of special-purpose research codes, is neither necessary nor desirable in a general-purpose context. Furthermore, a bearingless rotor flexbeam must undergo deformation-induced rotations of the order of the collective pitch angle — a rotation too large to be classified as "moderate." Thus, the bearingless rotor problem demands a large-deflection analysis without artificial restrictions on rotations due to deformation, the degree of nonlinearity, or the values of blade properties. 3) *Restriction to a fixed number (usually one) of configurations* (e.g., *isolated hingeless blade or coupled bearingless rotor and body or a single blade/root configuration).* This restriction is unacceptable because of the need to analyze different types of configurations with one code and one set of assumptions. This one code should be able to treat all currently known blade/root mechanisms and, at the same time, model configurations that do not yet exist. The user should be able to "construct" a new configuration with simple building blocks and without artificial limitations on the process. For maximum flexibility in treating these different configurations, the finite-element method is the preferred approach. Moreover, the existence of many different blade/hub configurations for helicopters requires a capability to analyze arbitrary configurations of structures, parts of which may be rotating. Thus the code must be "multibody" in philosophy.

Approach

To overcome the aforementioned limitations of the previous methods of aeroelastic stability analysis, the General Rotorcraft Aeromechanical Stability Program (GRASP) has been developed. GRASP combines the finite-element and multibody approaches and incorporates multiple levels of substructures to provide a powerful tool for rotorcraft analysis. GRASP has been designed around the concept of a collection of flexible and rigid bodies connected in an arbitrary manner. Libraries of elements, constraints, and solution algorithms appropriate for the helicopter aeroelastic stability problem were designed and built in.

The element library fosters the treatment of the blades as beams; construction of arbitrary mechanisms to treat blade/root kinematics with beam elements and rigid bodies; treatment of the fuselage as either a rigid body, a collection of beam elements, or a modal representation obtained from some other source; and treatment of both static and dynamic induced inflow by means of blade-element/momentum theory. The constraint library allows for arbitrary connection of elements and includes constraints that allow for compliance in the constrained relative motion between elements, and a rotating/nonrotating interface, all without kinematical approximations such as

small-angle assumptions. The solution procedures include nonlinear static equilibrium and linearized stability about equilibrium, both presently limited to the hovering flight condition.

It should be noted that these physical modeling assumptions and solution procedures, while adequate for aeromechanical stability analysis in axial flight and ground contact, are not adequate for comprehensive rotorcraft dynamic analysis as defined by Johnson.[7] The analysis methodology used in GRASP, although a viable approach for application to nonlinear dynamics in forward flight, would require considerable effort to be implemented in GRASP. Such an effort is not currently planned.

The main flexible-body element is the *aeroelastic beam*, which is an elastic, variable-order, kinematically nonlinear beam element that may be subject to inertial, gravitational, and aerodynamic loads. The beam static equations are never written out explicitly, but rather are formed from simple hierarchical expressions in terms of force and moment stress resultants that are obtained from the principle of virtual work. Although it is assumed in the equations that the strains are small relative to unity, there are no small-angle assumptions in the beam equations, nor is there any truncation of kinematically nonlinear effects through an ordering scheme. The element dynamic matrix coefficients are formed from numerical quadrature of exact linearized equations.

Certain features of a general-purpose code that, while not actually requirements, are very desirable have been incorporated in GRASP. 1) The user is able to increase the accuracy of the analysis without having to add more elements. The *aeroelastic beam* finite element developed for GRASP uses a variable-order (or "p-version") approach, which is based on high-order polynomial displacement functions.[14,15] 2) The equations of motion are formed as much as possible by the program itself, minimizing the possibility of errors in the equations. 3) GRASP has a user interface capable of handling the required generality without the user's having to know the form of the equations of motion or even the number of degrees of freedom. 4) GRASP is able to model both large and small problems with the same code. Thus, the number of degrees of freedom is not fixed *a priori*. This not only implies the need for a great deal of flexibility in assembling the equations of motion for the system, but also a need to manage data in core with a flexibility not inherent in the FORTRAN language.

Early in the development of the code, the requirement that the number of elements and degrees of freedom be kept arbitrary determined that the structuring and managing of data in the code be accomplished in such a way that the sizes of data structures be established from the input data. The philosophy of in-core data management adopted for the present development is discussed in Reference 16. The program possesses a library of routines called the Information Manager which was designed to support high-level management of data structures.

The purpose of this paper is to introduce GRASP to the technical community. The following subjects will be discussed: the design of the program, its capabilities, numerical results correlating GRASP with an existing special-purpose program and with experimental data, and planned enhancements

Program Design

In order to fulfill all of the requirements for a program with the degree of generality specified in the introduction, a modern approach to the design of the program was needed. Modern techniques were used in the design of the method of modeling structures as well as for the design of the software itself.

Software Design

The GRASP program was written using modern program design methods. GRASP is written almost exclusively in ANSI-standard FORTRAN 77 with machine dependencies isolated to a few of the lowest-level routines. The primary principle guiding the design and implementation of GRASP is clarity. At the implementation level, clarity is enhanced by adhering to coding standards that include extensive use of comments, definition of all variables, structured coding, and format conventions related to indentation, spacing, and overall subroutine structure.

Clarity is built in at the design level by emphasizing modularity. As much as possible, subroutines and data structures are designed to serve a single purpose. The same is true for larger packages of subroutines. The effects of modularity are evident at different levels. 1) *Higher-level Modularity*. At the highest level, the software is composed of two programs: one (called Build Input) that builds an input file in an internal format, and another (GRASP) that performs the desired computations. Each of these programs is composed of a main library and several supporting libraries. There are supporting libraries for handling errors, managing information structures, providing utility functions and providing high-level mathematical functions for information structures. (These high-level libraries correspond to program and relocatable libraries supported by the computer operating system.) 2) *Lower-level Modularity*. At a lower level, modularity is evident in the division of the main library of GRASP into a package of executive routines and a collection of packages for carrying out the requested computations. These packages can be thought of as forming a library of solution methods. Similarly, there are collections of routines associated with each of the elements and constraints that can be thought of as forming an element library and a constraint library. (These low-level libraries are imbedded in the main high-level library and do not directly correspond to computer operating system program or relocatable libraries.)

Dist Special
A-1

The GRASP Information Manager is an important part of the overall design of the program in several ways: 1) *Modularity*. The Information Manager library increases the modularity of the entire program by removing the data management functions from the main library of computational subroutines into a library of subroutines dedicated to a specific set of functions. 2) *Natural Data Structures*. As discussed in Reference 16, there is a natural association between the representation of a structural model and certain relational organizations of the data referred to as data structures. Information Manager provides a collection of information structures, including arrays, matrices, stacks, queues, lists, trees, and variable length tables, that can be conveniently used to perform calculations related to the structural model. 3) *Dynamic Data Structures*. The requirement that the number of elements and the number of degrees of freedom be kept arbitrary dictated that the sizes of data structures (*e.g.*, the dimensions of matrices) had to be problem dependent, and hence could not be determined until the data structures were ready to be created. Reference 16 discusses methods for dealing with the requirement for dynamic data structures. A comprehensive approach is used in GRASP, with all major data structures residing in a block of core that is located beyond the address of the last word of the program. Besides containing all of the data used by the program, this area also contains all of the tables, pointers, and records necessary to keep track of the data and operate on it.

Model Representation

A primary requirement in the representation of any structure is the ability to write the full, nonlinear equations of motion for bodies which may be experiencing large kinematic motions relative to one another. The basic approach used in GRASP is borrowed from the spacecraft-oriented research described in Reference 17, with additional emphasis on multiple levels of substructures.

First, a frame of reference for the problem is established (*e.g.*, an inertially-fixed frame for a hovering helicopter), and the complete system being modeled, which is called the model or a *model-type* subsystem, is associated with it. The model is then thought of as being composed of a number of subsystems, each of which in turn may be composed of a set of subsystems. The process terminates when a subsystem consists of a single finite element, which has no subsystems. The result of this method of modeling a system is an hierarchically-ordered set (tree) of subsystems with the model at the root, and an element, or *element-type* subsystem, at each of the leaves. With this modeling scheme, a subsystem may have many subordinate (child) subsystems but only one superordinate (parent) subsystem. Any subsystem that is neither at the root nor at a leaf is called a *system-type* subsystem.

A *subsystem* consists of a frame of reference, a collection of child subsystems, a collection of nodes, and a collection of constraints. The subsystem performs several functions. First, it is the basic unit in the hierarchical scheme, containing the complete definition of a model for the portion of the system it represents. Second, it is a repository for the degrees of freedom associated with that portion of the system. This includes its frame and nodal degrees of freedom as well as the independent generalized coordinates of any child subsystems. Finally, through its constraints, it is associated with the computational process of transforming between the generalized coordinates of its parent and its own generalized coordinates. Both the generalized displacements and forces are transformed. For perturbation problems, the coefficient matrices are transformed.

An *element* is a special subsystem which has no child subsystems. It may also have additional non-nodal degrees of freedom (*e.g.*, the *aeroelastic beam* internal degrees of freedom). Computationally, the elements are the source of virtual work in the problem. An element provides the generalized forces given the generalized displacements. For perturbation problems, an element provides the coefficient matrices providing the perturbation in generalized forces associated with perturbations in the generalized coordinates and their time derivatives.

The *frame of reference* for a subsystem provides the primary means of establishing the coordinate system for that subsystem. It also introduces six (independent until constrained) degrees of freedom which define the position and orientation of the subsystem's frame relative to its parent's frame. The position and orientation of a frame may be selected to define a natural coordinate system for the subsystem (*e.g.*, a hub-centered frame for a helicopter rotor).

There are two types of nodes used in GRASP. *Structural nodes* provide measures for local displacement and rotation of the structure. *Air nodes* determine the induced inflow velocity (only one *air node* is used per rotor). Nodes are local to a subsystem and create degrees of freedom for that subsystem. Their only connection with other subsystems is through constraints. Nodes provide the physically identifiable points that form the basis for connectivity in the model. Elements are connected to nodes, and the physical constraints restrict the motion at one or more nodes.

Constraints can be thought of as the glue that holds the model together. Constraints are used to model both physical constraints (*e.g.*, pins or clamps) and to eliminate all of the dependent degrees of freedom introduced into the model. For example, if two frames of reference are to move as if they were rigidly connected to one another, a constraint is required to eliminate the dependent frame degrees of freedom. As another example, the assembly of a finite-element model may require a *structural node* in some subsystem to be connected to an element. This is accomplished by constraining the nodes in the subsystem and in the element to move as if they were rigidly connected. The set of constraints for a subsystem must be sufficient to reduce the total number of degrees of freedom to only the independent

degrees of freedom. Similarly, for the complete model, all dependent degrees of freedom must be eliminated.

In the course of building a model, GRASP routinely generates additional internal degrees of freedom (internal in the sense that they are not specified by the user). When it does so, it also generates internal constraints to eliminate the additional dependent degrees of freedom. For example, if the user specifies a constraint between a node in a grandparent subsystem and a node in a grandchild subsystem, GRASP will create an internal node in the parent subsystem. GRASP will also create internal constraints (*e.g.*, between the node in the grandparent subsystem and the node in the parent subsystem). (Actually, more than one internal node and internal constraint are produced. However, the details are beyond the scope of this paper.)

Solution and Modeling Capabilities

As noted in the introduction, GRASP was developed to provide a tool for determining the equilibrium deflections and stability of arbitrary rotorcraft configurations in hover or vertical flight (with no periodic coefficients or forces). The representation selected for the model allows great generality in the configurations that can be analyzed and permits essentially arbitrary kinematic motions of components relative to one another. This general framework along with the library-oriented software design means that the detailed capabilities and limitations of the program are those associated with the members of the libraries. The capabilities and restrictions associated with the solution and modeling libraries are described below. This is followed by an example of GRASP modeling.

Solutions

The solution library presently contains two solutions related to the hovering or vertical flight conditions (for which there are no periodic coefficients in the equations of motion):

Solve Steady-State. The steady-state solution provides values for all of the model degrees of freedom which result in model equilibrium. The primary requirement is that the model correspond to a physical system which admits a time-invariant, steady-state solution. That is, the structure must not be subject to time-varying forces. This may be accomplished by requiring that any rotating substructure be rotationally isotropic and have a constant angular velocity, that the gravity vector be parallel to the axis of rotation, and that the axis of symmetry for the axisymmetric flow field be coincident with the axis of rotation.

This solution uses the IMSL[18] subroutine ZXSSQ, to minimize the sum of the squares of the residuals in the equations of motion using a Levenberg-Marquardt algorithm. Actually, the subroutine is used at two levels. It is used at an outer level to iterate to a solution of the equations of motion of the complete model, excluding the equations for the *aeroelastic beam* internal degrees of freedom. A separate, inner-level call to the subroutine is made for each *aeroelastic beam* element within each residual evaluation of the outer level to obtain the solution for the internal degrees of freedom in that *aeroelastic beam*.

Solve Asymmetric Eigenproblem. The asymmetric eigensolution provides the complex eigenvalues and eigenvectors for all model degrees of freedom associated with the equations of motion, $M\ddot{q}+C\dot{q}+Kq = 0$, linearized about a steady-state deformation. The term "asymmetric" refers to the nonsymmetry of the coefficient matrices C and K. The coefficient matrix M, which is both symmetric and positive-definite, contains contributions from the mass of the structural model and from the "apparent mass" of the air. The coefficient matrix C contains contributions from structural and aerodynamic damping and inertial forces. The coefficient matrix K contains contributions from structural stiffness and effective stiffness from aerodynamic and inertial forces. Currently, the asymmetric eigensolution must be computed by using the steady-state solution obtained for an identical model. This solution procedure prohibits one, for example, from obtaining the steady-state deformations of an isolated blade, then applying that solution to a coupled, rotor/fuselage configuration. Plans are being made to relax this restriction. Like the steady-state solution, this solution requires that the model correspond to a physical system which is not subject to time-varying forces.

This solution factors the matrix M using the IMSL routine LUDECP which decomposes a matrix using the Cholesky algorithm. The linearized, second-order equations of motion are transformed to provide an identity mass matrix. These linearized, second-order differential equations are cast into first-order form. The IMSL routine EIGRF is then used to obtain the eigenvalues and eigenvectors. EIGRF balances the dynamic matrix, converts it to Hessenberg form, and then uses the QR algorithm to obtain the eigensolution. Finally, the eigenvectors are transformed back to the original coordinate system.

Nodes

The GRASP node library currently contains two nodes:

Structural Node. The *structural node* introduces six degrees of freedom which define the change in position and orientation relative to its undeformed state, (which is defined as a fixed position and orientation relative to the subsystem reference frame). A *structural node* moves with the deformation of the structure. The *structural node* can be thought of as a massless, infinitesimal, rigid body that is physically attached to the structure being modeled at a specified point.

Air Node. The *air node* introduces four degrees of freedom that are associated with the flow of air through a rotor disk. The meaning of these degrees of freedom can be understood by looking at the distribution of inflow velocity that they produce: $q_1 + q_2 r + q_3 r \cos\psi + q_4 r \sin\psi$, where the q_i represent the degrees of freedom, r is the radial position in the axisymmetric flow field, and ψ is the azimuthal position. Thus, q_1 is a uniform, induced inflow velocity and q_2, q_3, and q_4 are induced inflow velocity gradients in the radial direction and cosine and sine cyclic directions, respectively. For steady-state problems, q_3 and q_4 are not involved. For the asymmetric eigenproblem, q_2 is not involved. The position and orientation of the flow field are assumed to remain inertially fixed.

Elements

The GRASP element library currently contains three elements: the *aeroelastic beam*, the *air mass*, and the *rigid-body mass*.

Aeroelastic Beam. The *aeroelastic beam* element represents a slender beam (without shear deformability) that is subject to elastic, inertial, gravitational, and aerodynamic forces. Distributed beam properties are permitted a cubic variation along the length of the element and are specified at the ends and at the one-third points. The primary assumption in the derivation of the element equations[19] is that strains remain small relative to unity. There are no small-angle approximations made and all kinematically nonlinear effects are included. One current limitation is that orientation angles[18] (of type body-three: 1–2–3) are used in the description of finite rotation inside the beam element. Thus, rotations due to the deformation of beam elements may not exceed 90°.

The element degrees of freedom consist of a frame of reference which coincides with the root of the element in its undeformed state, *structural nodes* at the root and tip, an *air node*, and internal degrees of freedom. The internal degrees of freedom result from using higher-order polynomials to increase the accuracy of the beam deformation calculations. This method of increasing the accuracy of an element is more convenient than having to add elements to increase the accuracy, and it also turns out to be more efficient[14] (given the same number of degrees of freedom). As the default condition, axial and torsional deformations (in excess of a built-in pretwist) are represented by linear polynomials, while bending deflections are represented by cubic polynomials. The additional degrees of freedom may be added selectively to reflect the dynamics of the element. For example, if a beam is very stiff in bending and extension but soft in torsion, additional torsional degrees of freedom may be added without having to include any more bending or extensional degrees of freedom.

The inertial section properties of the element include the mass per unit length and the first and second moments of mass about the principal structural axes (assumed to coincide with the principal inertial axes) for the elastic center. The geometric properties include the beam length (not distributed) and the pretwist. The elastic properties include the axial stiffness, the first area moments of inertia of axial stiffness, the bending stiffnesses, the torsional stiffness, and eight other section integrals. Also contained among the elastic properties is a structural damping parameter which is constant over the length of the beam.

The aerodynamic forces on the beam element are calculated from quasi-steady strip theory using lift, drag, and moment coefficients that are functions of angle of attack, and are read from tables. As these tables are defined by the data input to GRASP, they may be as general as piecewise cubics in the angle of attack. Spanwise scale factors for the lift, drag, and moment may also be specified (at the beam one-third points) to allow for tip loss or other similar effects. The chord width, the pitch angle of the zero-lift-line, and the offset of the aerodynamic center from the elastic axis allow for a cubic distribution over the length of the beam element, as do the structural section properties.

The *aeroelastic beam* element also calculates the blade element contributions to the *air node*. These contributions are combined with the momentum contributions from the *air mass* element during the solution process.

Air Mass. The *air mass* element models the momentum air flow through an axisymmetric rotor disk, and is defined by specifying the tip radius of the disk and the root cutout radius, if any. In this element, the frame degrees of freedom are suppressed because the frame must be inertial. The only degrees of freedom associated with the element are represented by a single *air node*. For steady-state problems, the residuals corresponding to q_1 and q_2 are calculated from momentum considerations,[20] while q_3 and q_4 are not involved. For the asymmetric eigenproblem, only the momentum terms[21] involving q_1, q_3, and q_4 contribute to the element coefficient matrices M and C, while q_2 is not involved.

Rigid-body Mass. The *rigid-body mass* element is defined by specifying its total mass and its principal moments of inertia about the mass center. The single *structural node* associated with this element is located at the mass center, and has its axes aligned with the principal axes. The frame of reference coincides with the node in the undeformed state.

Constraints

All of the constraints are based on purely kinematical relationships. There are no restrictions to small or moderate displacements or rotations in the constraint equations. However, it is necessary to avoid the singularity that occurs for deformation-induced rotations of 180° in the finite-rotational kinematics based on Rodrigues parameters.[22] (The finite-rotational kinematics of the *aeroelastic beam* are

somewhat different and have been discussed earlier.) The constraint library presently provides the user with 11 constraints. These constraints may be categorized as follows: frame constraints, node constraints, and element connectivity constraints.

Frame Constraints. The three types of frame constraints are *fixed frame*, *periodic frame*, and *rotating frame*. Each of these three constraints eliminates the child frame (dependent) degrees of freedom in favor of the parent frame (independent) degrees of freedom.

A *fixed frame* constraint permanently prescribes the position and orientation of a subsystem reference frame relative to its parent's reference frame.

If a subsystem is to be replicated N times about an axis of symmetry, a *periodic frame* constraint may be used. It is assumed, without loss of generality, that the x_1 axis of the parent frame is always the axis of symmetry. As for the fixed frame constraint, the position and orientation (except for the change in azimuth orientation) of each child frame relative to its parent is fixed.

The *rotating frame* constraint relates the degrees of freedom of a frame that is rotating at a constant angular velocity about its x_1 axis to the degrees of freedom of its parent frame. In a manner similar to the other two frame constraints, the position and orientation (except for the angular rotation) of the rotating frame is fixed relative to its parent.

Node Constraints. There are five types of constraints between nodes: *periodic structure*, *prescribed*, *rotating structure*, *rigid body*, and *screw*.

The *periodic structure* constraint performs the same function for a node as the periodic frame constraint performs for a frame. There is, however, an additional restriction that the independent node must be coincident with the independent subsystem reference frame.

The *prescribed* constraint sets a specified degree of freedom in a *structural node* to a prescribed value. By using a sequence of prescribed constraints, the user can create physical constraints such as clamps and pins. Prescribed constraints are also used to eliminate unnecessary *air node* degrees of freedom. During processing of the *air mass connectivity* constraint, these constraints are used internally to eliminate dynamic degrees of freedom from steady-state problems and static degrees of freedom from dynamic problems.

The *rotating structure* constraint relates the degrees of freedom of a rotating, dependent node to the degrees of freedom of a nonrotating, independent node. For this constraint, the dependent node must be coincident with the rotating reference frame. For both the periodic structure and rotating structure constraints, the axes of symmetry and rotation are associated with the parent and child frames, respectively, and not with the nodes.

The *rigid body* constraint forces two nodes to behave as if they are two points on a single rigid body.

In the *screw* constraint, the dependent node is only allowed to translate along and/or rotate relative to the independent node about a single, specified axis. Compliances (damping and stiffness) may be specified for both screw translation and rotation. By introducing nodes and using a sequence of screw constraints, the user can create physical constraints such as gimbals.

Element Connectivity Constraints. The element connectivity constraints are designed to connect an *element-type* subsystem to a *system-type* subsystem, and include the *aeroelastic beam connectivity*, the *air mass connectivity*, and the *rigid-body mass connectivity* constraints. Although they are specified by the user as a single constraint, GRASP actually generates a series of simpler constraints for each.

The *aeroelastic beam connectivity* constraint generates a *fixed frame* constraint between the element frame and the parent frame. It also generates internal constraints between the element root and tip *structural nodes* and the user-specified nodes in their respective subsystems. Finally, it generates an internal constraint relating the element *air node* to a user-specified *air node*.

The *air mass connectivity* constraint creates a *fixed frame* constraint between the element frame and the parent frame. Even though the element has no frame degrees of freedom, frame information is required to locate the position and orientation of the flow field center. An internal constraint is also generated which relates the element *air node* to a user-specified *air node*.

The *rigid-body mass connectivity* constraint generates a *fixed frame* constraint between the element frame and the parent frame. It also generates an internal constraint between the element center-of-mass node and a user-specified *structural node*.

GRASP Modeling Example

As an example of how GRASP may be used to model rotorcraft, consider the rotor-fuselage configuration investigated by Ormiston.[3,23] The physical model is shown in Fig. 1. It consists of a rigid fuselage that has pitch and roll degrees of freedom, and a rotor that has N (where $N > 2$) rigid blades which are rotating at a constant angular velocity. The blades are rigid, and are allowed to rotate about centrally-located, spring-restrained flap and lag hinges.

Fig. 2 shows the hierarchical set of subsystems used to represent the physical model. Under the name of each subsystem, and in parentheses, is the subsystem type. Table 1 summarizes the nodes and constraints associated with each of these subsystems. A description of each subsystem follows.

Subsystem PRBMODEL. At the root of the tree is the *model-type* subsystem PRBMODEL that represents the complete physical model. This subsystem is not explicitly defined in the hierarchy input to GRASP, but rather is generated by the program. The frame of reference is inertially fixed and is initially located at the fuselage center of mass with its x_1 axis vertical.

Subsystem NROT. The first explicitly defined subsystem is the *system-type* subsystem NROT, which represents the fuselage, inflow velocity field, and the rotor integrated into a single entity. The frame of reference associated with this subsystem is inertially fixed and is initially located at the fuselage center of mass, with the x_1 axis aligned with the gravity vector and inflow velocity field and the x_3 axis passing through the tail boom. This subsystem consists of two nodes and six constraints, and has three children.

The first node is the *structural node* NRNOD, which is initially coincident with the subsystem frame of reference. This is the node to which the *rigid-body mass* element representing the fuselage will be attached. The other node is the *air node* AMSDOF, which is located such that the axis of symmetry of the flow field is coincident with the rotor mast (the frame x_1 axis). This *air node* contains the degrees of freedom where the momentum and blade element contributions to the inflow distribution will be combined.

The first constraint defined for this subsystem is the *fixed frame* constraint FFR1, which specifies that the subsystem frame of reference is coincident with the model reference frame. *Prescribed* constraints PRE1 through PRE4 lock out all of the fuselage translations and yaw rotation, leaving only fuselage pitch and roll. PREAIR1, a *prescribed* constraint, eliminates the uniform inflow degree of freedom in the static solution and the collective dynamic inflow mode in the asymmetric eigensolution.

Subsystem FUSEL. The first child of subsystem NROT is FUSEL. This *element-type* subsystem is a *rigid-body mass* that represents the fuselage. There are no user defined nodes for this subsystem (however, GRASP generates the center-of-mass node internally). The only constraint required is the *rigid-body mass connectivity* constraint RMC1, which constrains the element center-of-mass node and the *structural node* NRNOD in subsystem NROT to be coincident. This has the effect of attaching a *rigid body mass* element to the *structural node* NRNOD.

Subsystem AMSS. The second child of subsystem NROT is AMSS, which is another *element-type* subsystem. AMSS is an *air mass element* that represents the induced velocity through the rotor disk. Again, although there are no user-defined nodes for an element, GRASP generates an *air node*. The *air mass connectivity* constraint AMC1, which is the only constraint required in the subsystem, connects the element *air node* with the *air node* AMSDOF in subsystem NROT. This has the effect of attaching an *air mass* element to the *air node* representing the induced inflow, AMSDOF.

Subsystem RROT. The third child of subsystem NROT is RROT, which is a *system-type* subsystem that represents an axisymmetric substructure consisting of N identical, equally-spaced rotor blades. In this subsystem, the reference frame, initially located at the top of the rotor mast, participates in the nominal rotation of the rotor. Subsystem RROT consists of one node and two constraints, and has only one child.

The only node defined for the subsystem is the *structural node* RRNOD, which is initially coincident with the subsystem frame of reference. Physically, this node can be thought of as being the center of the rotor system.

The frame constraint for this subsystem is the *rotating frame* constraint RFR1, which specifies that the frame of reference for subsystem RROT is offset from its parent reference frame by a distance along the x_1 axis equal to the rotor mast height. This constraint also specifies that the RROT reference frame is rotating at a constant angular velocity relative to its parent frame. The *rotating structure* constraint RST1 connects node RRNOD in this subsystem to node NRNOD in subsystem NROT. Since node RRNOD belongs to subsystem RROT and its frame of reference is rotating relative to its parent, RRNOD is also rotating relative to node NRNOD.

Subsystem RBLADE. Subsystem RBLADE is the only child of subsystem RROT, and represents a substructure consisting of one of the N identical, equally-spaced blades that make up the rotor system. The reference frame for this subsystem is initially located at the rotor hub center and has its x_1 axis aligned with the blade flapping direction and its x_3 axis aligned with the blade axis. This subsystem consists of one node and two constraints, and also has only one child.

The only node defined for this subsystem is the *structural node* HUBNOD, which is initially coincident with the subsystem frame of reference, and therefore is also located at the center of the rotor hub.

The first constraint for subsystem RBLADE is the *periodic frame* constraint PFR1, which specifies that the origin of the reference frame for RBLADE is coincident with the parent frame but allows a rotation about the x_3 axis to simulate flap-lag coupling. This constraint also specifies that there are N subsystems identical to RBLADE spaced equally about the x_1 axis of the parent subsystem (RROT) frame. The second constraint is the *periodic structure* constraint PST1, which connects the N copies of the node HUBNOD associated with subsystem RBLADE to node RRNOD in subsystem RROT.

Subsystem BLADE. The only child of subsystem RBLADE is *system-type* subsystem BLADE, which represents a single, generic rotor blade that includes a rigid blade

as well as flap and lag hinges. The subsystem frame of reference is initially at the hub center with the x_1 axis aligned in the flapping direction and the x_3 axis aligned with the blade. This subsystem consists of three nodes and four constraints, and has one child.

Structural nodes INTNOD and BLDR are initially coincident with the frame of reference, which is located at the blade root, while *structural node* BLDT is offset from the frame by a distance along the x_3 axis equal to the length of the beam element. Physically, BLDR and BLDT are located at the root and tip of the blade. INTNOD is coincident with BLDR and allows a second *screw* constraint to be used for the lag hinge.

The frame constraint for this subsystem is *fixed frame* constraint FFR2, which specifies that the frame of reference for subsystem BLADE is coincident with the parent frame. For this problem, the lag and flap hinges are defined using *screw* constraints. *Screw* constraint LAG permits node INTNOD to rotate about the x_1 axis of node HUBNOD (from subsystem RBLADE). Another *screw* constraint, FLAP, permits node BLDR to rotate about the x_2 axis of node INTNOD. The *rigid body* constraint RBC1 requires node BDLT to move as if it were rigidly connected to node BLDR. The effect of this last constraint is to establish the rigidity of the blade in the absence of *aeroelastic beam* internal degrees of freedom.

Subsystem BEAM. The only child of subsystem BLADE is BEAM, an *aeroelastic beam* element representing the rotor blade. As with the other *element-type* subsystems, no nodes need to be defined. The only constraint required is *aeroelastic beam connectivity* constraint ABC1, which connects the element root and tip nodes to *structural nodes* BLDR and BLDT in subsystem BLADE. It also connects the element *air node* to *air node* AMSDOF in subsystem NROT. Blade pitch angle is introduced by defining the orientation of the element root node relative to node BLDR to include a rotation about the x_3 (beam) axis.

Numerical Results

Two sets of numerical results from GRASP are presented. The first set is compared with output from an existing, reliable, special-purpose, coupled rotor/body aeromechanical stability program used by Ormiston to obtain the results presented in References 3 and 23. This problem exercises each member of GRASP's solution, node, element and constraint libraries. By recreating the model in a special-purpose program, it also illustrates GRASP's modeling flexibility. The second set of results is compared with experimental data[24–26] for static and dynamic behavior of an end-loaded cantilevered beam which undergoes large deflections. This problem emphasizes the accuracy of GRASP's beam element for large deflections.

Rotor-Fuselage Problem

GRASP results were obtained for the rotor/body model described in the previous GRASP modeling example. The model used a rotor height (above the body center of mass) of 0.2 times the rotor radius, the small-body inertia properties as given in Reference 23, and the following rotor/blade properties: solidity of 0.05, Lock number of 10, fundamental rotating flap and lag frequencies of 1.05 and 0.5 respectively, and drag coefficient of 0.0079. The present results contain body pitch and roll degrees of freedom, rotor blade flap and lead-lag, and, where specified, inflow dynamics. GRASP calculations were done with a large number of blades in the rotor model (holding solidity and Lock number constant) to suppress blade-element apparent-mass effects that are included in GRASP but omitted in Ormiston's program.

Eigenvalues were obtained for pitch angles of 0°, 5°, 10°, and 15°. The differences between the results are virtually imperceptible using normal scales for the real and imaginary axes of the root-locus plots. To amplify the differences as much as possible, exaggerated scales were selected to produce Figs. 3–7. As can be seen, the correlation at low thrust is excellent. Only the body mode in Fig. 6 shows a difference at zero thrust; this mode is quite close to the real axis where very slight changes in any physical parameters can produce large changes in the roots. Slight deviations as pitch angle increases can be seen in all the modes. The most noticeable deviations at large thrust were found in the body mode. These were caused by slightly different assumptions that were made in the structural and aerodynamic modeling of the two analyses. For example, in Ormiston's program small-angle assumptions were made for the spring-restraint moments and for pitch and inflow angles in the aerodynamics, while in GRASP exact kinematics were used. Clearly, small differences such as these in the modeling do produce small differences in the results. Also, the changes in the trends produced by the addition of dynamic inflow are virtually identical in the two programs.

Princeton Experiment

One of the first correlation attempts with GRASP started with a basic experiment carried out at Princeton University[24–26] (under Aeroflightdynamics Directorate sponsorship). This experiment consisted of measuring the static deformation and fundamental flatwise and edgewise natural frequencies of a uniform, nonrotating, cantilever beam with a mass attached to the tip (Fig. 8). The beam was slender and was sufficiently flexible to undergo large displacements (still at small strains) due to the presence of the tip mass. Beam load angle and mass of the tip weight were varied throughout appropriate ranges.

Beam Properties. The beam was made from 7075 aluminum (density 0.1014 lb/in^3) with a rectangular cross section of 0.1251 by 0.4999 in. and a length of 19.985 in. (See References 24–26 for a complete description of the appara-

tus and experimental methodology). The mass per unit length, derived from the density and cross-sectional dimensions, is 1.6424×10^{-5} lb-sec^2/in^2. The mass moments of inertias about the x_1, x_2, and x_3 axes are 2.1420×10^{-8} lb-sec^2-in., 2.1420×10^{-8} lb-sec^2-in., and 3.4204×10^{-7} lb-sec^2-in., respectively. The moments of inertia were estimated for the 2-lb. tip mass as $I_1 = I_2 = 0.0033676$ and $I_3 = 0.0026784$, and for the 3-lb. tip mass as $I_1 = I_2 = 0.0065673$ and $I_3 = 0.0053169$ where I_i is expressed in lb-sec^2-in.

Determining the appropriate values of bending stiffness proved to be more difficult. Both static and dynamic results are very sensitive to the value of the stiffnesses, and so great care was taken to model the stiffnesses as accurately as possible. The use of the standard value of the modulus of elasticity gives only fair results for both the statics and dynamics. Attempted inference of equivalent beam properties from classical linear formulas for deflection versus load for the two uncoupled cases, load angles of 0° and 90°, yields contradictory information. At a load angle of 0° (the edgewise-bending case), linear theory is too stiff, but at a load angle of 90° (the flatwise-bending case), linear theory is too soft. This seems to suggest that there is no one value of E that will yield accurate flatwise and edgewise bending stiffnesses. Also, the experimental data do not show any deflection without a tip mass, although some deflection should have been measurable, at least in the flatwise-bending case.

With these observations in mind, equivalent beam bending stiffnesses (*i.e.*, EI's) were deduced for flatwise and edgewise bending entirely from uncoupled (load angles of 0° and 90°) static data where the experimental data were assumed to have deflections for no tip mass subtracted out. This latter point is not explicitly stated in References 24–26 but from the present investigation appears to be true. A simple planar elastica[27] model was derived in which an equation for tip deflection as a function of tip mass and bending stiffness, EI, was produced. A least-squares method was used to find the best EI to fit this theoretical model to the experimental data for both the flatwise and edgewise directions. The two values of E inferred from the bending stiffnesses and the cross section geometry were averaged and multiplied by the cross sectional area to obtain the axial stiffness. A value of Poisson's ratio equal to 0.31 was assumed and the shear modulus, G, was inferred from E. The torsional section constant reported in Reference 26 was used. The following stiffnesses resulted: axial stiffness = 6.2856×10^5 lb-in^2; flatwise stiffness = 8.4487×10^2 lb-in^2; edgewise stiffness = 1.2689×10^4 lb-in^2; torsional stiffness = 1.0538×10^3 lb-in^2.

GRASP Model. The GRASP model for the Princeton experiment is depicted in Fig. 9. Subsystem PRNCTN, the *model-type* subsystem generated internally by GRASP, represents the complete structure. The first explicitly defined subsystem is CANTBEAM. The frame of reference is defined to be coincident with the model frame except for a rotation about the x_3 axis, which is interpreted as the beam load angle. The subsystem contains two *structural nodes* named ROOT and TIP. ROOT is coincident with the CANTBEAM subsystem frame of reference, and has all of its degrees of freedom prescribed to zero (cantilever beam boundary conditions). TIP is defined to be located 19.985 in. from the frame along the x_3 axis.

The first child of CANTBEAM is an *aeroelastic beam* element named BLADE. An *aeroelastic beam connectivity* constraint associates the elements root and tip nodes with the nodes ROOT and TIP in the subsystem CANTBEAM. The definition of the element includes specifying the orders of the polynomials used to represent the displacements. The typical approach in finite element programs would be to use several elements with the transverse displacements approximated by cubic polynomials and the axial displacement and torsion approximated by linear polynomials. Instead, for this analysis we use *one element* with eighth-order polynomials for bending and sixth-order polynomials for axial displacement and torsion. This yields a total of 32 element degrees of freedom (6 of which will be constrained out by the clamped-end condition). Essentially the same results were obtained when the order of each polynomial was reduced by one.

Subsystem WEIGHT, the second child of CANTBEAM, is a *rigid-body-mass* element that is defined to be coincident with the node TIP. Its definition specifies the mass and the mass moments of inertias about all three principal axes.

GRASP Results. GRASP expresses static rotations in terms of Rodrigues parameters, so a minor amount of postprocessing is needed to convert the GRASP output to the projected angle which was measured in Reference 25. Also, all GRASP deflections had the no-load deflections subtracted out before the results were plotted with the experimental data. All frequencies calculated by GRASP were converted from rad/sec to Hz.

Two tip-loading conditions are presented here. First, results are presented for a 2-lb. tip mass. Fig. 10 shows the static deflections vs. load angle. The correlation for flatwise and edgewise is excellent. For torsional deflection the GRASP calculations cut right through the middle of the experimental scatter. Fig. 11 displays the flatwise and edgewise frequencies vs. load angle. The GRASP results are only slightly offset from the experimental values, and follow the trend exactly. The average error is approximately 0.5%.

The 3-lb. tip mass results are presented next. Again, excellent correlation with the static deflection is shown in Fig. 12. Notice the slight rise in edgewise deflection with load angle around 30° shown by both the experiment and GRASP. The torsional data have much less scatter than in the 2-lb. case. Again, GRASP calculations correlate excellently. Fig. 13 shows the flatwise and edgewise frequencies. The GRASP predictions are again slightly low for both of the frequencies, but follow the trends very nicely.

Overall, GRASP does a much better job of determining both the static and dynamic the behavior of the beam under load than the theory used in Reference 26. This is primarily due to the use in GRASP of the exact small-strain kinematics modeled by the beam element equations of Reference 20.

Planned Enhancements

Since GRASP is composed of libraries of solutions, elements, and constraints, enhancements which extend the capabilities of GRASP result from adding new members to or improving the existing members of these libraries. The following sections describe the enhancements which are planned to extend the capabilities of the libraries. Other enhancements are also described, which are not intended to add new capabilities but will make the program easier to use.

Enhancements to the Libraries

Solution Library. In the solution library, the planned enhancements include adding a symmetric eigensolution and a reference deformations solution. The symmetric eigensolution will provide the user with the capability of using a significantly faster eigenvalue utility to obtain a set of approximate eigenvectors for any portion of the model. These approximate eigenvectors can then be used to obtain a reduced set of generalized coordinates for the *sub-space reduction* constraint (see below).

The reference deformations solution will relax the restriction to using identical models for the asymmetric eigensolution and the steady-state solution. It will make it possible to specify an arbitrary deformation state for any portion of the model. For example, the deformations of an isolated rotor blade could be calculated using the steady-state solution, then used to specify the deformation state of a portion of a more complex structure.

Element Library. Planned additions to the element library include a *direct input* element and a *composite aeroelastic beam* element. The *direct input* element will allow creation of an element that is defined by the linear equation $M\ddot{q}+C\dot{q}+Kq=0$. This element will provide the capability to model simple springs and dashpots as well as the ability to represent a portion of the model in terms of its modes and frequencies.

The *composite aeroelastic beam* element will provide the same basic capability as the *aeroelastic beam* element, but will more accurately represent the behavior of beams built up from composite materials. Section rotations from independent polynomials will be used to obtain a more general deformation field and Rodrigues parameters will be used in order to obtain a simpler formulation and an increased range of rotations due to deformation.[28] Also, a more general constitutive law will also be used in conjunction with an increased number of independent elastic constants.

Constraint Library. Five additions to the constraint library are planned. Two of these will provide connectivity for the new *direct input* and *composite aeroelastic beam* elements described above. The three other additions to the constraint library include a *pin* constraint, a *moving frame* constraint, and a *sub-space reduction* constraint. The *pin* constraint will allow the user to define a pin connection between two nodes with a single constraint.

The *moving frame* constraint will associate a frame and a *structural node* such that the frame follows the motion of the node, providing a more natural frame for measuring motion in a subsystem. Also, this constraint will make it possible to treat the rotating/nonrotating interface in a more general fashion.

The *sub-space reduction* constraint will be used to reduce the number of degrees of freedom in a model by transforming from the coordinate space defined by the original degrees of freedom to a sub-space of generalized coordinates whose basis is specified by a linear transformation. Typically the linear transform will be the eigenvectors from a symmetric eigensolution performed on that subsystem.

Other Enhancements

A number of enhancements have been planned to increase the useability of the program. These enhancements include implementing improved error messages that include data that can better help pinpoint problems. Also, an interactive preprocessor is planned that will provide much of the "boilerplate" for the input file and prompt the user for values of the input data. Finally, an interactive postprocessor is planned that will provide automatic generation of plots and improved tabular output of results.

Concluding Remarks

In response to the limitations of previous methods for analyzing rotorcraft, GRASP has been developed. GRASP is a general-purpose program which treats the nonlinear static and linearized dynamic behavior of rotorcraft represented by arbitrarily connected rigid body and beam elements. Large relative motions and deformation-induced displacements and rotations are permitted (as long as the strains in the beam element are small). Periodic coefficients are not treated, restricting the solutions to rotorcraft in axial flight and ground contact conditions. Time-history solution procedures are not considered.

GRASP uses a modern approach for modeling structures, incorporating the features of several traditional methods. The basic approach which provides the foundation for large relative motion kinematics is derived from "multibody" research with an expanded emphasis on multiple levels of substructures. This is combined with the finite

element approach which provides flexible modeling through the use of libraries of elements, constraints, and nodes. The use of a variable-order polynomial beam element makes the finite element approach more effective. The incorporation of aeroelastic effects, including inflow dynamics and nonlinear aerodynamic coefficients for the beam element, further extends the capabilities.

One feature of the program which is especially important for rotor dynamics is the comprehensive treatment of the nonlinearities associated with large, deformation-induced displacements and rotations. No kinematical approximations are made on the magnitudes of the displacements and rotations. However, the strains in the *aeroelastic beam* element must remain small compared to unity, and shear deformation effects are not considered in the present element.

GRASP also uses a modern approach for programming. The design and implementation of the code emphasize ciarity and modularity at every level. One important feature is the use of an information manager which facilitates use of data structures that are natural for the problem and allows the dimensions of the data structures to be established based on the requirements of the particular model being analyzed.

The flexibility of GRASP has been demonstrated by using it to generate a model which essentially duplicates an existing, special-purpose, coupled rotor/body, aeromechanical stability program. The GRASP results correlate very well with those of the special-purpose program. The accuracy of the GRASP solution methods and the powerful capabilities of the *aeroelastic beam* element have been demonstrated by comparing the results from a simple single-element model with experimental results for an end-loaded cantilevered beam that is undergoing large deflections. The agreement between the calculated and experimentally measured results is excellent.

All of the basic features of GRASP described in this paper, excluding the planned enhancements, have been implemented in the current version of the code and are presently being subjected to extensive testing. Modular testing of the code has been conducted in parallel with development since the beginning of the project. As a particular system capability was developed, that capability was also tested to greatest extent possible at that time. Since the completion of the coding for the current version in September 1985, most of the effort by the project team has been concentrated in the area of system testing. Another major emphasis at the present time is the development of a theoretical manual for GRASP. This manual will provide detailed analytical derivations for the theoretical basis of GRASP in a single source. The current version of the code (written for the CRAY X-MP), a users' manual, and a programmers' manual are presently available.

Acknowledgements

Analysis development was initiated by the first author in April 1980. The second author began design in September 1980, and initiated the design of the code. Actual coding began in December 1980. The third author joined the project team in June 1982 and the fourth in January 1985. The first author served as project manager and chief analyst, the second as chief designer and analyst, and the third and fourth as programmers and analysts. After approximately 20 man-years of work, the current version of GRASP consists of over 170,000 lines of heavily-commented FORTRAN 77 code in over 900 subroutines. The development of GRASP was a team effort. Significant technical support by Robert A. Canfield and Steven L. Pucci is gratefully acknowledged. Program configuration control by Dr. Victor J. Alesi, Selvina J. Samson, Nghia T. Ton, Nhung T. Duong, and Anna Chou is acknowledged along with documentation and conversion support from Laura B. Biggs, Darryl L. Wooten, Howard F. Cunningham, Charlotte McGary, and Donna Diebert.

References

[1]Ormiston, R. A., and Hodges, D. H., "Linear Flap-Lag Dynamics of Hingeless Helicopter Rotor Blades in Hover," *Journal of the American Helicopter Society*, 17, (2), Apr. 1972, pp. 2–14.

[2]Hodges, D. H., and Dowell, E. H., "Nonlinear Equations of Motion for the Elastic Bending and Torsion of Twisted Nonuniform Rotor Blades", NASA TN D–7818, 1974.

[3]Ormiston, R. A., "Aeromechanical Stability of Soft In-plane Hingeless Rotor Helicopters," Paper No. 25, Third European Rotorcraft and Powered Lift Aircraft Forum, Aix-en-Provence, France, Sept. 1977.

[4]Hodges, D. H., "An Aeromechanical Stability Analysis for Bearingless Rotor Helicopters," *Journal of the American Helicopter Society*, 24, (1), Jan. 1979, pp. 2–9.

[5]Davis, J. M., Bennett, R. L., and Blankenship, B. L., "Rotorcraft Flight Simulation with Aeroelastic Rotor and Improved Aerodynamic Representation," USAAMRDL TR 74–10, June 1974.

[6]Bielawa, R. L., "Aeroelastic Analysis for Helicopter Rotor Blades with Time Variable Nonlinear Structural Twist and Multiple Structural Redundancy — Mathematical Derivation and Program User's Manual," NASA CR–2368, 1976.

[7]Johnson, W., "Assessment of Aerodynamic and Dynamic Models in a Comprehensive Analysis for Rotorcraft," *Computers and Mathematics with Applications*, 12A, (1), Jan. 1986, pp. 11–28.

[8]Hurst, P. W., and Berman, A., "DYSCO: An Executive Control System for Dynamic Analysis of Synthesized Structures," *Vertica*, 9, (4), 1985, pp. 307–316.

[9]Friedmann, P., and Straub, F., "Application of the Finite Element Method to Rotary-Wing Aeroelasticity," Paper No. 24, Fourth European Rotorcraft and Powered Lift Aircraft Forum, Stresa, Italy, Sept. 1978.

[10]Sivaneri, N. T., and Chopra, I., "Dynamic Stability of a Rotor Blade Using Finite Element Analysis," *AIAA Journal*, 20, (5), May 1982, pp. 716–723.

[11]Magnus, K., (ed.), "Dynamics of Multibody Systems," IUTAM Symposium, Munich, Germany, Aug. 29–Sept. 3, 1977.

[12]Kane, T. R., and Levinson, D. A., "Multibody Dynamics," *Journal of Applied Mechanics*, 50, Dec. 1983, pp. 1071–1078.

[13]Rosen, A., and Friedmann, P., "The Nonlinear Behavior of Elastic Slender Straight Beams Undergoing Small Strains and Moderate Rotations," *Journal of Applied Mechanics*, 46, March 1979, pp. 161–168.

[14]Hodges, D. H., and Rutkowski, M. J., "Free-Vibration Analysis of Rotating Beams by a Variable-Order Finite Element Method," *AIAA Journal*, 19, (11), November 1981, pp. 1459–1466.

[15]Szabo, B. A., "Some Recent Developments in Finite Element Analysis," *Computers and Mathematics with Applications*, 5, 1979, pp. 99–115.

[16]Kunz, D. L., and Hopkins, A. S., "Structured Data in Structural Analysis Software", Paper No. 85–0742, Proceedings of the 26th Structures, Structural Dynamics and Materials Conference, Orlando, Florida, Apr. 1985, pp. 480–493.

[17]Hopkins, A. S., "The Motion of Interconnected Flexible Bodies," Doctoral Dissertation, School of Engineering and Applied Science, University of California at Los Angeles, UCLA–ENG–7513, Feb. 1975.

[18]The International Mathematical and Statistical Library, IMSL Inc., Houston, Texas, 1984, Ch. E, L, and Z.

[19]Hodges, D. H., "Nonlinear Equations for Dynamics of Pretwisted Beams Undergoing Small Strains and Large Rotations", NASA TP–2470, 1985.

[20]Gessow, A.,and Myers, G. C., *Aerodynamics of The Helicopter*, Frederick Unger Publishing Company, New York, 1967, pp. 67–68.

[21]Pitt, D. M., and Peters, D. A., "Theoretical Predictions of Dynamic Inflow Derivatives," *Vertica*, 5, (1), March 1981, pp. 21–34.

[22]Kane, T. R., Likins, P. W., and Levinson, D. A., *Spacecraft Dynamics*, McGraw-Hill, 1983, Ch. 1.

[23]Ormiston, R. A., "Rotor-Fuselage Dynamic Coupling Characteristics of Helicopter Air and Ground Resonance," Proceedings of The Theoretical Basis of Helicopter Technology, Nanjing Aeronautical Institute, Nanjing, China, Nov. 6–8, 1985.

[24]Dowell, E. H., and Traybar, J., "An Experimental Study of the Non-linear Stiffness of a Rotor Blade Undergoing Flap, Lag and Twist Deformations," AMS Report No. 1194, Princeton University, Jan. 1975.

[25]Dowell, E. H., and Traybar, J., "An Experimental Study of the Non-linear Stiffness of a Rotor Blade Undergoing Flap, Lag and Twist Deformations," AMS Report No. 1257. Princeton University, Dec. 1975.

[26]Dowell, E. H., Traybar, J., and Hodges, D. H., "An Experimental-Theoretical Correlation Study of Non-linear Bending and Torsion Deformations of a Cantilever Beam," *Journal of Sound and Vibration*, 50, (4), 1977, pp. 533–544.

[27]Love, A. E. H., *A Treatise on the Mathematical Theory of Elasticity*, Fourth Ed., Dover Publications, New York, 1944, pp. 381–426.

[28]Hodges, Dewey H., "Finite Rotation and Nonlinear Beam Kinematics," *Vertica*, to be published, 1986.

Table 1 Subsystem nodes and constraints for the rotor-fuselage configuration.

Subsystems	Nodes	Constraints
PRBMODEL		
NROT	NRNOD, AMSDOF*	FFR1, PRE1–PRE4, PREAIR1
FUSEL		RMC1
AMSS		AMC1
RROT	RRNOD	RFR1, RST1
RBLADE	HUBNOD	PFR1, PST1
BLADE	INTNOD, BLDR, BLDT	FFR2, LAG, FLAP, RBC1
BEAM		ABC1

*air node

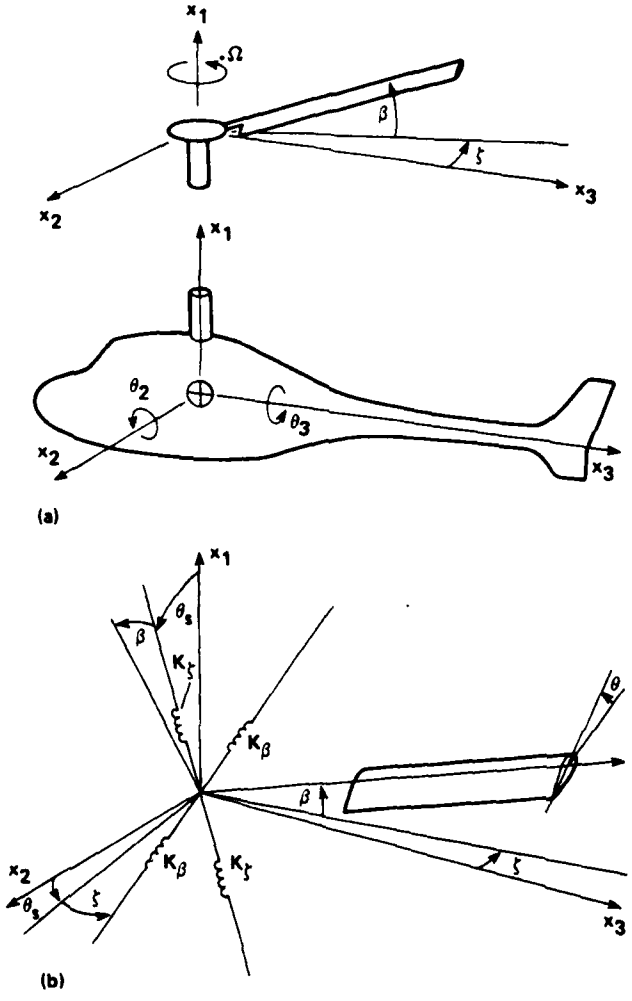


Fig. 1 Schematic of sample rotor-fuselage configuration, a) rotor and fuselage, b) individual blade.

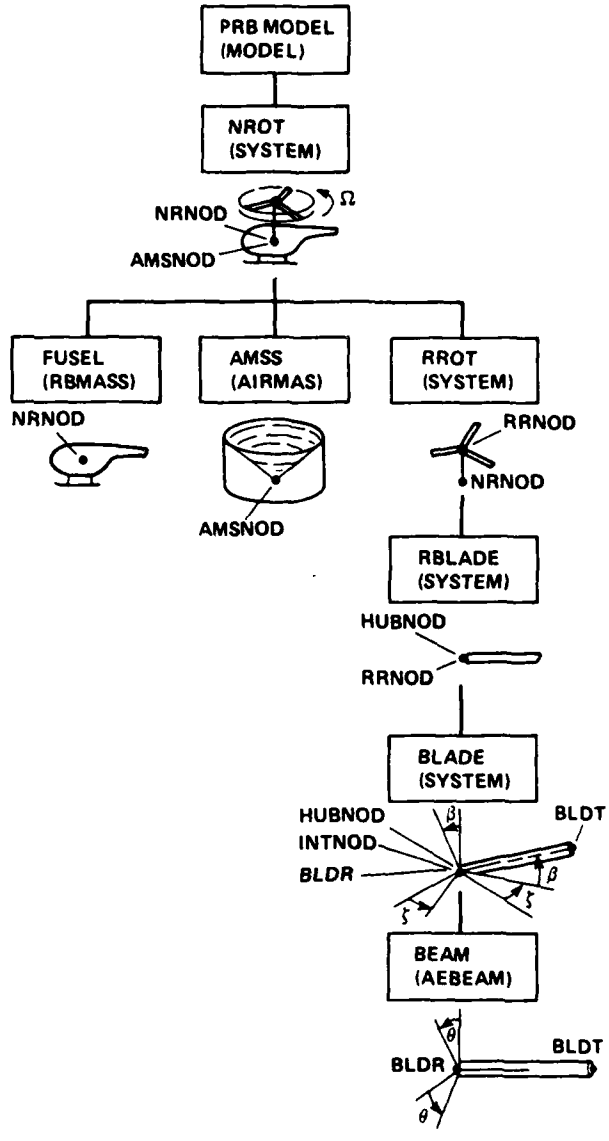


Fig. 2 Hierarchical GRASP model of the rotor-fuselage configuration.

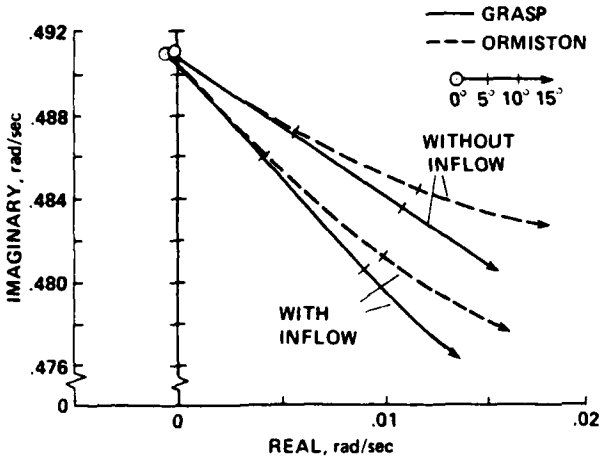


Fig. 3 GRASP correlation with Ormiston's coupled rotor/body model: behavior of the regressing lead-lag mode with and without inflow dynamics.

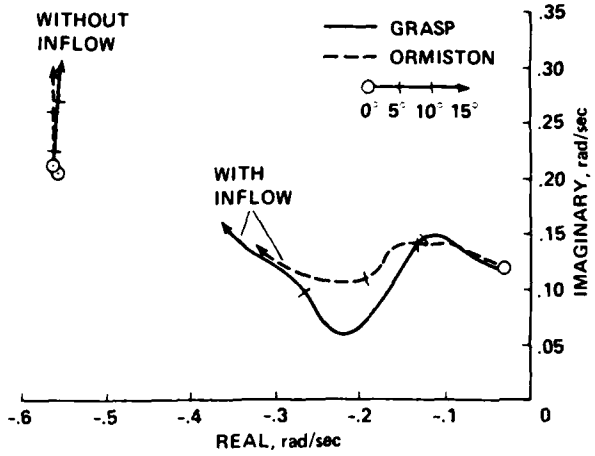


Fig. 5 GRASP correlation with Ormiston's coupled rotor/body model: behavior of the regressing flap mode with and without inflow dynamics.

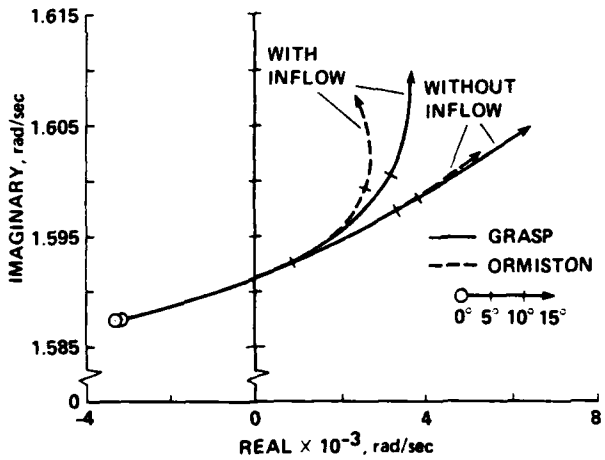


Fig. 4 GRASP correlation with Ormiston's coupled rotor/body model: behavior of the progressing lead-lag mode with and without inflow dynamics.

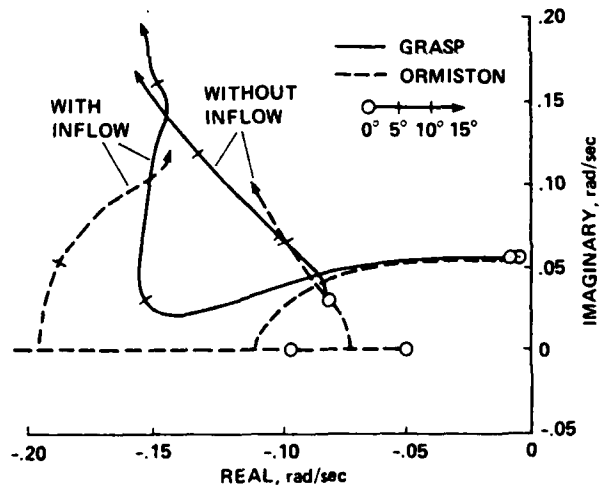


Fig. 6 GRASP correlation with Ormiston's coupled rotor/body model: behavior of a body mode with and without inflow dynamics.

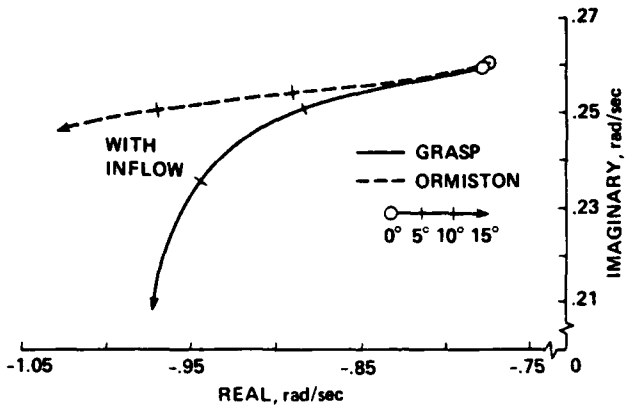


Fig. 7 GRASP correlation with Ormiston's coupled rotor/body model: behavior of the inflow mode.

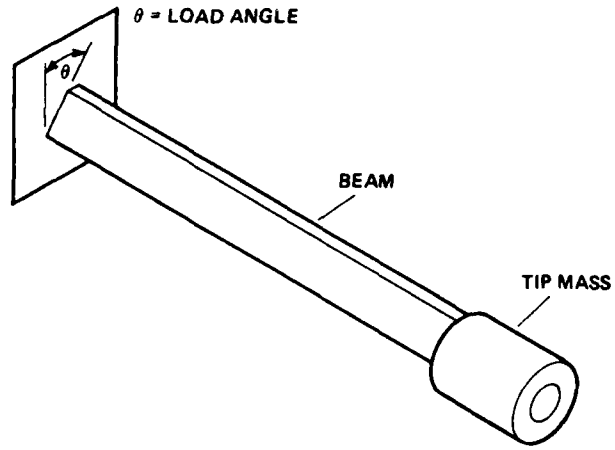


Fig. 8 Schematic of the Princeton beam experimental apparatus.

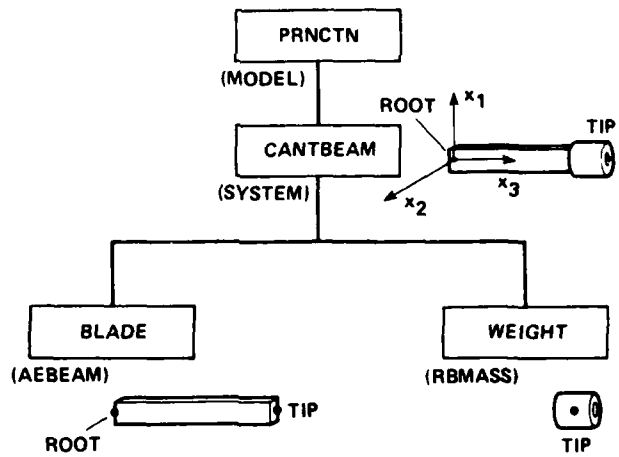


Fig. 9 Hierarchical GRASP model of the Princeton beam.

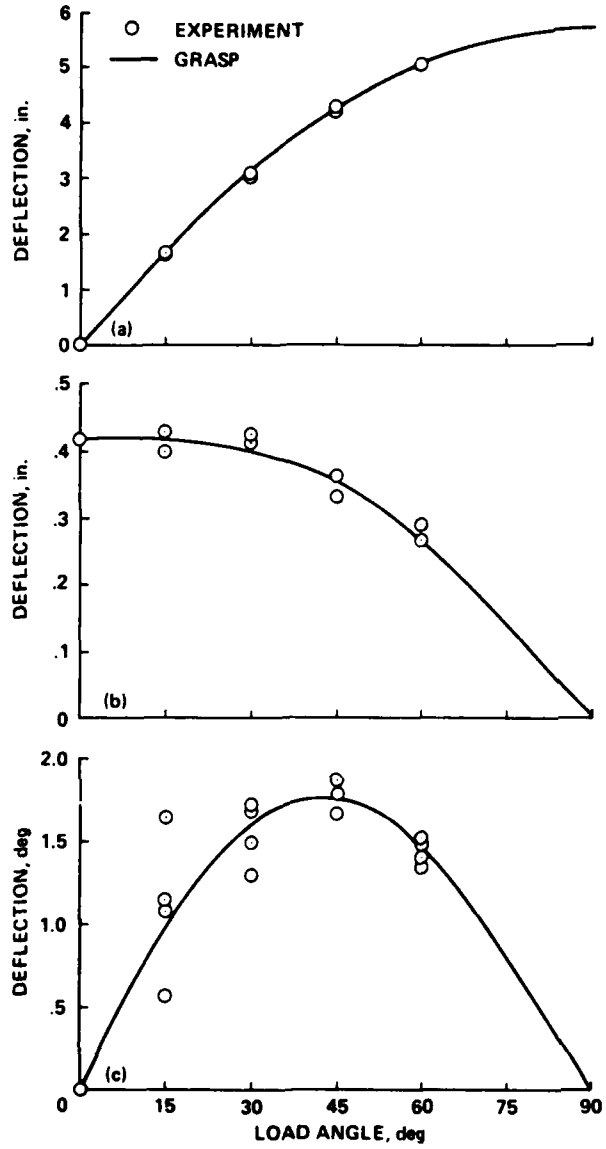


Fig. 10 GRASP correlation with the Princeton experiment (2-lb. tip mass): static deflections, a) flatwise, b) edgewise, c) torsional.

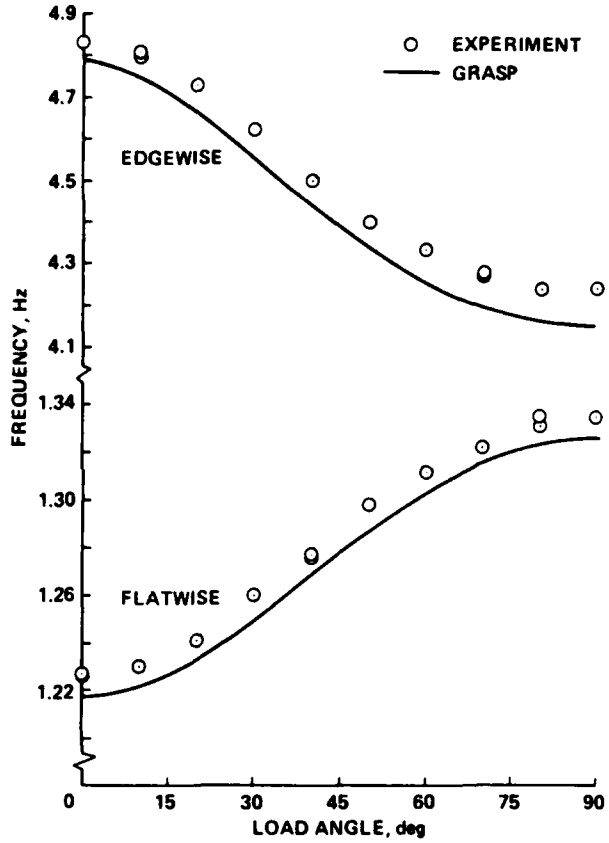


Fig. 11 GRASP correlation with the Princeton experiment (2-lb. tip mass): first flatwise and first edgewise frequencies.

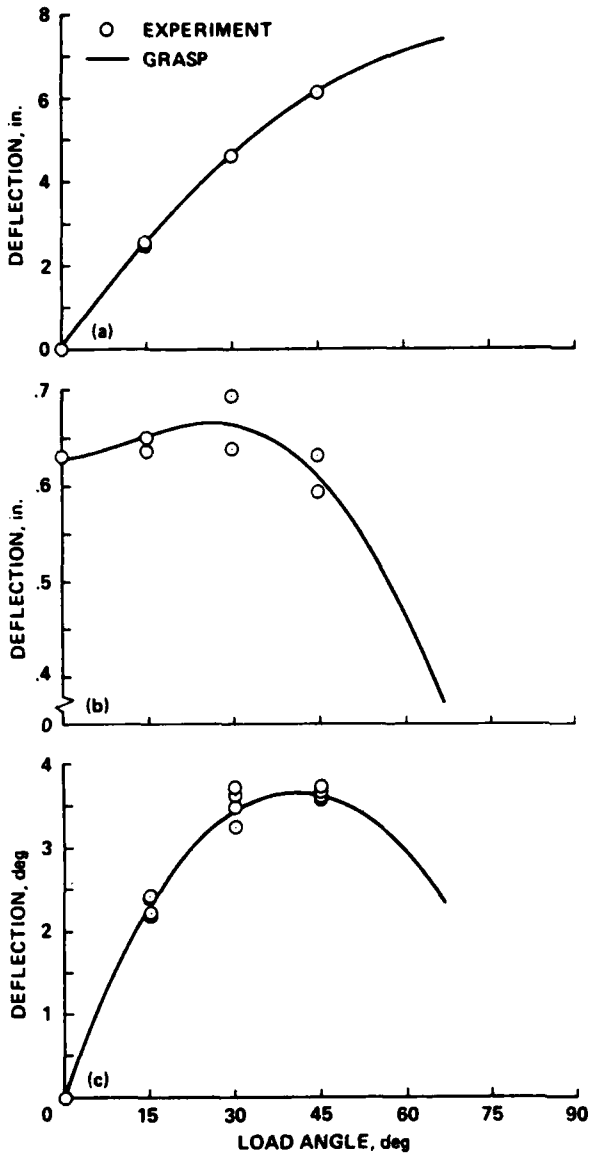


Fig. 12 GRASP correlation with the Princeton experiment (3-lb. tip mass): static deflections, a) flatwise, b) edgewise, c) torsional.

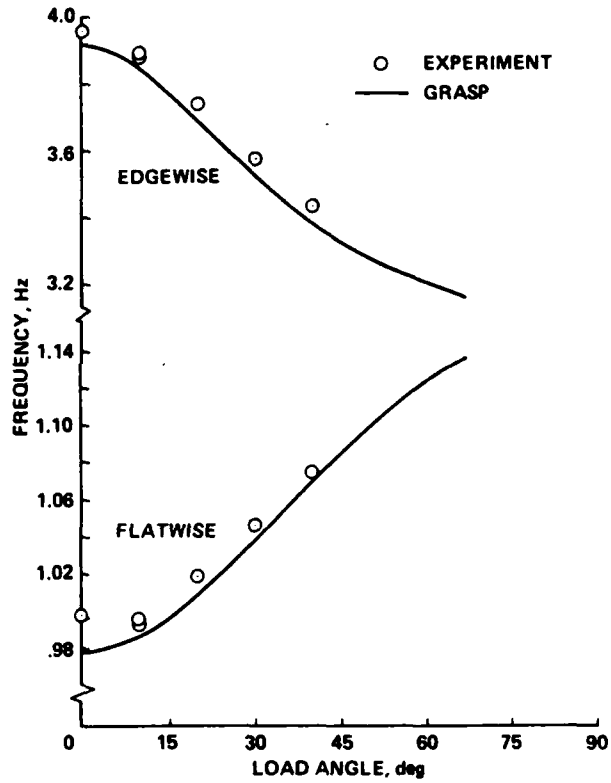


Fig. 13 GRASP correlation with the Princeton experiment (3-lb. tip mass): first flatwise and first edgewise frequencies.

END

12-86

DTIC