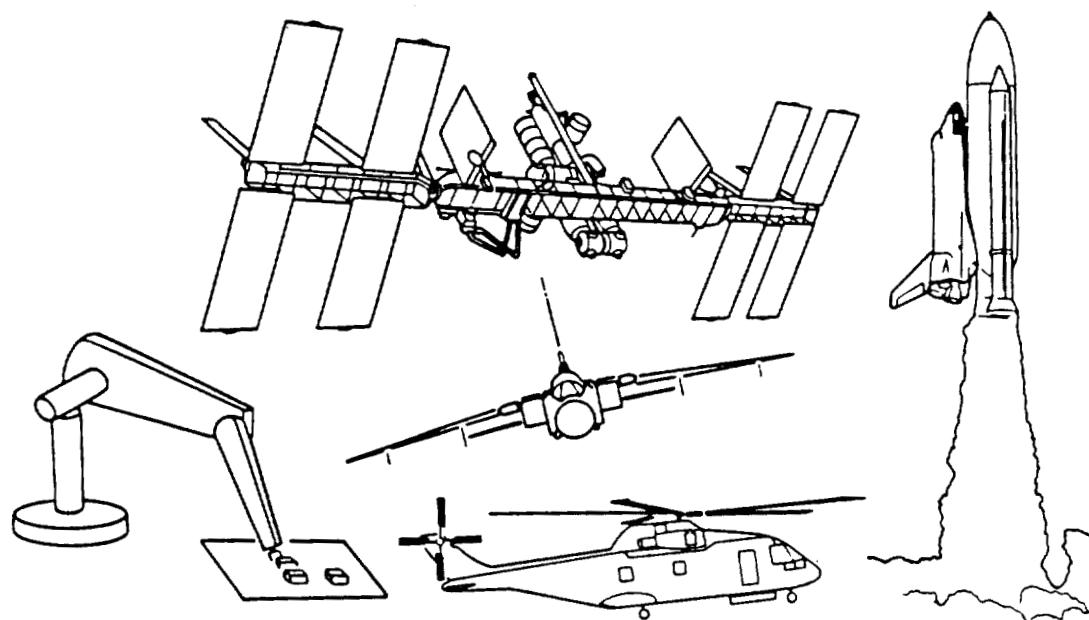# NASA Technical Memorandum 101578, Part One

WORKSHOP on COMPUTATIONAL ASPECTS in the CONTROL of FLEXIBLE SYSTEMS

Held at the Royce Hotel in Williamsburg, Virginia

July 12-14, **1988**

Sponsored by the NASA Langley Research Center

Proceedings Compiled by Larry Taylor
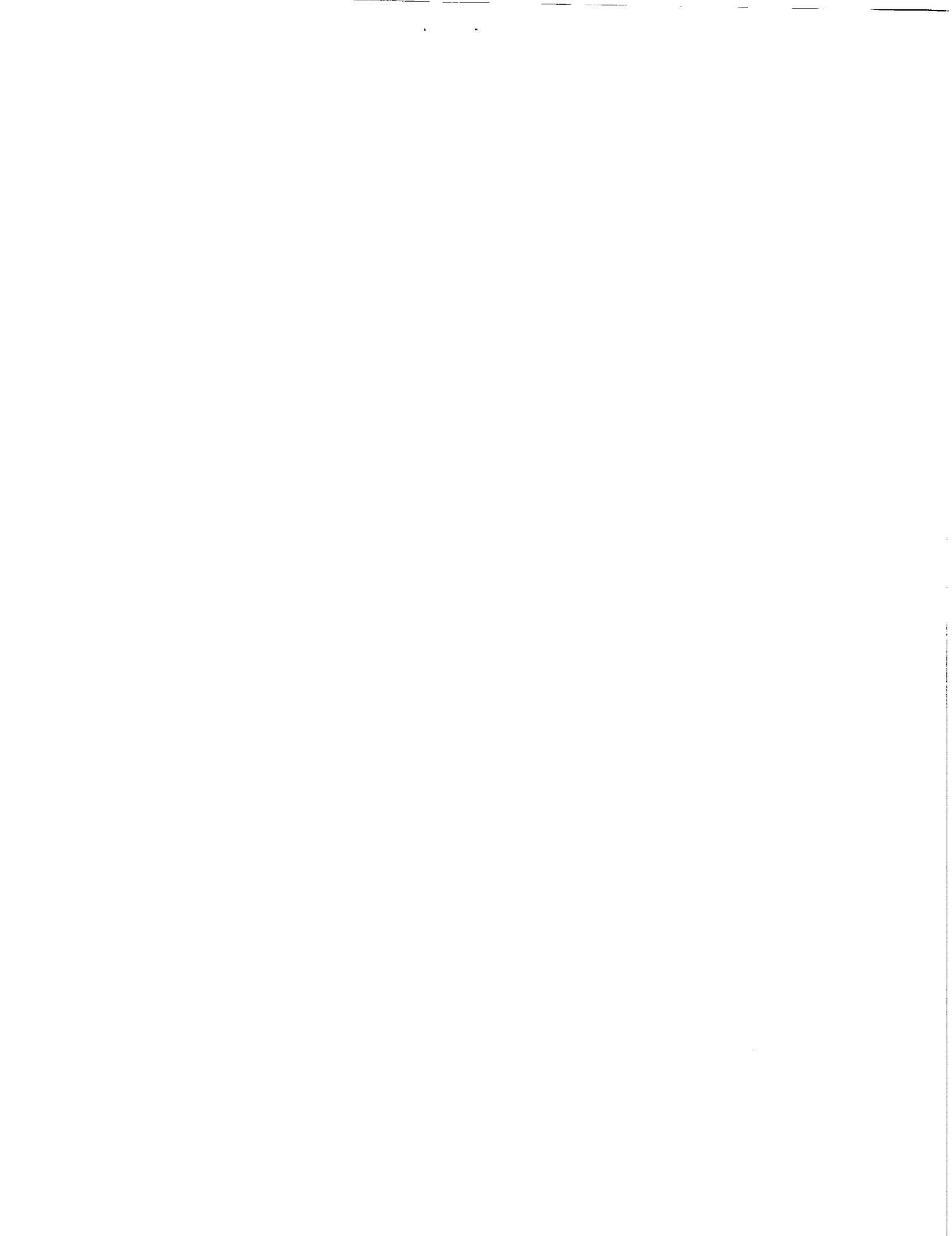
Table of Contents

******* PART ONE *******

Computational Efficiency and Capability

####### PART TWO #######

Modeling and Parameter Estimation

Control Synthesis and Optimization Software

National Aeronautics and
Space Administration

Langley Research Center
Hampton. Virginia
23665-5225

NASA

April 12, 1988

TO: Invited Workshop Participants

FROM: Lawrence W. Taylor, Jr., Chairman

SUBJECT: Workshop on Computational Aspects in the Control of Flexible Systems

BACKGROUND: As aerospace vehicles and robotic systems become larger and more
flexible, the attendant complexity results in an increased demand for high fidelity dynamic
models. This in turn can cause an excessive computational burden for model development,
systems analysis and real-time simulation.

A number of software packages are available for modeling flexible structures and for control
analysis, but there remain unsatisfied needs for more efficient and more comprehensive
software which is easy to use for modeling, analysis, synthesis and simulation. Low cost
parallel processing promises significant increases in computational speed.

GOAL: To assess the state of the technology in software tools for simulation analysis and
synthesis for the control of flexible aerospace systems; establish capabilities and performance
of these tools when applied to specific example problems; and to identify gaps and
shortcomings of software tools.

APPROACH: A workshop will be held at the Royce Hotel in Williamsburg, Virginia,
July 12-14 , 1988. This workshop is being organized under the auspices of the Office of
Aeronautics and Space Technology (OAST) in NASA Headquarters, and is being chaired by
the following people:

Lawrence W. Taylor, Jr.   Virginia B. Marks
NASA Langley Research Center   NASA Langley Research Center
M/S 489   M/S 479
Hampton, VA 23665-5225   Hampton, VA 23665-5225
(804)865-3716   (804)865-2077

Presentations and demonstrations will be given as requested in the Call for Papers (below). In
addition, a panel of experts will be assembled to summarize workshop presentations, lead a
discussion on the state of the technology concerning computational aspects in the control of
flexible structures and recommend future direction of research and areas of concentration. A
proceedings will be published and distributed following the workshop which will include
presentation materials with brief explanations and an address list of workshop attendees.

CALL FOR PAPERS: Presentations will be scheduled for 30 minute time slots including a period for questions and answers. We encourage submissions of one-page abstracts of proposed presentations for the following categories:

MODELING SOFTWARE FOR FLEXIBLE STRUCTURES: Special model formulations, model building, model reduction, modeling articulated, flexible structures.

SYSTEMS ANALYSIS AND SYNTHESIS SOFTWARE: Control synthesis, time-varying system analysis, and nonlinear system analysis

ROBOTIC SYSTEMS APPLICATIONS: Modeling experiences, needed software advances, comparison simulation and actual experience.

SPACECRAFT APPLICATIONS: Comparisons of actual and expected system stability and performance, integrated design techniques.

AIRCRAFT APPLICATIONS: Techniques by which flexibility is treated, capabilities of high fidelity simulation, comparison between actual and expected flight system stability and performance.

SIMULATION COMPUTERS: Workstations for analysis and synthesis, parallel processing for simulation.

SOFTWARE/COMPUTER DEMONSTRATIONS: Software developers and vendors are encouraged to demonstrate the capabilities of their wares. Sun and Microvax workstations will be available and space will be provided for additional equipment and displays. For details call Larry Taylor, (804)865-3716.

PARTICIPATION: Attendance at this conference by nonpresenters is also encouraged to facilitate thorough discussions in all areas. Special arrangements are not required for non-U.S. citizens attending the workshop in Williamsburg. However, if a non-U.S. citizen desires access to the Langley Research Center for any reason, a letter of endorsement from your Embassy in Washington, DC, must be forwarded to NASA Headquarters before plans can be arranged. The address is:

National Aeronautics and Space Administration
International Affairs Division
Code XIC
Washington, DC 10546-0001

RESERVATIONS: A block of rooms has been reserved at the Royce Hotel for attendees of the Workshop on Computational Aspects in the Control of Flexible Systems. The special room rates for this workshop are:

Types of Room	Rate
Single	$60.00 + 6.5% tax
Double	$70.00 + 6.5% tax

The cut-off date for these rooms is June 11, 1988. Reservations requested beyond the cut-off date are subject to availability. Rooms may be available after this date, but not necessarily at the same rate. To make reservations, call (804)229-4020 and request a room reserved for the Workshop on Computational Aspects in the Control of Flexible Systems or return the
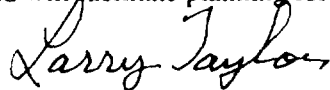
enclosed reservation form before the cut-off date. Reservations must be arranged by the individual attendees, keeping in mind that during the summer Colonial Williamsburg is a popular vacation site.

REGISTRATION: A registration form for the workshop is enclosed. Please complete this form and return it by June 11, 1988.

DEADLINES: All submissions must be accompanied by the title, full name, affiliation, complete address and telephone number of each co-author in regular sessions and each participant in panel sessions.

MAY 1, 1988: Indicate interest in demonstrating software packages or computers by calling Larry Taylor, Chairman.

JUNE 1, 1988: Submit one-page abstract for proposed technical presentations to Gin Marks, Vice Chairman.

JUNE 11, 1988: Registration forms due to Gin Marks.
Cut-off date for reservations at the Royce Hotel at the special group rate.

JUNE 22, 1988: Authors are notified of presentation acceptance. Demonstrators are notified on demonstration acceptance.

JULY 1, 1988: Finalized agenda is sent to all registered atttendees.

JULY 12, 1988: First day of workshop. Authors are asked to provide camera ready copies of presentation material with brief explanations included for workshop proceedings.

Early response to the Call for Papers and submission of your registration forms is appreciated and will facilitate planning for a successful workshop.

Lawrence W. Taylor, Jr.
Chairman

Enclosures:
Preliminary Agenda
Registration Form
Reservation Request
Local Attractions
Golf Facilities
"Williamsburg Great Entertainer"

3

COMPUTATIONAL ASPECTS WORKSHOP ORGANIZATION

Chairman	Larry Taylor
Administrator	Trish Johnson
Aero Program Chairman	Jerry Elliott
Robotics Program Chairman	Jack Pennington
Space Program Chairman	Jerry Newsom
Computational Facilities Coord.	George Tan
Meeting Site Contract Coord.	Pat Gates
Mail List Secretary	Trish Johnson
Proceedings Compiler	Larry Taylor
Consultant	Gin Marks

5

Attendance List for the

WORKSHOP ON

COMPUTATIONAL ASPECTS

IN THE CONTROL OF FLEXIBLE SYSTEMS

JULY 12-14, 1988

ROYCE HOTEL, EMPIRE BALLROOM

WILLIAMSBURG, VIRGINIA

7

Dr. Willard W. Anderson
NASA Langley Research Center
Mail Stop 479
Hampton, VA 23665-5225
804-865-3049

Dr. Ernest S. Armstrong
NASA Langley Research Center
Mail Stop 499
Hampton, VA 23665-5225
804-865-4848

Prof. A. V. Balakrishnan
University of California at LA
School of Eng. and Applied Science
6731 Boelter Hall
Los Angeles, CA 90024-1600
213-825-2180

W. Keith Belvin
NASA Langley Research Center
Mail Stop 499
Hampton, VA 23665-5225
804-865-3801

Dr. Douglas Bernard
Jet Propulsion Laboratory
Mail Code 198-326
4800 Oak Grove Drive
Pasadena, CA 91109
818-354-2597

Dr. Paul Blelloch
Structural Dynamics Research Corp.
11055 Roselle Street
San Diego, CA 92121-1276
619-450-1553

John Butwin
Hughes Aircraft - Space & Comm Grp
P.O. Box 92919
Los Angeles, CA 90009
213-606-1787

C. W. Chang
COMTEK
305 Susan Newton Lane
Tabb, VA 23602

Amir A. Anissipour
The Boeing Company
P.O. Box 3707, M/S 9W-38
Seattle, WA 98124-2207
206-277-4390

Peter M. Bainum
Howard University
Dept. of Mechanical Engineering
Washington, DC 20059
202-636-6612

Michael Barsky
NASA Langley Research Center
Mail Stop 161
Hampton, VA 23665-5225
804-865-4591

Alex Berman
Kaman Aerospace Corporation
D-435, Building 2
P.O. Box 2
Bloomfield, CT 06002-0002
203-243-7215

Dr. Saroj K. Biswas
Temple University
Dept of Electrical Engineering
Philadelphia, PA 19122
215-787-8403

Kerman Buhariwala
Spar Aerospace Limited
1700 Ormont Drive
Weston, Ontario M9L 2W7
CANADA
416-745-9680 X4

John E. Byrne
Aastra Aerospace
1685 Flint Road
Downsview, Ontario M3J-2W8
CANADA
416-736-7070

Chaur-Ming Chou
University of Lowell
Dept of Mechanical Engineering
Lowell, MA 01854

Prof. Ajit K. Choudhury
Howard University
Dept of Electrical Engineering
Washington, DC 20059
202-636-7124

William W. Cimino
Boeing Aerospace
P.O. Box 3999
Mail Stop 82-24
Seattle, WA 98124
206-773-5191

William Clark
VPI&SU
Mechanical Engineering Department
Randolph Hall
Blacksburg, VA 24061
703-961-7038

Edward E. Coleman
The Boeing Company
P.O. Box 3707, M/S 9W-38
Seattle, WA 98124-2207
206-277-3189

John B. Dahlgren
Jet Propulsion Laboratory
Mail Stop 198-330
4800 Oak Grove Drive
Pasadena, CA 91109
818-842-2039

Dr. Arthur R. Dusto
Boeing Advanced Systems
P.O. Box 3707
M/S 33-12
Seattle, WA 98124-2207
206-241-4393

Dr. Yaghoob S. Ebrahimi
The Boeing Company
P.O. Box 3707, M/S 9W-38
Seattle, WA 98124-2207
206-277-2261

Dr. John W. Edwards
NASA Langley Research Center
Mail Stop 173
Hampton, VA 23665-5225
804-865-4236

Ko-Hui M. Fan
University of Maryland
Systems Research Center
College Park, MD 20742
301-454-8832

Shalom Fisher
Naval Research Laboratory
Code 8241
4555 Overlook Avenue, S.W.
Washington, DC 20375-5000

Ian T. Gallimore
NASA Langley Research Center
Mail Stop 161
Hampton, VA 23665-5225
804-865-4591

Dave Ghosh
NASA Langley Research Center (PRC)
Mail Stop 161
Hampton, VA 23665-5225
804-865-4591

Michael G. Gilbert
NASA Langley Research Center
Mail Stop 243
Hampton, VA 23665-5225
804-865-2388

Rafael Gluck
TRW Inc. - Space Technology Group
R4/1408
One Space Park
Redondo Beach, CA 90278
213-297-3655

Dr. Nesim Halyo
Information & Control Systems, Inc.
28 Research Drive
Hampton, VA 23666
804-865-0371

Prof. S. Hanagud
George Institute of Technology
School of Aerospace Engineering
Atlanta, GA 30332
404-894-3000

Brantley R. Hanks
NASA Langley Research Center
Mail Stop 161
Hampton, VA 23665-5225
804-868-3058

Dr. George R. Hennig
The Boeing Company
P.O. Box 3707, M/S 9W-38
Seattle, WA 98124-2207
206-277-4273

Garnett C. Horner
NASA Langley Research Center
Mail Stop 230
Hampton, VA 23665-5225
804-865-3699

Dr. T. Houlihan
The Jonathan Corporation
150 Boush Street
Norfolk, VA 23510
804-640-7140

Dr. Jerrold M. Housner
NASA Langley Research Center
Mail Stop 230
Hampton, VA 23665-5225
804-865-4423

Robert Howe
Applied Dynamics International
3800 Stone School Road
Ann Arbor, MI 48108
313-973-1300

Glenda L. Jeffrey
NASA Langley Research Center
Mail Stop 230
Hampton, VA 23665-5225
804-865-4513

Dexter Johnson
NASA Langley Research Center
Mail Stop 230
Hampton, VA 23665-5225
804-865-2738

Dr. Suresh M. Joshi
NASA Langley Research Center
MS 161
Hampton, VA 23665-5225
804-865-4591

Dr. Dinesh S. Joshi
Northrop Aircraft Division
Dept. 3836/82
One Northrop Avenue
Hawthorne, CA 90250-3277
213-332-7808

Prof. Walter Karplus
University of California at LA
School of Eng. and Applied Sciences
3732B Boelter Hall
Los Angeles, CA 90024
213-825-2929

David S. Kawg
Charles Stark Draper Laboratory
Mail Stop 4C
555 Technology Square
Cambridge, MA 02139

Ian W. Kaynes
Royal Aircraft Establishment
M&S Department, X33
Farnborough
Hampshire GU146TD
ENGLAND
0252-24461x5591

Ted F. Knaak
General Electric
Valley Forge Space Center
Box 8555, Bldg 100, Room U4248
Philadelphia, PA 19101
215-354-3672

Robert A. Laskin
Jet Propulsion Laboratory
Attn: 198-326/Robert A. Laskin
Mail Stop 198-326
4800 Oak Grove Drive
Pasadena, CA 91109
818-354-5086

Danette Lenox
NASA Langley Research Center (PRC)
Mail Stop 161
Hampton, VA 23665-5225
804-865-4591

Feiyue Li
Howard University
Dept of Mechanical Engineering
Washington, DC 20059
202-636-7124

Kyong Lim
NASA Langley Research Center (PRC)
Mail Stop 499
Hampton, VA 23665-5225
804-865-2026

Sang Seok Lim
#1-104 Henderson Street
Ottawa Ontario KIN7P4
CANADA

Dr. Jiguan G. Lin
Control Research Corporation
6 Churchill Lane
Lexington, MA 02173
617-863-0889

Michael Lou
Jet Propulsion Laboratory
Mail Stop 157-410
4800 Oak Grove Drive
Pasadena, CA 91109
818-354-3034

Leo H. McWilliams
Allied Signal Inc.
Engine Controls Division
717 North Bendix Drive
South Bend, IN 46620
219-231-3749

Larry Michaels
Applied Dynamics International
3800 Stone School Road
Ann Arbor, MI 48108
313-973-1300

K. Dean Minto
GE - Corporate R&D Center
KWD-207
Schenectady, NY 12301
518-387-6760

Dr. Raymond C. Montgomery
NASA Langley Research Center
Mail Stop 161
Hampton, VA 23665-5225
804-865-4591

Wendy Moore
NASA Langley Research Center
Mail Stop 161
Hampton, VA 23665-5225
804-865-4591

Ronald Moquin
Applied Dynamics International
3800 Stone School Road
Ann Arbor, MI 48108
313-973-1300

Nancy Nimmo
NASA Langley Research Center
Mail Stop 499
Hampton, VA 23665-5225
804-865-2371

Dr. Thomas E. Noll
NASA Langley Research Center
Mail Stop 243
Hampton, VA 23665-5225
804-865-3451

Prof. Irving J. Oppenheim
Carnegie-Mellon University
Department of Civil Engineering
Pittsburgh, PA 15213
412-268-2950

Linda Parada
Calspan Advanced Technical Center
P.O. Box 400
Buffalo, NY 14225

William M. Patten
VPI&SU
Mechanical Engineering Department
Randolph Hall
Blacksburg, VA 24061
703-961-7038

Sivakumar Tadikonda
Rutgers University
Dept. of Mechanical Engineering
P.O. Box 909
Piscataway, NJ 08855

George Tan
NASA Langley Research Center (PRC)
Mail Stop 161
Hampton, VA 23665-5225
804-865-4591

Lawrence W. Taylor, Jr.
NASA Langley Research Center
Mail Stop 489
Hampton, VA 23665-5225
804-865-3716

Joseph Turnbull
Charles Stark Draper Laboratory
Mail Stop 4C
555 Technology Square
Cambridge, MA 02139
617-258-2292

A. Galip Ulsoy
University of Michigan
Dept. of Mechanical Engineering
Ann Arbor, MI 48109-2125
313-936-0407

Robert H. Van Vooren
TRW Inc. - Space & Technology Group
R4/1098
One Space Park
Redondo Beach, CA 90278
213-535-8764

Richard Vandervoork
Dynacs Engineering
2280 N. U.S. 19
Suite 111
Clearwater, FL 34623
813-799-4124

Joseph E. Walz
NASA Langley Research Center
Mail Stop 499
Hampton, VA 23665-5225
804-865-2367

Sharon L. Welch
NASA Langley Research Center
Mail Stop 161
Hampton, VA 23665-5225
804-865-4591

James L. Williams
NASA Langley Research Center
Mail Stop 499
Hampton, VA 23665
804-865-3801

Jeffrey P. Williams
NASA Langley Research Center
Mail Stop 230
Hampton, VA 23665
804-865-4423

Stanley E. Woodard
NASA Langley Research Center
Mail Stop 230
Hampton, VA 23665-5225
804-865-4513

Shih-Chin Wu
COMTEK
305 Susan Newton Lane
Tabb, VA 23602

John W. Young
NASA Langley Research Center
Mail Stop 499
Hampton, VA 23665-5225
804-865-2367

Dr. Larry Zavodney
Ohio State University
Dept. of Engineering Mechanics
209 Boyd Lab
155 W. Woodruff Avenue
Columbus, OH 43210

PRECEDING PAGE BLANK NOT FILMED

SESSION I - NEEDS FOR ADVANCED CSI SOFTWARE

PRECEDING PAGE BLANK NOT FILMED

NASA'S CONTROLS-STRUCTURES INTERACTION PROGRAM

By

Brantley R. Hanks
NASA Langley Research Center
Hampton, Virginia

ABSTRACT

Spacecraft design is conducted conventionally by estimating sizes and masses of mission-related components, designing a structure to maintain desired component relationships during operations, and then designing a control system to orient, guide and/or move the spacecraft to obtain required performance. This approach works well in cases where a relatively high stiffness structural bus is attainable and where nonstructural components are massive relative to the structure.

Occasionally, very flexible, distributed-mass, structural components, such as solar arrays and antennas are attached to the structural bus. In these, the primary purpose is to maintain geometric relationships rather than support masses which are large relative to the structural mass. Because of their flexibility, potential interactions of such components with the spacecraft control system can reduce performance or restrict operations. This interaction, referred in this document as controls-structures interaction (CSI), also occurs in small components if precision pointing and/or surface shapes/orientations are critical performance factors and in very large systems where attaining a high structural stiffness is detrimental to launch and operations requirements. The degree of success in handling these situations in past designs is uncertain. Reduced performance and unexpected dynamic motions have been observed in operational spacecraft; but, in most cases, the spacecraft were not sufficiently instrumented to determine the cause.

Designing to avoid CSI generally requires either stiffening the structure (costly in mass, inertia and fuel consumption) or slowing down the control system response (costly in performance capability). Using the power available in the control system to reduce the interactive motions is theoretically possible; a great number of approaches to do so have been advanced in the literature. However, reduction of these approaches to practice on hardware has not been accomplished on any meaningful scale. The techniques generally require analytical representations of the system within the control loop. The fidelity, size, accuracy and computational speed of these analyses are integrally related to, and affect the performance of, the combined structure-control system. The structural hardware, the control hardware, and the analytical models cannot be separated in the process of verifying that the system performs as required. Furthermore, if improperly designed, the closed-loop system is subject not only to inadequate performance, but also to destructive dynamic instability.

Future NASA missions are likely to increase the likelihood of CSI because of increased size of distributed-mass components, greater requirements for surface and pointing precision, increased use of articulated moving components, and increased use of multi-mission science platforms (with multiple control systems on board). An SSTAC

develop the technology to solve the CSI problem. More recently, a NASA CSI Requirements Committee reviewed potential future NASA missions and found the need for CSI technology to be widespread.

A NASA program is about to start which has the objective to advance CSI technology to a point where it can be used in spacecraft design for future missions. Because of the close interrelationships between the structure, the control hardware, and the analysis/design, a highly interdisciplinary activity is defined in which structures, dynamics, controls, computer and electronics engineers work together on a daily basis and are co-located to a large extent. Methods will be developed which allow the controls and structures analysis and design functions to use the same mathematical models. Hardware tests and applications are emphasized and will require development of concepts and test methods to carry out.

Because of a variety of mission application problem classes, several time-phased, focus ground test articles are planned. They will be located at the Langley Research Center (LaRC), the Marshall Space Flight Center (MSFC) and at the Jet Propulsion Laboratory (JPL). It is anticipated that the ground tests will be subject to gravity and other environmental effects to the extent that orbital flights tests will be needed for verification of some technology items. The need for orbital flight experiments will be quantified based on ground test results and mission needs. Candidate on-orbit experiments will be defined and preliminary design/definition and cost studies will be carried out for one or more high-priority experiments.

22

A BRIEF OVERVIEW OF
THE CONTROLS-STRUCTURES
INTERACTION (CSI) PROGRAM

PRESENTED BY
BRANTLEY R. HANKS

23

THE NASA LANGLEY RESEARCH CENTER

THE NASA CONTROLS-STRUCTURES INTERACTION (CSI) PROGRAM

o A RESTRUCTURING OF THE COFS PROGRAM

o EMPHASIZES INCREASED GROUND TESTING AND ANALYSIS
  WITH A CONSERVATIVE FLIGHT EXPERIMENT SCHEDULE

o MISSION APPLICATIONS WEIGHTED TOWARD EARTH
  OBSERVATION SPACECRAFT FOR 2000+

o JOINT EFFORT OF NASA HEADQUARTERS AND THREE FIELD
  ORGANIZATIONS, LANGLEY, MARSHALL AND JPL

o MANAGED BY HEADQUARTERS CODE RM, SPECIFIC ROLES FOR
  EACH FIELD ORGANIZATION, OVERALL TECHNICAL
  COORDINATION BY LANGLEY

NASA CSI PROGRAM ORGANIZATION

25

```
┌─────────────────────┐
│      MISSION        │          ┌──────────────────────┐          ┌──────────────────────┐
│   APPLICATIONS      │──────────│  CSI PROGRAM MGR      │──────────│    INTERCENTER       │
│   ADV. COMMITTEE    │          │     CODE RM           │          │ TECH WORKING GROUP   │
└─────────────────────┘          └──────────────────────┘          │    LaRC, LEAD        │
                        ·                                           └──────────────────────┘
┌─────────────────────┐
│ UNIVERSITY/INDUSTRY │
│   ADV. COMMITTEE    │
└─────────────────────┘
```

| JPL
CSI OFFICE | LaRC
CSI OFFICE | MSFC |

o OPTICS-CLASS
  APPLICATIONS

o MICRO-PRECISION
  CSI DEVELOPMENT

o CSI TECH PROG
  COORDINATION

o ANALYSIS/DESIGN
  METHODS

o TEST METHODS

o GI PROGRAM

o FLIGHT
  QUALIFICATION
  METHODS/TESTS

o CASES FLIGHT
  EXPERIMENT
  (X-RAY PINHOLE
  OCCULTER)

LaRC CSI ORGANIZATION

CSI PROGRAM GENERAL OBJECTIVES

o REDUCE DYNAMIC RESPONSE FOR GIVEN MANEUVERS/LOADS
   WITHOUT INCREASING MASS OR CONTROL ENERGY

o DEVELOP ACCURATE METHODS FOR PREDICTION OF ON-ORBIT
   RESPONSE BASED ON ANALYSIS TUNED BY GROUND TESTS

o DEVELOP UNIFIED MODELING, ANALYSIS AND DESIGN METHODS
   WHICH PROVIDE BETTER AND FASTER RESULTS THAN CURRENT
   METHODS

o VERIFY THE CAPABILITY TO VALIDATE ON-ORBIT CSI
   PERFORMANCE BY GROUND-BASED METHODS

27

CSI PROGRAM ELEMENTS

CONFIGURATIONS & CONCEPTS
- QUANTIFY MISSION REQUIREMENTS & BENEFIT TRADE-OFFS
- EXPAND CONFIGURATION AND TECHNOLOGY OPTIONS

INTEGRATED ANALYSIS & DESIGN
- DEVELOP UNIFIED MODELING & ANALYSIS TECHNIQUES
- DEVELOP IMPROVED CSI SYSTEM DESIGN APPROACHES

GROUND TEST METHODOLOGY
- DEVELOP TEST METHODS FOR VERIFYING CSI DESIGNS
- VALIDATE THEORETICAL CSI TECHNICAL APPROACHES

IN-SPACE FLIGHT EXPERIMENTS
- INVESTIGATE PHENOMENA MASKED IN GROUND TESTS
- CALIBRATE PROPOSED VERIFICATION TEST & ANALYSIS
  METHODS

GUEST INVESTIGATOR PROGRAM
- PROVIDE MECHANISM/FUNDS FOR INCORPORATING IDEAS
  & CAPABILITIES OF NON-NASA RESEARCHERS

EVOLUTIONARY GROUND TEST SYSTEM

REACTION WHEELS

THROTTLABLE JETS

SEGMENTED REFLECTOR

MESH ANTENNA

BEAM SEGMENTS

POINTING MOUNT

SOLAR ARRAY

29
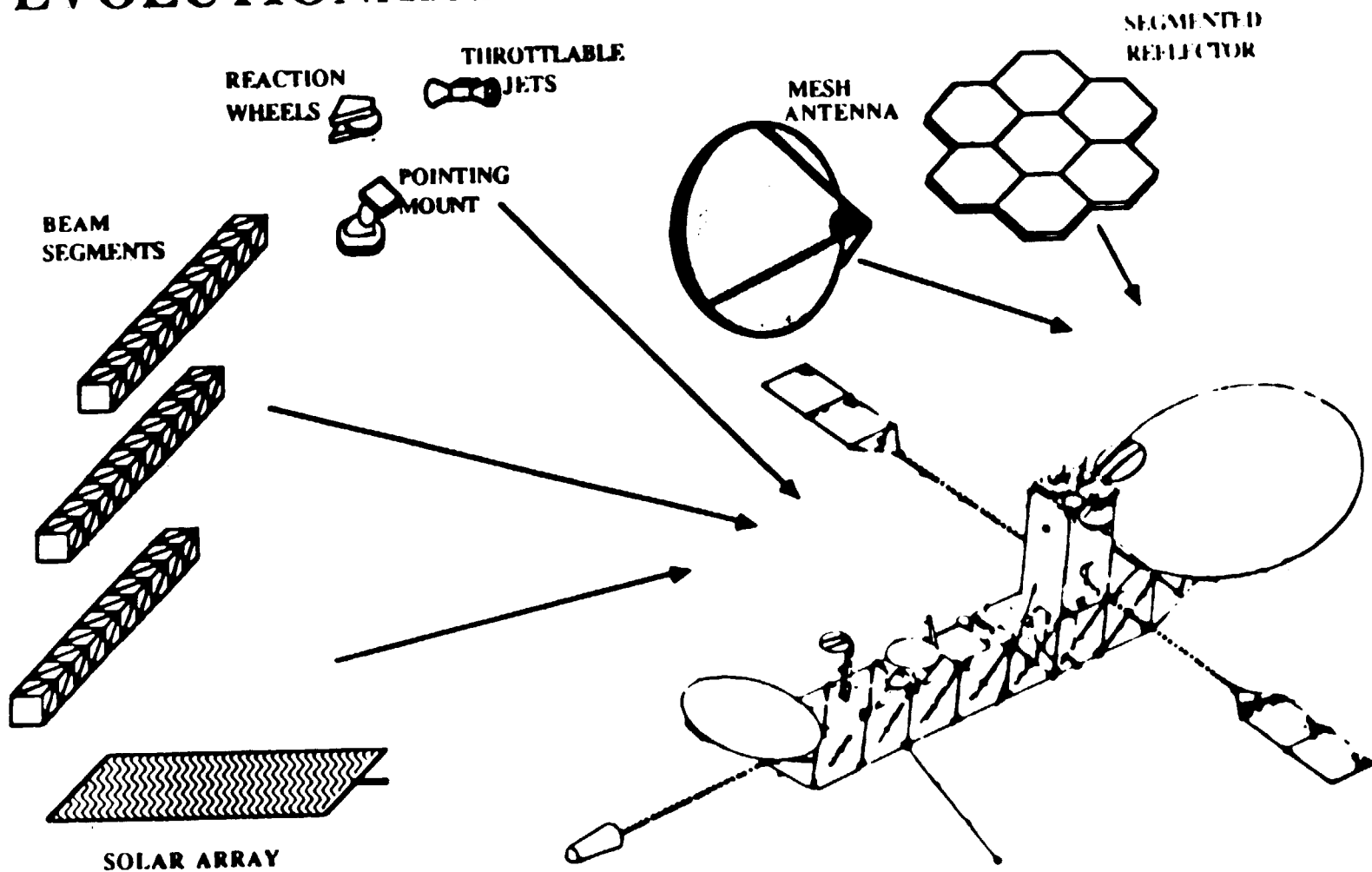
FLIGHT STRUCTURES CONTROL EXPERIMENT

BOOM TIP ASSEMBLY

LOWER BOOM
AMED ASSEMBLY

COALIGNED BOOM
BASE

MISSION PECULIAR
EQUIPMENT

PAYLOAD CARRIER
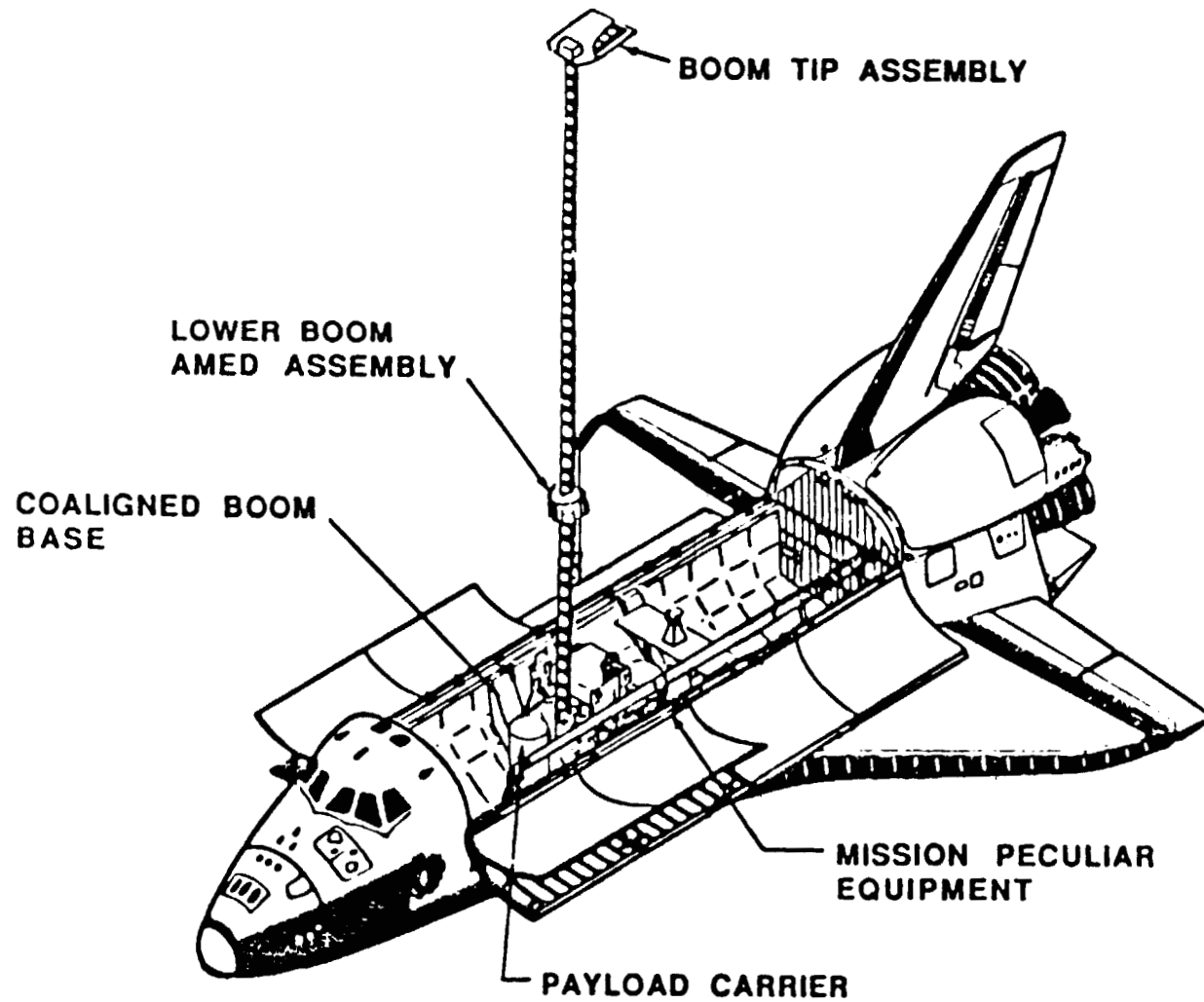
USEFUL WORKSHOP OUTPUT

CASES WHERE PROBLEMS WERE CAUSED BY THE FOLLOWING:
- INACCURATE MATH MODELS
- INACCURATE COMPUTATIONAL ALGORITHMS
- INABILITY TO TEST SYSTEM
- SLOW DESIGN ITERATION TURNAROUND
- FLEXIBLE STRUCTURE INTERACTION WITH CONTROLS

EXAMPLES OF SIGNIFICANT DESIGN IMPACT TO AVOID CSI PROBLEMS:
- BY LIMITING CAPABILITY
- BY REDUCING REQUIREMENTS
- BY "BEEFING-UP" DESIGN

QUANTIFIED EXAMPLES OF THE COMPUTATIONAL BURDEN
- ITERATION TIMES
- COMPUTER "HORSEPOWER" REQUIREMENTS

PRIORITIZED AREAS OF EXPECTED BENEFIT FROM RESEARCH

UPCOMING CSI PROGRAM EVENTS

o FIRST GI CONTRACTS TO BE ANNOUNCED - AUGUST

o GI/UNIVERSITY ENGR RESEARCH CENTERS/OUTREACH COORD
  MEETING - OCTOBER

o THIRD NASA/DOD CSI CONFERENCE, JANUARY 89

o NEXT GI PROPOSAL SOLICITATION - 1st QUARTER 89

32

COMPUTATIONAL CONTROLS FOR AEROSPACE SYSTEMS

JPL

GUY K. MAN

ROBERT A. LASKIN

A. FERNANDO TOLIVAR

12 JULY 1988

COMPUTATIONAL CONTROLS
OBJECTIVE

DEVELOP THE NEXT GENERATION GUIDANCE AND CONTROL
ANALYSIS AND DESIGN TOOLS TO ENABLE FUTURE MISSIONS
AND TO IMPROVE PRODUCTIVITY AND RELIABILITY.

34

TOOLS FOR CONTROL SYSTEM DEVELOPMENT

35

```
┌─────────────────┐
│                 │      ● SPACECRAFT CONFIGURATION
│   TRADE OFF     │        TRADE STUDIES
│                 │
│     STUDY       │
│                 │      ● CONTROL HARDWARE SELECTION
└─────────────────┘        AND PLACEMENT
         │
         ▼
┌─────────────────┐      ● PLANT MODELING
│                 │
│     DESIGN      │      ● DETAILED DESIGN
│                 │
│                 │      ● DESIGN ASSESSMENT
└─────────────────┘
         │
         ▼
┌─────────────────┐      ● FLIGHT AND GROUND SOFTWARE INTEGRATION
│                 │
│  VERIFICATION   │      ● PERFORMANCE DEMONSTRATION
│                 │
│                 │      ● HARDWARE IN THE LOOP
└─────────────────┘        REAL TIME VERIFICATION
         │
         ▼
┌─────────────────┐      ● REAL TIME OPERATIONS
│                 │
│   OPERATIONS    │      ● SYSTEM IDENTIFICATION
│                 │
│                 │      ● ANOMALY INVESTIGATION
└─────────────────┘
```
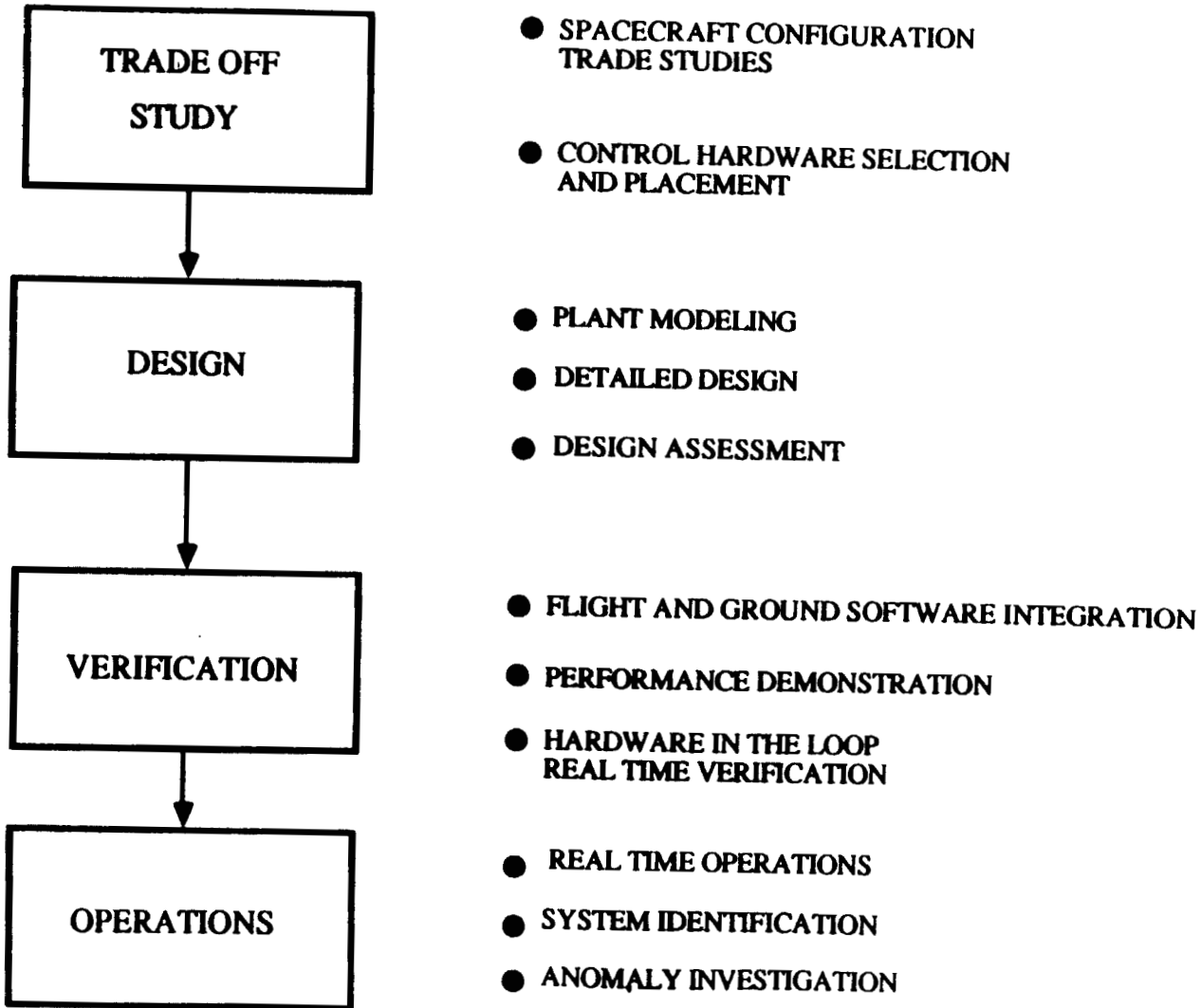
TOOLS ARE INDISPENSIBLE FOR CONTROL SYSTEM DEVELOPMENT

GOALS FOR NASA COMPUTATIONAL CONTROL

- IMPROVE QUICK-DESIGN TURN AROUND TIME BY A FACTOR OF 16
  (4 MONTHS → 1 WEEK)

- IMPROVE EVALUATION TURN AROUND TIME BY A FACTOR OF 40
  (10 MONTHS → 1 WEEK)

- ENABLE REAL TIME HARDWARE-IN-THE-LOOP SIMULATION OF
  COMPLEX SPACECRAFT

- ENABLE REAL TIME ANOMALY INVESTIGATION FOR OPERATIONS

- ENABLE TOOLS TO HANDLE 300 STATES BY 1992 AND 1000 STATES BY 1996

RATIONALE

- LACK OF QUICK-DESIGN TOOLS TO IMPACT SPACECRAFT DESIGN

- LACK OF EFFECTIVE EVALUATION TOOLS TO CHECK DESIGN MARGIN & PERFORMANCE

- LACK OF REAL TIME SIMULATION TOOL OF REALISTIC SPACECRAFT TO CERTIFY DESIGN

- LACK OF QUICK DIAGNOSTIC TOOLS FOR MISSION OPERATIONS

LACK OF PROPER TOOL CREATES INTOLERABLE RISK FOR FUTURE SPACECRAFT SYSTEMS

THE GALILEO CONTROL DESIGN PROBLEM

PROBLEM:

LACK OF QUICK-LOOK TOOL
LEADS TO FAILURE TO MEET
MISSION REQUIREMENTS

SPIN BEARING ASSEMBLY
(CLOCK CONTROL LOOP)

A NONCOLOCATED SENSOR AND
ACTUATOR POINTING SYSTEM

SCAN
PLATFORM

SCAN ACTUATOR
(CONE CONTROL LOOP)

ATTITUDE CONTROL
THRUSTERS

STAR 48 SRM
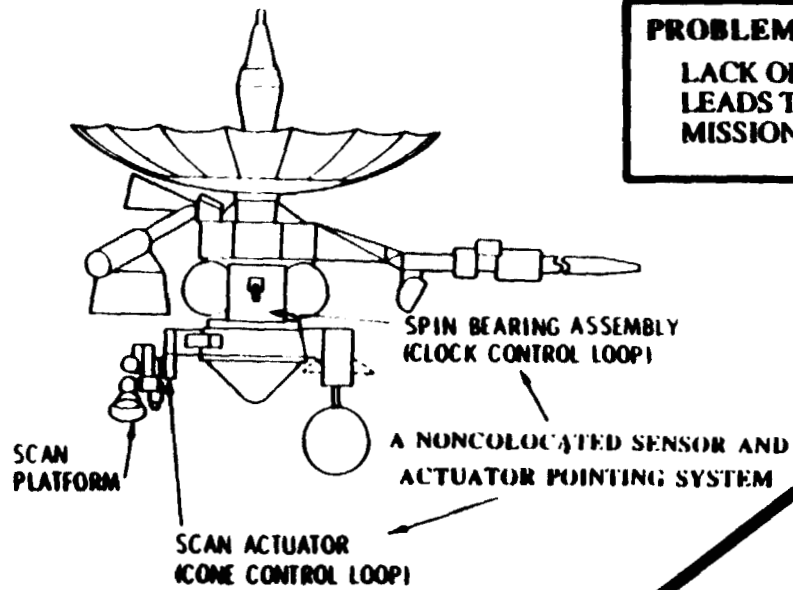
PROBLEM:

LACK OF EFFECTIVE EVALUATION
TOOL PROHIBITS US FROM IDENTIFYING
A MISSION CATASTROPHIC FAILURE
DURING VENUS ORBIT INSERTION
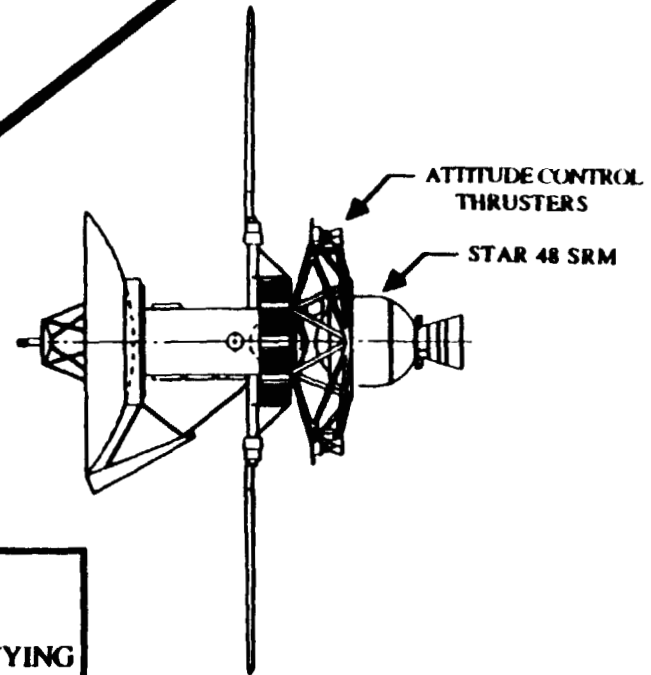
MAGELLAN SPACECRAFT VENUS ORBIT INSERTION PROBLEM

GALILEO CONTROL SYSTEM REAL TIME TESTING

**Galileo AACS
Test Area**

SECTION 343
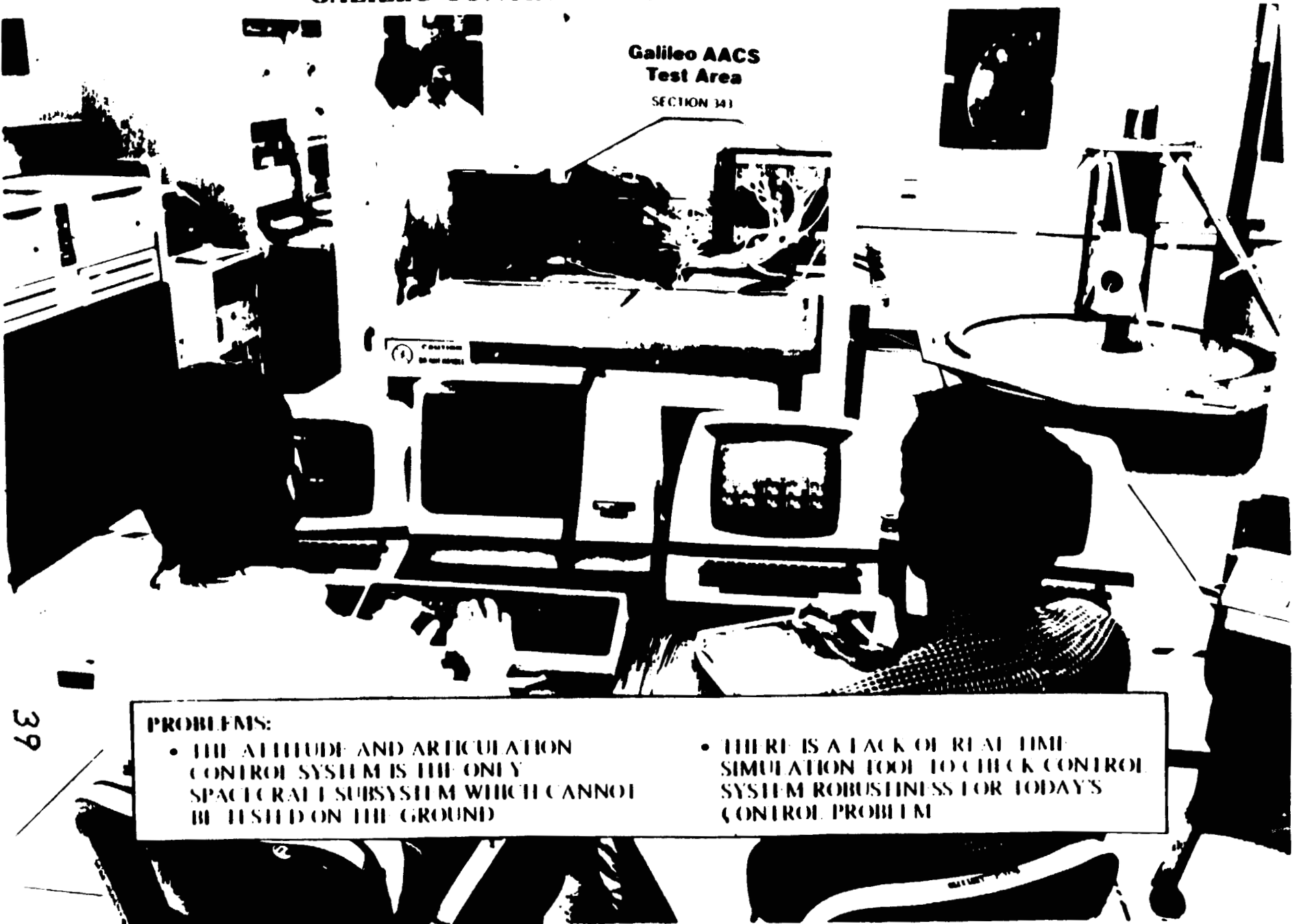
PROBLEMS:

- THE ATTITUDE AND ARTICULATION CONTROL SYSTEM IS THE ONLY SPACECRAFT SUBSYSTEM WHICH CANNOT BE TESTED ON THE GROUND

- THERE IS A LACK OF REAL TIME SIMULATION TOOL TO CHECK CONTROL SYSTEM ROBUSTNESS FOR TODAYS CONTROL PROBLEM

MISSION OPERATIONS SUPPORT IS INADEQUATE

40
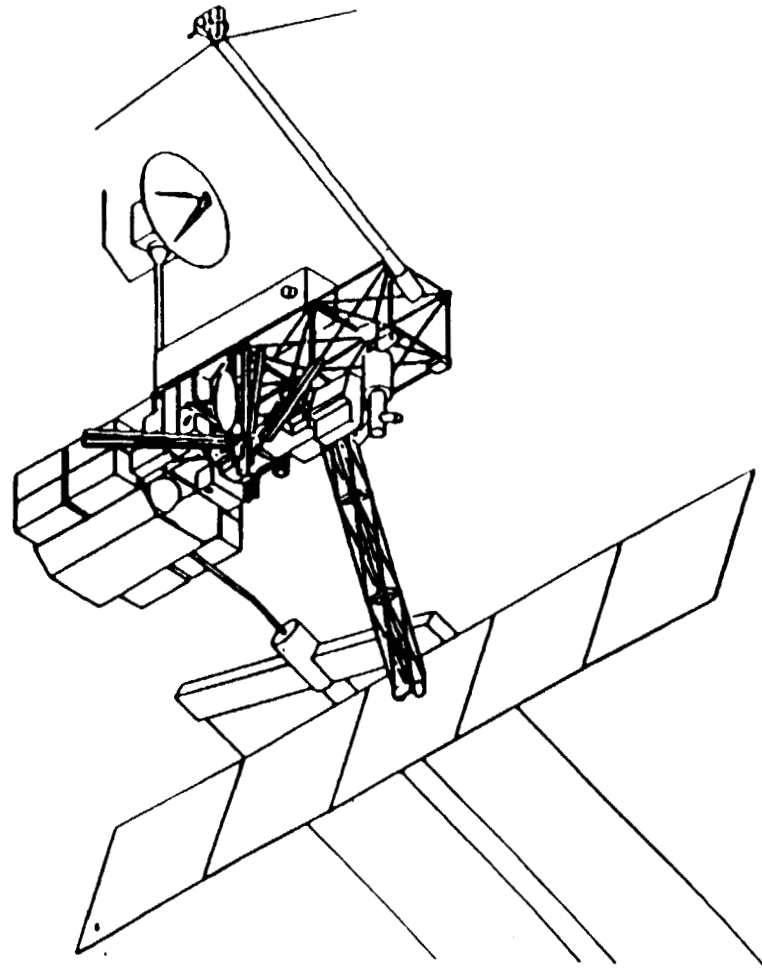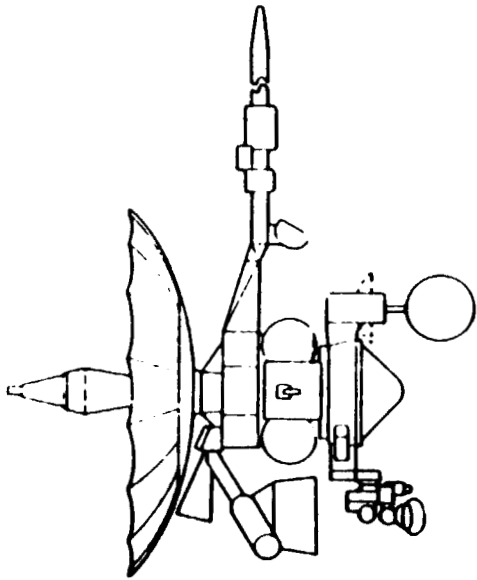
PROBLEM:
LACK OF QUICK DIAGNOSTIC TOOL
FOR ANOMALY INVESTIGATION
LEAD TO CONCERNS IN TURN
AROUND TIME FOR OPERATIONS

GROWTH IN SPACECRAFT MODELING COMPLEXITY
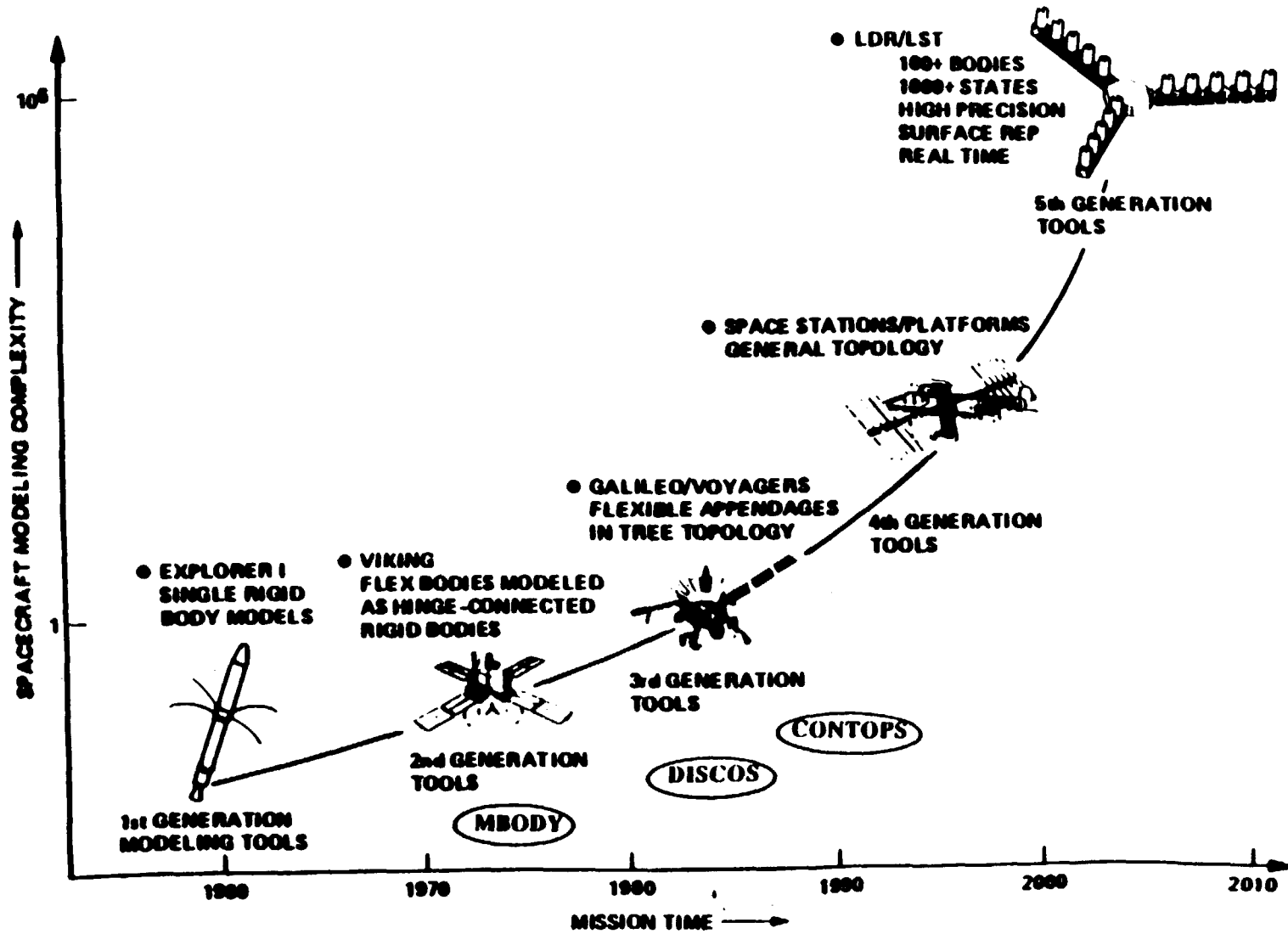
EVOLUTION OF EARTH OBSERVING PLATFORMS

LANDSAT
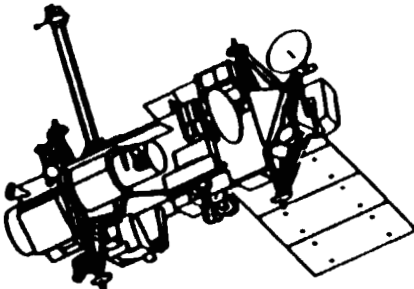
UARS
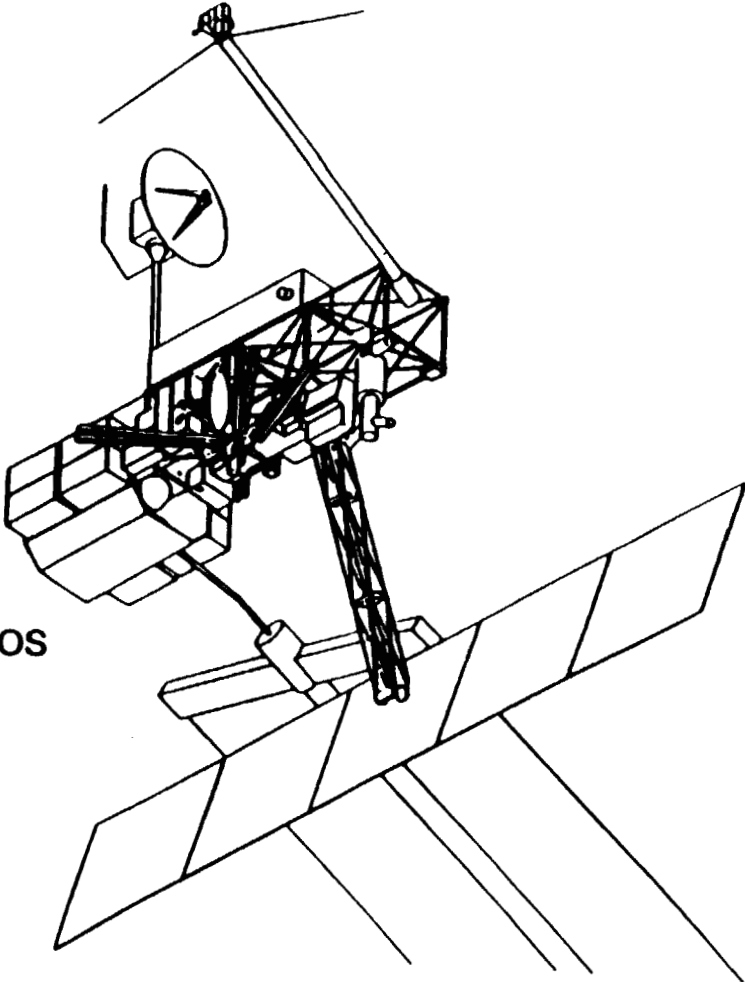
EOS
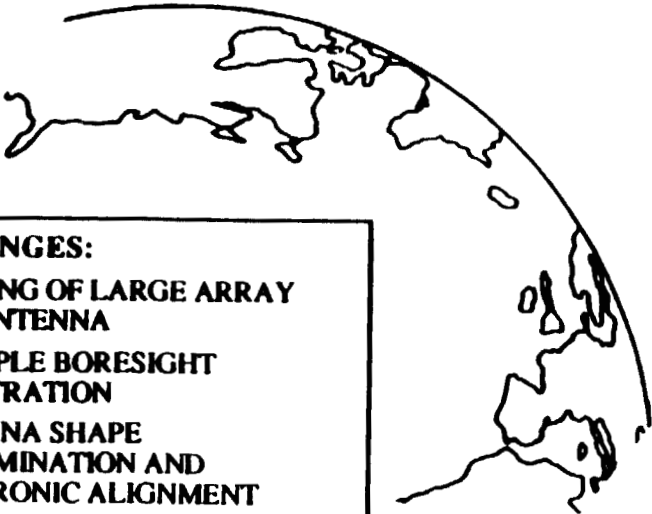
CHALLENGES:

- POINTING OF LARGE ARRAY AND ANTENNA
- MULTIPLE BORESIGHT REGISTRATION
- ANTENNA SHAPE DETERMINATION AND ELECTRONIC ALIGNMENT

42

ADVANCED ASTROPHYSICAL INSTRUMENTS

43
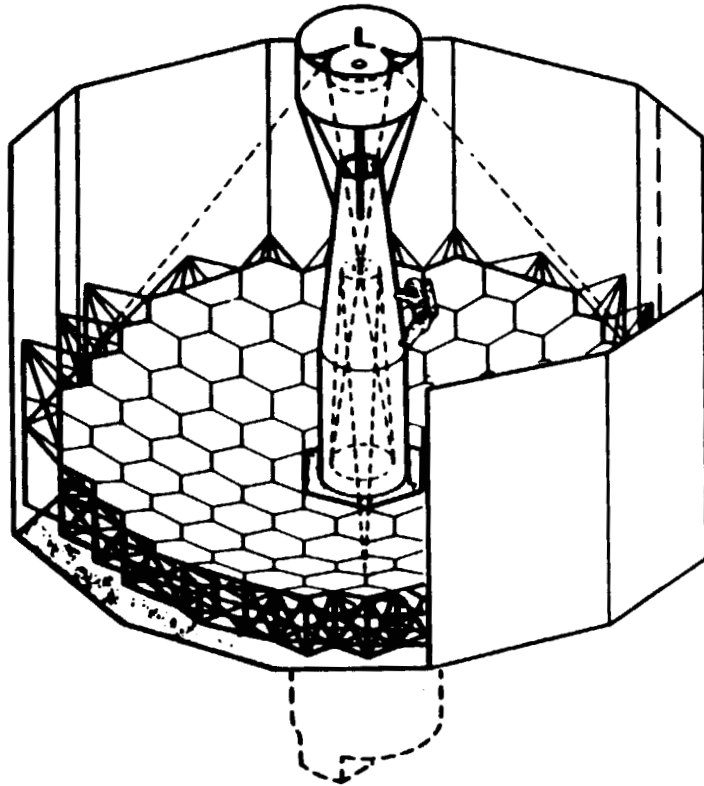
LDR
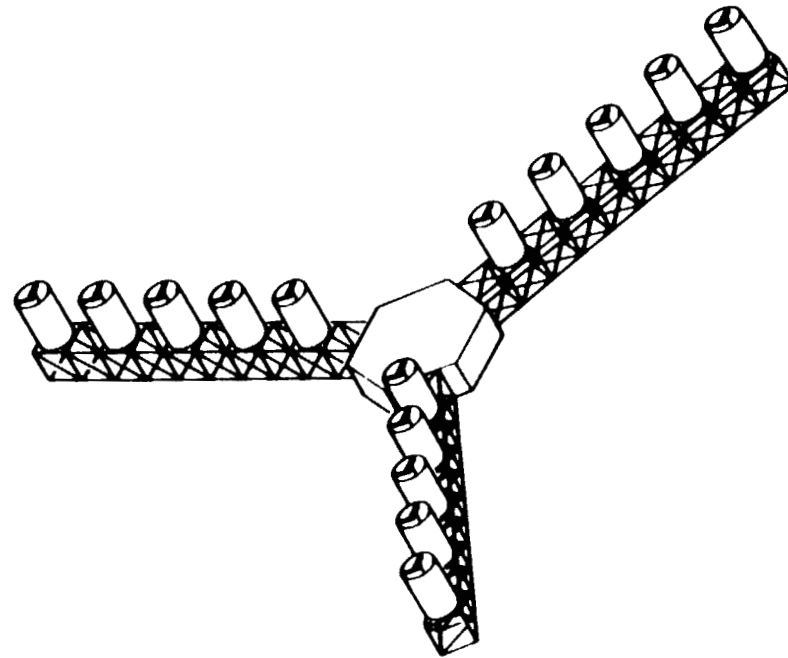
ASTROPHYSICAL INTEROMETER

CHALLENGES:
- SHAPE DETERMINATION AND ACTIVE CONTROL
- SUBWAVELENGTH PHASING OF OPTICAL PATHS
- DISTRIBUTED SENSING AND ACTUATION

CONTROL DESIGN AND ANALYSIS
NEEDS VS. CAPABILITIES

MODEL COMPLEXITY (NORMALIZED)

10^6

1

44

CURRENT CAPABILITIES

NUMBER OF IMPORTANT SYSTEM STATES

10 100 1000

1988 **2000** **CALENDAR YEAR**
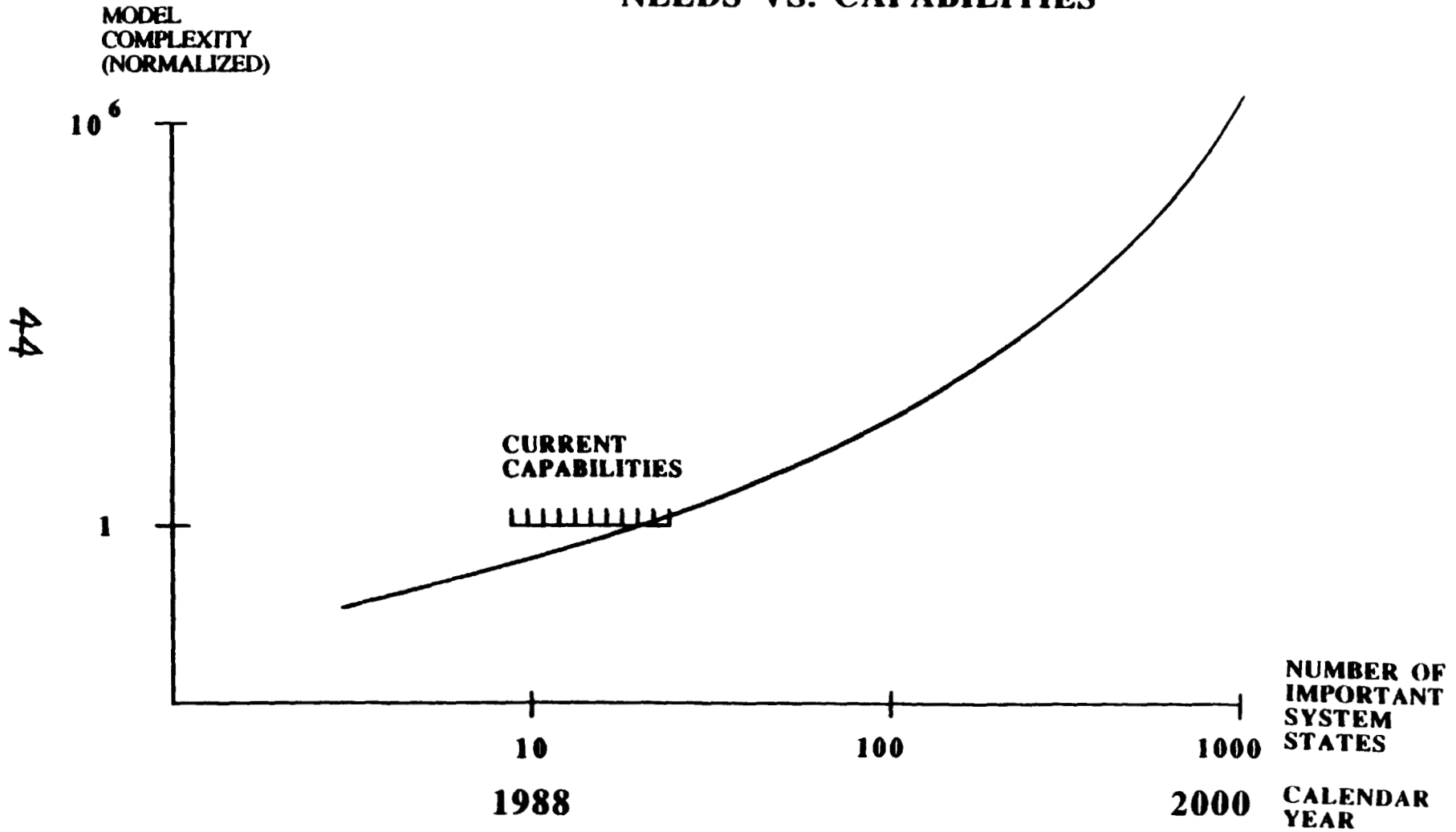
EXISTING TOOLS ARE A LIMITING FACTOR IN TODAY'S CONTROL DESIGN AND VERIFICATION, AND ARE INADEQUATE FOR FUTURE NEEDS

COMPUTATIONAL CONTROLS APPPROACH

45

A.
TECHNOLOGY ASSESSMENT
& REQUIREMENT DEFINITIONS

B.
EXISTING TOOLS
UPGRADE

EXISTING
TOOLS

NEXT
GENERATION
TOOLS

C.
NEXT GENERATION G&C
DESIGN & ANALYSIS TOOLS
DEVELOPMENT

88   89   90   91   92   93   94   95   FISCAL YEAR
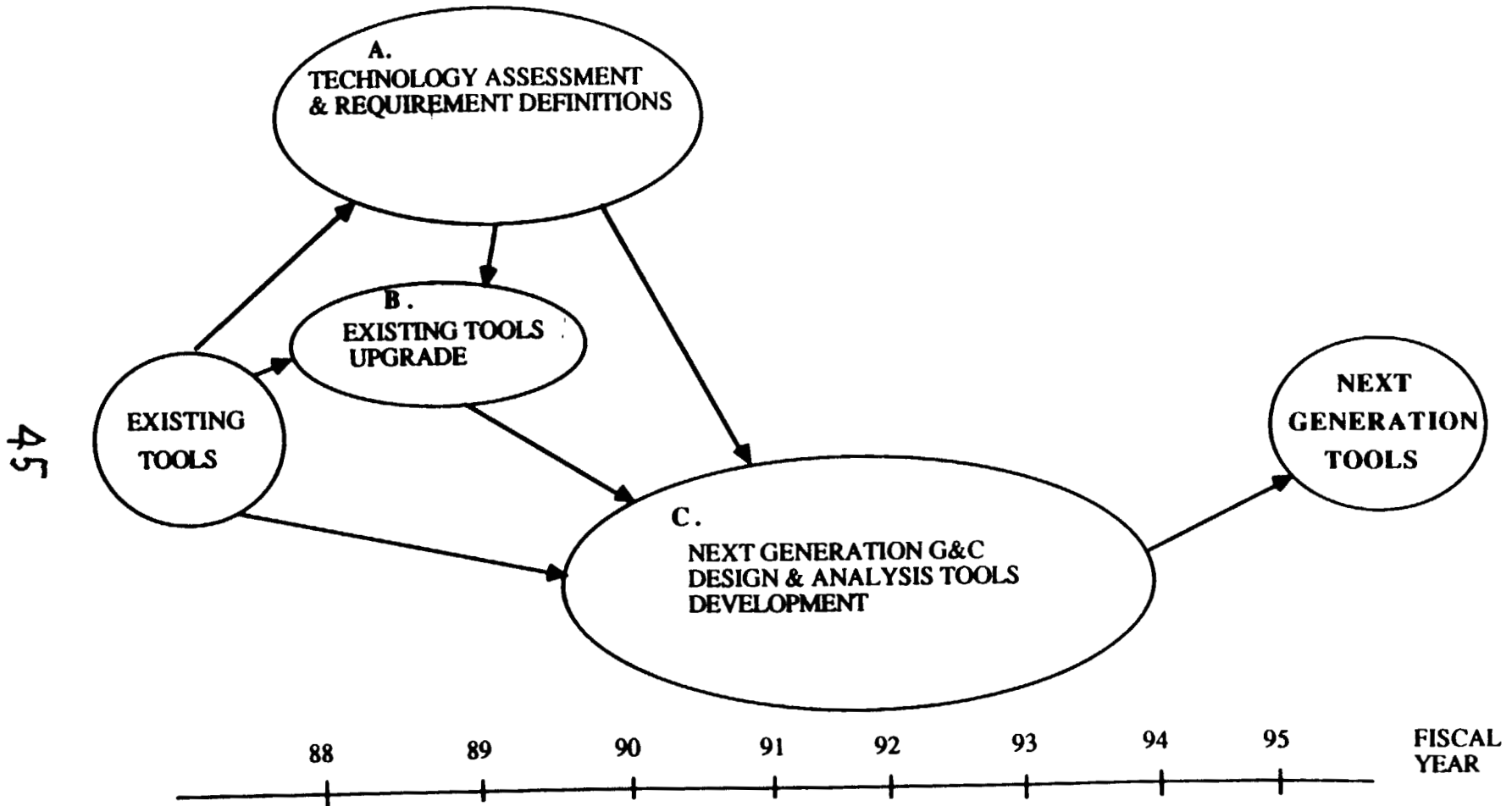
COMPUTATIONAL CONTROLS APPROACH CONT.

A. TECHNOLOGY ASSESSMENT & REQUIREMENT DEFINITIONS

- MULTIBODY SIMULATION TECHNOLOGY VERIFICATION

- CONTROL SYSTEM DESIGN/ANALYSIS TOOL ASSESSMENT

- REQUIREMENT DEFINITION AND ANALYSIS

B. EXISTING TOOLS UPGRADE

- UPDATE TOOLS WITH KNOWN DEFICENCIES

- UPGRADE TOOLS TO MEET NEAR TERM NEEDS

C. NEXT GENERATION TOOLS DEVELOPMENTS

- MULTIBODY SIMULATION TOOLS

- CONTROL SYSTEM OPTIMIZATION

- TOOLS FOR MODERN COMPUTING ENVIRONMENT

- ACCURATE SURFACE MODELING & REPRESENTATION TOOLS

- INTEGRATED CONTROL DESIGN ENVIRONMENT

46

MULTIBODY SIMULATION ASSESSMENT & VERIFICATION PLAN

PLAN SUMMARY:

● **ESTIMATED DURATION:**
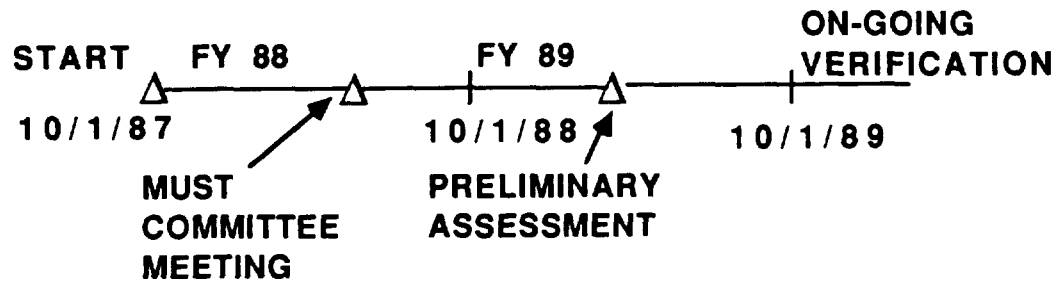
SCHEDULE:

1ST YEAR
- REQUIREMENT DEFINITION AND ANALYSIS
- ESTABLISH VERIFICATION LIBRARY
- TEST CASE DEVELOPMENT

2ND YEAR
- TEST CASE EXECUTION AND EVALUATION
- EXPERIMENT EXECUTION AND EVALUATION
- TEST REPORT GENERATION

FUTURE YEARS:
- CONTINUE TO BUILD VERIFICATION LIBRARY
- VERIFY NEW TOOLS AS THEY ARE DEVELOPED

DELIVERABLES:
- QUESTIONNAIRES
- REQUIREMENTS MATRIX
- TEST PLAN
- TEST CASE REPORT
- FINAL REPORT
- TWO WORKSHOPS

    - COMPUTATIONAL ASPECTS OF FLEXIBLE BODY SYSTEMS

    - FINAL REPORT TO THE COMMUNITY

47

JPL

MULTIBODY SIMULATION TECHNOLOGY ASSESSMENT AND VERIFICATION PLAN

	MILESTONES	FY'87	FY'88	FY'89
		O N D J F M A M J J A S	O N D J F M A M J J A S	O N D J F M A M J J A S
1	SETUP ORGANIZATIONAL MECHANISM			
2				
3	REQUIREMENT DEFINITION			
4	MISSION REQUIREMENTS			
5	SIMULATION REQUIREMENTS			
6				
7	CODE ANALYSIS & TEST CASE DEV			
8	SIMPLE AND COMPLEX MODELS			
9				
10	EXPERIMENT DESIGN			
11	GROUND (NASA)			
12	ON-ORBIT			
13				
14	TEST EXECUTION AND EVALUATION			
15	SIMULATION			
16	IN-ORBIT (LACE)			
17				
18	TEST REPORT GENERATION			
19	WORKSHOPS			
20	MUST COMMITTEE MEETINGS			
21				
22	MUST ASSESSMENT			
23	MUST VERIFICATION			

ADDITIONAL SOFTWARE DEVELOPMENTS WANTED FOR MODELING AND CONTROL OF FLEXIBLE SPACE SYSTEMS

By

Dr. Jiguan Gene Lin
Control Research Corporation
Lexington, Massachusetts

ABSTRACT

Existing modeling and control software packages are either inadequate or inefficient for applications to flexible space structures. Some additional software developments are wanted for effective design and evaluation of the control systems. The following will be discussed in this presentation.

1. Linear-quadratic optimal regulators as usual can be designed using various "modern control" design software packages. To design for active augmentation of (approximately) the specified amount of active damping to each "controlled modes," the common practice is to adjust repeatedly the state and control weights (i.e., the Q and R matrices) by mostly endless trial and error. The time consumed and effort spent in the trial-and-error repetition can be saved by using an analytical procedure for closely estimating the corresponding state and control weights. Various numerical examples have shown that this is possible. No software has been developed for automating such a time-saving analytical assignment procedure yet.

2. "Modal dashpots" are very effective output-feedback vibration controllers for flexible structures, not only effective for augmenting a small amount of active damping to a large number of vibration modes (like the so-called low-authority structural controllers), but also effective for quick suppression of large vibrations (like high-authority structural controllers). Recent numerical results on orbital SCOLE configuration have shown so. No software has been developed for facilitating the design process yet.

3. The actual performance of any control design needs to be evaluated against a faithful model of the flexible structure to be controlled. The potential of destabilization or serious performance degradation needs to be detected by numerical simulation of the structure with the control loops being closed. Except for some trivial cases, reduced=order normal-mode models are generally not appropriate: if they are computationally feasible to simulate the closed-loop system, then they are likely not accurate enough to represent the dynamics of the flexible structure; if they are satisfactorily accurate, then they are mostly too large for effective dynamic simulation even by a state-of-the-art mainframe computer. Besides, computing a very large number of normal modes is very expensive, and the accumulated computational errors in the natural frequencies and mode shapes grow very rapidly. The popular Guyan reduction technique is often used to reduce the large finite-element mass-stiffness model first. Such a reduction technique, unfortunately, introduces large additional errors which are proportional to the square of the natural frequency of the modes computed thereafter.

There is a trend towards some innovative use of non-normal modes (such as Ritz or Lanczos vectors) for representing the structures by a much smaller number of such modes. Available results are interesting and promising. Additional development effort is needed and will be very worthwhile.

ADDITIONAL SOFTWARE DEVELOPMENTS WANTED FOR
MODELING AND CONTROL OF FLEXIBLE SPACE SYSTEMS

Jiguan Gene Lin
Control Research Corp,
Lexington, Massachusetts

51

ADDITIONAL SOFTWARE DEVELOPMENTS **URGENTLY** WANTED

● Accuracy-Preserving Computationally Efficient
  Coordinate Reduction of Finite-Element Models,

        To ENABLE

  1. Pre-Design Open-Loop Dynamic Analysis of
     Realistic, Large, Flexible Space Structures
  AND

  2. Post-Design **Full-Order Closed-Loop** Evaluation of
     Control Systems for such Structures

● Analytical Selection of Control and State Weights,

        TO AID

  Design of Linear-Quadratic Regulators desired for
        Vibration control of Flexible Space Structures

52

ACCURACY-PRESERVING COMPUTATIONALLY EFFICIENT
COORDINATE REDUCTION OF FINITE-ELEMENT MODELS

DESIGN OF RELIABLE CONTROL SYSTEMS FOR FLEXIBLE SPACE SYSTEMS

NEEDS

1. CAREFUL PRE-DESIGN OPEN-LOOP DYNAMIC ANALYSIS OF
   THE SPACE STRUCTURE, AND

2. CAREFUL POST-DESIGN FULL-ORDER CLOSED-LOOP EVALUATION OF
   CONTROL SYSTEMS FOR THE STRUCTURE

NEEDS PRE-DESIGN OPEN-LOOP DYNAMIC ANALYSIS
   -- TO ASSESS EFFECTS OF DISTURBANCES ON SYSTEM PERFORMANCE,
      E.G., POINTING STABILITY, LINE-OF-SIGHT ERRORS, ...
   -- TO IDENTIFY STRUCTURAL MODES NEEDING ACTIVE CONTROL
   -- TO FORM A COMPUTATIONALLY FEASIBLE
      REDUCED-ORDER CONTROL-DESIGN MODEL
   -- TO ASSSESS EFFECTIVENESS OF CONTROL ACTUATORS AND SENSORS

NEEDS POST-DESIGN FULL-ORDER CLOSED-LOOP EVALUATION
   -- TO DETECT POSSIBLE INSTABILITY INTRODUCED BY
      REDUCED-ORDER CONTROL DESIGN
   -- TO VERIFY ACTUAL TIME-DOMAIN PERFORMANCE
   -- TO TEST ROBUSTNESS TO MODELING ERRORS, PARAMETER VARIATIONS,...
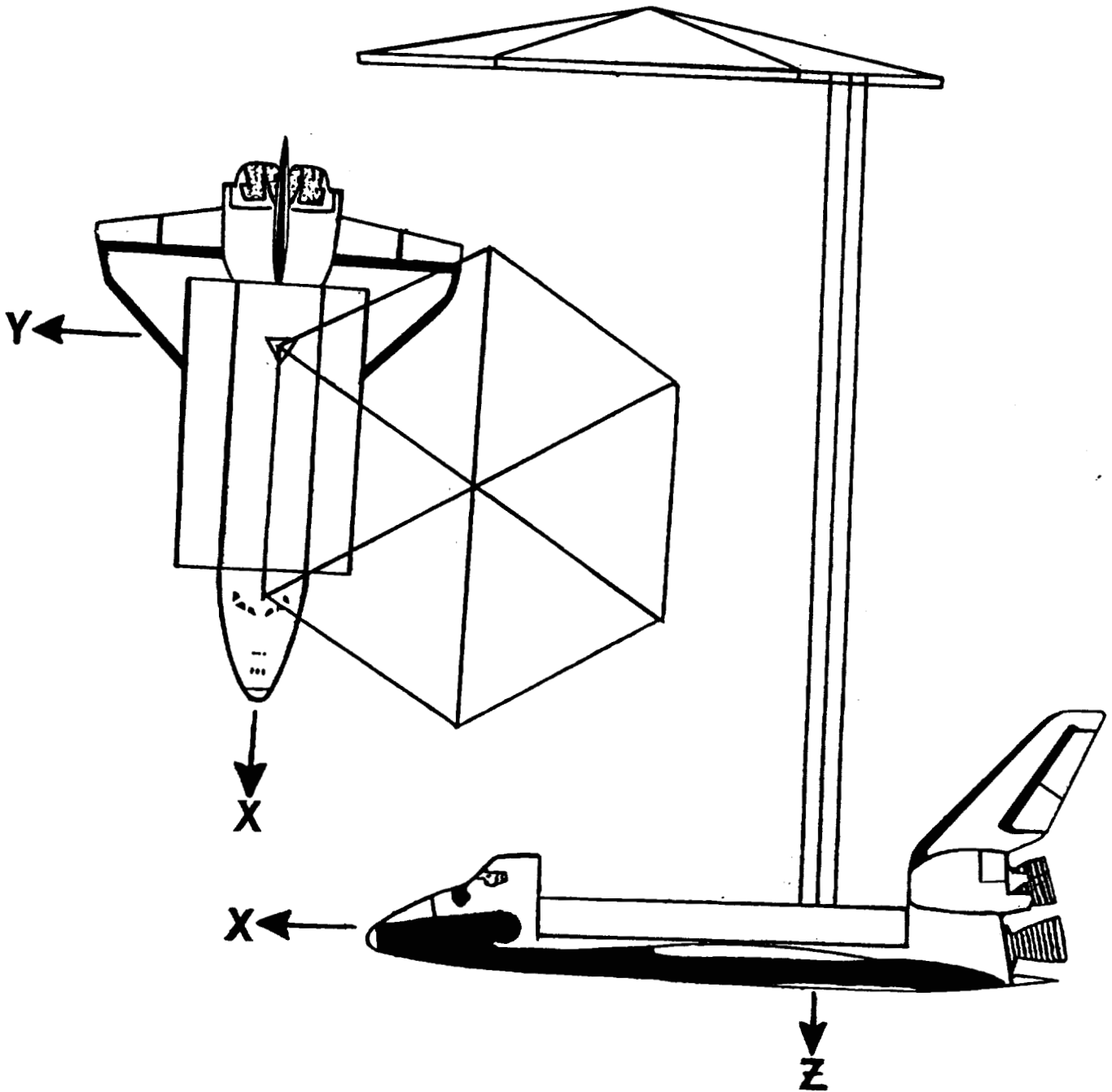
Fig. 1-1 Spacecraft Control Laboratory Experiment (SCOLE)--
the orbital Shuttle-Mast-Antenna configuration.

54
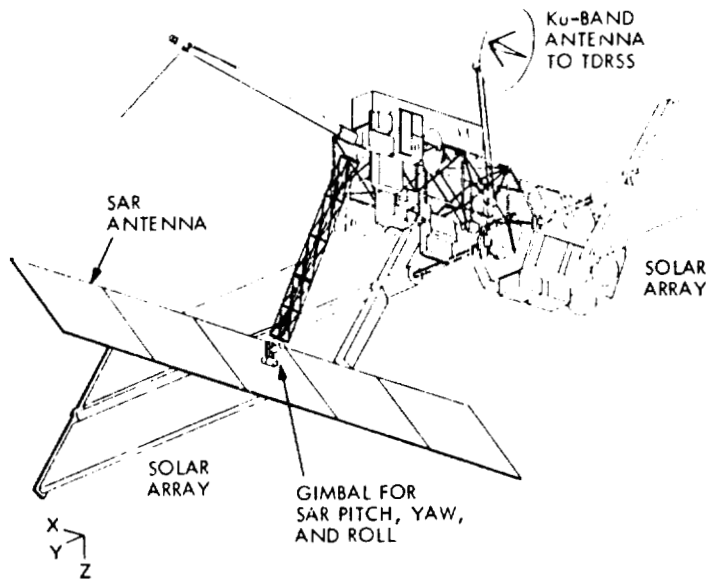
Figure 1. A Possible Eos Polar Platform Configuration
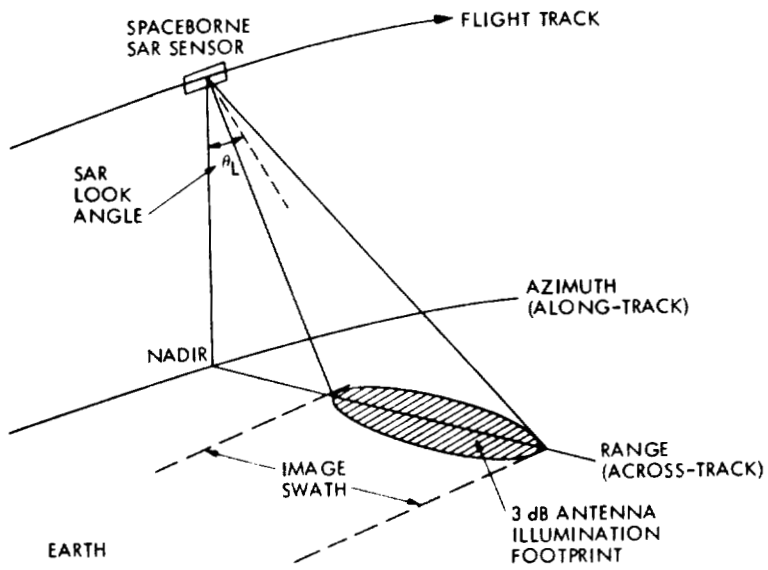
Figure 2. SAR Imaging Geometry

AIAA-87-0022 PAPER:
"EARTH OBSERVING SYSTEM INSTRUMENT POINTING CONTROL
MODELING FOR POLAR ORBITING PLATFORM,"
BY H.C. BRIGGS, T.KIA, S.A. McCABE, AND C.E. BELL

● NASTRAN STRUCTURAL DYNAMICS MODEL OF PLATFORM: 200 NODES
   -- 600 TO 1,200 DOFs
   -- 40 MODES BELOW 10 Hz

● SLEW OF SAR ANTENNA ABOUT ITS BOOM AXIS WAS SIMULATED
   -- TO ASSESS THE POINTING CONTROL AND STABILITY
        OF INSTRUMENTS MOUNTED ON THE CARRIER STRUCTURE
      DURING SLEWING OF ADJACENT INSTRUMENTS.

● "LARGE ANGLE TIME-DOMAIN SIMULATIONS CAN PRESENTLY BE CONDUCTED
USING DISCOS, BUT DUE TO EXECUTION EXPENSE AND THE DIFFICULTY OF USER
INTERFACE THIS APPROACH IS IMPRACTICAL FOR EOS STUDIES.

NEXT GENERATION SIMULATION TOOLS WHICH REDUCE THE NUMBER OF NUMERICAL
OPERATIONS FROM ORDER N^4 (DISCOS) TO N^3 (TREETOPS) AND BEYOND TO ORDER N
ARE URGENTLY NEEDED TO EFFICIENTLY AND COST EFFECTIVELY VERIFY
THE PERFORMANCE OF LARGE SYSTEMS OF MULTIPLE ARTICULATED AND ROTATING
ELEMENTS SUCH AS EOS PLATFORMS."

SOME SERIOUS TECHNICAL PROBLEMS

CURRENT REDUCED-ORDER MODELS ARE GENERALLY NOT APPROPRIATE FOR
REALISTIC, LARGE, FLEXIBLE SPACE STRUCTURES:

● <u>MODEL ACCURACY VS COMPUTATIONAL CAPABILITY</u>

    IF COMPUTATIONALLY FEASIBLE TO SIMULATE ON THE COMPUTER, THEN
      LIKELY NOT ACCURATE ENOUGH TO REPRESENT THE FLEXIBLE STRUCTURE;

   IF SATISFACTORILY ACCURATE, THEN
      MOSTLY TOO LARGE FOR EFFECTIVE DYNAMIC SIMULATION ON COMPUTER

● <u>COMPUTATIONAL EXPENSE AND ACCUMULATED ERRORS</u>

    COMPUTING A VERY LARGE NUMBER OF NORMAL MODES IS VERY EXPENSIVE;

    ACCUMULATED COMPUTATIONAL ERRORS IN THE NATURAL FREQUENCIES AND
      MODE SHAPES GROW VERY RAPIDLY.

● <u>WASTED EXPENSIVE MODAL COMPUTATIONS</u>

    MANY USELESS MODES COMPUTED,
      THEN IGNORED IN CONTROL DESIGN OR EVALUATION

    -- UN-RELATED TO DISTURBANCES CONCERNED,
             OR CONTROL ACTUATIONS CONSIDERED

● <u>ACCURACY-SACRIFICING COORDINATE REDUCTION</u>

    POPULAR GUYAN REDUCTION TECHNIQUE IS OFTEN USED FIRST
      TO REDUCE THE LARGE FINITE-ELEMENT STIFFNESS AND MASS MATRICES
-- LARGE ERRORS INTRODUCED THEREBY;
    INCREASE AS THE **SQUARE** OF FREQUENCIES OR HIGHER

INNOVATIVE RAYLEIGH-RITZ METHOD

● A TREND TOWARDS SOME INNOVATIVE USE OF NON-MORMAL MODES
(SUCH AS RITZ OR LANCZOS VECTORS) FOR REPRESENTING THE STRUCTURES BY
A MUCH SMALLER NUMBER OF GENERALIZED COORDINATES
-- AVAILABLE RESULTS INTERESTING AND PROMISING.
-- ADDITIONAL DEVELOPMENT AND EXTENTION EFFORTS NEEDED.

● RAYLEIGH-RITZ METHOD

-- ASSUMED SHAPES: $Q_1, Q_2, \ldots, Q_N$ (SMALL N)

-- APPROXIMATE THE STRUCTURAL DISPLACEMENT VECTOR X:

$$x = z_1 Q_1 + z_2 Q_2 \ldots + z_N Q_N = Q z$$

$$Q = \begin{bmatrix} Q_1, Q_2, \ldots, Q_N \end{bmatrix}, \qquad z = (z_1, z_2, \ldots, z_N)$$

-- REDUCE ORIGINAL FINITE-ELEMENT MODEL: $M \dfrac{D^2 x}{DT^2} + K x = F(T)$

TO
$$\boxed{(Q^T M Q) \frac{D^2 z}{DT^2} + (Q^T K Q) z = Q^T F(T)}$$

-- ORIGINAL LARGE MATRICES M AND K NOW REDUCED TO
        SMALLER ONES:
$$M_C = Q^T M Q, \qquad K_C = Q^T K Q$$

● WILSON-YUAN-DICKENS ALGORITHM

-- ASSUME $F(T) = B U(T),$ $U(T) = $ A SCALAR FUNCTION

-- GENERATE AND ORTHOGONALIZE THE ASSUMED SHAPES SEQUENTIALLY:

$$K Q_1^* = B \quad ============> Q_1$$
$$K Q_2^* = M Q_1 \quad =========> Q_2$$
$$\ldots\ldots\ldots\ldots\ldots\ldots$$
$$K Q_N^* = M Q_{N-1} \quad ========> Q_N$$

ADDITIONAL DEVELOPMENT AND EXTENTION EFFORTS WANTED

● COMPUTATIONAL PROBLEMS WITH ORTHOGONALIZATION

1. ACCUMULATED ROUNDOFF ERRORS CAN DESTROY THE ORTHOGONALITY OF THE RITZ VECTORS THUS GENERATED
   -- NEED TO RE-ORTHOGONALIZE WHENEVER ORTHOGONALITY IS LOST

2. COMPUTATIONAL INTENSIVE: PERFORM GRAM-SCHMIDT ORTHOGONALIZATION EVERY TIME A VECTOR Q_I^* IS GEANERATED
   -- NOUR-OMID AND CLOUGH'S SOLUTION WAS TO ORTHOGONALIZE ONLY WITH RESPECT TO TWO PREVIOUS VECTORS.

   -- THE MOST TROUBLESOME DRAWBACK OF THE LANCZOS ALGORITHM REAPPEAR:
       EASY LOSS OF ORTHOGONALITY OF THE LANCZOS VECTORS;
       RE-ORTHOGONALIZATION REQUIRED WHEN ORTHOGONALITY IS LOST

● EXTENSION BEYOND THE SPECIAL CASE OF SCALAR FORCES

   -- THE WILSON-YUAN-DICKENS ALGORITHM WAS FORMULATED FOR
       SCALAR FORCES;
     NOT DIRECTLY APPLICABLE TO THE GENERAL CASE OF
       MULTIPLE SIMULTANEOUS DISTURBANCE (OR CONTROL) FORCES

   -- SO WAS NOUR-OMID AND CLOUGH'S VERSION USING LANCZOS VECTORS

   -- BUT, SPACE SYSTEMS LIKELY BE SUBJECT TO MULTIPLE DISTURBANCES
       NOT ONE AT A TIME, BUT SIMULTANEOUSLY

   -- ALSO MOST CONTROL SYSTEMS USE MULTIPLE INDEPENDENT ACTUATORS
       TO APPLY FORCES/TORQUES TO THE STRUCTURES SIMULTANEOUSLY.

LINEAR-QUADRATIC REGULATORS (LQR) FOR FLEXIBLE SPACE STRUCTURES

● TRUNCATED MODAL MODEL OF THE FLEXIBLE STRUCTURE

$$\ddot{\eta}_I + 2 \zeta_I \omega_I \dot{\eta}_I + \omega_I^2 \eta_I = \phi_I^T B_F U \quad I = 1,\ldots,N$$

● PUTTING INTO STATE-SPACE FORM

$$\dot{x} = A x + B u$$

WITH $\quad x = \begin{bmatrix} \eta \\ \dot{\eta} \end{bmatrix} \quad \eta = \begin{bmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_N \end{bmatrix}$

● LQR DESIGN:

FIND A FEEDBACK GAIN MATRIX K SUCH THAT

$$J = \int_0^\infty (x^T Q x + u^T R u) \, DT$$

IS MINIMIZED WITH $\quad u = K x$

● GIVEN THE CONTROL AND STATE WEIGHTING MATRICES R AND Q, ANY "MODERN CONTROL" DESIGN PROGRAM, SUCH ORACLS, CTRL-C, CAN PRODUCE AN OPTIMAL SOLUTION K VIRTUALLY AUTOMATICALLY

DESIGN OF LINEAR-QUDRATIC REGULATORS FOR
ACTIVE AUGMENTATION OF SPECIFIIED DAMPING TO SPECIFIC MODES

APPRAOACH 1. CONSTRAINED OPTIMIZATION

OPTIMIZE THE PERFORMANCE INDEX J WITH THE SPECIFIED DAMPING RATIOS
AS CONSTRAINTS.

-- CONSTRAINED OPTIMIZATION IS PARTICULARLY COMPLICATED
WHEN DYNAMIC EQUATIONS ARE INVOLVED

APPROACH 2. ALPHA-SHIFT

SHIFT ALL POLES TO THE LEFT OF THE IMAGINERY AXIS BY A CONSTANT α.

-- SOME MODES MAY NOT GET ENOUGH DAMPING TO BE CLOSE TO THE SPECIFIED,
WHILE SOME OTHERS MAY GET TOO MUCH MORE THAN THE SPECIFIED.

APPROACH 3. TRIAL AND ERROR ON THE CONTROL AND STATE WEIGHTS

START WITH DIAGONAL R AND Q WITH SOME ARBITRARY NUMBERS, E.G., 1;
CARRY OUT THE DESIGN OF THE CORRESPONDING LQR;
EVALUATE THE CLOSD-LOOP POLES, AND HENCE THE DAMPING RATIOS.

TRY OTHER CONTROL AND STATE WEIGHTS,
REPEAT THE DESIGN-EVALUATION CYCLE,
UNTILL THE RESULTS ARE SATISFACTORY.

-- THE CONTROL AND STATE WEIGHTS USED MOSTLY ARE AD HOC;
THE TRIAL-AND-ERROR PROCESS IS MOSTLY ENDLESS,
VERY TIME CONSUMING

ADDITIONAL SOFTWARE DEVELOPMENT WANTED

SOFTWARE MODULES FOR AIDING DESIGNERS IN MAKING GOOD INITIAL CHOICES,
AND INTERMEDIATE ADJUSTMENTS, OF THE CONTROL AND STATE WEIGHTS

SO THAT,

THE RESULTING DESIGN OF LINEAR-QUADRATIC REGULATORS
CAN, WITHIN ONLY A FEW ITERATIONS, SATISFY CLOSELY
THE DESIGN SPECIFICATIONS,
E.G., ON DAMPING AUGMENTATION, STIFFNESS AUGMENTATION,
LINE-OF-SIGHT POINTING ACCURACY, ETC.

ADDITIONAL SOFTWARE DEVELOPMENTS **URGENTLY** WANTED

● ACCURACY-PRESERVING COMPUTATIONALLY EFFICIENT
  COORDINATE REDUCTION OF FINITE-ELEMENT MODELS,

  To ENABLE

  1. PRE-DESIGN OPEN-LOOP DYNAMIC ANALYSIS OF
     REALISTIC, LARGE, FLEXIBLE SPACE STRUCTURES
  AND

  2. POST-DESIGN **FULL-ORDER CLOSED-LOOP** EVALUATION OF
     CONTROL SYSTEMS FOR SUCH STRUCTURES

● ANALYTICAL SELECTION OF CONTROL AND STATE WEIGHTS,

  TO AID

  DESIGN OF LINEAR-QUADRATIC REGULATORS DESIRED FOR
          VIBRATION CONTROL OF FLEXIBLE SPACE STRUCTURES

63

SESSION II - SURVEY OF AVAILABLE SOFTWARE

65

FLEXIBLE STRUCTURE CONTROL EXPERIMENTS USING A REAL-TIME WORKSTATION FOR COMPUTER-AIDED CONTROL ENGINEERING

By

Michael E. Stieber
Communications Research Centre
Ottawa, Ontario, CANADA

ABSTRACT

A Real-Time Workstation for Computer-Aided Control Engineering has been developed jointly by the Communications Research Centre (CRC) and Ruhr-Universitaet Bochum (RUB), West Germany. The system is presently used for the development and experimental verification of control techniques for large space systems with significant structural flexibility.

The Real-Time Workstation (cf. Attachment 1) essentially is an implementation of RUB's extensive Computer-Aided Control Engineering package "KEDDC" on an INTEL micro-computer running under the RMS real-time operating system. The portable system supports system identification, analysis, control design and simulation, as well as the immediate implementation and test of control systems. A wealth of classical and modern control analysis and design methods are available to the user who interacts with KEDDC through a friendly dialog. The workstation can be configured both with analog and digital interfaces to the "real world" for data acquisition and control.

The Real-Time Workstation is currently being used by CRC to study control/structure interaction on a ground-based structure called "DAISY" (cf. Attachment 2), whose design was inspired by a reflector antenna. DAISY emulates the dynamics of a large flexible spacecraft with the following characteristics: rigid body modes, many clustered vibration modes with low frequencies and extremely low damping. DAISY presently has seven control actuators and eight sensors which are all "spacecraft-like."

The class of control algorithms currently investigated by experiments is "robust LQG" control. The Real-Time Workstation was found to be a very powerful tool for experimental studies, supporting control design and simulation, and conducting and evaluating tests within one integrated environment. It has dramatically increased the flexibility and turnaround of the experiments. As the Workstation all but eliminates the barriers between ideas on control systems and their experimental evaluation, analytical and experimental development can take place essentially simultaneously.

REAL-TIME WORKSTATION FOR
COMPUTER-AIDED CONTROL ENGINEERING

DAISY

- SYSTEM / SIGNAL ANALYSIS
- CONTROL DESIGN
- SIMULATION
- SYSTEM IDENTIFICATION
- *RT CONTROL OPERATION*

INTEL | RMX

IEEE 488 BUS

89
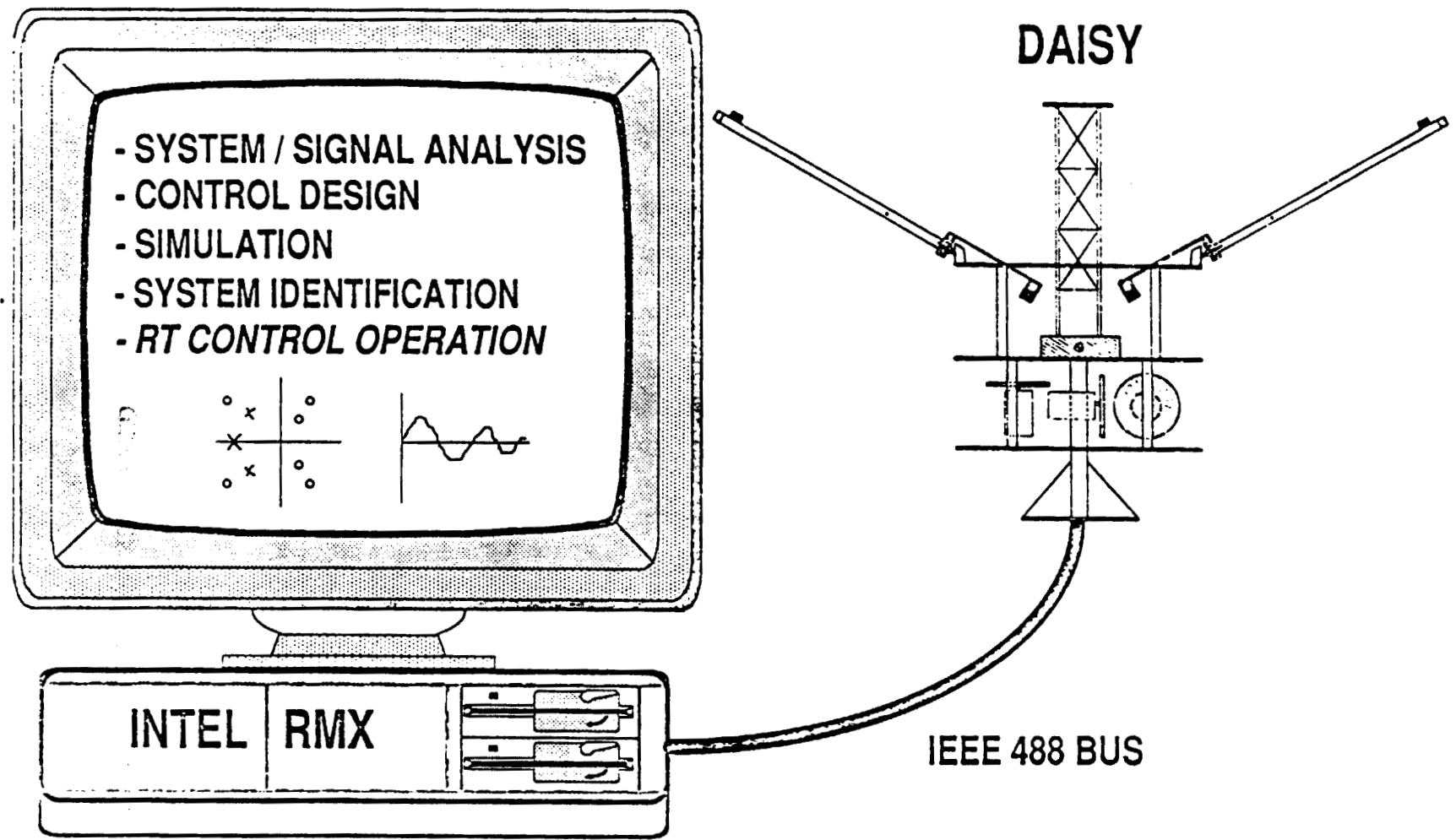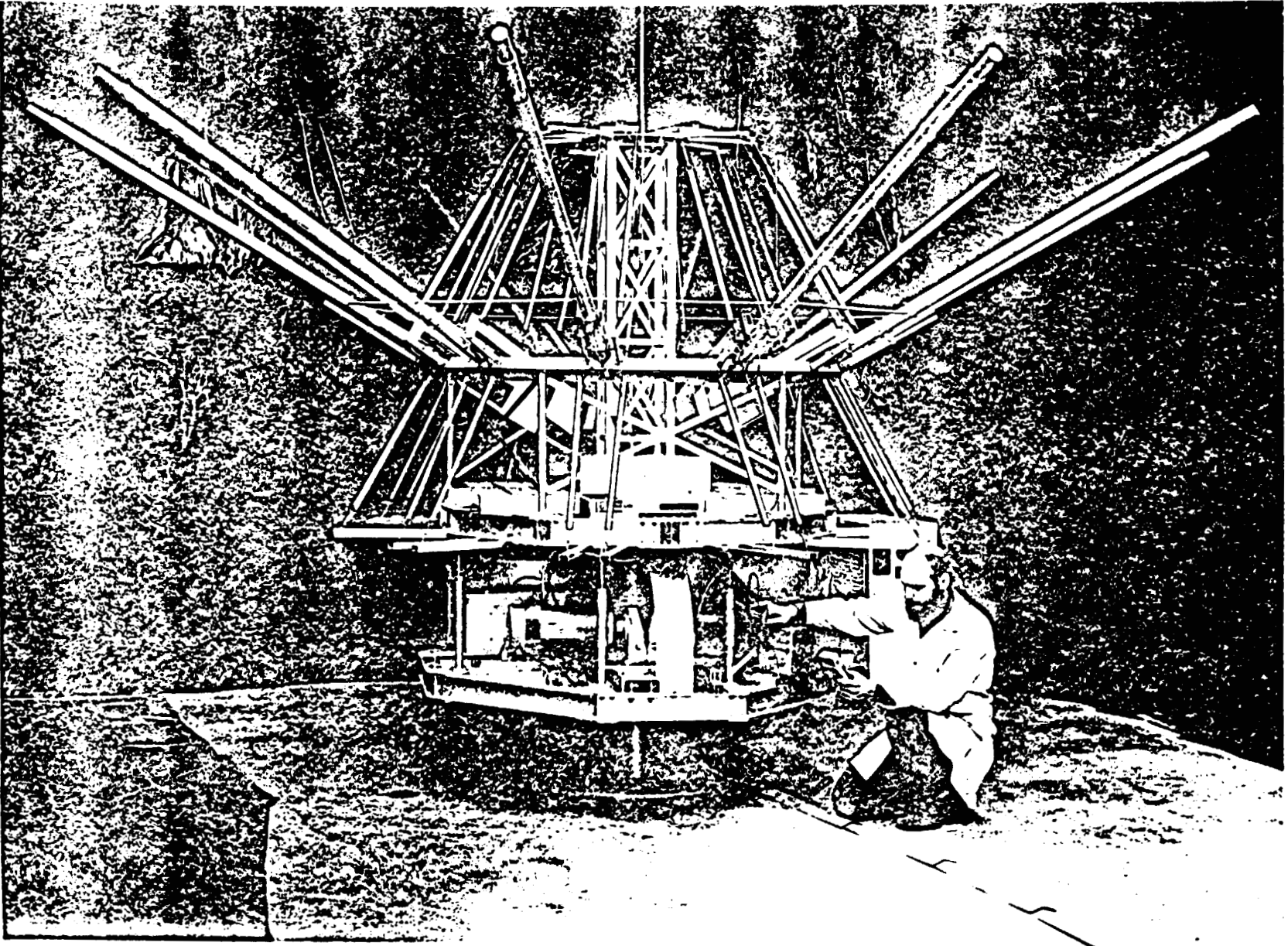
FLEXIBLE STRUCTURE CONTROL EXPERIMENTS

USING A REAL-TIME WORKSTATION FOR

COMPUTER-AIDED CONTROL ENGINEERING

MICHAEL E. STIEBER

SPACE MECHANICS DIRECTORATE

COMMUNICATIONS RESEARCH CENTRE, OTTAWA, CANADA

NASA WORKSHOP ON COMPUTATIONAL ASPECTS IN THE CONTROL OF FLEXIBLE STRUCTURES, JULY 12-14,1988

OUTLINE

1. INTRODUCTION

2. REAL-TIME WORKSTATION

- CAPABILITIES

- HOST ENVIRONMENT

3. FLEXIBLE STRUCTURE CONTROL EXPERIMENT

- CHARACTERISTICS

- APPLICATION OF REAL-TIME WORKSTATION

4. SUMMARY & CONCLUSIONS

SPACE-BASED RADAR

SPACE-FED PHASED ARRAY ANTENNA CONCEPT

72
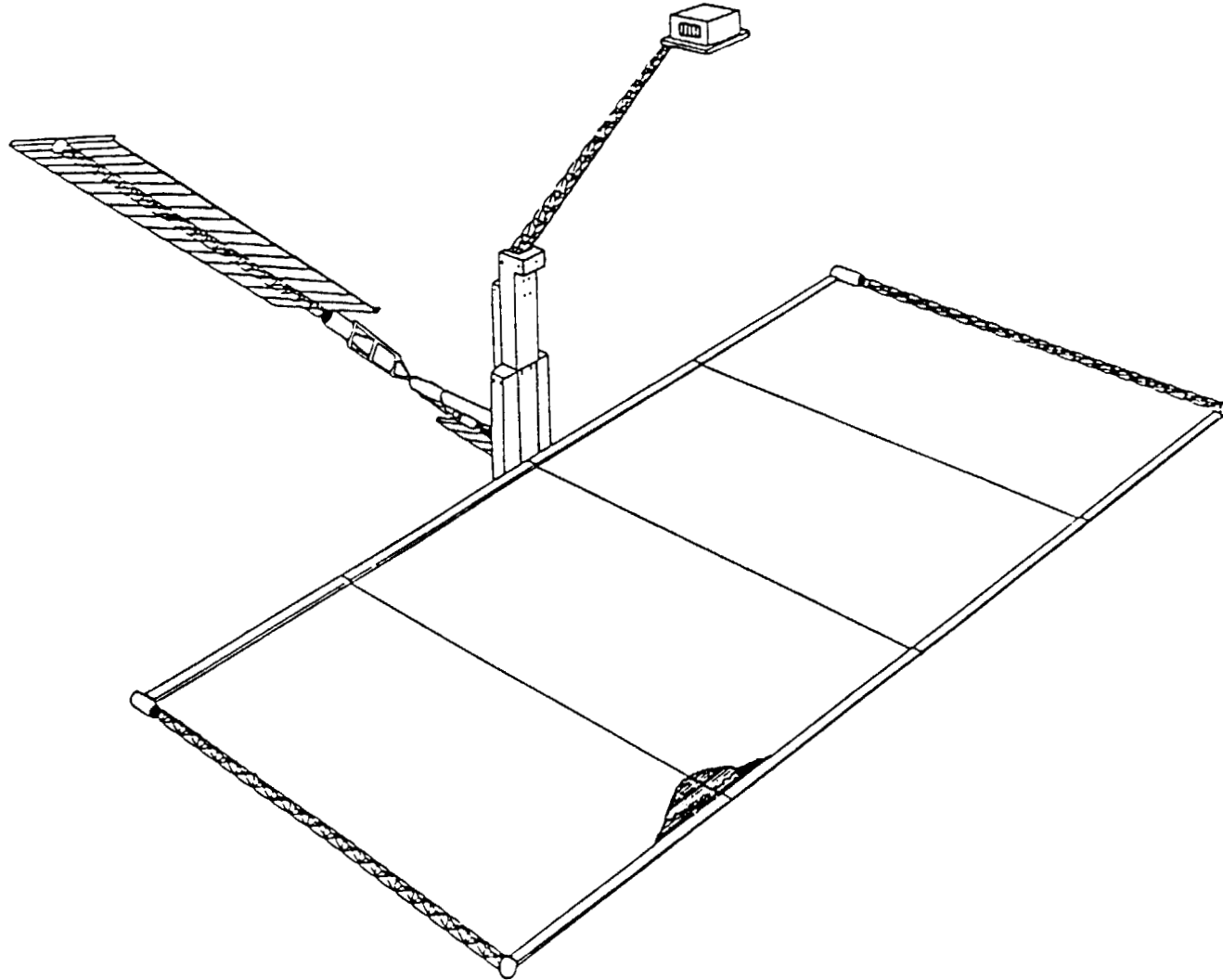
TECHNOLOGY DEVELOPMENT FOR

CONTROL OF FLEXIBLE SPACE STRUCTURES

- **ANALYTICAL STUDIES**
  DEVELOPMENT OF NEW TECHNIQES
  APPLICATION TO STRAWMAN PROBLEMS (SIMULATIONS)

- **GROUND-BASED EXPERIMENTS**
  VALIDATION AND DEMONSTRATION OF ANALYTICAL RESULTS

- **FLIGHT TEST**

SUPPORT BY CAD SYSTEMS ?

HOW DO CAD PACKAGES SUPPORT

CONTROL SYSTEM TECHNOLOGY DEVELOPMENT ?

MANY SUPPORT ANALYTICAL STUDIES

- NUMERICAL ANALYSIS

- GRAPHICS

FEW DIRECTLY SUPPORT EXPERIMENTAL STUDIES, WHICH REQUIRES:

- INTERFACE TO THE REAL WORLD

- DATA ACQUISITION

- IMPLEMENTATION & TEST OF REAL-TIME CONTROL SYSTEMS

74

REAL-TIME WORKSTATION

FLEXIBLE STRUCTURE CONTROL EXPERIMENT

- SYSTEM / SIGNAL ANALYSIS
- CONTROL DESIGN
- SIMULATION
- SYSTEM IDENTIFICATION
- *RT CONTROL OPERATION*

75
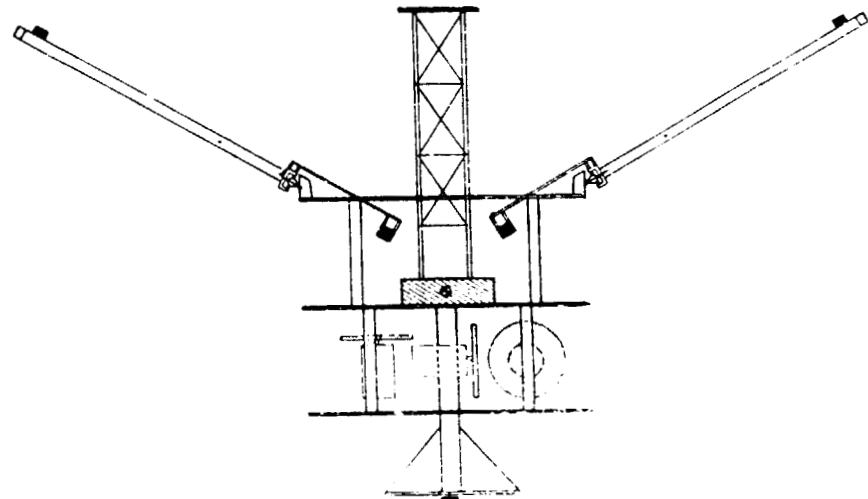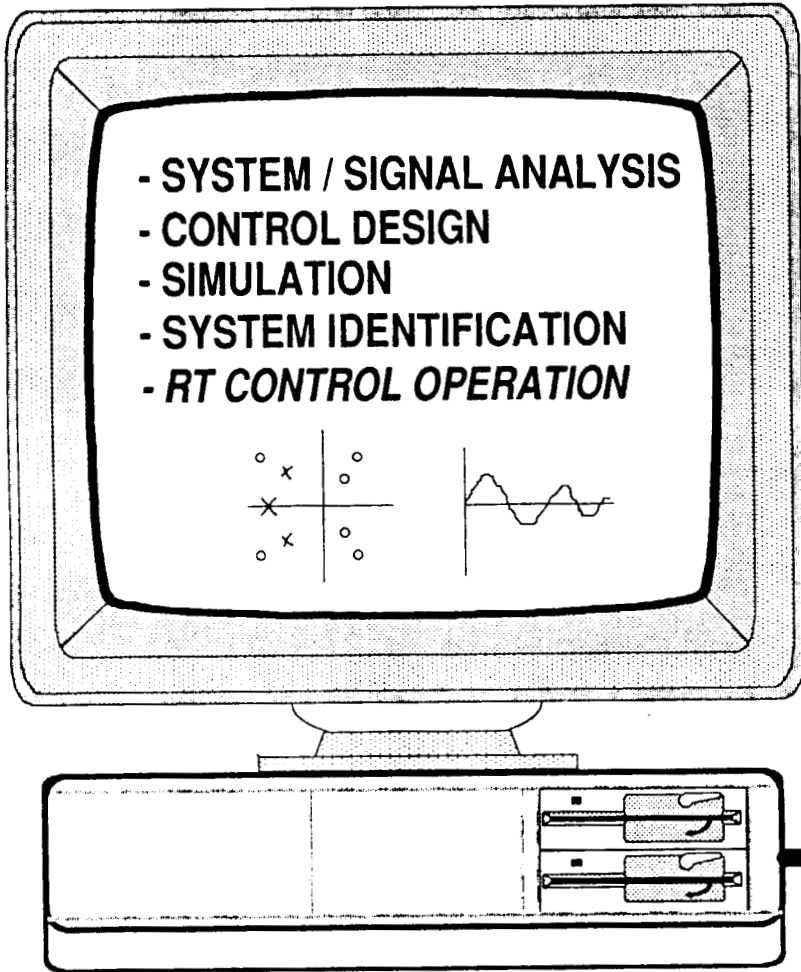
REAL-TIME WORKSTATION SOFTWARE

UNDERLYING CAD PACKAGE: *KEDDC*

- DEVELOPED BY DR. CHRISTIAN SCHMID

- AT RUHR-UNIVERSITY, BOCHUM, WEST GERMANY

- RT WORKSTATION A JOINT PROJECT OF RUHR-U. AND CRC

FEATURES

- MATURE

- COMPREHENSIVE

- **PORTABLE** (RUNNING UNDER 12 OPERATING SYSTEMS)

- MODULAR, OPEN SYSTEM

KEDDC

CORE MODULES

- MATRIX MANAGER

- SYSTEM MANAGER

- FREQUENCY MANAGER

- SIGNAL MANAGER

- POLYNOMIAL MATRIX MANAGER

- GRAPHICS MANAGER

CAPABILITY OF CORE PACKAGE

- INTERACTIVE 'CALCULATOR' -TYPE ENVIRONMENT

- 250 COMMANDS

- EXTENDED BY APPLICATIONS MODULES

HOST ENVIRONMENT

REQUIREMENTS FOR SELECTION

- REAL-TIME MULTI-TASKING OPERATING SYSTEM

- PORTABLE COMPUTER

- COMPATIBLE WITH FUTURE MICRO-PROCESSORS

SYSTEM CHOSEN (IN 1985): INTEL 286/310

- OPEN SYSTEM (MULTIBUS 1)

- CPU: INTEL 80286/80287

- OPERATING SYSTEM: INTEL RMX86

- UPGRADE TO 386-BASED RMX286 SYSTEM PLANNED

HOST ENVIRONMENT (CONT'D)

PERIPHERALS

- **GRAPHICS TERMINAL** (780 X 1024 RESOLUTION)

- **DOT MATRIX PRINTER**

REAL-TIME SIGNAL INTERFACE FOR DATA ACQ. AND CONTROL

- **IEEE 488 GPIB** (USED IN FLEXIBLE STRUCTURE CONTROL EXPERIMENT)

- **ANALOG SIGNALS**

DATA LINK TO REMOTE MAINFRAME

REAL-TIME WORKSTATION

FLEXIBLE STRUCTURE CONTROL EXPERIMENT "DAISY"

KEDDC
- SYSTEM / SIGNAL ANALYSIS
- CONTROL DESIGN
- SIMULATION
- SYSTEM IDENTIFICATION
- *RT CONTROL OPERATION*

INTEL | RMX

IEEE 488 BUS

PHONE LINE | **MAINFRAME**
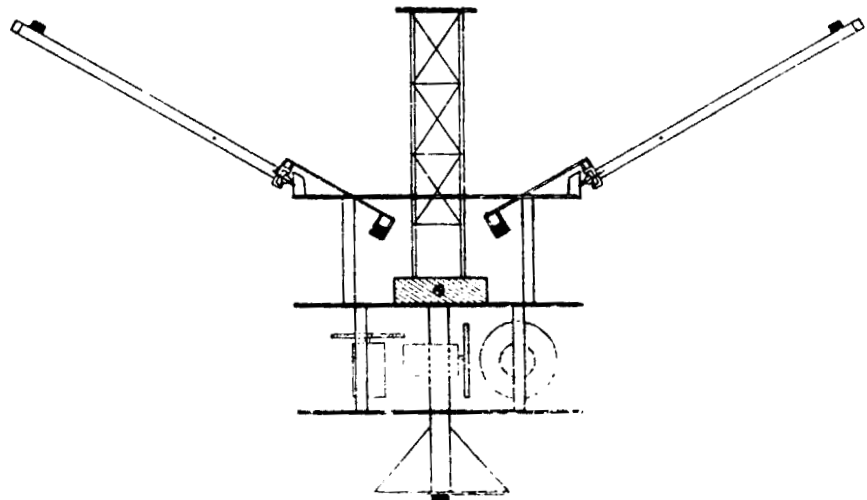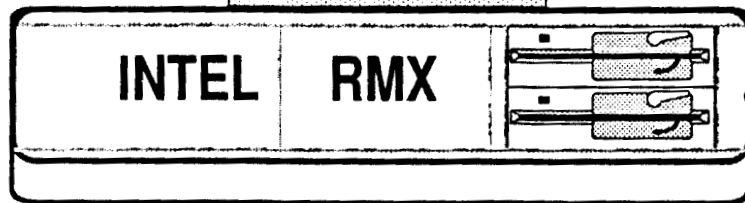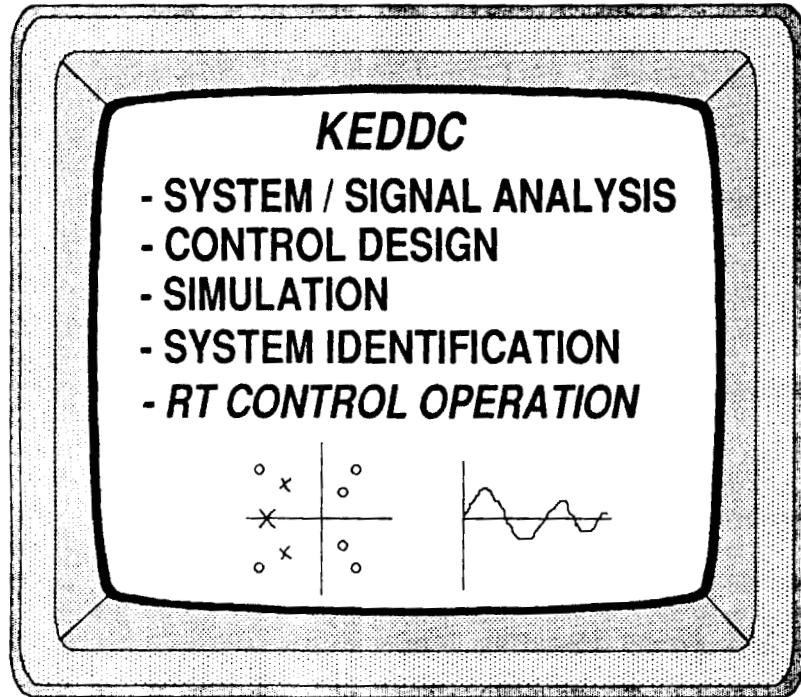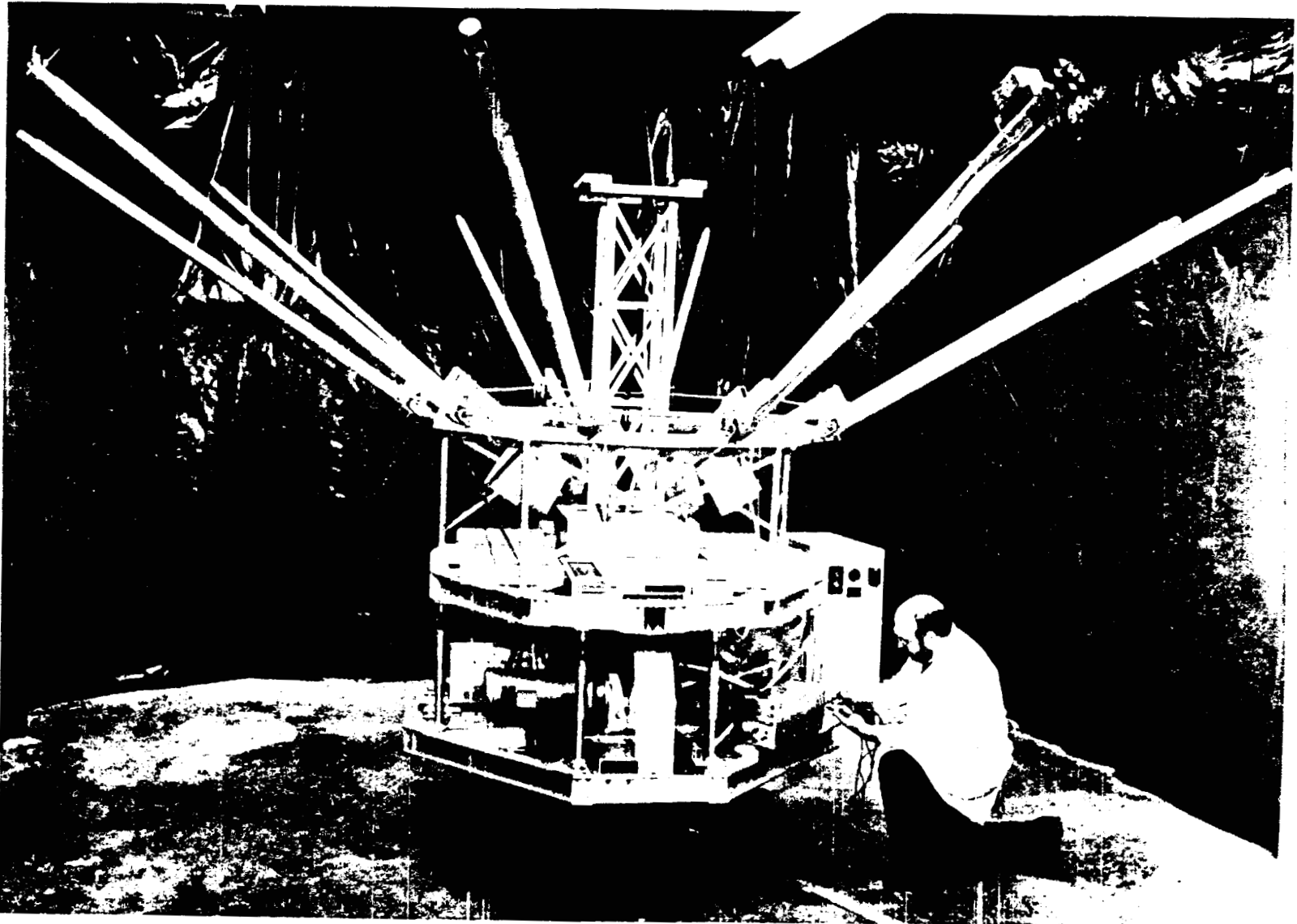
DAISY: A FLEXIBLE SPACECRAFT EMULATOR
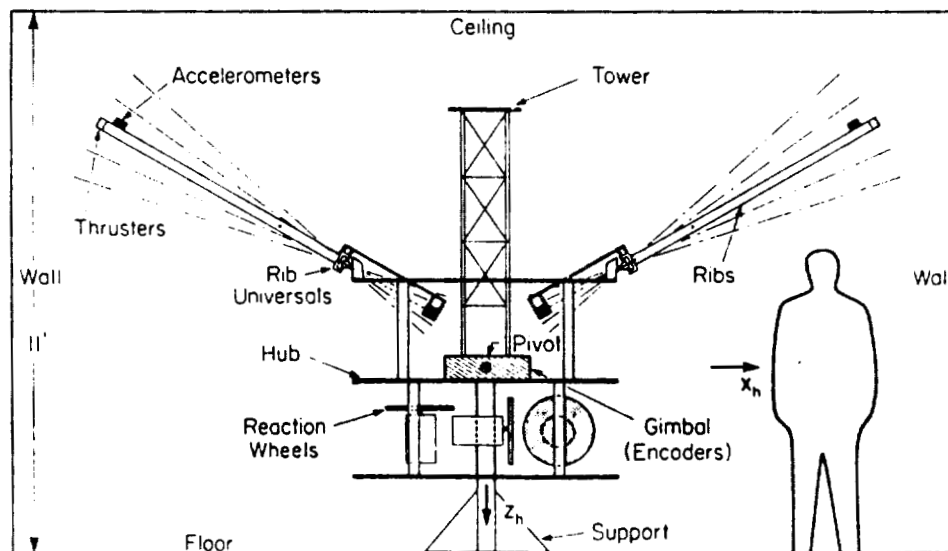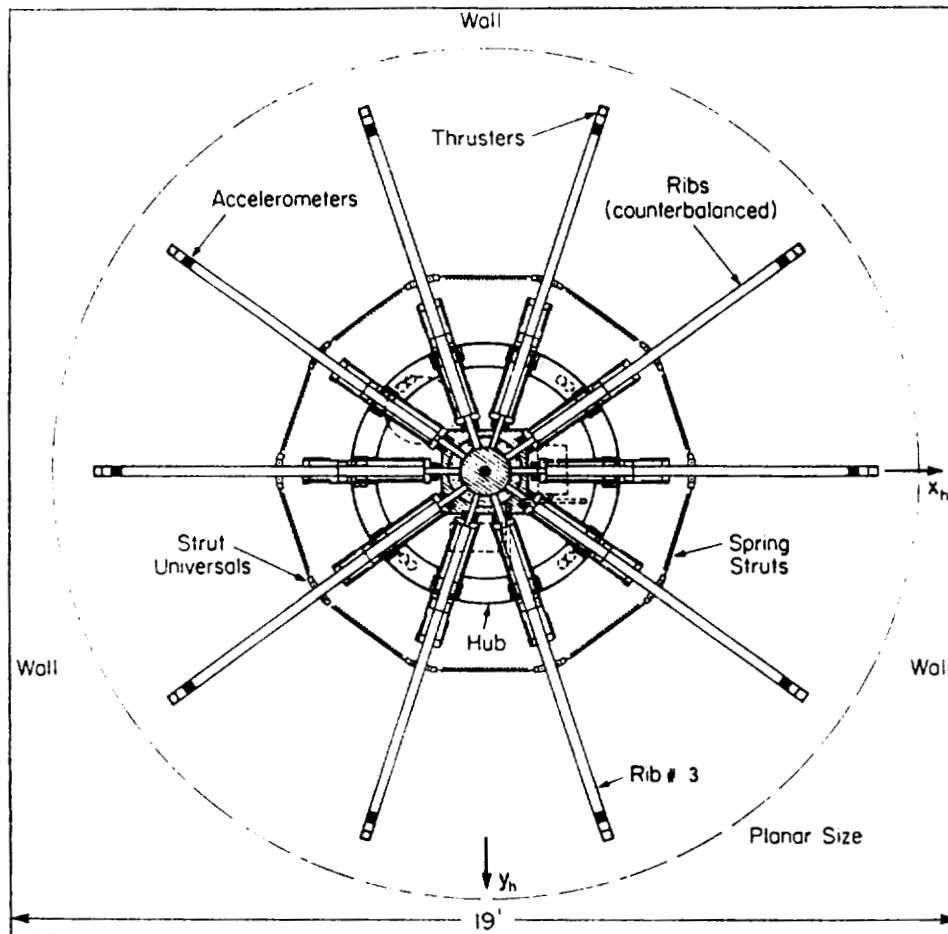
DAISY

EMULATES DYNAMICS OF A LARGE FLEXIBLE SPACE STRUCTURE

- 3 RIGID-BODY MODES
  (SLIGHT PENDULOSITY IN 2 RIGID-BODY MODES)

- 20 FLEXIBLE BODY MODES,
  LOW FREQUENCIES: 0.07 ... 0.11 Hz, IN CLUSTERS

- LOW DAMPING RATIO ACHIEVED
  RIBS: 0.008, HUB: 0.01 ... 0.05

SPACECRAFT - LIKE SENSORS AND ACTUATORS

- 3 REACTION WHEELS ON HUB

- THRUSTERS ON RIB(S)

- ENCODERS ON HUB GIMBAL

- ACCELEROMETERS ON RIB(S)

Principal Gains of DAISY
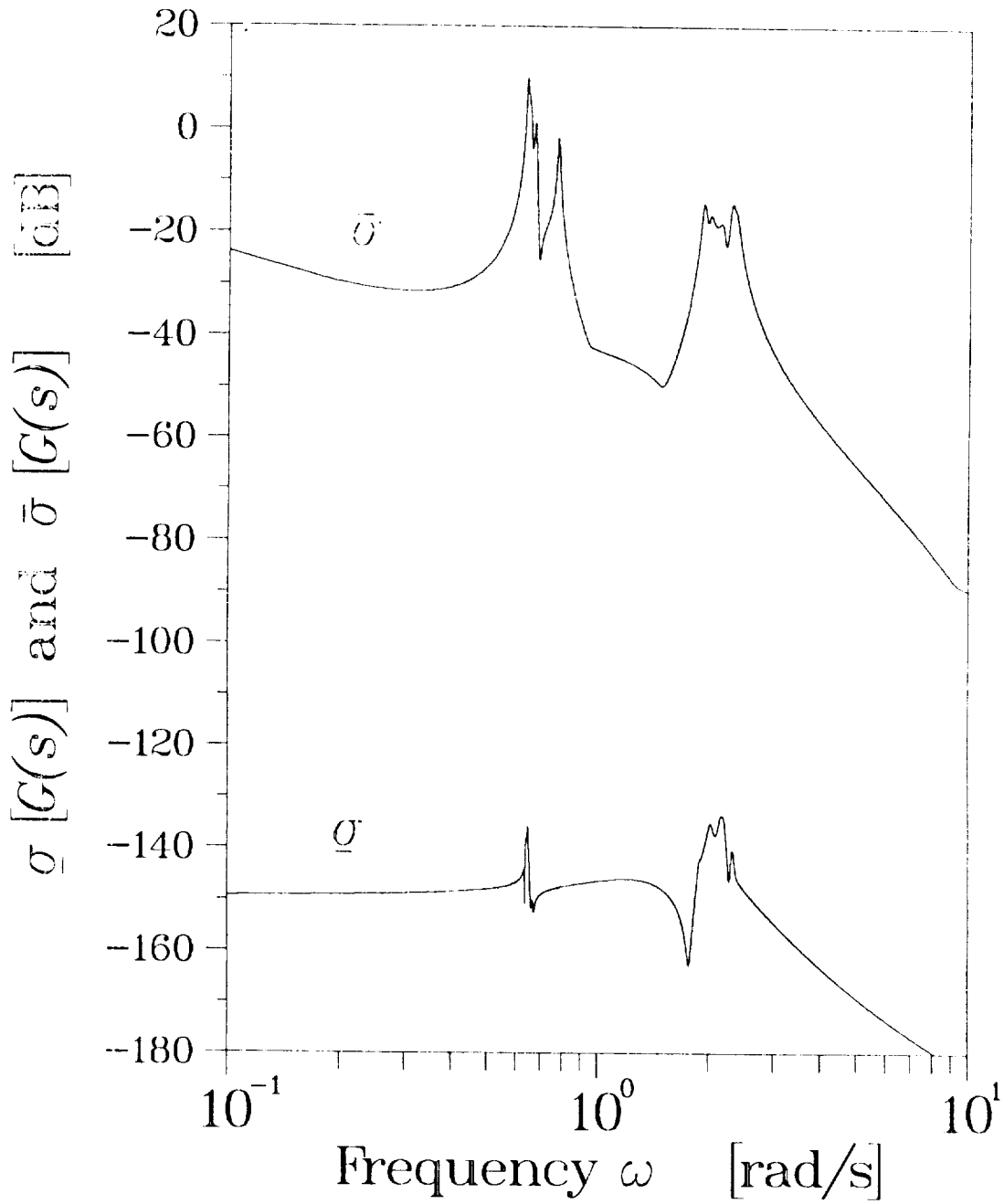
EXPERIMENTAL RESEARCH USING DAISY

PRESENT OBJECTIVE

DEVELOPMENT AND DEMONSTRATION OF

ROBUST CONTROL ALGORITHMS FOR FLEXIBLE STRUCTURES

STEPS (NOT NECESSARILY IN THIS ORDER)

- GIVEN: ANALYTICAL DYNAMICS MODEL

- SYSTEM-ORDER REDUCTION

- MODEL DISCRETIZATION

- SYNTHESIS OF CONTROL ALGORITHM

- SIMULATION

- EXPERIMENT

- EVALUATION OF ALGORITHM

TURNAROUND: 40 MIN

DESIGN EXAMPLE

SYSTEM EIGENVALUES AND TRANSMISSION ZEROS
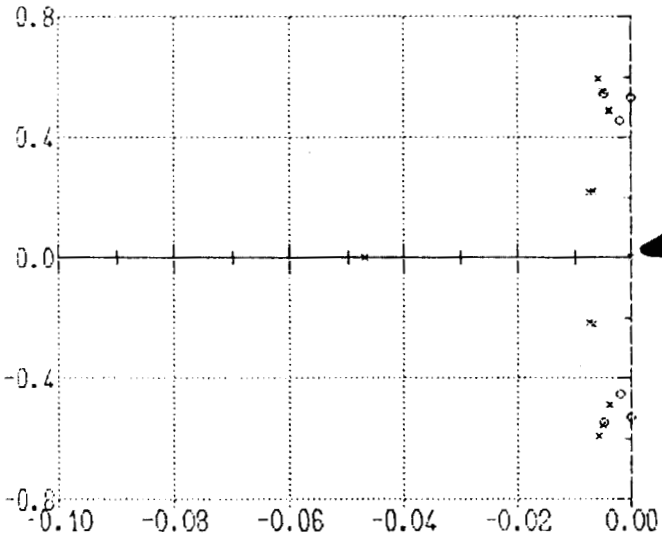
MODEL (SYSTEM MATRIX)

ANALYSIS
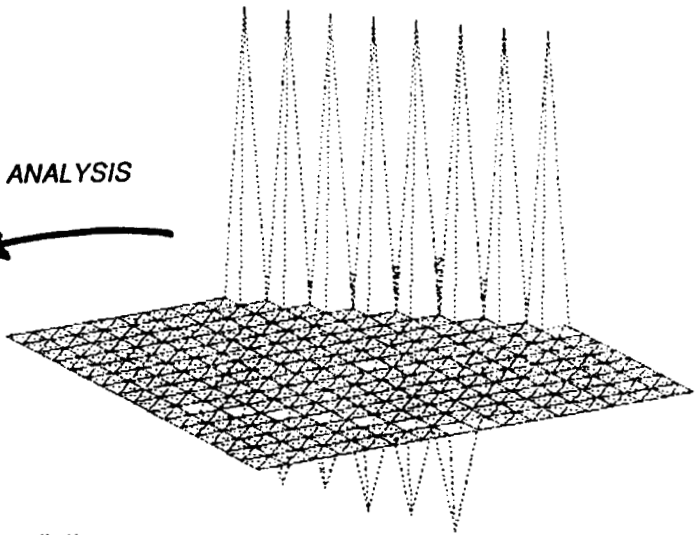
DESIGN

EIGENVALUES OF

CLOSED-LOOP SYSTEM
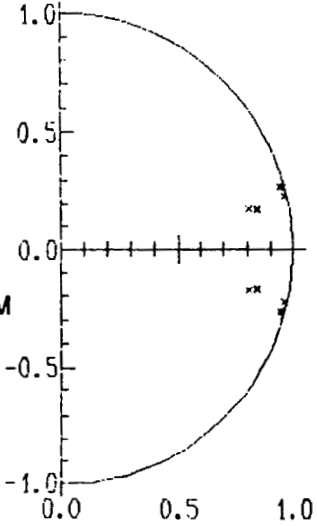
EXPERIMENT

TRACKING RESPONSE

98
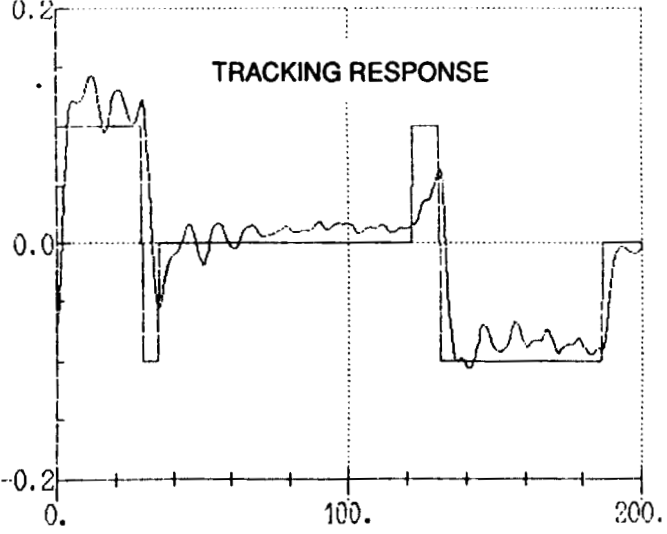
REAL-TIME CONTROL OPERATION

INTERACTIVE MONITOR

- INTERFACE BETWEEN USER AND REAL-TIME CONTROL ALGORITHM

- CONFIGURATION AND CONTROL OF REAL-TIME ALGORITHM

- DISPLAY AND RECORDING OF EXPERIMENTAL RESULTS
  SIGNALS: PLANT INPUT/OUTPUT, SETPOINTS, OBSERVER STATES, ...

- COMPLETE ENVIRONMENT FOR EFFICIENT EXPERIMENTATION

REAL-TIME CONTROL ALGORITHM

- EXECUTION TIME

  EXTREMES: 5 MILLISEC WITH 5TH-ORDER OBSERVER
  1.2 SEC WITH 50TH-ORDER OBSERVER, 10 INPUTS, 10 OUTPUTS

  TYPICAL FOR DAISY APPLICATION (20TH-ORDER, 5 INP, 5 OUTP): 20 MILLISEC

- HOST FAST ENOUGH FOR REAL-TIME CONTROL OF DAISY
  SAMPLING INTERVAL: 0.2 SEC ... 1 SEC

SUMMARY

EXPERIMENTAL RESEARCH ON CONTROL OF FLEXIBLE STRUCTURES
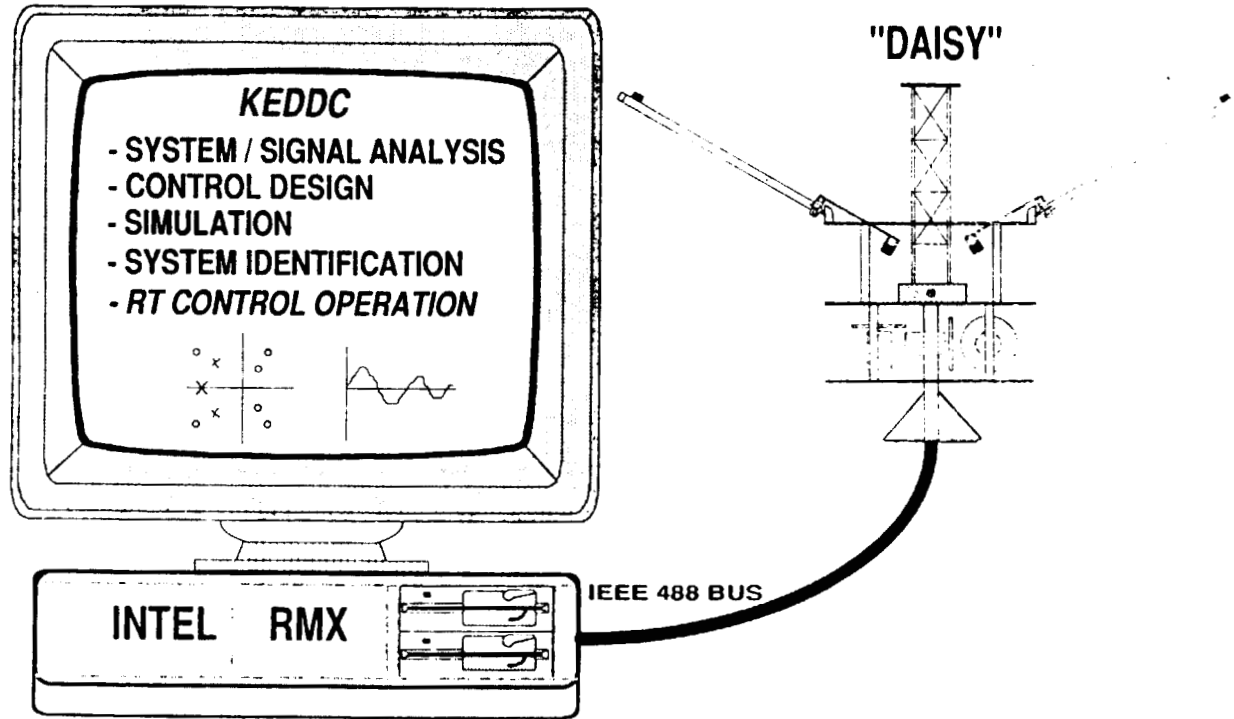
CONCLUSION

REAL-TIME WORKSTATION BRIDGES GAP BETWEEN THEORY AND EXPERIMENT!

CONSOLE: A CAD TANDEM FOR OPTIMIZATION-BASED DESIGN INTERACTING WITH USER-SUPPLIED SIMULATORS

By

Michael K. H. Fan, Li-Sheng Wang, Jan Koninckx and Andre L. Tits
University of Maryland
College Park, Maryland

ABSTRACT

The most challenging task when designing a complex engineering system is that of coming up with an appropriate system "structure." This task calls extensively upon the engineer's ingenuity, creativity, intuition and experience. After a structure has been (maybe temporarily) selected, it remains to determine the "best" value of a number of "design parameters." The engineer's input is still essential here, as multiple tradeoffs are bound to appear. However, except in the simplest cases, achieving anything close to optimal would be impossible without the support of numerical optimization. Providing such support while emphasizing tradeoff exploration through man-machine interaction is the purpose of interactive optimization-based design packages such as CONSOLE (Proceedings of American Control Conference 1988). A requirement for CONSOLE is that the parameters to be optimally adjusted vary over a continuous (as opposed to discrete) set of values.

CONSOLE employs a recently developed design methodology (International Journal of Control 43:1693-1721) which provides the designer with a congenial environment to express his problem as a multiple objective constrained optimization problem and allows him to refine his characterization of optimality when a suboptimal design is approached. To this end, in CONSOLE, the designer formulates the design problem using a high-level language and performs design task and explores tradeoff through a few short and clearly defined commands.

The range of problems that can be solved efficiently using a CAD tools depends very much on the ability of this tool to be interfaced with user-supplied simulators. For instance, when designing a control system one makes use of the characteristics of the plant, and therefore, a model of the plant under study has to be made available to the CAD tool. CONSOLE allows for an easy interfacing of almost any simulator the user has available.

To date CONSOLE has already been used successfully in many applications, including the design of controllers for a flexible arm and for a robotic manipulator and the solution of a parameter selection problem for a neural network (all under P. S. Krishnaprasad at the University of Maryland at College Park), the design of an RC controller for a radar antenna (under F. Emad at the University of Maryland at College Park), and the design of power filters (at the Westinghouse Defense and Electronics Center). In the case of the neural network application, CONSOLE was coupled to the nonlinear system simulator SIMNON.

CONSOLE :

A CAD Tandem for Optimization-Based Design Interacting with User-Supplied Simulators

Michael K.H. Fan

Li-Sheng Wang

Jan Koninckx

André L. Tits

Systems Research Center
University of Maryland, College Park

CONSOLE ━━━━━━━━━━━━━━━━━
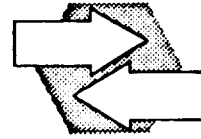
HISTORY

DELIGHT (Nye, Polak, Sangiovanni-Vincentelli, Tits) 1980 -
general purpose interactive package
+ optimization algorithms

DELIGHT.MaryLin (Fan, Nye, Tits) 1985 -
interactive optimization-based design package
for linear time-invariant systems

CONSOLE (Fan, Wang, Koninckx, Tits) 1987 -
interactive optimization-based design package
for engineering systems (with user-supplied
simulators)
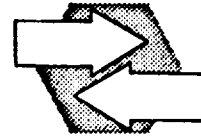
PARAMETRIC OPTIMIZATION IN DESIGN

Assume *structure* already chosen

Examples :

Circuit → Topology

Control System → Controller Structure

Earthquake Proof Building → Number and Position
of Beams

Remain to choose *best* value of finitely many parameters

Examples :

Circuit → R, C, W, A, ...

Control System → Controller Gains,
LQR/LQG Weighting Matrices,
Q-parameterization, ...

Earthquake Proof Building → Beam Thickness,
Amount of Steel, ...

COMPONENTS FOR PARAMETERIC OPTIMIZATION

1. Design Methodology (Nye, Tits)

Problem Formulation
Optimal in what Sense ?
Optimization Algorithm
User-Machine Interaction $\Big\}$ → **CONSOLE**

2. Model and Simulation Tool → Simulators

93
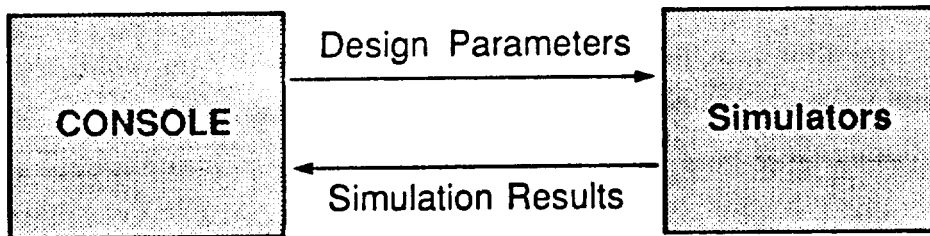
PROBLEM FORMULATION

Types of Specifications

Objectives - The smaller (larger) the better.

Soft Constraints - Aim for a *target* value. If unachievable,

the smaller (larger) the better.

Hard Constraints - Specified value *must* be achieved.

OPTIMAL IN WHAT SENSE ?

Degree of Satisfaction

D_G

D_B

f_1

f_2

G_1 G_2 B_2 B_1 f_1, f_2

Degree of Satisfaction

D_G

D_B

f_2'

f_1'

0 θ 1 f_1', f_2'

$$f'_i = \frac{f_i - G_i}{B_i - G_i} \qquad \min_x \ \max_i \ f'_i(x)$$

95
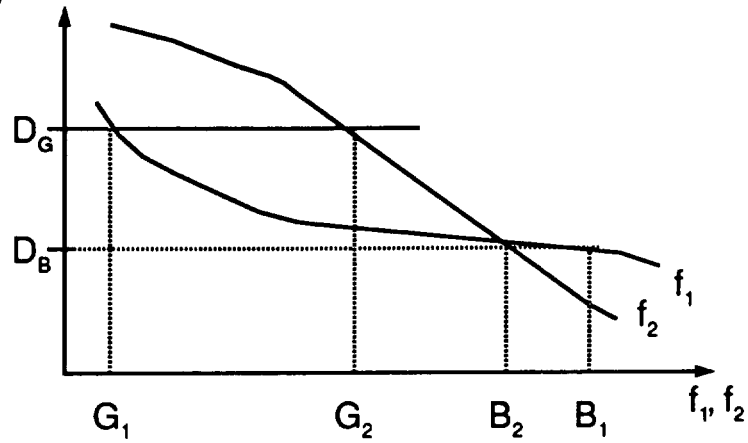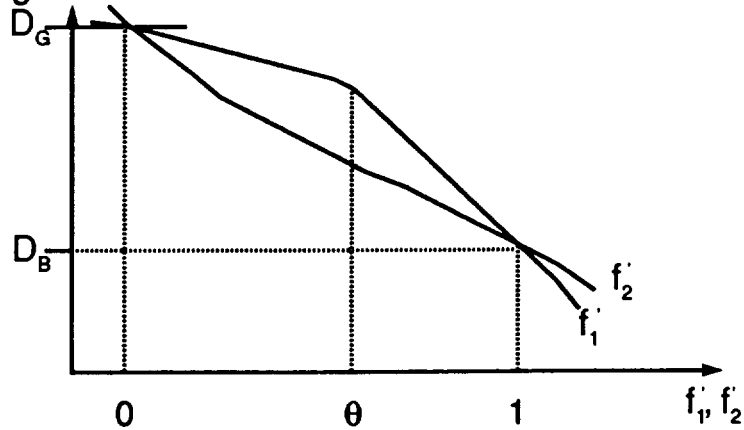
OPTIMIZATION ALGORITHM

Three Phase Feasible Direction Algorithm

Phase 1 (until all hard constraints are satisfied)

attempt to satisfy hard constraints (HC)

minimax on HC

Phase 2 (until all good values are achieved)

improve objectives (O) and soft constraints (SC)

minimax on O and SC

subject to satisfying HC

Phase 3

improve objectives

minimax on O

subject to satisfying HC and SC

$$\min_{x} \max_{i} f_i(x)$$

subject to

$$g_k(x) \leq 0, \quad \forall k$$

where

$$f_i(x) = \max_{\omega} \varphi_i(x,\omega)$$

$$g_k(x) = \max_{\omega} \psi_k(x,\omega)$$

97
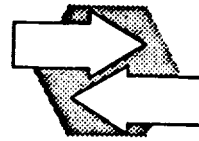
USER-MACHINE INTERACTION

Purpose

Progressively refine problem definition

Means

- Information on status of design conveyed graphically to user (Pcomb, Ecomb).

- User steers design to *his* optimal solution by adjusting good/bad values/curves.

C-2

CONSOLE =

CONvert + SOLvE

CONSOLE

| System Description File | | Problem Description File |

CONVERT

Two Binary Files

SIMULATOR ⟷ SOLVE

USER

OPTIMAL SOLUTION

100
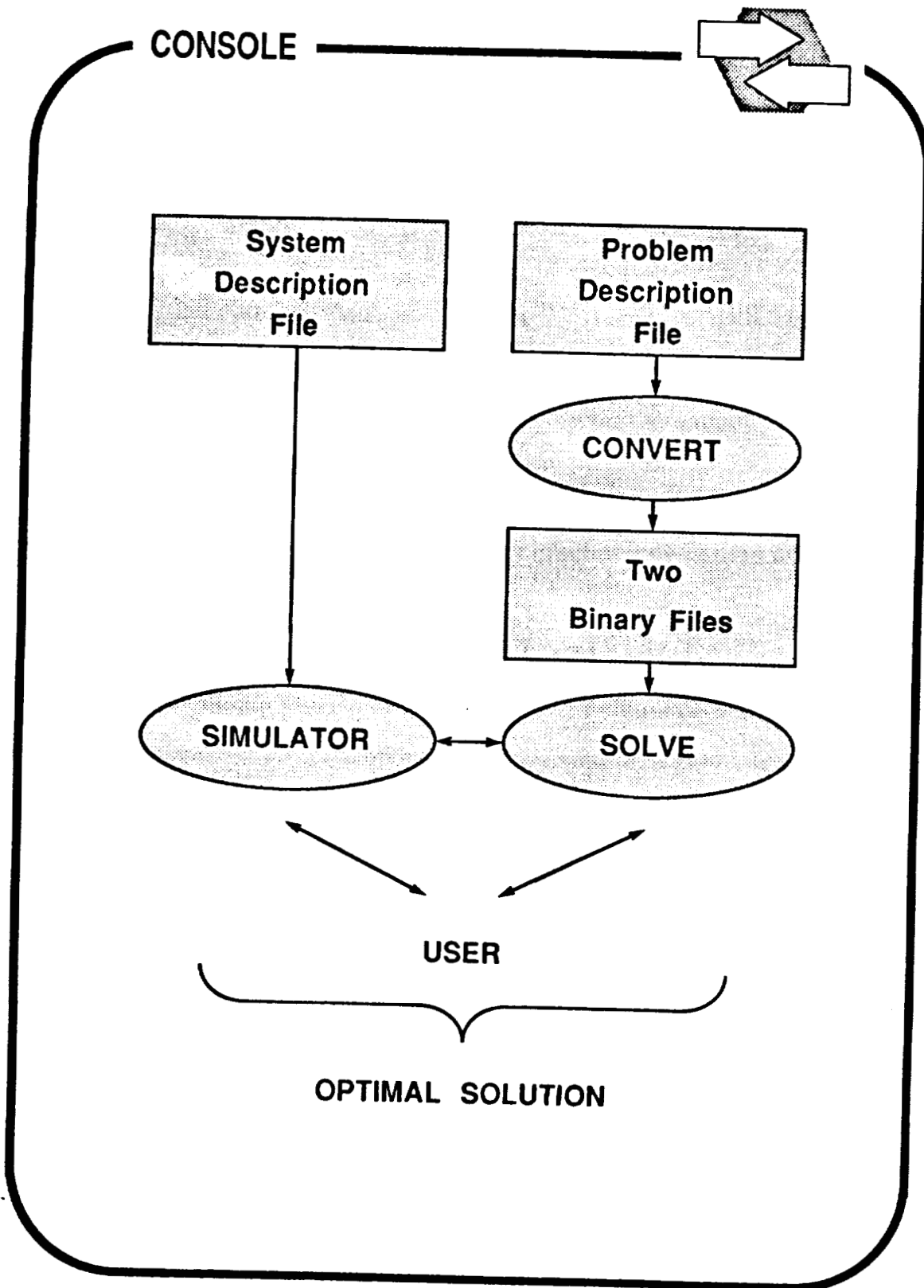
A SIMPLE DESIGN EXAMPLE

CONTROL SYSTEM
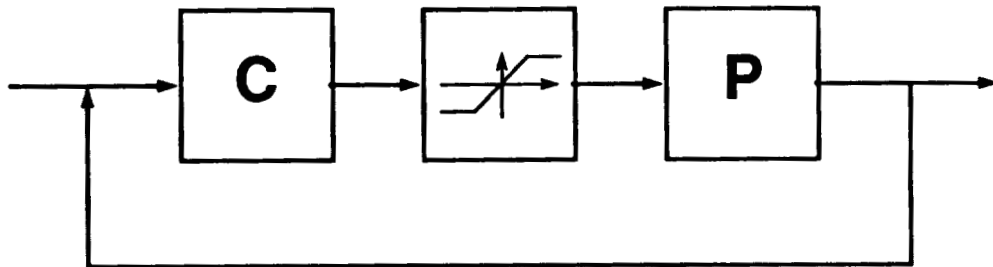
$$C(s) = K_p + \frac{K_I}{s} \qquad P(s) = \frac{s+1}{s^2}$$

DESIGN SPECIFICATION

TIME

SYSTEM DESCRIPTION FILE FOR THE EXAMPLE
(SIMNON*)

```
CONTINUOUS SYSTEM servo
STATE x1 x2 x3
DER dx1 dx2 dx3
x1:0
x2:0
x3:0
dx1 = x2
dx2 = if (e > 0.4) then 0.4
        else if (e < -0.4) then -0.4
        else e
dx3 = r - y
e = (r - y)*Kp + x3*Ki
y = x1+x2
r:1
Kp:0
Ki:0
END
```

*SIMNON was developed at the Lund Institute of Technology, Lund, Sweden
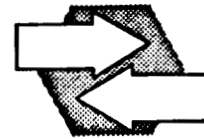
PROBLEM DESCRIPTION FILE FOR THE EXAMPLE

```
design_parameter Kp init=1 variation=5
design_parameter Ki

functional_objective "overshoot"
  for t from 0 to 20 by 0.1
  minimize {
    double simnon_time_response();
    return simnon_time_response(Kp,Ki,"y",t);
    }
  good_curve={
    if (t <= 4)  return 1.05;
    else         return 1.01;
    }
  bad_curve ={
    if (t <= 4)  return 1.1;
    else         return 1.02;
    }

functional_objective  "settling time"
  for t from 2 to 20 by .1
  maximize {

    ...
```

CONSOLE

MAIN FEATURES OF CONSOLE

Problem formulation is closely related to the character of a design problem.

Problem formulation syntax is strict, but easy to use.

Efficient iteration between **CONVERT** and user for debugging the PDF.

SOLVE is interactive, with short and clearly defined commands providing efficient communication between the program and the user.

Interactive graphics provide the user with easy-to-interpret information on the current design (Pcomb, Ecomb).

User-supplied simulators can easily be linked with **SOLVE** .

104

CONSOLE
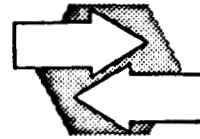
GLANCE AT APPLICATIONS

Design of a copolymerization reactor controller
(Butala, Choi, Fan)

Design of controllers for a flexible arm
(Wang, Krishnaprasad)

Design of a controller for a robotic manipulator
(Chen, Krishnaprasad)

H-infinity Design of a Sampled-Data Control Systems
(Yang, Levine)

Solution of a parameter selection problem
for a neural network
(Pati, Krishnaprasad *et al.*)

Design of an RC controller for a radar antenna
(Emad)

Design of power filters
(Glover, Walrath at Westinghouse Defense
and Electronics Center)

... and soon

Design of earthquake proof buildings
(Austin)

Design of controllers for X29 aircraft
(Reilly, Levine)

Design of circuits
(Westinghouse)

DESIGN OF A COPOLYMERIZATION REACTOR CONTROLLER

(**CONSOLE** + Copoly) (Butala, Choi, Fan)

Objectives and Constraints

> Molecular Weight
> Composition
> Final Volume
> Temperature
> Feed Flowrate

Manipulated Variables

> Temperature $= a_1 + a_2 t + a_3 t^2 + a_4 t^3$
>
> Feed Flowrate $= b_1 + b_2 t + b_3 t^2 + b_4 t^3$

Design Parameters $= a_i$'s and b_i's

Results

Pcomb (Iter= 22) (Phase 2) (MAX_COST_SOFT= 0.0766327)

SPECIFICATION	PRESENT	GOOD	G	B	BAD
FO1 (MN-MNs)^2	1.92e+06	0.00e+00 ==============*		l	2.50e+07
FO2 (CC-CCs)^2	3.88e-03	0.00e+00 ==============*		l	5.06e-02
C1 final vol	3.47e+00	4.00e+00 <--	l	l	4.10e+00
FC1 upper temp	3.53e+02	3.63e+02 <--	l	l	3.64e+02
FC2 lower temp	3.45e+02	3.28e+02 <----------------------------------		3.23e+02
FC3 upper flow	9.70e-03	7.00e-02 <--	l	l	7.50e-02
FC4 lower flow	6.00e-03	0.00e+00 <----------------------------------		-5.00e-03

DESIGN OF A DC DIRECT DRIVE MOTOR

(**CONSOLE** + Simnon) (Wang, Krishnaprasad)

Objective
  Position Profile

Design Parameters
  Feedback Gains

Results
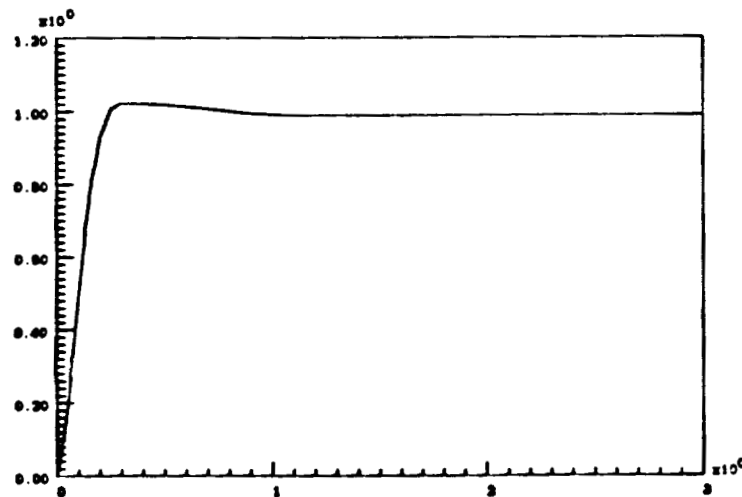
CONSOLE
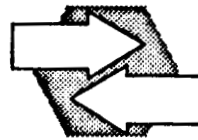
FUTURE ENHANCEMENTS

User Interface

More Powerful Optimization Algorithms

Gradient Computation

THE APPLICATION OF TSIM SOFTWARE TO ACT DESIGN AND ANALYSIS ON
FLEXIBLE AIRCRAFT

By

Ian W. Kaynes
Royal Aerospace Establishment
Farnborough, United Kingdom

ABSTRACT

The TSIM software is described. This is a package which uses an interactive FORTRAN-
like simulation language for the simulation on nonlinear dynamic systems and offers
facilities which include: mixed continuous and discrete time systems, time response
calculations, numerical optimization, automatic trimming of nonlinear aircraft
systems, and linearization of nonlinear equations for eigenvalues, frequency
responses and power spectral response evaluation.

Details are given of the application of TSIM to the analysis of aeroelastic systems under
the RAE Farborough extension FLEX-SIM. The aerodynamic and structural data for the
equations of motion of a flexible aircraft are prepared by a preprocessor program for
incorporation in TSIM simulations. Within the simulation the flexible aircraft model
may then be selected interactively for different flight conditions and modal reduction
techniques applied. The use of FLEX-SIM is demonstrated by an example of the flutter
prediction for a simple aeroelastic model.

By utilizing the numerical optimization facility of TSIM it is possible to undertake
identification of required parameters in the TSIM model within the simulation. The
optimizer is applied to the minimization of error between predicted and measured time
responses of the system; while possibly not so efficient as dedicated identification
software this has the great advantages that the identification is made directly
involving the simulation model without further reprogramming or data transfer and it
may be applied directly to nonlinear models. Examples are given of this analysis
applied to aircraft measured responses and to simulated responses of a controlled
aircraft with nonlinearities.

THE APPLICATION OF TSIM SOFTWARE TO ACT DESIGN
AND ANALYSIS ON FLEXIBLE AIRCRAFT

by

IAN KAYNES

ROYAL AEROSPACE ESTABLISHMENT
Farnborough, England

Head, Theoretical Dynamics Section,
Structural Dynamics Division,
Materials and Structures Department

PROGRAMME OBJECTIVES

1. Improvement of aeroelastic modelling techniques

2. ACT Design methods for structural applications

3. Assessment of structural impact of ACT

2.   RAE FLEX-SIM

110

RAE EXPERIMENTAL PROGRAMMES

1. Flight data from flexible aircraft
   (VC10, Tornado)

2. Wind tunnel experiments
   (GARTEUR, 'flying model', spoiler tests)

3.   RAE FLEX-SIM

AEROELASTIC MODELLING INPUT

a) STRUCTURAL MODAL DATA
   Calculated from mass and stiffness data by
   finite element or beam models AND/OR
   derived from ground resonance tests.
   Model reduction techniques used as appropriate.

b) AERODYNAMIC LOADINGS
   Calculated from geometric data by vortex lattice
   or RAE methods for steady and unsteady flow.

c) SENSOR and ACTUATOR DATA.
   Linearity assumed in these models.

4.   RAE FLEX-SIM

AEROSERVOELASTIC MODEL

Combination of structural, aerodynamic, sensor and actuator data with the control system model.

Expressed in a first order form compatible with stability and control representations to allow integration between the aeroelastician and the S&C specialists.

Software required for response prediction and control design activities on these models.

5.	RAE FLEX-SIM

TSIM

Time SIMulation

Non-linear dynamic simulation package

Originated and developed at RAE since late 1970s

Now documented, supported and developed as a commercial product by Cambridge Control

Used in RAE and in research organisations, aerospace industry and universities in Britain and overseas

6.	RAE FLEX-SIM

TSIM FACILITIES

Interactive program using FORTRAN-like simulation language and facilitating modification of model

Simulation of linear and non-linear equations
Mixed continuous and discrete time systems
Time response calculation
Linearisation of non-linear equations for:
>   Eigen values
>   Frequency responses
>   RMS response evaluation
Numerical optimisation
Automatic trimming of non-linear aircraft
Communication with other control design packages

| 7. | RAE FLEX-SIM |

SAMPLE OF TSIM SERIAL INTERACTION

```
SIM>
SIM>; Assign values to some TSIM variables:-
SIM>   ZPOSA 0.9  DAMPA 0.7  RTB 15
SIM>
SIM>; Enter the time response set-up module and
SIM>; define the required parameters:-
SIM>   SET TIME_RESP
SIM>
SET TIME_RESP: OUTPUT 1 NZB 2 BMR 3 TWG
SET TIME_RESP: SCALE 2 -0.8 0.8
SET TIME_RESP: RKUTTA 0.4, 0.002, 0.01
SET TIME_RESP: STEP BGO 0.0, -0.1, -0.6
SET TIME_RESP:
SIM>; Now run the time response module:-
SIM>   RUN TIME_RESP
```

| 8. | RAE FLEX-SIM |

113

FLEX-SIM: APPLICATION OF TSIM TO FLEXIBLE AIRCRAFT

PRE-PROCESSING FUNCTIONS:

    a)   structural data processing

    b)   aerodynamics calculations and modification

    c)   loads, actuator and sensor modelling

    d)   model reduction and combination

    e)   TSIM model generation

TSIM-CONCURRENT FUNCTIONS:

    f)   generation of aeroelastic input functions

    g)   order reduction and changes of flight
        conditions in the flexible aircraft model

    h)   flight loads and sensor response calculation

    i)   presentation of results

POST-PROCESSING FUNCTION:

    j)   analysis of aeroservoelastic results

9.	RAE FLEX-SIM

DEMONSTRATION LOAD ALLEVIATION - AIRCRAFT
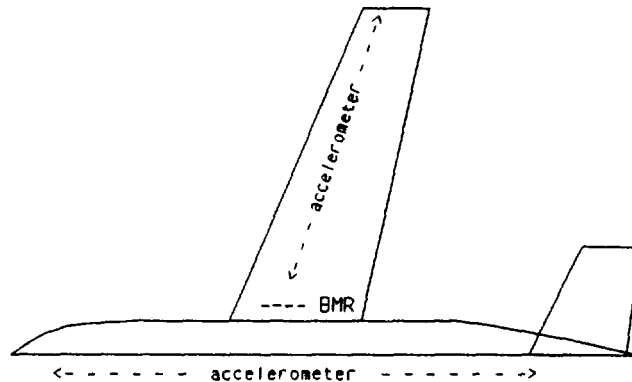
OBJECTIVE: reduction of wing loads in turbulence
through outboard wing controls

INVESTIGATION: sensor location and combination

10.	RAE FLEX-SIM

DEMONSTRATION LOAD ALLEVIATION – SYSTEM

Gust input

Actuator → Control demands → FLEXIBLE AIRCRAFT DYNAMICS → accelerations

First order filter

| 11. | RAE FLEX-SIM |

BASIC AIRCRAFT TIME RESPONSES
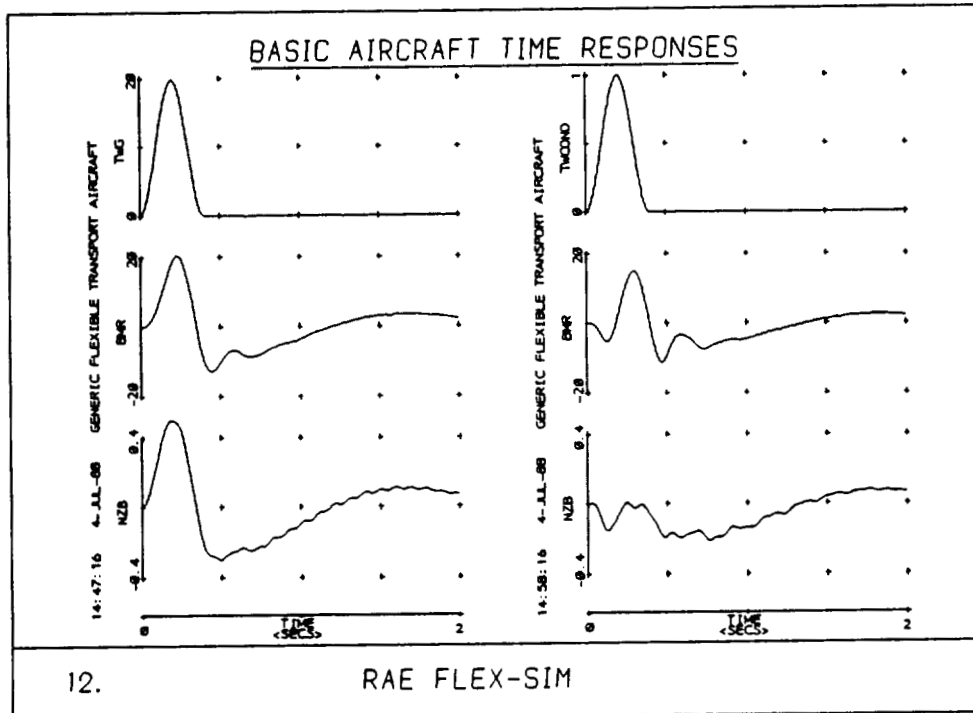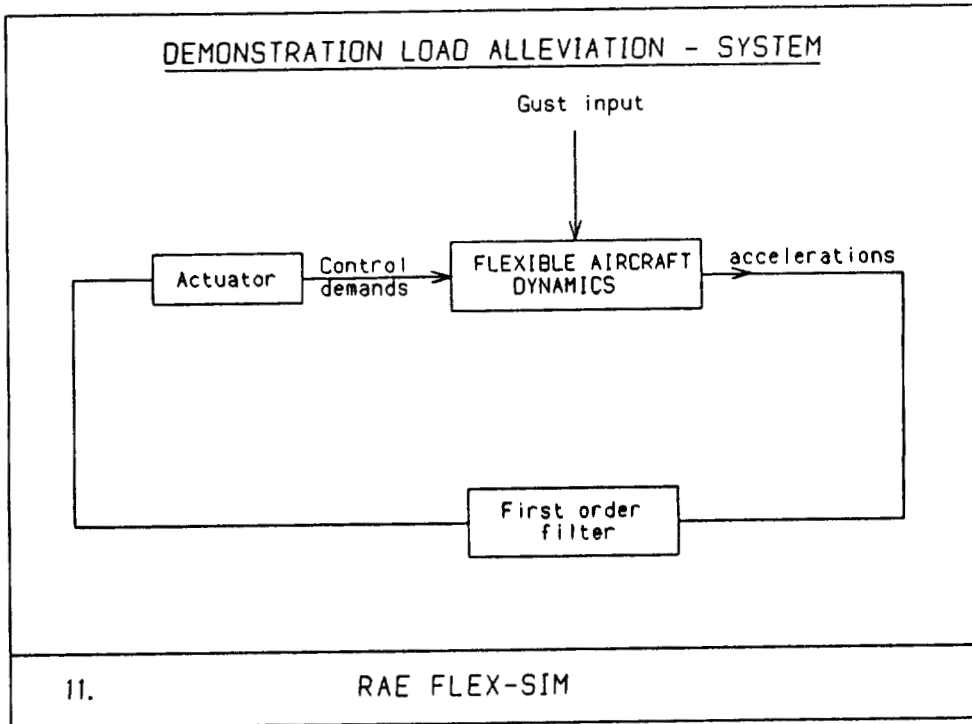
| 12. | RAE FLEX-SIM |

BASIC AIRCRAFT FREQUENCY RESPONSES
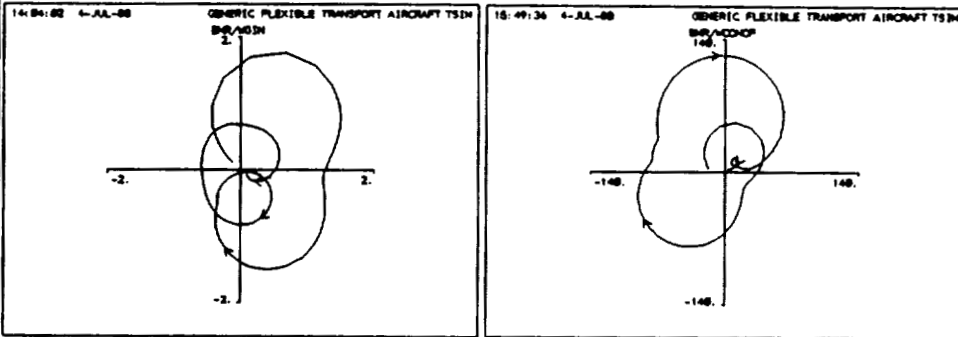
13.	RAE FLEX-SIM

GLA WITH ACCELEROMETER AT CG

Effect of variation of gain on gust responses

16:24:37   4-JUL-88   GENERIC FLEXIBLE TRANSPORT AIRCRAFT
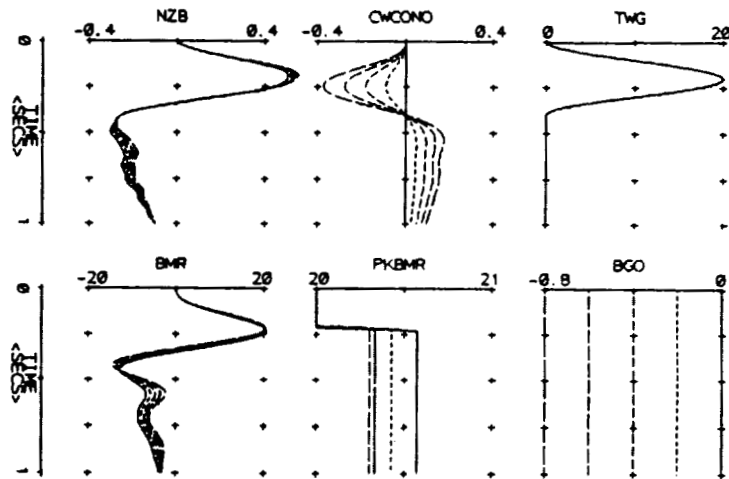
14.	RAE FLEX-SIM

GLA WITH ACCELEROMETER AT CG

Effect of variation of gain on PSD gust responses

11:05:33   6-JUL-88   GENERIC FLEXIBLE TRANSPORT AIRCRAFT
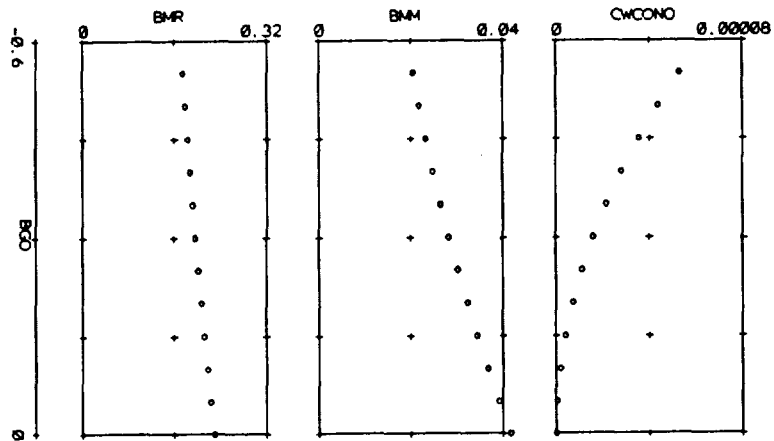
| 15. | RAE FLEX-SIM |

GLA WITH ACCELEROMETER IN FUSELAGE

Variation of eigen values with fuselage location
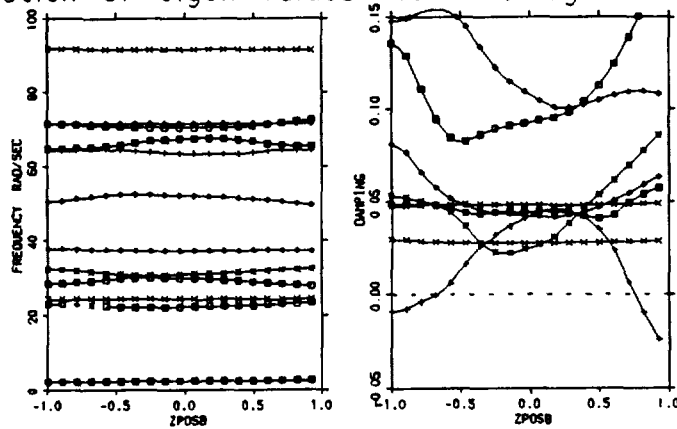
1-JUL-88 18:30:31 GENERIC FLEXIBLE TRANSPORT AIRCRAFT

| 16. | RAE FLEX-SIM |

117

GLA WITH ACCELEROMETERS ON WING AND AT CG

Variation of eigen values with spanwise position
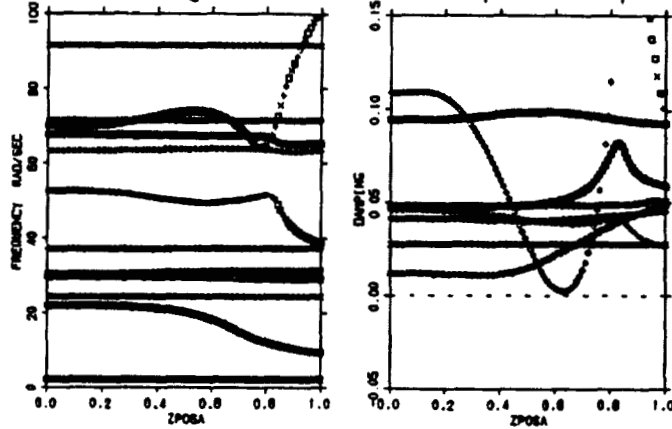
5-JUL-88 15:58:51 GENERIC FLEXIBLE TRANSPORT AIRCRAFT

17.   RAE FLEX-SIM

GLA WITH ACCELEROMETERS ON WING AND AT CG

Root locus with spanwise position

11:34:34   5-JUL-88   GENERIC FLEXIBLE TRANSPORT AIRCRAFT

18.   RAE FLEX-SIM

118

GLA WITH ACCELEROMETERS ON WING AND AT CG

Effect of wing tip accel gain variations, time

10:06:43   7-JUL-88   GENERIC FLEXIBLE TRANSPORT AIRCRAFT

19.   RAE FLEX-SIM

GLA WITH ACCELEROMETERS ON WING AND AT CG

Effect of wing tip accel gain variations, PSD

15:08:56   6-JUL-88   GENERIC FLEXIBLE TRANSPORT AIRCRAFT

20.   RAE FLEX-SIM

119

120
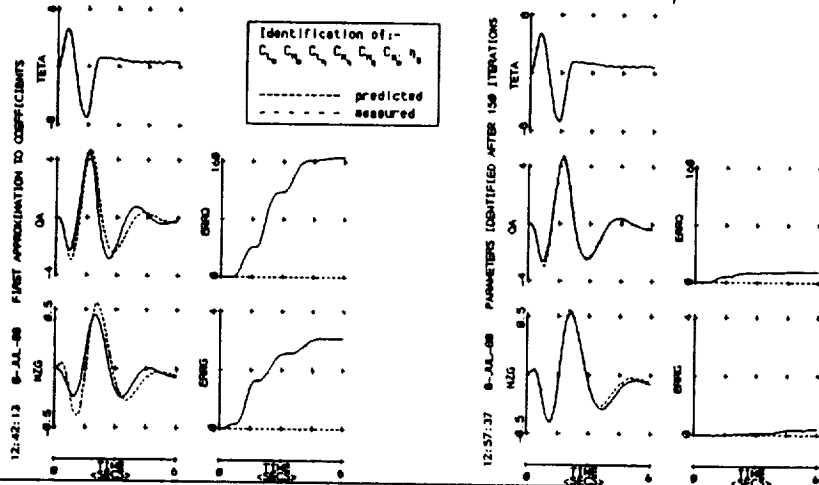
CONTROL/STRUCTURE INTERACTION METHODS FOR SPACE STATION POWER SYSTEMS

By

Paul Blelloch
Structural Dynamics Research Corporation
San Diego, California

ABSTRACT

The Structural Dynamics Research Corporation and the NASA Lewis Research Center have been working together to develop tools and methods for the analysis of control/structure interaction problems related to the space station power systems. Flexible modes of the solar arrays below 0.1 Hz, suggest that even for relatively slow control systems, the potential for control/structure interaction exists. The emphasis of the effort has been to develop tools which couple NASTRAN's powerful capabilities in structural dynamics with EASY5's powerful capabilities in control systems analysis. One product is an interface software package called CO-ST-IN for COntrol-STructure-INteraction. CO-ST-IN acts to translate data between NASTRAN and EASY5, facilitating the analysis of complex coupled problems. Interfaces to SDRC I-DEAS and MATRIXx are also offered. Beside transferring standard modal information, CO-ST-IN implements a number of advanced methods. These include a modal ordering algorithm that helps eliminate uncontrollable or unobservable modes from the analysis, an implementation of the more accurate mode acceleration algorithm for recovery of element forces and stresses directly in EASY5 and an implementation of fixed interface modes in NASTRAN, which reduces the error in the closed-loop model due to the use of truncated mode sets. A brief overview of the program will be presented, along with description of some of the methods used to facilitate rapid and accurate analyses.

CONTROL/STRUCTURE INTERACTION METHODS FOR SPACE STATION POWER SYSTEMS

presented by

Paul Blelloch, Ph.D.
SDRC WRO
San Diego, CA

supported by

NASA Lewis Research Center
Cleveland, OH

July 11, 1988

AGENDA

- **Quick Overview of CO-ST-IN Program**

- **Alternate Modal Representations**

- **Discussion**

SDRC

NOTES

SDRC has been working with the NASA Lewis Research Center to develop methods for the study of control/structure interaction problems related to space station power systems. We will discuss the software developed for this project, (CO-ST-IN) and if we have time we will briefly mention the important area of alternate modal representations to improve the accuracy of closed-loop models.

DATA NEEDS TO BE TRANSFERRED FROM STRUCTURES TO CONTROLS

Structural Dynamics

I-DEAS

NASTRAN

- Mode Shapes
- Open-loop flexible response and data recovery (linear)

Control Systems

MATRIXx

EASY5

- Classical Control
- Closed-loop rigid body response (non-linear)

SDRC
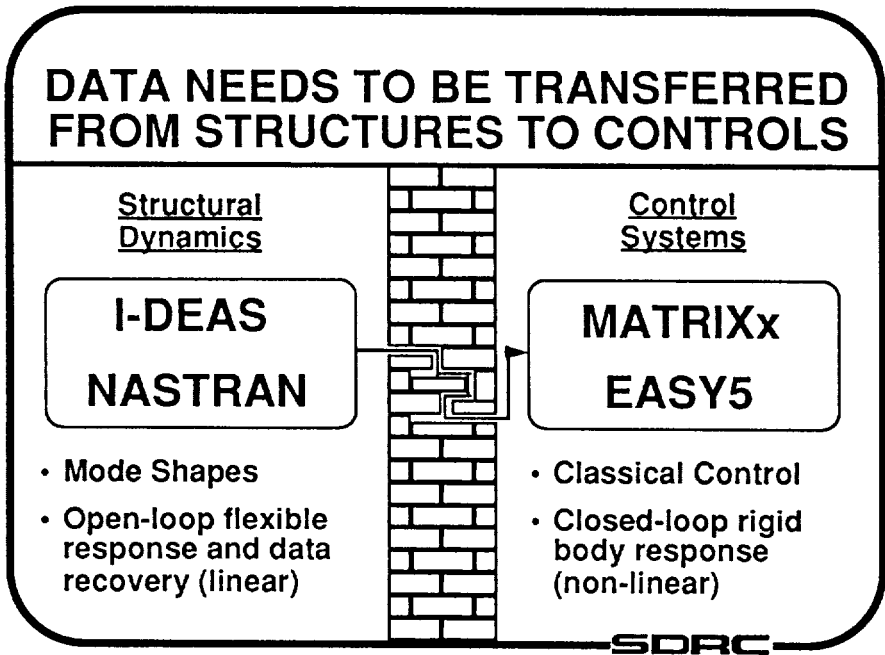
NOTES

Standard approaches to Control/Structure Interaction problems combine two separate disciplines, structural dynamics and control systems. Data is often passed manually from engineers in one group to engineers in the other. Furthermore, each group uses its own analysis tools. We use I-DEAS and NASTRAN for structural dynamics and MATRIXx and EASY5 for control systems.

124

SPACE STATION MODEL TOO LARGE FOR MANUAL TRANSFER OF DATA

Reaction Control System

126 Modes Below 1 Hz

Alpha Joints

Beta Joints

Attitude Control

13 Control Inputs and Outputs

1,890 numbers to be transferred

SDRC
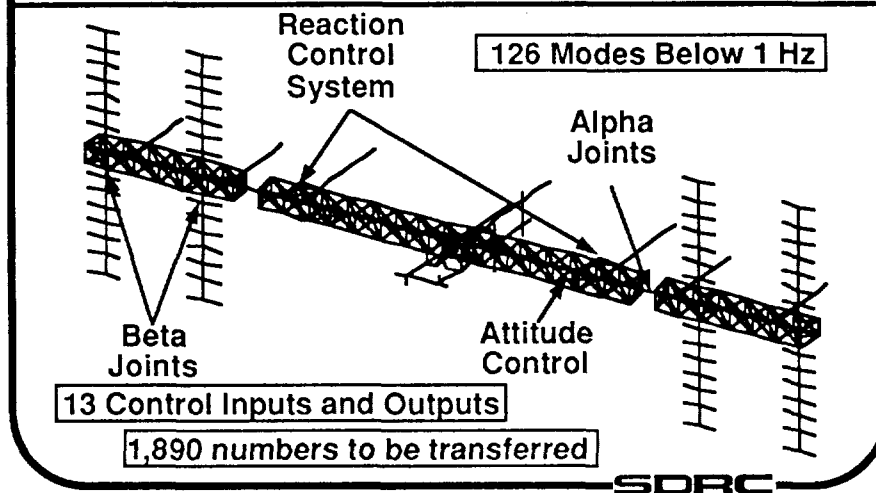
NOTES

The space station is a large complex structural system with a large number of closely spaced, low frequency modes and a large number of structural inputs and outputs. The size of the model makes manual transfer of data impractical. Model size also puts a large emphasis on practical model reduction algorithms.

CO-ST-IN TRANSFERS DATA

Structural Dynamics		Control Systems
I-DEAS **NASTRAN**	◄► **CO-ST-IN** ◄►	**MATRIXx** **EASY5**

- Closed-loop flexible response
- Closed-loop data recovery (non-linear)
- Stability analysis

SDRC
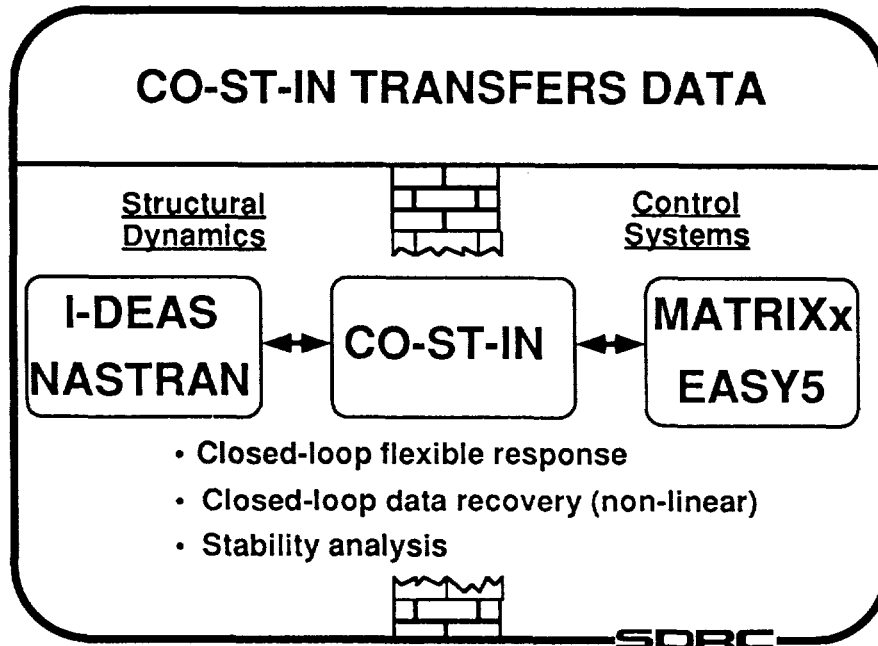
NOTES

CO-ST-IN stands for COntrol-STructure-INteraction. It automates the transfer of data back and forth among I-DEAS, NASTRAN, MATRIXx and EASY5. CO-ST-IN implements a number of special (non-standard) capabilities as well as the automated transfer of modal data.

MODAL ORDERING REDUCES MODEL SIZE

```
I-DEAS          Mode           MATRIXx
                Selection
NASTRAN                        EASY5
```

All modes in a frequency range	Select modes which interact strongly with inputs and outputs	Limited to smaller models
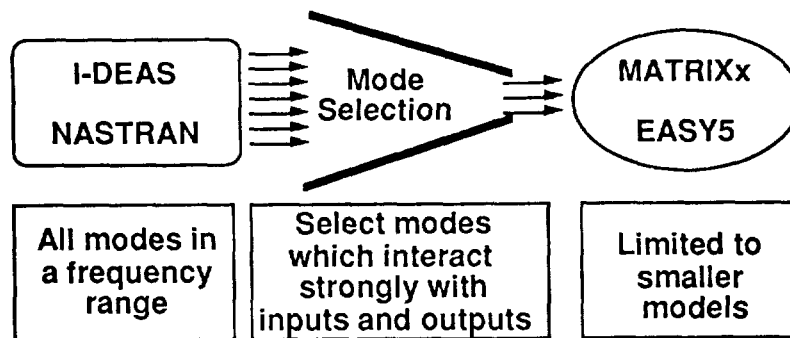
SDRC

NOTES

Since many control system algorithms fail for large models it is essential to select as small a model as possible. Large structural models usually contain a number of modes which do not interact significantly with inputs and outputs. Modal ordering can help eliminate these modes, resulting in an accurate reduced order model. We implement both Skelton's modal cost and the approximate balanced singular value as measures of modal influence. Inputs and outputs can be scaled to reflect their relative importance, and modes can be grouped when modal frequencies are close.

127

ELEMENT FORCES CAN BE CALCULATED IN EASY5 OR MATRIXx

NASTRAN Normal Modes and Static Analyses → Extract Data Recovery Matrices → EASY5 or MATRIXx Transient Analysis → Element Force Time Histories

- Forces available without returning to NASTRAN
- Mode acceleration and mode displacement options
- Applicable to preliminary studies
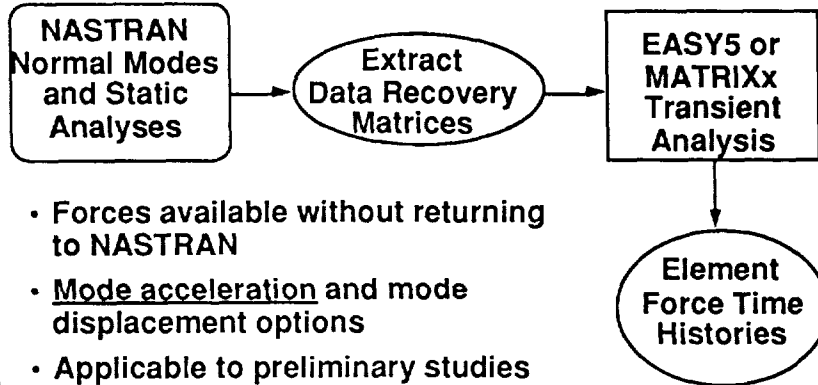
SDRC

NOTES

Calculating element forces and stresses directly in the control system routine can greatly accelerate turn around time. We transfer the appropriate matrices from NASTRAN to let us implement a mode acceleration technique. The mode acceleration formulation adds a static correction term to the standard mode displacement formulation which improves accuracy when using truncated mode sets. This approach is applicable to parameter studies, where quick turn around time is paramount.

128

ELEMENT FORCES CAN BE CALCULATED IN NASTRAN

```
┌─────────────┐      ╭─────────╮       ┌─────────────┐
│ EASY5 or    │      │ Extract │       │ NASTRAN     │
│ MATRIXx     │ ───▶ │Structural│ ───▶ │ Transient   │
│ Transient   │      │ Inputs  │       │ Analyses    │
│ Analysis    │      ╰─────────╯       └─────────────┘
└─────────────┘                               │
                                              ▼
 • Any NASTRAN solution can be used      ╭─────────╮
                                         │ Element │
 • Larger models are feasible            │Force Time│
                                         │ Histories│
 • Applicable to detailed stress analysis╰─────────╯
```
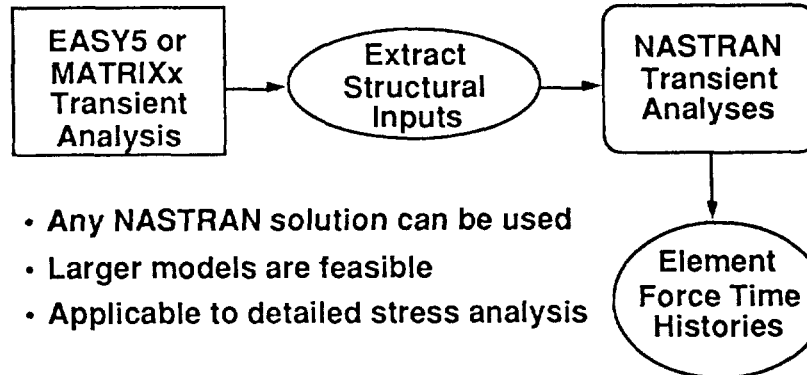
SDRC

NOTES

Detailed stress analyses fall into the realm of structural dynamicists. In order to facilitate the direct application of NASTRAN to this problem we extract the structural input forces from the controls routine and write these as NASTRAN bulk data. This allows the structural dynamicist the flexibility to choose any NASTRAN transient technique (including a direct transient) to recover element forces and stresses. This method increases turn around time, but is applicable to a detailed stress analysis after control system parameters have been fixed.

129

CO-ST-IN TRANSLATES
EASY5 AND MATRIXx OUTPUT

EASY5 or MATRIXx Output → CO-ST-IN → NASTRAN Bulk Data

CO-ST-IN → Universal Files
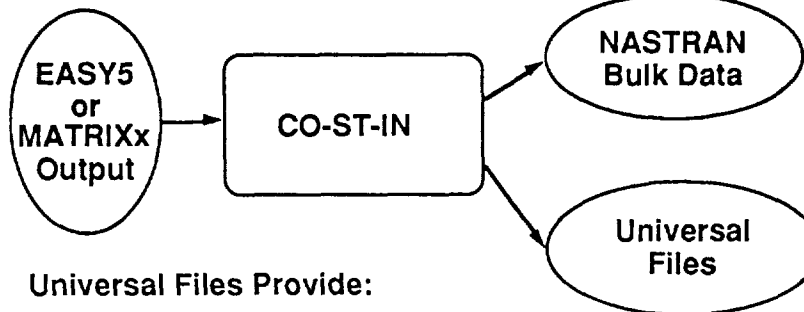
Universal Files Provide:

• Flexible plotting options

• Data management (SYSTAN)

SDRC

NOTES

As well as provide input to NASTRAN, other reasons for translating control system output include the requirement for more flexible plotting and data management. By translating time simulation output to I-DEAS Universal file format, we can store functions in a database, facilitating the application of powerful data management and plotting capabilities.

130

CO-ST-IN IS A TOOL FOR CONTROL/STRUCTURE INTERACTION

- Transfers data between structural dynamic (NASTRAN and I-DEAS) and control system software (EASY5 and MATRIXx).

- Uses modal ordering to reduce model size.

- Data recovery performed directly in controls routine using the more accurate mode acceleration method.

- EASY5 and MATRIXx output translated for plotting, data management and NASTRAN data recovery.

SDRC

NOTES

CO-ST-IN is simply a tool for transferring data among otherwise incompatible analysis programs. CO-ST-IN tries to be smart in what it transfers by using modal ordering to reduce model size and a mode acceleration technique to recover element forces and stresses directly in the controls routine. Control routine output is translated to I-DEAS Universal file format in order to facilitate data management and plotting.

AGENDA

- **Quick Overview of CO-ST-IN Program**

- **Alternate Modal Representations**

- **Discussion**

SDRC

The main focus of our investigation has been to determine the best possible structural models to use in control/structure interaction study. Here we define best to mean those which result in the most accurate closed-loop models while using a minimum number of dynamic states. This investigation has led us to examine the use of alternate (other than normal) modal representations.

132

FIXED INTERFACE MODES REPRE-SENT EFFECT OF INPUT FORCES

- Normal modes are calculated on basis of free-free boundary conditions
- Normal modes do not represent effect of input forces
- Fixed interface (cantilever, Craig-Bampton) modes result in more accurate closed-loop models
- Improvement even for "soft" controllers
- More pronounced for "stiffer" controllers

SDRC

NOTES

The basic problem with normal modes is that they are calculated on the basis of free-free boundary conditions. The result is that these modes poorly represent the local effects of forces and moments applied by control actuators at these boundaries. The use of fixed interface modes (sometimes called cantilever or Craig-Bampton modes) can help alleviate this problem by providing an accurate static representation at the location of control inputs. We have found that the use of fixed interface modes does result in more accurate closed-loop models, even for control frequencies which lie well below flexible frequencies. For stiffer controllers the differences are even more pronounced.

CLOSED-LOOP FREQUENCIES MORE ACCURATE

Normalized Frequency Error vs. Number of Flexible Modes, showing Fixed Interface Modes and Normal Modes.
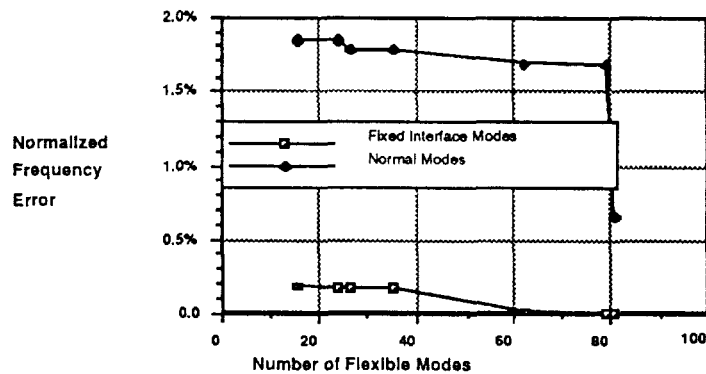
SDRC

NOTES

One measure of accuracy for the closed-loop model is the accuracy of closed-loop frequencies. Normalized error is defined as the distance of the approximate frequency from the exact frequency, divided by the magnitude of the exact frequency. In this case we are examining the accuracy of an alpha joint control frequency as we increase the number of open-loop modes, using either a fixed interface or a normal modes representation. In this case control frequencies are close to an order of magnitude below flexible frequencies and the results are consistent with other models that we've looked at.

134

CLOSED-LOOP FREQUENCY RESPONSE MORE ACCURATE

SDRC
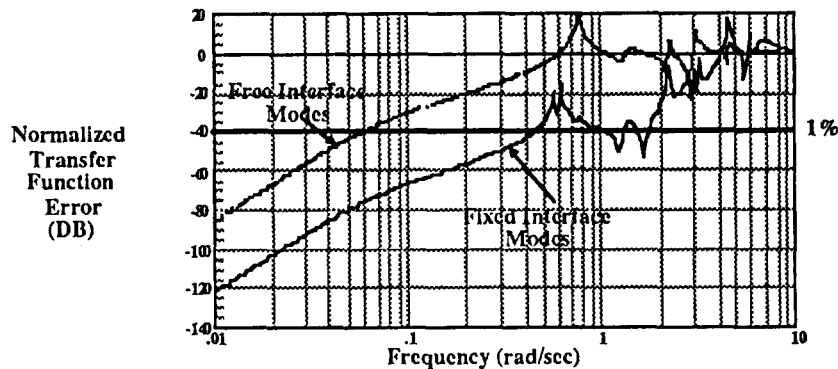
Another measure of accuracy for the closed-loop system is accuracy of the closed-loop frequency response. Here we examine the frequency response from an attitude command about the y-axis (roll) to response about the same axis. Normalized error at each frequency is defined as the distance of the approximate to the exact frequency response, divided by the magnitude of the exact response. Note that both representations are inaccurate at high frequencies (where modes are neglected), but that the fixed interface representation is more accurate at lower frequencies. The control frequency in this case is more than an order of magnitude lower than the flexible frequencies and again the results are consistent with other models we've examined.

135

SUMMARY

- CO-ST-IN transfers data between NASTRAN, IDEAS, EASY5 and MATRIXx

- Modal ordering reduces model size

- Mode acceleration data recovery performed in control simulation

- Fixed interface modal representations result in more accurate closed-loop models

SDRC

NOTES

Our work with NASA Lewis is on-going, and we will be continuing to develop methods which facilitate fast and accurate closed-loop structural analyses. We will also continue to place emphasis on the selection of improved structural representations for control/structure interaction studies.

A Brief List of CO-ST-IN Commands -

DAMP	- Define modal damping ratios.
DMPDAT	- Write all data to an unformatted file.
DRSC	- Scale data recovery outputs for modal ordering.
GYRO	- Define gyroscopic forces due to a spinning body.
INPT	- Define an absolute or relative structural input.
INPT1	- Define a generalized structural input.
INSC	- Scale inputs for modal ordering.
MACC	- Define an absolute or relative acceleration measurement.
MASC	- Scale acceleration measurements fro modal ordering.
MDSP	- Define an absolute or relative displacement measurement.
MDSC	- Scale displacement measurements for modal ordering.
MVEL	- Define an absolute or relative velocity measurement.
MVSC	- Scale velocity measurements for modal ordering.
OACC	- Define an absolute or relative acceleration output.
OASC	- Scale acceleration outputs for modal ordering.
ODSP	- Define an absolute or relative displacement output.
ODSC	- Scale displacement outputs for modal ordering.
OGRP	- Group modes for ordering.
ORDER	- Order modes on the basis of approximate balanced singular values.
ORDUSR	- User-defined modal ordering.
OVEL	- Define an absolute or relative velocity output.
OVSC	- Scale velocity measurements for modal ordering.
PARAM	- Define various problem parameters.
PID	- Define a PID controller.
PULSE	- Define a pulse train input.
REDDAT	- Read unformatted data file written by DMPDAT.
RCS	- Define a simple reaction control system for space station reboost.
RDRM2	- Read data recovery matrices from a NASTRAN Output2 file.
RDRM4	- Read data recovery matrices from a NASTRAN Output4 file.
RMN2	- Read modal data from a NASTRAN Output2 file.
RMU	- Read modal data from an I-DEAS Universal file.
RRESP	- Read response time histories from an EASY5 Plots file.
STITLE	- Define a problem subtitle.
TITLE	- Define a problem title.
WDRM4	- Write data recovery matrices in NASTRAN Output4 format.
WEAD	- Write an EASY5 Analysis Definition File.
WEMG	- Write an EASY5 Model Generation File.
WLODN	- Write NASTRAN FORCE and MOMENT cards for static solution.
WMATX	- Write matrices in MATRIXx format.
WRSPN	- Write structural input force response as NASTRAN Bulk Data.
WRSPU	- Write EASY5 responses in I-DEAS Universal File Format.

137

FLEXIBLE MISSILE AUTOPILOT DESIGN STUDIES WITH PC-MATLAB/386

By

Michael J. Ruth
Johns Hopkins University/Applied Physics Laboratory
Laurel, Maryland

ABSTRACT

Development of a responsive, high-bandwidth missile autopilot for airframes which have structural modes of unusually low frequency presents a challenging design task. Such systems are viable candidates for modern, state-space control design methods. The PC-MATLAB interactive software package provides an environment well-suited to the developement of candidate linear control laws for flexible missile autopilots. The strengths of MATLAB include: (1) Exceptionally high speed -- MATLAB's version for 80386-based PC's offers benchmarks approaching minicomputer and mainframe performance; (2) Ability to handle large design models of several hundred degrees of freedom, if necessary; and (3) Broad extensibility through user-defined functions. To characterize MATLAB capabilities, a simplified design example is presented. This involves interactive defintion of an observer-based state-space compensator for a flexible missile autopilot design task. MATLAB capabilities and limitations, in the context of this design task, are then summarized.

FLEXIBLE MISSILE AUTOPILOT DESIGN STUDIES
WITH PC-MATLAB/386

Michael J. Ruth
Johns Hopkins University / Applied Physics Laboratory
Laurel, Maryland

Workshop on Computational Aspects in the Control
of Flexible Systems

12-14 July 1988
Williamsburg, Virginia

141

PRESENTATION OVERVIEW

---

1. Introduction

2. MATLAB Background

3. Characteristics of MATLAB Environment

4. Classical Control Capabilities

5. Modern Control Design Example

6. Summary

142

INTRODUCTION

- JHU/APL acts as technical direction agent for US Navy weapon system programs

- A key task of APL's Guidance, Control, and Navigation Systems Group is the evaluation or conceptual design of missile guidance and control systems

- Analysis and design work requires a <u>flexible</u>, <u>interactive</u> linear modeling tool

- PC-MATLAB resident on 80386 engineering work-stations provides such a tool

- Work presented here shows general attributes of MATLAB, demonstrating use of PC-MATLAB/386 for linear design of a flexible missile autopilot

143

MATLAB BACKGROUND

- MATLAB (MATrix LABoratory) provides an interactive, matrix-oriented environment

- MATLAB is based on the EISPACK and LINPACK routines for matrix computations

- PC-MATLAB/386 is a high-performance MATLAB implementation for 80386-based workstations

- MATLAB built-in functions, plus higher-level functions developed for control system calculations, allow for effective controls design studies

144

HARDWARE AND SOFTWARE CONFIGURATION

- COMPAQ 386/20 computer

- Weitek 1167 numeric coprocessor

- PC-MATLAB/386 with Control Systems Toolbox

145

PC-MATLAB/386 ATTRIBUTES

■ **Interactive, high-level command environment**

■ **Very high processing speed**

■ **Easy extensibility via user-defined functions**

146

A MATLAB INTERACTIVE COMMAND LINE EXAMPLE

>> k = lqr(a,b,q,rho*r); eig(a-b*k), y = step(a-b*k,b,c,d,1,t); plot(t,y);

- The <u>single</u> <u>line</u> above, typed at the MATLAB command line prompt, does several things:

    - Computes a quadratic regulator gain vector

    - Displays the closed-loop eigenvalues -- often useful for confirming that actuator band-width requirements are not excessive

    - Computes and plots a unit step response

- By varying the control cost (rho) above, a very large family of compensators may quickly be considered

- The above command line suggests the power and utility available from a high-level, inter-active matrix language

PC-MATLAB/386 PROCESSING SPEED

- **MATLAB's LINPACK Benchmark: 460 double precision KFLOPS**

- **This processing speed is:**

    - **25 x faster than standard PC/AT**
    - **6 x faster than Mac II**
    - **3 x faster than MicroVax II**

- **Implication: the fast response time resulting from
  such performance allows for truly interactive
  design iterations on complex control laws**

148

MATLAB EXTENSIBILITY

- User-defined functions may be developed through creation of simple text files

- Some typical user-defined functions:

  - Frequency-response plotting routines

  - Application-specific linear transformations

  - Multivariable Nyquist criterion

- Complex state-space or transfer-function models also defined through user text files

149

AN EXAMPLE OF A USER-DEFINED COMMAND FILE

- **Below command set calculates and plots the maximum and minimum singular values of a plant and observer-based compensator, for a loop broken at plant input**

```
function [smin,smax] = svdinput(a,b,c,kcon,kobs,w);
%
jay = sqrt(-1);
[nn,xx]=size(a); i2=eye(nn); [ng,xx]=size(c*a*b); phi = '(s*i2-a)'';
for i = 1:nc;
    s = w(i)*jay; phieval = eval(phi);
    gs = c/phieval*b; ks = kcon / (phieval+b*kcon+kobs*c) * kobs;
    xx=svd(ks*gs); smin(i)=xx(ng); smax(i)=xx(1);
end;
%
%
% convert to decibels and plot output
%
smin=20*log10(smin); smax=20*log10(smax);
semilogx(w,smin,w,smax,'r--'); grid;
title('Max and Min Singular Values; Loop Broken at Plant Input ');
xlabel('Frequency  (rad/sec)'); ylabel('Magnitude  (db)');
```

- **Procedure requires only eleven lines of executable MATLAB code**

150

CLASSICAL CONTROL CAPABILITIES

- Frequency response

- Root locus

- Nyquist plots

- Development of dynamic compensators
  (lead-lag, notch filters, etc)

MODERN CONTROL DESIGN EXAMPLE

- Design plant describes tactical missile at a high-altitude flight condition

- Design plant includes single-plane rigid-body dynamics and effect of first flexible mode on sensed pitch rate

- Objective is to develop an autopilot to track commanded accelerations

- Design challenge is to achieve high closed-loop bandwidth in presence of low-frequency bending modes

152

DESIGN APPROACH

- Establish design goals for closed-loop responsiveness and stability

- Develop full-state feedback (LQR) gains for design plant

- Define linear observer to reconstruct full state vector

    - Use "robust observer" design (Doyle and Stein, 1979 IEEE Transactions on Automatic Control)

    - Adjust observer gains to recover original LQR loop transfer in desired frequency range

153

DESIGN PLANT MODEL

- Fifth-order state vector $\underline{x}$; $\dot{x} = \underline{A}x + \underline{b}u$

- $\underline{x} = [q_r \quad q_r/s \quad a/s \quad q_f/s \quad q_f]$

- First three state variables are associated with rigid-body airframe; the last two describe flexible mode dynamics

- Rate gyro measurement: $[1\ 0\ 0\ 0\ 1]\ ^*\ \underline{x}$

- (Integrated) accelerometer measurement: $[0\ 0\ 1\ 0\ 0]\ ^*\ \underline{x}$

$$A = \begin{bmatrix} 0 & -2.3557e+02 & 1.7967e+02 & 0 & 0 \\ 1.0000e+00 & 0 & 0 & 0 & 0 \\ 0 & 2.6158e+00 & -1.9951e+00 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0000e+00 \\ 0 & 0 & 0 & -2.4649e+04 & -3.1400e+00 \end{bmatrix}$$

$$b = \begin{bmatrix} -2.8031e+02 \\ 0 \\ 9.2587e-02 \\ 0 \\ 3.0723e+02 \end{bmatrix}$$

154

SOME OBSERVATIONS ON DESIGN PLANT MODEL

- Feedback of the first three states describes a very standard (rigid-body) autopilot topology, used by tactical missiles since 1950's

- Open-loop plant is characterized by lightly damped airframe (weathercock) poles, and by bending mode poles

  - Airframe pole frequency lies at nominal 2.5 Hz

  - Bending mode has nominal 25 Hz natural frequency

- Desired autopilot crossover frequency here will lie near the bending mode frequency

EFFECT OF STRUCTURAL MODE ON SENSED PITCH RATE
(RATE GYRO MEASUREMENT)

Response to Unit Fin Deflection

156
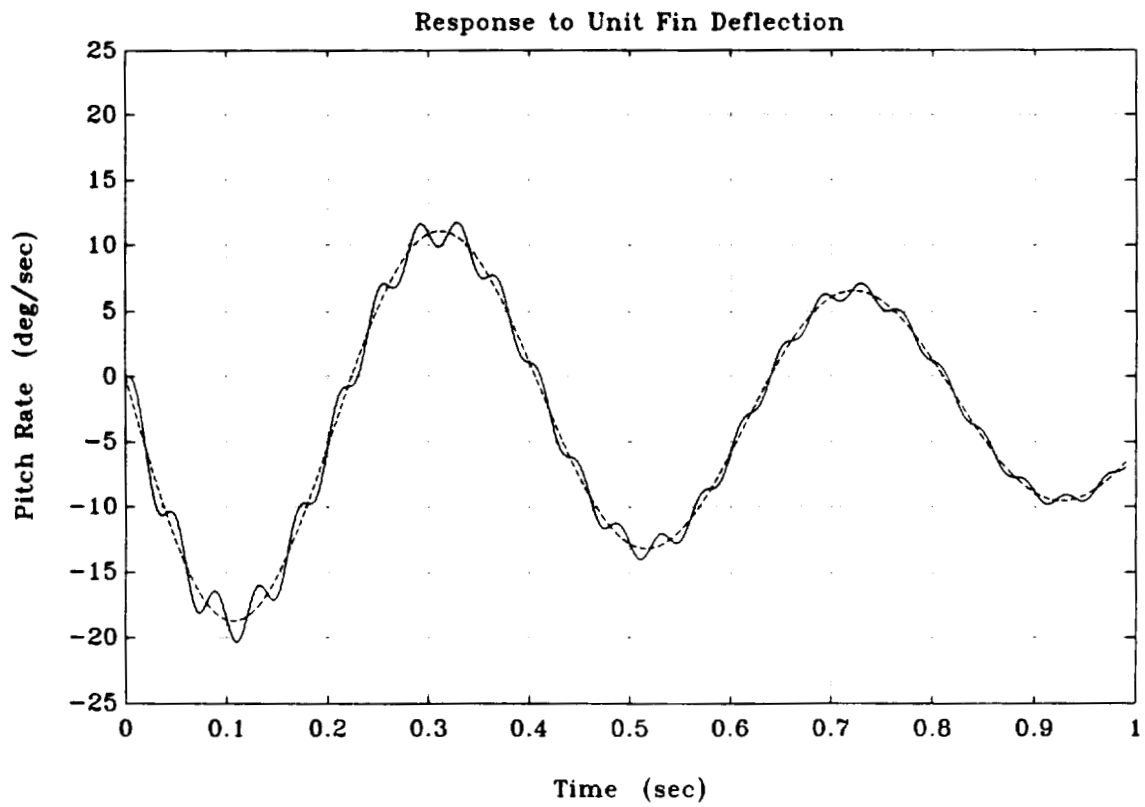
CONTROLLABILITY AND OBSERVABILITY PROPERTIES OF PLANT

- System ($\underline{A},\underline{b}$) is controllable

- System is unobservable if rate gyro alone, or accelerometer alone, is used as the measurement to reconstruct state vector

- Both sensor outputs thus should be used in the observer design

- Approach taken for this application:

    - Define a (non-square) design plant having one input (fin deflection) and two independent outputs (gyro and accelerometer)

    - Use extensions of loop transfer recovery (Williams and Madiwale, 1985 ACC) valid for non-square systems

157

FREQUENCY RESPONSE OF FULL-STATE FEEDBACK (LQR) SYSTEM (LOOP BROKEN AT PLANT INPUT)
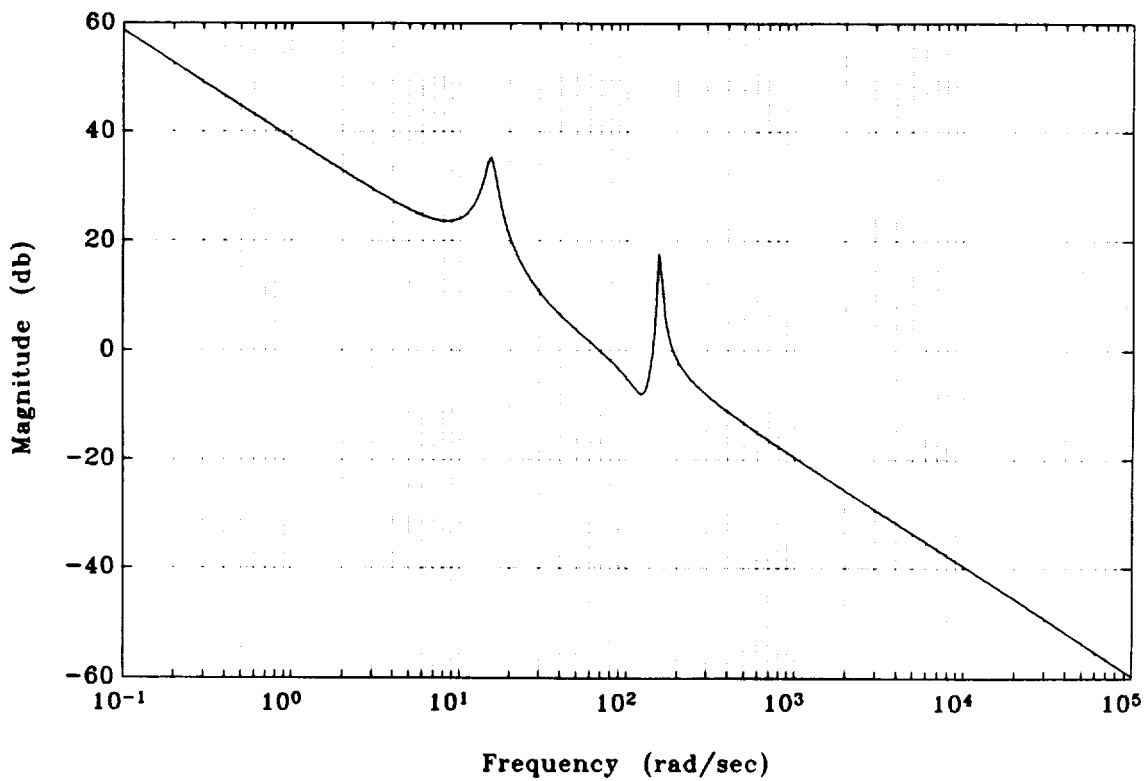
OBSERVATIONS ON LOOP TRANSFER RECOVERY PROCEDURE

- For this application, recovery at <u>both</u> the (rigid-body) airframe and bending mode frequencies may only be achieved with very high observer gains

- For practical ranges of observer gains, recovery at airframe frequencies is obtained at the cost of lessened robustness in the structural mode frequency range

- Use of a set of user-defined MATLAB files, to implement a range of observer gain calculations, makes evaluation of this robustness tradeoff straightforward

159

RECOVERY OF DESIRED FULL-STATE FEEDBACK SYSTEM
WITH MODEL-BASED COMPENSATOR

Asymptotic Loop Transfer Recovery Properties of Compensator

160
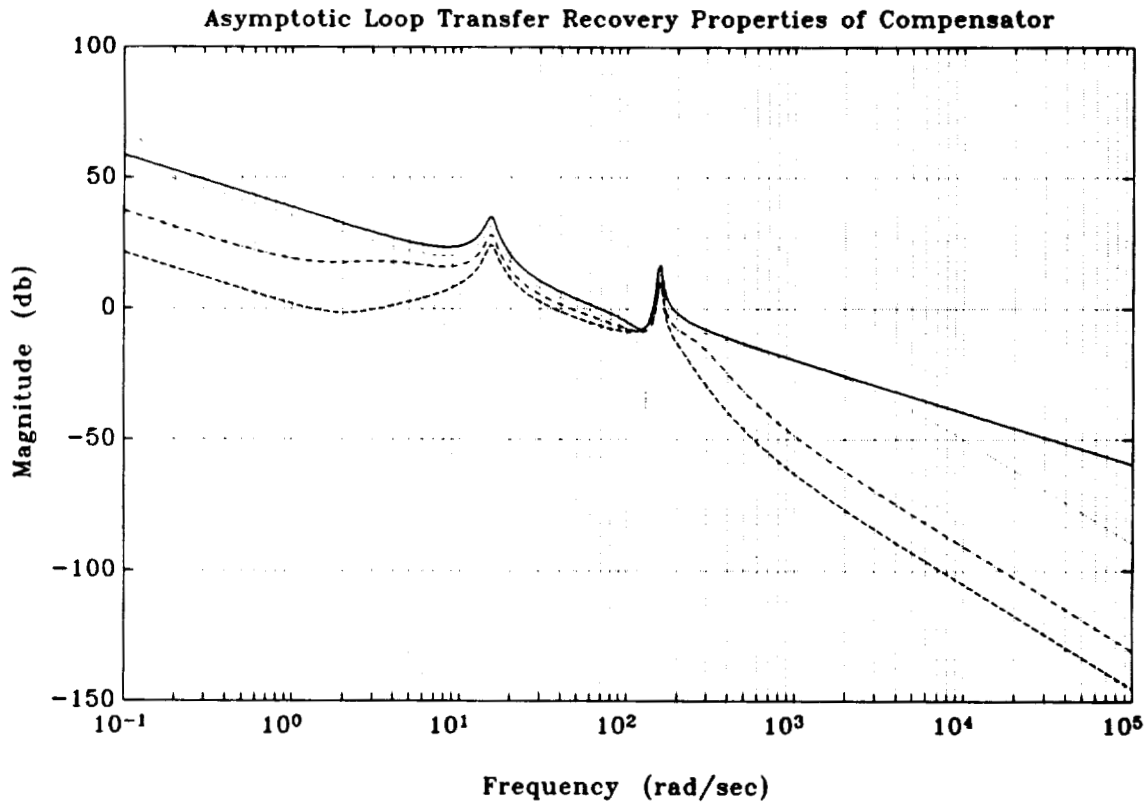
ACCELERATION STEP RESPONSE OF FINAL COMPENSATOR DESIGN

Response to 1-Gee Acceleration Command
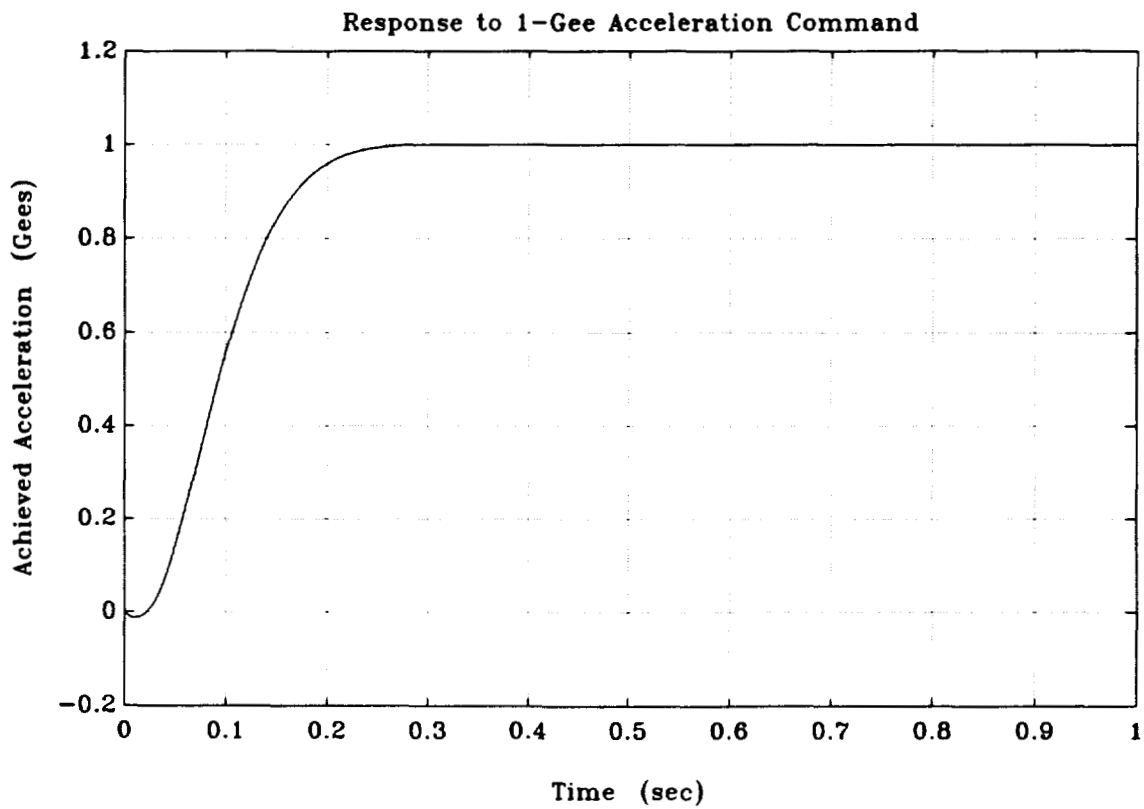
RESPONSE OF FLEXIBLE MODE STATE DURING ACCELERATION STEP RESPONSE

Pitch Rate Response Due to Flexible Mode

162
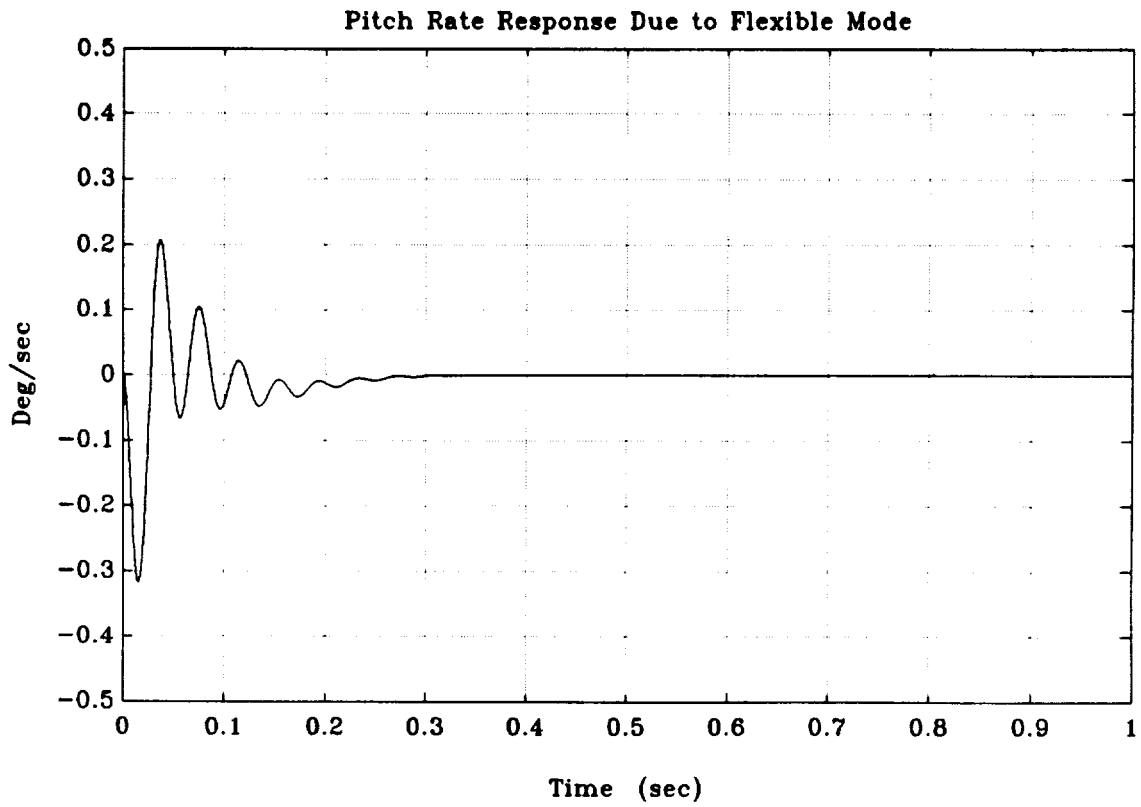
ACCELERATION STEP RESPONSE FOR CASE WHEN
BENDING MODE IS PERTURBED TO 25 % LOWER VALUE

Response to 1-Gee Acceleration Command
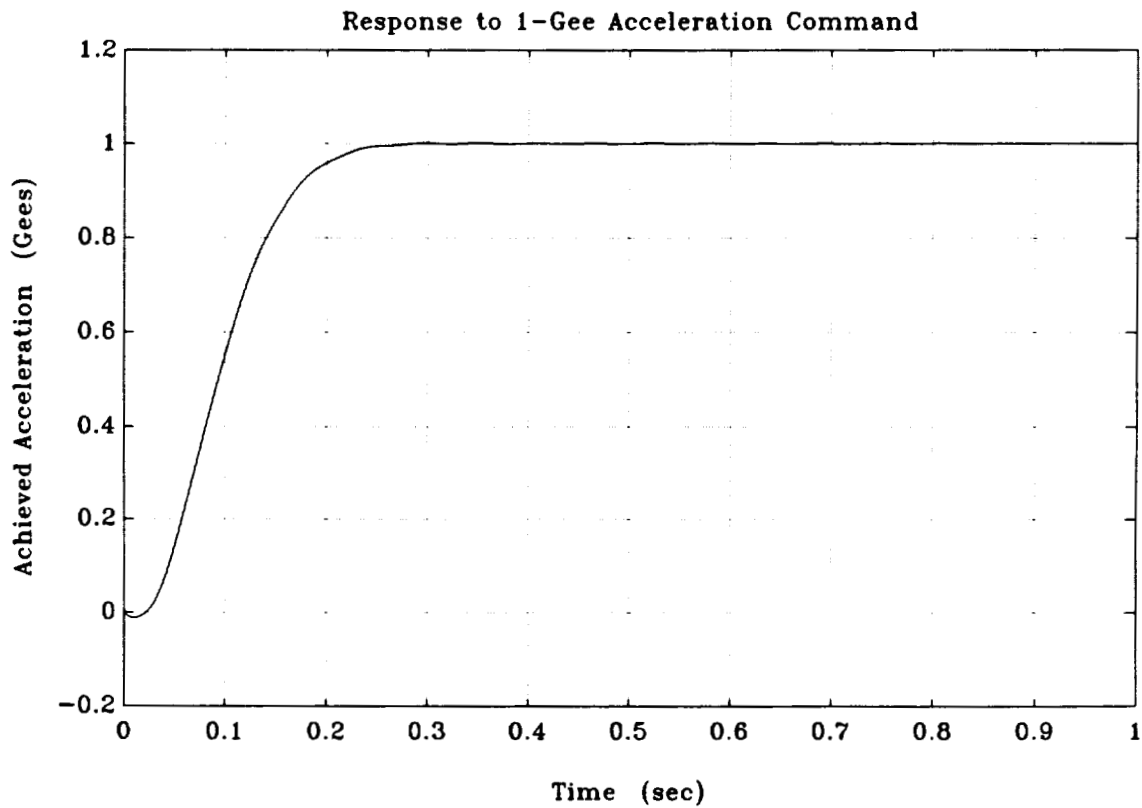
COMPARISON OF ACTUAL AND RECONSTRUCTED FLEXIBLE MODE STATE DURING STEP RESPONSE -- BENDING MODE PERTURBED TO 25 % LOWER VALUE

Actual (−) and Reconstructed (−−) Flexible Mode State
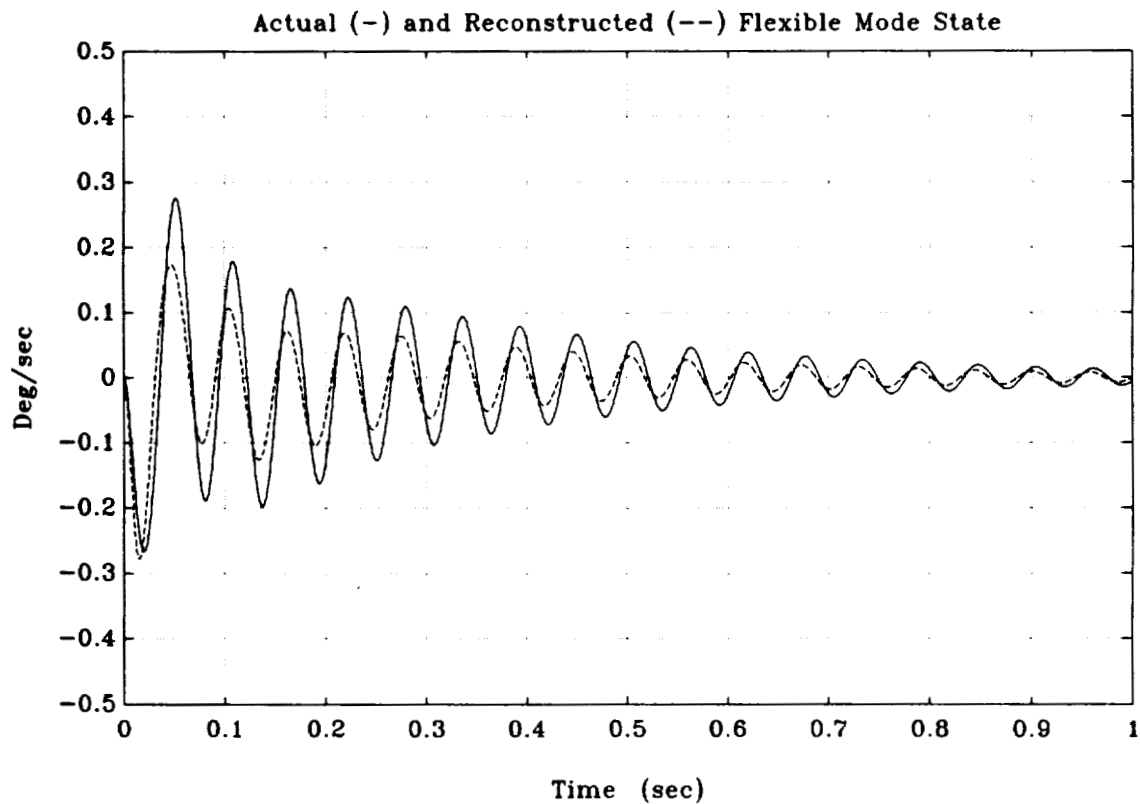
SUMMARY OF DESIGN RESULTS

- Model-based compensator yields a high-bandwidth autopilot, which is robust to at least a 25% perturbation in bending mode frequency

- A number of issues still not addressed:

  - Detailed noise sensitivity assessment
  - Effect of higher-frequency structural modes
  - Phase lag from actuator dynamics
  - Effect of structural modes on accelerometer measurement
  - Tolerance to uncertainties in aerodynamics

- Above concerns could also be addressed using MATLAB

165

SUMMARY: MATLAB APPLICABILITY FOR CONTROL DESIGN OF FLEXIBLE SYSTEMS

- MATLAB provides the necessary tools for a variety of control system design techniques

- Extensibility of MATLAB allows development of tools to implement recent modern control design methods, including loop transfer recovery

- Implementation for 80386-based machines (PC-MATLAB/386) has very high performance, allowing for interactive control design of complex systems such as flexible structures

- Any flexible structures control problem which can be cast into a state-space framework may benefit from design work with MATLAB

166

DYSCO - A SOFTWARE SYSTEM FOR MODELING GENERAL DYNAMIC SYSTEMS

By

Alex Berman
Kaman Aerospace Corporation
Bloomfield, Connecticut

ABSTRACT

The DYSCO program has been under development since 1979. It has been funded by Army and Air Force laboratories and by the Kaman Aerospace Corporation. It is presently available at a number of government and nongovernment installations. It has been used to analyze a very broad range of dynamics problems.

A principle feature of the software design of DYSCO is the separation of the executive from the technology. The executive, which controls all the operations, is "intelligent" in the sense that it "knows" that its function is to assemble differential equations and to prepare them for solution. The "technology library" contains FORTRAN routines which perform standard functions, such as, computing the equation coefficients of an element (or "component") given the local state at any time. The technology library also contains algorithms and procedures for solving the coupled system equations.

The system was designed to allow easy additional of technology to the library. Any linear or nonlinear structural entity, control system, or set of ordinary differential equations may be simply coded and added to the library, as well as algorithms for time or frequency domain solution.

The program will be described with emphasis on its usefulness in easily modeling unusual concepts and configurations, performing analysis of damage, evaluating new algorithms, and simulating dynamic tests. Illustrations of several typical and illustrative applications will be presented. A summary of the technology presently residing in the technology libraries at the various sites will also be given.

KAMAN
AEROSPACE
CORPORATION

891

DYSCO - A SOFTWARE SYSTEM
FOR MODELING GENERAL DYNAMIC SYSTEMS

ALEX BERMAN
KAMAN AEROSPACE CORPORATION

WORKSHOP ON COMPUTATIONAL ASPECTS IN THE CONTROL OF FLEXIBLE SYSTEMS
WILLIAMSBURG, VIRGINIA, JULY 12-14, 1988

691

o   DYNAMIC SYSTEM COUPLER (DYSCO)

o   INITIAL DEVELOPMENT - 1979

o   FUNDED BY ARMY, AIR FORCE, KAMAN

o   PRESENTLY OPERATIONAL ON IBM AND VAX

o   SIZE - 50000+ LINES OF CODE
              350+ SUBROUTINES
                4+ MEGABYTES OF STORAGE

DYSCO COUPLES AND SOLVES SECOND ORDER ODE

o $\quad M_I\ddot{X}_I + C_I\dot{X}_I + K_IX_I = F_I \quad$ (COMPONENT I)

o $\quad X_I = T_IX_S$

o $\quad M_S\ddot{X}_S + C_S\dot{X}_S + K_SX_S = F_S \quad$ (SYSTEM)

o   "COMPONENT" IS MORE GENERAL THAN "FINITE ELEMENT"

o M_I, C_I, K_I, F_I = ARBITRARY FUNCTIONS OF STATE

o X_I = ANY GENERALIZED DOF - PHYSICAL, MODAL, OTHER

o   COMPONENT MAY BE

-    FINITE ELEMENT
-    ASSEMBLY OF FINITE ELEMENTS (SUBSYSTEM, OUTPUT OF FE ANALYSIS)
-    SPECIAL SET OF EQUATIONS (E.G., HELICOPTER ROTOR, SPECIAL MECHANISM)
-    CONTROL ALGORITHM (MIMO, NON-SYMMETRICAL MATRICES, NONLINEAR)
-    FORCE ALGORITHM (M, C, K = NULL, AERO, ELECTROMAGNETIC)
-    ETC., ETC.

172

o   A MODEL IS A DESCRIPTION OF A COUPLED SET OF COMPONENT
    EQUATIONS

o   COMPONENT EQUATIONS ARE DEFINED BY

    -     NAME OF THE ALGORITHM IN "TECHNOLOGY LIBRARY"
    -     NAME OF DATA SET IN "MODELING DATABASE"

o   COMMAND "RUN" COUPLES EQUATIONS

o   NEXT STEP IS TO SPECIFY SOLUTION ALGORITHM
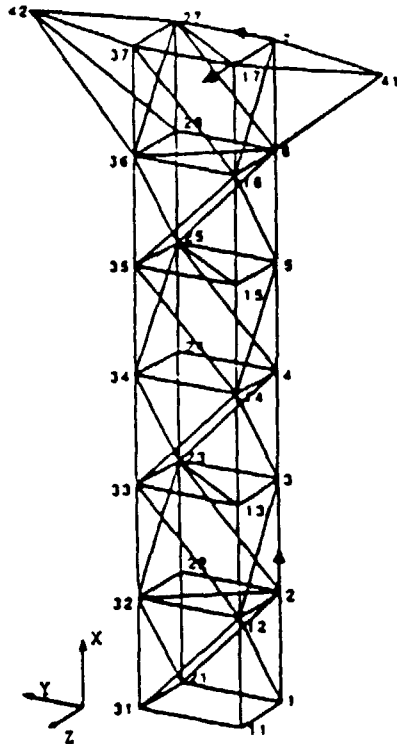
	COMPONENT	NO.	DATA SET
1	CTR4	1	ABCD1
2	CTR4	3	ABCD1
3	CTR4	5	ABCD1
4	CTR4	2	ABCD2
5	CTR4	4	ABCD2
6	CTR4	6	ABCD2
7	CSF1		TOPR
8	CSF1		TOPL
9	CSF1		CONTR
10	CLC1		GROUND

MODELING DATABASE

EXECUTIVE

TECHNOLOGY
MODULE
LIBRARY

EXTERNAL DATABASE

USER

174

KAMAN
AEROSPACE
CORPORATION
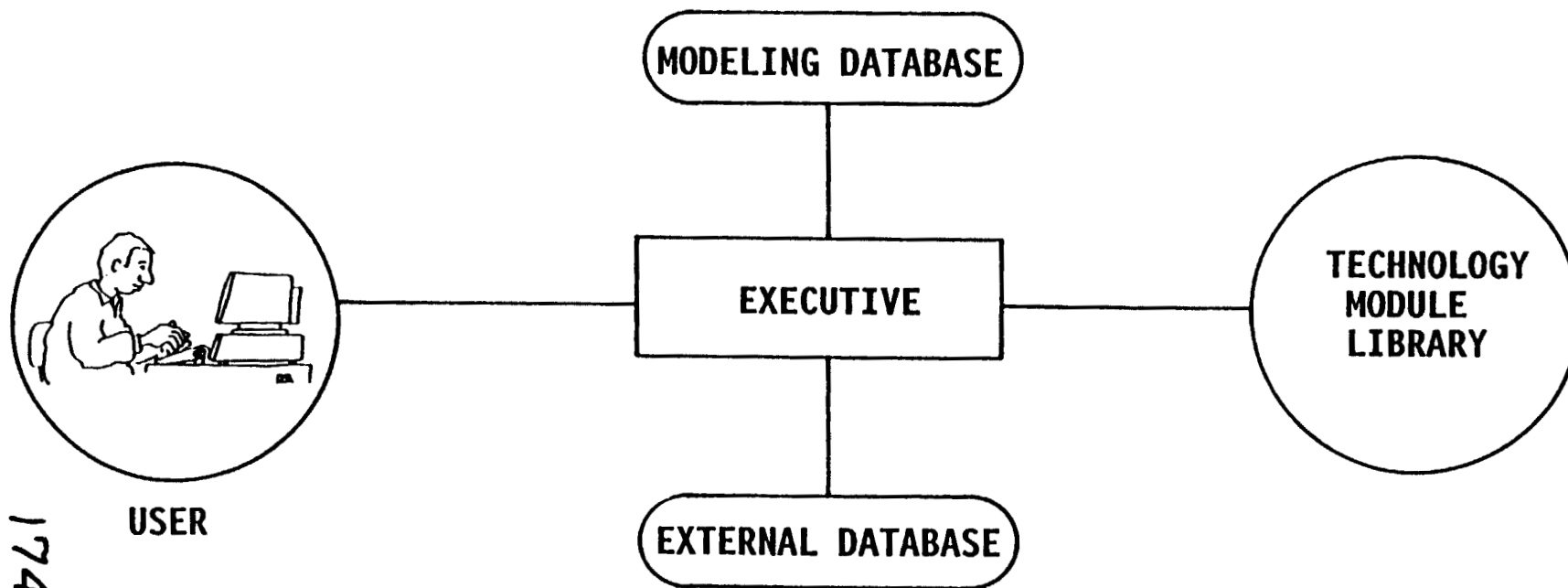
MODELING SCENARIO AND COMMAND RELATIONSHIP

USER INPUT   PROCESS   COMMAND   RDF/UDF

COMP. DATA → DEFINE COMPONENT → NEW → DS/C---

FORCE DATA → DEFINE FORCES → NEW → DS/F---

NAMES OF COMP. FORCES → DEFINE MODEL → NEW → DS/MODEL

RUN

FORM COUPLED SYSTEM

RERUN

SOL. DATA → DEFINE SOLUTION

EXECUTE SOLUTION

175
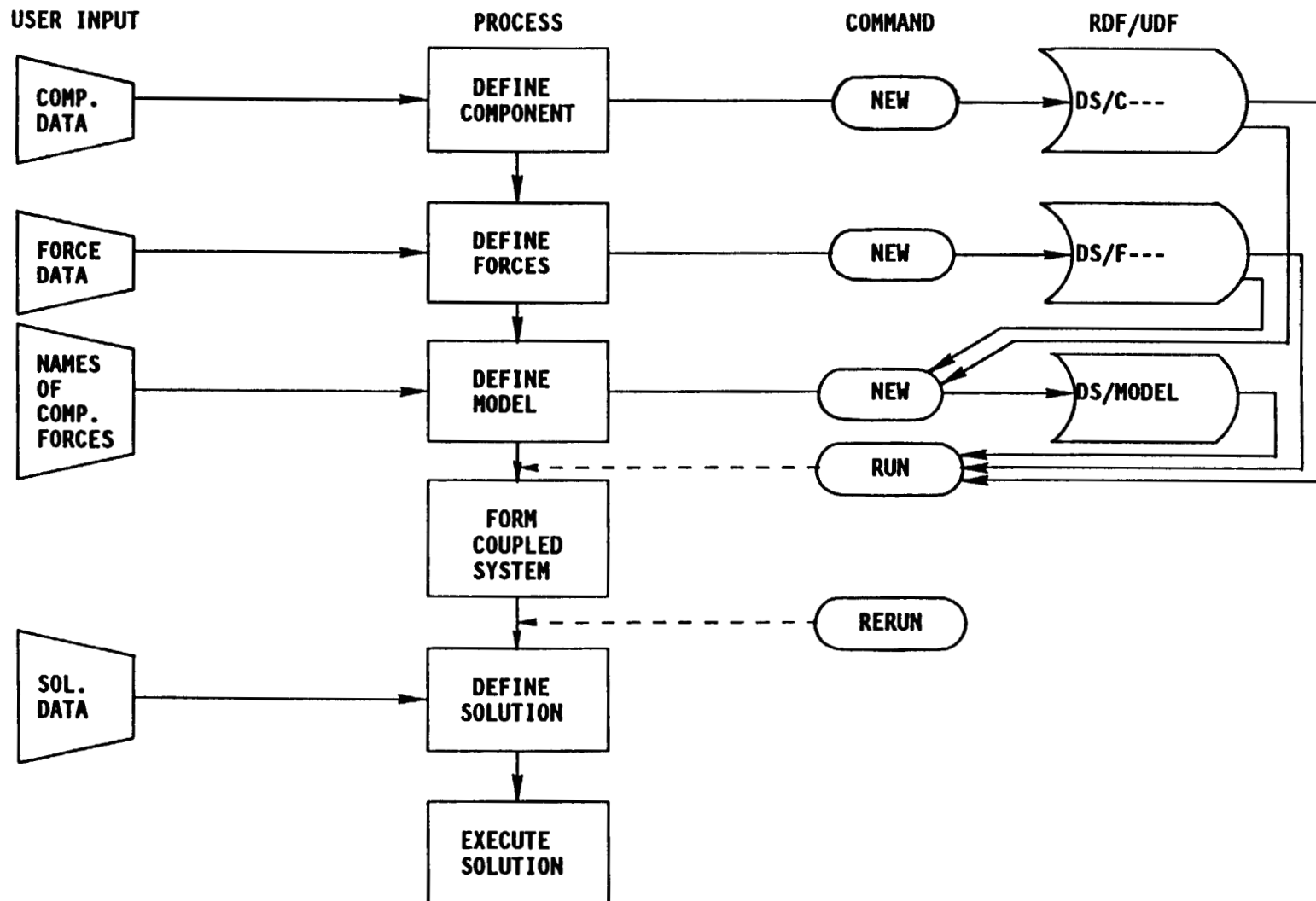
FUNCTION	COMPONENT	FORCE	SOLUTION
INPUT	C---I	F---I	S---I
DEFINITION	C---D	N/A	N/A
COEFFICIENT	C---C	F---C	N/A
ACTIVE	C---A	F---A	S---A
OUTPUT	N/A	N/A	S---O
LOADS	C---L	N/A	N/A

176

---I   INPUT DEFINITION

---D   DEFINE DEGREES OF FREEDOM

---C   COMPUTE CONSTANT COEFFICIENTS IN EQUATIONS

---A   COMPUTE NON-CONSTANT COEFFICIENTS, FUNCTION
         OF TIME AND STATE

---O   OUTPUT

---L   INTERNAL LOADS, FUNCTION OF STATE

178

MODELING		TECHNICAL MODULES		
USER INPUT	PROCESS	COMPONENT	FORCE	SOLUTION
INPUT	DEFINE COMPONENT	C---I		
INPUT	DEFINE FORCE		F---I	
PARTI-CULAR COMP. FORCES NAMES	DEFINE MODEL			
	FORM COUPLED SYSTEM	C---D C---C	F---C	
SOL. DATA	DEFINE SOLUTION			S---I
	EXECUTE SOLUTION	C---A	F---A	S---A S---O

KAMAN
**AEROSPACE
CORPORATION**

179
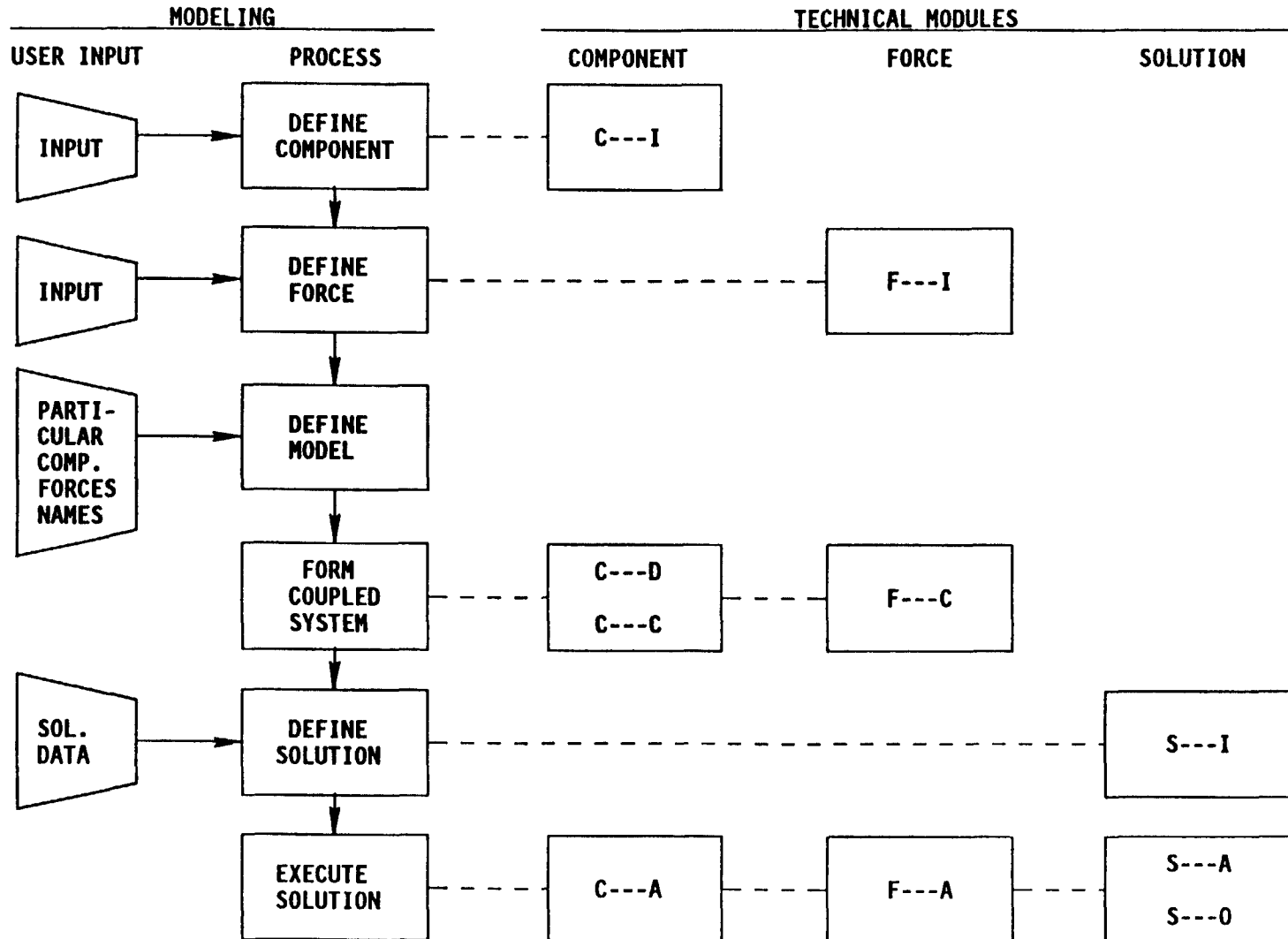
USER INPUT	EXECUTIVE FUNCTION	TECH. LIBRARY	DATA LIBRARY

RUN
MODEL NAME

→ READ DS/MODEL
READ DS/COMPONENTS ← DS/___

ALL
COMPONENTS
OF MODEL

IDENTIFY DOF
AND
CONSTRAINTS ← C___D

FORM ALL T_I AND
SYSTEM DOF

ALL
COMPONENTS
OF MODEL

TRANSFORM CONSTANT
COEFFICIENTS TO SYS-
TEM $M_S = M_S + T_I M_I T_I \ldots$ ← M_I, C_I, K_I, F_I | C___C

REQUEST SOLUTION
MODULE NAME

S___

USER INPUT   EXECUTIVE FUNCTION   TECH. LIBRARY

180

INPUT

$\dot{x}_S, x_S(T_0)$ ...

REQUEST SOLUTION
INPUT
$T = T_0$

$S\_\_\_I$

RETRIEVE COMP. STATE
$x_I = T_I x_S,\ \dot{x}_I = T_I \dot{x}_S(T)$

$x_I,\ \dot{x}_I,\ T$

ALL
COMPONENTS
OF MODEL

TRANSFORM UPDATED
COMP. M, C, K, F,
TO SYSTEM AND SUM
$M_S = M_S + T_I^T \Delta M T_I,\ \ldots$

$\Delta M_I, \Delta C_I, \Delta K_I, \Delta F_I$

$C\_\_\_A$

SOLVE FOR $\ddot{x}_S(T)$

$\ddot{x}_S(T)$

INTEGRATE $\ddot{x}_S$ TO
$\dot{x}_S, x_S, T = T + \Delta$

$\dot{x}_S, x_S(T+\Delta)$

$S\_\_\_A$

o   EXECUTIVE IS SPECIFICALLY BUILT TO MANAGE
     STRUCTURAL DYNAMIC ANALYSIS

o   IT UNDERSTANDS AND MANAGES

     -     INPUT: IDENTIFICATION, STORAGE, EDITING
     -     MODEL BUILDING: RETRIEVAL OF DATA, CALLS
           TO TECHNOLOGY LIBRARY
     -     ASSEMBLY OF EQUATIONS: APPLIES MPC, SPC
     -     SOLUTION OF EQUATIONS: CALLS TO TECHNOLOGY
           LIBRARY, RETRIEVAL OF LOCAL STATES,
           INTERFACE LOADS

o   EXECUTIVE INDEPENDENT OF ANY PARTICULAR AREA OF
     TECHNOLOGY

     -     UNIFORM ABSTRACT INTERFACES TO TECHNOLOGY
           LIBRARY

o   NEW TECHNOLOGY EASILY ADDED
    –   COMPONENT, FORCE, SOLUTION
    –   UNIFORM INTERFACES TO EXECUTIVE
    –   FORTRAN CODING

o   COMPONENTS ARE ANY SECOND ORDER ODE, SUCH AS,
    –   SINGLE SPRING, DAMPER, OR MASS
    –   ANY FINITE ELEMENT
    –   COMPLETE NASTRAN MODEL
    –   HELICOPTER ROTOR
    –   MIMO CONTROL ALGORITHM

o   SOLUTIONS ACT ON MODEL EQUATIONS, E.G.
    –   EIGENANALYSIS
    –   FREQUENCY RESPONSE
    –   TIME HISTORY
    –   HELICOPTER TRIM (PERIODIC SHOOTING)
    –   PERIODIC SYSTEM STABILITY
    –   STATE FEEDBACK OPTIMIZATION

o   VALIDATED INPUT AND EDITING
     -    USES KNOWLEDGE TABLE: TYPE, CHARACTERISTICS,
          EXISTENCE, RANGE
     -    PROMPTED INPUT
     -    INSTANTANEOUS VALIDATION
     -    ASSURED COMPLETE AND CONSISTENT DATA

o   SIMPLE EDITING OF MODEL
     -    CONFIGURATION CHANGES
     -    PARAMETER VARIATION
     -    DAMAGE ANALYSIS

o   INTELLIGENT COUPLING PROCEDURES
     -    RECOGNITION OF DOF NAMES
     -    MPC OPTIONALLY AUTOMATICALLY FORMED
     -    GENERAL MPC SOLVED FOR DOF EQUATIONS

183

o   CSF1 - LINEAR FINITE ELEMENT

       USER SUPPLIES: NAMES OF DOF
                 M, C, K, F

o   CFM3 - 3D MODAL STRUCTURE

       RIGID BODY, ELASTIC MODES (ALL OPTIONAL)
       DOF NAMES AUTOMATICALLY GENERATED
       AUTOMATIC COUPLING AT SPECIFIED NODES

o   CSB2 - GENERAL BAR ELEMENT* (NOT AVAILABLE IN GOVT VERSION)

       MAY BE USED AS A BEAM OR ROD ELEMENT
       SHEAR FACTORS, CONSISTENT MASS, RAYLEIGH DAMPING
       UP TO 12 DOF

184

o   CES1 - ELASTIC STOP

    NONLINEAR SPRING, DAMPING, WITH GAP

o   CGF2 - GENERAL FORCE

    -   POLYNOMIAL, FOURIER SERIES, OR TABULAR
    -   PERIODIC

o   CLC0 - SINGLE POINT CONSTRAINTS

o   CLC1 - MULTIPOINT CONSTRAINTS

o   CLC2 - ADVANCED MULTIPOINT CONSTRAINT

185

o   SEA4 - EIGENANALYSIS, REAL

o   SEA5 - COMPLEX EIGENANALYSIS

o   STH4 - TIME HISTORY
           - CONDITION CODES

o   SFD1 - FREQUENCY DOMAIN MOBILITY
           - RESPONSE PER UNIT FORCE

o   STCO - OPTIMIZER FOR LINEAR STATE FEEDBACK* (NOT
             AVAILABLE IN GOVT VERSION)
           - SOLVES MATRIX RICCARTI EQUATION
           - INTEGRATES SYSTEM STATE EQUATIONS

o   SII3 - INTERFACE AND INTERNAL LOADS
           - RESIDUAL FORCES AT INTERFACES
           - FORCES, STRAIN ENERGY, BENDING MOMENTS

o   CRR2, CRR3 - HELICOPTER ROTOR

o   CCE0, CCE1 - ROTOR CONTROL SYSTEM

o   CRD3 - ROTOR DAMAGE

o   CFM2 - HELICOPTER FUSELAGE

o   CLG2 - NONLINEAR LANDING GEAR

o   CLS2 - LIFTING SURFACE

o   FRA0, FRA2, FRA3 - ROTOR AERODYNAMICS

o   FFA0, FFC2 - FUSELAGE AERODYNAMICS

o   STH3 - TIME HISTORY, HELICOPTER CONTROLS

o   STR3 - HELICOPTER TRIM

o   SSF3 - FLOQUET STABILITY

187

o   DYSCO DOES NOT COMPETE WITH FE CODES

o   DYSCO COMPLEMENTS FE CODES

o   FE ANALYSIS FOR DETAILED STRUCTURAL ANALYSIS

o   DYSCO CAN START WITH FE MODEL AND:

- MODIFY CONFIGURATION
- SIMULATE DAMAGE
- ADD CONTROL ALGORITHMS
- ADD SPECIAL COMPONENTS
- PERFORM SOLUTIONS ON ALL MODIFICATIONS
- STUDY EFFECTS OF CHANGE
- ANALYZE CONFIGURATIONS NOT POSSIBLE (OR CONVENIENT) WITH FE CODES

o   DYSCO CAN ALSO MODEL STRUCTURES ON ITS OWN

KAMAN
AEROSPACE
CORPORATION

o   SIMPLE PROBLEMS ARE EASY AND INEXPENSIVE TO SOLVE

o   PROBLEMS NOT CONVENIENTLY MODELED ELSEWHERE CAN BE
     SOLVED

o   PHENOMENA CAN BE BETTER UNDERSTOOD:
     -    START WITH SIMPLE REPRESENTATION
     -    GRADUALLY INCREASE COMPLEXITY
     -    VARY PARAMETERS
     -    VARY CONFIGURATION

o   NOVEL CONCEPTS CAN BE EASILY MODELED AND EVALUATED

o   NEW ALGORITHMS CAN BE TESTED AND EVALUATED

189

o   PACOSS TOWER DYNAMIC ANALYSIS

o   TRUSS STRUCTURE WITH ACTIVE ELEMENTS - VIBRATION CONTROL

o   PIEZOELECTRIC SENSORS/ACTUATORS ON BEAM - VARY CONTROL
      LAWS, ADD ELASTIC STOP, STABILITY, TIME, FREQUENCY
      DOMAIN

o   POINTING-TRACKING SYSTEM - MOTOR DRIVEN MIRRORS - MOVING,
      ACCELERATING TARGET, VARY CONTROL GAINS

o   ROTORCRAFT TRIM - DAMAGED BLADE - INTERNAL LOADS

o   RAIL GUN PNEUMATIC ACCELERATOR - GAS PRESSURE - BOLT MOTION

o   ALGORITHM EVALUATION - REDUCED MODELS, SYSTEM IDENTIFICATION,
      SIMULATE EFFECTS OF MEASUREMENT ERRORS

1990

o   DYSCO COUPLES THE EQUATIONS OF INDIVIDUAL COMPONENTS TO
     FORM THE EQUATIONS OF A MODEL

o   EACH COMPONENT AND MODEL ARE OF THE FORM

$$M\ddot{X} + C\dot{X} + KX = F$$

$$M, C, K, F = F(T, \dot{X}, X)$$

o   M, C, K, F MAY BE ARBITRARY FUNCTIONS OF TIME OR STATE

o   X MAY REPRESENT PHYSICAL, MODAL, OR ANY GENERALIZED DOF

o   EACH COMPONENT IS REPRESENTED BY FORTRAN SUBROUTINES IN
     THE TECHNOLOGY LIBRARY

KAMAN
AEROSPACE
CORPORATION

o   DYSCO USES AN "INTELLIGENT" PROCEDURE FOR
     COUPLING DEGREES OF FREEDOM

o   DEGREES OF FREEDOM OF COMPONENTS MAY BE:
     -    PHYSICAL COORDINATES
     -    MODAL DISPLACEMENTS
     -    ANY GENERALIZED COORDINATES

o   COUPLING INCLUDES
     -    PHYSICAL TO PHYSICAL
     -    PHYSICAL TO MODAL
     -    MODAL TO MODAL
     -    SINGLE POINT CONSTRAINTS
     -    MULTIPLE POINT CONSTRAINTS
     -    ANY LINEAR RELATIONSHIPS

o   EFFECTS SIMULATED
     -    RIGID PHYSICAL LINKAGES
     -    OPTICAL BEAM COORDINATES
     -    CONTROL ALGORITHMS

X, X_I ARE VECTORS OF THE DOFs OF THE SYSTEM (MODEL) AND
THE COMPONENTS

T_I IS A TRANSFORMATION MATRIX

$$X_I = T_I X$$

THE EQUATION OF THE MODEL IS

$$MX + CX + KX = F$$

WHERE

$$M = \sum T_I^T M_I T$$

$$C = \sum T_I^T C_I T$$

$$K = \sum T_I^T K_I T$$

$$F = \sum T_I^T F_I$$

EACH T_I IS AUTOMATICALLY FORMED IN DYSCO

o   DYSCO USES A UNIQUE PROCEDURE WHERE VARIABLE NAMES (A4, I4) ARE
     RECOGNIZED AND PROCESSED

o   NAMES ARE AUTOMATICALLY FORMED OR USER SUPPLIED

o   LIKE NAMES IN COMPONENTS IMPLY CONNECTION

o   SIMPLE EXAMPLE:

     X1   X2   X3   X4
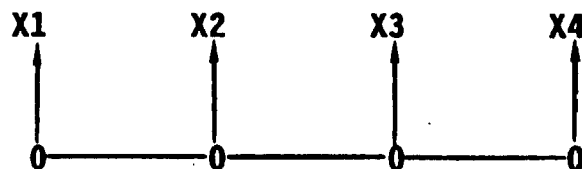
TO CONSTRAIN X3 TO GROUND BY A SPRING, κ, USER SIMPLY ADDS
COMPONENT TO MODEL WITH FOLLOWING INFORMATION

         NO OF DOF = 1
         NAME = X3
         M = C = F = NULL
         K = κ

KAMAN
AEROSPACE
CORPORATION

X1   X2   X3   X4   X5   X6

SENSOR LOCATIONS X1, X4, X6
ACTUATOR LOCATIONS X3, X4

$F_{X3} = A*X1 + B*X4 + C*X6 + D*\dot{X}4$

$F_{X4} = E*\dot{X}1 + F*X6 + G*\dot{X}6$

THIS MAY BE REPRESENTED BY COMPONENT WITH

DOF = [X1, X3, X4, X6]

M = 0

$$C = - \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & D & 0 \\ E & 0 & 0 & G \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$K = - \begin{bmatrix} 0 & 0 & 0 & 0 \\ A & 0 & B & C \\ 0 & 0 & 0 & F \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

SENSOR AND ACTUATOR LOCATIONS MAY BE CHANGED BY EDITING
DOF NAMES. GAINS MAY BE CHANGED BY EDITING MATRICES.

KAMAN
AEROSPACE
CORPORATION

o   COUPLING MAY ALSO USE OPTIONAL LINEAR RELATIONS

o   SINGLE POINT, MULTIPOINT, CONNECTIVITY CONSTRAINTS

$$X1 = 1.0 * X2$$
$$X10 = 0$$

$$A * X10 + B * Y100 + C * Z20 = 0$$

o   REPRESENTATION OF MECHANICAL LINKAGES

o   CONVERSION TO CONVENIENT PARAMETERS
     -    OPTICAL BEAM ANGLE AS FUNCTION OF MIRROR DOF
     -    TIP DISPLACEMENT OF BEAM AS FUNCTION OF MODAL DOF

THE DOMAIN IS "COUPLED DYNAMIC EQUATIONS"

IT EXECUTES TECHNICAL MODULES IN "PARALLEL"
(RATHER THAN IN SEQUENCE)

SIMPLE COMMANDS PERFORM NUMEROUS MODULE
EXECUTIONS (E.G., RUN)

INVALID COMMAND SEQUENCES ARE NOT ACCEPTED
INVALID DATA USAGE IS NOT POSSIBLE

ALL DATA PLACED ON FILES OR EDITED IS ASSURED TO BE
VALID (E.G., CONSISTENCY AND FORMAT)

o   COMPONENT - ALGORITHM FOR COMPUTING M, C, K, F FOUND IN
    TECHNOLOGY LIBRARY

         NAME: C---

o   DATA SET - SPECIFIC SET OF DATA TO BE USED WITH A COMPONENT FOUND
    IN DATA LIBRARY. INPUT BY USER. USER SUPPLIES DATA SET "NAME"

o   MODEL - COLLECTION OF COMPONENTS AND ASSOCIATED DATA SETS

                            SAMPLE MODEL

COMPONENT	DATA SET
CSF1	STRUT1
CSF1	STRUT2
CFM2	FUSELAGE
CSF1	CONTROL 3
CES1	K100

```
*************************     MODEL PATIE12     *****************************

CATSTAR: SAME AS PATIE11 BUT CLC1 REWRITTEN FOR PHIP1 IMPLICIT

INDEX     COMP     NO.     DATA SET          FORCE     DATA SET

  1       CSF1             PRIMARY           NONE
  2       CSF1             BASE P-S          NONE
  3       CSF1             SECONDRY          NONE
  4       CSF1             BASE S-M          NONE
  5       CSF1             MOTOR             NONE
  6       CSF1             TRUSS-PR          NONE
  7       CSF1             TRUSS-DI          NONE
  8       CSF1             TRUSS-MO          NONE
  9       CFM2      .1     SHUTTLE           NONE
 10       CSF1             CONTROL1          NONE
 11       CSF1             TARGET            NONE
 12       CSF1             NOMAL             NONE

 13       CLC1             OPTICS3           NONE

*******************************************************************************

*******************************************************************************
                        GLOBAL   VARIABLES

 NO INPUT REQUIRED
*******************************************************************************
```

```
**************       CONTROL1/CSF1       ***************

CONTROL LAW 1 FOR GIMBALLED MIRROR

****************************************************************
INPUT FOR COMPONENT CSF1. FINITE ELEMENT

    1 NCDF       - NUMBER OF DOF          =          3
    2 CDFLI      - (DOF) DOF NAME
                 PHIP1000   DSTR1000   MTH 1000
    3 CM         - (REAL) MASS MATRIX VALUES
                 NULL MATRIX

    4 CC         - (REAL) DAMPING MATRX VALUES
                 GENERAL MATRIX

      ROW     1       NULL ROW
      ROW     2
              3.70000E+02   0.00000E+00   0.00000E+00

      ROW     3
              -3.70000E+02   0.00000E+00   0.00000E+00
    5 CK         - (REAL) STIFFNESS MTRX VALUES
                 GENERAL MATRIX

      ROW     1       NULL ROW
      ROW     2
              1.23300E+06   0.00000E+00   0.00000E+00
      ROW     3
              -1.23300E+06   0.00000E+00   0.00000E+00
    6 CF         - (REAL) FORCE VECTOR VALUES
                 0.00000E+00   0.00000E+00   0.00000E+00

****************************************************************
```

201

```
**************        HELIUM1 /CAG1        **************

HELIUM ACCUMULATOR FOR ET1 MODEL

*****************************************************************
INPUT FOR COMPONENT CAG1. ADIABATIC GAS

    1 NCDF      - NUMBER OF DOF          =            3
    2 CDFLI     - (DOF) DOF NAME
          PROJ1000   BOLT1000   MAGZ1000
    3 GVECT     - (REAL) INITIAL GAS VECTOR
          4.55000E+03   4.00000E+00   1.66700E+00   2.07703E+00
          8.40000E+04
    4 AREA      - (REAL) MATRIX FOR AREA CALC
          GENERAL MATRIX

      ROW    1
          2.00000E+00   2.00000E+00   3.00000E+00

      ROW    2
          1.00000E+00   1.00000E+00   6.66670E-01
    5 GOVDOF    - (DOF) AREA EXIST CRITERIA
          PROJ1000   BOLT1000   BOLT1000
    6 AECV      - (REAL) GOVDOF CRITICAL VALU
          -1.00000E+04  -1.00000E+04   1.00000E+00
    7 PECDOF    - PRESSURE EX CRITERIA=      PROJ1000
    8 PECVAL    - CRITICAL VALUE        =   7.75000E+00
*****************************************************************
```

202

o   TECHNOLOGY LIBRARY CONTAINS

COMPONENT REPRESENTATIONS (C ...)
FORCE ALGORITHMS (F ...)
SOLUTION ALGORITHMS (S ...)

o   DATA LIBRARIES CONTAIN

DATA ASSOCIATED WITH PARTICULAR C ..., F ...
AND IDENTIFIED BY DATA SET NAME
SUPPLIED BY USER

KAMAN
AEROSPACE
CORPORATION

203

MODELING DATA LIBRARY

COMPONENT/FORCE/MODEL/CASE/AIRFOIL

• • •

DYSCO
EXECUTIVE
SYSTEM

AIRFOIL/INDUCED VELOC. TABLES

• • •

EXTERNAL DATA LIBRARY

| CRR2 |
| CFM2 |
| C--- |
| |
| • • • |
| FRAØ |
| FFAØ |
| F--- |
| • • • |
| STH3 |
| STR3 |
| S--- |
| • • • |
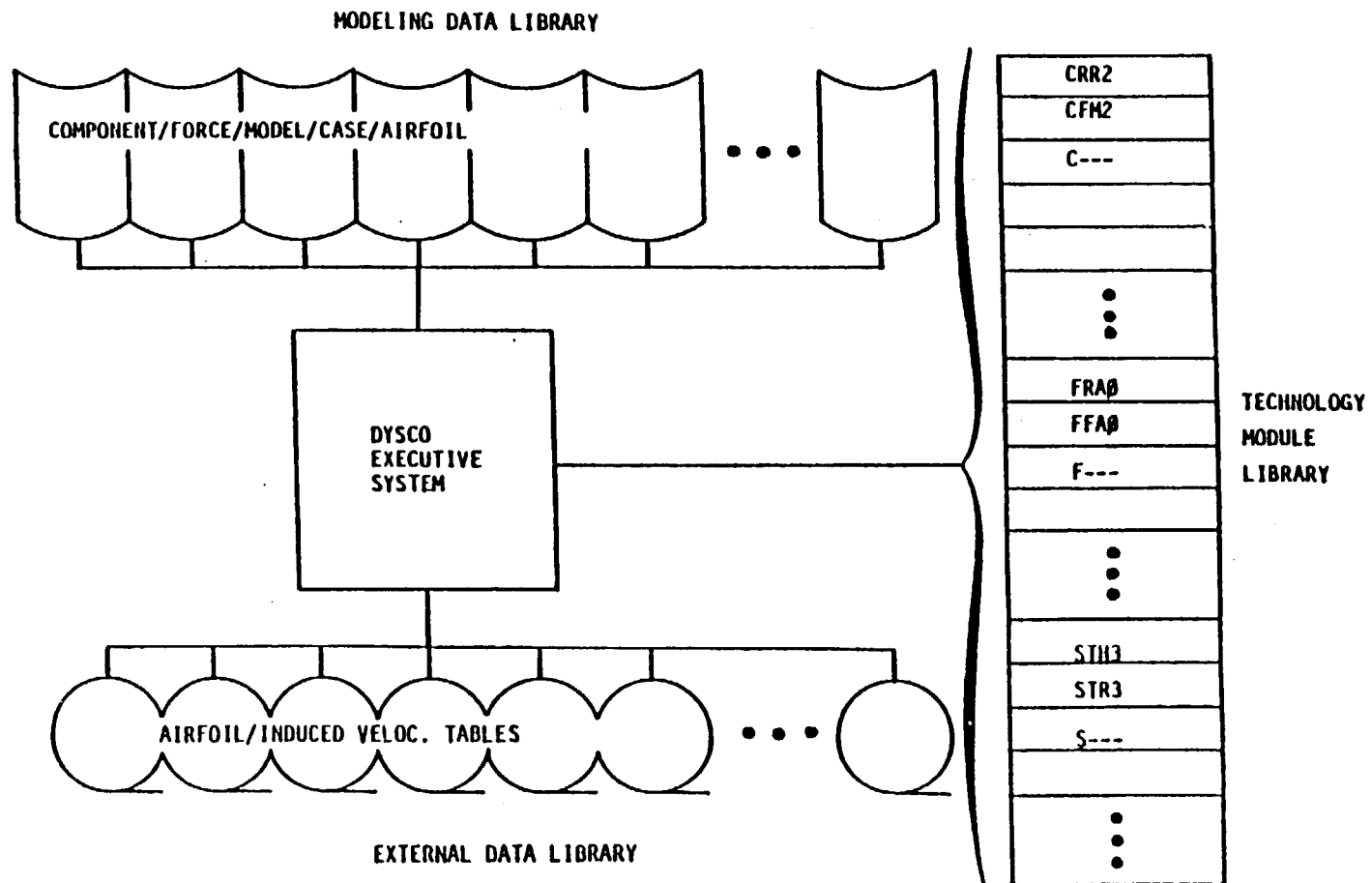
TECHNOLOGY
MODULE
LIBRARY

ANY TECHNOLOGY MODULE MAY BE ADDED TO LIBRARY IF:

COMPONENT, FORCE

    M, C, K, F MAY BE COMPUTED AS FUNCTIONS
    OF LOCAL STATE VECTOR AND TIME BY A
    FORTRAN PROGRAM

SOLUTION

    ALGORITHM MAY BE WRITTEN IN FORTRAN, GIVEN
    SYSTEM M, C, K, F, AS ABOVE

<u>COMMAND</u>

NEW   ALLOWS USER TO MODIFY MODEL
            ALLOWS USER TO CREATE NEW COMPONENT/FORCE INPUT DATA

RERUN   NEW SOLUTION FOR MODEL JUST RUN

RUN   FORMS EQUATIONS OF MODEL AND EXECUTES A SOLUTION

EDIT   ALLOWS USER TO TO MODIFY MODEL AND PERFORM VALIDATED
            EDIT OF INPUT DATA

205

KAMAN
AEROSPACE
CORPORATION

DYSCO ACHIEVES ITS MULTIPURPOSE CAPABILITY BY A
COMPLETE SEPARATION OF ABSTRACT AND SPECIFIC DATA.

SINCE DYSCO DOES NOT TREAT SPECIFIC PHYSICAL COMPO-
NENTS, FORCES, OR SOLUTIONS, IT CAN SOLVE PROBLEMS
INVOLVING ANY COMPONENTS, FORCES, OR SOLUTIONS.

206

THE USER PERCEIVES A "MODEL" MADE UP OF COMPONENTS

HE THEN:

    SELECTS APPROPRIATE REPRESENTATIONS FOR EACH
    COMPONENT FROM THE LIBRARY

    SELECTS APPROPRIATE FORCE ALGORITHMS FOR EACH
    COMPONENT

    SELECTS APPROPRIATE SOLUTION METHODS.

NOTE THAT THE ORIGINAL MODEL FORMULATION REQUIRES AN APPROPRIATE LEVEL
OF ENGINEERING JUDGEMENT.

FOR EACH COMPONENT AND FORCE, THE DATA MUST BE IDENTIFIED AS RESIDING
ON A USER FILE OR THE DATA MUST BE PLACED ON A FILE USING DYSCO.

THE MODEL AND VARIATIONS MAY THEN BE FORMED INTERACTIVELY AND SPECI-
FIED SOLUTIONS MAY BE CARRIED OUT.

KAMAN
AEROSPACE
CORPORATION

DYSCO

ILLUSTRATIVE APPLICATIONS

208
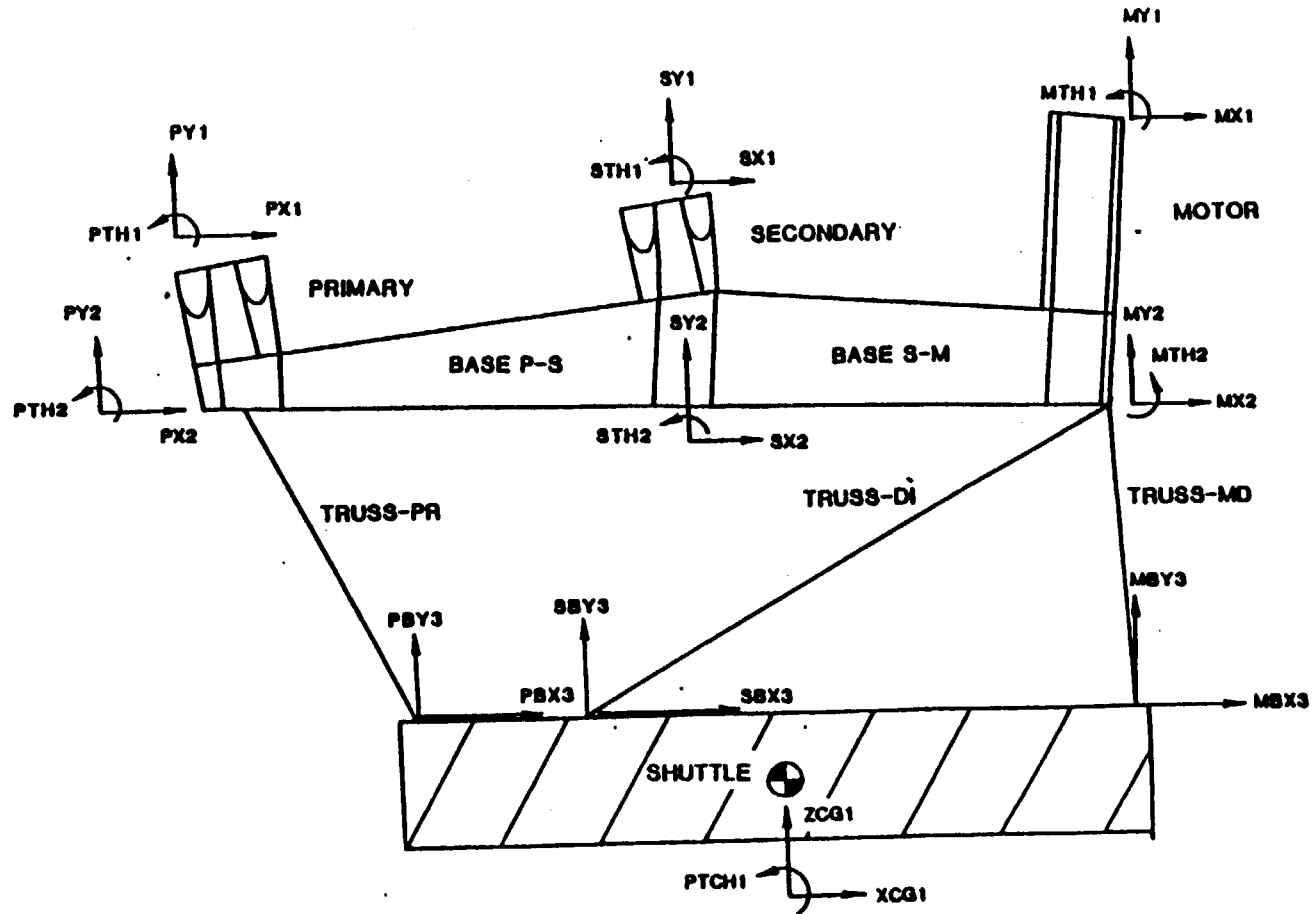
MODEL 1   18 DOF (GROUNDED)
      2-5 VARIOUS BASE SHAKES
      6   21 DOF (3 BASE DOF)

KAMAN
AEROSPACE
CORPORATION

I. MAX TORQUE FOR MOTOR AT MAX RATE .05 R/s - 370 IN #

II. DESIGN TORQUE T = 185 IN #

III. INITIAL TRIAL GAINS

1. DISPLACEMENT GAIN
IF ϕ IS AT A MAX ALLOWABLE ERROR OF 15 μR, LET MOTOR BE
DRIVEN AT MAX SPEED

T = 185 IN # AT 15 μR

$K_2 = 1.233 \times 10^7$ IN #/R OF ϕ

= 0 IN # AT 0

2. VELOCITY GAIN

IF $\dot{\phi}$ IS AT MAX RECESSION OR APPROACH OF .05 R/s, LET
MOTOR BE DRIVEN AT MAX SPEED

T = 185 IN # AT .05 R/s

$K_1 = 3.7 \times 10^3$ IN # SEC/R OF ϕ

= 0 IN # AT 0

210

I. MIRROR AND MOTOR

1. THET9 = θ = TARGET L.O.S. FROM INERTIAL REFERENCE

2. DSTR1 = $\delta*$ = ACTUAL DRIVE ANGLE OF MIRROR

3. MTH1 = θ_M = STRUCTURAL VIBRATION OF THE MOTOR MOUNT

4. $2\delta* + \theta_M$ = OX' (BENT OPTICAL AXIS) FROM INERTIAL REFERENCE

5. $\phi = \theta - \theta_M - 2\delta*$ = OPTICAL MISPOINT

211
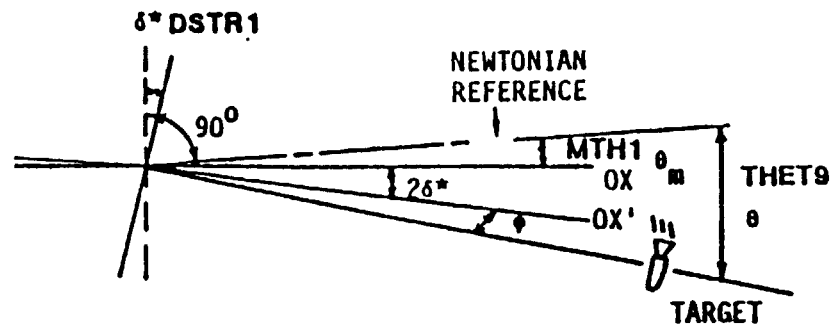
CASE 1 TIME HISTORY   *KAMAN*
STATIONARY TARGET AT ZERO ———————INITIAL MISPOINT DSTR = 15 μrad
INITIAL LINES-OF-SIGHT STATIONARY   CONTROLS:
    Solid = Optical Mispoint of Flat Mirror   Displ. Gain = Max Avail.
                                                       Vel. Gain = Max Avail.

212
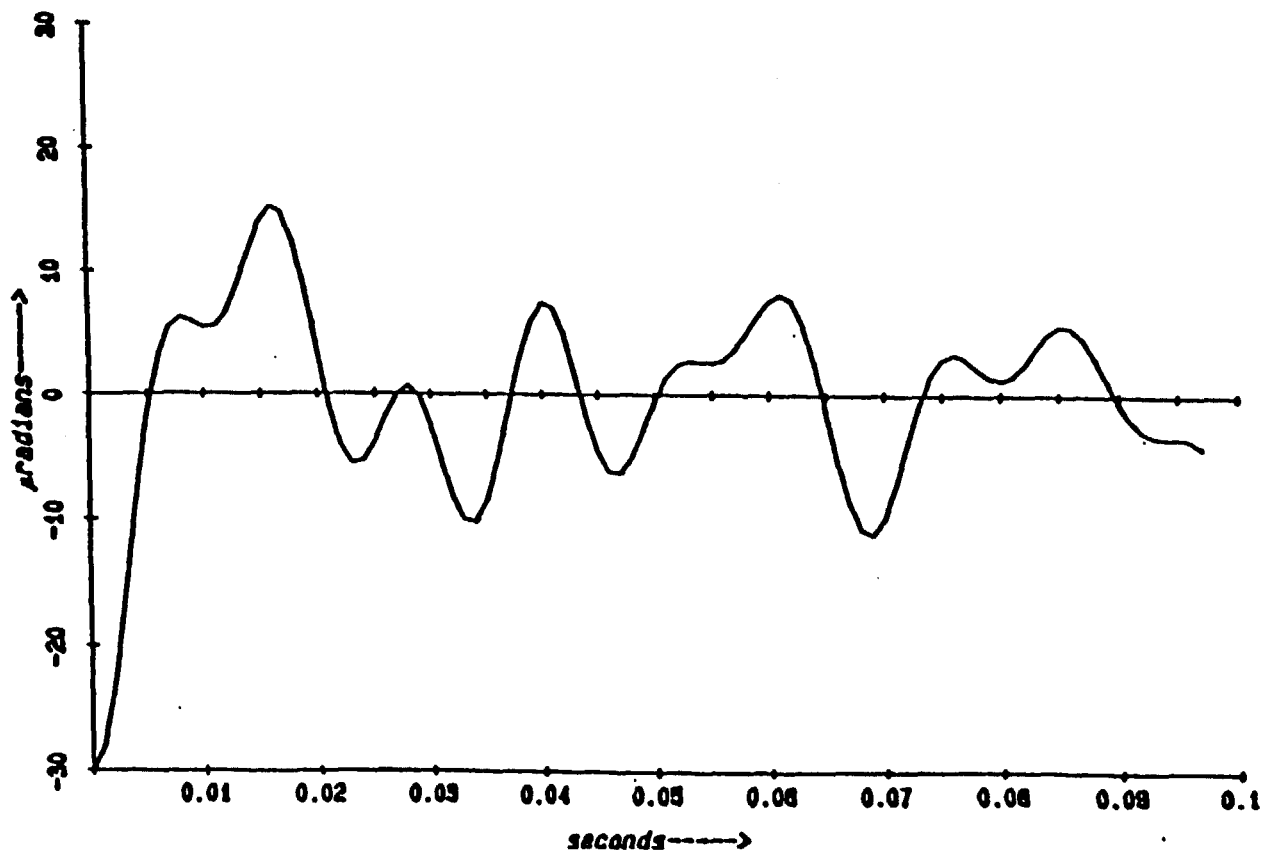
CASE 8 TIME HISTORY   *KAMAN*
ACCELERATING TARGET -------INITIAL MISPOINT.OSTR = 15 ,rad
INITIAL LINES-OF-SIGHT APPROACHING   CONTROLS:
    Solid - Target   Displ. Gain - .65Max
    Dotted - Pointing Axis L.O.S.   Vel. Gain -.81 Max

213

CASE 8 TIME HISTORY

ACCELERATING TARGET -------INITIAL MISPOINT DSTR = 15 μrad

INITIAL LINES-OF-SIGHT APPROACHING

    Solid = Target

    Dotted = Driven Mirror Angle

    Dashed = Motor Mount Response

KAMAN

CONTROLS:

Displ. Gain = .85 Max

Vel. Gain = .81 Max

214

SPRING DETENT

x_1 ← → x_2 ⟿ Ψ_o, P_o, T_o, R, Y

m_1

a

$\ddot{x}_1$

m_2 ← F_2 (PRELOAD)

$\ddot{x}_2$

k_2

x_2 SNUBBER

x_3 SNUBBER

x_2 SNUBBER

m_3 ↓ $\ddot{x}_3$ x_3

ℓ

c

F_3 k_3

(PRELOAD)

b

P_o = 4550 psi

Ψ_o = 4 cu in

Y = 1.667 (Helium)
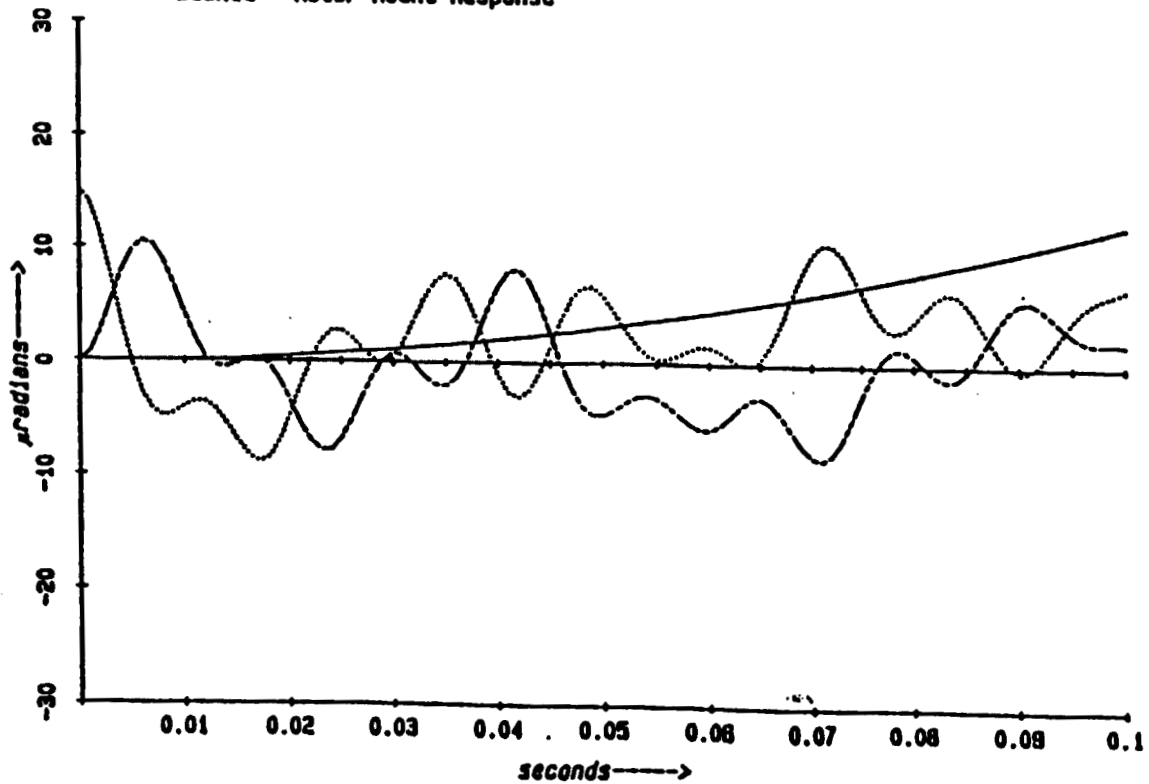
x_3 SNUBBER
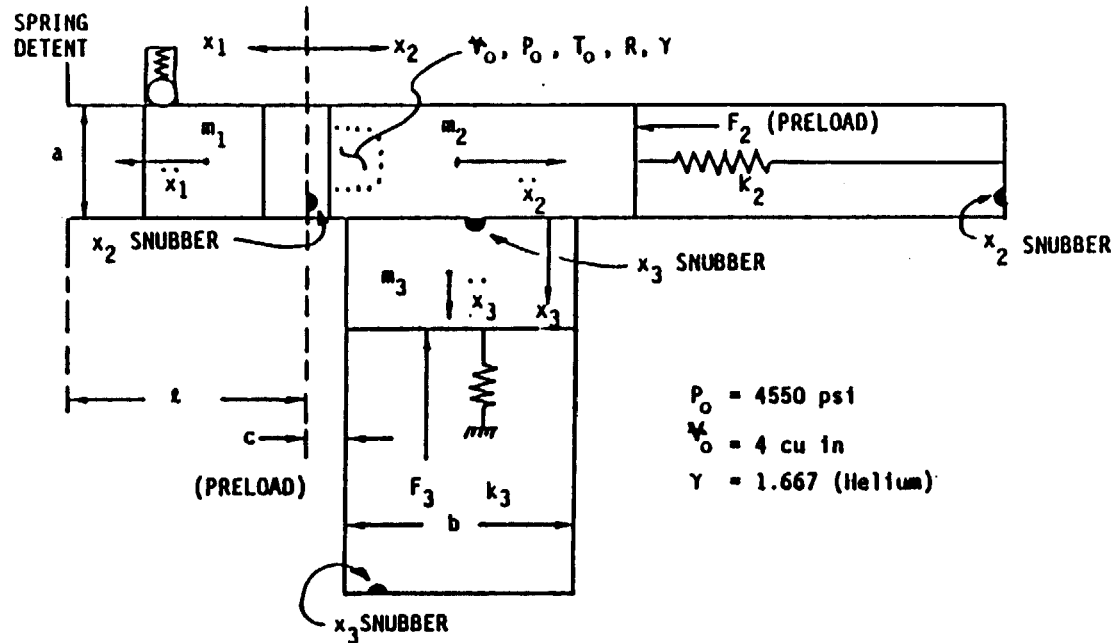
m_1 = PROJECTILE m_2 = BREECH BOLT m_3 = MAGAZINE STACK

EQUATION OF MOTION –

$$\begin{bmatrix} m_1 & & 0 \\ & m_2 & \\ 0 & & m_3 \end{bmatrix} \begin{Bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \\ \ddot{x}_3 \end{Bmatrix} + [\,0\,] \{\dot{x}\} \begin{bmatrix} 0 & & 0 \\ & k_2 & \\ 0 & & k_3 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{Bmatrix} A_1 P(t) \\ A_2 P(t) - F_2 \\ A_3 P(t) - F_3 \end{Bmatrix}$$

A_3 = 0 UNLESS
x_2 > C BEFORE
PRESSURE VENTS

215

216

P-INITIAL = 4550 psi

PRESS VENTS AT t = ____ msec

P-FINAL = 112.1 psi

HELIUM PRESSURE (psi)

TIME (MILLESEC)
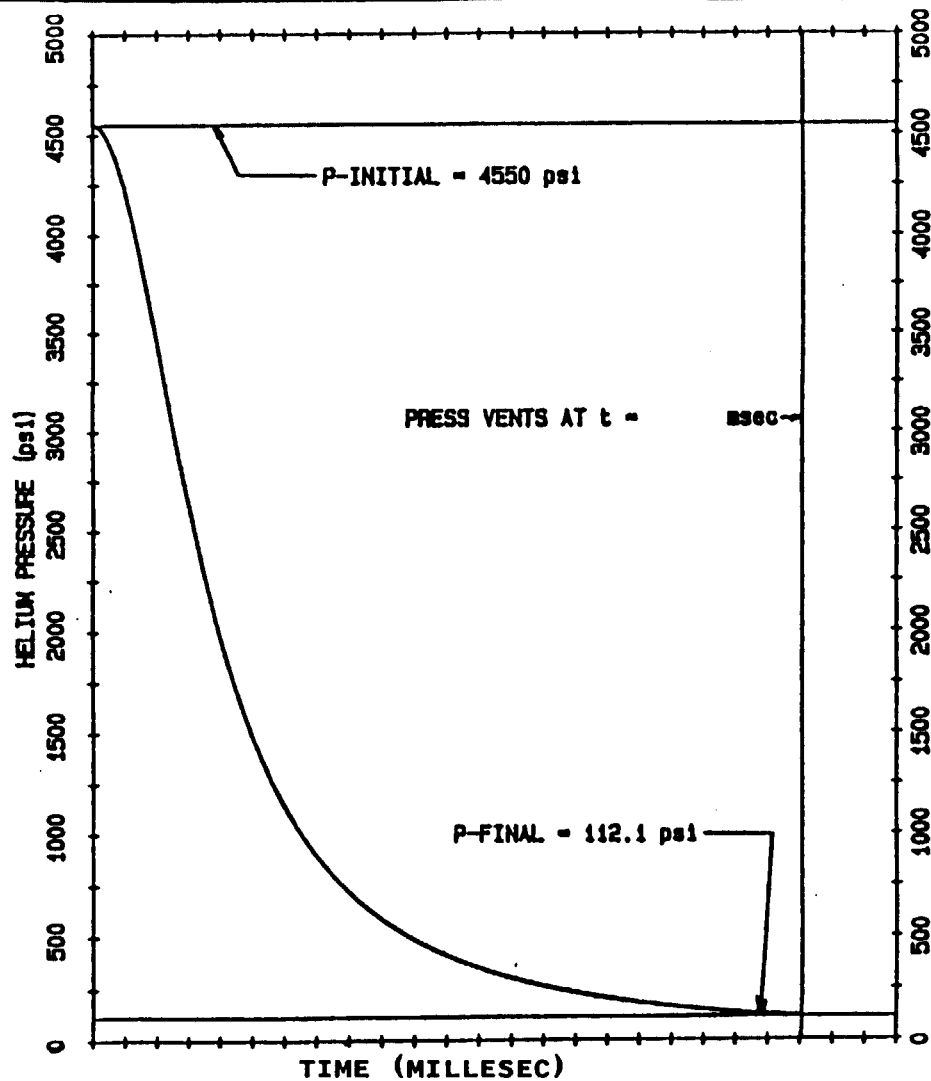
217
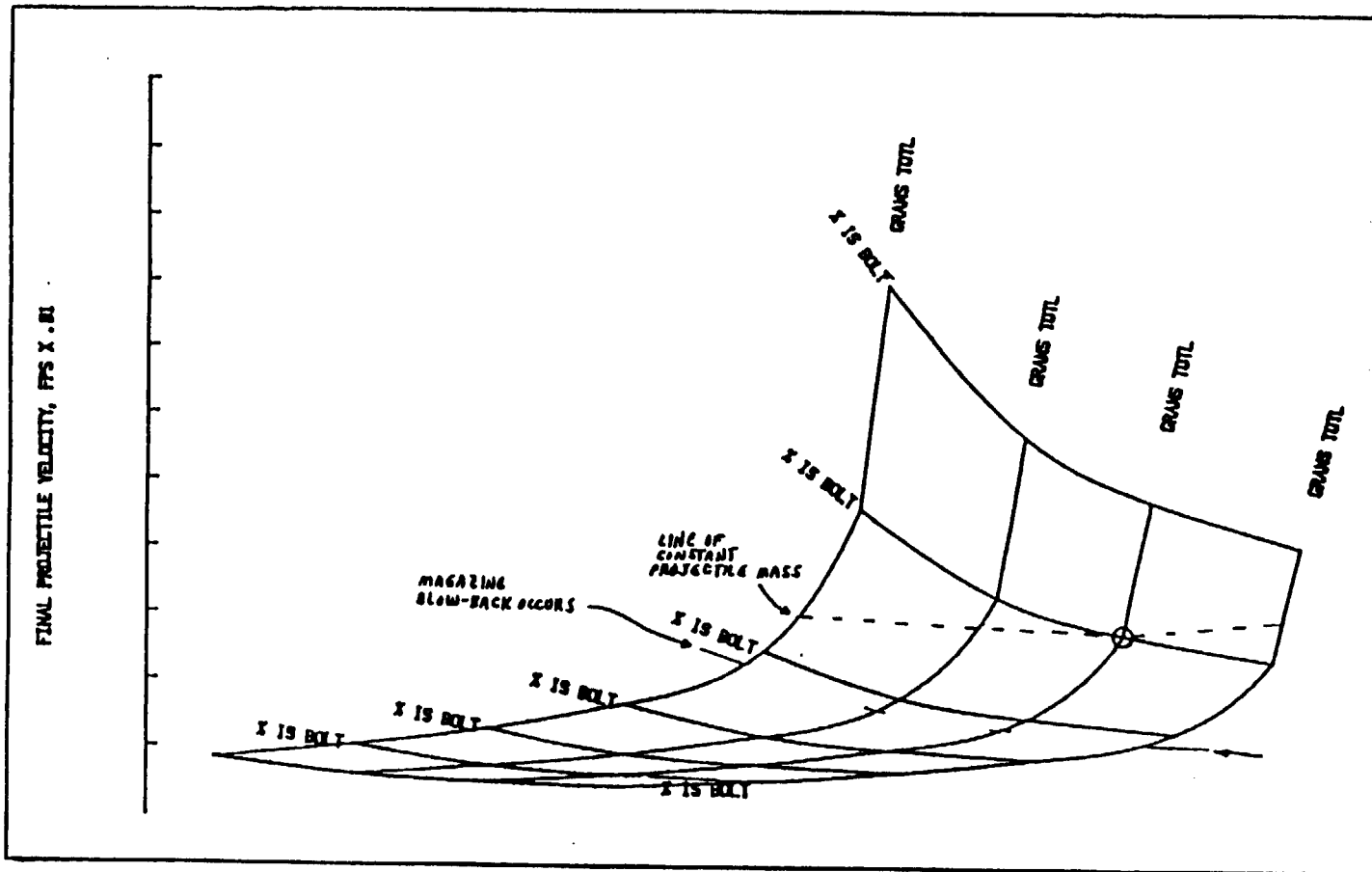
ET PREACCELERATOR, BEHAVIOR VS. TOTAL
MASS AND PERCENT ALLOTTED TO BOLT

MODEL FOR HARMONIC BOLT RETURN AND IMPACT WITH FRESHLY LOADED PROJECTILE
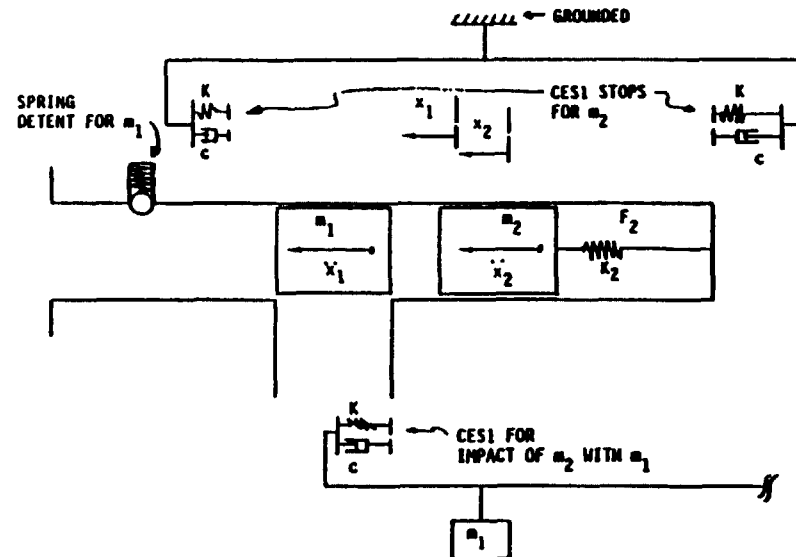
VARIOUS COEFFICIENTS OF RESTITUTION CAN BE MODELLED BY JUDICIOUS CHOICE OF K AND C

**************************** MODEL IMPACT 2 ****************************

IMPACT MODEL PLUS BARREL FIT/MAGAZINE FRICTION AND BALL SPRING DETENT

INDEX	COMP	NO.	DATA SET	FORCE	DATA SET
1	CSF1		BR1DYN	NONE	
2	CES1		BOLT2.0	NONE	
3	CSD1		BALL2	NONE	
4	CDF1		IMPMU	NONE	

****

219
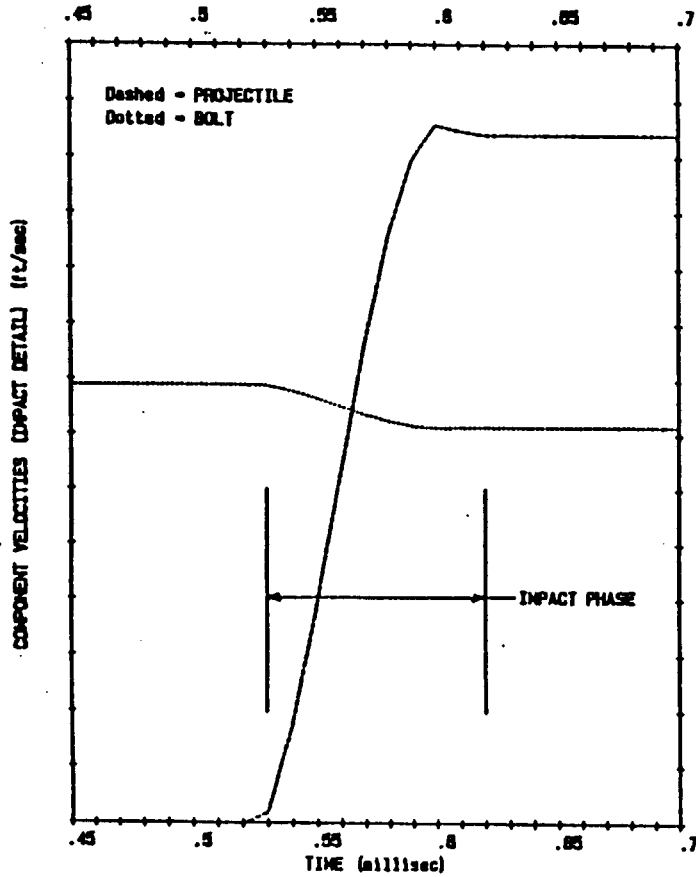
ET PRE-ACCELERATOR
TIME DOMAIN VELOCITIES
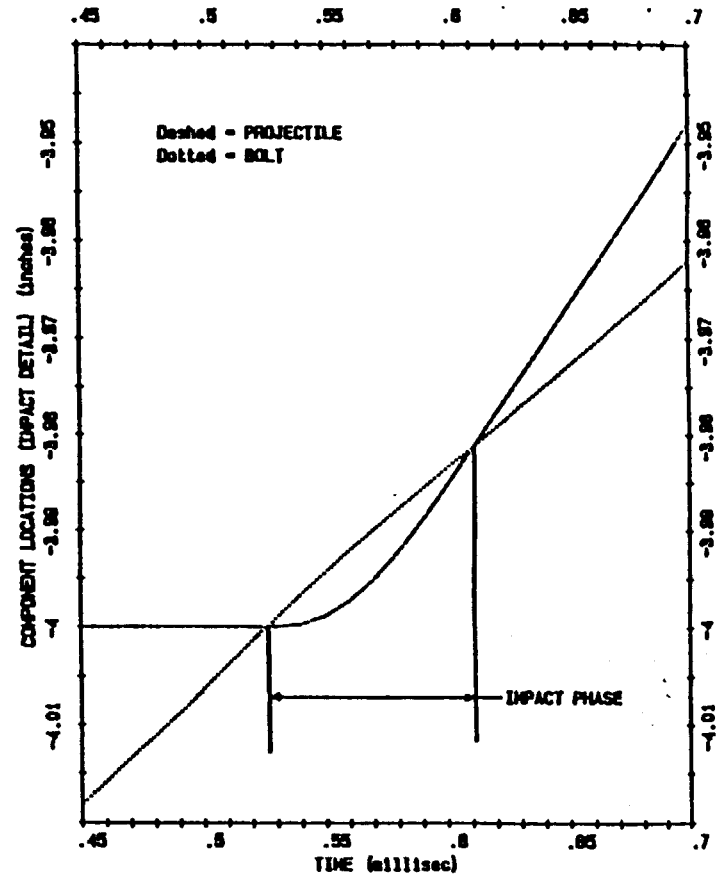
ET PRE-ACCELERATOR
TIME DOMAIN DISPLACEMENTS

MODELING AND CONTROL SYSTEM DESIGN AND ANALYSIS TOOLS FOR FLEXIBLE STRUCTURES

By

Amir A. Anissipour
Russell A. Benson
Edward E. Coleman

The Boeing Company
Seattle, Washington

ABSTRACT

Design and analysis of control systems for flexible structures require accurate math models of flexible structures and software design with the analysis tools capable of handling these models while maintaining numerical accuracy. Since aeroelastic models of flexible structures tend to be large (e.g., 100 states), the availability of tools to handle such large models is crucial. Initial model development is based on aerodynamic mathematical models, wind tunnel data, mathematical structural models, and ground shake test results. Eventually, flight test data are used to update and refine the model. This paper describes Boeing software tools used for the development of control laws of flexible structures.

The Boeing Company has developed a software tool called Modern Control Software Package (MPAC). MPAC provides the environment necessary for linear model development, analysis, and controller design for large models of flexible structures. There are two features of MPAC which are particularly appropriate for use with large models: (1) numerical accuracy and (2) label-driven nature. With the first feature MPAC uses double precision arithmetic for all numerical operations and relies on EISPAC and LINPACK for the numerical foundation. With the second feature, all MPAC model inputs, outputs, and states are referenced by user-defined labels. This feature allows model modification while maintaining the same state, input, and output names. In addition, there is no need for the user to keep track of a model variable's matrix row and column locations.

There is a wide range of model manipulation, analysis, and design features within the numerically robust and flexible environment provided by MPAC. Models can be built or modified using either state space or transfer function representations. Existing models can be combined via parallel, series, and feedback connections; and loops of a closed-loop model may be broken for analysis. Analysis tools available include: eigenvalue/eigenvector, controllability matrix, observability matrix, transfer function generation, frequency response and singular value plots, covariance response to white noise or atmospheric turbulence models, model simulation using step, sinusoidal, random, or user-defined inputs. Control system design tools include: root locus, LQG full state feedback gain matrix computation, LQG full-order estimator design, and robust low order controller (SANDY) design as developed by Dr. Uy-Loi Ly at Stanford.

The existing Boeing Company structural analysis and design software package, ATLAS, has been extended in order to form a state-space model for input to MPAC. The new capability, a module named DYFORM, is an outgrowth of earlier work under a NASA contract for Integrated Application of Active Controls. The structural and theoretical aerodynamic mathematical model originates within ATLAS in exactly the same fashion as for conventional flutter and dynamic loads analyses. The DYFORM module is then used to construct the state-variable model as required by MPAC. Its capabilities include (1) control surfaces and/or gust vector as inputs, (2) sensors and/or loads quantities as outputs, (3) formulation in body-fixed or inertial axes, (4) modification of the theoretical aerodynamics using wind tunnel/flight test data from rigid or flexible-model tests, and (5) use of S-plane rational airloads expressions to formulate the state model including augmented states to represent unsteady aerodynamic effects.

MPAC has been used for yaw damper design (including active flexible mode suppression) of the Boeing 767 and 747 airplanes. The flexible structural models of these planes, as large as 100 states, have been handled by MPAC without loss of numerical accuracy.

The Boeing Company plans for the development of a system identification and parameter estimation (SIPE) software tool. The system identification algorithms employ a multiple stepwise regression technique to determine the structure of the system. The parameter estimation algorithms update the current model using maximum likelihood estimation. The SIPE routines will be compatible with MPAC and RF_DATA (a data correction and reformatting program also developed by Boeing). The SIPE routines will be flexible, allowing the user to select gradient methods, integration algorithms, and Riccati solution algorithms. The MPAC compatible model structure slated for the SIPE package will be applicable to any dynamic system. Aerodynamic, aeroelastic, ground effects, and sensor noise modeling will all be possible.

222

INTRODUCTION TO MPAC:

A Control Law Design Tool Well Suited for

Flexible Structure Applications

Edward E. Coleman

The Boeing Company
Boeing Commercial Airplanes
P.O. Box 3707
Seattle, Washington 98124-2207

223

Tool Requirements for Models of Flexible Structures:

o Large model capacity (more than 100 states)

o Efficient user interface for handling large models

o Numeric robustness

o Model reduction techniques

224

BOEING COMMERCIAL AIRPLANES
GUIDANCE AND CONTROL RESEARCH

MPAC - Multivariable control design and analysis PACkage

o Programmable "calculator" for synthesis, manipulation, and analysis of continuous and discrete linear dynamic system models

o MPAC supports:

- Model development

- Dynamic system analysis

- Controller synthesis

o MPAC was originally developed as a batch process tool. An interactive interface is currently being developed for MPAC to improve its ease of use and efficiency.

225

MPAC Features:

Label Driven Model Format:

- User defined state, input, and output labels of up to 8 characters.

Numeric Robustness:

- Built on Eispac, LINPACK, and ORCALS
- Double precission computation throughout
- Handles models up to 256 elements (states, inputs, and outputs)

Modular Structure:

- Each command is a seperate subroutine
- User need learn only those comands he/she wants to use
- Wide range of available commands
- Provission for customized, user defined commands

BOEING COMMERCIAL AIRPLANES

GUIDANCE AND CONTROL RESEARCH

| Model Input Output |
| Model Properties |
| Model Modification |
| Model Reduction |
| Model Transformation |
| Connection |
| MATLAB |
| EASY |

| Time Response |
| Frequency Response |
| Covariance Response |

| Create Controller |
| Eigenstructure |
| Discrete Design |
| Continuous Design |

| User Commands |

| All commands |
| Last command |

Execution mode : BUILD_CMD_FILE

AUTO-EXEC ?   Yes

AUTO-VIEW ?   Yes

Position MDL file

No.	Comments
↑ 1	-> ▲
2	->
3	->
↓ 4	->
5	->

Task Menu - Command Definition Level

Enter command data / Write command

View results

Print results

Plot results

View CMD file

Cancel (new command)

Up (to File Definition Level)

BOEING COMMERCIAL AIRPLANES
GUIDANCE AND CONTROL RESEARCH

COMPUTE THE ASYMPTOTIC POLES OF FULL STATE FEEDBACK GAIN FOR THE reg

COMPUTE THE ASYMPTOTIC POLES OF KALMAN FILTER GAIN FOR THE reg

COMPUTE THE COMPANION MODAL MODEL OF THE reg, replacement_option

COMPUTE THE CONJUGATE MODAL MODEL OF THE reg, replacement_option

COMPUTE THE CONTROLLABILITY MATRIX OF THE reg

COMPUTE THE COVARIANCE RESPONSE OF THE reg TO turbulence_model AND covariance_print_option

COMPUTE THE COVARIANCE RESPONSE OF THE reg TO WHITE NOISE AND covariance_print_option

COMPUTE THE DISCRETE COVARIANCE RESPONSE OF THE reg TO WHITE NOISE AND covariance_print_option

COMPUTE THE EIGENVALUES OF THE reg

COMPUTE THE EIGENVECTORS OF THE reg

COMPUTE THE FREQUENCY RESPONSE OF THE reg

COMPUTE THE OBSERVABILITY MATRIX OF THE reg

COMPUTE THE RESIDUES OF THE reg

COMPUTE THE ROOT LOCUS OF THE reg

COMPUTE THE SETPOINT MATRICES FOR THE reg

COMPUTE THE SIMILARITY TRANSFORM OF THE STATE MODEL FOR THE reg

COMPUTE THE SINGULAR VALUES OF THE reg

COMPUTE THE TRANSFER FUNCTION(S) OF THE reg

COMPUTE THE TRANSMISSION ZEROS OF THE reg

CONSTRUCT THE DRYDEN TURBULENCE MODEL FOR THE reg

CONSTRUCT THE EQUATIONS OF MOTION FOR THE reg

COPY THE STATE MODEL OF THE reg TO THE destination_reg

CREATE STATE ESTIMATOR FOR THE reg

CREATE controller_type FOR THE reg

CREATE controller_type FOR THE reg WITH MODEL FOLLOWING

DESIGN THE DISCRETE FSF (FULL STATE FEEDBACK) GAIN MATRIX FOR THE reg

DESIGN THE DISCRETE STATE ESTIMATOR (KALMAN FILTER) GAIN MATRIX FOR THE reg

DESIGN THE EMF (EXPLICIT MODEL FOLLOWING) GAIN MATRIX FOR THE reg

DESIGN THE FSF (FULL STATE FEEDBACK) GAIN MATRIX FOR THE reg

DESIGN THE LTS (LINEAR TRACKING SYSTEM) GAIN MATRIX FOR THE reg

DESIGN THE STATE ESTIMATOR (KALMAN FILTER) GAIN MATRIX FOR THE reg

DESIGN ROBUST LOW-ORDER CONTROLLER

Cancel

| Model Input Output |
| Model Properties |
| Model Modification |
| Model Reduction |
| Model Transformation |
| Connection |
| MATLAB |
| EASY |

No.	Comments
1	->
2	->
3	->
4	->
5	->

BOEING COMMERCIAL AIRPLANES
GUIDANCE AND CONTROL RESEARCH

Model Input Output

Model Properties

Model Modification

Model Reduction

Model Transformation

Connection

MATLAB

EASY

DESIGN THE STATE ESTIMATOR (KALMAN FILTER) GAIN MATRIX FOR THE reg

DESIGN ROBUST LOW-ORDER CONTROLLER

DISCONNECT THE CLOSED LOOP OF THE reg

FEEDBACK CONNECTION: a_reg IS CONNECTED TO THE b_reg AND IS PLACED INTO THE destination_reg

FORM THE CLOSED LOOP SYSTEM: CONNECT THE controller_option TO THE cls_reg

FORM THE pim_reg WITH THE IDEAL MODEL (AND PLACE IT INTO THE PLANT + IDEAL MODEL)

FORM THE pse_reg WITH THE estimator_option (AND PLACE IT INTO THE PLANT + STATE ESTIMATOR)

LOAD THE G (FULL STATE FEEDBACK GAIN) MATRIX

LOAD THE S (ESTIMATOR GAIN) MATRIX

LOAD THE STATE MODEL OF THE reg

LOAD transfer_fn_type TRANSFER FUNCTION INTO THE reg

MODIFY THE G MATRIX BY modification_method

MODIFY THE S MATRIX BY modification_method

MODIFY THE STATE MODEL OF THE reg

PARALLEL CONNECTION: a_reg IS CONNECTED TO THE b_reg AND IS PLACED INTO THE destination_reg

PRINT THE G (FULL STATE FEEDBACK GAIN) MATRIX

PRINT THE S (ESTIMATOR GAIN) MATRIX

PRINT THE STATE MODEL OF THE reg

READ THE STATE MODEL FOR THE reg (IN MATLAB FORMAT)

READ THE STATE MODEL FOR THE reg (MPAC FORMAT)

REDUCE THE ORDER OF THE reduce_reg

RESTORE THE ORIGINAL G MATRIX (AFTER MODIFY COMMAND)

RESTORE THE ORIGINAL S MATRIX (AFTER MODIFY COMMAND)

RUN MATLAB

SERIES CONNECTION: a_reg IS CONNECTED TO THE b_reg AND IS PLACED INTO THE destination_reg

SIMULATE THE LINEAR STATE MODEL OF THE reg

SIMULATE THE LINEAR STATE MODEL OF THE reg WITH A lts_controller_reg (LINEAR TRACKING SYSTEM)

TRANSFORM reg USING transform_option

WRITE THE STATE MODEL FOR THE reg (IN MATLAB FORMAT)

WRITE THE STATE MODEL FOR THE reg (IN EASY5 FORMAT)

WRITE THE STATE MODEL FOR THE reg (MPAC FORMAT)

XXX nopt

Cancel

No.	Comments
1	->
2	->
3	->
4	->
5	->

BOEING COMMERCIAL AIRPLANES
GUIDANCE AND CONTROL RESEARCH

COMMAND:
*COMPUTE THE OBSERVABILITY MATRIX OF THE reg

Model Input Output

Model Properties

Model Modification

Model Reduction

Model Transformation

Connection

MATLAB

EASY

Time Response

Frequency Response

Covariance Response

Create Controller

Eigenstructure

Discrete Design

Continuous Design

User Commands

reg

PLANT
REDUCED PLANT
CONTROLLER
REDUCED CONTROLLER
FEEDFORWARD CONTROLLER
LTS CONTROLLER
STATE ESTIMATOR
REDUCED STATE ESTIMATOR
IDEAL MODEL
PLANT + IDEAL MODEL
PLANT + STATE ESTIMATOR
CLOSED LOOP SYSTEM
BROKEN LOOP SYSTEM
MODEL 1
MODEL 2
MODEL 3

Cancel

Execution mode :   BUILD_CMD_FILE

AUTO-EXEC ?   Yes

AUTO-VIEW ?   Yes

Position MDL file

No.	Comments
1	->
2	->
3	->
4	->
5	->

Task Menu - Command Definition Level

Enter command data / Write command
View results
Print results
Plot results
View CMD file
Cancel (new command)
Up (to File Definition Level)

BOEING COMMERCIAL AIRPLANES
GUIDANCE AND CONTROL RESEARCH

This is an example MPAC command file. The output file generated using this command file is given on the following pages.

```
*MPAC READ PLANT
LAT2.MDL

*COMPUTE EIGENVALUES OF PLANT

*DEFINE PLANT
DELETE STATE   PSI
CREATE STATE BETA_INT.dot:  1. BETA
CREATE OUTPUT PHI_CRIT:    1. PHI.dot   5.0 PHI
CREATE OUTPUT BETA_CRIT:   1. BETA.dot  3.2 BETA   4. BETA_INT
END

*DESIGN GAIN MATRIX FOR PLANT
.001
2, 2
'AIL'         1.
'RUD'         2.
'PHI_CRIT'    4.
'BETA_CRIT'   1.
'WLOCUS'    'RHO'  1.,  1.,  1
$    FIRST CUT LATERAL GAIN LOCUS
$         AIL=1. RUD=2.
$       PHI_CRIT=4. BETA_CRIT=1.

*CREATE CONTROLLER FOR PLANT
'NODIRECT'

*FORM PLANT + CONTROLLER

*PRINT CLOSED-LOOP SYSTEM

*COMPUTE EIGENVALUES OF CLOSED-LOOP SYSTEM

*MPAC WRITE CLOSED-LOOP SYSTEM
CLOSED_LOOP.MDL
```

BOEING COMMERCIAL AIRPLANES
GUIDANCE AND CONTROL RESEARCH

MPAC output file example. Output file generated using command file on previous

```
          MPAC RELEASE VERSION 4.00    05 MAY 1987        (CONFIGURATION CONTROL I

*******************************************************
*=====================================================*
*          MPAC INPUT/OUTPUT FILE DESCRIPTION         *
*=====================================================*
*******************************************************

      COMMAND FILE ----------------- example.cmd
      MODEL FILE --------------------
      INPUT BINARY FILE ------------
      OUTPUT BINARY FILE -----------
      MPAC OUTPUT FILE ------------- example.out
      MPAC GGP PLOT FILE ----------- example.ggp
      MPAC USER DATA FILE NO.1 -----
      MPAC USER DATA FILE NO.2 -----
      MPAC USER-DEFINED UBIN FILE --
      TIME OF MPAC JOB EXECUTION --- Tuesday, July 5, 1988   3:50:36 pm (PST)


*********************************************************

******************************************************************************
******************************************************************************
******************************************************************************
******                                                                  ******
******   MODERN CONTROL THEORY ANALYSIS/SYNTHESIS SOFTWARE PACKAGE      ******
******                                                                  ******
******         APOLLO-VERSION: MPAC 4.00 ON APOLLO FORTRAN 8.40         ******
******                                                                  ******
******************************************************************************
******************************************************************************
******************************************************************************

                                                    *************
                                                    * 07/05/88  *
                                                    * 15:50:54  *
                                                    *************

      ***********************************
      ************* TASK 1 *************
      ***********************************
      ******                     ******
      ******    *MPAC READ PLANT   ******
      ******                     ******
      ***********************************
      ************* TASK 1 *************
      ***********************************

      *** MODEL READ FROM FILE:  LAT2.MDL ***

      ELAPSED TIME (SEC):        0.24
```

BOEING COMMERCIAL AIRPLANES
GUIDANCE AND CONTROL RESEARCH

```
**************************************************
****************** TASK 2 ********************
**************************************************
******                                    ******
******    *COMPUTE EIGENVALUES OF PLANT   ******
******                                    ******
**************************************************
****************** TASK 2 ********************
**************************************************
```

SAMPLING TIME : DELTA =   0.0000

_____

```
********  EIGENVALUES OF PLANT  **********************
```

COUNT	REAL PART	IMAG PART	DAMPING	FREQ (RAD/S)	FREQ (HZ)
1	0.0000	0.0000	0.0000	0.0000	0.0000
2	-1.2989E-02	0.0000	1.000	1.2989E-02	2.0673E-03
3	-0.1403	1.676	8.3426E-02	1.682	0.2677
4	-0.1403	-1.676	8.3426E-02	1.682	0.2677
5	-1.946	0.0000	1.000	1.946	0.3097

ELAPSED TIME (SEC):   0.14


```
******************************
********** TASK 3 ************
******************************
******                  ******
******   *DEFINE PLANT   ******
******                  ******
******************************
********** TASK 3 ************
******************************
```

```
******************************
```
DELETED	STATE	PSI				
CREATED	STATE	BETA_INT.:	1.000	BETA		
CREATED	OUTPUT	PHI_CRIT:	1.000	PHI.	5.000	PHI
CREATED	OUTPUT	BETA_CRIT:	1.000	BETA.	3.200	BETA
```
******************************
```

ELAPSED TIME (SEC):   0.10

BOEING COMMERCIAL AIRPLANES
GUIDANCE AND CONTROL RESEARCH

```
*************************************************
****************** TASK 4 ********************
*************************************************
******                               ******
******   *DESIGN GAIN MATRIX FOR PLANT   ******
******                               ******
*************************************************
****************** TASK 4 ********************
*************************************************
```

DESIGN PARAMETERS:

    ALPHA = 1.00000E-03

    CONTROL VARIABLE   CONTROL WEIGHT (R)
        AIL   1.0000
        RUD   2.0000

    CRITERIA VARIABLE   CRITERIA WEIGHT (Q)
        PHI_CRIT   4.0000
        BETA_CRIT   1.0000

```
=========================================
===== STEADY STATE RICCATI SOLUTION =====
=========================================
```

	1	2	3	4	5
1	15.95	-2.100	-6.876	-5.891	9.802
2	-2.100	1.105	4.050	0.7099	-0.3128
3	-6.876	4.050	15.73	2.397	-2.262
4	-5.891	0.7099	2.397	3.222	-3.365
5	9.802	-0.3128	-2.262	-3.365	14.39

    SAMPLING TIME : DELTA =   0.0000

```
********  EIGENVALUES OF A + B*G  *********************
********  OPTIMAL CL EIGENVALUES     ******************
```

COUNT	REAL PART	IMAG PART	DAMPING	FREQ (RAD/S)	FREQ (HZ)
1	-0.9492	0.0000	1.000	0.9492	0.1511
2	-1.574	1.453	0.7346	2.142	0.3409
3	-1.574	-1.453	0.7346	2.142	0.3409
4	-4.442	2.532	0.8688	5.113	0.8138
5	-4.442	-2.532	0.8688	5.113	0.8138

```
================================
===== FEEDBACK GAIN MATRIX =====
================================
```

	BETA	P	PHI	R	BETA_INT
AIL	5.470	-2.605	-9.524	-2.000	1.093
RUD	-3.065	-0.5127	-2.059	2.056	-2.722

ELAPSED TIME (SEC) :   1.17

```
*************************************************
****************** TASK 5 ******************
*************************************************
******                                    ******
******    *CREATE CONTROLLER FOR PLANT    ******
******                                    ******
*************************************************
****************** TASK 5 ******************
*************************************************


***********************************************************
***** FULL STATE FEEDBACK CONTROLLER *****
***** NO MODEL FOLLOWING *****
***** DIRECT F.B. STATES TO PLANT *****
BETA          P              PHI          R              BETA_INT
***********************************************************

ELAPSED TIME (SEC):      0.50


*********************************************
**************** TASK 6 ****************
*********************************************
******                              ******
******    *FORM PLANT + CONTROLLER    ******
******                              ******
*********************************************
**************** TASK 6 ****************
*********************************************


***********************************************************
***** FULL STATE FEEDBACK CONTROLLER *****
***** NO MODEL FOLLOWING *****
***********************************************************

ELAPSED TIME (SEC):      2.16
```

BOEING COMMERCIAL AIRPLANES
GUIDANCE AND CONTROL RESEARCH

```
************************************************
*************** TASK 7 ******************
************************************************
******                              ******
******    *PRINT CLOSED-LOOP SYSTEM    ******
******                              ******
************************************************
*************** TASK 7 ******************
************************************************


****************************************************************
   ***** CLOSED LOOP SYSTEM *****
****************************************************************

   SAMPLING TIME : DELTA =    0.0000
   _____
```

```
=============
===== A =====
=============
            BETA        P          PHI         R          BETA_INT
          ------------------------------------------------------------
BETA       -0.2401     5.9648E-02 -4.8458E-02 -0.9132     -0.1075
P          -4.579      -8.932      -25.99      2.8649E-02  -3.112
PHI        0.0000       1.000       0.0000     8.7813E-02   0.0000
R          7.526        0.4025      2.215      -3.809       4.611
BETA_INT   1.000        0.0000      0.0000      0.0000      0.0000


=============
===== B =====
=============
            AIL        RUD
          --------------------------
BETA       2.5320E-03  4.0504E-02
P          2.284        2.060
PHI        0.0000       0.0000
R          0.1228      -1.645
BETA_INT   0.0000       0.0000


=============
===== C =====
=============
            BETA        P          PHI         R          BETA_INT
          ------------------------------------------------------------
PHI_CRIT   0.0000       1.000       5.000      8.7813E-02   0.0000
BETA_CRIT  2.960       5.9648E-02 -4.8458E-02 -0.9132       3.892
AIL+       5.470       -2.605      -9.524      -2.000        1.093
RUD+      -3.065       -0.5127     -2.059       2.056       -2.722
AIL=       5.470       -2.605      -9.524      -2.000        1.093
RUD=      -3.065       -0.5127     -2.059       2.056       -2.722


=============
===== D =====
=============
            AIL        RUD
          --------------------------
PHI_CRIT   0.0000       0.0000
BETA_CRIT  2.5320E-03  4.0504E-02
AIL+       0.0000       0.0000
RUD+       0.0000       0.0000
AIL=       1.000        0.0000
RUD=       0.0000       1.000

ELAPSED TIME (SEC):     0.12
```

237

```
******************************************************************
************************** TASK 8 **************************
******************************************************************
******                                                     ******
******   *COMPUTE EIGENVALUES OF CLOSED-LOOP SYSTEM   ******
******                                                     ******
******************************************************************
************************** TASK 8 **************************
******************************************************************
```

SAMPLING TIME : DELTA =   0.0000

```
********  EIGENVALUES OF CLOSED-LOOP SYSTEM  *********
```

COUNT	REAL PART	IMAG PART	DAMPING	FREQ (RAD/S)	FREQ (HZ)
1	-0.9492	0.0000	1.000	0.9492	0.1511
2	-1.574	1.453	0.7346	2.142	0.3409
3	-1.574	-1.453	0.7346	2.142	0.3409
4	-4.442	2.532	0.8688	5.113	0.8138
5	-4.442	-2.532	0.8688	5.113	0.8138

ELAPSED TIME (SEC):   0.18

```
*****************************************************
***************** TASK 9 *****************
*****************************************************
******                                        ******
******   *MPAC WRITE CLOSED-LOOP SYSTEM   ******
******                                        ******
*****************************************************
***************** TASK 9 *****************
*****************************************************
```

*** MODEL WRITTEN TO FILE: CLOSED_LOOP.MDL ***

ELAPSED TIME (SEC):   0.32

TOTAL JOB ELAPSED TIME (SEC):   10.24

238

PROPOSED SIPE TOOLBOX

A Graphic/Engineering Software Concept

for Modeling

Amir A. Anissipour

The Boeing Company

Boeing Commercial Airplanes
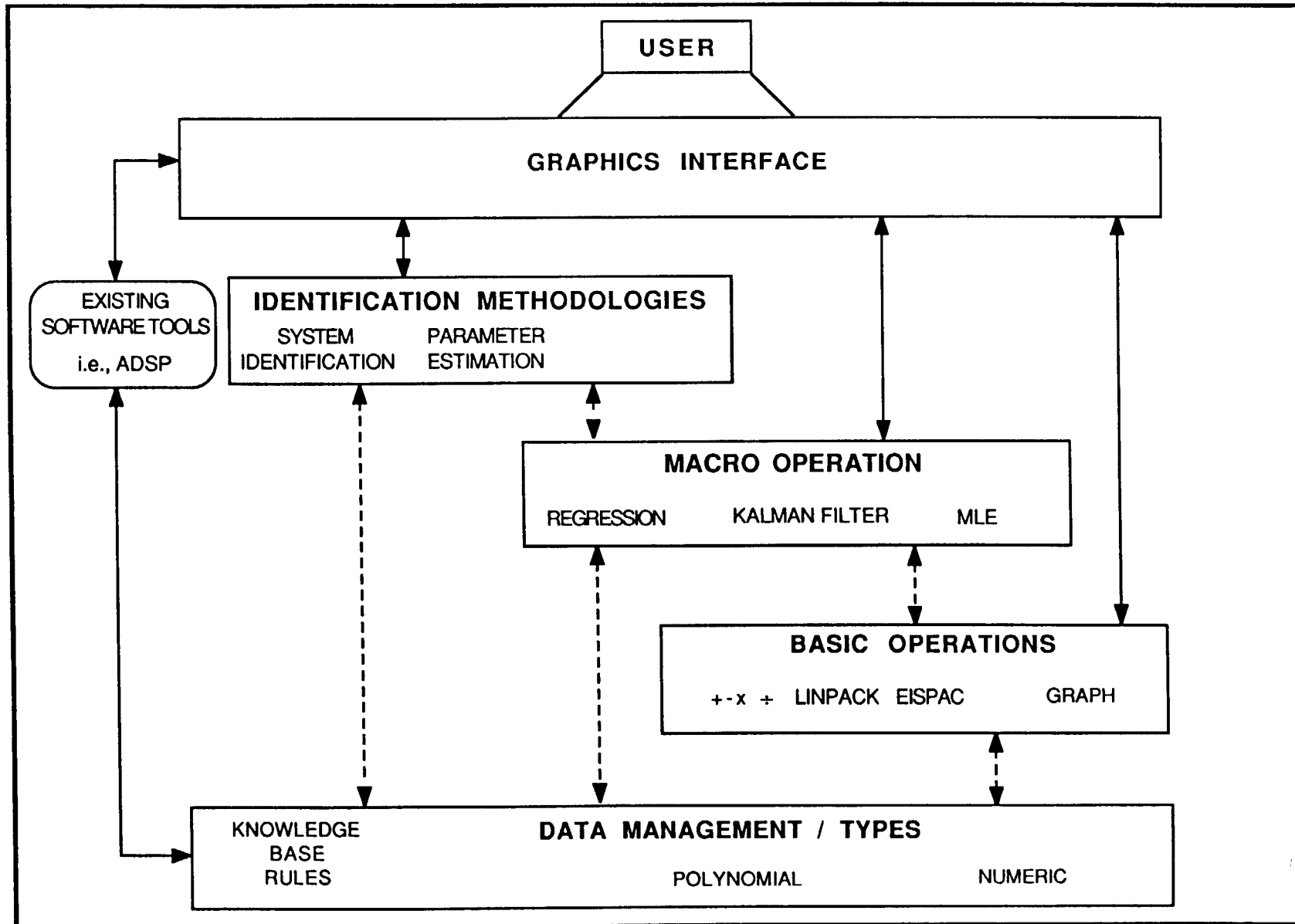
P.O. Box 3707

Seattle, Washington 98124-2207

239

CONCEPT OBJECTIVES

CREATE A SUPERIOR COMPUTATIONAL FRAMEWORK FOR MODELING

- SUPPORT LINEAR AND NON-LINEAR SYSTEM MODELS

- HANDEL HIGH-ORDER MODELS

- BASIS FOR FUTURE ENHANCEMENTS

- USER DEFIEND ANALYSIS

- CONSOLIDATE NASA DRYDEN AND NASA LARC METHODS

- INTERACTIVE GRAPHICS ENVIROMENT FOR HIGH PRODUCTIVITY AND VISIBILITY

SIPE TOOLBOX ARCHITECTURE

USER

GRAPHICS INTERFACE

EXISTING
SOFTWARE TOOLS

i.e., ADSP

IDENTIFICATION METHODOLOGIES

SYSTEM   PARAMETER
IDENTIFICATION   ESTIMATION

MACRO OPERATION

REGRESSION   KALMAN FILTER   MLE

BASIC OPERATIONS

+ - x ÷   LINPACK EISPAC   GRAPH

KNOWLEDGE
BASE
RULES

DATA MANAGEMENT / TYPES

POLYNOMIAL   NUMERIC

241

LUMPED MASS FORMULATIONS

FOR

MODELING FLEXIBLE BODY SYSTEMS

Rajiv Rampalli

Director, Product Technology Group
Mechanical Dynamics, Inc.
3055, Plymouth Road
Ann Arbor, Mi 48105-3203

ABSTRACT

This paper presents the efforts of Mechanical Dynamics, Inc. in obtaining a general formulation for flexible bodies in a multibody setting. The efforts being supported by MDI, both in house and externally are summarized. The feasibility of using lumped mass approaches to modeling flexibility in a multibody dynamics context is examined. The kinematics and kinetics for a simple system consisting of two rigid bodies connected together by an elastic beam are developed in detail. Accuracy, efficiency and ease of use using this approach are some of the issues that are then looked at.

The formulation is then generalized to a "superelement" containing several nodes and connecting several bodies. Superelement kinematics and kinetics equations are developed.

The feasibility and effectiveness of the method is illustrated by the use of some examples illustrating phenomena common in the context of spacecraft motions.

SCOPE OF THE PRESENTATION

- **PROFILE OF MECHANICAL DYNAMICS, INC.**

- **MDI EFFORTS TO MODEL FLEXIBILITY**

- **LUMPED MASS APPROACHES TO FLEXIBILITY**

- **EXAMPLES**

PROFILE OF MECHANICAL DYNAMICS, INC.

- COMPANY BACKGROUND

    - HISTORY

    - PRODUCTS & SERVICES

    - CUSTOMERS

- CURRENT PRODUCTS

    - ADAMS

    - ADAMS / MODAL

    - POST PROCESSORS

- SERVICES

    - CONSULTING

    - TRAINING

    - HOTLINE

- AVAILIBILITY OF PRODUCTS

245

MDI EFFORTS IN FLEXIBILITY

- **INTERNAL R&D**

    - **LUMPED MASS APPROACHES TO FLEXIBILITY**

- **EXTERNAL R&D**

    - **UNIVERSITY FUNDED RESEARCH IN MODAL APPROACHES**

- **INTERFACE TO FEA PROGRAMS**

    - **NASTRAN**

    - **ANSYS**

246

LUMPED MASS APPROACH TO FLEXIBILITY

PART 1
(BASE PART)

BEAM

PART 2

1

2'

2
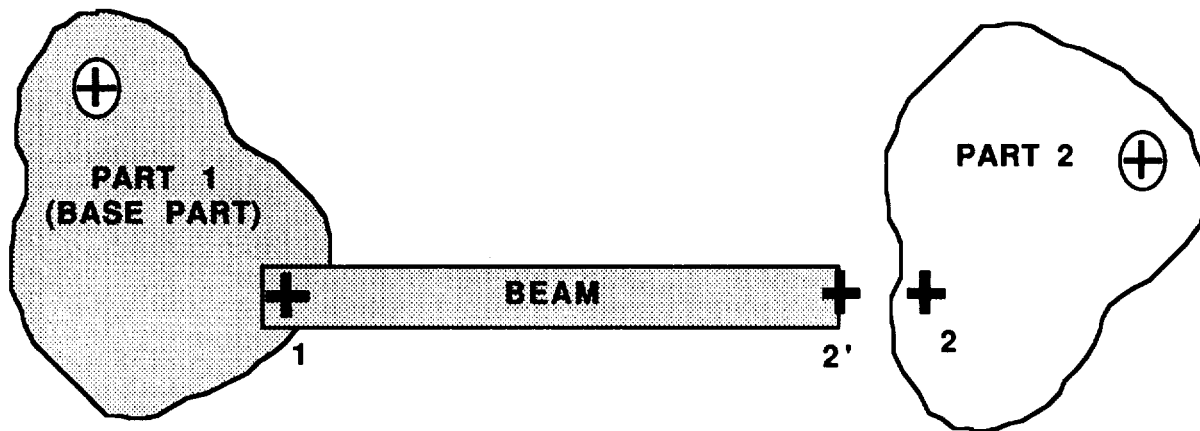
EXPLODED VIEW OF INITIAL CONFIGURATION
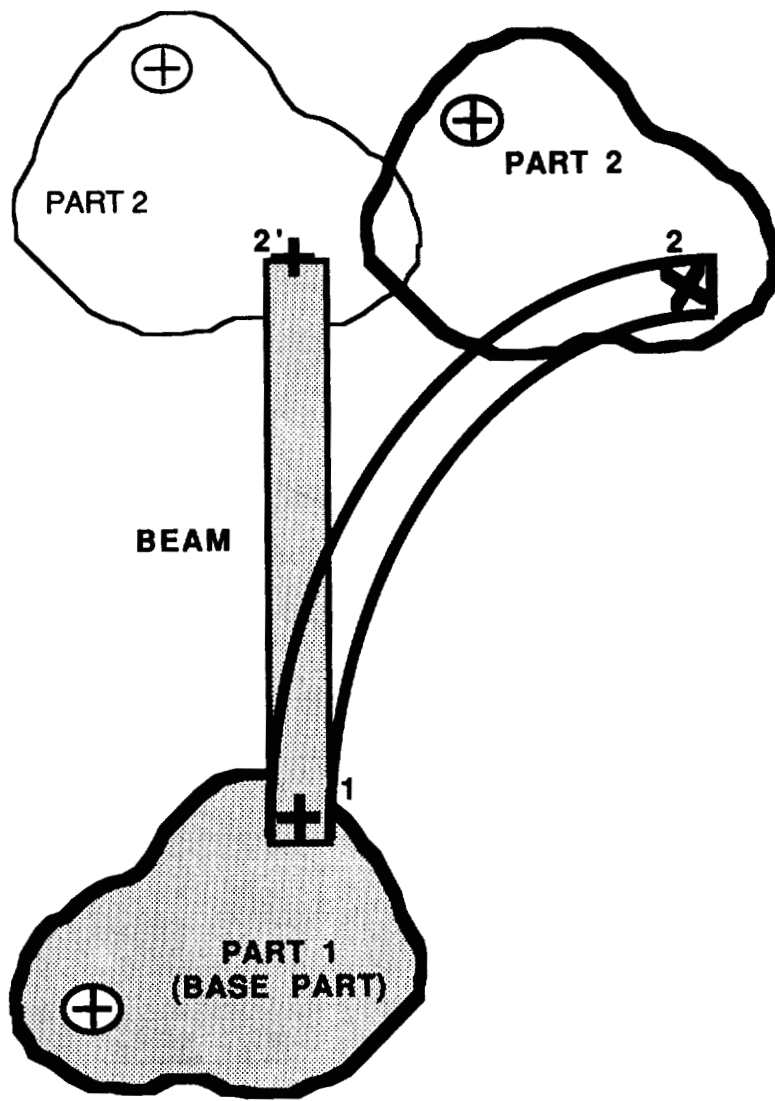
BEAM DEFORMATION DURING MOTION

249

TRANSLATIONAL DISPLACEMENT COMPUTATION
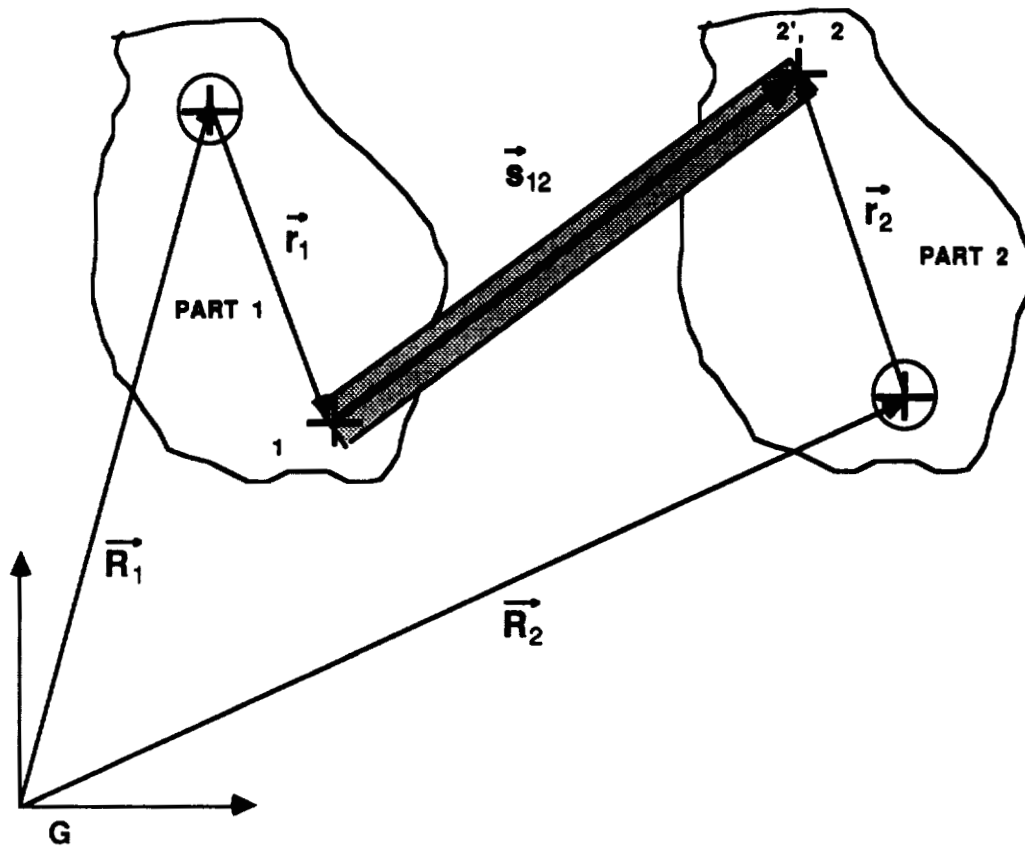
$$\vec{S}_{12} = \vec{R}_2 + \vec{r}_2 - \vec{R}_1 - \vec{r}_1$$

$$_1S_{12} = A^{1G} \left[\ R_2 + A^{2G} r_2 - R_1 - A^{1G} r_1 \ \right]$$

$$_1\Delta_{2'2} = {}_1S_{12} - {}_1L$$

$$_1L \ = \ \{ L \ \ 0 \ \ 0 \ \}^T$$

250

ANGULAR DISPLACEMENT COMPUTATION

SPACE 1-2-3 ANGLES ARE USED FOR MEASURING ANGLES

$$
\mathbf{A}^{2'2} = \begin{bmatrix} C_2\,C_3 & S_1\,S_2\,C_3 - S_3\,C_1 & C_1\,S_2\,C_3 + S_3\,S_1 \\ C_2\,S_3 & S_1\,S_2\,S_3 - C_3\,S_1 & C_1\,S_2\,S_3 + C_3\,S_1 \\ -S_2 & S_1\,C_2 & C_1\,C_2 \end{bmatrix}
$$

$$\beta_2 = \text{SIN}^{-1}(-a_{31})$$

IF $\quad \beta_2 \neq \prod/2 \quad$ THEN

$$\beta_1 = \text{ARCTAN2}(a_{32}, a_{33})$$

$$\beta_3 = \text{ARCTAN2}(a_{12} + a_{13}, a_{13} - a_{22}) - \beta_1$$

ELSE IF $\quad \beta_2 = \prod/2 \quad$ THEN

$$\beta_3 = \text{ARCTAN2}(a_{21}, a_{11})$$

$$\beta_3 = \text{ARCTAN2}(a_{12}, a_{13}) + \beta_3$$

$$\underline{\beta}_{2'2} = [\ \beta_1 \quad \beta_2 \quad \beta_3\ ]^T$$

VELOCITY COMPUTATION
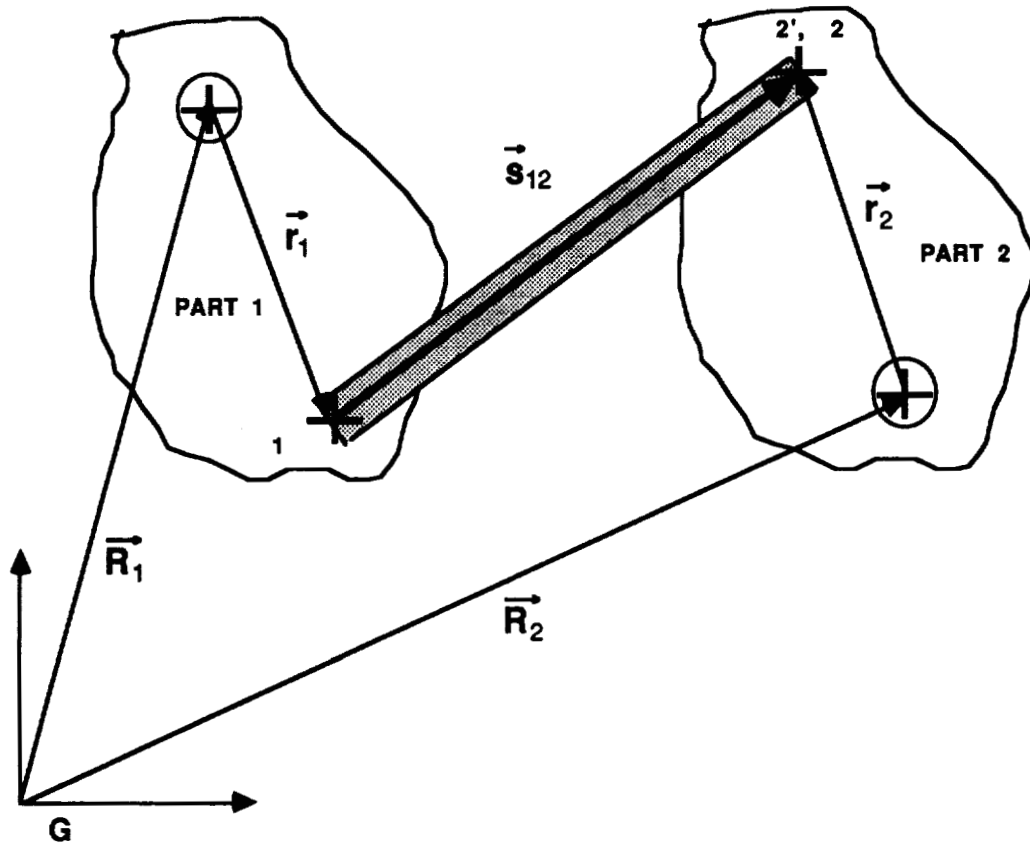
$$^{(1)}\vec{V}_{2'2} = {}^{(G)}\vec{V}_{21} - \vec{W}_1 \times \vec{S}_{21}$$

$$_1\underline{V}_{2'2} = A^{1J} \left[A^{1G}(\underline{R}_I - \underline{R}_J) - A^{JI}\underline{\omega}\tilde{~}_I\underline{r}_1 \right.$$
$$\left. - \underline{\omega}\tilde{~}_{-J}(A^{JI}\underline{r}_1 + A^{JG}\underline{R}_1 - A^{JG}\underline{R}_1) \right]$$

$$\vec{\omega}_{12} = \vec{\omega}_2 - \vec{\omega}_1$$

$$_1\underline{\omega}_{12} = A^{1J} \left[A^{JI}\underline{\omega}_I - \underline{\omega}_J \right]$$

252

FORCE COMPUTATION

FORCES AT ORIGIN OF COORDINATE SYSTEM ON REF. FRAME 1

$$_1\underline{F}_2 = \quad - [\ \underline{K}_{11}\ _1\underline{\Delta}_{2'2} + \underline{K}_{12}\ \underline{\beta}_{12}\ ]$$
$$\quad\quad - [\ \underline{C}_{11}\ _1\underline{V}_{2'2} + \underline{C}_{12}\ _2\underline{\omega}_{12}\ ]$$

$$_1\underline{T}_2 = \quad - [\ \underline{K}_{21}\ _1\underline{\Delta}_{2'2} + \underline{K}_{22}\ \underline{\beta}_{12}\ ]$$
$$\quad\quad - [\ \underline{C}_{21}\ _1\underline{V}_{2'2} + \underline{C}_{22}\ _1\underline{\omega}_{12}\ ]$$

$\underline{K}$ IS THE STANDARD MATRIX FOUND IN ANY STRUCTURAL ANALYSIS TEXT

FORCES AT ORIGIN OF COORDINATE SYSTEM ON REF. FRAME 2

SINCE THE BEAM IS MASSLESS, APPLYING LAWS OF EQUILIBRIUM :

$$\vec{F}_1 + \vec{F}_2 = \vec{0}$$

$$_1\underline{F}_1 = -\ _1\underline{F}_2$$

$$\vec{T}_1 + \vec{T}_2 + \vec{S}_{12} \times \vec{F}_2 = \vec{0}$$

$$_1\underline{T}_1 = - [\ _1\underline{T}_2 + _1\underline{\tilde{S}}_{12}\ _1\underline{F}_2\ ]$$

ACCURACY OF METHOD

- DIRECTLY RELATED TO DEGREE OF DISCRETIZATION

- METHOD DOES **NOT** YIELD **WRONG** ANSWERS

- DEGREE OF DISCRETIZATION DEPENDENT ON FREQUENCY CONTENT DESIRED. **ADAMS/MODAL** WILL COMPUTE EIGENVALUES AND EIGENVECTORS FOR ANY ADAMS MODEL. CAN ANIMATE LINEAR MODEL USING SELECTED SET OF MODE SHAPES AND FREQUENCIES.

R.R.Ryan (313) 936-2922
University of Michigan

FLEXIBLE BODY DYNAMICS
Benchmark Problem #1: Uniform Cantilever Beam Spin-up

The inset of Fig. 1 shows a uniform, homogeneous, cantilever beam supported by a circular hub of radius r. At time $t = 0$, the system is at rest in a Newtonian reference frame and the beam is undeformed. Subsequent to this initial time, the hub is made to rotate about a vertical axis $X - X$, passing through the center of the hub, in such a way that Ω, the angular speed of the hub, is given by

$$\Omega(t) = \begin{cases} (2/5)\left[t - (7.5/\pi)\sin(\pi t/7.5)\right] & \text{rad/sec} \quad 0 \leq t \leq 15 \text{ sec} \\ 6 \text{ rad/sec} & t > 15 \text{ sec} \end{cases}$$

which represents a smooth transition from zero hub motion to a constant angular speed of 6 rad/sec. The beam has a length L, Young's modulus E, shear modulus G, mass per unit length ρ, and a circular cross-section of area A and area moment of inertia I.

The solid line in the figure below shows the time history of the displacement of the beam tip, in the plane of rotation, relative to a line fixed in the hub and originally parallel to the centroidal axis of the beam. This result was obtained using the theory and algorithm presented in Refs.[1] and [2] with three assumed modes and the following parameter values

$$r = 0\,\text{m} \qquad \rho = 1.2\,\text{kg/m}$$
$$L = 10\,\text{m} \qquad A = 4 \times 10^{-4}\,\text{m}^2$$
$$E = 7 \times 10^{10}\,\text{N/m}^2 \qquad G = 3 \times 10^{10}\,\text{N/m}^2 \qquad I = 2 \times 10^{-7}\,\text{m}^4$$

All external forces were neglected and the assumed modal functions were chosen to be equal to the first three eigenfunctions of an identical uniform cantilever beam with its root fixed in a Newtonian reference frame. The numerical integration was carried out using a $4^{th} - 5^{th}$ order, variable step-size, Runge-Kutta-Merson method with a print step and initial time step of .03 seconds and an error tolerance of 1×10^{-6}. The dashed line result was produced with an algorithm based on the assumed-mode formulation utilized in most flexible multibody programs. This result was verified by Fidelis Eke [(818) 354-2916] at Jet Propulsion Labs using DISCOS.
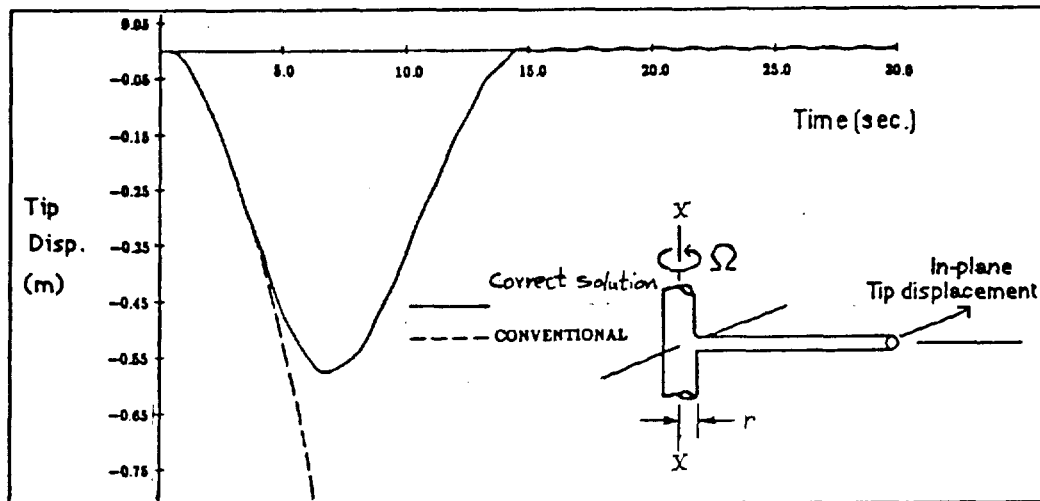

Fig. 1 Spin-up of Homogeneous Uniform Cantilever Beam

[1] Ryan, R.R., "Flexibility Modeling Methods in Multibody Dynamics," Ph.D. dissertation, Stanford University, Available from University Microfilms, Ann Arbor, Michigan, 1986.
[2] Kane, T.R., Ryan, R.R., and Banerjee, A.K., "Dynamics of a Cantilever Beam Attached to a Moving Base," Journal of Guidance, Control, and Dynamics, Vol.10, No.2, March-April, 1987.
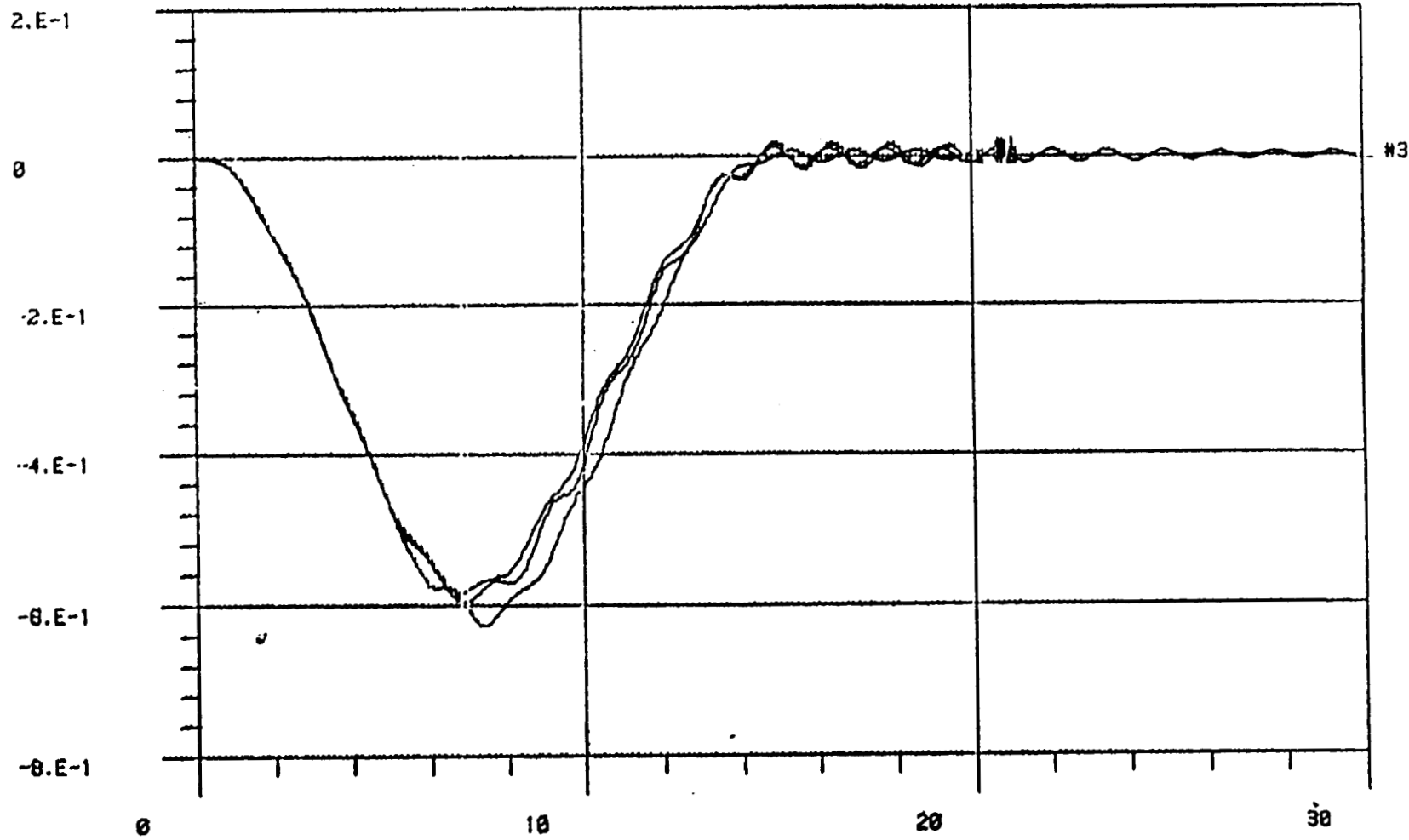
UNIFORM BEAM SPIN-UP PROBLEM
BEAM TIP DEFLECTION VS. TIME
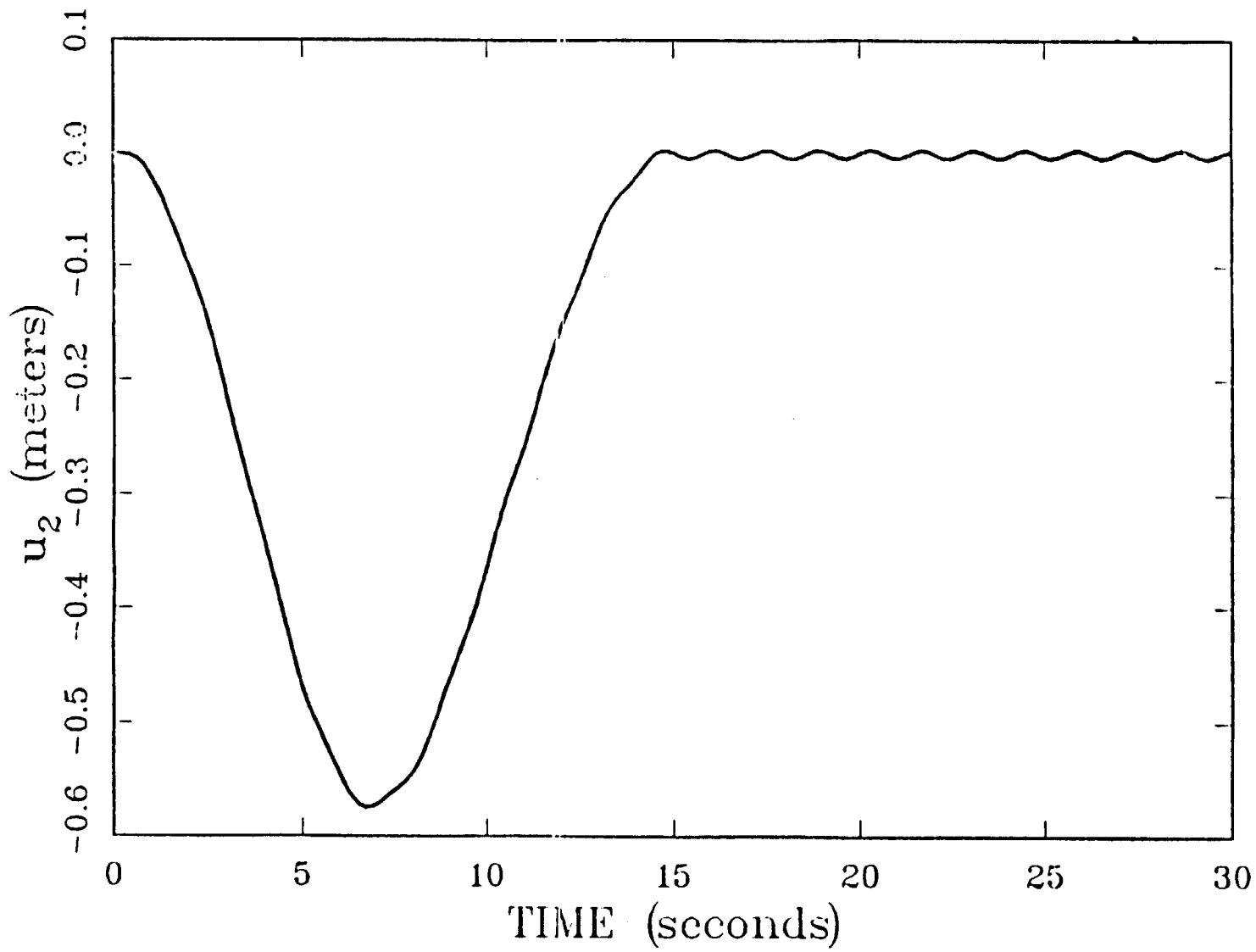
UNIFORM BEAM SPIN - UP PROBLEM : ADAMS APPROACH
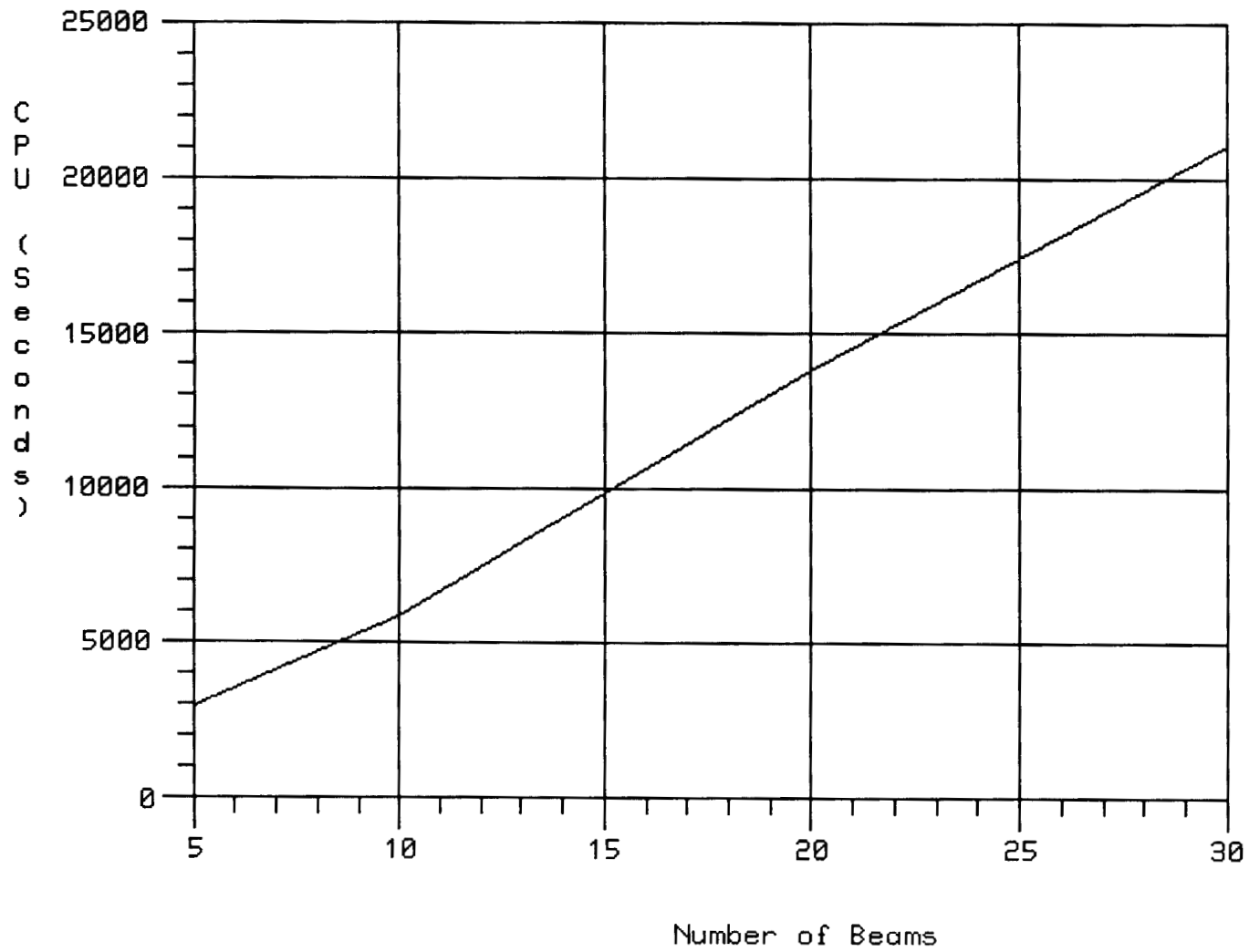
BEAM TIP DEFLECTION Vs. TIME

EFFICIENCY OF METHOD

- THIS METHOD IS USABLE FOR SMALL TO MEDIUM SIZE PROBLEMS
  (MEDIUM = 300 RIGID AND FLEXIBLE DOF)

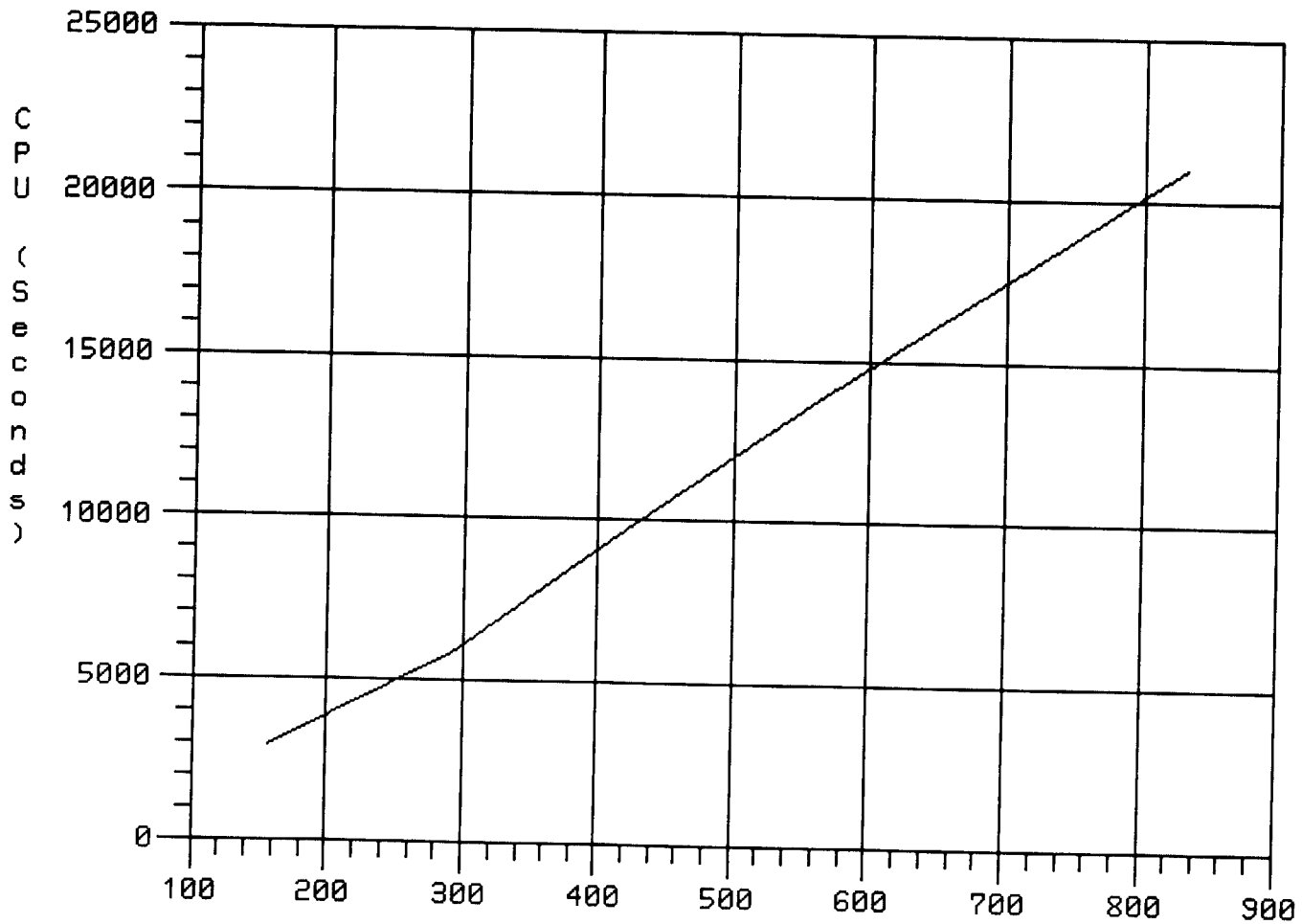- FOR LARGER PROBLEMS IT *MAY* PROVE TO BE MORE CPU INTENSIVE
  THAN DESIRABLE.

- THE CPU TIME TAKEN FOR A SIMULATION IS LINEARLY
  PROPORTIONAL TO THE NUMBER OF FLEXIBLE BEAMS IN
  THE SYSTEM

Number of Beams   Vs. CPU Time

259

Number of Equations   Vs.   CPU
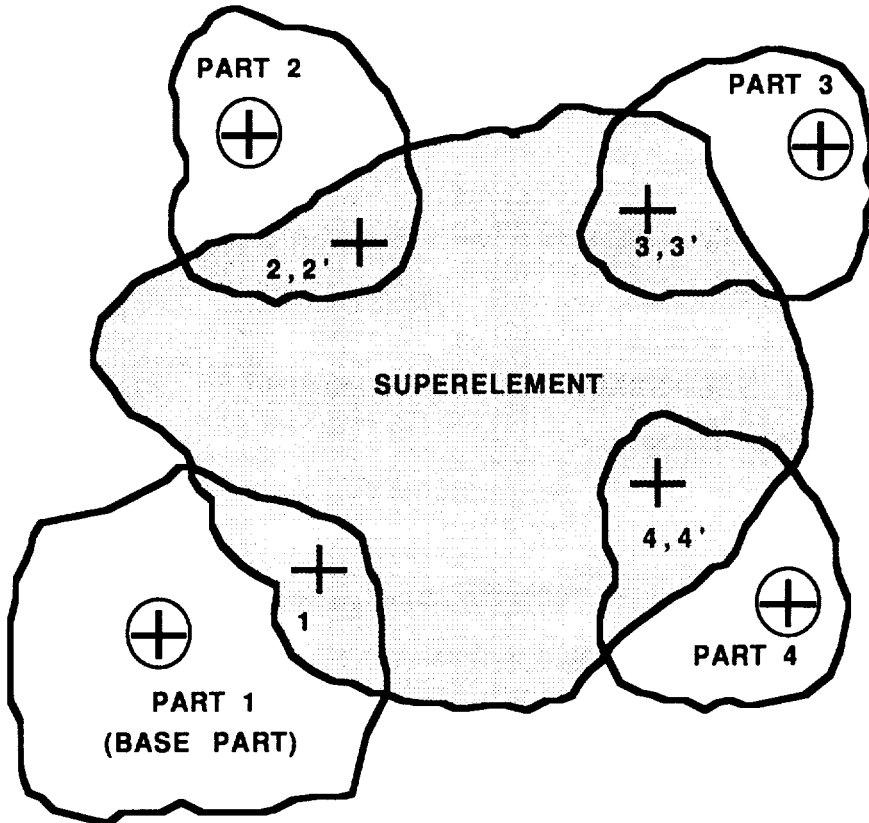
260

Number of Equations

EASE OF USE

- THE RESULTING PROGRAM IS EXTREMELY EASY TO USE.

- USERS DO NOT NEED STRONG FEA BACKGROUND TO CREATE
  MODELS OF STRUCTURES

- RECOGNITION AND SELECTION OF PROPER MODES IN AN ART.
  THE RESULTS ARE ONLY AS GOOD AS THE SELECTED MODES.
  DIFFICULTY ALLEVIATED IN THIS APPROACH.

GENERALIZATION TO SUPERELEMENTS

NODES

- **LOCATION OF NODES 2, 3, 4 WRT. TO A KNOWN REFERENCE FRAME**

- **MASS AND INERTIA PROPERTIES FOR EACH NODE OBTAINED FROM MASS MATRIX.**

LOCATION AND ORIENTATION OF COORDINATE SYSTEMS

2 ON PART 2	**2' ON PART 1**
3 ON PART 3	**3' ON PART 1**
4 ON PART 4	**4' ON PART 1**

ASSEMBLY

- **PRE-TENSION AND INITIAL DISPLACEMENTS AT CONNECTION POINTS 2-2', 3-3', 4-4'**

FLEXIBILITY PROPERTIES

- **STIFFNESS MATRIX**

- **DAMPING MATRIX**

DISPLACEMENT COORDINATES

$$_1\Delta = [\ _1\Delta_{2'2}\ \ _1\Delta_{3'3}\ \ _1\Delta_{4'4}\ ]^T$$

$$\beta = [\ \beta_{2'2}\ \ \beta_{3'3}\ \ \beta_{4'4}\ ]^T$$

VELOCITY COORDINATES

$$_1V = [\ _1V_{2'2}\ \ _1V_{3'3}\ \ _1V_{4'4}\ ]^T$$

$$_1\omega = [\ _1\omega_{2'2}\ \ _1\omega_{3'3}\ \ _1\omega_{4'4}\ ]^T$$

FORCE DEFINITION AT COORDINATE SYSTEMS   2, 3, 4

$$_1F = [\ _1F_2\ \ _1F_3\ \ _1F_4\ ]^T$$

$$_1T = [\ _1T_2\ \ _1T_3\ \ _1T_4\ ]^T$$

263

FORCE COMPUTATION AT COORDINATE SYSTEMS 2, 3, 4

$$_1F = -[K_{11}\,_1\Delta + K_{12}\,\beta]$$
$$-[C_{11}\,_1V + C_{12}\,_1\omega] + F_0$$

$$_1T = -[K_{21}\,_1\Delta + K_{22}\,\beta]$$
$$-[C_{21}\,_1V + C_{22}\,_1\omega] + T_0$$

FORCE AT COORDINATE SYSTEM 1

$$\vec{F}_1 + \vec{F}_2 + \vec{F}_3 + \vec{F}_4 = \vec{0}$$

$$_1F_1 = -[\,_1F_2 + \,_1F_3 + \,_1F_4\,]$$

$$\vec{T}_1 + \vec{T}_2 + \vec{T}_3 + \vec{T}_4 + \vec{S}_{12} \times \vec{F}_2 + \vec{s}_{13} \times \vec{F}_3 + \vec{s}_{14} \times \vec{F}_4 = \vec{0}$$

$$_1T_1 = -[\,_1F_2 + \,_1F_3 + \,_1F_4\,]$$

$$-[\,_1\bar{s}_{12}\,_1F_2 + \,_1\bar{s}_{13}\,_1F_3 + \,_1\bar{s}_{14}\,_1F_4\,]$$

264

A Comparison of Software for the Modeling and Control of Flexible Systems

Lawrence W. Taylor, Jr.
NASA Langley Research Center

Computational Aspects Workshop
July 12-14, 1988   Williamsburg, VA

Memorandum

To: Whom it may concern

From: 161/Chief Scientist, GCD

Subject: Development of Software for the Control of Flexible Systems

[handwritten annotation]

I propose a cooperative effort among specialists who use or develop
software for simulating and analyzing the control of flexible, aerospace
systems. A comparison of existing software for modeling control
systems and flexible structures, applied to several example problems
would be quite valuable. The comparison would indicate computational
efficiency and capabilities with respect to handling nonlinearities and
graphical output.

Because of the diversity of applications of such software, I believe that
the proposed cooperative effort can transcend projects involving
specific applications. Comparisons of software capability and efficiency
can be made and gaps can be identified. In this way the results of the
cooperative effort can provide guidance for individual projects.

Enclosed are several charts which outline the objectives and approach of
the proposed cooperative effort. I would appreciate your suggestions
and expressions of interest in this matter.

Sincerely,

Lawrence W. Taylor, Jr.
Mail Stop 161
NASA Langley Research Center
Hampton, VA 23665
(804)-865-4591

267

OBJECTIVE

"To Evaluate Software for the Control,
Analysis, Simulation and Design of
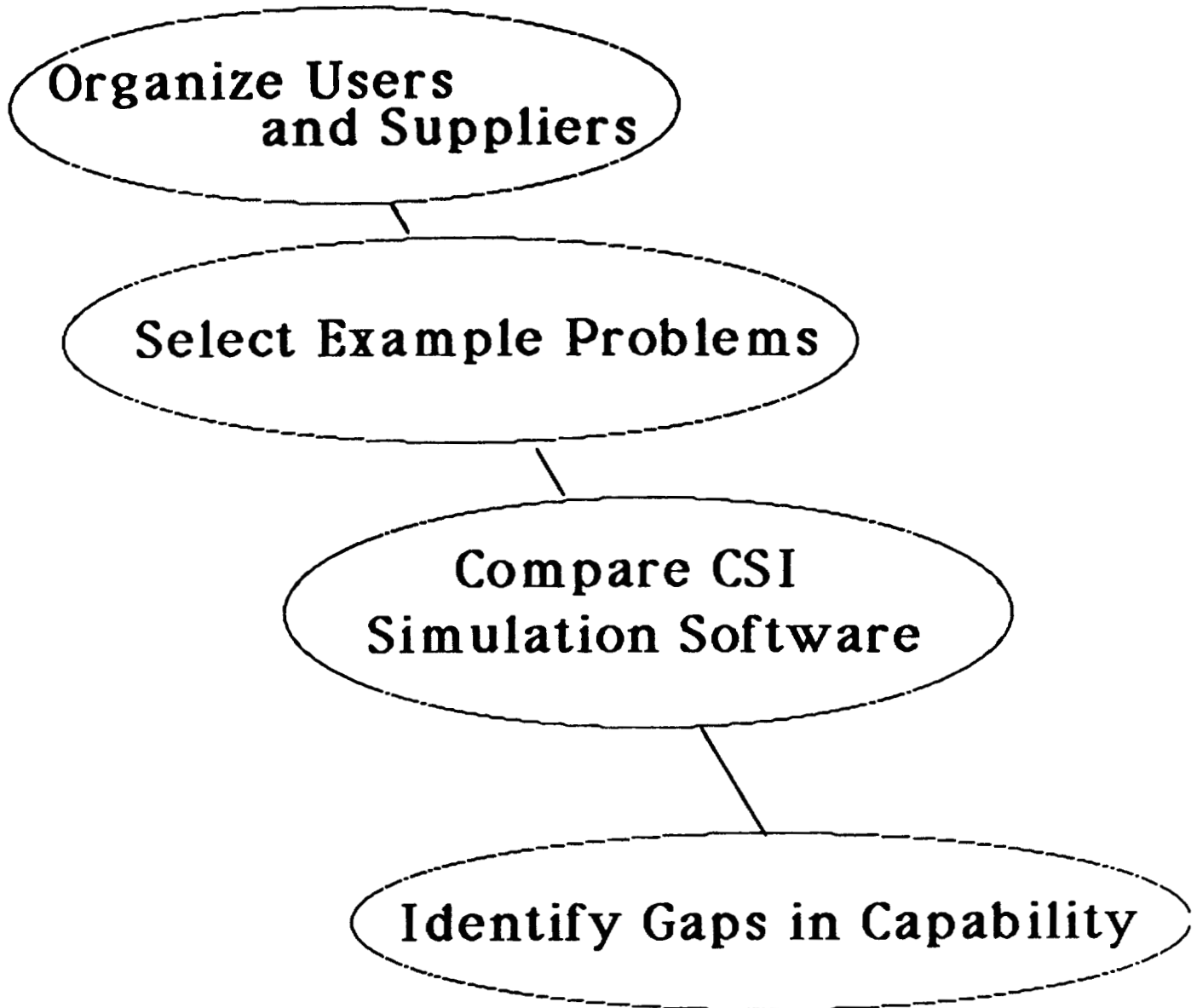of Flexible Aerospace Systems....

Which includes:

- Control Law Dynamics
- Actuator/Sensor Dynamics
- Structural Dynamics

And Which is <u>Efficient</u> and <u>Accurate</u>

And Which is <u>Easy to Use</u>."

APPROACH

Organize Users and Suppliers

Select Example Problems

Compare CSI Simulation Software

Identify Gaps in Capability

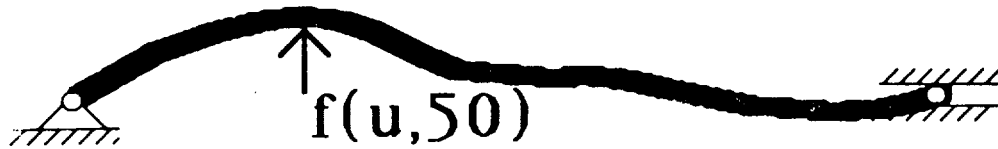# Example Problems

- Uniform Beam - Pinned-Pinned

- SCOLE (L. Taylor-LaRC)

- Translation/Rotation/Flex(Juang-LaRC)

- SAFE(70% Deployment)(L.Taylor-LaRC)

- Pinhole Occulter (Henry Waites-MSFC)

- Manned Space Station
  - a. Reboost
  - b. Solar Dynamic Pointing
  - c. MRMS Operation
  - d. Docking

Others?

Pinned-Pinned Beam

$$f(u,50)$$

$$m\ddot{u} + c|\ddot{u}"|\ddot{u}" + EIu"" = 0$$

$$u(x,0) = A\sin(\pi x/L)$$

$$A = 1.3 \text{ Ft.}$$

$$L = 130 \text{ Ft.}$$

$$EI = 40,000,000 \text{ Lb/Ft}^2$$

$$m = .09556 \text{ Slugs/Ft}$$

$$c = 280.32$$

$$f_{50}(s) = \frac{K\dot{u}(50,s)}{(1 + Ts)}$$

$$K = -.5$$

$$T = .2 \text{ Sec}$$

Problem:

1. Calculate Time History of u(65,t)   0<t<5.26
2. Plot Time History
3. Calculate Modal Characteristics   1<k<10
4. Express Final Shape in Modal Coordinates

CSI Simulation Software

- NASTRAN

- DISCOS

- TREETOPS(CONTOPS)

- EAL
- LATDYN
- DADS
- Multi-MACS

- ORACLS
- EISPAC
- LINPAC
- Matrix$_x$
- CTRL-C
- SYSPAC

Information Sheet

NAME of SOFTWARE: DISCOS

RESPONSIBLE PERSON: Harry Frisch
NASA Goddard Space Center
Bldg. 11, Rm. S221A
Greenbelt, MD 20771

CAPABILITIES:

1. Nonlinear Kinematics?	Yes ☑ No ☐	_____
2. Finite Element Modeling?	Yes ☑ No ☐	_____
3. All Rotational and Translational D.O.F.?	Yes ☑ No ☐	_____
4. Linked to Distributed Parmeters?	Yes ☐ No ☐	_____
5. Large Amplitudes?	Yes ☑ No ☐	_____
6. Nonlinear Damping?	Yes ☐ No ☑	_____
7. Control Law Dynamics?	Yes ☐ No ☑	_____
8. Sensors and Actuator Dynamics?	Yes ☐ No ☑	_____
9. Nonlinear Joints?	Yes ☐ No ☑	_____
10. Distributed Parameter System?	Yes ☐ No ☑	_____
11. Optimal Control Synthesis?	Yes ☐ No ☑	_____
12. Sensitivity Functions for P.E. & Design?	Yes ☐ No ☑	_____

DOCUMENTATION: Complete.

SIZE: 500K

LANGUAGE: Fortran 77 +

INTERACTIVE: No.

GRAPHICS: Plots

AVAILABILITY: Free, Nonproprietary

273

Information Sheet

NAME of SOFTWARE: LATDYN

RESPONSIBLE PERSON: Jerry Housner
                               Mail Stop – 230
                               NASA Langley Research Center
                               Hampton, VA   23665

CAPABILITIES:

1. Nonlinear Kinematics?	Yes ☑	No ☐	2-D.O.F.
2. Finite Element Modeling?	Yes ☑	No ☐	2-D.O.F.
3. All Rotational and Translational D.O.F.?	Yes ☑	No ☐	2-D.O.F.
4. Linked to Distributed Parmeters?	Yes ☑	No ☐	2-D.O.F.
5. Large Amplitudes?	Yes ☑	No ☐	2-D.O.F.
6. Nonlinear Damping?	Yes ☑	No ☐	Add Code
7. Control Law Dynamics?	Yes ☑	No ☐	Add Code
8. Sensors and Actuator Dynamics?	Yes ☑	No ☐	Add Code
9. Nonlinear Joints?	Yes ☑	No ☐	Add Code
10. Distributed Parameter System?	Yes ☑	No ☐	2-D.O.F.
11. Optimal Control Synthesis?	Yes ☐	No ☑	
12. Sensitivity Functions for P.E. & Design?	Yes ☐	No ☑	

DOCUMENTATION: 2-D.O.F. Written, 3-D.O.F. Under Development

SIZE: 400K

LANGUAGE: Fortran 77

INTERACTIVE: Yes

GRAPHICS: Time Histories, Line Drawing, PSD, Movies

AVAILABILITY: Free, Nonproprietary

Information Sheet

NAME: Multibody Analysis & Control Synthesis (MACS)

RESPONSIBLE PERSON: Lawrence W. Taylor
NASA Langley Research Center
Hampton, VA 23665
(804)-865-4591

CAPABILITIES:

1. Nonlinear Kinematics?	Yes ☑ No ☐	_____
2. Finite Element Modeling?	Yes ☑ No ☐	_____
3. All Rotational and Translational D.O.F.?	Yes ☑ No ☐	_____
4. Linked to Distributed Parmeters?	Yes ☑ No ☐	_____
5. Large Amplitudes?	Yes ☑ No ☐	_____
6. Nonlinear Damping?	Yes ☐ No ☑	Not Yet
7. Control Law Dynamics?	Yes ☑ No ☐	_____
8. Sensors and Actuator Dynamics?	Yes ☑ No ☐	_____
9. Nonlinear Joints?	Yes ☐ No ☑	Not Yet
10. Distributed Parameter System?	Yes ☐ No ☑	_____
11. Optimal Control Synthesis?	Yes ☐ No ☑	_____

DOCUMENTATION: Incomplete

SIZE: 25K Core Memory

LANGUAGE: FORTRAN 77

INTERACTIVE: No

GRAPHICS: None

AVAILABILITY: No Charge

275

Information Sheet

NAME of SOFTWARE: TREETOPS (CONTOPS)

RESPONSIBLE PERSON: Ramen Singh
Dynacs Engineering Company
2280 U.S. 19 No., Suite 111
Clearwater, FL 34623

CAPABILITIES:

1. Nonlinear Kinematics?	Yes ☑ No ☐ _____
2. Finite Element Modeling?	Yes ☑ No ☐ _____
3. All Rotational and Translational D.O.F.?	Yes ☑ No ☐ _____
4. Linked to Distributed Parmeters?	Yes ☑ No ☐ _____
5. Large Amplitudes?	Yes ☑ No ☐ _____
6. Nonlinear Damping?	Yes ☑ No ☐ _____
7. Control Law Dynamics?	Yes ☑ No ☐ _____
8. Sensors and Actuator Dynamics?	Yes ☑ No ☐ _____
9. Nonlinear Joints?	Yes ☑ No ☐ _____
10. Distributed Parameter System?	Yes ☑ No ☐ _____
11. Optimal Control Synthesis?	Yes ☐ No ☑ _____
12. Sensitivity Functions for P.E. & Design?	Yes ☐ No ☑ _____

DOCUMENTATION: Complete. Course Available.

SIZE: 600K

LANGUAGE: Fortran 77 +

INTERACTIVE: Yes. Sun, MicroVAX, Masscomp

GRAPHICS: Plots, Windows, Movies

AVAILABILITY: Free, Nonpropriety

276

Information Sheet

NAME of SOFTWARE: _____

RESPONSIBLE PERSON: _____

_____

_____

_____

CAPABILITIES:

1. Nonlinear Kinematics? Yes ☐ No ☐ _____
2. Finite Element Modeling? Yes ☐ No ☐ _____
3. All Rotational and Translational D.O.F.? Yes ☐ No ☐ _____
4. Linked to Distributed Parmeters? Yes ☐ No ☐ _____
5. Large Amplitudes? Yes ☐ No ☐ _____
6. Nonlinear Damping? Yes ☐ No ☐ _____
7. Control Law Dynamics? Yes ☐ No ☐ _____
8. Sensors and Actuator Dynamics? Yes ☐ No ☐ _____
9. Nonlinear Joints? Yes ☐ No ☐ _____
10. Distributed Parameter System? Yes ☐ No ☐ _____
11. Optimal Control Synthesis? Yes ☐ No ☐ _____
12. Sensitivity Functions for P.E. & Design? Yes ☐ No ☐ _____

DOCUMENTATION: _____

SIZE: _____

LANGUAGE: _____

INTERACTIVE: _____

GRAPHICS: _____

AVAILABILITY: _____

Solution Characteristics

- Time Required – CPU Sec.

- Accuracy
  a. Modal Characteristics
  b. Time Histories (PSD)

- Memory Required

- Input

- Output

PANEL   DISCUSSION

1.   Should <u>We</u> Compare Software for

Control/Modeling?

2.   Should <u>We</u> Establish Example Problems?

3.   Should This Workshop be Repeated?

......................Changed?

....................Merged?

SESSION III - COMPUTATIONS EFFICIENCY AND CAPABILITY

PRECEDING PAGE BLANK NOT FILMED

PAGE 280 INTENTIONALLY BLANK

A FINITE ELEMENT APPROACH FOR LARGE MOTION DYNAMIC ANALYSIS OF MULTIBODY STRUCTURES IN SPACE

By

Che-Wei Chang
The COMTEK Company
Grafton, Virginia

ABSTRACT

A three-dimensional finite element formulation for modeling the transient dynamics of constrained multibody space structures with truss-like configurations is presented. Convected coordinate systems are used to define rigid-body motion of individual elements in the system. These systems are located at one end of each element and are oriented such that one axis passes through the other end of the element. Deformation of each element, relative to its convected coordinate system, is defined by cubic flexural shape functions as used in finite element methods of structural analysis. The formulation is oriented toward joint dominated structures and places the generalized coordinates at the joint. A transformation matrix is derived to integrate joint degree-of-freedom into the equations of motion of the element. Based on the derivation, a general-purpose code LATDYN (Large Angle Transient DYNamics) has been developed. Two examples are presented to illustrate the application of the code. For the spin-up of a flexible beam, results are compared with existing solutions available in the literature. For the deployment of one bay of a deployable space truss (the "Minimast"), results are verified by the geometric knowledge of the system and converged solution of a successively refined model.

LATDYN

Large Angle Transient DYNamics

(Finite-Element-Based)

A NASA Facility for Research

in

Applications and Analysis Techniques

for Space Structure Dynamics

Presented by

Che-Wei Chang

COMTEX

284

TALK OUTLINE

* Motivation

* Capability

* Theory

* Modelling

* Present LATDYN
  (verifications)

* Future LATDYN

* Conclusions

CAPABILITIES

* **Three-Dimensional**

* **Deformable Bodies**

* **Multi-Connection Joints**

* **Large Angular Motion**

* **Variable Constraints**

* **Impacts & Joint-Lock**

* **Experimental Data**

* **User's Control Strategy**

BACKGROUND THEORY

1. Corotational Axes
   (convected system)

2. F-E Connectivity through

   Joint Kinematics

3. Numerical Integrations

287

DEFORMED FINITE ELEMENT AND ELEMENT COORDINATE SYSTEMS

Convected axis

End 2

Element coordinates move
with cross-section

End 1
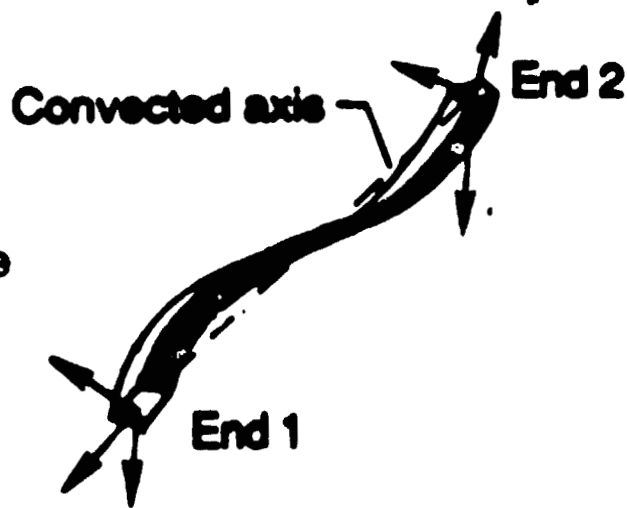
Deformations are measured from convected axes

Deformation : u
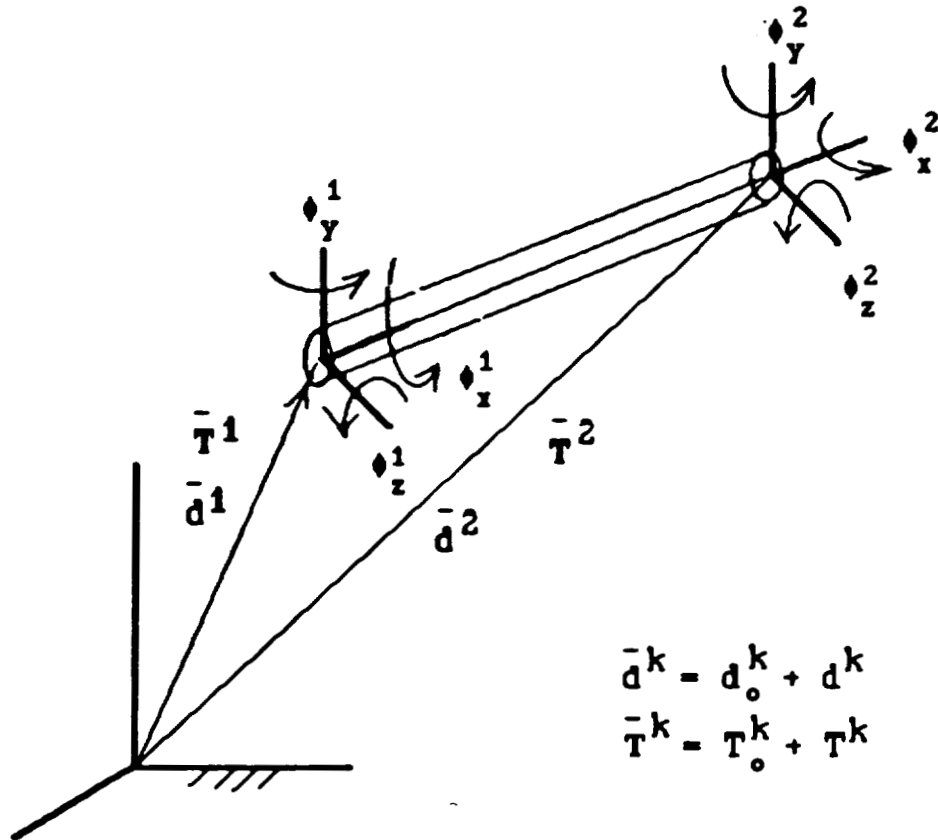
$$\bar{d}^k = d_o^k + d^k$$
$$\bar{T}^k = T_o^k + T^k \qquad : k = 1 \text{ or } 2$$

$$\Phi = [\phi_x^1, \phi_y^1, \phi_z^1, \phi_x^2, \phi_y^2, \phi_z^2,$$

$$u_x^1, u_y^1, u_z^1, u_x^2, u_y^2, u_z^2]^T$$

$$\boxed{u = N\,\Phi}$$

$$U = [\, u_x \, , u_y \, , u_z \,]^T$$

$$\boxed{\Phi = \Phi\,(d,T)}$$

$$d = [\, d^{1^T} , \, d^{2^T}]^T \; :$$

$$T = T(\theta^1, \theta^2)$$

289

Internal Force

because

$$\varepsilon = D \Phi \qquad \& \qquad \sigma = E \varepsilon$$

$\therefore$

$$\sigma = E D \Phi$$

$$\delta\varepsilon = D \delta\Phi = DB \delta q$$

virtual work done by internal force

$$\delta W = -\delta U$$

$$= -\int_{dv} \delta\varepsilon^T \sigma \; dv$$

$$= \delta q^T \{-\int_{dv} (DB)^T D\Phi \; dv\}$$

Total displacement $\bar{\mathbf{u}}$

$$\bar{d}^k = d_o^k + d^k$$
$$\bar{T}^k = T_o^k + T^k$$
$$: k = 1 \text{ or } 2$$
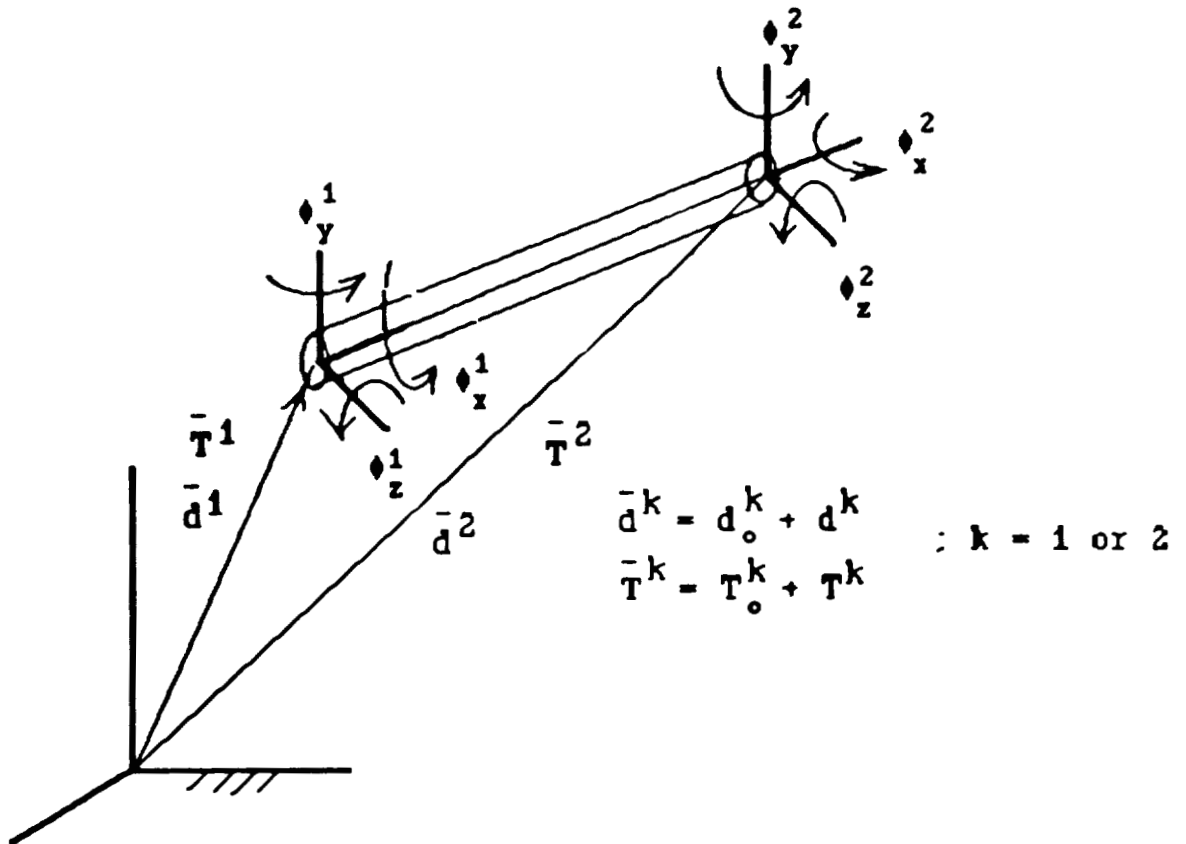
$$\bar{\mathbf{u}} = \bar{\mathbf{d}}^1 + \mathbf{T_c}\,\mathbf{u}$$

$$\delta\bar{\mathbf{u}} = \mathbf{C}\,\delta\mathbf{q}$$

$$\dot{\bar{\mathbf{u}}} = \mathbf{C}\,\dot{\mathbf{q}}$$

$$\ddot{\bar{\mathbf{u}}} = \mathbf{C}\,\ddot{\mathbf{q}} + \dot{\mathbf{C}}\,\dot{\mathbf{q}}$$

C-4

291

Inertia

virtual work done by inertia force

$$\delta W = \int_{dv} \{ \delta \bar{u}^T (-\rho \ddot{\bar{u}}) \} \, dv$$

$$= \delta q^T \{ - \{ \int_{dv} \rho \, C^T C \, dv \} \ddot{q} \}$$

$$+ \delta q^T \{ - \{ \int_{dv} \rho \, C^T \dot{C} \, dv \} \dot{q} \}$$

TYPICAL INTERCONNECTION OF TWO JOINT BODIES
THROUGH FLEXIBLE BEAM
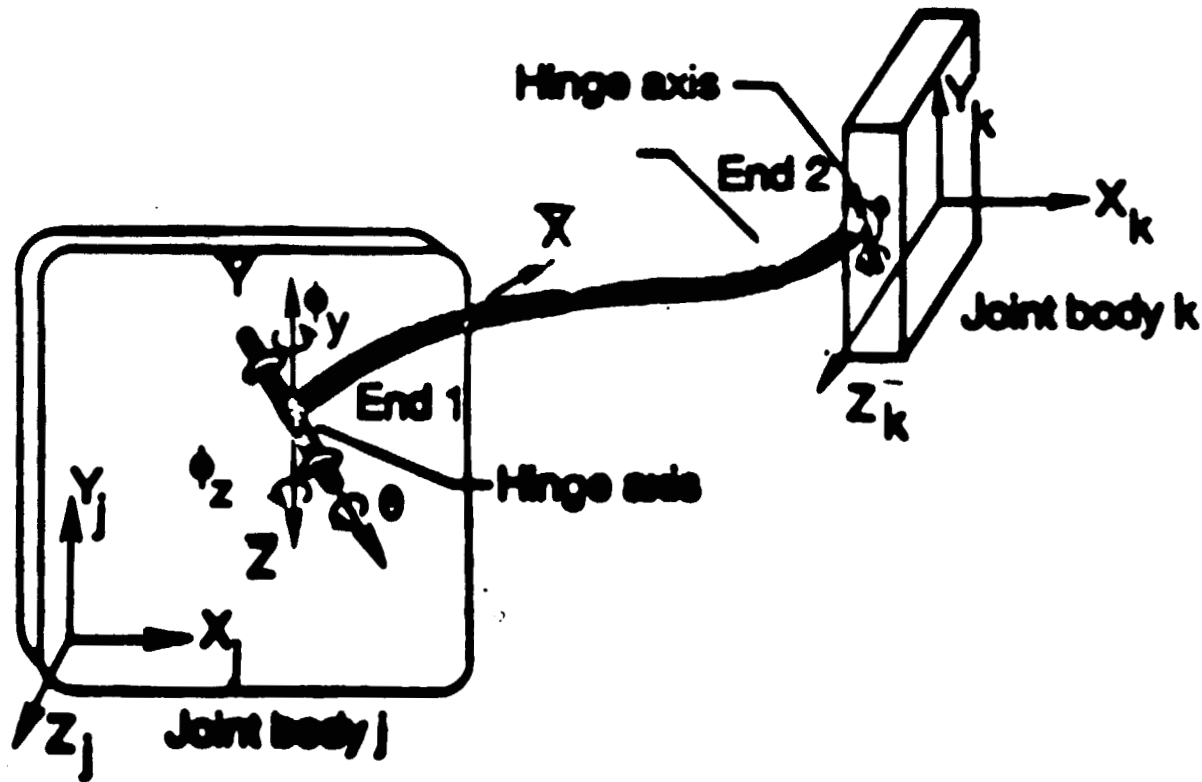
Element EQ's of Motion

in terms of nodal disp.

$$M\ddot{q} = F^{E} + F^{I} + g$$

GENERIC JOINT BODY WITH VARIOUS TYPES OF HINGE CONNECTIONS

Joint body coordinates

Revolute connection

Joint body

Axle

Universal

Ball

$$M\ddot{q} = F^{\Sigma} + F^{I} + g$$

Joint Kinematics

offset

q

joint dof's
p

body dof's
$\bar{p}$
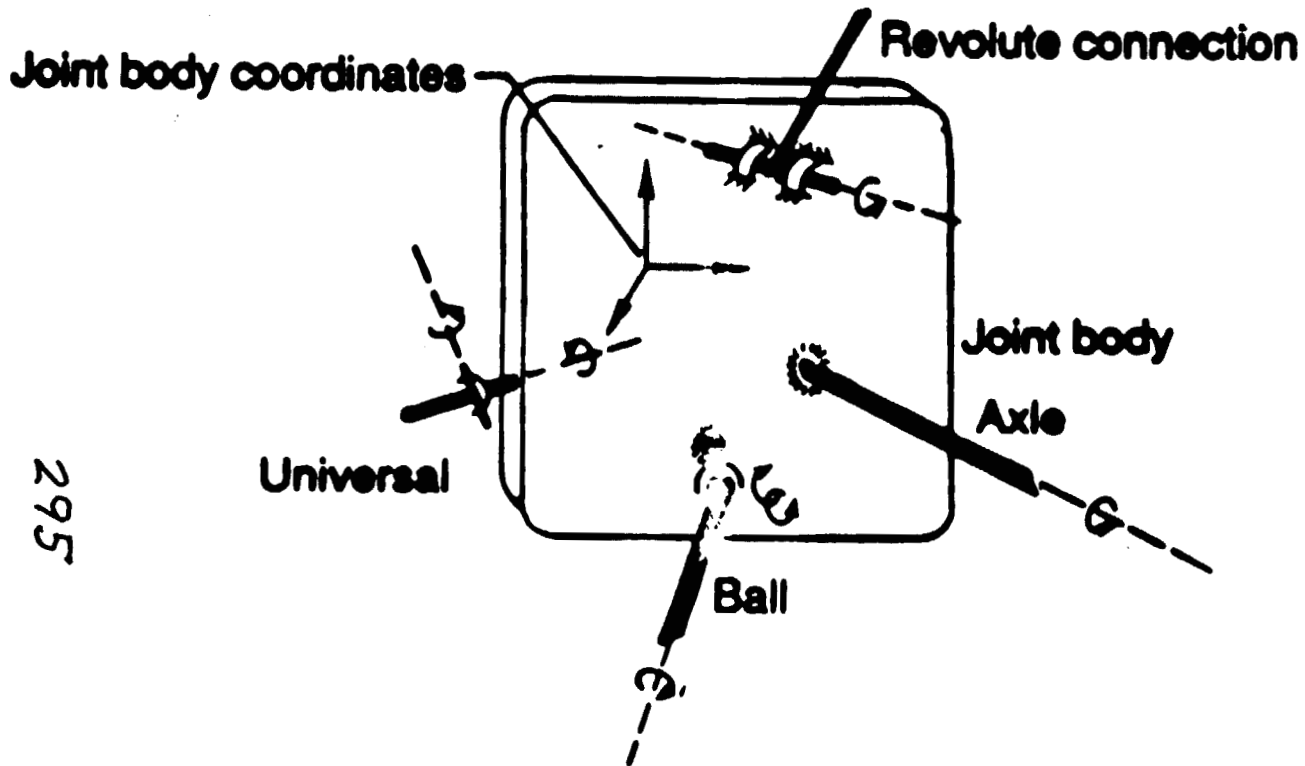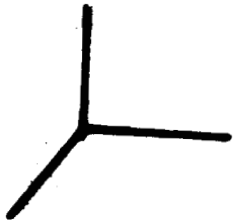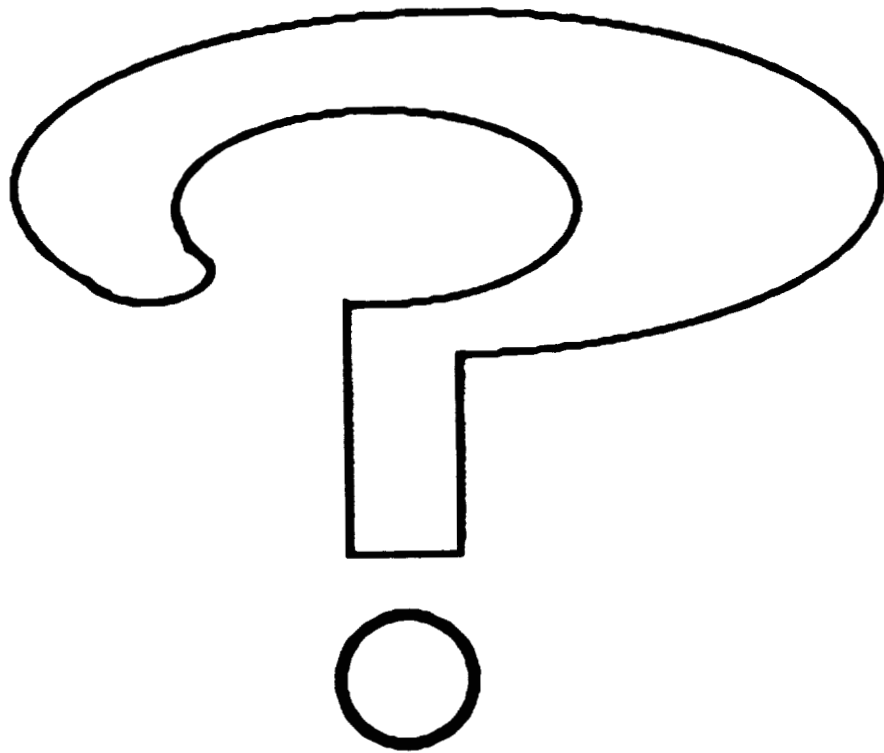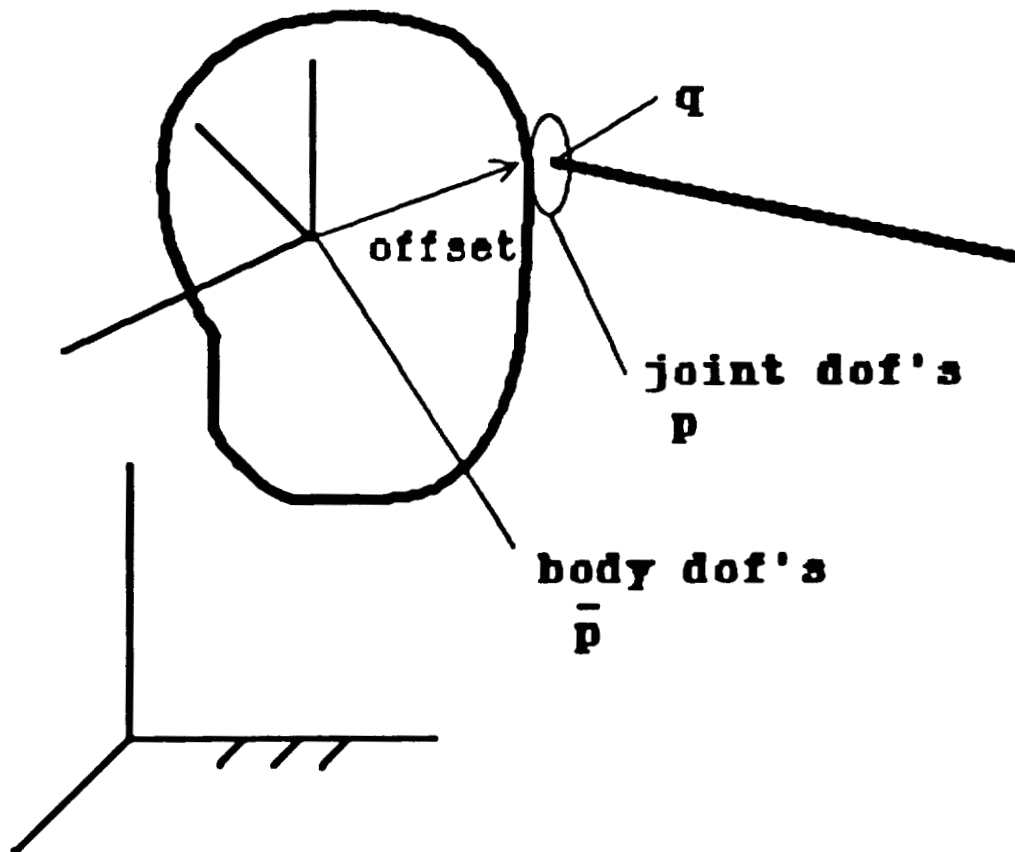
$$q = q(\bar{p}, p)$$
$$= q(\bar{q})$$

$$\delta q = H \, \delta \bar{q}$$

$$\dot{q} = H \, \dot{\bar{q}}$$

$$\ddot{q} = H \, \ddot{\bar{q}} + \dot{H} \, \dot{\bar{q}}$$

Multi-Joint Body
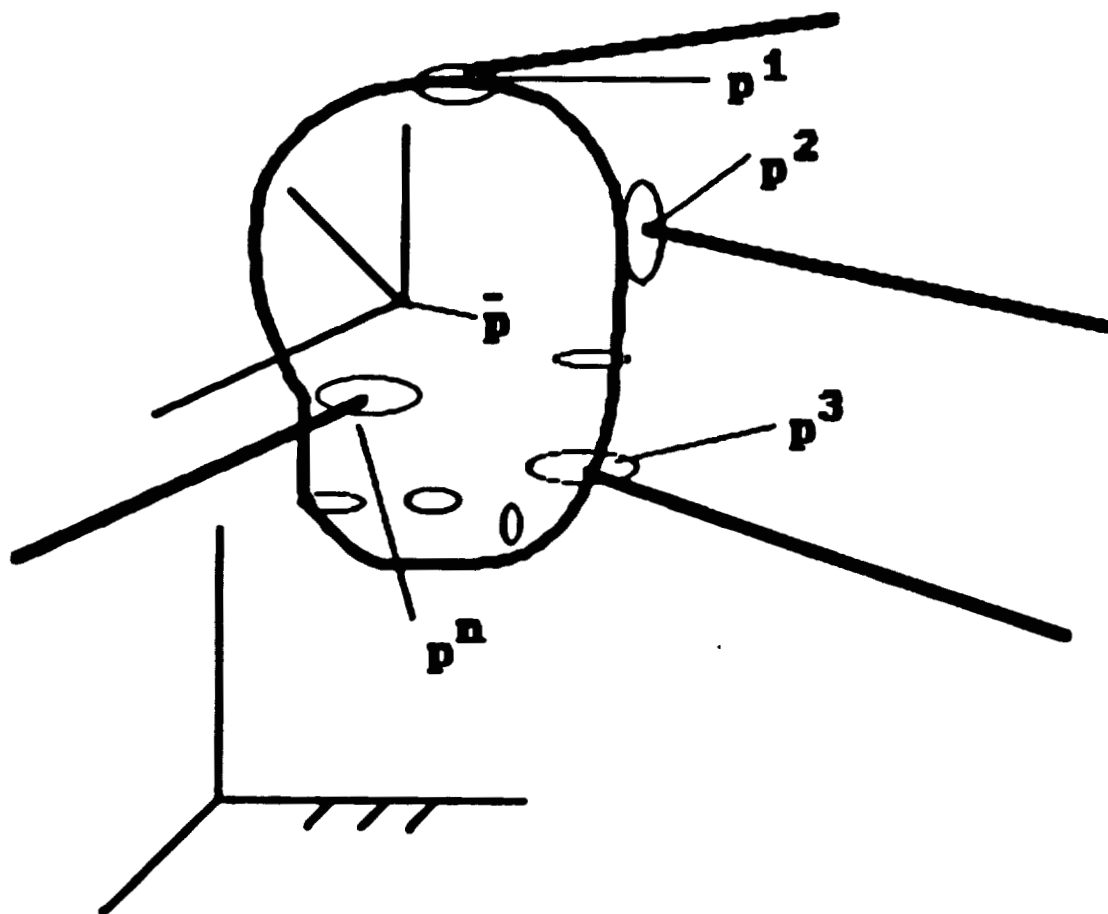
generalized coords.

$$\bar{q} = [\bar{p}^T, p^{1^T}, p^{2^T}, \ldots, p^{n^T}]^T$$

298

<u>Rigid Chain</u>

$$q = q (\bar{p} . p_1 . p_2 . p_3)$$
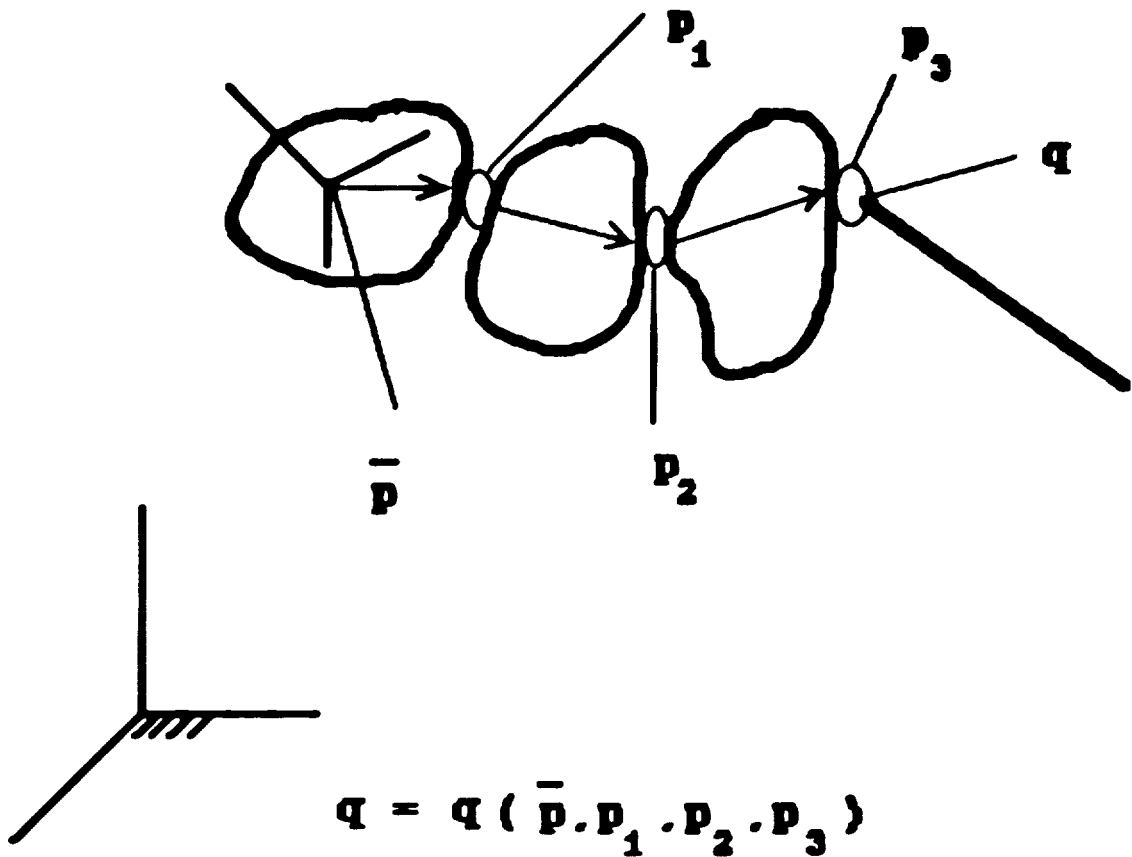
generalized coord.

$$\bar{q} = [\bar{p}^T . p_1^T . p_2^T . p_3^T]^T$$

<u>Generalized Coordinates</u>

at each joint body:

      3 translational disp.

      3 orientational disp.

  +   No. of relative(joint)

      d-o-f's

System EQ's of Motion

in terms of joint body
and joint dof's

$$\bar{M}\ddot{\bar{q}} = \bar{F}^{\Sigma} + \bar{F}^{I} + \bar{g}$$

Equations of Motion and Their Numerical Integration

At n^{th} time step,

$$M^n a^n + f'' + g^n = F^n$$

Newmark-Beta Integrator at k^{th} iteration:

$$a_k^n = a_{k-1}^n + \left[M_{k-1}'' + \frac{h}{2} G_{k-1}^n + \beta h^2 K_{k-1}^n \right]^{-1} R_k^n \qquad \text{:Update Accelerations}$$

$$R_k^n = \text{iterative residual} = F^n - f_{k-1}^n - M_{k-1}^n a_{k-1}^n$$

$$V_k^n = V^{n-1} + \left(\frac{h}{2} \right) \left(a^{n-1} + a_k^n \right) \qquad \text{:Update Velocities}$$

302

Equations of Motion and Their Numerical Integration (cont'd)

Split into translational and rotational d.o.f.

Translational displacements are

$$d_k^n = d^{n-1} + hv^{n-1} + \left(\tfrac{1}{2} - \beta\right) h^2 a^{n-1} + \beta h^2 a_k^n \qquad \text{:Update Translational d.o.f.}$$

Rotational motions are given by transformation matrix:

$$T_k^n = \left[1 + h\overline{\omega}_k^n + \tfrac{1}{2} h \left(\overline{\omega}_k^n\right)^2\right] T^{n-1} \qquad \text{:Update hinge body}$$

transformation

$$\overline{\omega}_k^n = \omega_k^n + \omega^{n-1}$$

303

Modelling Techniques

* How does user work with LATDYN ?

* How does program model a system ?

```
                    ┌───────────┐
                    │   user    │
                    └───────────┘
                          │
                          ▼
    ┌─────────────────────────────────────────────┐
    │  ┌──────────────┐    ┌──────────────────┐    │
    │  │  modelling   │    │   programmable   │    │
    │  │  commands    │    │    language      │    │
    │  └──────────────┘    └──────────────────┘    │
    └─────────────────────────────────────────────┘
                          │
                          ▼
         ⟨          preprocessor          ⟩
              │                     │
              ▼                     ▼
    ┌─────────────────────────────────────────────┐
    │   ┌──────────┐         ┌────────────────┐    │
    │   │  model   │ ──────▶ │    user's      │    │
    │   │  data    │         │  subroutines   │    │
    │   └──────────┘         └────────────────┘    │
    │        ⇅                      ⇅              │
    │   ⟨            LATDYN                  ⟩      │
    └─────────────────────────────────────────────┘
                          │
                          ▼
                  ┌───────────────┐
           ┌──────│    dynamic    │──────┐
           │      │    results    │      │
           ▼      └───────────────┘      ▼
    ⟨ postprocessor ⟩              ┌──────────┐
        │     │                   │  output  │
        ▼     └──────────────────▶└──────────┘
   ┌──────────┐                        │
   │ graphics │                        │
   └──────────┘                        │
        │                              │
        ▼                              ▼
         ⟨            user            ⟩
```

305
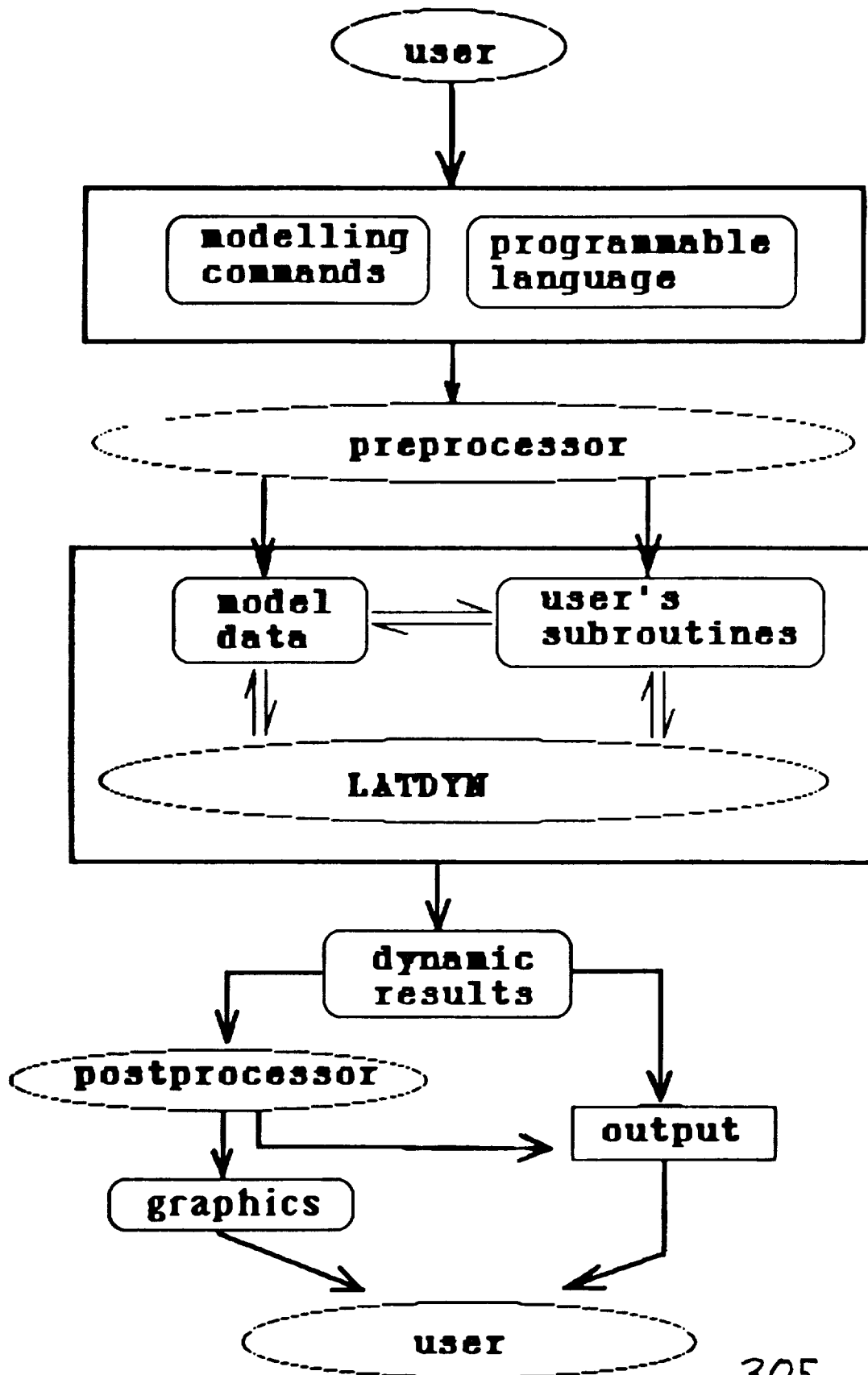
Defining the Model

1. Numerical Control

2. Flexible Bodies

* material properties

* element properties

* grid points

3. Rigid Connections

* body geometry & mass

* joint connections

4. Forcing Elements

* Forcing functions

* spring-damper-actuators

5. Initial Conditions

6. programmable language
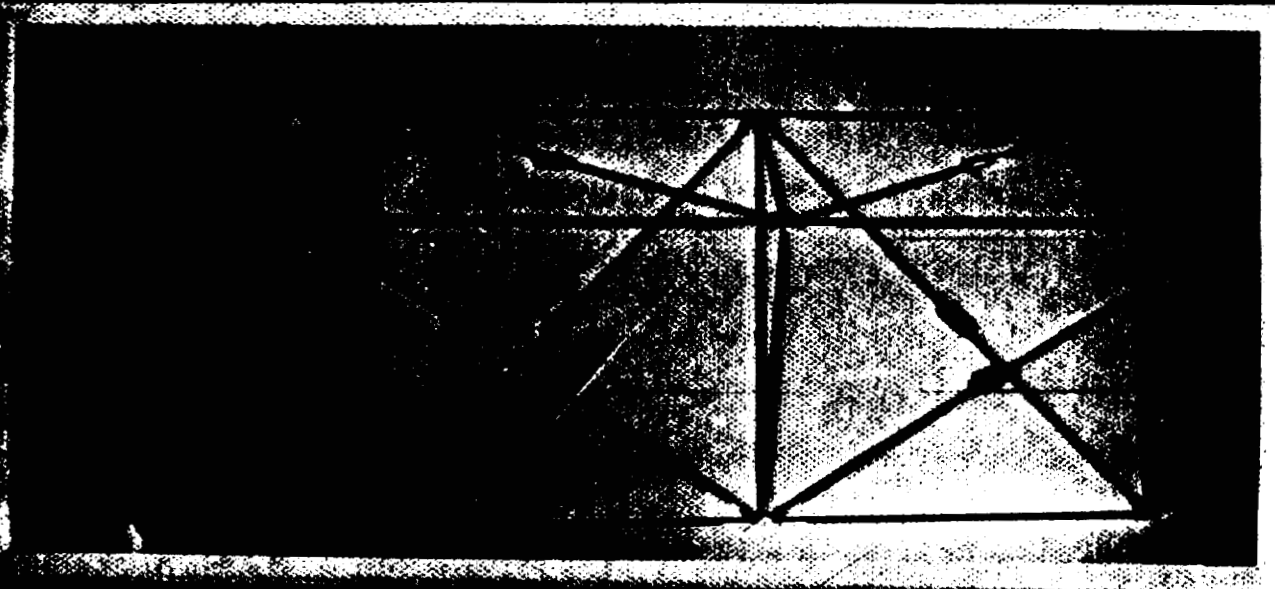
307

ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

6-JUN-88 12:26:38
UNITS = IN
DISPLAY:No stored OPTION

SDRC I-DEAS 3.9: Pre/Post Processing

DATABASE: CORNERBODY B FOR MINIMAST
VIEW:WHOLE1 (modified)
Task: Free Mesh Generation

Conventional

F-E Model of corner body
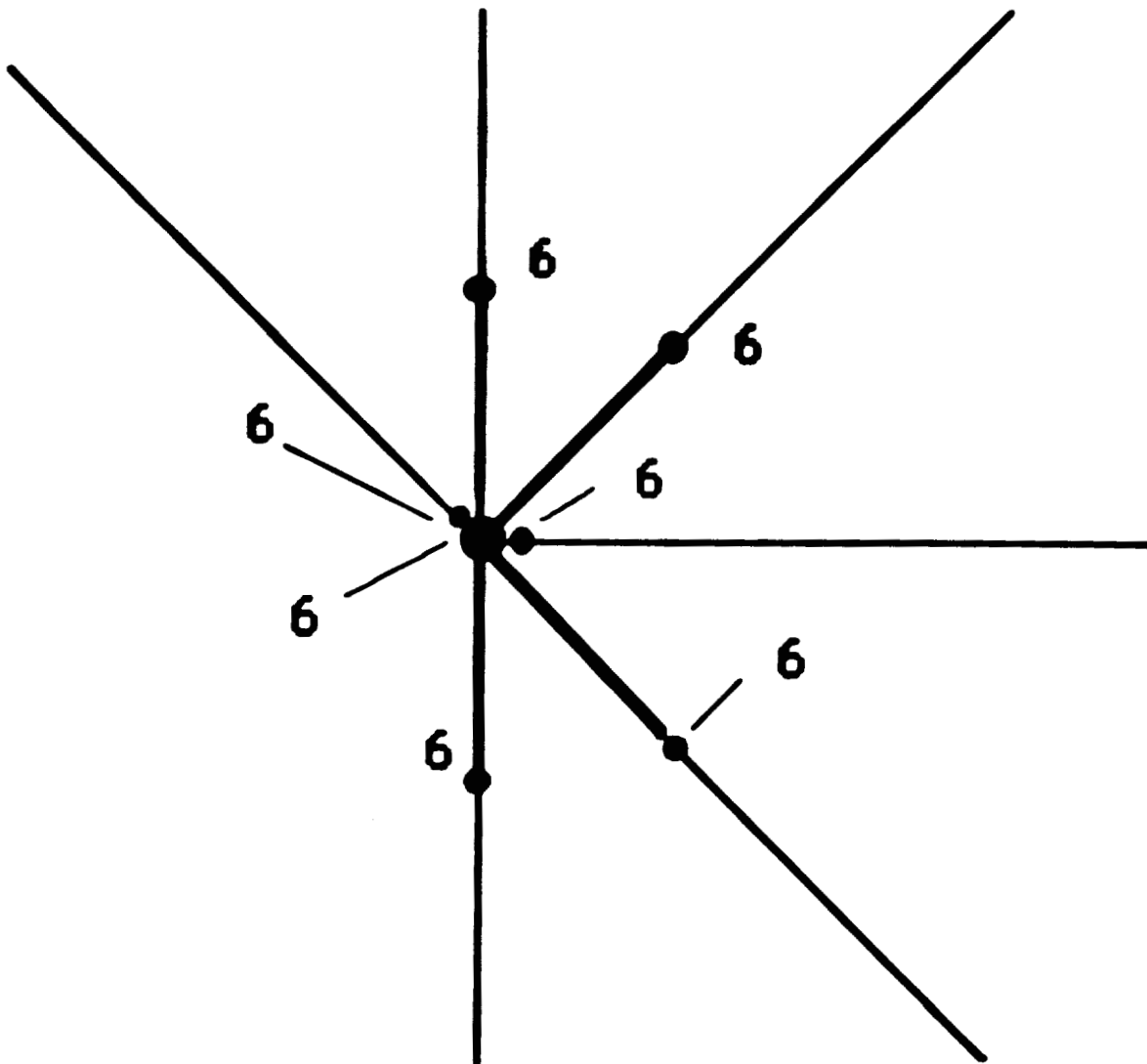
● grid (6-dof's)

42-dof's

32-constraints

310

LATDYN

F-E Model of corner body

● grid (6-dof's)

○ hinge (1-dof)
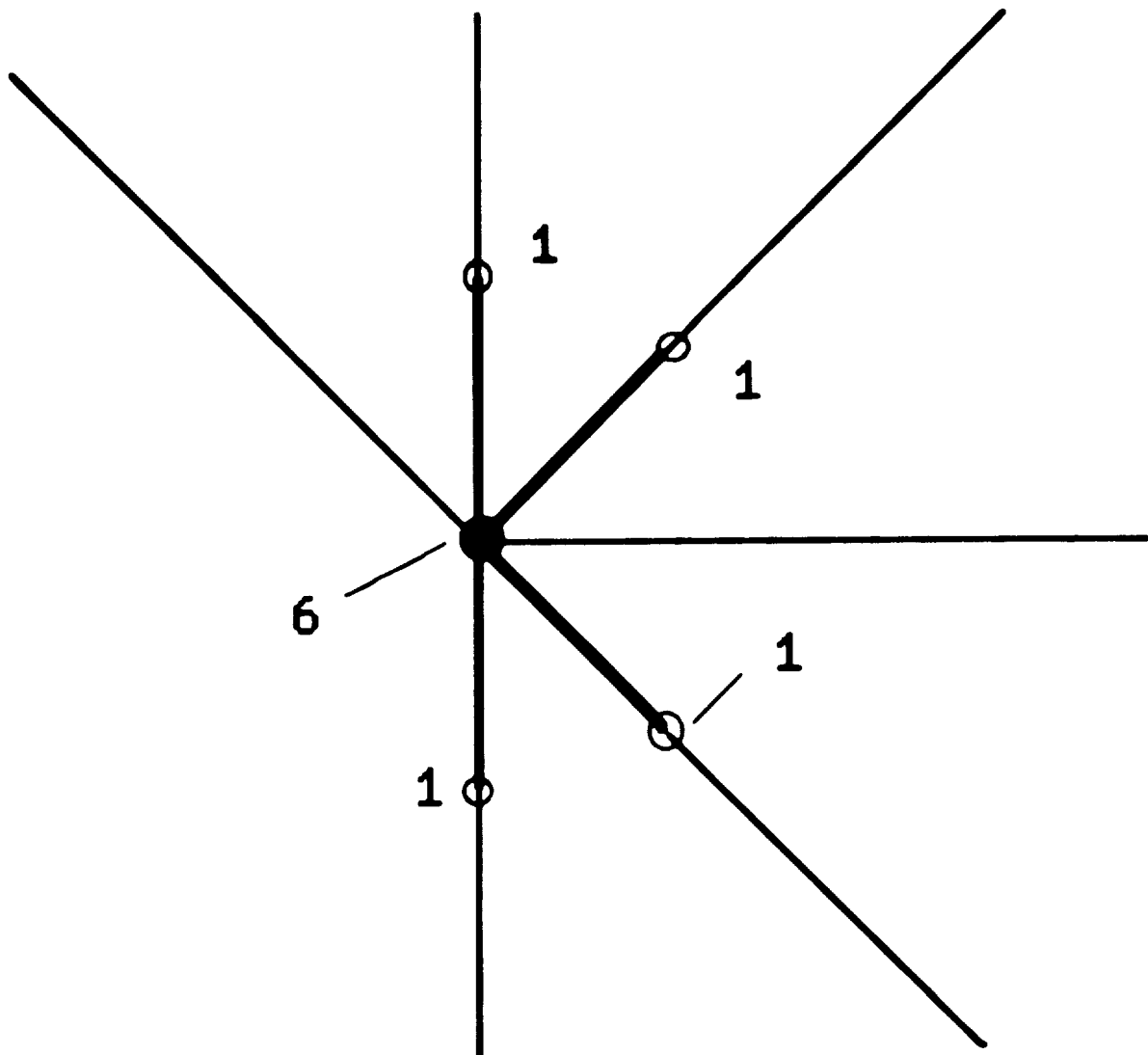
10-dof's

0-constraint

311

3-D LATDYN Model of Mini-Mast
Locking Joint
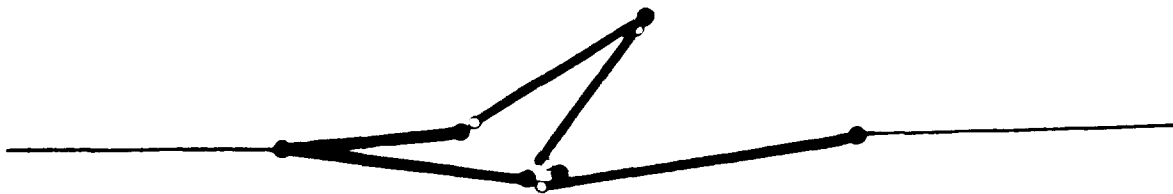
Note: closed loop

Target is that user will not have to specify how rigid members are formulated.

Program will determine most efficient arrangement, and will cut closed loops and implement constraints automatically.

Present LATDYN

* three-dimensional

* Euler-Bernoulli beam
  elements

* hinge connections

* Newmark-β explicit &
  implicit methods

* constraints & joints

* external forcing function
  & spring-damper-actuator

313

COMPARISON OF RESULTS FOR FLEXIBLE BEAM SPIN-UP ON A PLANE

3/4
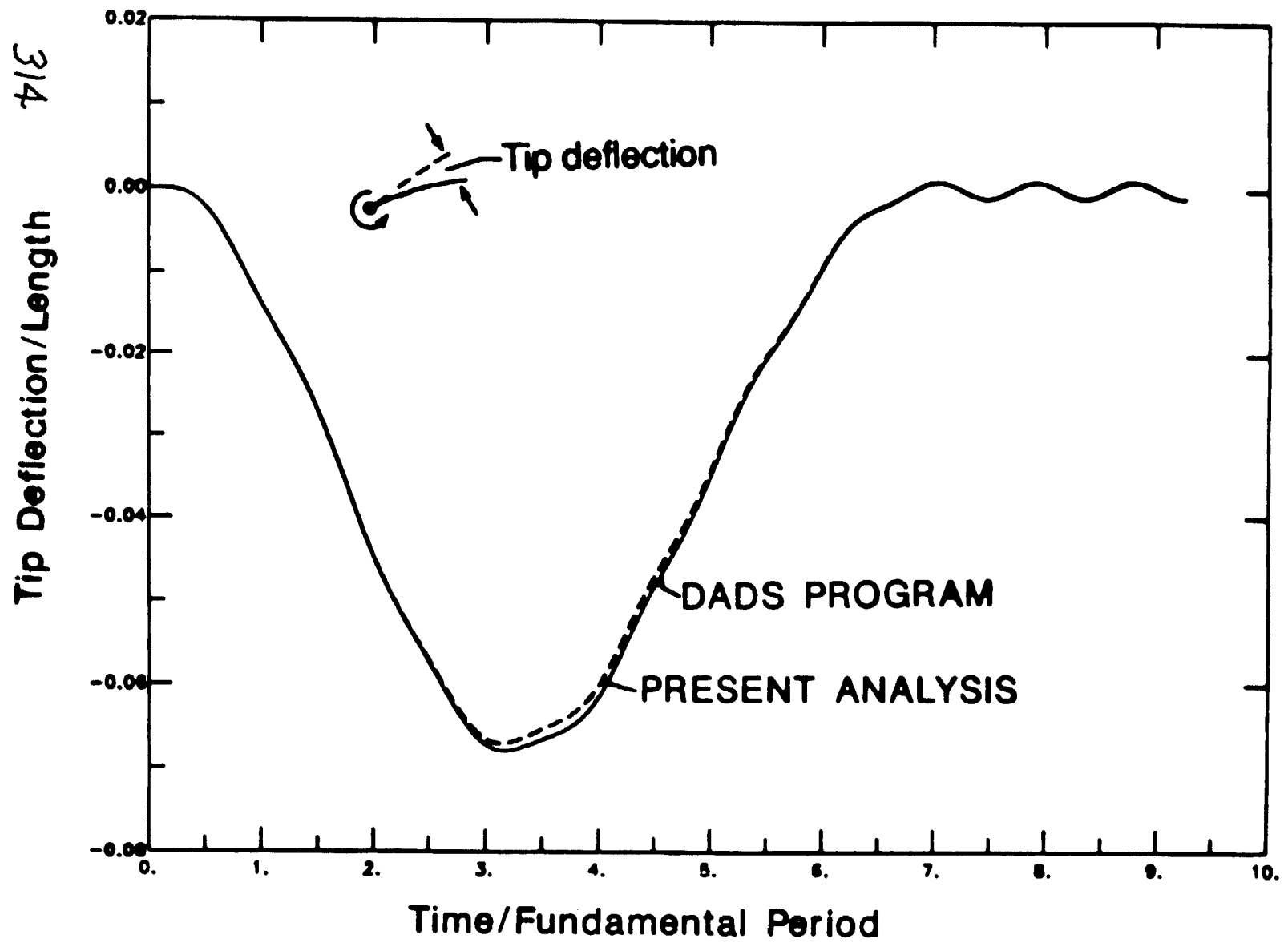
CONVERGENCE OF SOLUTION FOR FLEXIBLE BEAM SPIN-UP ON A PLA
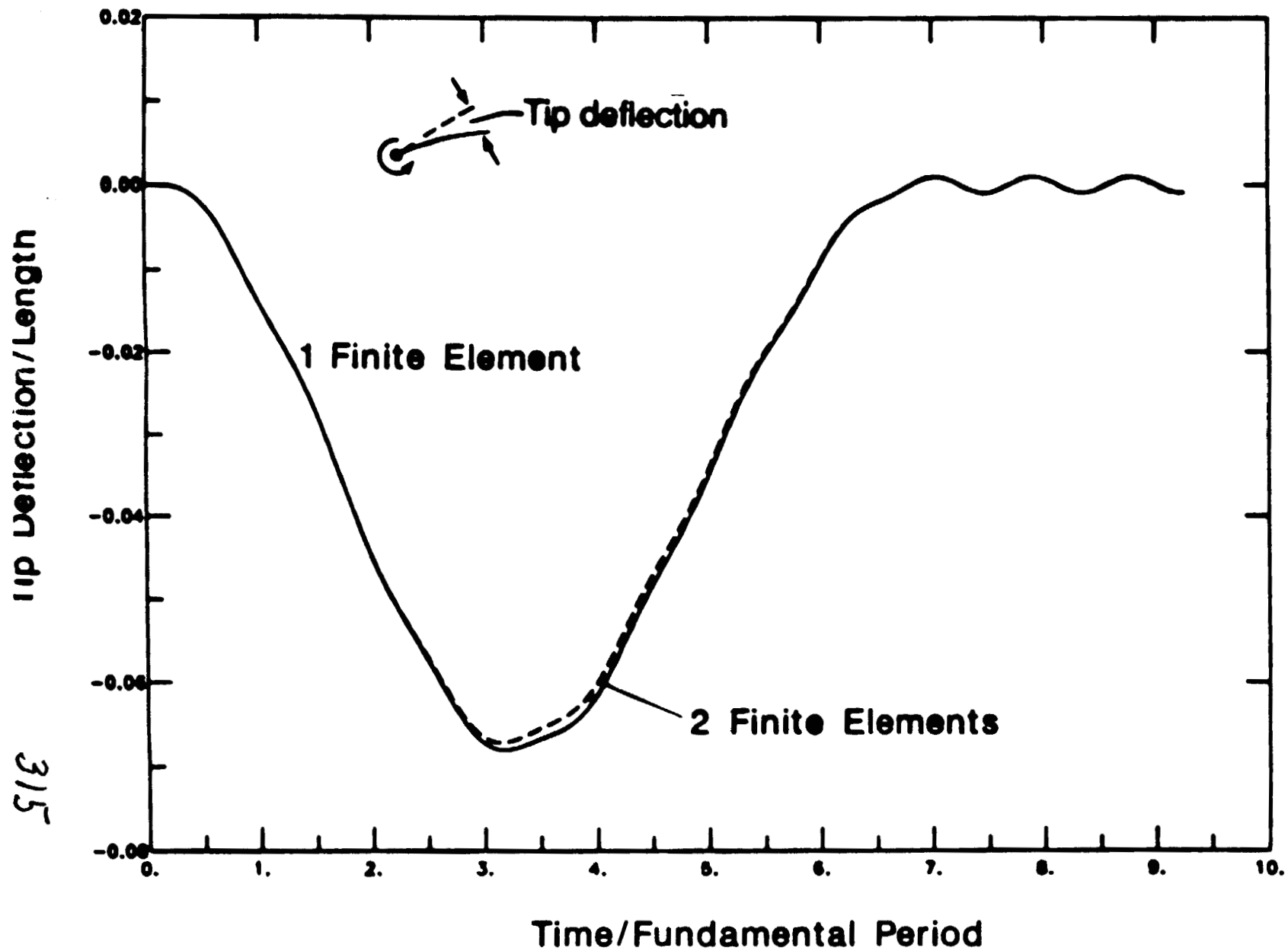
upper triangle

longeron

lower triangle

Apply
Driving Function

upper triangle

longeron

z

lower triangle
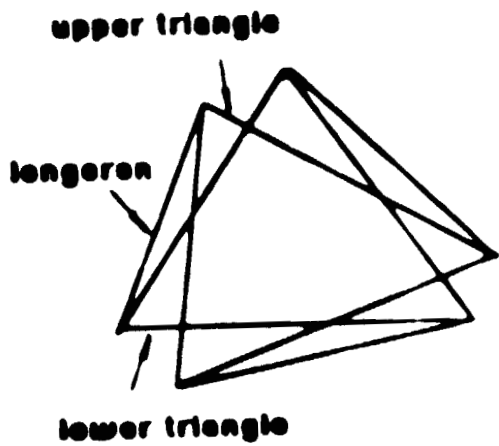
Fully Packed
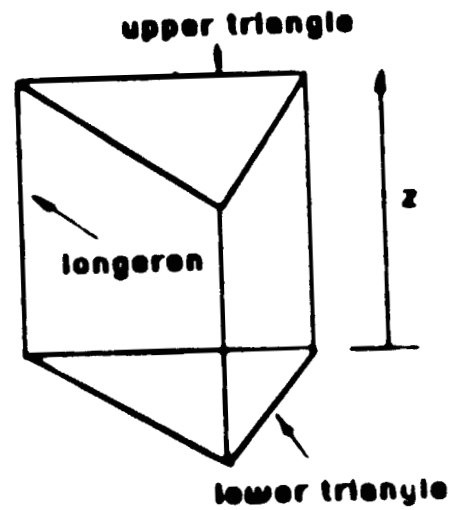
Fully Deployed
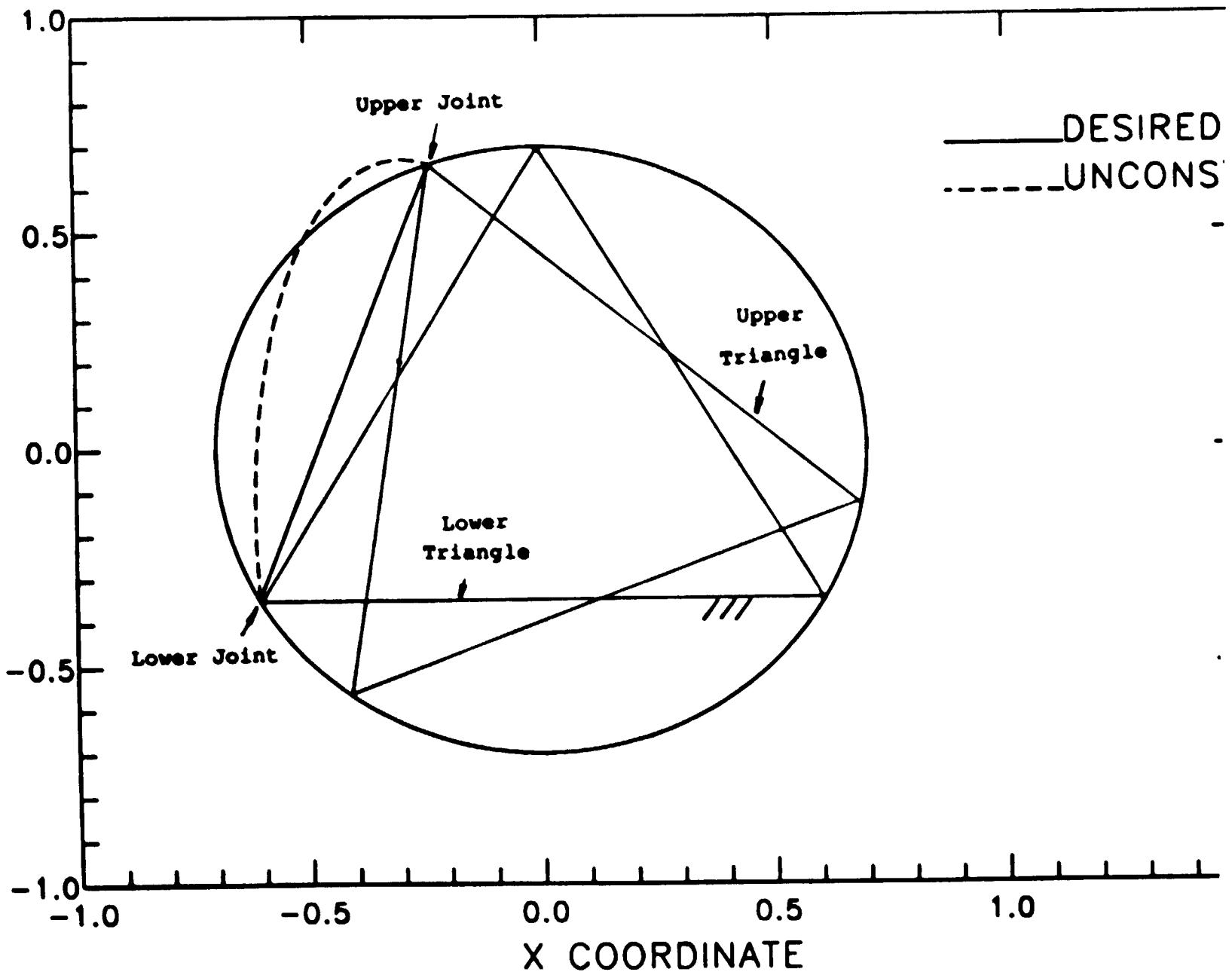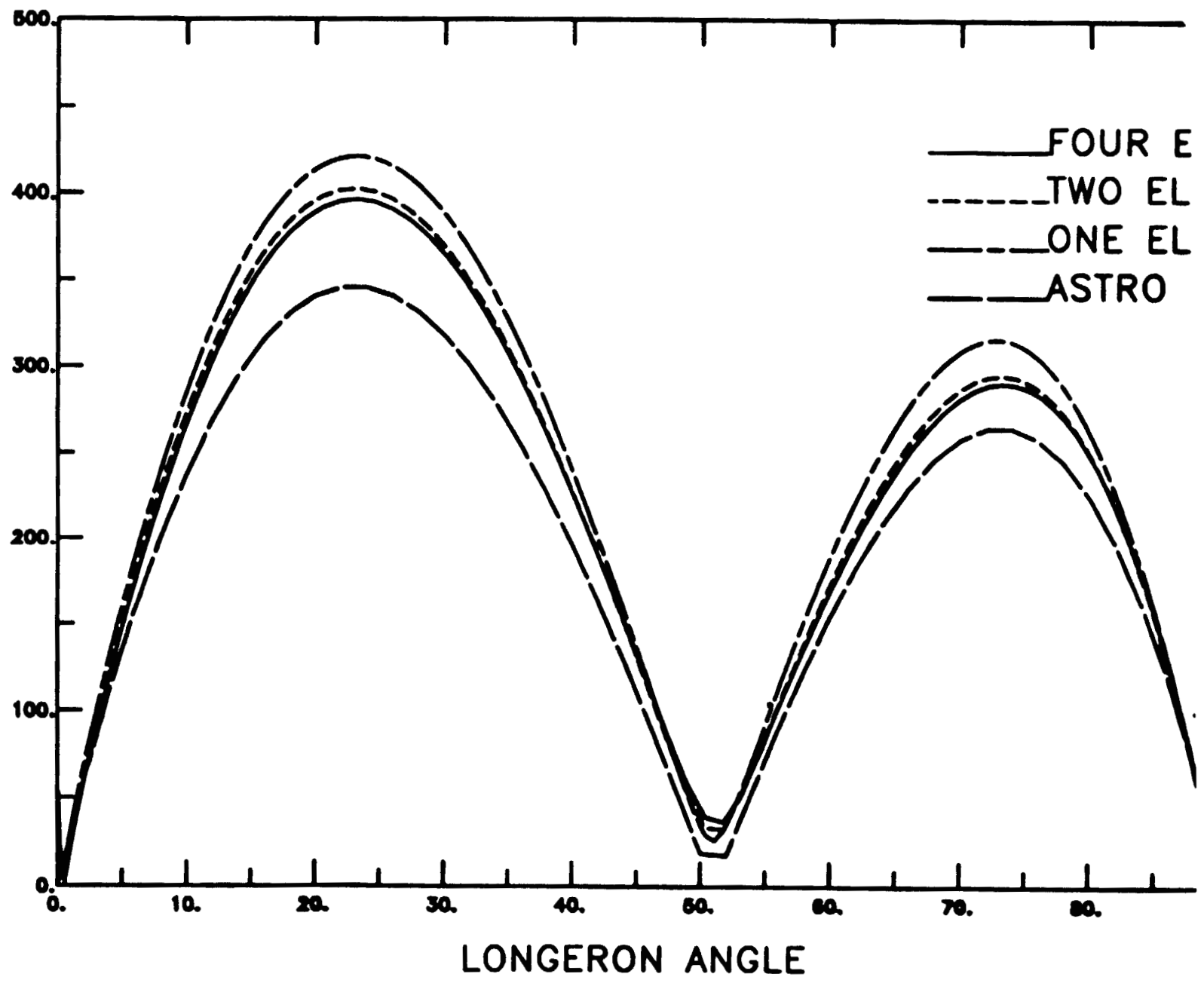
AT UPPER JOINT

RESULTANT BENDING MOMENT

LONGERON ANGLE

FOUR I
TWO EI
ONE EI
ASTRO

319
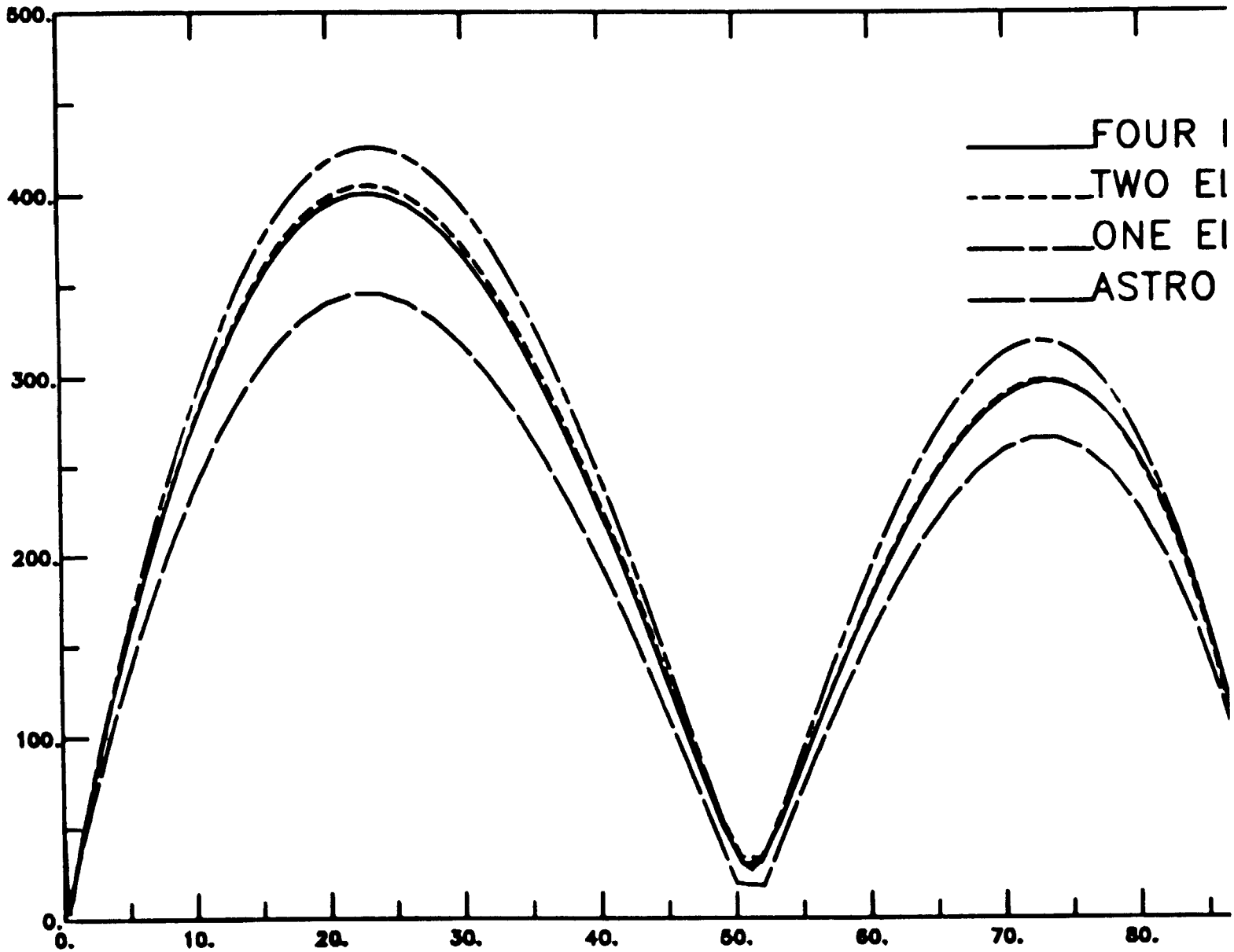
Future LATDYN

* various elements

* various joint connections

* various integrations

    (parallel version)

* control and structure

    interactions

320

Conclusions

* A finite-element-based research code is developed.

* It provides a modelling, calculation, and analysis tool for researcher & Engr.

* To analyze complex space structures and/or mechanisms.

* In the simulation of Control design as well as structural dynamics.

ENHANCED ELEMENT-SPECIFIC MODAL FORMULATIONS FOR FLEXIBLE MULTIBODY DYNAMICS

By

Robert R. Ryan
University of Michigan
Ann Arbor, Michigan

ABSTRACT

The accuracy of current flexible multibody formalisms based on assumed modes is examined in the context of standard spacecraft motions involving structural components undergoing both slow and fast overall translational and rotational motions as well as small deformations. Limitations of current techniques in treating (1) element-specific coupling behavior of large motion and small deformation, and (2) motion-induced structural stiffness variations, are noted.

The roles of nonlinear and linear elastic structural theories in accurately predicting transient large-displacement dynamic behavior of flexible multibody systems are examined in detail. Coupling effects between deformation and overall motion are carefully scrutinized in the context of assumed-mode discretization techniques. Consistently linearized beam, plate, and shell formulations involving in-plane stretch variables are proposed and shown to yield very accurate simulation results and extremely fast modal convergence for most motions involving small strains. In some particular cases, however, in which membrane stiffness dominates bending stiffness, a nonlinear strain formulation is required in order to capture proper coupling between deformation and overall motion. Unfortunately, with standard component modes, algorithmic formalisms involving nonlinear strain-displacement expressions show very slow modal convergence. A procedure involving use of constraint modes is proposed to alleviate this problem.

PRECEDING PAGE BLANK NOT FILMED

NASA – OAST WORKSHOP

Computational Aspects in the Control of Flexible Systems

ENHANCED ELEMENT-SPECIFIC MODAL FORMULATIONS

FOR

FLEXIBLE MULTIBODY DYNAMICS

July 12-14, 1988

Robert R. Ryan

University of Michigan

PRECEDING PAGE BLANK NOT FILMED

329
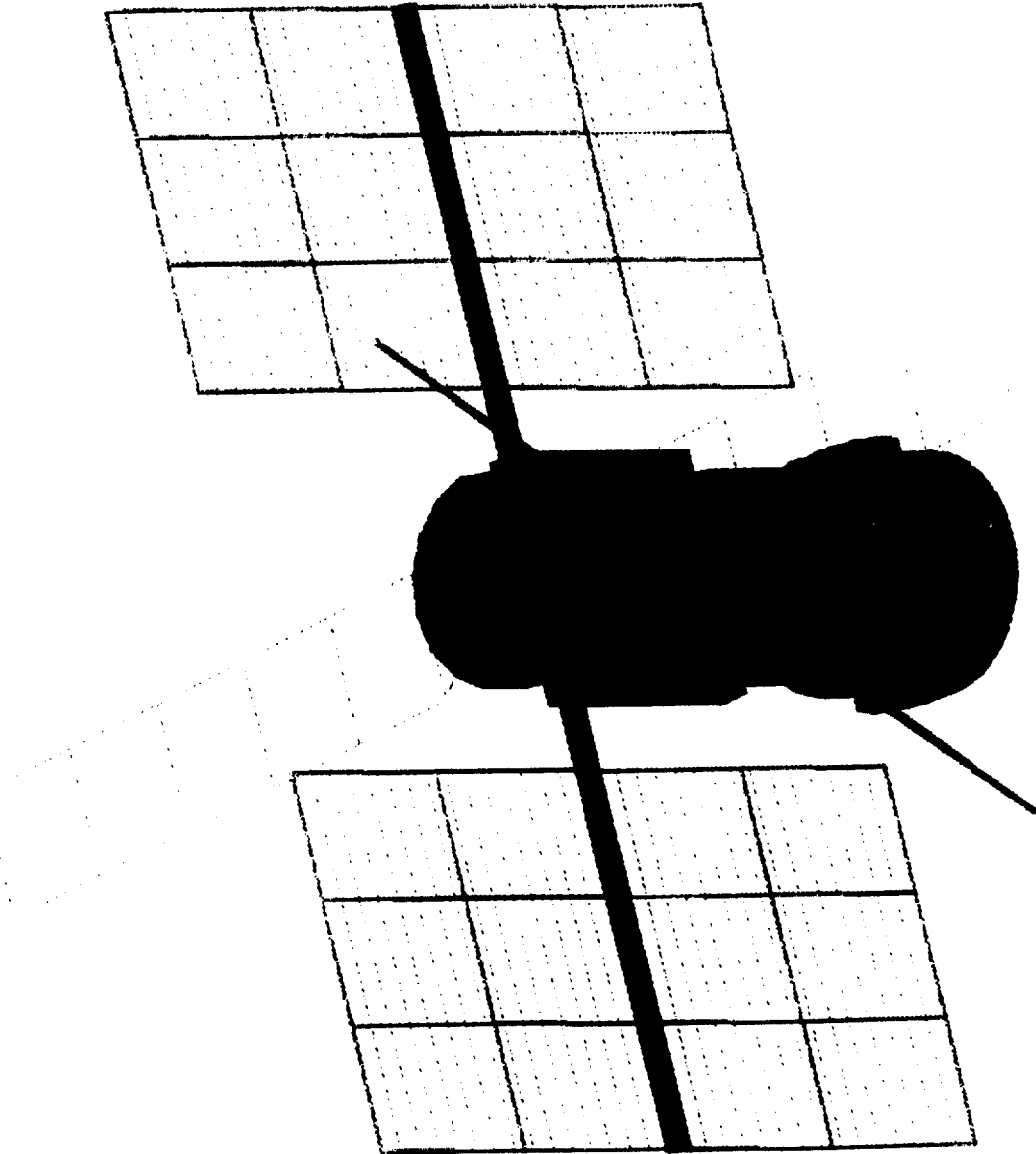
///////// Outline \\\\\\\\\

I. Limitations of Existing Flexible Multibody Formalisms
- Examples
- Verification

II. Linear and Nonlinear Element-Specific Formulations
- Consistently-Linearized Beam, Plate, Shell Multibody Models
- Second-Order Beam, Plate Models

III. Simulation Results
- Membrane/Bending Problems
- Convergence

Current Flexible Multibody Formalisms - Modal Approach
"Limitations"

- Do Not Account For Large-Displacement Element-Specific Behavior

- Inadequate Account of Motion-Induced Stiffness Variations

$$\lambda_i, \phi_{k,i} \qquad S_{ij} = \int_0^L \rho\phi_{1i}\phi_{1j}\,dx \qquad \text{*Solution*}$$
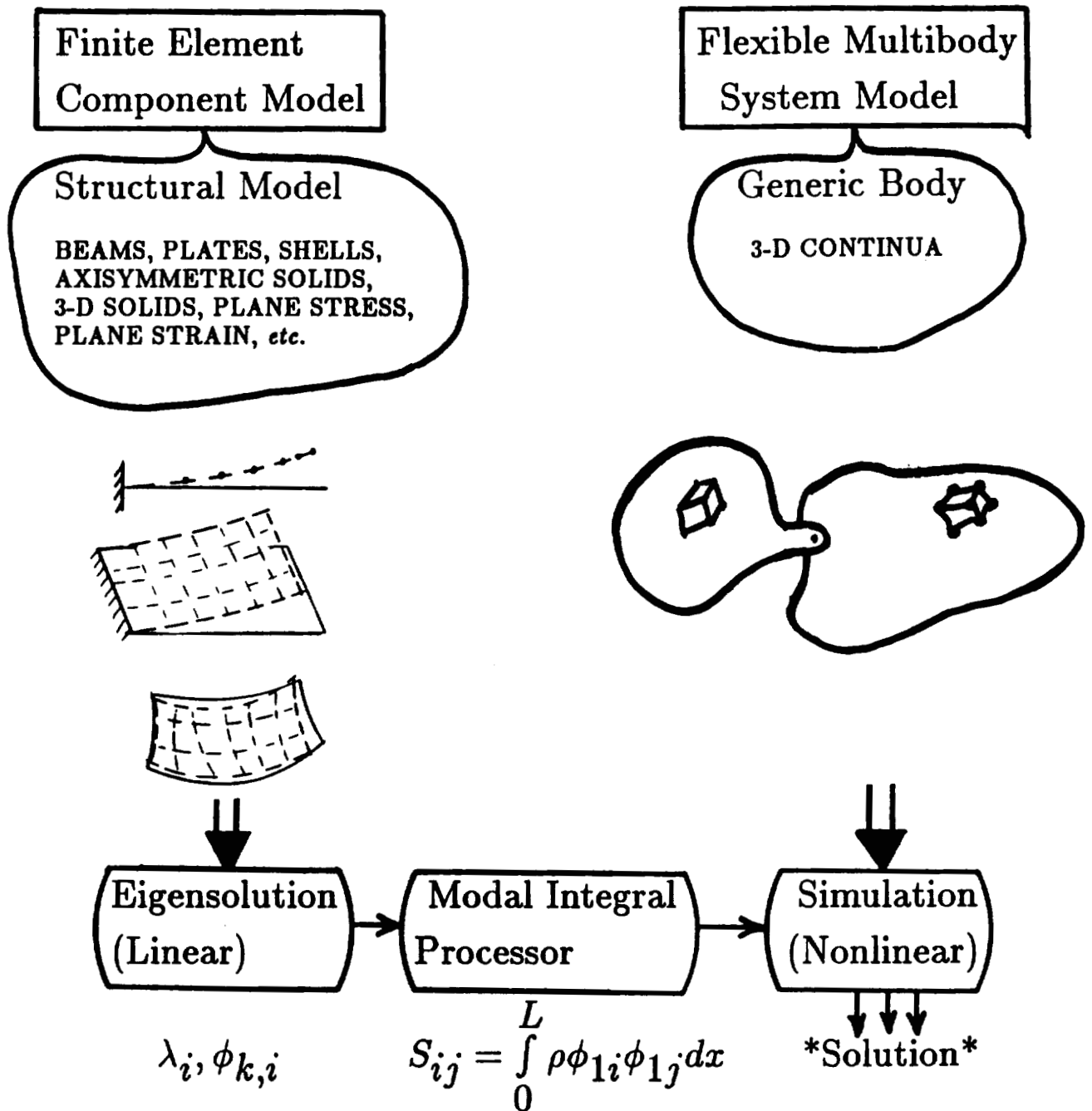
Slow Repositional Maneuver of Channel Beam
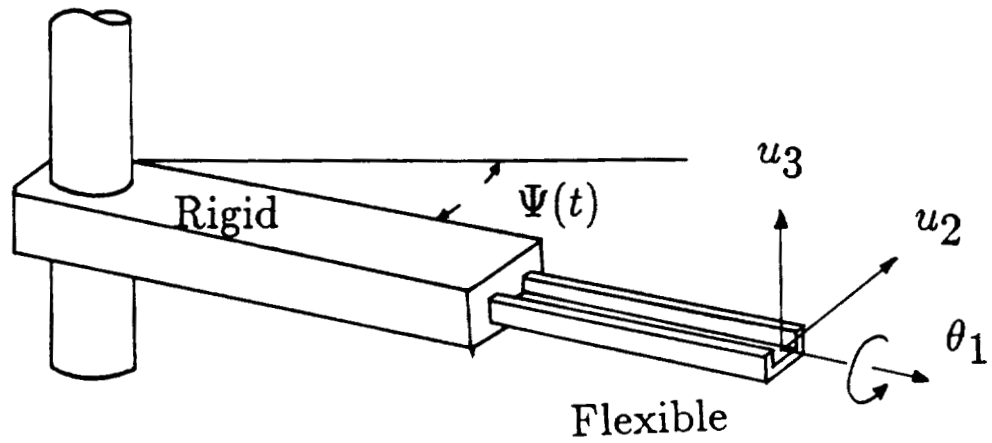
Repositional Maneuver Angle

$\Psi(t)$

0.0   15.0   30.0

Time (sec)

333
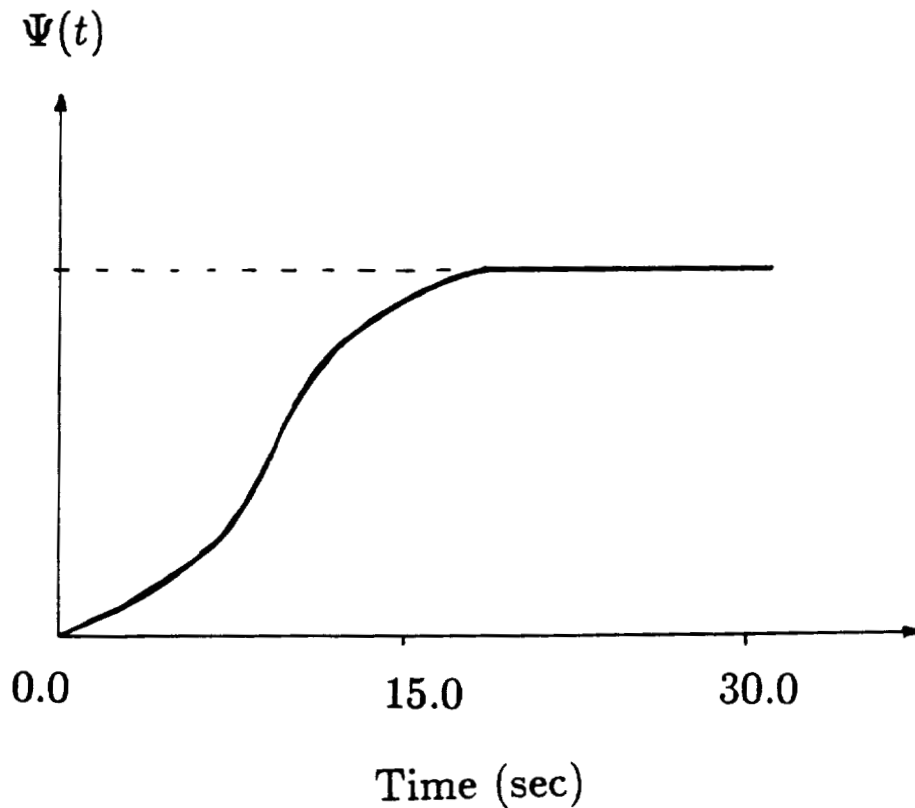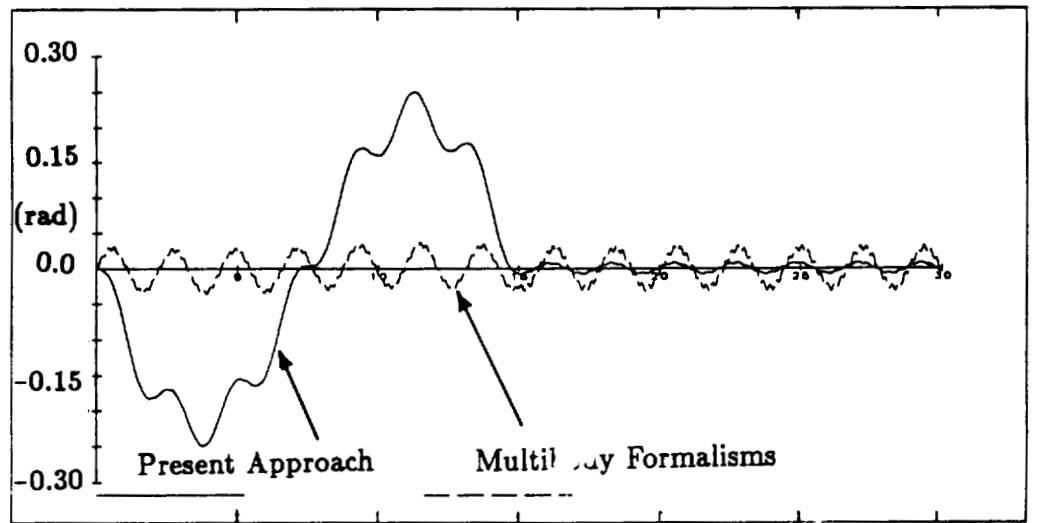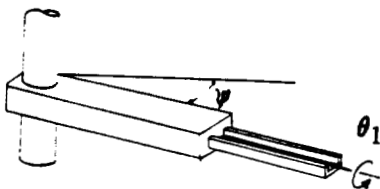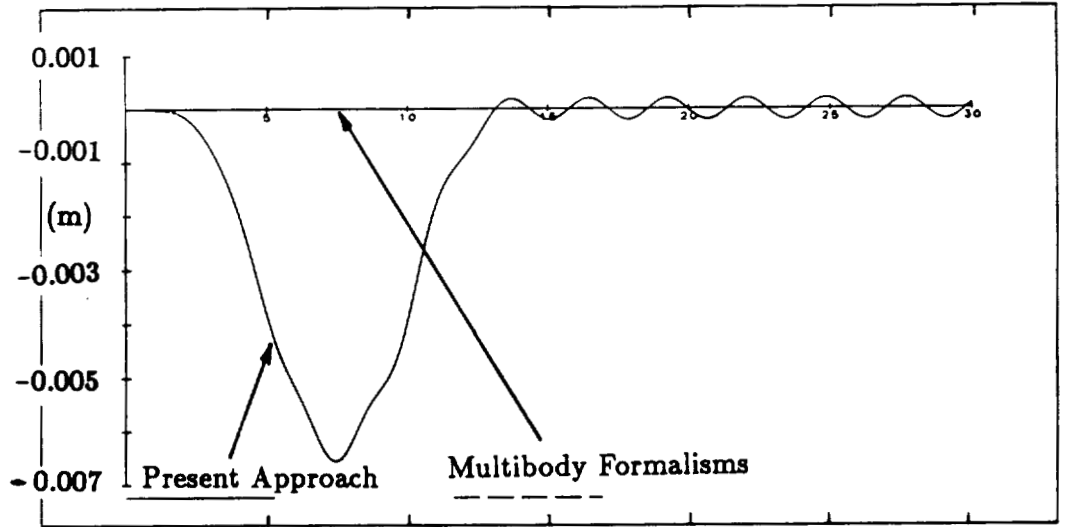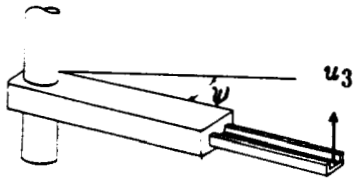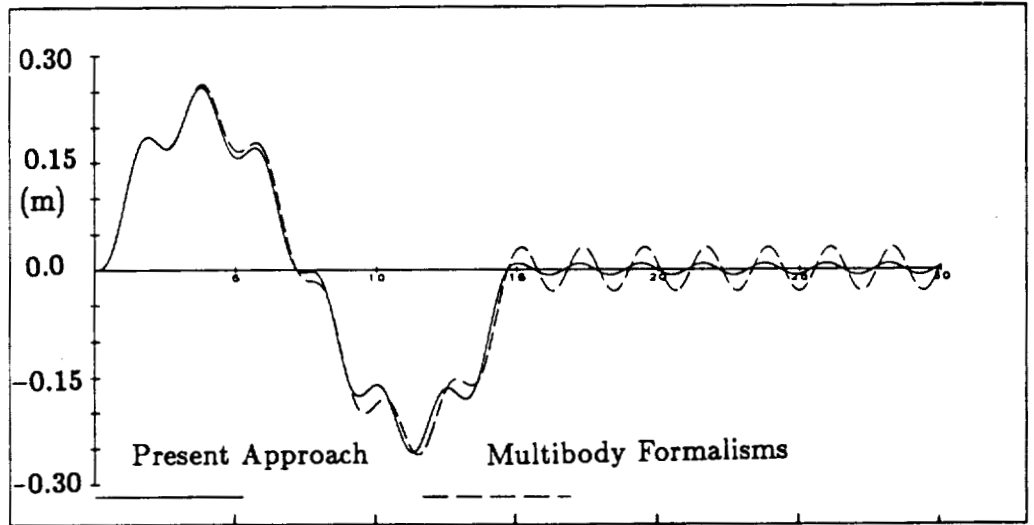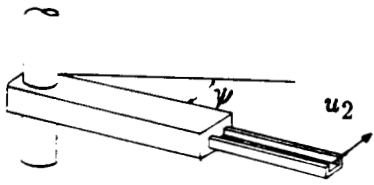
BEAM WITH OFFSET TIP MASS

Equipment

λ, ϕ

Tip Disp.

Time

Tip Disp.

Time

Different
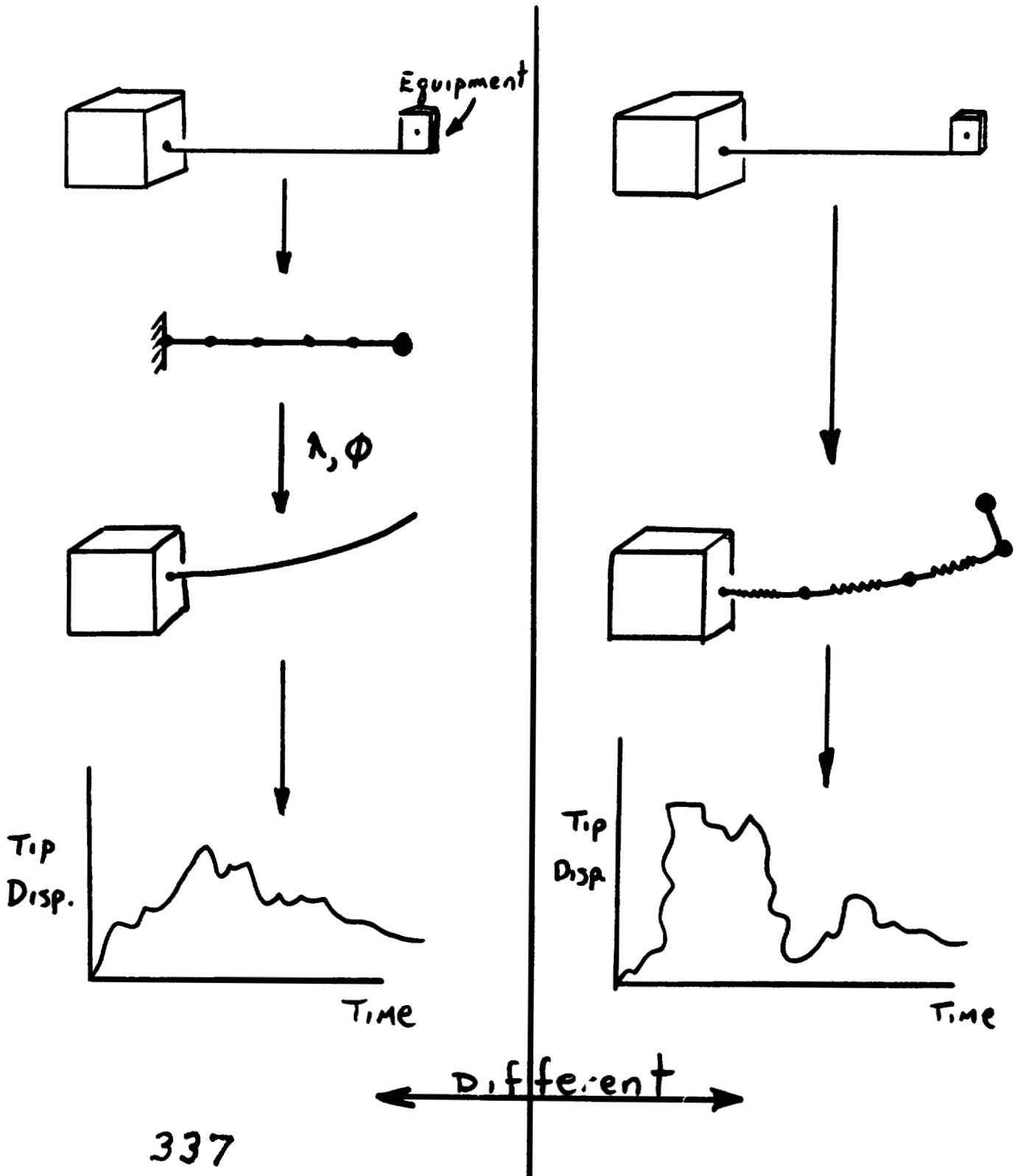
337

Constant Speed Spin — Buckling Analysis
Conventional Vs. Enhanced Modal Theory
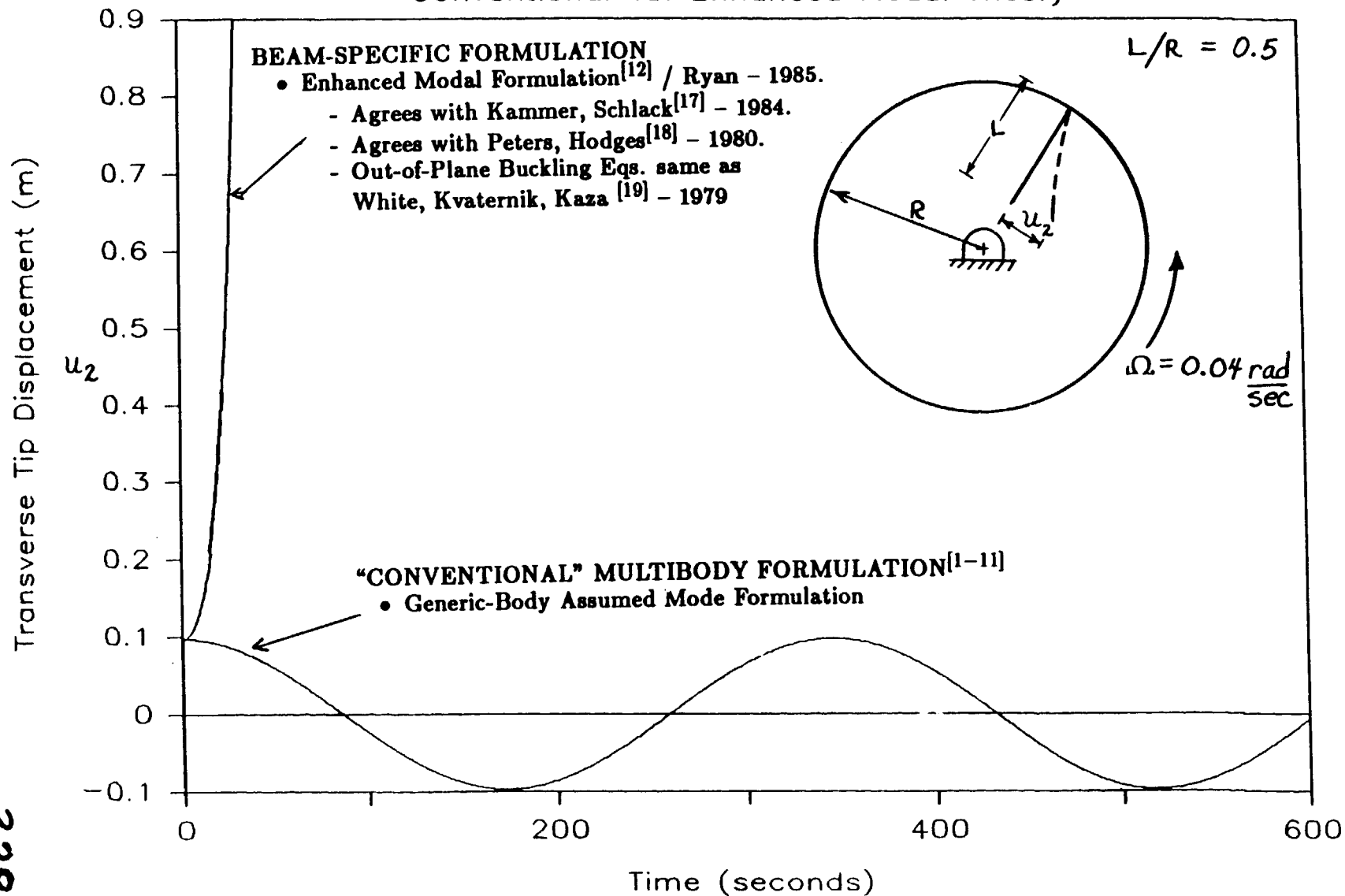
BEAM-SPECIFIC FORMULATION
- Enhanced Modal Formulation[12] / Ryan – 1985.
  - Agrees with Kammer, Schlack[17] – 1984.
  - Agrees with Peters, Hodges[18] – 1980.
  - Out-of-Plane Buckling Eqs. same as White, Kvaternik, Kaza [19] – 1979

$L/R = 0.5$

$\Omega = 0.04 \frac{rad}{sec}$

"CONVENTIONAL" MULTIBODY FORMULATION[1–11]
- Generic-Body Assumed Mode Formulation

Transverse Tip Displacement (m)

u_2

Time (seconds)

- Discrete Representations

- Nonlinear Finite Element Methods

- Linear and Nonlinear
  Enhanced Modal Approaches

Consistently-Linearized Multibody Structural Theories

- Beam

$$s \stackrel{\triangle}{=} u_1 + \frac{1}{2}\int_0^x \left[\left(\frac{\partial u_2}{\partial \sigma}\right)^2 + \left(\frac{\partial u_3}{\partial \sigma}\right)^2\right] d\sigma$$
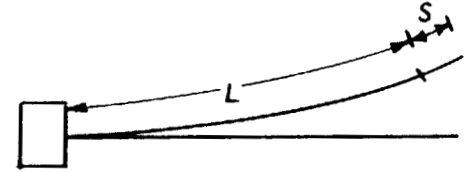
- Plate

$$r \stackrel{\triangle}{=} u_2 + \frac{1}{2}\int_0^y \left(\frac{\partial u_3}{\partial \eta}\right)^2 d\eta$$

$$s \stackrel{\triangle}{=} u_1 + \frac{1}{2}\int_0^x \left(\frac{\partial u_3}{\partial \xi}\right)^2 d\xi$$
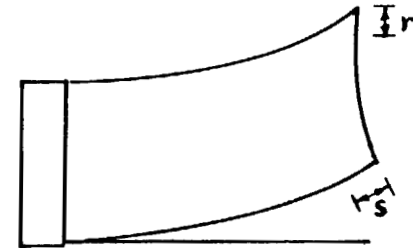
$$M\ddot{q} + G\dot{q} + (K_I + K_L + K_g)q = F$$

Advantages:
- Excellent Convergence
- Captures Motion-Induced Bending Stiffness Variation
- Ease of Modal Reduction/Controls
- Easily-Implemented – Linear in Deformation

Disadvantages:
- Doesn't Capture Motion-Induced Membrane Stiffness

Second-Order Structural Multibody Theories

- **Beam**

$$U_b = \frac{1}{2}\int_0^\ell E\left\{ I_{zz}\left(\frac{\partial^2 u_2}{\partial x^2}\right)^2 + I_{yy}\left(\frac{\partial^2 u_3}{\partial x^2}\right)^2 \right\}dx$$

$$U_s = \frac{1}{2}\int_0^\ell EA\left\{ \left[\left(\frac{\partial u_1}{\partial x}\right) + \frac{1}{2}\left(\frac{\partial u_2}{\partial x}\right)^2 + \frac{1}{2}\left(\frac{\partial u_3}{\partial x}\right)^2\right]^2 \right\}dx$$

- **Thin Rectangular Plates:**

$$U_b = \frac{1}{2}\int_0^b \int_0^a \beta\left\{ \left(\frac{\partial^2 u_3}{\partial x^2} + \frac{\partial^2 u_3}{\partial y^2}\right)^2 \right.$$

$$\left. -2(1-\nu)\left[\left(\frac{\partial^2 u_3}{\partial x^2}\right)\left(\frac{\partial^2 u_3}{\partial y^2}\right) - \left(\frac{\partial^2 u_3}{\partial x \partial y}\right)^2\right] \right\}dxdy$$

$$U_s = \frac{1}{2}\int_0^b \int_0^a \gamma\left\{ \left(\frac{\partial u_1}{\partial x}\right)^2 + \left(\frac{\partial u_1}{\partial x}\right)\left(\frac{\partial u_3}{\partial x}\right)^2 + \left(\frac{\partial u_2}{\partial y}\right)^2 \right.$$

$$+ \left(\frac{\partial u_2}{\partial y}\right)\left(\frac{\partial u_3}{\partial y}\right)^2 + \frac{1}{4}\left[\left(\frac{\partial u_3}{\partial x}\right)^2 + \left(\frac{\partial u_3}{\partial y}\right)^2\right]^2$$

$$+ 2\nu\left[\left(\frac{\partial u_1}{\partial x}\right)\left(\frac{\partial u_2}{\partial y}\right) + \frac{1}{2}\left(\frac{\partial u_2}{\partial y}\right)\left(\frac{\partial u_3}{\partial x}\right)^2 + \frac{1}{2}\left(\frac{\partial u_1}{\partial x}\right)\left(\frac{\partial u_3}{\partial y}\right)^2\right]$$

$$+ \frac{(1-\nu)}{2}\left[\left(\frac{\partial u_1}{\partial y}\right)^2 + 2\left(\frac{\partial u_1}{\partial y}\right)\left(\frac{\partial u_2}{\partial x}\right) + \left(\frac{\partial u_2}{\partial x}\right)^2\right.$$

$$\left.\left. + 2\left(\frac{\partial u_1}{\partial y}\right)\left(\frac{\partial u_3}{\partial x}\right)\left(\frac{\partial u_3}{\partial y}\right) + 2\left(\frac{\partial u_2}{\partial x}\right)\left(\frac{\partial u_3}{\partial x}\right)\left(\frac{\partial u_3}{\partial y}\right)\right]\right\}dxdy$$

$$M\ddot{q} + G\dot{q} + (K_I + K_L + K_n)q = F$$

Advantages:
- Captures Important Motion-Induced Bending AND Membrane Stiffness Variations for Small Strain

Disadvantages:
- Poor Convergence With Standard Modes
- Order Reduction Results in Very Inaccurate Models
- Very Costly to Incorporate

SMOOTH SPIN-UP MOTION (0 - 6 rad/sec in 15 seconds)

Transverse
Tip
Displacement
(meters)

u_2 (m)

2nd Order Nonlinear

Correct

-5.E-1

Consistently-linearized

Linear

-1

0   10   Time (seconds)   20   30
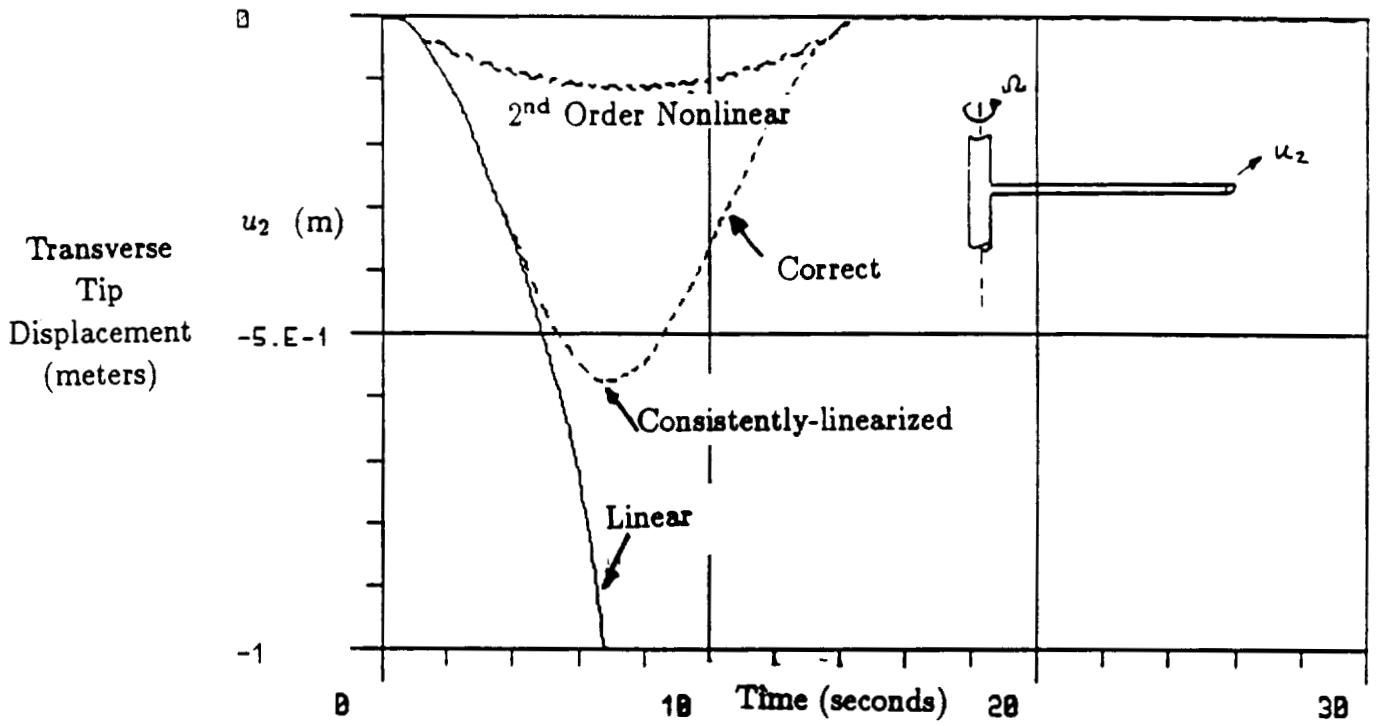
SMOOTH SPIN-UP MOTION (0 - 6 rev/min in 30 seconds)

2.5

Midpoint
Transverse
Displacement
(inches)

u_3 (in)

2

Linear

1.5

Consistently-linearized

1

Correct

5.E-1

0

2nd Order Nonlinear

-5.E-1

0   20   Time (seconds)   40   60

345
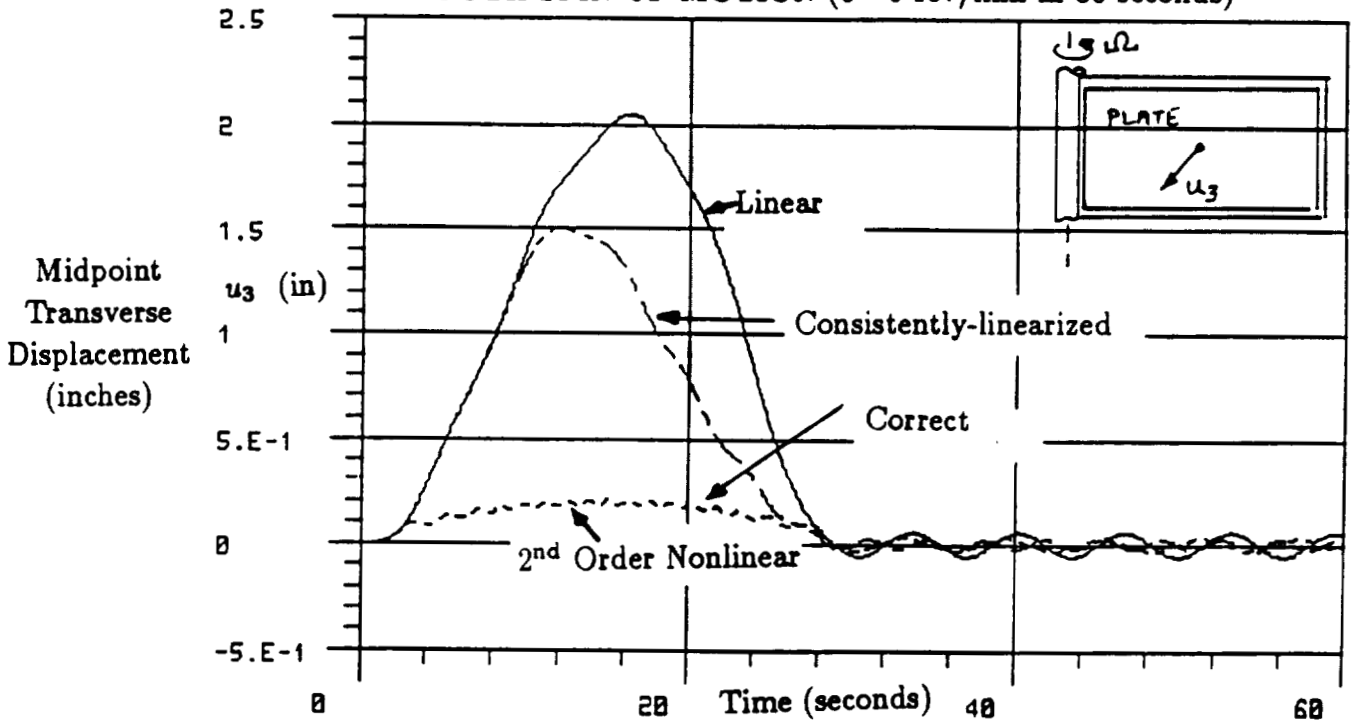
SIMPLY SUPPORTED RECTANGULAR PLATE SPIN-UP MOTION
Assumed Mode Approach with Nonlinear Strain Expression

Transverse Deflection of Center of Plate - Smooth Spin-up From 0 - 6 rad/sec in 15 seconds
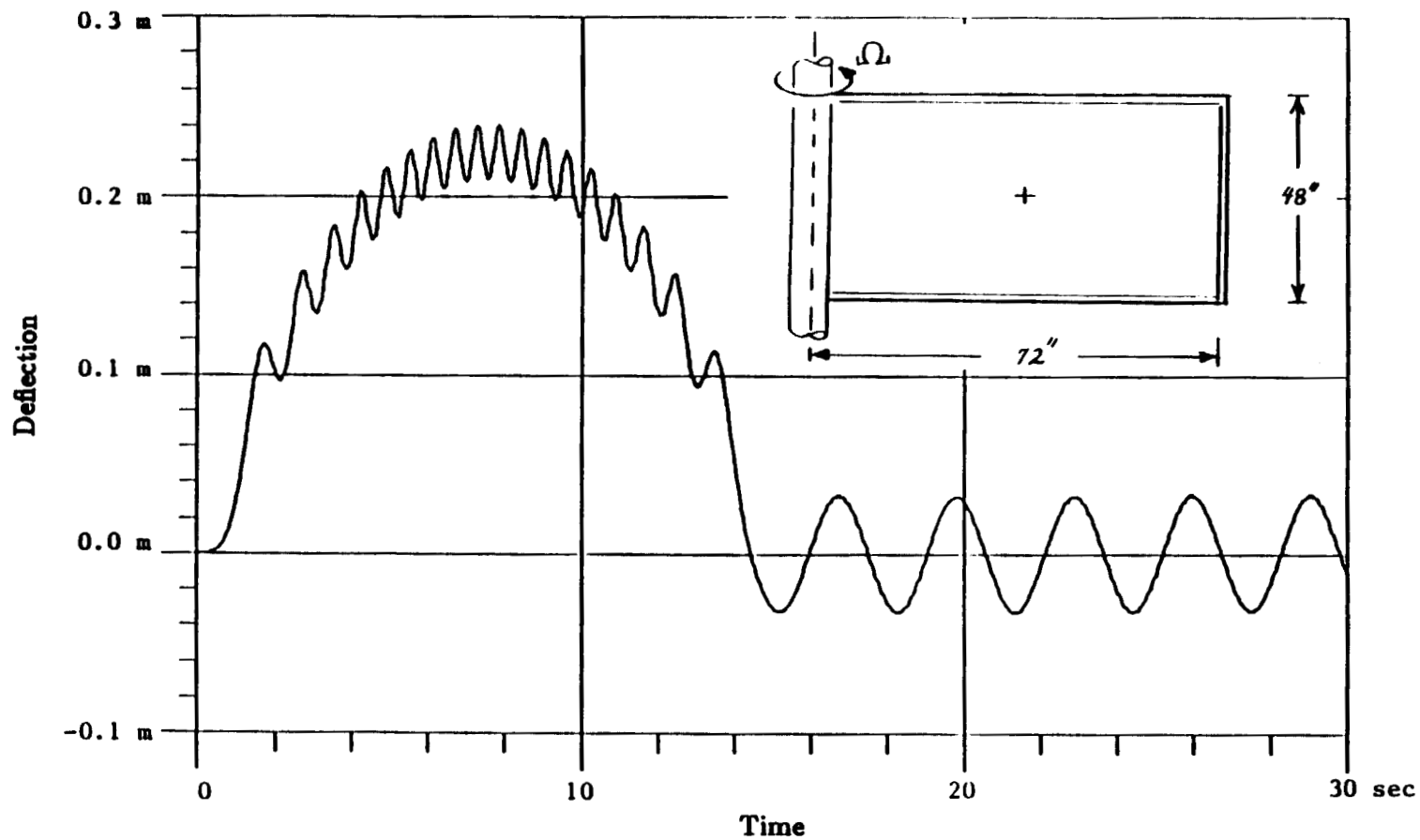
Results obtained with 3 Assumed Stretch Modes and 3 Assumed Bending Modes

Static Analysis of a Square Plate with Uniform Pressure Distribution Considering only Membrane Stiffness

Uniform Pressure p

All Four Edges
Simply Supported
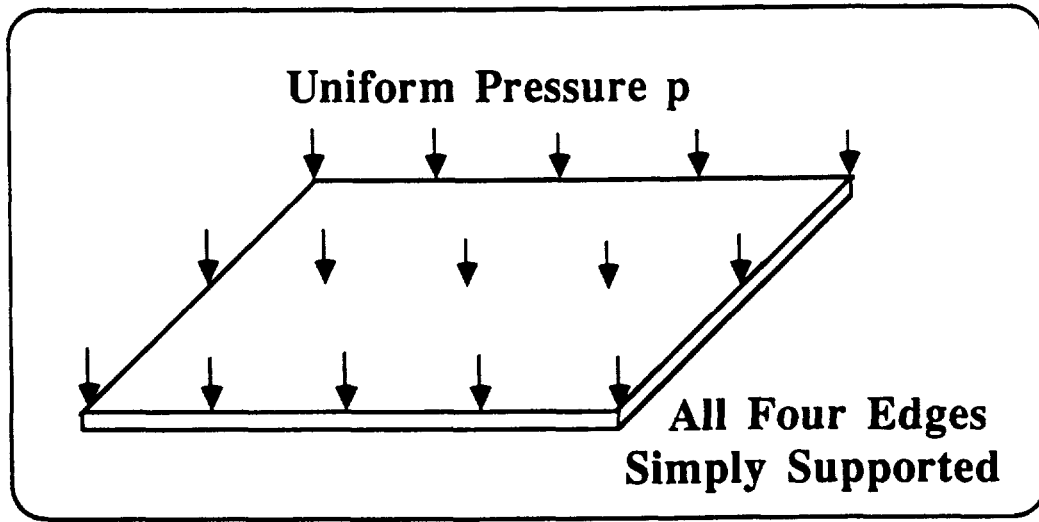
Maximum inertia force per area in the middle
of the plate during the spin-up motion
is used as uniform pressure distribution.

Results of Maximum Lateral Deflection

ω \ case	Static	Dynamic	
		Present	Conventional
$\pi/5$	0.18"	0.21"	2.1"
π	0.32"	0.37"	Divergence

Fig. 19 – Static Deflections under High Pressure Loads

347

CONCLUSIONS

- Existing Flexible Multibody Formalisms Are Limited in Their Ability to Treat Coupled Large Displacement/Small Deformation.

- Alternative Approaches Include Taking Explicit Account of Constraints Geometrically or Within a Nonlinear Strain Measure.

- Consistently-Linearized Models Work Well For Most Problems But Cannot Capture Motion-Induced Membrane Stiffness Variations.

- Second Order Structural Theories Account for Motion-Induced Stiffness Variations But Converge Slowly With Standard Mode Functions.

- Nonlinear Constraint Functions Serve Well as Modal Functions in Order to Improve Convergence in Second Order Structural Theories.

EFFICIENCY AND CAPABILITIES OF MULTI-BODY SIMULATIONS

R.J. VanderVoort
DYNACS Engineering Co., Inc.
Clearwater, Fl

ABSTRACT

Simulation efficiency and capability go hand in hand. The more capability you have the lower the efficiency will be. ~~Section 1 of this paper discusses~~ The efficiency and ~~section 2 deals with~~ capabilities. The lesson ~~we have~~ learned about generic simulation is: Don't rule out any capabilities at the beginning but keep each one on a switch so it can be bypassed when warranted by a specific application.

1. EFFICIENCY

Efficiency means different things to different people. For the person running simulations interactively on a terminal quick turn around time is efficiency. For the person making 10,000 Monte-Carlo runs low cost is efficiency. For the person running real time simulations minimum CPU time is efficiency.

Three aspects of a simulation should be considered when dealing with efficiency; hardware, software and modeling.

Hardware A fast processor will reduce CPU time for a given simulation but this doesn't necessarily equate to improved efficiency. For example, the Monte-Carlo simulation may take 10 minutes on a super computer and 2 weeks on a PC but if time is free on the PC then that may be an efficient solution. We will not discuss hardware related issues except for two points. 1.) Fast hardware is of primary importance to the real time simulation because it means higher fidelity models can be incorporated 2.) Vector processors and parallel processors should use custom algorithms that take full advantage of the special machine architecture.

Software A fast algorithm will also reduce CPU time but again this doesn't necessarily equate to improved efficiency. For example, it is generally accepted that an ad-hoc simulation is much faster than a generic simulation. The cost of developing and testing the ad-hoc simulation may exceed the run time saving thereby reducing overall efficiency.

Recent work in the area of symbolic programming has shown that significant savings can be achieved by symbolicaly forming the equation of motion and numerically solving them. Other algorithms have been proposed that promise similar savings. There is one point that software developers should keep in mind. With generic simulations the user must have complete flexibility in retaining or deleting different parts of his model. This is because generic simulations are often used for model development and validation. In that environment an analyst will add or delete certain features to determine the effect on performance and whether or not the feature should be retained in the model.

More on this subject in section 2.

Modeling This is the domain of the simulation user and the area in which many improvements in efficiency can be made. For example, deleting a high order mode in a flexible body model has a compound effect. It reduces the model complexity and at the same time allows a bigger integration step size both of which reduce run time. Often times the reduced fidelity is justified by the savings in run time.

The point to be made is that the analyst is the end authority on the "correct" model for a given application. The more flexibility he has in changing his model the easier it is for him to select the best model for the job.

2. CAPABILITIES

Capability in our context is synonymous with flexibility and not with complexity. A simulation may be very detailed and complex but if it can't be changed then it's only useful in a narrow range of applications and has limited capability.

In our experience with TREETOPS and DCAP we have found that it is much easier to generate a model and obtain a response than it is to predict the correct response. In other words, when we don't get the expected response the simulation is usually correct and our expectation is wrong. This is not entirely unexpected because it is very difficult, even for an expert, to solve the equations of anything but the simplest dynamical systems. The solution to this dilemma is flexibility. Start with simple models that have known analytic solutions. Then add complexity one step at a time while gaining confidence in your model and insight into the behavior of your system.

For multibody systems with flexible bodies the same arguments apply but the complexity of the model increases more rapidly than for rigid bodies. The person doing software development makes assumptions that simplify the resulting equations of motion. If this is done carelessly then terms are dropped that may prove essential in specific applications. On the other hand, if simplifications are not made then the computation burden becomes too great.

The lesson we learned is that you must retain as many terms as possible in the kinematics but they must have associated switches so you can easily add or delete them from a specific application. This is done for two reasons. 1.) to give you insight into the effect of various model elements on system response and 2.) to allow the selection of the most efficient model for a given application.

Dynacs

352

HARDWARE

SOFTWARE

MODELING
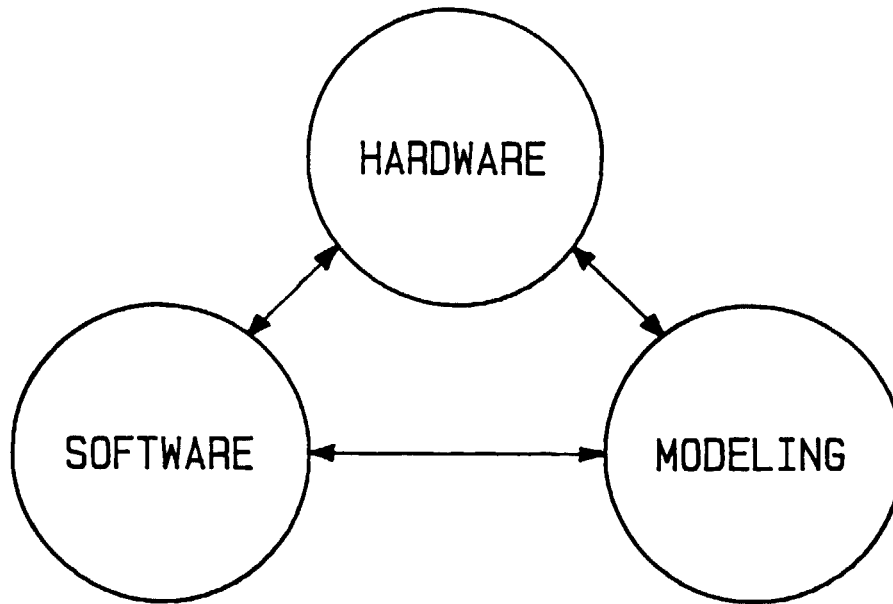
- BYPASS TERMS
- MULTI-RATE ALGORITHMS
- SYMBOLIC PROGRAMMING
- ADHOC SIMULATION
-

- BYPASS TERMS
- INTEGRATION TYPE & STEP SIZE
- REDUCED ORDER
-
-

Dynacs

353

	COORDINATE TRANSFORMATIONS	MASS MATRIX FORMULATION (M)	NON-LINEAR TERMS $(\underline{\omega} \times \underline{\underline{I}} \cdot \underline{\omega})$	CONSTRAINT FORMULATION (A)
PERFORM ALL COMPUTATIONS				
COMPUTE ONLY ON FIRST PASS OF R-K INTEGRATION				
COMPUTE ONLY ON FIRST PASS OF NTH R-K STEP				
BYPASS COMPUTATIONS				

Dynacs

354

	EQUATION FORMULATION	EQUATION SOLUTION	PROCESSING HARDWARE
CURRENT STATUS	NUMERIC	NUMERIC	SERIAL
	⋁		
FIRST STEP	SYMBOLIC	NUMERIC	SERIAL
			⋁
SECOND STEP	SYMBOLIC	NUMERIC	PARALLEL

$$M\ddot{q} = f + A^{\tau}\lambda$$

$$A\dot{q} = B$$

SIMULATION CAPABILITY-MENUS

Dynacs

BODIES	SENSORS	ACTUATORS	CONSTRAINTS	DEVICES	CONTROLLERS
1. RIGID	1. RATE GYRO	1. REACTION JET	1. CLOSED LOOP	1. SPRINGS	1. CONTINUOUS
2. FLEXIBLE	2. RESOLVER	2. HYDRAULIC CYLINDER	2. VELOCITY-TIME	2. DAMPERS	2. DISCRETE
	3. ANGULAR ACCELEROMETER	3. REACTION WHEEL	3. VELOCITY -DIRECTION	3. COULOMB DAMPER	3. BLOCK DIAGRAM (FREQUENCY DOMAIN)
	4. VELOCITY	4. TORQUE MOTOR	4. RATE-TIME	4. QUADRATIC SPRING/DAMPER	4. MATRIX (STATE SPACE)
	5. POSITION		5. RATE-DIRECTION	5. SOLID DAMPER	
	6. ACCELEROMETER	5. MOMENT	6. CUT JPINT	6. HARDSTOP	5. USER
	7. TACHOMETER	6. BRAKE		7. CONTACT SPRINGS	
	8. INTEGRATING RATE GYRO	7. LOCK			
	9. SUN SENSOR	8. SINGLE GIMBAL CMG			
	10. STAR SENSOR	9. DOUBLE GIMBAL CMG			
	11. IMU	10. MAGNETIC			
	12. POSITION VECTOR				
	13. VELOCITY VECTOR				

Dynacs

356

- ● **HIGH LEVEL**

    — LUMPED MASS SWITCH

- ● **MID LEVEL**

    — FIRST ORDER SWITCH
    — SECOND ORDER SWITCH
    — THIRD ORDER SWITCH

- ● **LOW LEVEL**

    — ONE SWITCH FOR EACH TERM

Dynacs

BODY j

$\underline{b}^j_{o2}$

BODY j
REFERENCE
FRAME
(b^j_o)

$\underline{b}^j_{o1}$

$\underline{b}^j_{o3}$

$\underline{r}^{j}_{o}$

$\underline{U}^{j}_{o}$

NODAL BODY j_o
REFERENCE
FRAME
$(b^j_{o_o})$

"NODAL BODY" j_o

NODAL BODY j_o
MASS CENTER

$\underline{\rho}^{j}_{o}$

$\underline{\rho}_o$

dm

ELEMENTAL
MASS

$R^{b^j_o}$

R^{j}

$\underline{j}$

INERTIAL
REFERENCE
FRAME

$\underline{i}$

$\underline{k}$

$$f_K + f_K^* = 0 \quad ; \quad K = 1, \dots, N$$

$$f_K = \sum_{\delta=1}^{NB} \left\{ \int_{\delta} \underline{V}_K^{\dot\delta} \cdot d\underline{f}^{\dot\delta} \right\}$$

$$-f_K^* = \sum_{\delta=1}^{NB} \left\{ \sum_{0=1}^{NNB^{\dot\delta}} \int_{\delta_0} \underline{V}_K^{\dot\delta} \cdot \underline{\ddot{R}}^{\dot\delta} dm \right\}$$

FROM NEWTON'S LAW

$$\int_{b_{\dot\delta}} d\underline{f}^{\dot\delta} - \underline{\ddot{R}}^{\dot\delta} dm = \underline{0}$$

NOTE: OPEN DOT DENOTES LOCAL TIME DERIVATIVE, SOLID DOT DENOTES INERTIAL TIME DERIVATIVE.

358

$$\underline{V}_K^{\dot\delta} = \underline{V}_K^{b\dot\delta} + \underline{\omega}_K^{\dot\delta} \times (\underline{r}_0^{\dot\delta} + \underline{u}_0^{\dot\delta} + \underline{\rho}_0) + \underline{V}_K^{\dot{\dot\delta}\dot\delta}$$

$$\underline{\ddot{R}}^{\dot\delta} = \underline{\ddot{R}}^{b\dot\delta} + \underline{\dot{\omega}}^{\dot\delta} \times (\underline{r}_0^{\dot\delta} + \underline{u}_0^{\dot\delta} + \underline{\rho}_0) + \underline{\overset{\circ\circ}{u}}_0^{\dot\delta} + \underline{\overset{\circ\circ}{u}}_0^{\prime\dot\delta} \times \underline{\rho}_0 + 2\underline{\dot{\omega}}^{\dot\delta} \times \underline{\overset{\circ}{u}}_0^{\dot\delta}$$

$$+ (\underline{\dot{\omega}}^{\dot\delta} \times \underline{\overset{\circ}{u}}_0^{\prime\dot\delta}) \times \underline{\rho}_0 + \underline{\omega}^{\dot\delta} \times (\underline{\omega}^{\dot\delta} \times (\underline{r}_0^{\dot\delta} + \underline{u}_0^{\dot\delta} + \underline{\rho}_0))$$

$$+ \underline{\omega}^{\dot\delta} \times (\underline{\overset{\circ}{u}}_0^{\prime\dot\delta} \times \underline{\rho}_0) + \underline{\overset{\circ}{u}}_0^{\prime\dot\delta} \times (\underline{\omega}^{\dot\delta} \times \underline{\rho}_0) + \underline{\overset{\circ}{u}}_0^{\prime\dot\delta} \times (\underline{\overset{\circ}{u}}_0^{\prime\dot\delta} \times \underline{\rho}_0)$$

359

TRANSLATIONAL D.O.F.

LET K CORRESPOND TO THE i-TH TRANSLATIONAL D.O.F. OF THE q-TH HINGE.

$$\underline{V}_K^{b\delta} = \begin{cases} \underline{g}_i^q & ; \delta \in E(q) \\ \underline{0} & ; OTHERWISE \end{cases}$$

$$\underline{\omega}_K^\delta = \underline{0}$$

$$\underline{V}_K^{\ddot{q}\delta} = \underline{0}$$

ROTATIONAL D.O.F.

LET K CORRESPOND TO THE i-TH ROTATIONAL D.O.F. OF THE q-TH HINGE.

$$\underline{V}_K^{b\delta} = \begin{cases} \underline{\ell}_i^q(\theta^q) \times \left[\underline{q}\,\underline{x}^\delta - (\underline{r}_{hq} + \underline{u}_{hq}) \right] & ; \delta \in E(q) \\ \underline{0} & ; OTHERWISE \end{cases}$$

$$\underline{\omega}_K^\delta = \begin{cases} \underline{\ell}_i^q(\theta^q) & ; \delta \in E(q) \\ \underline{0} & ; OTHERWISE \end{cases}$$

$$\underline{V}_K^{\ddot{q}\delta} = \underline{0}$$

MODAL D.O.F.

LET K CORRESPOND TO THE i-TH MODAL D.O.F. OF THE q-TH BODY.

$$\underline{V}_K^{b\delta} = \begin{cases} -\underline{\phi}_i^{\prime q}(\underline{r}_{hq}) \times {}^{hq}\underline{z}^{b\delta} + \underline{\phi}_i^{\prime q}(\underline{r}_{p(q)}) \times {}^{p(q)}\underline{S}^{b\delta}(1-\delta_{iq}) \\ \qquad + \underline{\phi}_i^q(\underline{r}_{hq}\,)(1-\delta_{iq}) - \underline{\phi}_i^q(\underline{r}_{hq}) \quad ; \delta \in E(q) \\[4pt] \underline{0} \quad ; OTHERWISE \end{cases}$$

$$\underline{\omega}_K^\delta = \begin{cases} -\underline{\phi}_i^{\prime q}(\underline{r}_{hq}) + \underline{\phi}_i^{\prime q}(\underline{r}_{hq}\,)(1-\delta_{iq}) & ; \delta \in E(q) \\ 0 & ; OTHERWISE \end{cases}$$

$$\underline{V}_K^{\ddot{q}\delta} = \begin{cases} \underline{\phi}_i^q(\underline{r}^\delta) + \underline{\phi}_i^{\prime q}(\underline{r}^\delta) \times \underline{\rho}_0 & ; \delta = q \\ \underline{0} & ; OTHERWISE \end{cases}$$

$\underline{g}_i^g$ = i-TH TRANSLATION AXIS OF THE g-TH HINGE, FIXED IN $L(g)$, BODY INBOARD OF THE g-TH BODY

$\underline{l}_i^g(\theta^g) \equiv$ i-TH ROTATION AXIS (EULER AXIS) OF THE g-TH HINGE

$^g\underline{\chi}^j \equiv$ VECTOR LOCATING BODY j REFERENCE WRT BODY g REFERENCE

$\underline{r}_{hg} \equiv$ VECTOR LOCATING UNDEFORMED HINGE ATTACH POINT ON BODY g WRT BODY g REFERENCE.

$\underline{U}_{hg} \equiv$ DEFORMATION AT $\underline{r}_{hg}$ ON BODY g $\left(\underline{U}_{hg} = \sum\limits_{i=1}^{NM_g} \underline{\phi}_i^g(\underline{r}_{hg}) \eta_i^g \right)$

$^{hg}\underline{z}^{bj} \equiv$ VECTOR LOCATING BODY j REFERENCE WRT BODY g HINGE ATTACH POINT ON BODY g.

$^{P(g)}\underline{S}^{bj} \equiv$ VECTOR FROM POINT $P(g)$ ON BODY g (LEADING TO BODY j) TO BODY j REFERENCE. $\left(\text{IF } j=g, {}^{P(g)}\underline{S}^{bj} = \underline{0} \right)$

$E(g) \equiv$ SET OF All BODIES OUTBOARD OF THE g-TH BODY INCLUDING BODY g

"GENERALIZED INERTIAL FORCE"

(EXPRESSION FOR $-f_K^*$)

$$-f_K^* = \sum_{\delta=1}^{NB} \left\{ m_\delta (\underline{\ddot{R}}^{b\delta} + \underline{\ddot{L}}_\delta) \cdot \underline{V}_K^{b\delta} + (\underline{\dot{H}}^\delta + m_\delta \underline{L}_\delta \times \underline{\ddot{R}}^{b\delta}) \cdot \underline{\omega}_K^\delta + \sum_{\sigma=1}^{NNB_\delta} \int_{b_{\delta\sigma}} \underline{V}_K^{q\delta} \cdot \underline{\ddot{R}}_\delta \, dm \right\}$$

MODAL TERMS

K CORRESPONDING TO THE i-TH MODAL D.O.F. OF THE δ-TH BODY

(I.) $m_\delta \underline{\alpha}_i^\delta \cdot \underline{\ddot{R}}^{b\delta}$

(II.) $\left(\underline{h}_i^\delta + \sum_{K=1}^{NM_\delta} \underline{Y}_{Ki}^\delta \eta_K^\delta + \sum_{K=1}^{NM_\delta} \sum_{m=1}^{NM_\delta} \underline{Z}_{mKi}^\delta \eta_m^\delta \eta_K^\delta \right) \cdot \underline{\dot{\omega}}^\delta$

(III.) $2\underline{\omega}^\delta \cdot \left[\sum_{K=1}^{NM_\delta} \underline{Y}_{Ki}^\delta \dot{\eta}_K^\delta + \sum_{K=1}^{NM_\delta} \sum_{m=1}^{NM_\delta} \underline{Y}_{mKi}^\delta \eta_m^\delta \dot{\eta}_K^\delta \right]$

(IV.) MODAL MASS (ASSUMED BODY BASIS)

— SCALAR REPRESENTATION FOR "LUMPED APPROACH"

$$\left\{ \sum_{o=1}^{NNB_o^{\,i}} \left(\{ \phi_{o_i}^{\,i} \}^T \{ \phi_{o_i}^{\prime\,i} \}^T \right) \left[\begin{array}{c|c} \left[m_o^{\,i}{}_{DIAGONAL} \right] & \left(-m_o^{\,i}\,\tilde{\ell}_o^{\,i} \right) \\ \hline \left[m_o^{\,i}\,\tilde{\ell}_o^{\,i} \right] & \left(J^{b_{io}} + \tilde{u}_o^{\prime\,i} J^{b_{io}} - J^{b_{io}} \tilde{u}_o^{\prime\,i} - \tilde{u}_o^{\prime\,i} J^{b_{io}} \tilde{u}_o^{\prime\,i} \right) \end{array} \right] \left\{ \begin{array}{c} \{ \phi_{o_k}^{\,i} \} \\ \hline \{ \phi_{o_k}^{\prime\,i} \} \end{array} \right\} \right\} \ddot{\eta}_k^{\,i}$$

(V.)

$$\underline{\omega}^{\,i} \cdot \left[\underline{\underline{W}}_i^{\,i} + \sum_{k=1}^{NM^{\,i}} \underline{\underline{W}}_{ki}^{\,i} \eta_k^{\,i} + \sum_{k=1}^{NM^{\,i}} \sum_{m=1}^{NM^{\,i}} \underline{\underline{W}}_{mki}^{\,i} \eta_m^{\,i} \eta_k^{\,i} \right] \cdot \underline{\omega}^{\,i}$$

$$\sum_{o=1}^{NNB_o^{\,i}} \left(\underline{\omega}^{\,i} + \underline{\dot{u}}_o^{\prime\,i} \right) \cdot \left[\underline{\underline{T}}_i^{b_{io}} + \sum_{k=1}^{NM^{\,i}} \underline{\underline{T}}_{ki}^{b_{io}} \eta_k^{\,i} + \sum_{k=1}^{NM^{\,i}} \sum_{m=1}^{NM^{\,i}} \underline{\underline{T}}_{mki}^{b_{io}} \eta_m^{\,i} \eta_k^{\,i} \right] \cdot \left(\underline{\omega}^{\,i} + \underline{\dot{u}}_o^{\prime\,i} \right)$$

$$(\text{VI.}) \quad \underline{\omega}^{\dot{a}} \cdot \left[\sum_{K=1}^{NH_{\dot{a}}} \underline{W}_{Ki}^{\dot{a}} \, \dot{\eta}_K^{\dot{a}} + \sum_{K=1}^{NH_{\dot{a}}} \sum_{M=1}^{NH_{\dot{a}}} \underline{W}_{MKi}^{\dot{a}} \, \eta_M^{\dot{a}} \, \dot{\eta}_K^{\dot{a}} \right] ;$$

$$(\text{VII.}) \quad \sum_{0=1}^{NB_{\dot{a}}} \left(\underline{\omega}^{\dot{a}} \times \underline{u}_0^{\prime \dot{a}} \right) \cdot \left[\underline{D}_i^{b_{\dot{a}o}} + \sum_{K=1}^{NH_{\dot{a}}} \underline{D}_{Ki}^{b_{\dot{a}o}} \, \eta_K^{\dot{a}} + \sum_{K=1}^{NM_{\dot{a}}} \sum_{M=1}^{NM_{\dot{a}}} \underline{D}_{MKi}^{b_{\dot{a}o}} \, \eta_M^{\dot{a}} \, \eta_K^{\dot{a}} \right]$$

RATE OF CHANGE OF BODY δ ANGULAR MOMENTUM

(CONSOLIDATED EXPRESSION FOR $\underline{\dot{H}}^{\delta}$)

$$\underline{\dot{H}}^{\delta} = \underline{\underline{\dot{I}}}^{\delta'} \cdot \underline{\dot{\omega}}^{\delta} + \sum_{i=1}^{NM^{\delta}} \left(\underline{\dot{h}}_i^{\delta} + \sum_{k=1}^{NM^{\delta}} \underline{Y}_{ki}^{\delta} \eta_k^{\delta} + \sum_{k=1}^{NM^{\delta}} \sum_{m=1}^{NM^{\delta}} \underline{Z}_{mki}^{\delta} \eta_m^{\delta} \eta_k^{\delta} \right) \ddot{\eta}_i^{\delta} + \underline{\omega}^{\delta} \times \underline{\underline{I}}^{\delta'} \cdot \underline{\omega}^{\delta}$$

$$+ \left[2 \sum_{i=1}^{NM^{\delta}} \left\{ \underline{\underline{M}}_i^{\delta} + \sum_{k=1}^{NM^{\delta}} \underline{\underline{P}}_{ki}^{\delta} \eta_k^{\delta} \right\} \dot{\eta}_i^{\delta} + \underline{\underline{K}}^{\delta} \right] \cdot \underline{\omega}^{\delta} + \underline{R}^{\delta}$$

WHERE :

$$\underline{\underline{I}}^{\delta'} = \underline{\underline{I}}_R^{\delta} + \sum_{i=1}^{NM^{\delta}} \left\{ \underline{\underline{M}}_i^{\delta} + \underline{\underline{N}}_i^{\delta} \right\} \eta_i^{\delta} + \sum_{i=1}^{NM^{\delta}} \sum_{k=1}^{NM^{\delta}} \underline{\underline{P}}_{ki}^{\delta} \eta_k^{\delta} \eta_i^{\delta}$$

Dynacs

DEFINE: $\quad \vec{b}^i = \begin{bmatrix} b^i_{\underline{o}1} \\ b^i_{\underline{o}2} \\ b^i_{\underline{o}3} \end{bmatrix} \quad ; \quad b^{iT} = \left(b^i_{\underline{o}1} \ \ b^i_{\underline{o}2} \ \ b^i_{\underline{o}3} \right) \quad ,$ BODY o REFERENCE BASIS

VARIABLE	ORDER	D.O.F. ASSOCIATION	DEFINITION
$\underline{\alpha}^i_i$	0	MODAL	$\dfrac{1}{m^i_o}\left\{ \displaystyle\sum_{o=1}^{NMB^i_o} m^i_o \phi'^i_{o\,i} - m^i_o \underline{\ell}^i_o \times \phi'^i_{o\,i} - \displaystyle\sum_{k=1}^{NM^i_o} m^i_o (\phi'^i_{o\,k} \times \underline{\ell}^i_o) \times \phi'^i_{o\,i} \right\} \eta^i_k$
$\underline{\underline{I}}^{b_{oo}}$	0,1,&,2	ROTATIONAL / MODAL	$b^T \left\{ J^{b_{oo}} + \tilde{u}'^i_o J^{b_{oo}} - J^{b_{oo}} \tilde{u}'^i_o - \tilde{u}'^i_o J^{b_{oo}} \cdot \tilde{u}'^i_o \right\} b$
			(box) $J^{b_{oo}}$ IS INERTIA MATRIX (3×3) OF NODAL BODY oo WRT NODAL BODY i. REFERENCE FRAME, b_{io}
$\underline{h}^i_i$	0	ROTATIONAL / MODAL	$b^T \left\{ \displaystyle\sum_{o=1}^{NMB^i_o} \left[m^i_o \tilde{r}^i_o \{ \phi'^i_{o\,i} \} - \tilde{r}^i_o m^i_o \tilde{\ell}^i_o \{ \phi'^i_{o\,i} \} \right.\right.$ $\left.\left. + m^i_o \tilde{\ell}^i_o \{ \phi'^i_{o\,i} \} + J^{b_{oo}} \{ \phi'^i_{o\,i} \} \right] \right\}$
			(box) $\{x\}$ DENOTES COLUMN MATRIX

VARIABLE	ORDER	D.O.F. ASSOCIATION	DEFINITION
$\underline{Y}^{i}_{ki}$		ROTATIONAL/ MODAL	$b^{T}\left\{\sum_{o=1}^{NNB^{i}}\left[m^{i}_{o}\tilde{\phi}^{i}_{o_{k}}\{\phi^{i}_{o_{i}}\}+\tilde{r}^{i}_{o}\left(m^{i}_{o}\tilde{l}^{i}_{o}\{\phi^{i}_{o_{k}}\}\right)\{\phi^{i}_{o_{i}}\}\right.\right.$ $-\tilde{\phi}^{i}_{o_{k}}m^{i}_{o}\tilde{l}^{i}_{o}\{\phi^{i}_{o_{i}}\}-m^{i}_{o}\left(\tilde{l}^{i}_{o}\{\phi^{i}_{o_{k}}\}\right)\{\phi^{i}_{o_{i}}\}$ $\left.\left.+\left(\tilde{\phi}^{i}_{o_{k}}J^{bi}_{o}-J^{bi}_{o}\tilde{\phi}^{i}_{o_{k}}\right)\{\phi^{i}_{o_{i}}\}\right]\right\}$
$\underline{Z}^{i}_{mki}$		ROTATIONAL/ MODAL	$b^{T}\left\{\sum_{o=1}^{NNB^{i}}\left(\tilde{\phi}^{i}_{o_{m}}\left(m^{i}_{o}\tilde{l}^{i}_{o}\{\phi^{i}_{o_{k}}\}\right)\{\phi^{i}_{o_{i}}\}-\tilde{\phi}^{i}_{o_{m}}J^{bi}_{o}\tilde{\phi}^{i}_{o_{k}}\{\phi^{i}_{o_{i}}\}\right)\right\}$
$\underline{Y}^{i}_{ki}$		MODAL	$b^{T}\left\{\sum_{o=1}^{NNB^{i}}\left[m^{i}_{o}\tilde{\phi}^{i}_{o_{k}}\{\phi^{i}_{o_{i}}\}-\tilde{\phi}^{i}_{o_{k}}m^{i}_{o}\tilde{l}^{i}_{o}\{\phi^{i}_{o_{i}}\}\right]\right\}$
$\underline{Y}^{i}_{mki}$		MODAL	$b^{T}\left\{\sum_{o=1}^{NNB^{i}}\left[\tilde{\phi}^{i}_{o_{k}}\left(m^{i}_{o}\tilde{l}^{i}_{o}\{\phi^{i}_{o_{m}}\}\right)\{\phi^{i}_{o_{i}}\}\right]\right\}$
$\underline{\underline{W}}^{i}_{i}$		MODAL	$b^{T}\left\{\sum_{o=1}^{NNB^{i}}\left(m^{i}_{o}\tilde{\phi}^{i}_{o_{i}}\tilde{r}^{i}_{o}-\left(m^{i}_{o}\tilde{l}^{i}_{o}\{\phi^{i}_{o_{i}}\}\right)\tilde{r}^{i}_{o}\right)\right\}b$
$\underline{\underline{W}}^{i}_{ki}$		MODAL	$b^{T}\left\{\sum_{o=1}^{NNB^{i}}\left[\left[\left(m^{i}_{o}\tilde{l}^{i}_{o}\{\phi^{i}_{o_{k}}\}\right)\{\phi^{i}_{o_{i}}\}\right]\tilde{r}^{i}_{o}+m^{i}_{o}\tilde{\phi}^{i}_{o_{i}}\tilde{\phi}^{i}_{o_{k}}\right.\right.$ $\left.\left.-\left(m^{i}_{o}\tilde{l}^{i}_{o}\{\phi^{i}_{o_{i}}\}\right)\tilde{\phi}^{i}_{o_{k}}\right]\right\}b$

367

VARIABLE	ORDER	D.O.F. ASSOCIATION	DEFINITION
$\underline{\underline{W}}{}^{i}_{mki}$		MODAL	$b^T \left\{ \sum_{o=1}^{NNB^i_o} \left[\left(\left(m^i_o \tilde{l}^i_o \{\phi^{ii}_{ok}\} \right) \{\phi^{ii}_{oi}\} \right)^{\sim} \tilde{\phi}^i_{om} \right] \right\} b$
$\underline{\underline{T}}{}^{bio}_{i}$		MODAL	$b^T \left\{ \tilde{\phi}^i_{oi} m^i_o \tilde{l}^i_o + J^{bio} \tilde{\phi}^{\prime i}_{oi} \right\} b$
$\underline{\underline{T}}{}^{bio}_{ki}$		MODAL	$b^T \left\{ -\tilde{\phi}^i_{oi} \left(m^i_o \tilde{l}^i_o \{\phi^{ii}_{ok}\} \right)^{\sim} + \tilde{\phi}^{\prime i}_{ok} J^{bio} \tilde{\phi}^{\prime i}_{oi} - J^{bio} \tilde{\phi}^{\prime i}_{ok} \tilde{\phi}^{\prime i}_{oi} \right\} b$
$\underline{\underline{T}}{}^{bio}_{mki}$		MODAL	$b^T \left\{ -\tilde{\phi}^{\prime i}_{om} J^{bio} \tilde{\phi}^{\prime i}_{ok} \tilde{\phi}^{\prime i}_{oi} \right\} b$
$\underline{W}{}^{i}_{ki}$		MODAL	$b^T \left\{ \sum_{o=1}^{NNB^i_o} \left[\tilde{\phi}^{\prime i}_{ok} m^i_o \tilde{l}^i_o \{\phi^i_{oi}\} \right] \right\}$
$\underline{W}{}^{i}_{mki}$		MODAL	$b^T \left\{ \sum_{o=1}^{NNB^i_o} \left[-\tilde{\phi}^{\prime i}_{ok} \left(m^i_o \tilde{l}^i_o \{\phi^{\prime i}_{om}\} \right)^{\sim} \{\phi^{\prime i}_{oi}\} \right] \right\}$
$\underline{D}{}^{bio}_{i}$		MODAL	$b^T \left\{ J^{bio} \{\phi^{\prime i}_{oi}\} \right\}$
$\underline{D}{}^{bio}_{ki}$		MODAL	$b^T \left\{ \tilde{\phi}^{\prime i}_{ok} J^{bio} \{\phi^{\prime i}_{oi}\} - J^{bio} \tilde{\phi}^{\prime i}_{ok} \{\phi^{\prime i}_{oi}\} \right\}$
$\underline{D}{}^{bio}_{mki}$		MODAL	$b^T \left\{ -\tilde{\phi}^{\prime i}_{om} J^{bio} \tilde{\phi}^{\prime i}_{ok} \{\phi^{\prime i}_{oi}\} \right\}$

368

VARIABLE	ORDER	D.O.F. ASSOCIATION	DEFINITION
$\underline{\underline{I}}{}^{a}_{R}$	0	ROTATIONAL	$b^T \left\{ \sum\limits_{o=1}^{NNB^a} \left[J^{b_io} + m^i_o \tilde{r}^i_o \tilde{r}^i_o - \tilde{r}^i_o m^i_o \tilde{l}^i_o - m^i_o \tilde{l}^i_o \tilde{r}^i_o \right] \right\} b$
$\underline{\underline{M}}{}^{a}_{i}$	1	ROTATIONAL	$b^T \left\{ \sum\limits_{o=1}^{NNB^a} \left[-m^i_o \tilde{r}^i_o \tilde{\phi}^{o_i}_{o_i} + \tilde{r}^i_o (m^i_o \tilde{l}^i_o \{\phi^{o_i}_{o_i}\}\tilde{\;}) - m^i_o \tilde{l}^i_o \tilde{\phi}^{o_i}_{o_i} \right. \right.$ $\left. \left. - J^{b_io} \tilde{\phi}^{o_i}_{o_i} \right] \right\} b$
$\underline{\underline{N}}{}^{a}_{i}$	1	ROTATIONAL	$b^T \left\{ \sum\limits_{o=1}^{NNB^a} \left[-m^i_o \tilde{\phi}^{o_i}_{o_i} \tilde{r}^i_o - \tilde{\phi}^{o_i}_{o_i} m^i_o \tilde{l}^i_o + (m^i_o \tilde{l}^i_o \{\phi^{o_i}_{o_i}\}\tilde{\;}) \tilde{r}^i_o \right. \right.$ $\left. \left. + \tilde{\phi}^{o_i}_{o_i} J^{b_io} \right] \right\} b$
$\underline{\underline{P}}{}^{a}_{ki}$	2	ROTATIONAL	$b^T \left\{ \sum\limits_{o=1}^{NNB^a} \left[-m^i_o \tilde{\phi}^{o_i}_{o_k} \tilde{\phi}^{o_i}_{o_i} + \tilde{\phi}^{o_i}_{o_k} (m^i_o \tilde{l}^i_o \{\phi^{o_i}_{o_i}\}\tilde{\;}) \right. \right.$ $\left. + (m^i_o \tilde{l}^i_o \{\phi^{o_i}_{o_k}\}\tilde{\;}) \tilde{\phi}^{o_i}_{o_i} - \tilde{\phi}^{o_i}_{o_k} J^{b_io} \tilde{\phi}^{o_i}_{o_i} \right] \bigg\} b$
$\underline{\underline{K}}{}^{a}$	MULTIPLE	ROTATIONAL	$\underline{\underline{K}}{}^{a} = \sum\limits_{i=1}^{NM^a} \underline{\underline{K}}{}^{a}_{i} \ddot{\eta}^a_i + \sum\limits_{i=1}^{NM^a} \sum\limits_{k=1}^{NM^a} \underline{\underline{K}}{}^{a}_{ki} \dot{\eta}^a_k \dot{\eta}^a_i$ $+ \sum\limits_{i=1}^{NM^a} \sum\limits_{k=1}^{NM^a} \sum\limits_{m=1}^{NM^a} \underline{\underline{K}}{}^{a}_{mki} \dot{\eta}^a_m \dot{\eta}^a_k \dot{\eta}^a_i$

369

VARIABLE	ORDER	D.O.F. ASSOCIATION	DEFINITION
$\underline{\underline{K}}_i^{\delta}$		ROTATIONAL	$b^T\left\{\sum\limits_{o=1}^{NNB^{\delta}}\left[J^{bio}\,\tilde{\phi}_{oi}^{\prime\delta} + \tilde{\phi}_{oi}^{\prime\delta}\,J^{bio}\right]\right\}b$
$\underline{\underline{K}}_{ki}^{\delta}$		ROTATIONAL	$b^T\left\{\sum\limits_{o=1}^{NNB^{\delta}}\left[\tilde{\phi}_{ok}^{\prime\delta}\,J^{bio}\,\tilde{\phi}_{oi}^{\prime\delta} + J^{bio}\,\tilde{\phi}_{ok}^{\prime\delta}\,\tilde{\phi}_{oi}^{\prime\delta} + \tilde{\phi}_{oi}^{\prime\delta}\,\tilde{\phi}_{ok}^{\prime\delta}\,J^{bio}\right.\right.$ $\left.\left. - \tilde{\phi}_{oi}^{\prime\delta}\,J^{bio}\,\tilde{\phi}_{ok}^{\prime\delta} + 2\,\tilde{r}_o^{\delta}\left[\tilde{\phi}_{oi}^{\prime\delta}\left(m_o^{\delta}\,\tilde{l}_o^{\delta}\{\phi_{ok}^{\prime\delta}\}\right)\right]^{\sim}\right]\right\}b$
$\underline{\underline{K}}_{mki}^{\delta}$		ROTATIONAL	$b^T\left\{\sum\limits_{o=1}^{NNB^{\delta}}\left[2\,\tilde{\phi}_{ok}^{\prime\delta}\left[\tilde{\phi}_{oi}^{\prime\delta}\left(m_o^{\delta}\,\tilde{l}_o^{\delta}\{\phi_{om}^{\prime\delta}\}\right)\right]^{\sim} + \tilde{\phi}_{om}^{\prime\delta}\,J^{bio}\,\tilde{\phi}_{ok}^{\prime\delta}\,\tilde{\phi}_{oi}^{\prime\delta}\right.\right.$ $\left.\left. - \tilde{\phi}_{oi}^{\prime\delta}\,\tilde{\phi}_{ok}^{\prime\delta}\,J^{bio}\,\tilde{\phi}_{om}^{\prime\delta}\right]\right\}b$
$\underline{R}^{\delta}$	MULTIPLE	ROTATIONAL	$\sum\limits_{o=1}^{NNB^{\delta}}\left\{\underline{\omega}^{\delta}\times\underline{\underline{I}}^{bio}\cdot\underline{\dot{u}}_o^{\prime\delta} + \underline{\dot{u}}_o^{\prime\delta}\times\underline{\underline{I}}^{bio}\cdot\underline{\dot{u}}_o^{\prime\delta}\right.$ $\left. + (\underline{\dot{r}}_o + \underline{\dot{u}}_o^{\delta})\times\left(\underline{\dot{u}}_o^{\prime\delta}\times\left(\underline{\dot{u}}_o^{\prime\delta}\times(m_o^{\delta}\,\underline{l}_o^{\delta} + \underline{u}_o^{\prime\delta}\times m_o^{\delta}\,\underline{l}_o^{\delta})\right)\right)\right\}$

EXPLICIT MODELING AND COMPUTATIONAL LOAD DISTRIBUTION
FOR CONCURRENT PROCESSING SIMULATION OF THE SPACE STATION

By

R. Gluck
TRW Space and Technology Group
Redondo Beach, California

ABSTRACT

Analytical simulation of the dynamics/control interaction of large space structures is essential during the design process as full-scale tests of these vehicles in the laboratory are impractical. Furthermore, the operational manifests of large space systems on-orbit may call for significant changes in their mass and stiffness distributions as well as for substantial growth during the vehicles' lifetimes, and these can be studied only by analytical simulation.

Current methodologies for simulating large space structures involve implicit mathematical models and solutions on serial digital computers. These methodologies require unacceptably long computer processing time and exorbitant costs as the models become larger and more complex. Potential orders-of-magnitude reductions in simulation time and cost of multibody dynamic systems can be attained using: (1) enhanced analytical models for simulation, and (2) special-purpose, concurrent computational hardware and system software.

This paper deals with two important aspects of concurrent processing under development at TRW. These are: (1) the derivation of explicit mathematical models of multibody dynamic systems, and (2) a balanced computational load distribution (BCLD) among loosely coupled computational units (processors) of a concurrent processing system. The developed methodologies will be demonstrated in the paper by way of an application to the Phase 1 of the Space Station - a task being performed by TRW under NASA/JSC contract NAS9-17778.

The mathematical model of the Space Station consists of three interconnected flexible bodies capable of undergoing large, rigid-body motion with respect to each other. Body 1 is the main central body and contains the pressurized modules inboard of the two Alpha gimbals. Bodies 2 and 3 are the starboard and port bodies connected to Body 1 at the Alpha gimbals and include all components on the transverse booms outboard of the Alpha gimbals (including the solar arrays). The control systems in the model maintain Body 1 in a prescribed 3-axis attitude control mode, while producing large-angle rotations of the flexible solar arrays to position them normal to the sun-line.

PRECEDING PAGE BLANK NOT FILMED

The BCLD methodology for concurrent processing developed by TRW enploys a static allocation strategy in which a separate software package is used off-line and at leisure prior to the execution of the simulation program. The load distribution, in this methodology, is carried out in a manner transparent to the user who, nevertheless, exercises control over the procedure with pre-selected constraint conditions.

The distributed model of the Space Station is now complete and ready to undergo benchmark tests on TRW's Custom Architectured Parallel System during the June-July 1988 timeframe.

TRW

NASA/OAST Workshop on Computational Aspects
in the Control of Flexible Systems

Explicit Modeling and Computational Load Distribution for Concurrent Processing Simulation of the Space Station

373

Dr. R. Gluck

July 12-14, 1988
Williamsburg, Virginia

Introduction

This paper presents the application of concurrent processing technology
developed at TRW Space & Technology Group over the past several
years to the simulation of the Space Station. The effort is funded by
NASA Johnson Space Center under Contract NAS9-17778 and monitored
by Mr. John W. Sunkel. The period of performance extends from
April 1987 to November 1988

374

OBJECTIVE

This project was established to provide NASA with quantitative data to determine the cost effectiveness of utilizing a specialized processing system such as the Custom Architectured Parallel Processing System (CAPPS) for development and verification of the operational Space Station flight control system. The CAPPS is a concurrent processor consisting of loosely coupled, high speed array processors [computational units (CUs)] - each containing its own input/output capability and memory banks. The specially designed CUs are capable of concurrent computation and communication, thereby placing a very low overhead on the latter operation. Furthermore, the system's architecture provides for direct communication between each CU and any other CU, facilitating considerable flexibility in adapting the CAPPS architecture to a specific simulation problem.

375

Objective

TRW

The objective of this project is to develop, verify and
demonstrate the simulation of an explicit model of the
controls/structure interaction of the Space Station on CAPPS

376

PROJECT WORK BREAKDOWN STRUCTURE

Applying previously developed application and system software at TRW to the concurrent processing of an explicit model of the control/structure interaction of the Space Station on CAPPS, the project described herein consists of three sequential tasks as stated in the figure. This paper focuses on the completed Tasks (a) and (b). Task (c), at the time of this writing, is in progress.

377

Project Work Breakdown Structure

378

a.

Space Station Model Development	**2.** Space Station Model Report

b.

A Balanced Computational Load Distribution of the Space Station Model for the CAPPS	**3.** Computational Load Distribution Report

c.

Simulation of the Space Station Model on the CAPPS	**4.** Final Report

Project Management

3

The Space Station Model Development

TRW

Engineering & Test Division
TRW Space & Technology Group

379
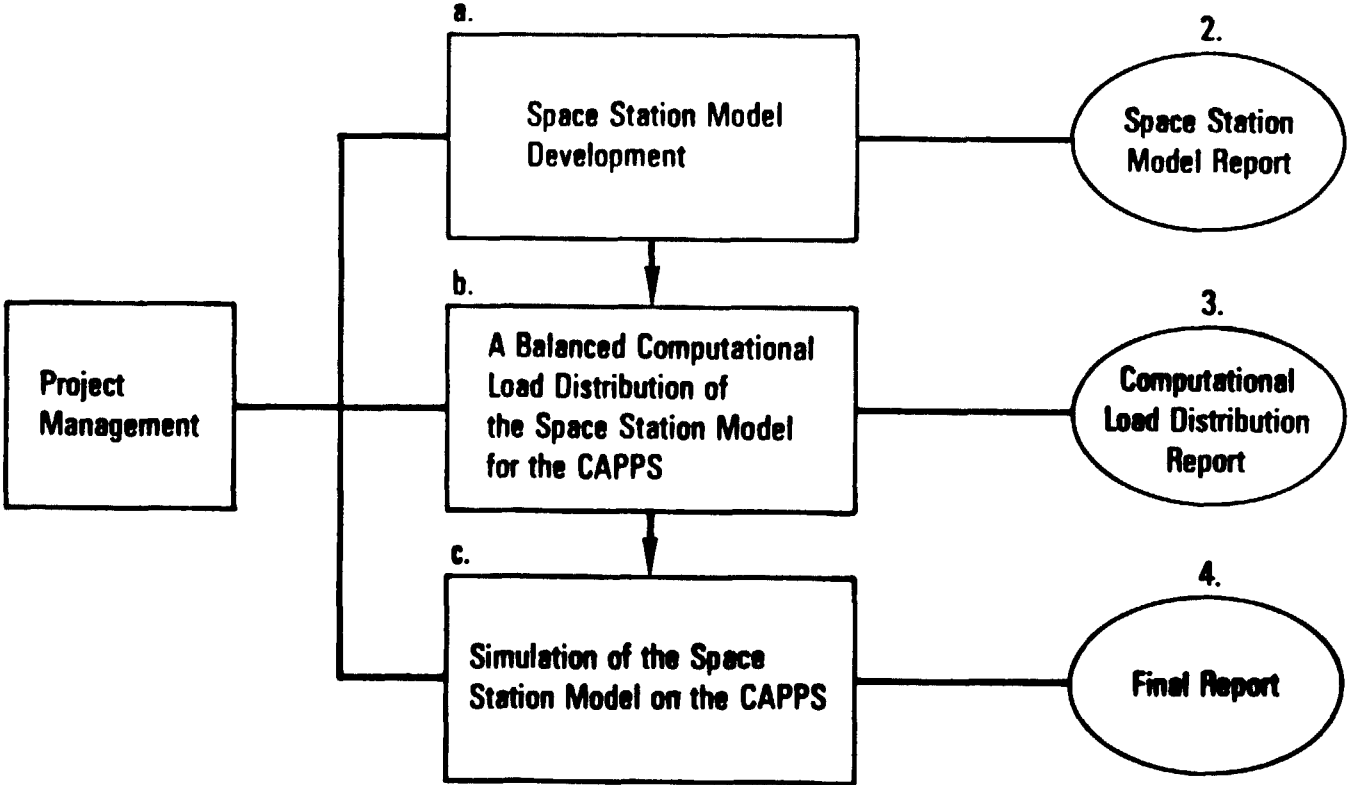
- Methodology

- Model's Contents

ADVANTAGES OF EXPLICIT EQUATIONS OF MOTION

The derivation of the equations of motion by symbol manipulation has several important attributes compared to the conventional (implicit) methodology. Symbol manipulation, i.e., the nonnumerical computation with a digital computer, yields scalar equations of motion specifically tailored to complex dynamical systems, where the analyst has the freedom and insight to incorporate any required fidelity in the model. Furthermore, the output of symbol manipulation is a completely portable FORTRAN code in the format of $A(x,t)\ddot{x} = b(x,t)$, which can be delivered via file to either serial or concurrent processors without requiring any programming. This reduces development cost by at least one order-of-magnitude or more compared to that of a special-purpose implicit code. Finally, symbolically derived scalar equations of motion require a substantially reduced simulation time compared to those of conventional codes. Benchmark tests conducted at TRW yielded improvements in run times by factors of approximately 4 and 3 for rigid-body and flexible-body systems, respectively.

Advantages of Explicit Equations of Motion

- Useful engineering insights into the dynamic characteristics of the system

- No major programming effort required to perform simulation

- Large reduction of time required for simulation as compared to that required for implicit formulation

  - Implicit formulation requires the derivation of the equations of motion to be performed numerically at each integration step
  - Explicit formulation requires the derivation to be performed only once

5

ADVANTAGES OF USING PROGRAM SMP TO DERIVE EXPLICIT EQUATIONS OF MOTION

Program SMP was selected for the TRW symbol manipulation methodology following a thorough analysis which proved it superior in both versatility and speed to other available symbol manipulation codes such as MACSYMA, Reduce and FORMAC. The SMP program is implemented in the C language and is available on a variety of mainframes and workstations. Its capability of handling very large amounts of data is ideally suited for the derivation of explicit equations of motion of multibody spacecraft. The program's other attributes are listed in the accompanying figure.

Advantages of Using SMP to Derive
Explicit Equations of Motion

- Relieves the drudgery and distasteful tasks of manual algebraic manipulation

- Reduces cost and time by orders-of-magnitude as compared
  to manual derivation

- Allows the analysts to fully participate in the derivation process to achieve
  the most efficient mathematical model

- Leads to equations with no wasteful operations, such as additions of zeros,
  multiplications by unity, and dot product of orthogonal vectors

6

PROGRAM SYMBOD

Program SYMBOD for the derivation of the explicit equations of motion of multibody flexible dynamical systems was developed within the general framework of SMP. A menu is provided to the analyst for introducing the system's topology and appropriate logic is available for the definition/computation of the essential elements of Kane's Dynamical Equations as shown in the figure; however, the analyst can, at his discretion, override the logic imbedded in each of the program's modules. The procedure is considered to be optimal in the sense that it leaves with the analyst the tasks he is best trained to perform, while transferring to the computer the manually prohibitive algebraic manipulation and long derivation operations.

Two methodologies were incorporated in Program SYMBOD for dealing with the presence of m geometrical and velocity constraints in a multibody dynamic system. In the first methodology (by Wampler et al, Reference 1), the dynamic equations governing a constrained system are generated symbolically directly from expressions comprising the equations governing the system without constraints. This methodology for constraint elimination (which requires a symbolic inversion of a matrix of order m) is applied when the number of constraint equations in the system is small (m<6) and no working constraints are involved; otherwise, the Lagrange multiplier methodology is used, where the stabilized penalty procedure of Reference 2 offers an attractive way for stabilizing the constraint equations now retained in the mathematical model.

Top Level Flow Diagram for the Explicit Formulation of Kane's Dynamical Equations with the SMP Program (Program SYMBOD)

385
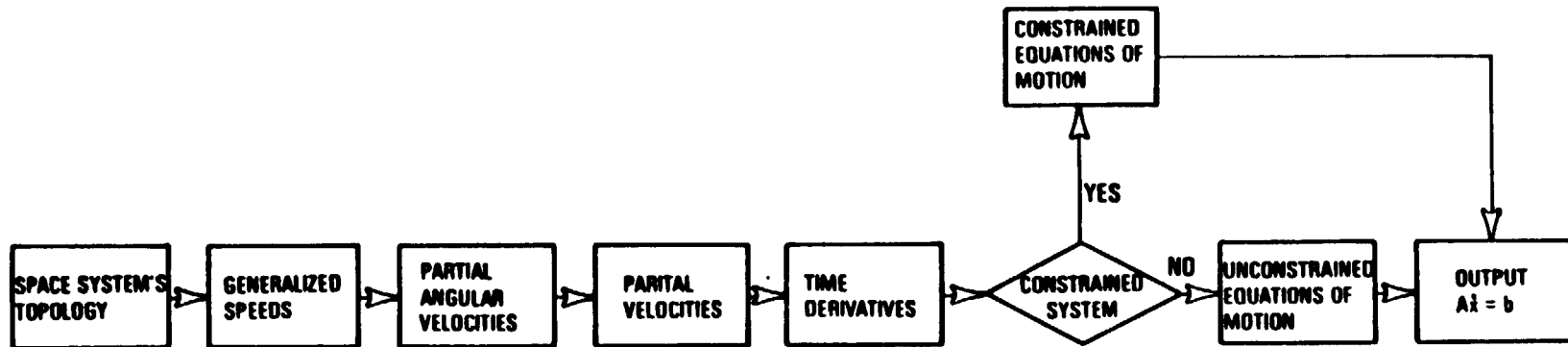
ATTRIBUTES OF PROGRAM SYMBOD

Program SYMBOD contains several innovations which combine to produce an efficient mathematical model. These are listed in the figure and explained briefly below.

The generation of equations of motion by symbol manipulation requires a systematic method of reducing the number of algebraic operations in the formation of Kane's equations. Frequently, the intermediate computations of expressions, such as the velocity terms, produce a multitude of expressions so large that their storage requirements exceed the computer's capacity. A procedure for systematically introducing new intermediate symbols to replace recurring combinations of algebraic subexpressions was developed. This procedure eliminates repetitious calculations and results in efficient computational algorithms requiring fewer arithmetic operations.

The formulation of Kane's dynamical equations associated with the flexible-body degrees-of-freedom (dof) of a body are iterative in the number of assumed admissible functions required to represent the flexibility. The totality of the flexible-body dof for each body was, therefore, represented in Program SYMBOD by a single dof of that body. This allows postponement of the final selection of the required number of assumed admissible functions until after the development of the explicit mathematical model (including the control system) is completed, i.e., the assumed admissible functions in this formulation need not be selected prematurely.

Program SYMBOD provides for direct elimination (by command) of superfluous higher order terms in the explicit equations of motion when these terms are inconsistent with basic assumptions made in the formulation or with the variance of input parameters.

Attributes of Program SYMBOD

- Systematic introduction of intermediate variables for algebraic subexpressions
    - Eliminates repetitious calculations
    - Reduces computer memory requirements
    - Produces efficient computational equations involving fewer arithmetic operations

- Totality of flexible-body degrees-of-freedom (dof) of each flexible body is represented by a single dof of that body. Selection of desired flexible functions in simulation is performed **after** control system is specified

- Elimination of superfluous higher-order terms in equations

- Error-free translation of explicit equations of motion into FORTRAN
    - Eliminates manual mistyping of equations
    - Eliminates debugging of code

TRW

Engineering & Test Division
TRW Space & Technology Group

The Space Station Model Contents
(A joint effort of NASA & TRW)

388

Space Station Model Simulation Objective

389

Simulate a transient maneuver involving a large-angle rigid-body motion of the flexible solar arrays connected to their respective transverse booms, while the central body is maintained in a three-axis attitude control mode

SPACE STATION MODEL

The mathematical model of the Space Station consists of three interconnected flexible bodies capable of undergoing large, rigid-body motion with respect to each other. Body 1 is the main central body and contains the pressurized modules inboard of the two Alpha gimbals. Bodies 2 and 3 are the starboard and port bodies connected to Body 1 at the Alpha gimbals, and include all components on the transverse booms outboard of the Alpha gimbals (including the solar arrays).

The three-body Space Station model contains eight (8) large-motion, rigid-body degrees of freedom, three translational and three rotational for the central body and one rotational for each of the extraneous bodies relative to the central body. Full coupling between the rigid- and flexible-body degrees of freedom are facilitated in the model. The flexibility of Body 1 is described by 1 to 45 "free-free" natural modes, excluding the six rigid-body modes, used here as assumed admissible functions. The flexibilities of Bodies 2 and 3 are each described by 1 to 45 "fixed-free" natural modes serving also as assumed admissible functions. The assumed spatial admissible functions are general 3-dimensional functions which satisfy the boundary conditions of the body in question but do not provide solutions to its differential equations of motion.

390

Space Station Model

Hinge 3
Alpha Gimbal

Hinge 2
Alpha Gimbal

b_3^2

b_3^1

b_3^3

b_2^2

b_1^2

b_2

Body
Reference
Frame, β_2

Body
Reference
Frame, β_1

b_1^1

Body
Reference
Frame, β_3

b_1^3

Body 1
Main Central Body

Body 2
Starboard Body

n_3

n_2

Hinge 1

n_1

Newtonian
Reference
Frame, N

Body 3
Port Body
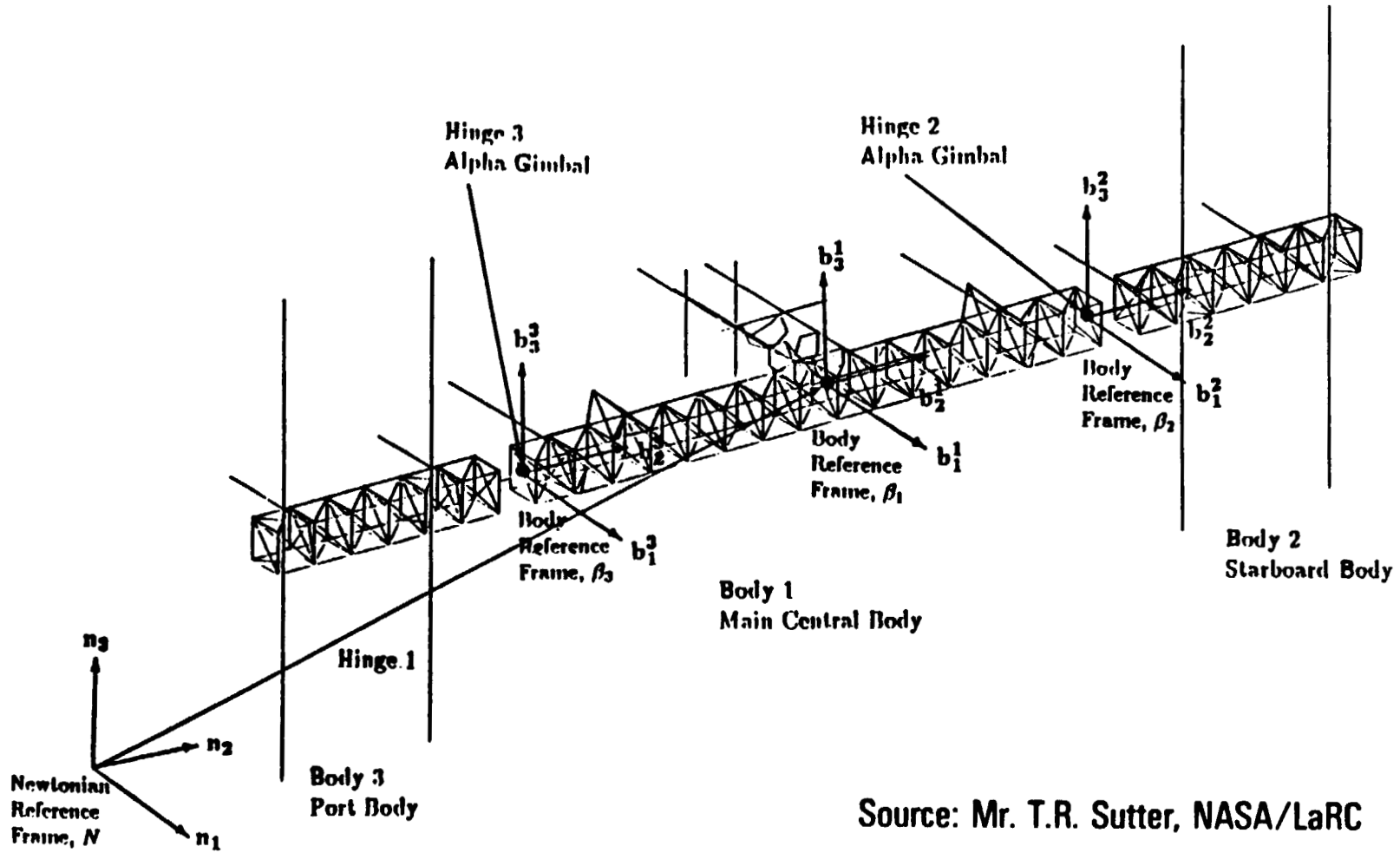
Source: Mr. T.R. Sutter, NASA/LaRC

391

11

FREQUENCY SPECTRA OF THE SPACE STATION MODEL

The assumed admissible functions in the Space Station model were obtained from finite element models developed for each of the bodies. These included an unconstrained (free-free) model for the central body and two constrained (fixed-free) models for the starboard and port bodies, cantilevered at the Alpha gimbals. A MSC/NASTRAN code was used to obtain the natural modes of vibration within a 10 Hz frequency bandwidth. The spectrum of natural frequencies for each of the three finite element models is shown in the figure. Note that these are characterized by a number of low frequency modes (below 1 Hz) spaced closely together.

Frequency Spectra of the Space Station Model

393

Central Body Frequencies, Hertz

Starboard Body Frequencies, Hertz

Port Body Frequencies, Hertz

12
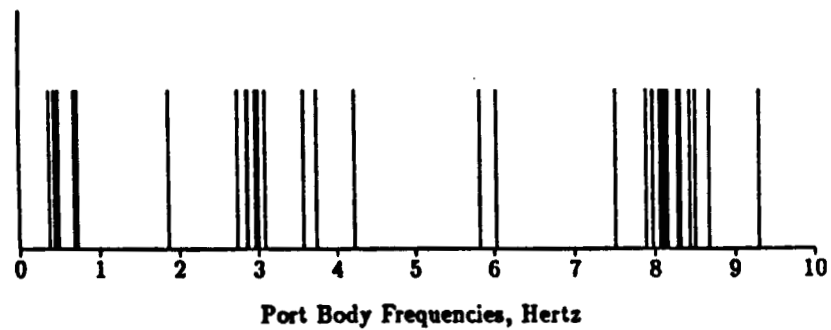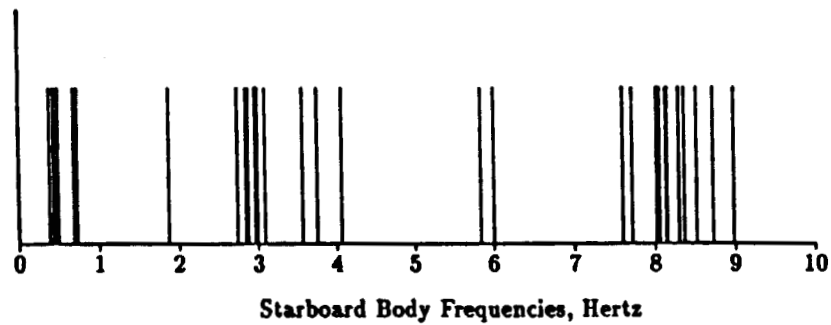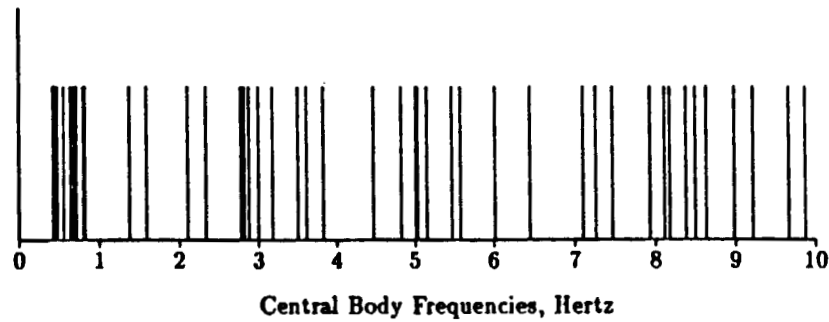
ATTITUDE CONTROL SYSTEM FOR SPACE STATION MODEL

The attitude control system of the Space Station was designed to regulate its orientation and keep its longitudinal axis aligned with the local vertical vector while maintaining its plane perpendicular to the velocity vector. The control system consists of attitude sensing instrumentation, control moment gyros, and electronics to cause corrective control moments to be applied to the Space Station's central body whenever it moves away from the commanded attitude. The attitude and rate sensors and the control moment gyros (CMG's) are co-located at the origin of the coordinate system of the central body placed at its undeformed center of mass.

A block diagram of the control law for the X_2 axis is shown in the figure. Similar control laws were designed for the remaining two axes (the three control laws are un-coupled from each other). Attitude sensing instrumentation provides electronic represen-tations of the attitude angle θ_2 and its time rate of change. The sensed attitude angle is subtracted from the commanded attitude angle (θ_{2c}) to form the attitude error signal (θ_{2e}). The electronic controller mechanizes a control law, specified in the form of a transfer function, to produce a commanded control moment (M_{2c}) based on the error signal. The CMG generates control moments (M_2) according to the commanded moments to drive the attitude error towards zero. External disturbances are not considered in this simulation and the commanded attitude is set nominally to zero.

Attitude Control System for Space Station Model - X_2 Axis

TRW

395
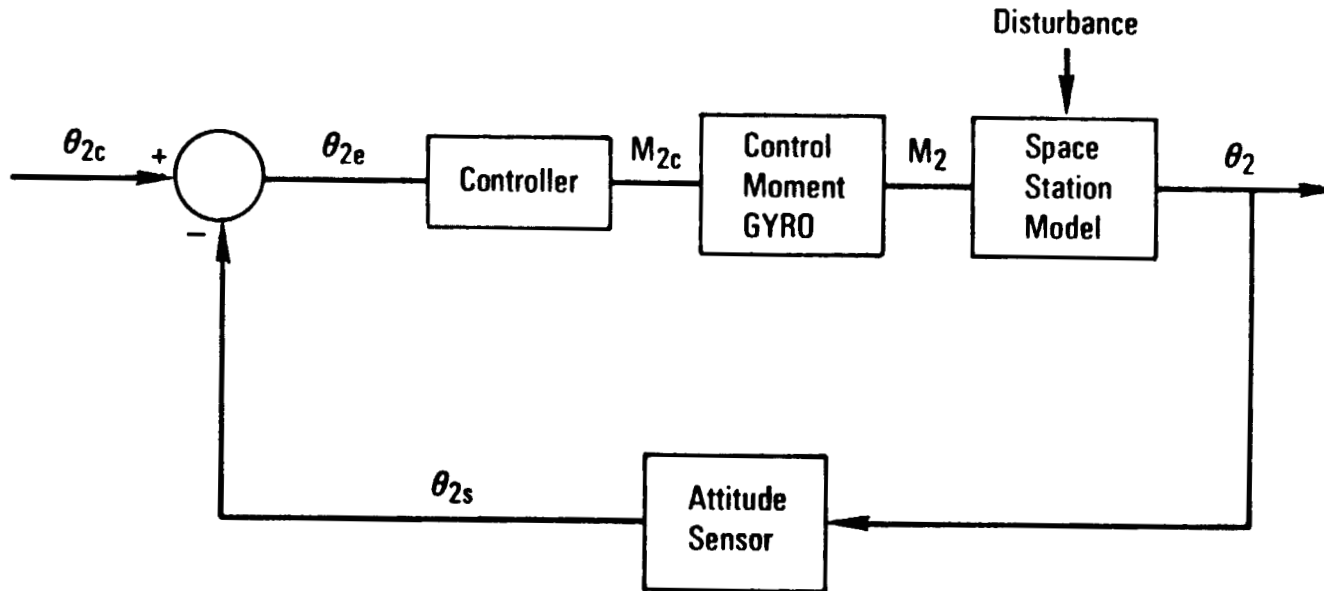
Symbol	Description
θ_2	— Central-Body Attitude
θ_{2s}	— Senced Attitude
θ_{2c}	— Commanded Attitude

Symbol	Description
θ_{2e}	— Attitude Error
M_2	— Control Moment
M_{2c}	— Commanded Contrtol Moment

Source: Mr. J.W. Young, NASA/LaRC

ARTICULATED-BODY CONTROL FOR SPACE STATION MODEL

In addition to the attitude control system, the Space Station model includes a second control system to maintain the solar arrays pointing in a direction perpendicular to the sun line. The control law is based on an angular position and rate feedback scheme with options provided to rewind the solar arrays during eclipse. The commanded angular position of the Alpha gimbal is utilized in a second order transfer function to calculate the controller's motor torque. Input and output parameters for the articulated body control system are shown in the figure.

Articulated-Body Control for Space Station Model

TRW

397

Inputs:
— Joint Angles & Rates →

— Joint Rate Commands →

— Unit Sun Vector →

Position, Velocity, Body Rates
— & Body to Inertial Frame →
Transformation

— Time →

[Articulated Body Control Subsystem]

Outputs:
— Joint Drive Torques →

Joint Angle Status Flags,
— Joint Angles, Rates & →
Accelerations

— Measured Joint Angles & Rates →
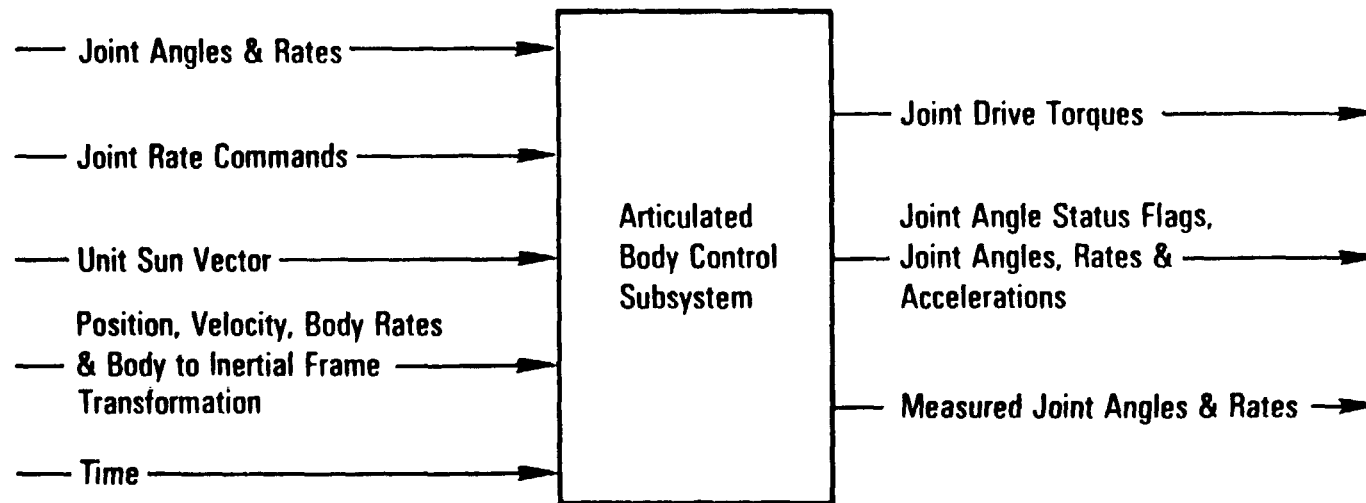
Source: Mr. J. Mapor, LEMSCO; Houston, Texas

TRW

A Balanced Computational Load
Distribution Methodology

398

c-5

MATHEMATICAL CHARACTERISTICS OF THE SPACE STATION MODEL

The balanced computational load distribution methodology described herein is aimed
at a broad class of multibody dynamic systems, which includes every variety of spacecraft,
robot, rotary aircraft and mechanism. This class is characterized by a set of first-
order, ordinary differential equations, known as Kane's Dynamical Equations, as depicted
in the figure.

The methodology for computational load allocation adopted here takes advantage of the
fact that the mathematical model involved, although generally very complex, remains
essentially unchanged for many hundreds (if not thousands) of simulation runs made in the
course of the development and verification of the dynamic system in question. For these
simulation runs, which feature different combinations of initial conditions, input
functions and parameter values, it is possible to distribute the computational load
statically, off-line, and thereby gain a significant advantage in execution speed during
simulation compared to that achievable with a dynamic load allocation methodology.

369

- KANE'S DYNAMICAL EQUATIONS CONSTITUTE A SET OF
  FIRST-ORDER ORDINARY DIFFERENTIAL EQUATIONS IN
  THE MATRIX FORM

$$A(q,t)\,\dot{u} = b(q,u,t)$$

$$\dot{q} = f(q,u,t)$$

WHERE **u** IS THE VECTOR OF GENERALIZED SPEEDS,
q IS THE VECTOR OF GENERALIZED COORDINATES,
t IS THE TIME, AND A DOT INDICATES TIME DIFFEREN-
TIATION

- THE FORM OF THE EQUATIONS IS QUITE GENERAL, AND
  IS THE FORM ADDRESSED BY THE LOAD BALANCING
  METHODOLOGY

400

LOAD BALANCING DIAGRAMS

The basic input to the load balancing software is the sequential FORTRAN code developed using Kane's Dynamical Equations and the symbol manipulation program SYMBOD/SMP.

The code is inspected by the user to determine large scale operations that may be done in parallel. This provides the software with a top level load distribution that it may then refine and balance. Mathematical models of the sort considered here willl have certain computational features that are ideally suited to parallel execution, and these may be used to provide a preliminary code division into tasks for each processor.

LOAD BALANCING
DIAGRAMS

TRW

Engineering & Test Division
TRW Space & Technology Group

402

```
                    ┌──────────────────┐
                    │  LOAD BALANCING  │
                    └──────────────────┘
┌──────────────┐             │
│ SEQUENTIAL   │             ▽
│ FORTRAN      │────────▷
│ CODE         │             │
└──────────────┘             ▽
                    ┌──────────────────┐
                    │ APPLY TOP LEVEL  │
                    │ LOAD DISTRIBUTION│
                    └──────────────────┘
                             │
                             ◁──────────────────┐
                             ▽                   │
                    ┌──────────────────┐         │
                    │  CODE DIVISION   │         │
                    ├──────────────────┤         │
                    │   DIVIDE CODE    │         │
                    │   INTO TASKS     │         │
                    └──────────────────┘         │
                             │                   │
                             ▽                   │
                    ┌──────────────────┐         │
                    │ TASK ASSIGNMENT  │         │
                    ├──────────────────┤         │
                    │  ASSIGN TASKS    │         │
                    │ TO PROCESSORS    │         │
                    └──────────────────┘         │
                             │                   │
                             ▽                   │
                        ╱─────────╲    F         │
                       ╱ ACCEPTABLE ╲────────────┘
                       ╲ LOAD BALANCE?╱
                        ╲─────────╱
                             │ T
                             ▽
                       ┌──────────┐
                       │   STOP   │
                       └──────────┘
```
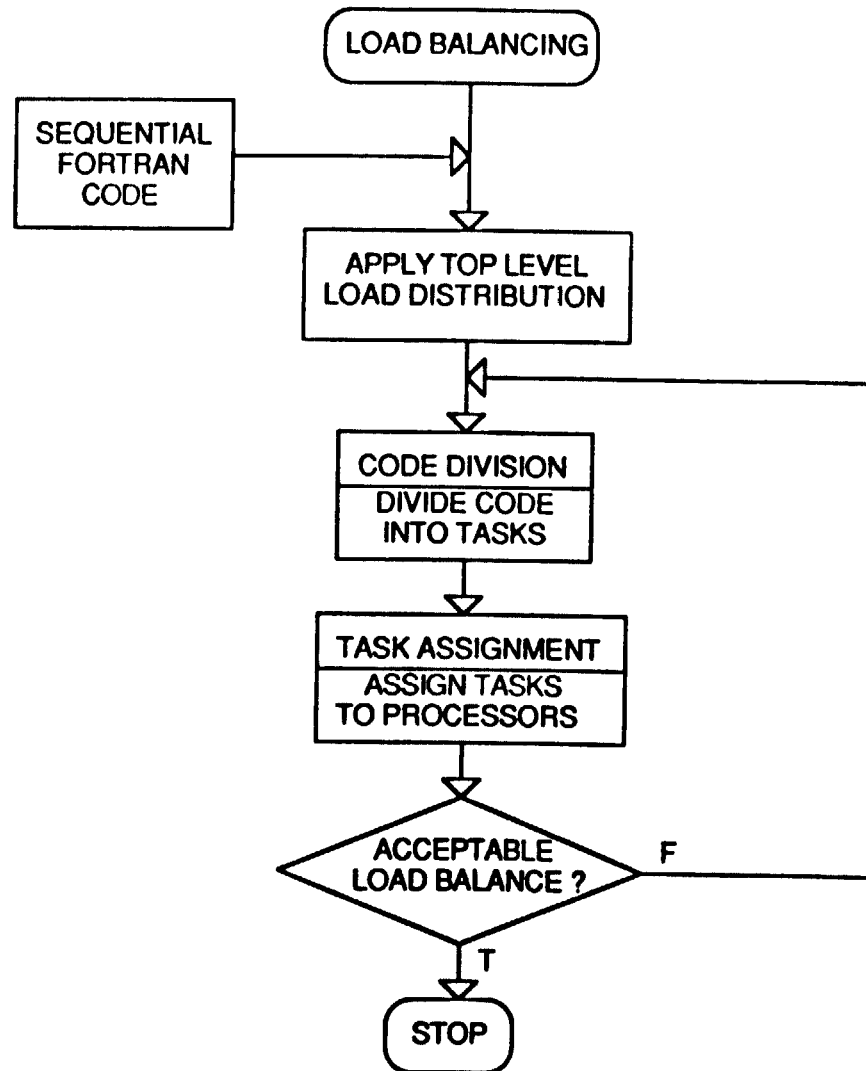
LOAD BALANCING DIAGRAMS (CONT'D)

The code is then divided into smaller tasks using the program TASK. This parti-
tioning incorporates the divisions already given in the top level distribution, and
results in tasks which are no smaller in size than some predetermined "grain" size. The
user chooses whether or not the code division is to be fine-grained or coarse-grained.
Different choices will be a result of, e.g., a different number of available processors,
the need to examine values for intermediate variables in a convenient way, etc.

Once tasks are obtained, TASK checks to see that the size and execution time for
tasks that are to be executed in parallel are approximately the same (according to
criteria determined, in part, by the user), in order to have a balanced computational
load. Those tasks found to be too large are divided while those that are too small are
merged with tasks to be executed before or afterwards. This division and merging
continues until the criteria mentioned above are satisfied.

403

TRW

404

18
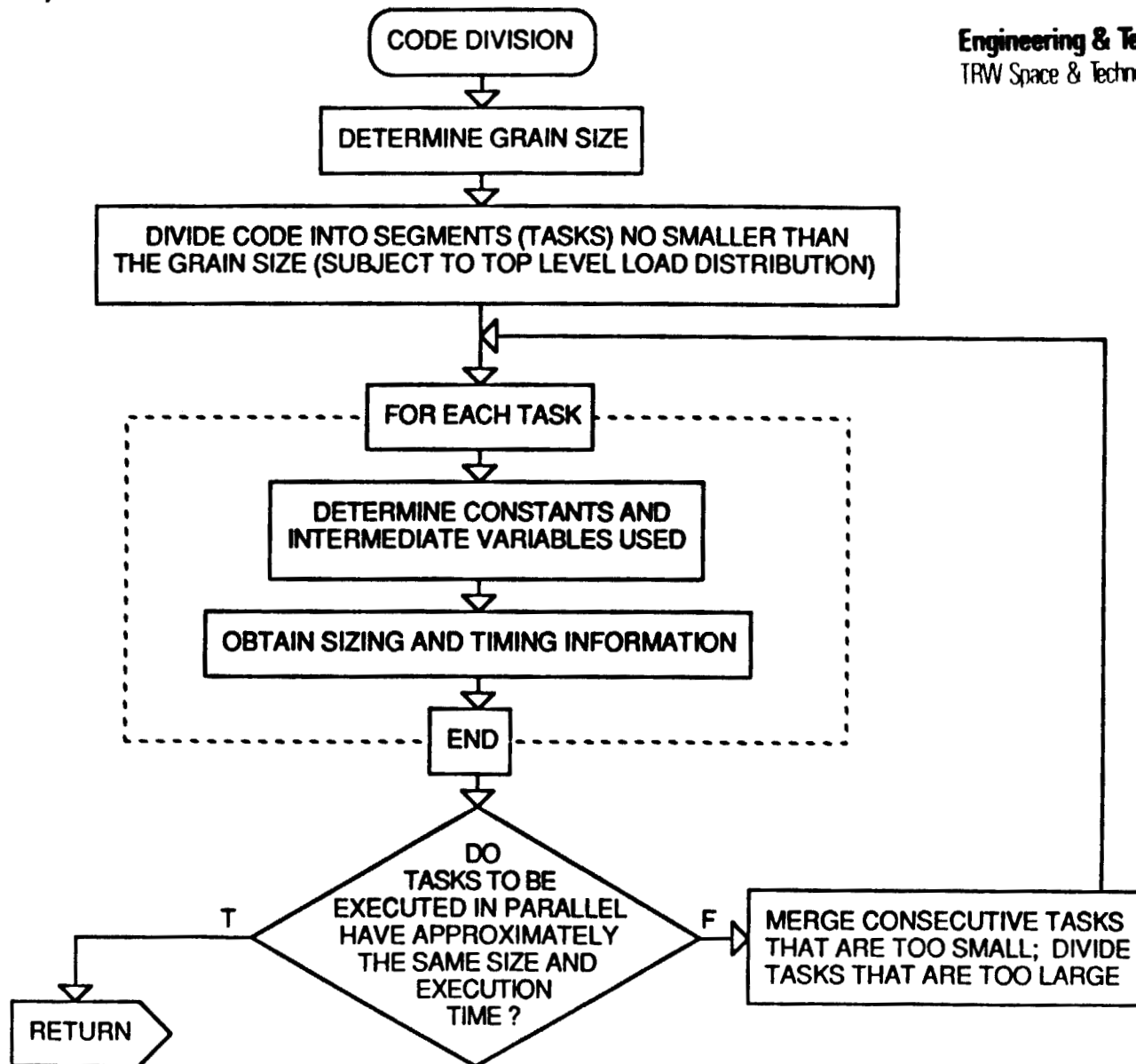
LOAD BALANCING DIAGRAMS (CONT'D)

At the conclusion of the code division, the tasks are evaluated in terms of how much communication they require with other tasks, and how many variables and parameters they share with other tasks. The program ASSIGN takes the results of the first of these evaluations and constructs what is called the connectivity matrix, with each entry indicating how much communication from task S, say, to task T is required, where S and T range over all tasks. ASSIGN uses the second evaluation to produce the parameter overlap matrix, where each entry indicates the number of parameters shared by the two tasks.

Tasks to be executed in parallel are now assigned to processors by ASSIGN. The connectivity and parameter overlap of each task with tasks already assigned to the processors is examined and a task is assigned to a particular processor according to a set of previously specified constraints.

At the end of the assignments, ASSIGN checks the computational load for balance among the processors. If the result is acceptable, then the software is finished. If not, then the constraints may be relaxed (in a way predefined by the user) and task assignment may be attempted again. If the constraints have been relaxed as far as possible, and the load is still not acceptably balanced, then TASK will attempt a different code division (and subsequent task assignment using ASSIGN) where the grain size may be different than before.

LOAD BALANCING DIAGRAMS (CONTINUED)

TRW

Engineering & Test Division
TRW Space & Technology Group

406

```
           ( TASK ASSIGNMENT )
                    │
                    ▽
   ┌─────────────────────────────────────────────┐
   │ DETERMINE CONNECTIVITY, PARAMETER OVERLAP,   │
   │ AND MEMORY SIZE CONSTRAINT INFORMATION        │
   └─────────────────────────────────────────────┘
                    │
                    ▽
   ┌─────────────────────────────────────────────┐
   │ OBTAIN CONNECTIVITY AND PARAMETER OVERLAP MATRICES │
   └─────────────────────────────────────────────┘
                    │
                    ▽
   ┌─────────────────────────────────────────────┐
   │ FOR EACH SET  S  OF TASKS TO BE EXECUTED IN PARALLEL │
   └─────────────────────────────────────────────┘
                    │
                    ▽
        ┌──────────────────────────┐
        │ FOR EACH TASK IN  S       │
        └──────────────────────────┘
                    │
                    ▽
   ┌─────────────────────────────────────────────┐
   │ DETERMINE CONNECTIVITY AND PARAMETER OVERLAP  │
   │ WITH TASKS ALREADY ASSIGNED TO EACH PROCESSOR │
   └─────────────────────────────────────────────┘
                    │
                    ▽
   ┌─────────────────────────────────────────────┐
   │ ASSIGN TASK TO A PARTICULAR PROCESSOR         │
   │ SUCH THAT CONSTRAINTS ARE SATISFIED           │
   └─────────────────────────────────────────────┘
                    │
                    ▽
              ┌─────────┐
              │   END   │
              └─────────┘
                    │
                    ▽
              ┌─────────┐
              │   END   │
              └─────────┘
                    │
                    ▽
```

```
      T        ACCEPTABLE      F        RELAX          T
   ◁────────< LOAD BALANCE ? >────────< CONSTRAINTS ? >────────
   │                                                  F │
┌─────────┐                                             │
│ RETURN  >                                             │
└─────────┘
```

19

DATA FLOW GRAPH

The data dependencies in the Space Station model are shown in the figure. The graph depicts the functional form of the model's equations.

The state vector, $\underline{x}_n$, at a given time t_n is composed of the generalized speeds ($\underline{u}_n$), the generalized coordinates ($\underline{q}_n$) and the control variables ($\underline{c}_n$).

The generalized coordinates are used in calculating time derivatives for all the state variables, as are the generalized speeds (though these are not used in the computation of the matrix $\underline{A}_n$). The control variables affect only the control torques and thus influence only the vector $\underline{b}_n$ and the derivative of $\underline{u}_n$.

Time derivatives of $\underline{q}_n$ and $\underline{c}_n$ are found directly from the generalized coordinates and speeds. Gaussian elimination is used to solve the matrix equation $\underline{A}_n \, \dot{\underline{u}}_n = \underline{b}_n$ giving the derivative of $\underline{u}_n$.

The time derivatives of each of the components of the state vector are collected to give the derivative of $\underline{x}_n$. This is then integrated using the Adams-Bashforth algorithm to give the state vector at time $t_{n+1}:\underline{x}_{n+1}$.

407

DATA FLOW GRAPH

STATE VECTOR (TIME t_n)

x_n

q_n GENERALIZED COORDINATES

u_n GENERALIZED SPEEDS

c_n CONTROL VARIABLES

A_n

b_n

LU DECOMPOSITION ALGORITHM

$\dot{q}_n$

$\dot{c}_n$

$\dot{u}_n$

$\dot{x}_n$

ADAMS-BASHFORTH ALGORITHM

x_{n+1}

TRW
Engineering & Test Division
TRW Space & Technology Group
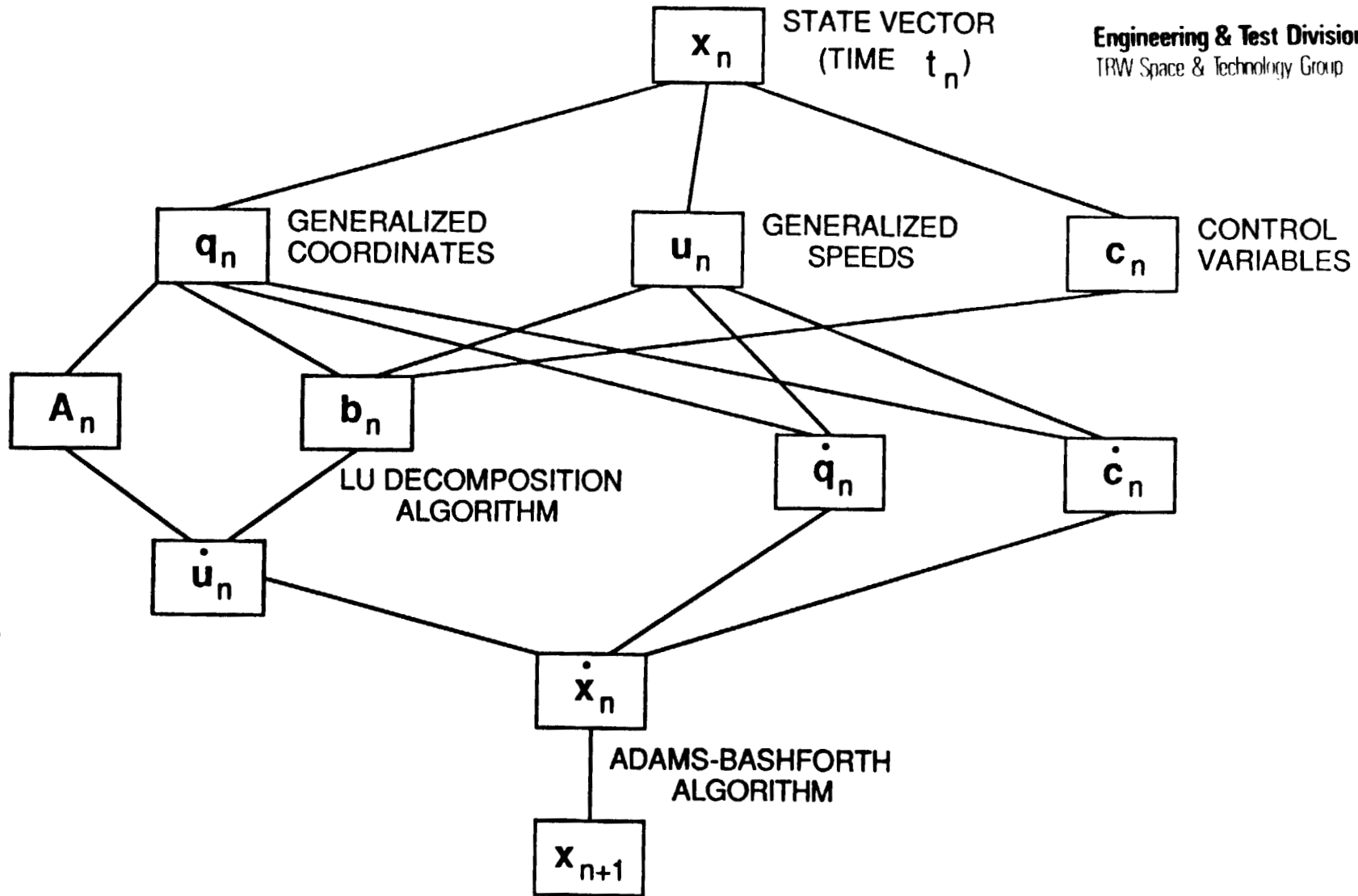
408

TOP LEVEL LOAD BALANCING (SPACE STATION MODEL)

Some opportunities for parallel execution of the Space Station model code are immediately apparent from even a casual inspection of the model, as shown in the figure.

The coordinate transformation matrices between frames in the three bodies, and between the body frames and an inertial frame are used frequently and must be calculated first. Each matrix, however, is calculated by a processor only if that processor will subsequently use it.

The outputs of the control subroutine are used only in the computation of the vector $\underline{b}$. Thus, this subroutine may be executed in parallel with sections of code computing general intermediate variables used by both $\underline{A}$ and $\underline{b}$ (such as partial angular velocities, partial velocities, etc.).

The calculation of elements of $\underline{A}$ and $\underline{b}$ may also be done in parallel, as may be the computation of the time derivative of $\underline{q}$.

A general division of the code may also be made according to whether computations involving Body 2 or Body 3 are needed. When possible, therefore, a given processor will compute quantities related only to Body 2 or only to Body 3, thus reducing interprocessor communication.

TOP LEVEL LOAD BALANCING
(SPACE STATION MODEL)

TRW

Engineering & Test Division
TRW Space & Technology Group

410
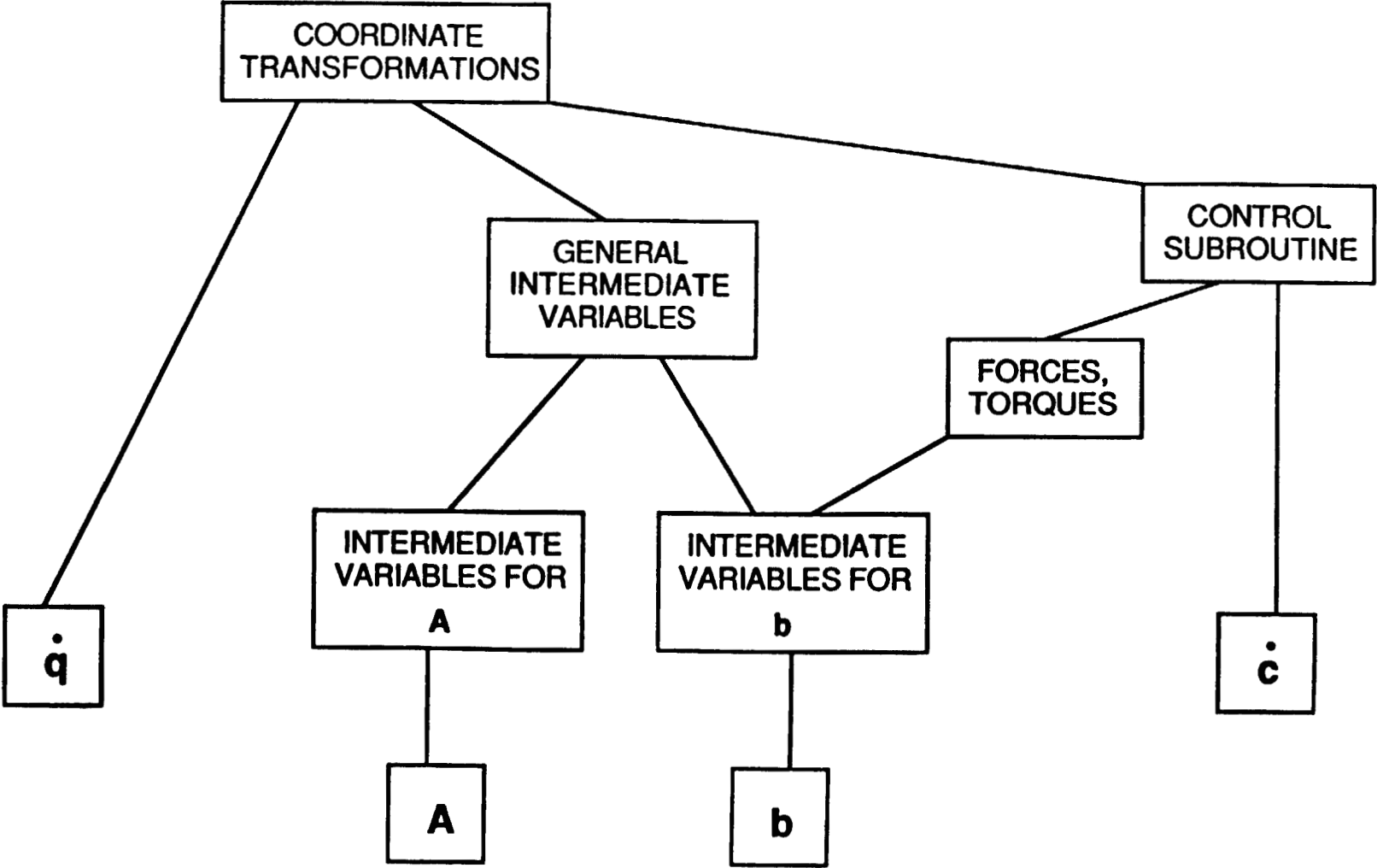
AN OVERVIEW OF THE CAPPS SIMULATION METHODOLOGY

An overview of the CAPPS simulation methodology is shown in the figure. The methodology is divided into four phases. The analysis phase constitutes the development of the mathematical model and requires an intensive interaction between the analyst and the previously described Program SYMBOD/SMP. The derivation phase follows, in which Program SMP carries out the instructions imbedded in SYMBOD to yield a matrix of ordinary differential equations in FORTRAN format. In addition to providing an accurate reflection of the analyst's intentions in the derivation of the equations of motion, this procedure also leads to equations which are virtually free of wasteful operations (such as additions of zeros, multiplications by unity and taking dot products of orthogonal vectors), as well as superfluous high order terms. The FORTRAN equations are delivered via file to the CAPPS computational load distribution software to begin the processing phase which is described in more detail below. It should be noted that the procedure completely eliminates the costly and time consuming programming effort which is normally required at this stage. The CAPPS system software transforms the derived equations from their original FORTRAN format to a binary format executable in concurrent operations by the CAPPS's CUs.
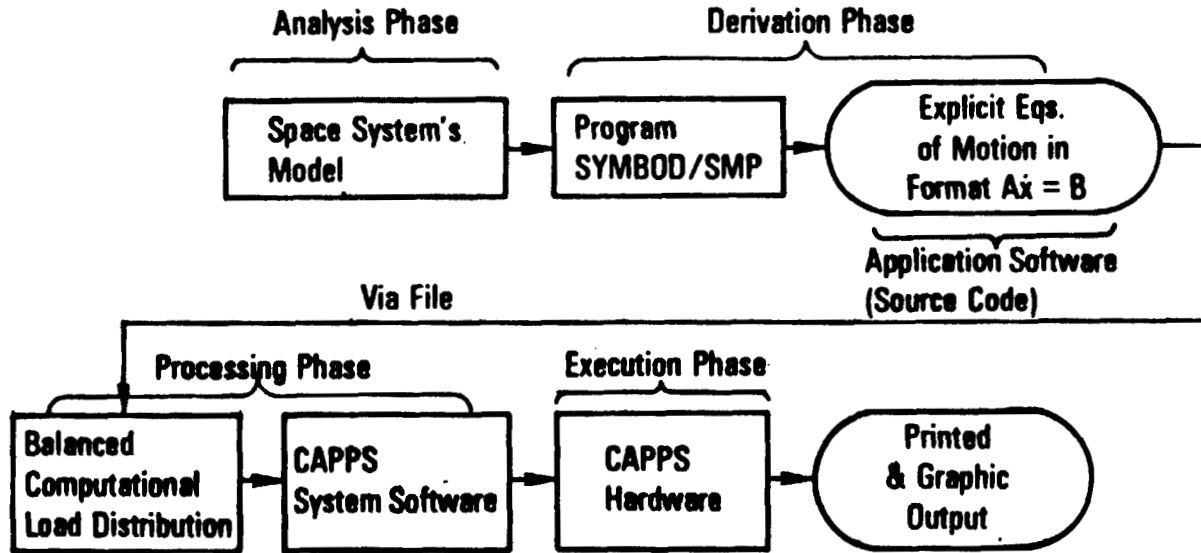
An Overview of the CAPPS Simulation Methodology

TRW

An Efficient Mathematical Model is Fed into a very Fast Computing System

REFERENCES

(1) C. Wampler, K. Buffinton, and J. Shu-hui, "Formulation of Equations of Motion for
     Systems Subject to Constraints"; ASME Journal of Applied Mechanics, Vol. 52, June
     1985.

(2) K. C. Park and J. C. Chiou, "Stabilization of Computational Procedures for
     Constrained Dynamical Systems"; University of Colorado, College of Engineering,
     Report No. Cu-CSSC-87-05, June 1987.

413

SIMULATION OF FLEXIBLE STRUCTURES WITH IMPACT: EXPERIMENTAL VALIDATION

By

A. Galip Ulsoy
University of Michigan
Ann Arbor, Michigan

ABSTRACT

The dynamics of flexible systems, such as mechanisms and robotic manipulators, is becoming increasingly important due to high-speed operation, high accuracy requirements, and lightweight designs. Such flexible devices can undergo impact during operation, and this may lead to: (1) component failure due to high stresses, and (2) excitation of higher structural modes due to impulsive forces. The latter issue, the simulation of the dynamic behavior of flexible structures with impact, is experimentally and numerically investigated.

A radially rotating flexible beam attached to a rigid shaft is considered. Both experimental and numerical studies are undertaken. Experimental studies show excellent agreement with simulation studies using both the momentum balance (coefficient of friction), and spring-dashpot (impact pair) models. The simulation studies are even capable of predicting the high-speed multiple impacts which occur due to structural flexibility and which were experimentally observed using high-speed video photography. The results of the studies show that a simple momentum balance (coefficient of restitution) method for simulating the impact is sufficiently accurate in predicting the dynamic behavior of the system for most engineering applications. The momentum balance method cannot simulate the impact force which develops during the contact duration, but is computationally very efficient. The spring-dashpot model is more difficult to develop and requires significantly larger computation time, but can simulate impact forces and stresses due to impact.

The momentum balance (coefficient of restitution) method, although strictly not applicable to flexible bodies, has been shown to provide an accurate and computationally efficient method for simulating the dynamic behavior of flexible structures with impact when contact stresses are not needed.

SIMULATION AND CONTROL PROBLEMS IN ELASTIC ROBOTS

By

S. S. K. Tadikonda and H. Baruh
Rutgers University
New Brunswick, New Jersey

ABSTRACT

Computational issues associated with modeling and control of robots with revolute joints and elastic arms are considered. A manipulator with one arm and pinned at one end is considered to investigate various aspects of the modeling procedure and the model, and the effect of coupling between the rigid-body and the elastic motions.

The rigid-body motion of a manipulator arm is described by means of a reference frame attached to the "shadow beam," and the linear elastic operator denoting flexibility is defined with respect to this reference frame. The small elastic motion assumption coupled with the method of assumed modes is used to model the elasticity in the arm. The complete model coupling the rigid-body and the elastic motion is highly nonlinear, and contains terms up to quartic in powers of the amplitudes of the assumed modes. It is shown that only terms up to quadratic in these model amplitudes need to be retained.

An important aspect of the coupling between the rigid-body and the elastic motion is the centrifugal stiffening effect. This effect stiffens the elastic structure, as to be expected on physical grounds, gives rise to a time-varying inertia term for the rigid-body motion, and, in general, results in an effective inertia term smaller than the rigid-body inertia term. In fact, this reduction in inertia determines the limitation of the small motion assumption. If the elastic behavior is excited sufficiently so as to cause a vanishing effective rigid-body motion inertia term, one should either modify the manipulator model, or consider the forcing profiles that excite the elastic motion least. The Fourier series expansion of a few such profiles is examined to provide insight in this regard.

Simulation results are presented for an elastic beam pinned at one end and free at the other, and rotating in a horizontal plane, and control issues such as the order of the model, number of sensors, and modal extraction are examined within this context. It is shown that the effect of centrifugal stiffening is pronounced on the rigid-body motion during transition, and ignoring it in the control model leads to gross inaccuracies in response. The effect of including varying amounts of flexibility on the response is studied.

417

SIMULATION AND CONTROL PROBLEMS

IN ELASTIC ROBOTS

S. S. K. Tadikonda[*]

H. Baruh[**]

Mechanical & Aerospace Engineering Dept.,
Rutgers University, New Brunswick, NJ.

* Graduate Assistant

** Assistant Professor

OBJECTIVE

To investigate modeling, control, and computational issues associated with elastic manipulators

SCOPE

Revolute joints

Actuators at joints only

Shadow beam approach

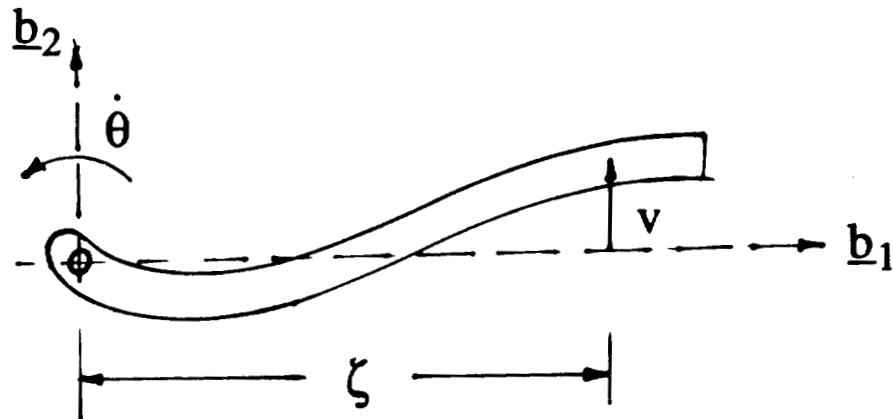Small elastic motion, and limit of such an assumption
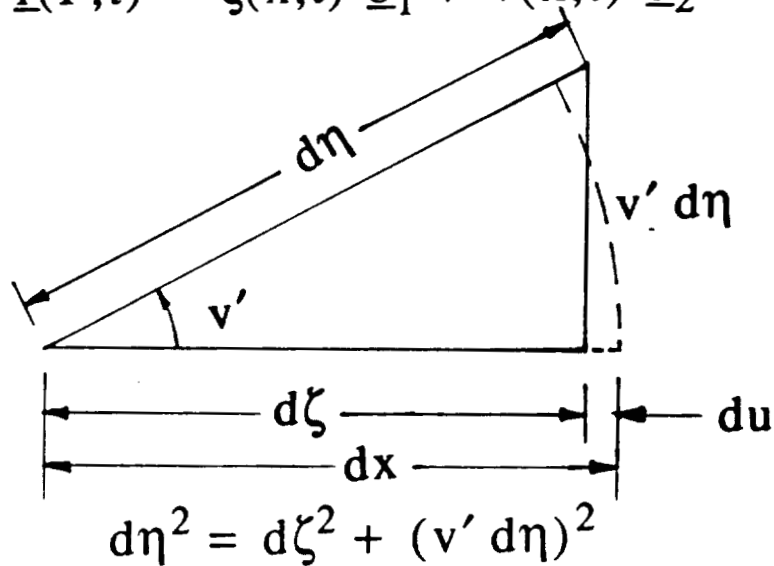
Nonlinear model

Control issues

Illustrative example

Pinned - free link

Reference frame located at the pin joint; describes rigid-body motion. Elastic motion is defined with respect to this frame

$$\underline{r}(P,t) = \zeta(x,t)\ \underline{b}_1 + v(x,t)\ \underline{b}_2$$

$$d\eta^2 = d\zeta^2 + (v'\ d\eta)^2$$

Notes: x is the position of the point in the undeformed configuration
The beam rotates in a horizontal plane

u(x,t) is obtained by integrating

$$d\zeta = [\ 1 - (\frac{\partial v}{\partial x})^2\ ]^{1/2}\ d\eta$$

where

$$\eta(x,t) = x + s(x,t)$$

$$\zeta(x,t) = x - u(x,t)$$

On integration,

$$\zeta(x,t) \approx \eta - 1/2\int_0^\eta [\ (\frac{\partial v}{\partial \sigma})^2\ ]\ d\sigma$$

or

$$u(x,t) \approx -s(x,t) + 1/2\int [(\frac{\partial v}{\partial \sigma})^2\ ]\ d\sigma$$

s(x,t) : axial vibration term

Integral : results in centrifugal stiffening term

Neglect axial vibration

Elastic Displacement

$$\underline{u}(x,t) = - u(x,t)\ \underline{b}_1 + v(x,t)\ \underline{b}_2$$

Position

$$\underline{r} = \zeta(x,t)\ \underline{b}_1 + v(x,t)\ \underline{b}_2$$

Velocity

$$\underline{\dot{r}} = - \frac{\partial u}{\partial t}\ \underline{b}_1 + \frac{\partial v}{\partial t}\ \underline{b}_2 + \underline{\dot{\theta}} \times \underline{r}$$

Kinetic Energy

$$K = 1/2 \int \underline{\dot{r}} \cdot \underline{\dot{r}}\ dm$$

Potential Energy

$$V = 1/2 \int_0^L EI(x)\ (v'')^2\ dx$$

Lagrangian

$$L = K - V$$

Notes : ($\cdot$) corresponds to partial derivative with respect to time, ( )′ corresponds to spatial derivative, ρ is the mass per unit length, and EI(x) is the

flexural rigidity

$$L = 1/2 \int_0^L \rho \, (\dot{v}^2 + \dot{\theta}^2 \, v^2 + 2 \, x \, \dot{v} \, \dot{\theta} + x^2 \, \dot{\theta}^2) \, dx$$

$$- \, 1/2 \int_0^L EI(v'')^2 dx$$

$$- \, 1/2 \, \dot{\theta}^2 \int_0^L \rho \, x \int_0^x (v')^2 d\sigma \, dx$$

$$+ \, 1/2 \int_0^L [-1/2 \int_0^x \frac{d}{dt}((v')^2) d\sigma]^2 \rho \, dx$$

$$+ \, 1/2 \, \dot{\theta} \int_0^L \rho \, v \int_0^x \frac{d}{dt}((v')^2) \, d\sigma \, dx$$

$$- \, 1/2 \, \dot{\theta} \int_0^L \rho \, \dot{v} \int_0^x (v')^2 d\sigma \, dx$$

$$+ \, 1/2 \, \dot{\theta}^2 \int_0^L \rho/4 \, [\int_0^x (v')^2 d\sigma]^2 \, dx$$

Assumed Modes

$$v(x,t) = \sum_{i=1}^{N_1} \phi_i(x) \, a_i(t)$$

$\phi_i(x)$: Admissible functions

424

Define

$$m_{ij} = \int_0^L \rho \phi_i(x) \, \phi_j(x) \, dx$$

$$k_{ij} = \int_0^L EI(x) \, \phi_i''(x) \, \phi_j''(x) \, dx$$

$$s_{ij}(x) = \int_0^x \phi_i'(x) \, \phi_j'(x) \, dx$$

$$p_{ij} = \int_0^L \rho \, x \, s_{ij}(x) \, dx$$

$$s_{ijkl} = \int_0^L s_{ij}(x) \, s_{kl}(x) \, dx$$

$$q_{ijk} = \int_0^L \rho \, \phi_k(x) \, s_{ij}(x) \, dx$$

$$r_i = \int_0^L \rho \, x \, \phi_i(x) \, dx$$

Then,

$$\int_0^x (v')^2 \, dx = s_{ij}(x) \, a_i \, a_j$$

$$\int_0^L \rho \, x \, [\int_0^x (v')^2 \, d\sigma] \, dx = p_{ij} \, a_i \, a_j$$

$$\int_0^L [\int_0^x \frac{d}{dt}((v')^2) \, d\sigma]^2 \rho \, dx = 4 \, s_{ijkl} \, a_i \, \dot{a}_j \, a_k \, \dot{a}_l$$

$$\int_0^L \rho \, v \, [\int_0^x \frac{d}{dt}((v')^2) \, d\sigma] \, dx = 2 \, q_{ijk} \, a_i \, \dot{a}_j \, a_k$$

425

Substituting in the Lagrangian,

$$L = 1/2 \, I_o \, \dot{\theta}^2 + 1/2 \, m_{ij} \, \dot{a}_i \, \dot{a}_j + \dot{\theta} \, r_i \, \dot{a}_i$$

$$- 1/2[k_{ij} + (p_{ij} - m_{ij})\dot{\theta}^2] \, a_i \, a_j$$

$$+ 1/2 \, s_{ijkl} \, a_i \, \dot{a}_j \, a_k \, \dot{a}_l + \dot{\theta}^2/8 \, s_{ijkl} \, a_i \, a_j \, a_k \, a_l$$

$$+ \dot{\theta} \, q_{ijk} \, a_i \, \dot{a}_j \, a_k - \dot{\theta}/2 \, q_{ijk} \, a_i \, a_j \, \dot{a}_k$$

Example :
Beam parameters

> Cross-section : 6 in x 3/8 in
> Length =   3.6576 m (12 ft)
> ρ =   4.015 kg/m
> EI = 756.65 N . m

Admissible   functions   :   Normalized eigenfunctions of a pinned-free beam

$$m_{ij} = \delta_{ij} \, ; \quad k_{ij} = \omega_i^2 \, \delta_{ij}$$

where δ_{ij} is the Kronecker delta

Notes : The summation convention, $\sum\sum m_{ij} \, a_i \, a_j = m_{ij} \, a_i \, a_j$, etc., will be employed for conciseness - i.e., repeated indices in an expression indicate summation over appropriate range.

Natural frequencies and Centrifugal stiffening coefficients

i	ω_i	P_{ij}		
		$j = 1$	$j = 2$	$j = 3$
1	15.82	6.397	1.861	-0.366
2	51.282	1.861	17.905	6.195
3	106.983	-0.366	6.195	35.999

Coriolis terms, q_{ijk}

i	j	$k = 1$	$k = 2$	$k = 3$
1	1	-0.152	0.143	0.008
1	2	0.415	-0.144	0.169
1	3	0.077	0.347	-0.143
2	1	0.415	-0.144	0.169
2	2	-0.175	0.152	-0.117
2	3	0.883	-0.196	0.145
3	1	0.077	0.347	-0.143
3	2	0.883	-0.196	0.145
3	3	-0.178	0.171	-0.152

Other coupling terms s_{ijkl}

Note: $s_{ijkl} = s_{jikl} = s_{ijlk} = s_{jilk}$

i	j	k	$l = 1$	$l = 2$	$l = 3$
1	1	1	0.669	0.099	0.001
1	1	2	0.099	1.800	0.444
1	1	3	0.001	0.444	3.570

1	2	1	0.099	0.275	-0.050
1	2	2	0.275	0.397	0.594
1	2	3	-0.050	0.594	0.901
1	3	1	0.001	-0.050	0.183
1	3	2	-0.050	-0.005	-0.074
1	3	3	0.183	-0.074	-0.113
2	1	1	0.099	0.275	-0.005
2	1	2	0.275	0.397	0.594
2	1	3	-0.050	0.594	0.901
2	2	1	1.800	0.397	-0.005
2	2	2	0.397	5.010	0.147
2	2	3	-0.005	0.147	9.94
2	3	1	0.444	0.594	-0.074
2	3	2	0.594	1.470	1.500
2	3	3	-0.074	1.500	3.160
3	1	1	0.001	-0.050	.183
3	1	2	-0.050	-0.005	-0.074
3	1	3	0.183	-0.074	-0.113
3	2	1	0.444	0.594	-0.074
3	2	2	0.594	1.470	1.500
3	2	3	-0.074	1.500	3.160
3	3	1	3.57	0.901	-0.113
3	3	2	0.901	9.94	3.160
3	3	3	-0.113	3.160	20.000

Notes : The magnitudes of the terms q_{ijk} and s_{ijkl} are small. In addition, they are multiplied by the cubic and quartic powers of modal amplitudes. Hence they will be dropped from further development.

428

Retaining terms only up to quadratic in modal amplitudes,

$$L = 1/2 \, [I_o - (p_{ij} - m_{ij}) \, a_i \, a_j] \, \dot{\theta}^2 + 1/2 \, m_{ij} \, \dot{a}_i \, \dot{a}_j$$

$$- 1/2 \, k_{ij} \, a_i \, a_j$$

The equation for rigid-body motion is

$$\frac{d}{dt} [(I_o - (p_{ij} - m_{ij}) \, a_i \, a_j) \, \dot{\theta}] = T$$

And the elastic motion is described by

$$m_{ij} \, \ddot{a}_j + [k_{ij} + (p_{ij} - m_{ij}) \, \dot{\theta}^2] \, a_j = T \, \phi_i'(0),$$

$$i = 1, \, 2, \, ..., \, N_1$$

Measurements at x = 0

$$\theta_1 = \theta + v'(0,t)$$

$$\dot{\theta}_1 = \dot{\theta} + \dot{v}'(0,t)$$

429

Choices for the control model

1. Ignore elastic effects completely

$$\text{Control model}: \quad I_o \, \ddot{\theta} = T$$
$$\theta = \theta_1 \qquad\qquad \dot{\theta} = \dot{\theta}_1$$

2. Rigid-body model, with the shadow frame angle properly extracted

$$\text{Control model}: \quad I_o \, \ddot{\theta} = T$$
$$\theta = \theta_1 - v'(0,t)$$
$$\dot{\theta} = \dot{\theta}_1 - \dot{v}'(0,t)$$

3. A few elastic modes are included, and the modal coordinates are approximated

Control model:
$$\frac{d}{dt}[(I_o - (p_{ij} - m_{ij}) \, a_i \, a_j) \, \dot{\theta} \,] = T$$
$$i,j \leq N_2, \quad N_2 < N_1$$

Notes : N_1 is the number of modeled modes. N_2 is the number of modes used for controller design. $N_1 = 3$ for the following simulation results.

4. Appropriate number of sensors used to obtain accurate modal coordinates.

$$N_2 = N_1$$

Control synthesis

Computed torque method

Pointwise-optimal control method

Open-loop Maneuver
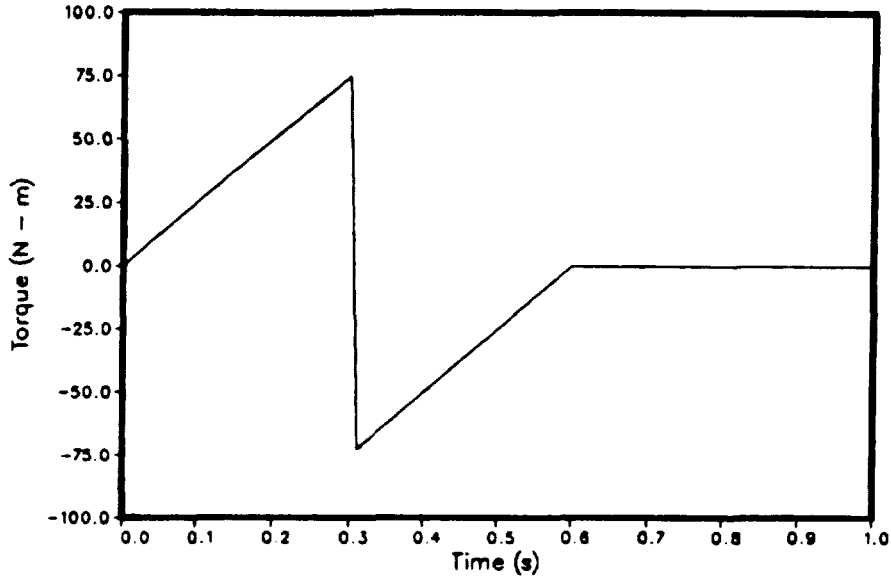
Fig. 1 : Torque Profile for Open-Loop Maneuver
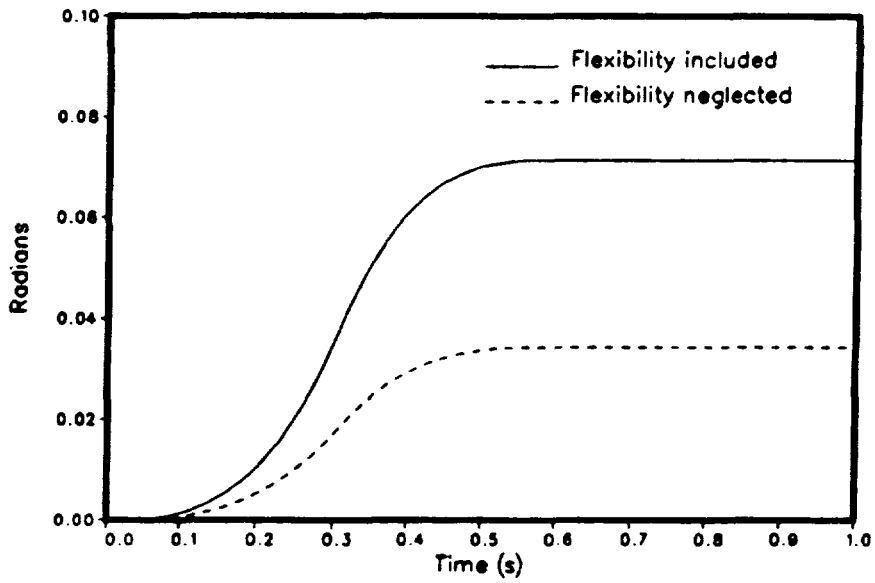
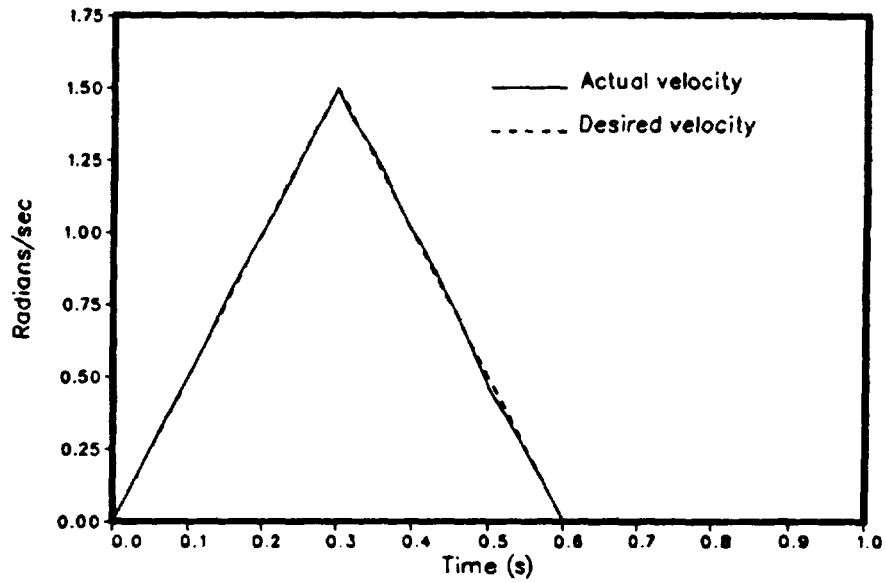Fig. 2 : Position Response of the Beam for the Torque Above

432

Feedback Control
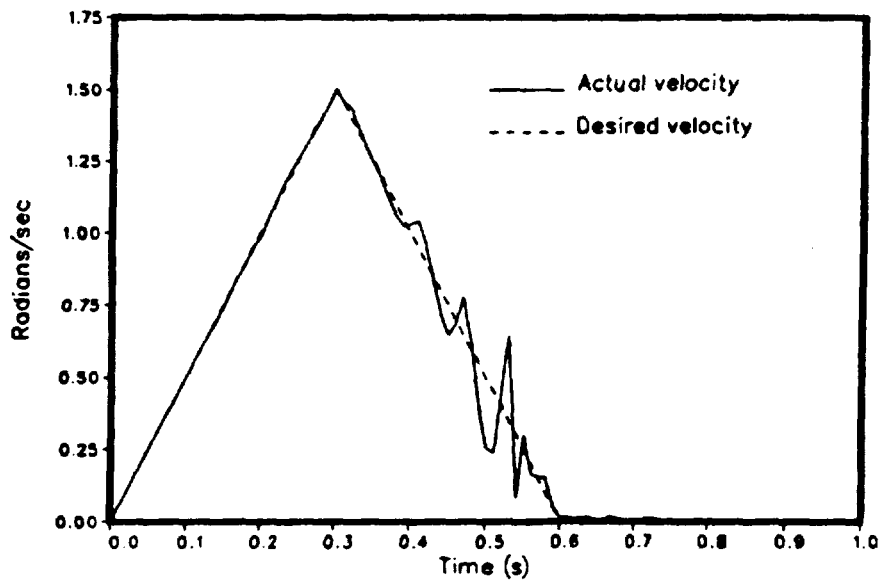

Fig. 3 : Rigid Model, Velocity response

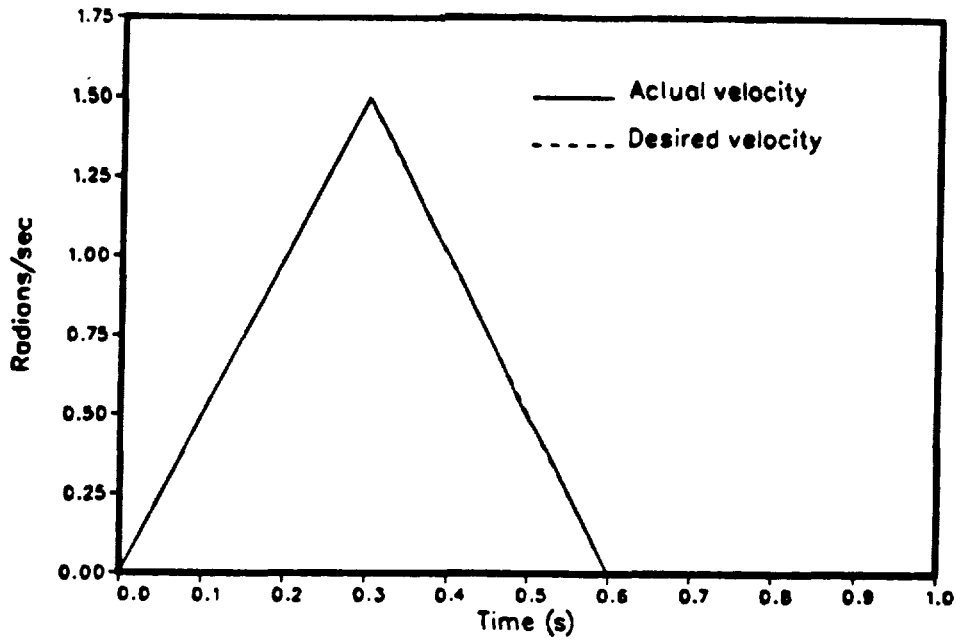Fig. 4 : One Flexible Mode Included in the Model

433

Feedback Control, contd.

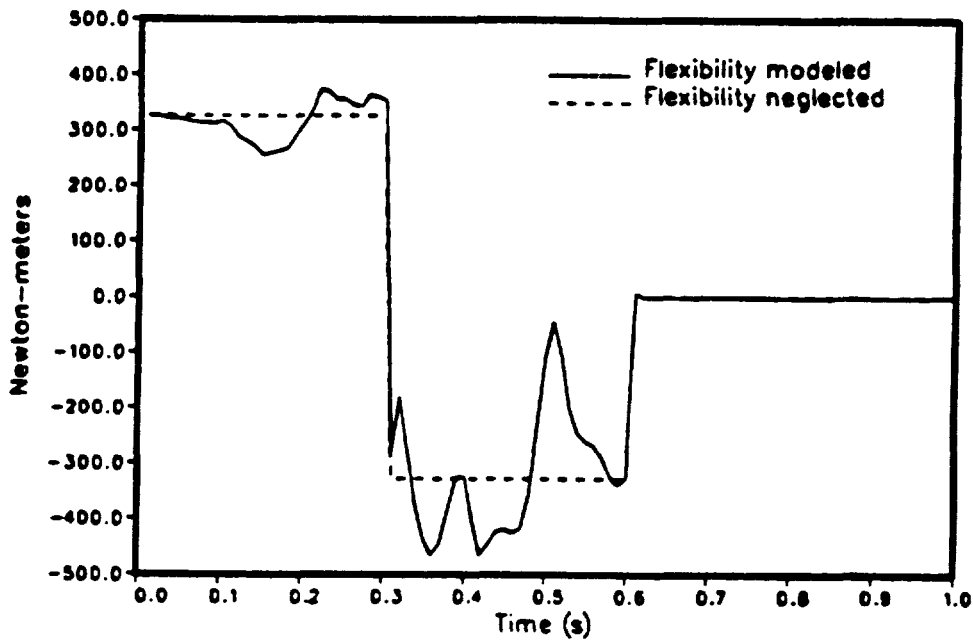Fig. 5 : Three Flexible Modes Included in the Model

Fig. 6 : Comparison of Open- and Closed-Loop
Torques

434

Effects of centrifugal stiffening

1. Provides a strong coupling between the rigid-body and elastic motion

2. Increases the stiffness of the structure

3. Reduces the effective rigid-body inertia term. Can cause it to vanish if the elastic motion is large. May have to modify the model, or vary the torque profiles.

Torque profiles and their Fourier coefficients

	Coefficients of	
	$\sin(n\omega_0 t)$	$\cos(n\omega_0 t)$
	$\dfrac{(-1)^{n-1}}{n\omega_0}$	0
	$\dfrac{(1-(-1)^n)}{3n\omega_0}$	$\dfrac{2((-1)^n - 1)}{3\pi\omega_0 n^2}$
	$\dfrac{(1-(-1)^n)}{n\omega_0}$	0
	$n = 2m+1$	
	$\dfrac{8}{3\pi\omega_0 n^2}$	0

$$\int_0^T \int_0^T \text{Torque} \cdot dt \quad \text{is the same for all cases}$$

$$\omega_0 = \frac{2\pi}{T}$$

436

Computational Issues for Control of multi-link flexible robot arm

1. The dynamic model can be arrived at by modeling each link independently and imposing constraints at the joints

2. The link geometry may not be simple

3. s_{ijkl}, q_{ijk}, may not be negligible, and the control model may include all the terms

4. The choice of admissible functions for each of the links may be different

5. Sampling rates - should not excite elastic motion

6. Control input computation may pose formidable burden.

The above issues can be adequately addressed by selecting pointwise-optimal control law for control input computations, where, the inputs can be computed at least one time step ahead.

Conclusions

1. A complete model for control of a flexible link is developed

2. Modeling issues are examined within the context of an example

3. Several control issues are investigated

4. It is shown that centrifugal stiffening effect on rigid-body motion is significant

5. There is a strong coupling between rigid-body and elastic motions; ignoring this coupling results in gross inaccuracies in response.

LINEARIZED FLEXIBILITY MODELS IN MULTIBODY DYNAMICS AND CONTROL

By

William W. Cimino
Boeing Aerospace
Seattle, Washington

ABSTRACT

This presentation discusses simulation of structural response of multi-flexible-body systems by linearized flexible motion combined with nonlinear rigid motion. Advantages and applicability of such an approach for accurate simulation with greatly reduced computational costs and turnaround times are described, restricting attention to the control design environment. Requirements for updating the linearized flexibility model to track large angular motions are discussed. Validation of such an approach by comparison with other existing codes is included. Application to a flexible robot manipulator system is described.

PRECEDING PAGE BLANK NOT FILMED

PAGE 440 INTENTIONALLY BLANK

Linearized Flexibility Models in Multibody Dynamics and Control

12 July 1988

William W. Cimino

Boeing Aerospace
Seattle, Washington

442

- Some controls requirements of multibody codes

- Introduction to SADACS

- Validation

- Applications

  - Spacecraft

  - Robot manipulators

443

Some Controls Requirements

1) General purpose dynamic module

2) Models can be merged in any configuration without creating new structural models

3) Very fast (computationally inexpensive)
   - Short simulation turnover time
     - Time domain analysis with nonlinear controllers
       - Sensitivity studies
       - Stability analysis
   - Control design iteration

444

SADACS
Spacecraft Appendage Dynamics and Control Simulation

- **Dynamics simulation of multi-flexible-body systems**

- **Designed for controls engineers/controls environment**

- **Approximate code to address controls requirements**

  - **Linearized flexible modal analysis with "configuration update"**

SADACS is designed for controls environment

- Used as general purpose dynamics module in a control simulation environment

- Allows multibody systems to be merged in any desired configuration without creating new structural models

- Very fast (computationally inexpensive) for system design and sensitivity/stability analysis

446

- **How fast is SADACS?**

  - **Problem dependent**

  - **Large complex models   100-500 times faster than DISCOS**

- **Why is it fast?**

  - **System modes**
    - **Diagonalized, linear, constant coefficient flexible equations of motion**
    - **Truncation (with residualization) to increase Δt**
    - **Use 'explicit' integration**

- **Propagate linear system until 'update'**

CPU Time Comparisons for 3-Body Problem

CODE	RUN TIMES	REMARKS
DISCOS	> 26 hours	No component modal truncation
DISCOS	5 hours 9 minutes	With component modal truncation
SADACS	10 minutes	With system modal truncation

Other Test Problems :

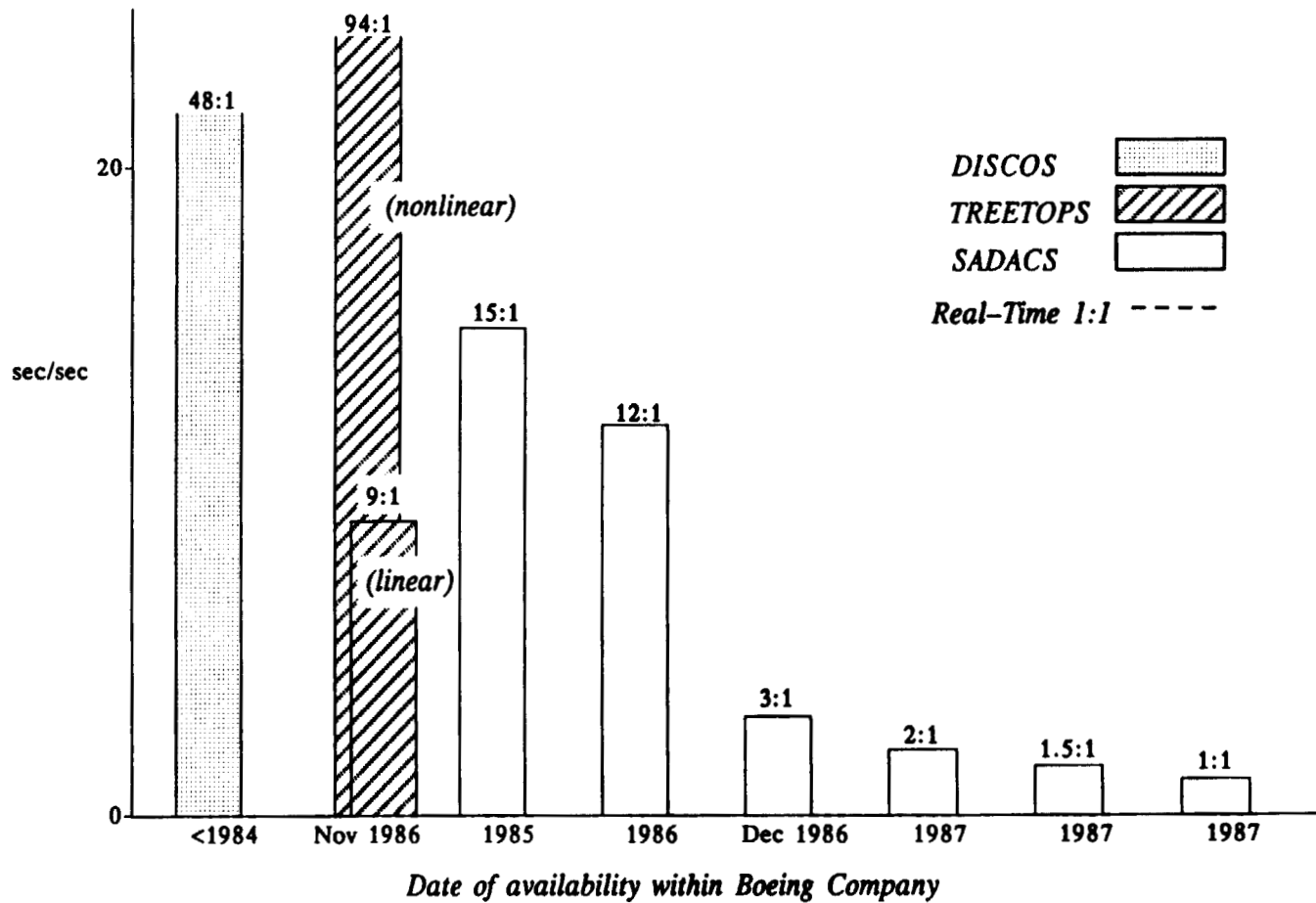*High Speed Simulation of Flexible Mulitbody Dynamics*
Presented at MSFC, April 22–24, 1986

CODE	RUN TIMES	REMARKS
DISCOS	$\simeq$ 50 Hours	With component modal truncation
SADACS	$\simeq$ 15 minutes	With system modal truncation

Multi–Flexible–Body Run–Times
3–Body simulation CPU (seconds) / Real–Time (seconds)

Date of availability within Boeing Company

SADACS Structure

1) **Nonlinear rigid body code**
   (SD/EXACT, TREETOPS, MBDYN, etc.)

2) **Linear flexible dynamics**

   - **System mode formulation**

   - **Retain truncated modes quasi-statically**

3) **System mode update/restart**

450

Simplified Diagram of the SADACS Code

Inputs to the system (generalized forces and torques) are applied to both the rigid and flex motion solvers. The rigid motion solver computes the nonlinear rigid body response. The flex motion solver uses a "system mode" formulation to compute the linearized flexible response. The outputs are combined in the motion summer and tested for an "update condition". If an update is not required the outputs are passed out to the simulation. If an update is required, a new eigensolution is performed on the new configuration and the mode shapes, frequencies, and system mode state vector are adjusted.
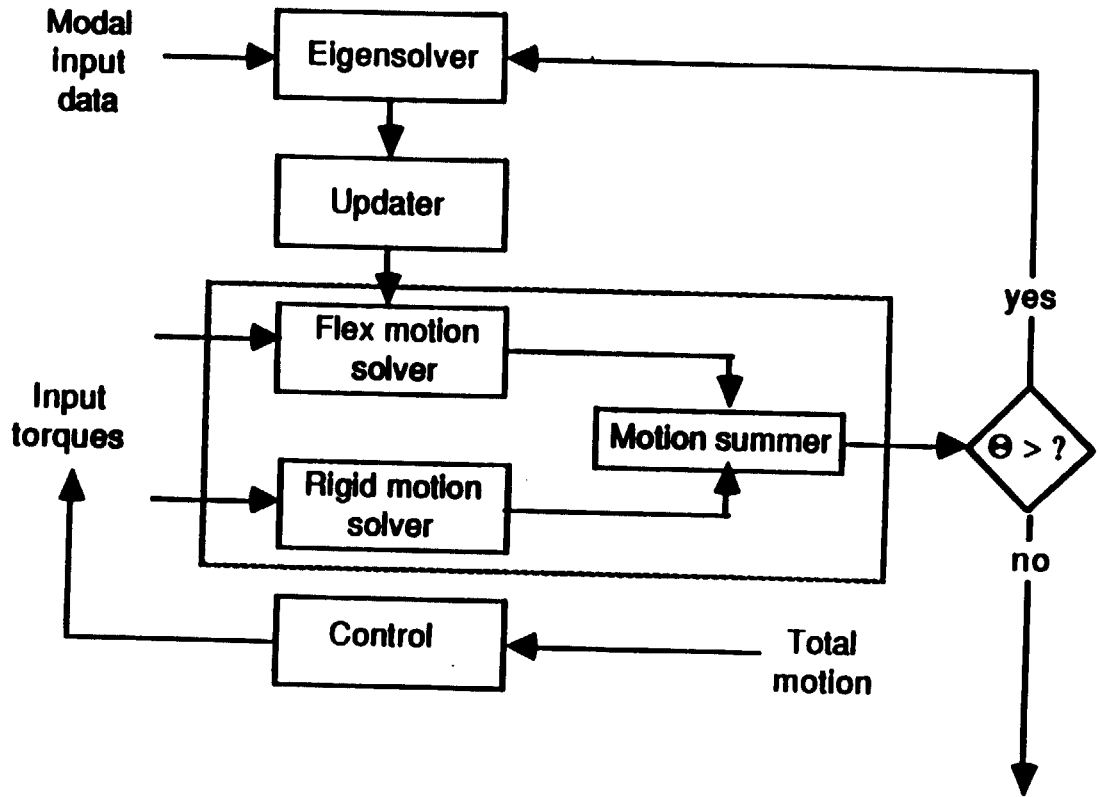
Simplified Diagram of SADACS Code

The Difficult Technical Problem:

Large angle motions with linear flexibility

- System modes change (shape, frequency) with angular position

- Track changes by 'updating' system modes

    - Update at predetermined angles or time

        - Shape
        - Frequencies
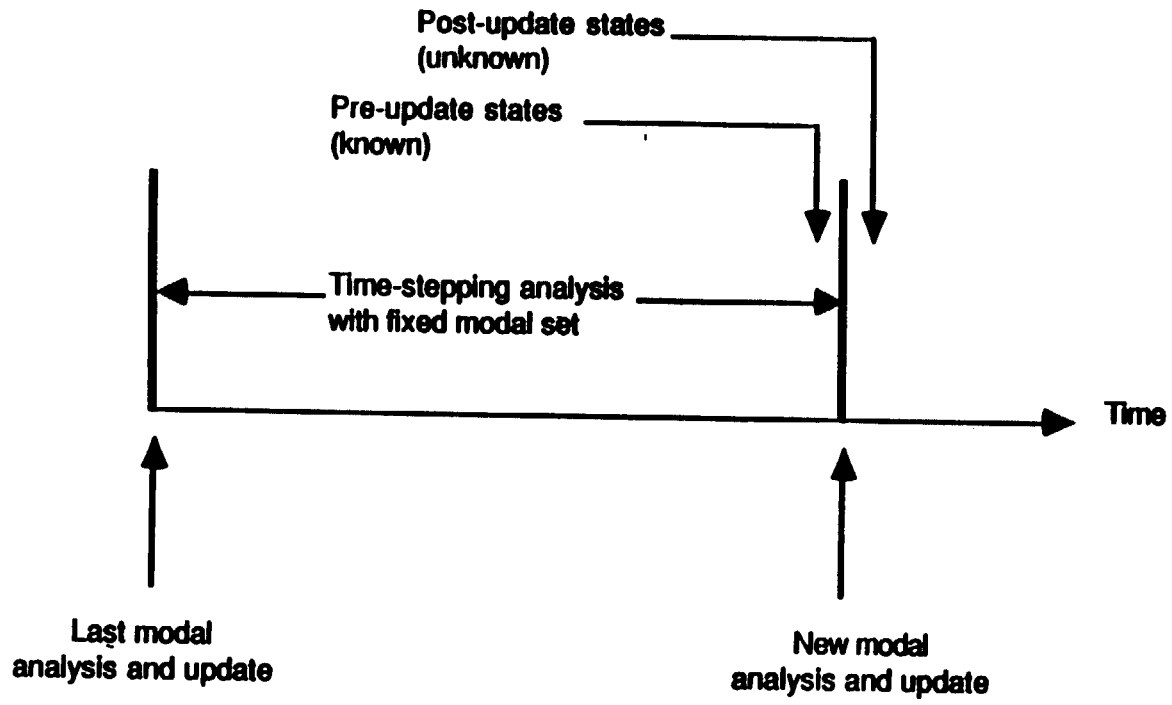   - Transfer functions

    - Restart dynamic analysis

Updating Time-Line Overview

A fixed modal set is maintained during a given epoch (time between updates). At the end of the current epoch the pre-update states are known. Following the eigensolution on the new system matrices, the new modes shapes and frequencies are known. The difficult part of the update is then to assign new values to the post-update states.

454

THIS PAGE INTENTIONALLY LEFT BLANK

455

456

Post-update states
(unknown)

Pre-update states
(known)

Time-stepping analysis
with fixed modal set

Time

Last modal
analysis and update

New modal
analysis and update

Updating Time-Line Overview

Why is updating a problem?

When gimbal rotations and rates (which include structural deformations) developed in one configuration are imposed on a new configuration, they excite the structure in shapes (modes) that would not have occurred in a 'continuous' solution, and in addition fail to preserve energy.

The problem is nonlinearity-induced trading of excitation, or coupling, between the modes.

457

Example problem: chosen to emphasize 'trade' in modal participation

- **Want update that**

    1) **Doesn't ring**

    2) **Maintains energy**

    3) **Tracks frequencies**

    4) **Correct shapes**

- **Coupling of flex into rigid neglected**

    - **SADACS not intended for problems where flex nonlinearities drive rigid motion**

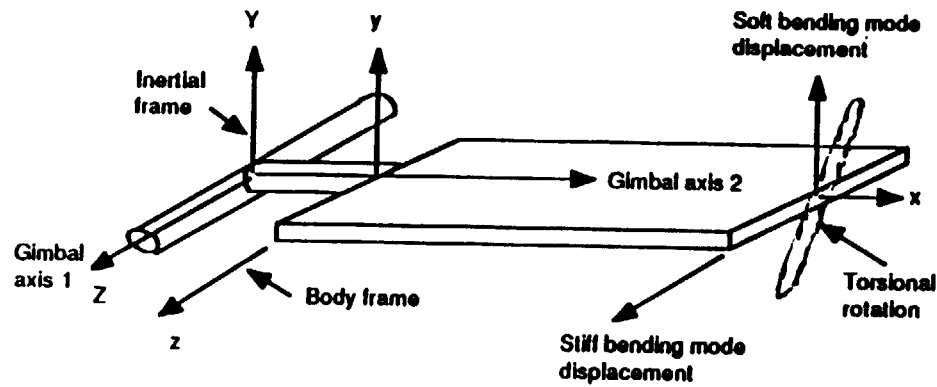- **Address update entirely with component modal variables**

Example Problem

This figure shows the system used to examine the update. The system has two flexible modes with coordinates q_1 (soft mode) and q_2 (stiff mode).

- **Rigid coordinates Θ_1, Θ_2**

- **Flex coordinates q_1, q_2**

- **Soft mode = 1.59 Hz**

- **Stiff mode = 3.18 Hz**

- **$\dot{\Theta}_2$ = 9 deg/sec (constrained)**

- **Initial conditions:**
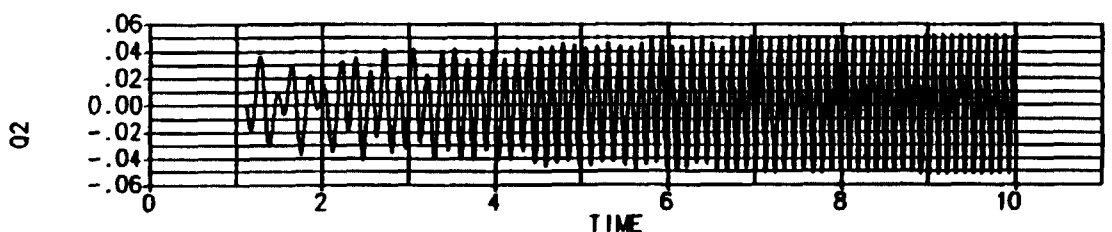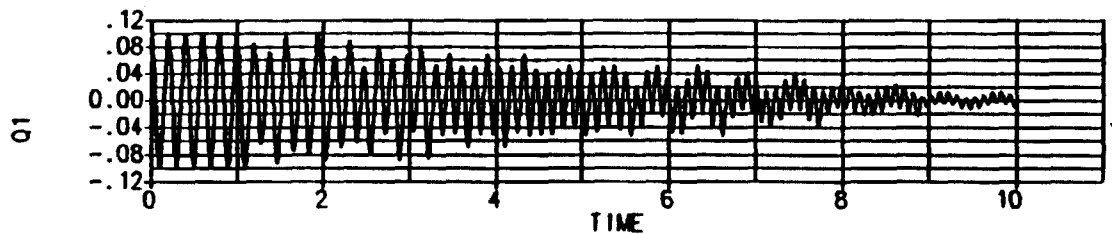 $$\Theta_1 = -.03 \text{ rad}$$
  $$q_1 = .1$$

SADACS vs. Continuous Solution

The top set of plots show the flexible coordinate response when the
equations of motion include flex/rigid coupling and are integrated
in a continuous manner. The bottom set of plots show the flexible
coordinate response using the SADACS approach.

461
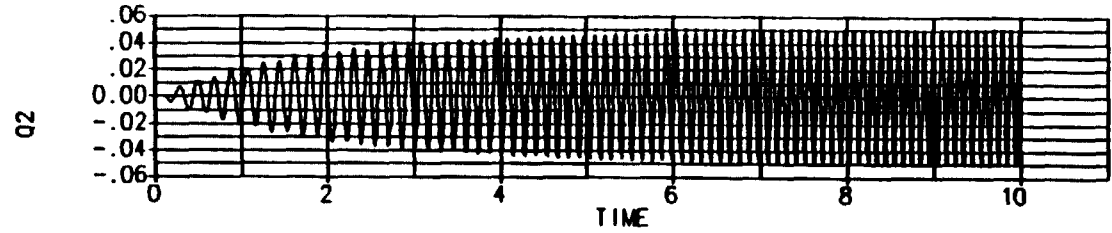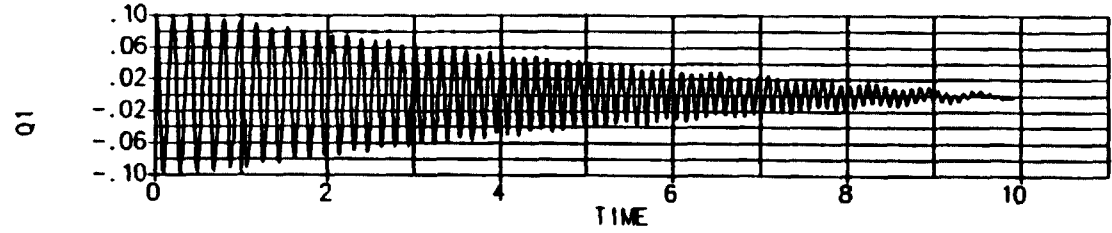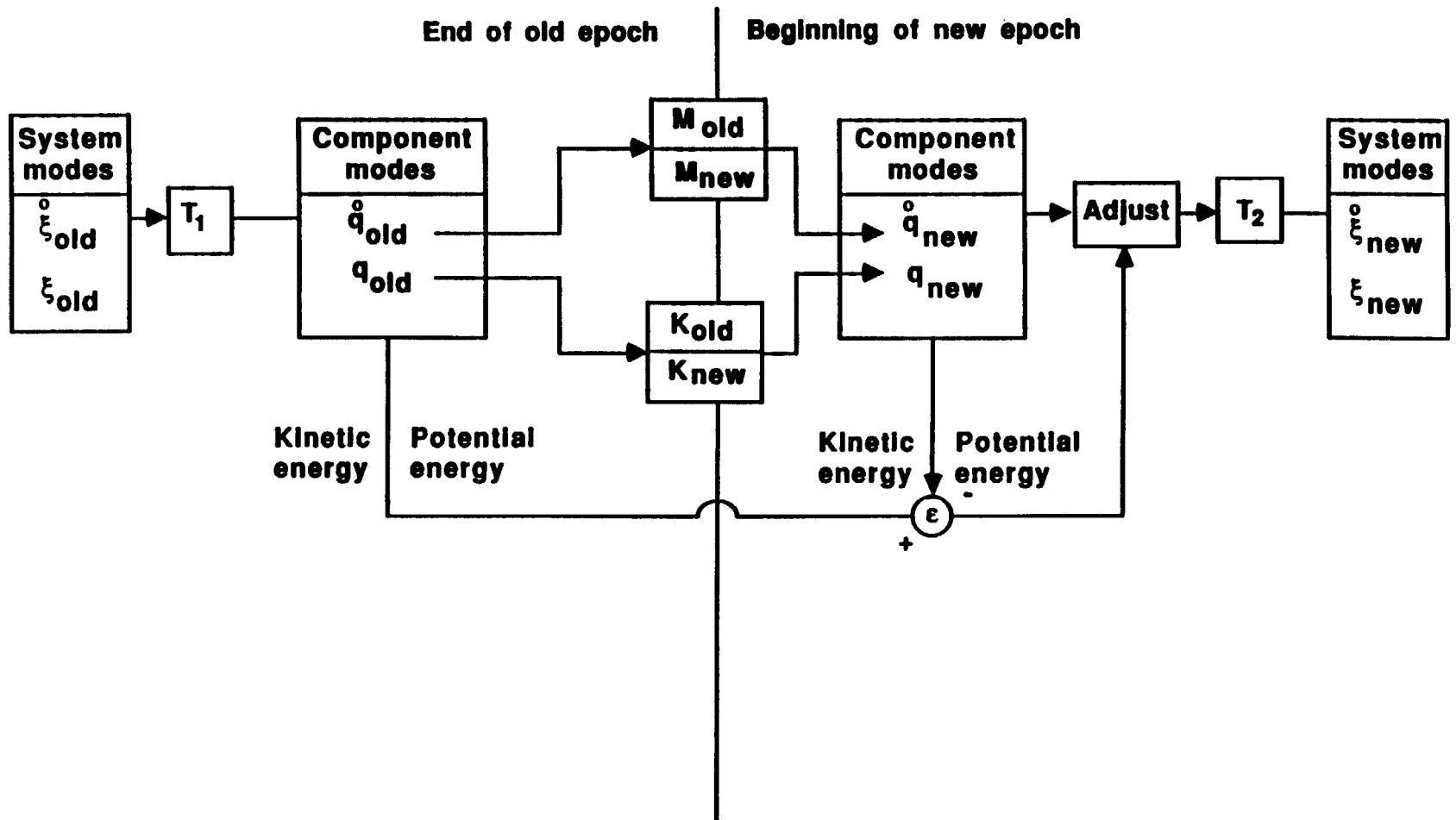
Continuous Solution
with Nonlinear Loads
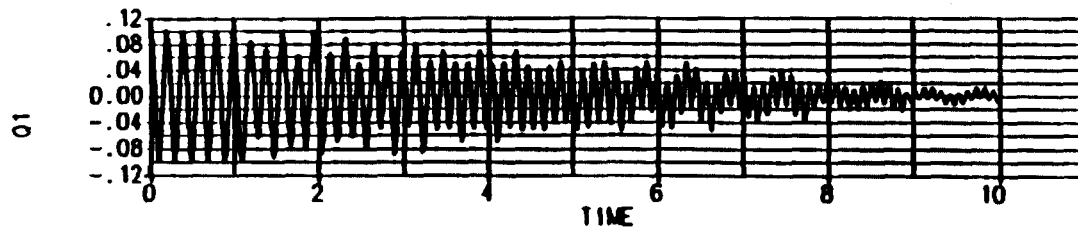
Updated Solution Based on Momentum
with Energy Balance

Momentum/Stress Update with Energy Balance

End of old epoch   **Beginning of new epoch**
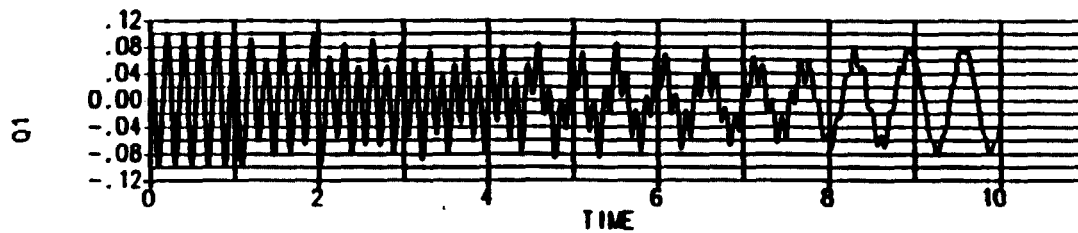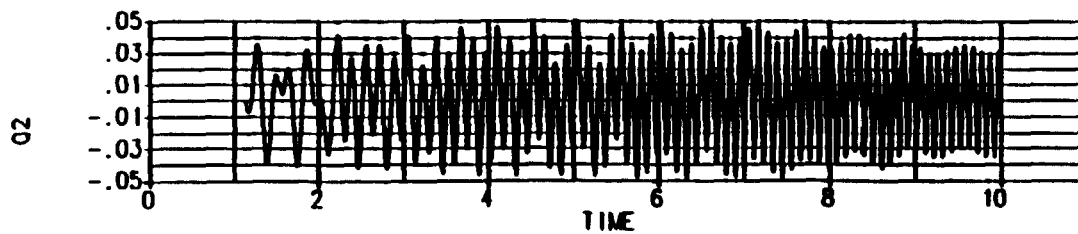
463

88m-34-108b
6-7-88

Updated Solution Based on Momentum with Energy Balance
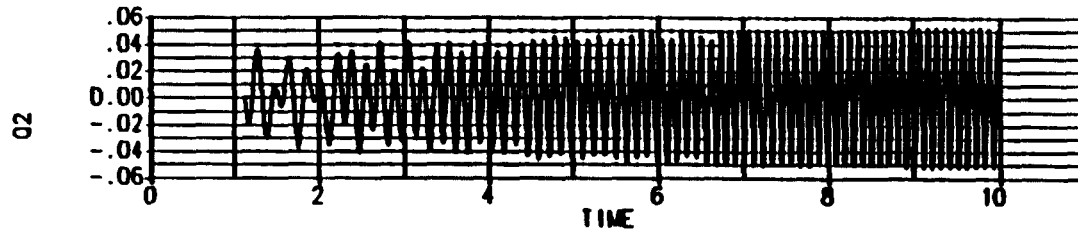
Updated Solution Based on Phase Variables with Energy Balance

464

FB2 Topology Capabilities

CLUSTERED

CLOSED LOOP

MANIPULATOR

TREE

MANIPULATOR

LINKED
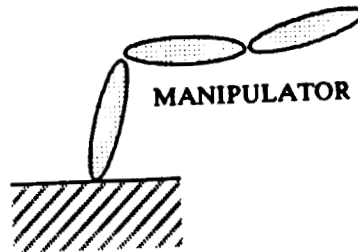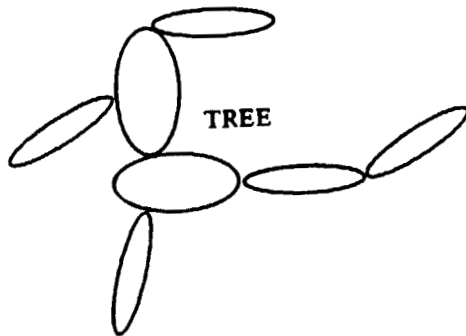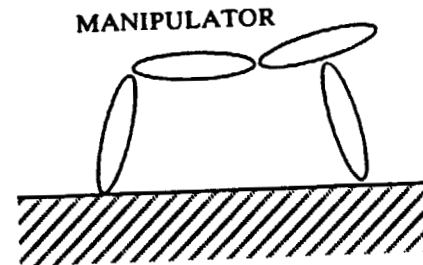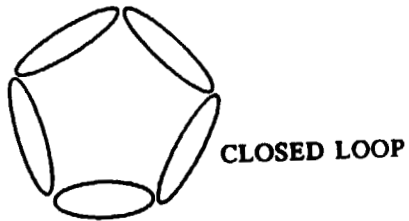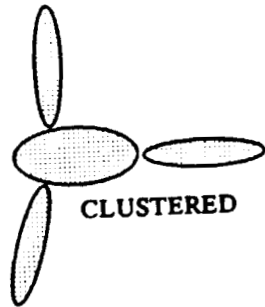
SADACS program capabilities
Summary

- Rigid body analyser

    - based upon the code used (MBDYN, TREETOPS, SD/EXACT etc)

- Flex body analyser (FB2)

    - Number of bodies   no limit
    - Number of flexural modes/body   no limit
    - Number of gimbals   no limit

    - Configurations :   cluster
                                            linked
                                            tree
                                            closed loop
                                            manipulator
                                            multiple closed loop
                                            multiply grounded manipulator

    - Degrees of freedom at gimbals $0 \rightarrow 6$ (totally locked to totally free)

DISCOS-SADACS Comparison:
Main Body Sensor X Rotation Due To Appendage Command

022P1-RJ-136/New

467

3
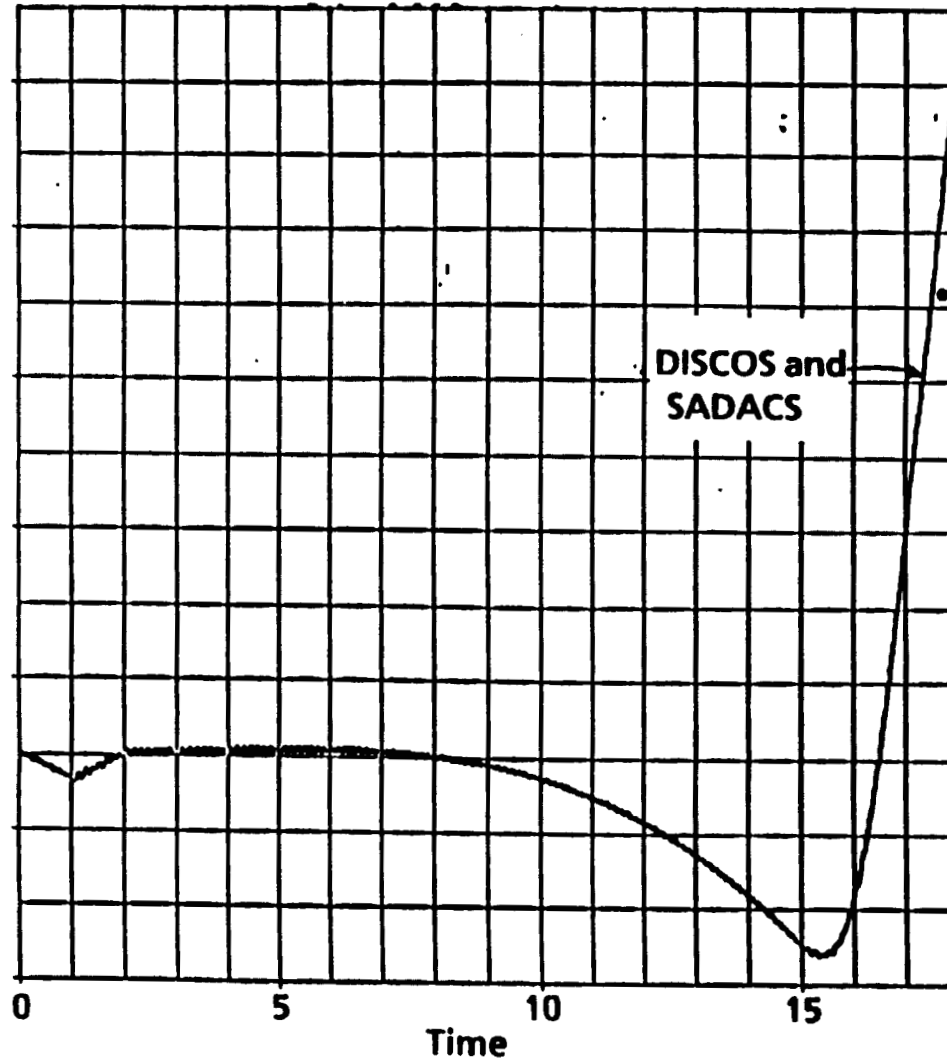
DISCOS-SADACS Comparison:

Main Body Sensor X Rotation Due To Appendage Command

468

022P1-RJ-137/New

4
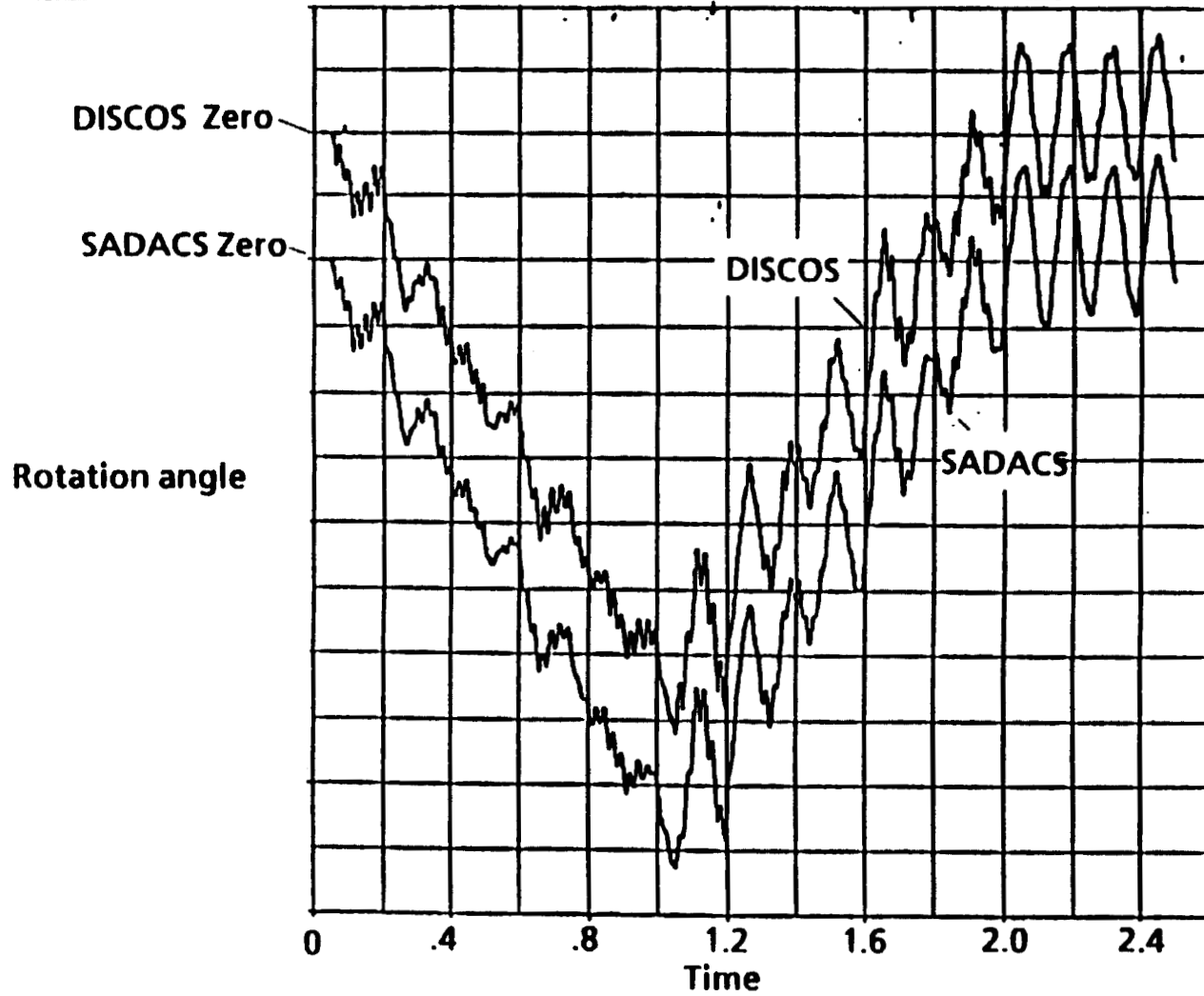
Controller Comparison - Closed Loop

The figure below shows the closed-loop response of a flexible model
of the SPAR robot manipulator with three different controllers. The
top plot is the first joint angle (waist) and the bottom plot is the
z motion of the end-effector (up and down). The three different
controllers are feedforward, semi-adaptive gain, and fixed gain.

Joint Angle

End-effector
Motion (Z)

470
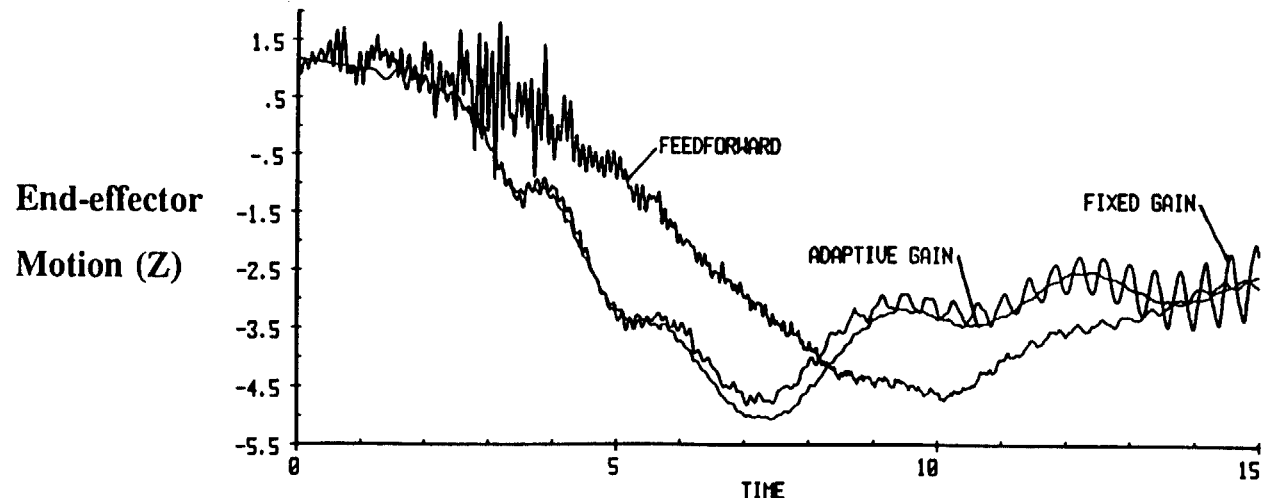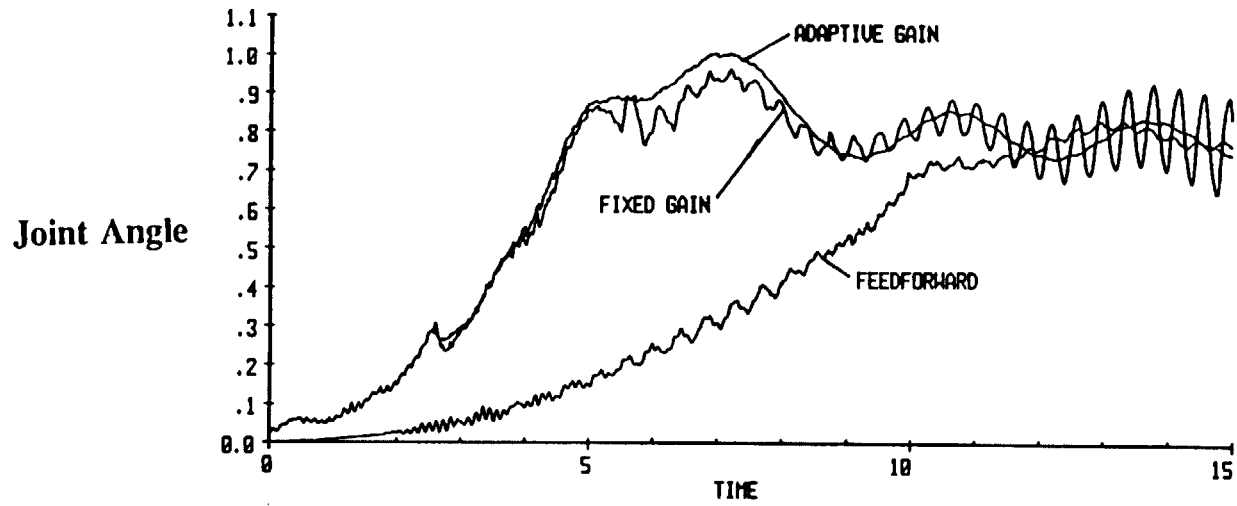
CONTROLLER COMPARISON - CLOSED LOOP

Applicability

- Large body of common problems

  - Non-spinners

  - Problems not dominated by nonlinear
    flexible response

- Each new problem should be validated against
  'full code' (TREETOPS/DISCOS)

Conclusions

- **SADACS fast, efficient multi-flexible-body simulation code**

- **Designed for use in controls environment**

- **New 'update' procedure improves accuracy, efficiency, works better**

- **Numerical example compared well with 'truth code' solution (DISCOS)**

472

SIMULATION OF SHUTTLE FLIGHT CONTROL SYSTEM STRUCTURAL INTERATION WITH RMS DEPLOYED PAYLOADS

By

Joe Turnball
C. S. Draper Laboratories
Cambridge, MA

ABSTRACT

In support of NASA/JSC, the C. S. Draper Laboratory (CSDL) has implemented a simulation of the system made up of the Orbiter, Remote Manipulator System (RMS), and payload grappled by the RMS. CSDL has used the simulation to study the stability of this overall system when its attitude is under control of the Orbiter's On-orbit Flight Control System (FCS). CSDL has also used the simulation to study the dynamics of the system when the RMS and its associated command software are in active control of the relative Orbiter to payload position and orientation.

The simulation models all of the following elements:

- RMS boom bending (represented by two cubic bending models)

- RMS boom Torsion

- RMS joint gearbox compliance (represented by a non-linear wind-up model)

- Flexibility at the RMS to Orbiter interface

- Flexibility at the RMS to payload interface

- Joint motor dynamics

- Joint servo-loop dynamics

- RMS on-board computer command logic

- Data transfer delays between the RMS sensor and the RMS on-board computer and between the RMS on-board computer and RMS joint servos

- On-orbit flight control nonlinear control logic

- Reaction Control System (both Primary and Vernier) jet forces and moments.

The Draper RMS Simulation (DRS) has close to a decade of development effort behind it. During that time, it has been used to analyze a wide range of RMS questions. Payload weights have run from zero (i.e., an unloaded arm) to weights in excess of the original design limit of the arm (65,000 lbs.). Types of interactions studied have ranged from interactions between failure detection algorithms in the RMS command software and high frequency motor transients to interactions between the On-orbit FCS and the

fundamental bending mode of the composite system with a 20,000 to 20,000 lb payload (.0.05 to 0.2 hz).

For all its complexity the DRS is reasonably economical. A run simulating one minute of real time costs on the order of $10 when run as a low priority over night batch job. Nevertheless, increases in economy can be of benefit for flight control/structural interaction studies which will involve increasing numbers of simulations with longer and longer simulation durations. Consequently, an effort has been under way for the last several years at CSDL on a so called Limited Singing and Dancing (LSAD) simulation that would sacrifice high frequency motor dynamics but retain good representation of bending modes pertinent to the interaction of the On-orbit FCS with the Orbiter/RMS/Payload structural system. LSAD shows approximately a ten-fold increase in economy as compared to similar DRS simulations

Simulation of Shuttle Flight Control System Interaction with RMS Deployed Payloads

A Presentation by Joseph Turnbull to the

WORKSHOP ON COMPUTATIONAL ASPECTS IN THE

CONTROL OF FLEXIBLE SYSTEMS

July 12–14, 1988

Orbiter / RMS / Payload
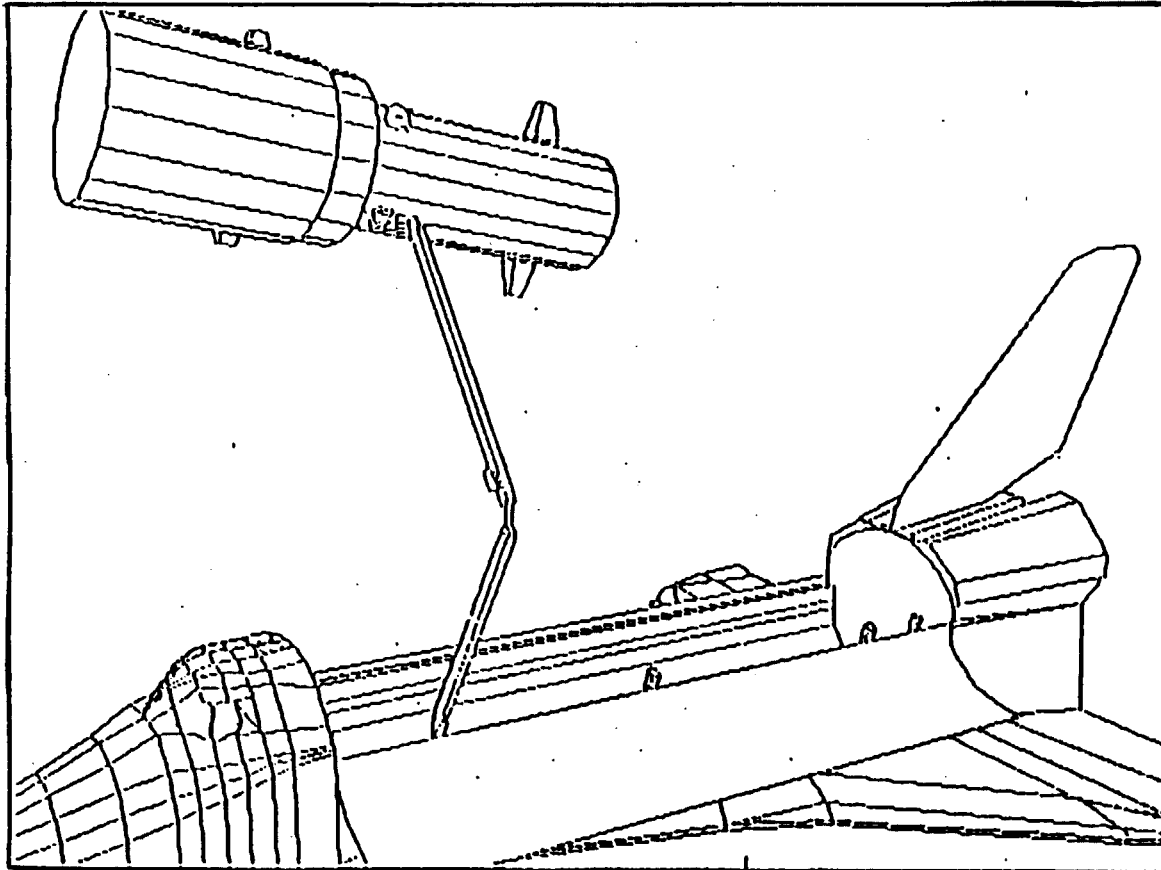
476

The Draper RMS Simulation (DRS)

Modeled in the DRS are:

- Transverse bending in the long booms

  Two cubic modes each in–plane and cross axis

- Torsion in the long booms

  Modeled as a torsional spring

- RMS to Orbiter stiffness

  Lumped with this are the flexibilities of the short links between the shoulder pitch joint and the Orbiter

- RMS to payload interface stiffness

  Lumped with this are the flexibilities of the short links between the wrist pitch joint and the payload

- Orbiter and payload as rigid bodies

JFT 7/12/88

DRS Model Elements (cont)

- RMS joint non-linear gearbox compliance

- Joint servo and motor dynamics

- Data transfer delays between the RMS sensors and the
  on-board computer and between the RMS on-board computer
  and the RMS joint servos

- RMS on-board computer command logic

- On-orbit flight control non-linear control logic

- Reaction Control System (both Primary and Vernier)
  jet forces and moments

478

Fundamental DRS Equation of Motion

$$\ddot{x} = A^{-1} [u - Kx]$$

where:

x is the state vector (dimension 25)

A is the "inertia" matrix

K is the "stiffness" matrix

u is the "torque" vector (a function, in part, of the servo loop)

Equation is integrated using a first order predictor/corrector scheme with a 1 ms integration step size.

479

480

DRAPER RMS SIMULATION
SERVO MODEL
APRIL1988

CURRENT SATURATION FLAG

DIRECT DRIVE CMDS (±3.385 V OR 0V)

JOINT TORQUE (TOJOV)

JOINT ANGLE (GAMMA)

joint stiction/friction

GR

GR

Delta

Σ

gearbox stiffness

gearbox efficiency

gearbox fwd/back logic

GEARBOX

Trim Integrator

KTR / s

LT

XCN1

$\dfrac{1}{\tau_1 s + 1}$ Low Pass

XCN2

RATE COMMAND

KDA

LD

DAC

Σ

analog demand

XT2

$\dfrac{K_2 \tau_2 s}{\tau_2 s + 1}$

L1

limit

$\dfrac{KI \tau_1 s}{\tau_1 s + 1}$

ANALOG TACH FILTER

K_A

Current Limiter

motor current

MDA limit +LA

-LA

Current limit level

Motor Current Limit Logic

Σ

V_B Back Emf

$\dfrac{1}{R_L + R_S}$

I_M

K_T

XMT

Σ

$\dfrac{1}{I_M}$

$\dfrac{1}{S}$

$\dfrac{1}{S}$

K_B

MOTOR & BRAKE

Motor Torque

TBR
TMS

motor/brake stiction/friction

DIGITAL TACH

BIAS

Σ

Tach Quantizer

KD

$\dfrac{1}{T2}[\Phi(aT2) - \Phi((a-1)T2)]$
Average rate over interval T2   T2 = 8.75 MS

Motor Fwd/Back Logic

CURRENT ATTENUATION COMMAND

BRAKE ON COMMAND

FORWARD/BACKWARD FLAG

RWM   4/14/88

DRS Verification

- Extensive simulation to flight comparison has been done.

- Model changes and parameter adjustments have been made to produce an excellent sim to flight agreement.

- Attention has been paid to both the low frequency bending and high frequency transients.

Simulation to Flight Overplot
Cross-axis Bending Excited by PRCS Jet Firing
8000 lb PFTA Grappled by RMS

482

JFT 7/12/88
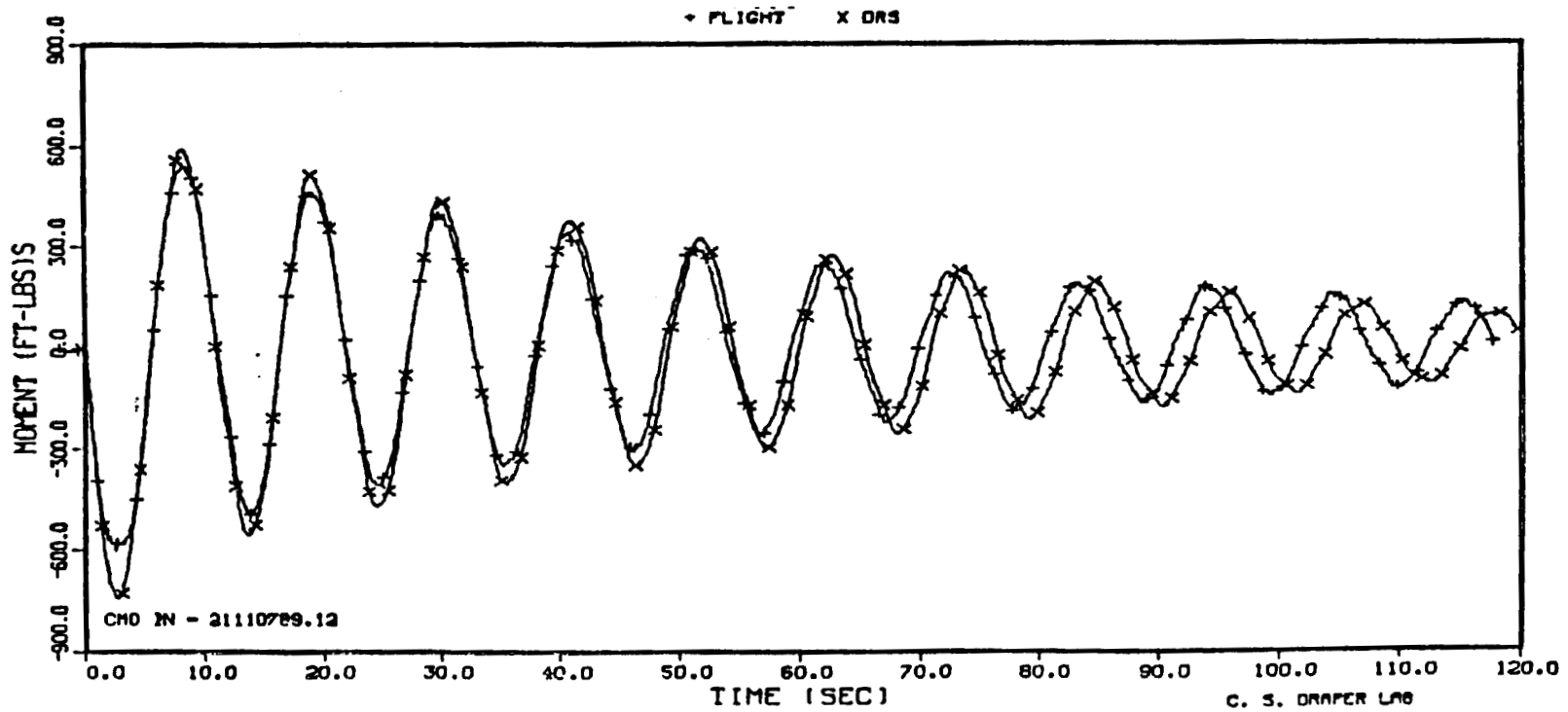
Simulation to Flight Overplot
In-plane Bending Excited by PRCS Jet Firing
8000 lb PFTA Grappled by RMS

+ FLIGHT   X ORS

CMD IN - 31110805.31

MOMENT (FT-LBS)S

TIME (SEC)

C. S. DRAPER LAB

JFT 7/12/88
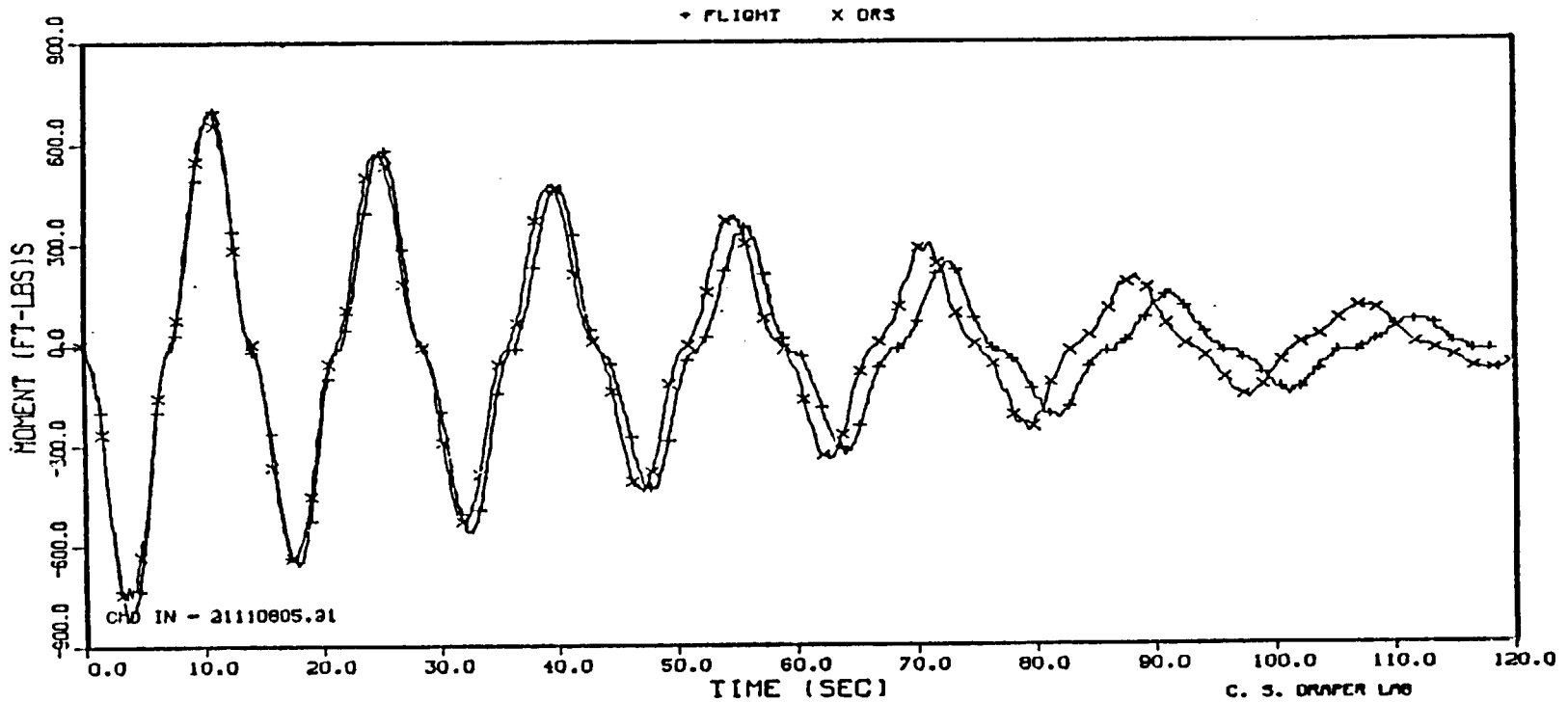
Simulation to Flight Overplot
Joint Angle & Motor Rate for
Shoulder Pitch Single Mode Drive
8000 lb PFTA Grappled by RMS

ENCODER ANGLE
SHOULDER PITCH
O ORS + FLIGHT

TACHOMETER RATE
SHOULDER PITCH

JFT 7/12/88
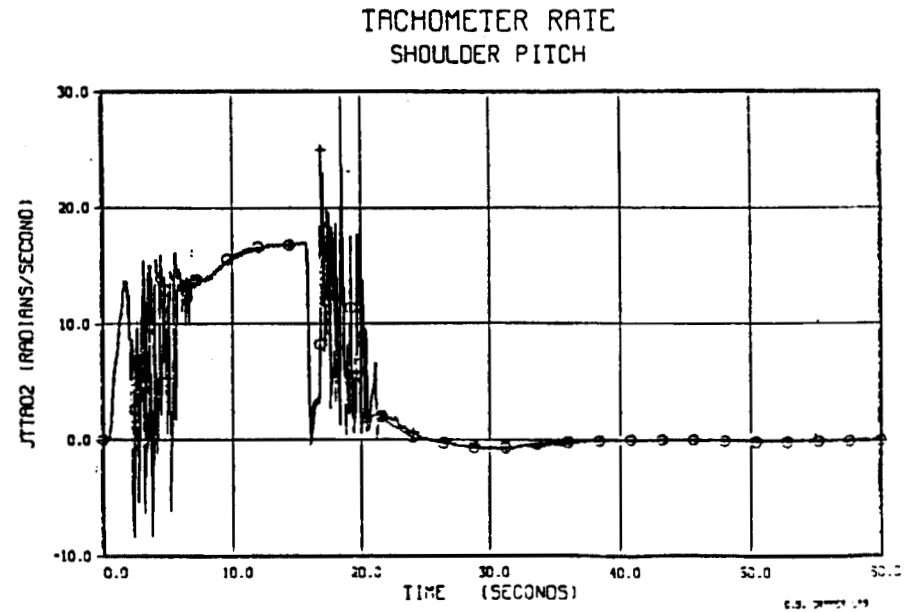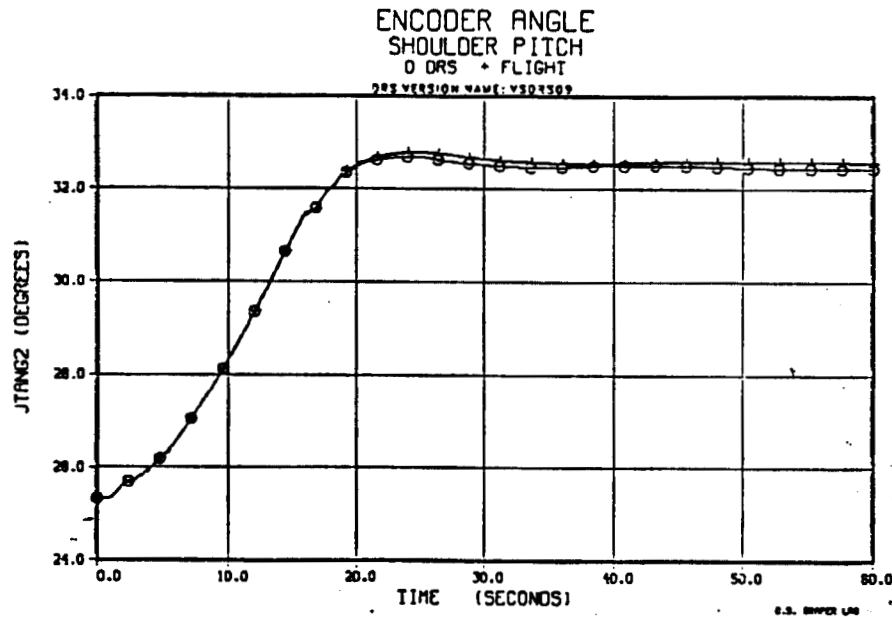
Simulation to Flight Overplot
Motor Rate Start and Stop Transients for
Shoulder Pitch Single Mode Drive
8000 lb PFTA Grappled by RMS

TACHOMETER RATE
SHOULDER PITCH
0 ORS • FLIGHT

TACHOMETER RATE
SHOULDER PITCH

485

JFT 7/12/88
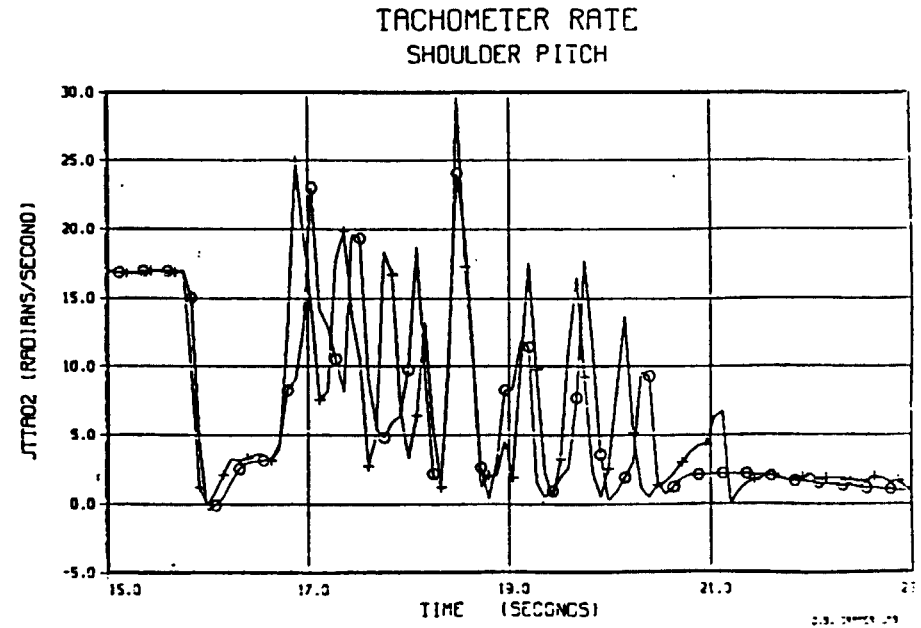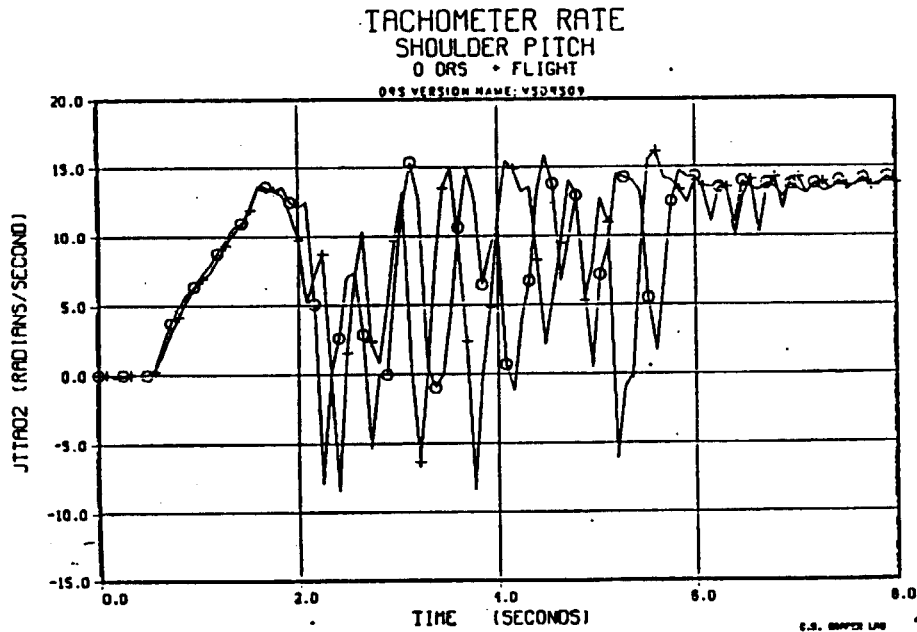
Simulation Applications of the DRS

- Arm dynamics and performance analyses

    - Arm motion and loads during arm maneuvers

    - Payload tip off rates at payload release

    - Payload and arm motion during capture

    - Interaxis coupling during six joint
      coordinated motion

    - Post-flight estimation of joint brake
      effectiveness

    - Evaluation of on-board algorithm for detecting
      and arresting joint runaway malfunctions

486

Simulation Applications of the DRS (cont)

- FCS interaction

  - Stability analyses — Self sustaining limit cycles are possible because of the relative values of FCS bandwidth, phase lag and fundamental bending frequency

  - Estimation of accelerations at the arm to payload interface due to PRCS jet activity

FCS Stability Analyses

- Stability is dependent on payload position and attitude relative to the Orbiter

- For a candidate position and attitude:

  - Apply open—loop PRCS pulses to excite the funamental flex modes of the system and then activate the FCS closed loop attitude hold, or

  - Simulate three axis attitude maneuvers,

  - Observe whether a self—sustained limit cycle results.

Example of a FCS Interaction Test with
No Self-sustained Limit Cycle

SJB24333   ESTIMATED, ACTUAL AND DESIRED ORBITER ATT. RATE
           IN DEG/SEC (solid=est,dash=true,dot=desired)
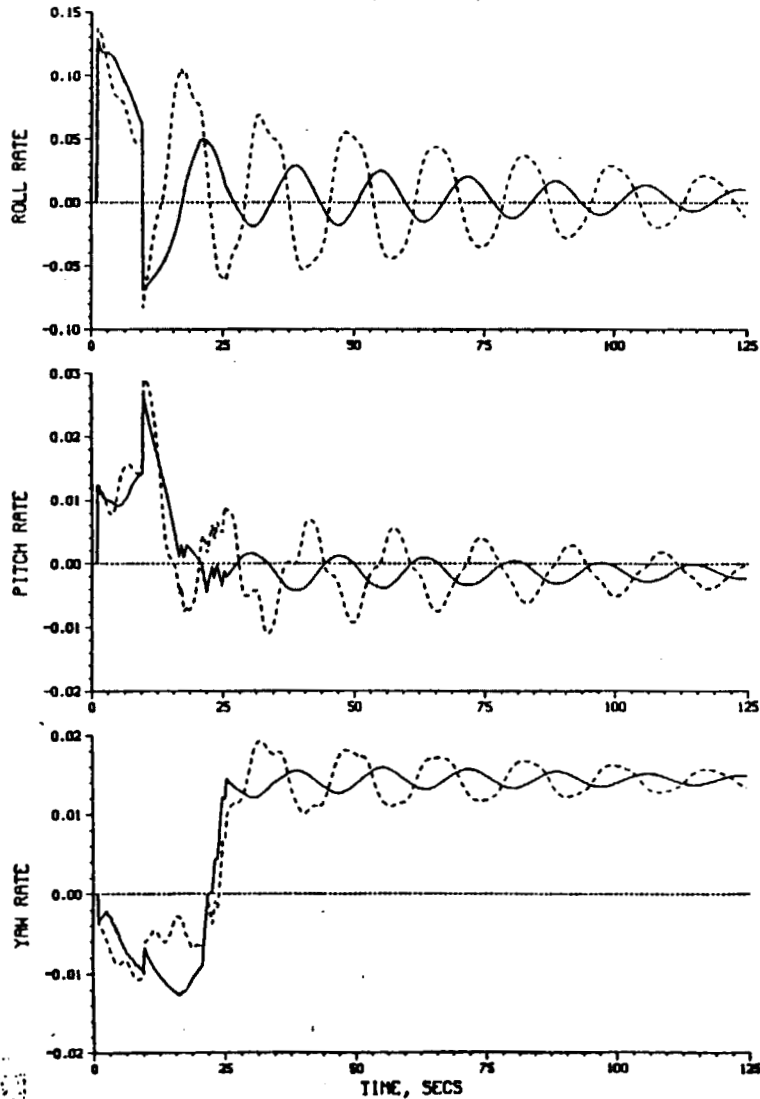
SJB24333   PHASE PLANE PLOTS
           DAP VARIABLES IN DEG AND DEG/SEC

489
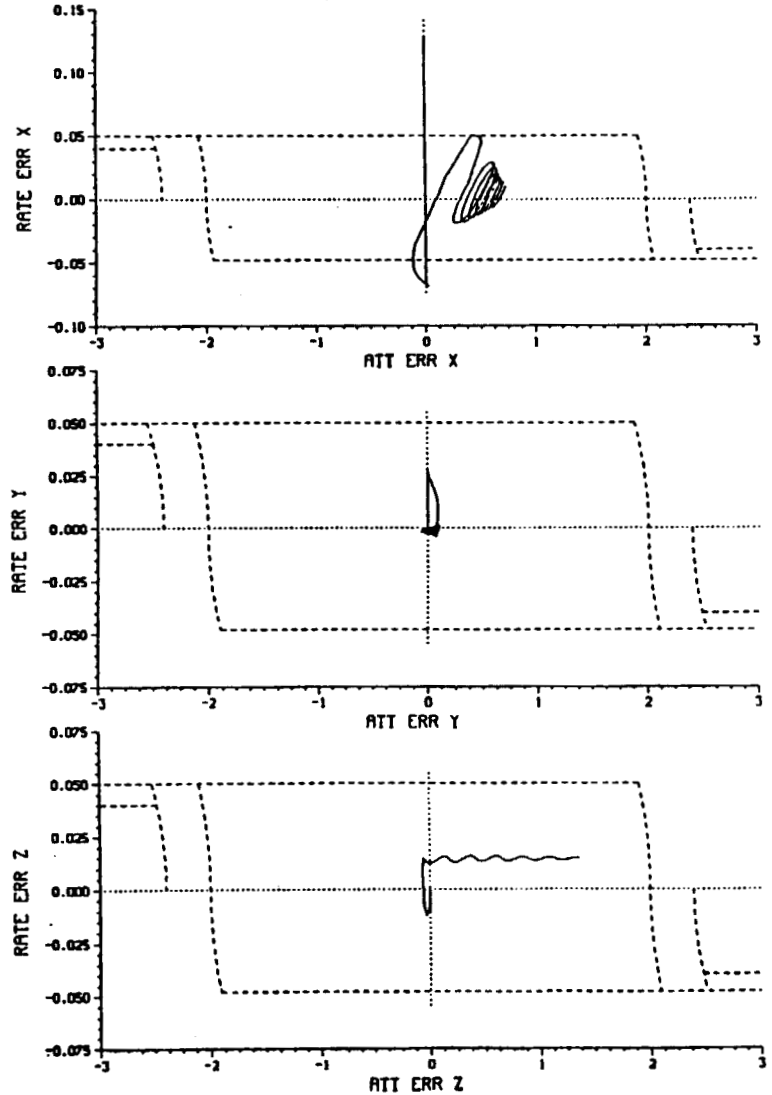
Example of a FCS Interaction Test with
A Self-sustained Limit Cycle

LOW50$30   ESTIMATED, ACTUAL AND DESIRED ORBITER ATT. RATE
           IN DEG/SEC   (solid=est,dash=true,dot=desired)
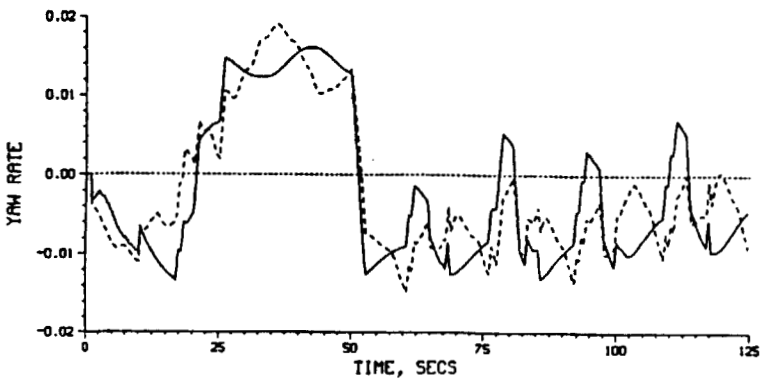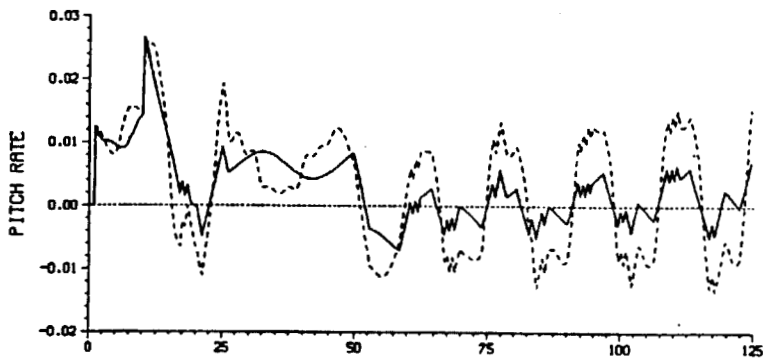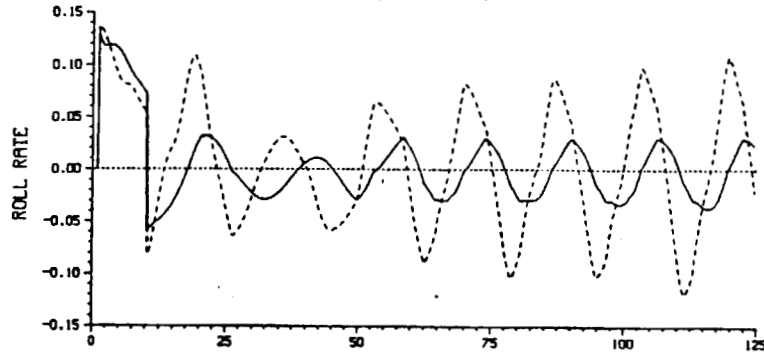
LOW50$30   PHASE PLANE PLOTS
           DAP VARIABLES IN DEG AND DEG/SEC

490
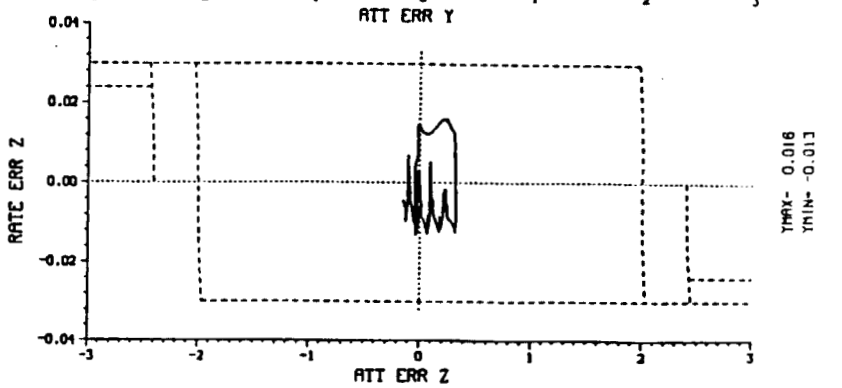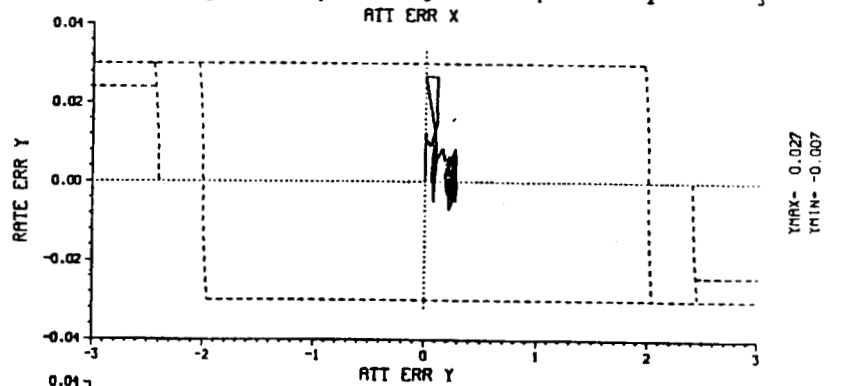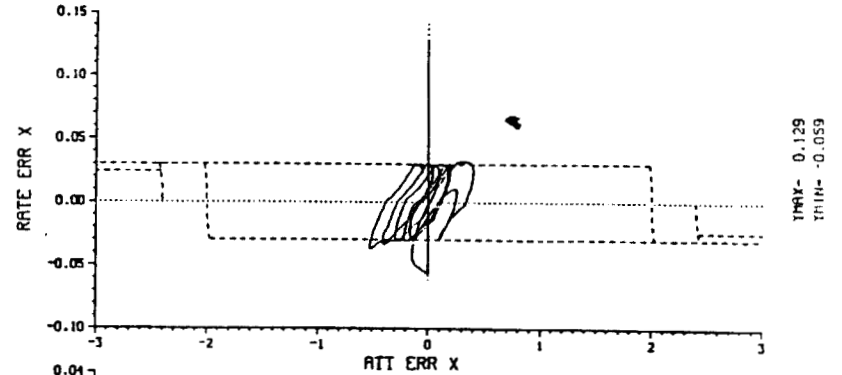
Cost of Running the DRS

- Simulation of one minute of FCS/RMS interaction costs about $15 when run at low priority over night.

- Normal priority for faster turn around is a factor of six more costly.

- For all its fidelity and capability, the DRS is reasonably efficient and economical.

- Nevertheless, the possible need to run hundreds of FCS interaction cases to map out stability regions was the motivation to develop an even more efficient simulation tailored to the needs of FCS interaction studies.

491

Draper's LSAD Simulation

- All flexibility between the orbiter and the payload is lumped into six relative degrees of freedom. The arm is assumed to be massless. LSAD state vector has dimension six.

- A simplified algorithm is used to model the response of the joint servos. This algorithm can operate at an 80 ms time step as opposed to the 1ms DRS time step.

- Fidelity in the low frequency modes has been retained.

- Features have been added:

    - Ability to submit sets of position and attitude variation cases in a single batch,

    - Ability to start a simulation with the arm in an excited state.

- Cost of an LSAD run is about a factor of 10 less than the cost of a DRS run.

DRS to LSAD Comparison of Orbiter Rates
During an FCS Stability Test

DRS STABILITY TEST
32K PL. B20501. BRAKES ON
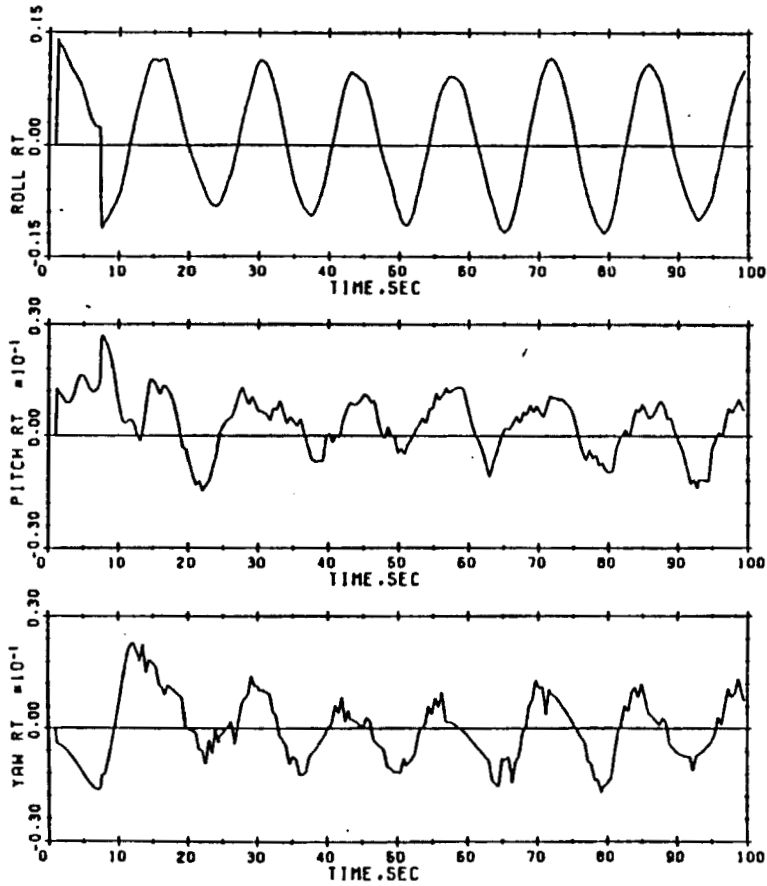
LSAD STABILITY TEST. RCS EXCITATION
32K PL.BRKS.B20501   DB= 2.000. RL= 0.010. ORB/JET/ARM PLOT
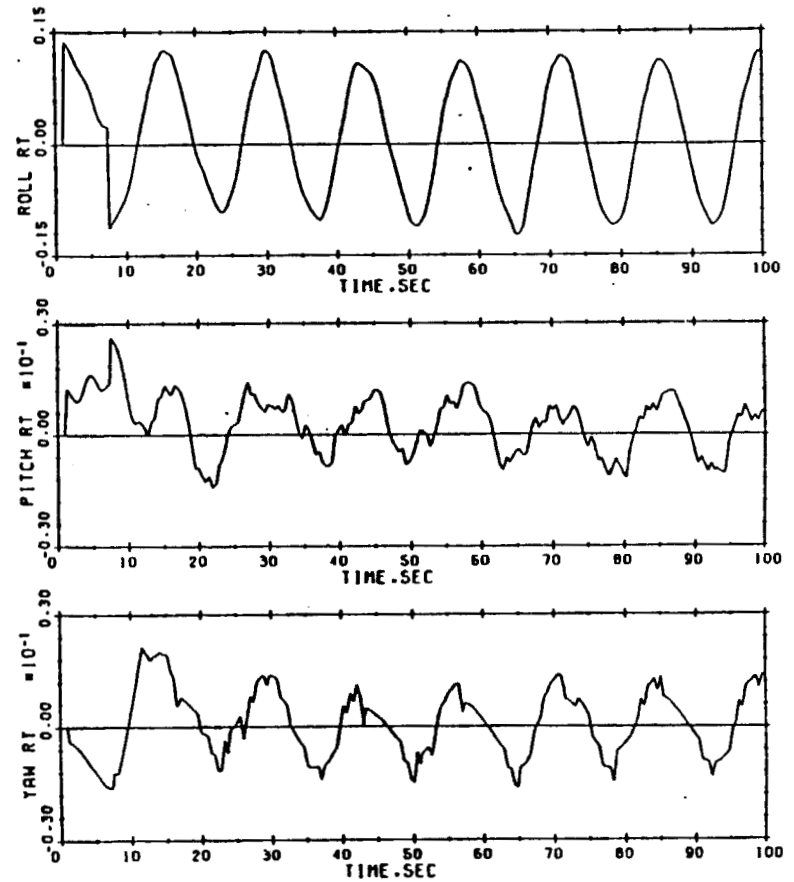
JFT 7/12/88

Draper Lab Contacts:

RMS general — Joe Turnbull (617) 258—2292

DRS — Rick Metzinger (617) 258—2912

LSAD — Tim Barrows (617) 258—2427

FCS Interaction — Darryl Sargent (617) 258—2296

494

A PERFORMANCE COMPARISON OF INTEGRATION ALGORITHMS IN SIMULATING FLEXIBLE STRUCTURES

By

R. M. Howe
The University of Michigan
Applied Dynamics International
Ann Arbor, Michigan

ABSTRACT

Modeling of the dynamic vibration modes of a flexible structure can be achieved either by using a generalized coordinate for each mode considered in the simulation, or by discretizing the structure into a sufficiently large number of segments to provide the necessary modal accuracy. The accuracy and stability considerations in choosing appropriate numerical integration algorithms are different, depending on which modeling approach is utilized. In the generalized coordinate approach the frequency and shape of each mode is assumed to be known. The integration method should provide an accurate match to the modal frequency and damping, and should also exhibit sinusoidal transfer function errors which are acceptably small, especially for frequencies in the vicinity of the modal resonance. Since only those modes considered necessary for the required simulation fidelity are included as generalized coordinates, integrator stability for modes of higher frequency does not become in an issue.

On the other hand, when the discretized structure approach is used, high frequency modes not of interest to the simulation will nevertheless be present. In this case it is important that the integration method not only provide satisfactory characteristic root and transfer function accuracy for the lower modes of interest, but also provide stable solutions with satisfactory damping for the higher modes which are not of interest.

In this paper asymptotic formulas for the characteristic root errors as well as transfer function gain and phase errors are presented for a number of traditional integration methods and for several new integration methods. Normalized stability regions in the λh plane are compared for the various methods. In particular, it is shown that a modified form of Euler integration with root matching is an especially efficient method for simulating lightly-damped structural modes. The method has been used successfully for structural bending modes in the real-time simulation of missiles. Performance of this algorithm is compared with other special algorithms, including the state-transition method. A predictor-corrector version of the modified Euler algorithm permits it to be extended to the simulation of nonlinear models of the type likely to be obtained when using the discretized structure approach.

Performance of the different integration methods is also compared for integration step sizes larger than those for which the asymptotic formulas are valid. It is concluded that many traditional integration methods, such as RD-4, are not competitive in the simulation of lightly damped structures.

A Performance Comparison of Integration Algorithms in Simulating Flexible Structures

R. M. Howe

The University of Michigan, Ann Arbor, Michigan

and

Applied Dynamics International, Ann Arbor, Michigan

ABSTRACT

In this paper a number of integration algorithms, including several new methods, are considered for the simulation of flexible structures. The effectiveness of the different algorithms is assessed by considering the characteristic root errors which they produce, the sinusoidal transfer function gain and phase errors, the stability regions, and the execution times. The suitability of the various algorithms for simulations with real-time inputs is also noted. When the structural modes in a simulation are represented by generalized (normal) coordinates, the selection criteria for integration methods are somewhat different than the criteria when the structure is discretized into a sufficiently large number of segments to provide the necessary modal accuracy. In this paper asymptotic formulas for the characteristic root errors as well as transfer function gain and phase errors are presented for a number of traditional integration methods and for several new integration methods. Normalized stability regions in the λh plane are compared for the various methods. In particular, it is shown that a modified form of Euler integration with root matching is an especially efficient method for simulating structural modes. The method has been used successfully for structural bending modes in the real-time simulation of missiles. A predictor version of the modified Euler algorithm permits it to be extended to the simulation of nonlinear models of the type likely to be obtained when using the discretized structure approach.

1. Introduction

Modeling of the dynamic vibration modes of a flexible structure can be achieved either by using a generalized coordinate for each mode considered in the simulation, or by discretizing the structure into a sufficiently large number of segments to provide the necessary modal accuracy. In this latter case the mathematical model for a flexible structure with N degrees of freedom has the following general form:

$$M(q)\ddot{q} + C(q,\dot{q}) + K(q) = F(t) \tag{1}$$

where q is an N-component position state vector, $M(q)$ is the mass matrix, $C(q,\dot{q})$ is the coriolis and centrifugal acceleration vector, $K(q)$ is the elastic and gravity force vector, and $F(t)$ is the external force vector. When the vibration modes of the structure are represented by normal

(generalized) coordinates, a coordinate x representing the time-varying amplitude of a given mode with undamped natural frequency ω_n and damping ratio ζ obeys the equation

$$\ddot{x} + 2\zeta\omega_n\dot{x} + \omega_n^2 x = \omega_n^2 \phi(t) \tag{2}$$

Here $\phi(t)$ is the generalized force associated with the coordinate x. When a number of modes are present, there will in general also be terms in Eq. (2) which couple the mode of amplitude x with other structural modes.

The accuracy and stability considerations in choosing appropriate numerical integration algorithms for solving differential equations of the type shown in (1) or (2) will be different. In the generalized coordinate approach of Eq. (2) the frequency and shape of each mode is assumed to be known. The integration method should provide an accurate match to the modal frequency and damping, and should also exhibit sinusoidal transfer function errors which are acceptably small, especially for frequencies in the vicinity of the modal resonance. Since only those modes considered necessary for the required simulation fidelity are included as generalized coordinates, integrator stability for higher frequency modes which are not of interest does not become an issue.

On the other hand, when the discretized structure approach represented by Eq. (1) is used, high frequency modes which are unimportant in the simulation will nevertheless be present. In this case it is important that the integration method not only provide satisfactory characteristic root and transfer function accuracy for the lower modes of interest, but also provide stable solutions with satisfactory damping for the higher modes which are not of interest.

In this paper asymptotic formulas for the characteristic root errors as well as transfer function gain and phase errors are presented for a number of traditional integration methods and for several new integration methods. Normalized stability regions in the λh plane are compared for the various methods, where λ is an eigenvalue asociated with the linearized perturbation equations of the structure and h is the integration step size. . In particular, it is shown that a modified form of Euler integration with root matching is an especially efficient method for simulating lightly-damped structural modes. The method has been used successfully for structural bending modes in the real-time simulation of missiles. Predictor versions of the modified Euler algorithm permit it to be extended to the simulation of nonlinear models of the type likely to be obtained when structures are represented by means of discretization. The stability regions in the λh plane for the modified Euler methods are especially well suited to the requirements when using the discretized structure approach.

2. Dynamic Error Measures for Integration Algorithms

In comparing different integration methods for the simulation of flexible structures it is important to utilize meaningful performance measures which permit general conclusions to be drawn regarding the expected dynamic errors associated with each method. Our dynamic error analysis will be based on linearized perturbation equations derived from the original nonlinear differential equations used to model the structure. Thus we will assume that the system

eigenvalues are known, as well as the transfer functions relating specific input-output pairs. We will further assume that the simulation uses a fixed integration step size h. This is necessary in the case of a real-time simulation. It is likely to be true over a large number of steps even when a variable-step integration method is used in simulating a flexible structure. For linearized equations and a fixed integration step size we can apply the method of z transforms to anayze the dynamic errors resulting from specific integration algorithms [1,2]. There are two error measures which quite useful in predicting overall dynamic accuracy in the simulation. The first is the fractional error in each characteristic root (eigenvalue) of the digital simulation, defined as

$$\text{Fractional error in characteristic root} = e_\lambda = \frac{\lambda^* - \lambda}{\lambda} \tag{3}$$

where λ is the characteristic root of the continuous system being simulated and λ^* is the equivalent characteristic root for the digital simulation. For the case of complex roots (of which there will be many conjugate pairs in the simulation of a flexible structure) it is more appropriate to determine the fractional error, e_ω, in root frequency and the damping ratio error, e_ζ. Thus we define

$$e_\omega = \frac{\omega_d^* - \omega_d}{\omega_d} \quad , \qquad e_\zeta = \zeta^* - \zeta \tag{4}$$

Here ω_d^* and ω_d represent the frequencies of the digital and continuous system roots, respectively, while ζ^* and ζ represent the damping ratios for the digital and continuous system roots, respectively.

The second dynamic error measure of significance is the fractional error in digital system transfer function for sinusoidal inputs of frequency ω. For any input-output pair let $H(s)$ be the transfer function of the continuous system and $H^*(z)$ be the z transform of the digital system that results when a particular integration algorithm is used. Then the fractional error in sinusoidal transfer function is given by [3]

$$\frac{H^*(e^{j\omega h})}{H(j\omega)} - 1 = e_M + je_A \tag{5}$$

For simulations of any reasonable accuracy the magnitude of this fractional error will be small compared with unity, in which case it is easily shown that the real part, e_M, is equal approximately to the fractional error in gain and the imaginary part, e_A, is equal to the phase error of the transfer function [3].

For any numerical integration algorithm the integrator transfer function for sinusoidal inputs of frequency ω can be written approximately as [3]

$$H_I^*(e^{j\omega h}) \cong \frac{1}{j\omega h\left[1 + e_I(j\omega h)^k\right]} \quad , \quad \omega h \ll 1 \tag{6}$$

499

where h is the integration step size. Since $1/(j\omega h)$ is the ideal integrator transfer function, it is apparent that the term $e_I(j\omega h)^k$ represents the integrator error. For Adams-Bashforth predictor and Adams-Moulton two-pass predictor-corrector algorithms of order 2, 3, and 4, integration methods that are candidates for simulation of flexible structures, the error coefficient e_I and algorithm order k are listed in Table 1.

Table 1. Integrator Transfer Function Error Parameters for AB Predictor and AM Predictor Corrector Algorithms

Method	Error Coefficient, e_I	Algorithm Order, k
AB-2	$\dfrac{5}{12}$	2
AB-3	$\dfrac{3}{8}$	3
AB-4	$\dfrac{251}{720}$	4
AM-2	$\dfrac{1}{12}$	2
AM-3	$\dfrac{1}{24}$	3
AM-4	$\dfrac{19}{720}$	4

In terms of e_I and k the following formula for e_λ, the fractional error in characteristic root as defined earlier in Eq. (3), can be derived [3]:

$$e_\lambda = \frac{\lambda^* - \lambda}{\lambda} \cong -e_I(\lambda h)^k \quad , \qquad |\lambda h| << 1 \tag{7}$$

It is apparent that e_λ is directly proportional to the integrator error coefficient, e_I. For complex characteristic roots equivalent asymptotic formulas for the root frequency and damping errors, e_ω and e_ζ, as defined in Eq. (4), can be derived [3]. As in Eq. (7), the errors are proportional to $e_I |\lambda h|^k$.

For digital simulation of a first order system with transfer function $H(s) = 1/(s-\lambda)$ the fractional error in the transfer function for sinusoidal inputs, as defined in Eq. (5), can also be derived in terms of the integrator error parameters e_I and k [3]. From this result the following asymptotic formulas are obtained for e_M, the fractional error in transfer function gain, and e_A, the transfer function phase error:

For k odd, $e_M \cong (-1)^{\frac{k+1}{2}} \dfrac{\omega \lambda e_I}{\omega^2 + \lambda^2} (\omega h)^k$, $e_A \cong (-1)^{\frac{k+1}{2}} \dfrac{\omega^2 e_I}{\omega^2 + \lambda^2} (\omega h)^k$, $\omega h \ll 1$ $\qquad$ (8)

For k even, $e_M \cong -(-1)^{\frac{k}{2}} \dfrac{\omega^2 e_I}{\omega^2 + \lambda^2} (\omega h)^k$, $e_A \cong (-1)^{\frac{k}{2}} \dfrac{\omega \lambda e_I}{\omega^2 + \lambda^2} (\omega h)^k$, $\omega h \ll 1$ $\qquad$ (9)

Here the errors are proportional to $e_I (\omega h)^k$. Comparable asymptotic formulas for e_M and e_A can be derived for digital simulation of a second-order system with transfer function $H(s) = 1/(s^2 + 2\zeta\omega_n s + \omega_n^2)$ [3]. Again, the gain and phase errors are proportional to $e_I(\omega h)^k$.

The transfer function $H(s)$ for any order linear system with both real and complex roots can be represented as the product of first and second-order transfer functions. In this case it can be shown that the asymptotic formulas for the digital system transfer function gain and phase errors is simply the sum of the individual first and second-order subsystem gain and phase errors, respectively, for predictor and predictor-corrector methods of the type shown in Table 1. If we simulate a flexible structure with a given integration method, this permits us to compute the linearized system gain and phase errors at the frequency ω for any input-output pair as a function of integration step size h. In view of the reemerging popularity of frequency-domain methods for designing multiple input/multiple output control systems, this is a quite useful result. It permits us to estimate ahead of time for a given step size and integration method whether the simulation errors will be satisfactorily small. Conversely, for a given transfer function accuracy requirement, it allows us to compute the maximum allowable step size h for the simulation.

It should be noted that the methodology outlined above for determining characteristic root and transfer function errors for any order of linearized system from the simple integrator model given by Eq. (6) does not work in the case of multiple-pass, single step methods such as Runge-Kutta. This is because the results of individual passes within a single step in such methods depend on the particular form of the system transfer function. Asymptotic formulas for the root error parameters e_λ, e_ω, and e_ζ can, of course, be derived separately for RK-2, RK-3, RK-4, and variations of these methods [3].

3. Modified Euler Integration Algorithms

In this section we describe some modifications of simple Euler integration which have potential advantages over conventional integration methods such as those listed in Table 1. First we introduce the concept of state variables defined at both integer and half integer sample times. Assume that the simulation of a mechanical degree of freedom with position state x, velocity state y, and acceleration a involves integrating the following simple state equations:

$$\dot{y} = a, \quad \dot{x} = y \qquad (10)$$

Next assume that successive data points are defined at integer time samples in representing the acceleration a and position x, and at half-integer sample times in representing the velocity y. The following modified Euler algorithms can then be used for integration:

$$y_{n+1/2} = y_{n-1/2} + ha_n \ , \quad x_{n+1} = x_n + hy_{n+1/2} \tag{11}$$

The basic concept behind this modification of standard Euler integration is very simple; instead of the using the state variable derivative defined at the beginning of the integration step, the method uses a state variable derivative defined halfway through the step. For this algorithm it is easy to show that the integrator error coefficient defined in Eq. (6) is given by $e_I = 1/24$ and the order of the method is $k = 2$. Thus the accuracy of this single-pass algorithm is twice that of the two-pass AM-2 predictor-corrector. However, the algorithm does require that the velocity states be defined at half-integer sample times.

Let us apply this modified Euler method to the second-order system represented by Eq. (2) for the generalized coordinate x. We can replace Eq. (2) by the following two state equations:

$$\dot{y} = \omega_n[\phi(t) - x - 2\zeta y] \ , \quad \dot{x} = \omega_n y \tag{12}$$

By analogy with Eq. (11) the modified Euler difference equations become:

$$y_{n+1/2} = y_{n-1/2} + \omega_n h(\phi_n - x_n - 2\zeta y_n') \ , \quad x_{n+1} = x_n + \omega_n h y_{n+1/2} \tag{13}$$

Since y_n is not explicitly computed, it is necessary to substitute an estimate y_n' in the damping term on the right side of the $y_{n+1/2}$ equation. There are many ways in which the y_n' estimate can be computed. In Table 2 we list four candidate methods.

Table 2. Methods for Estimating the Velocity y_n in Modified Euler Integration

Method	Formula for the Estimate, y_n'
1. Average of $y_{n+1/2}$ and $y_{n-1/2}$	$y_n' = \dfrac{y_{n+1/2} + y_{n-1/2}}{2}$
2. Extrapolation using $y_{n-1/2}$ and $y_{n-3/2}$	$y_n' = \dfrac{3}{2}y_{n-1/2} - \dfrac{1}{2}y_{n-3/2}$
3. Integration using $\dot{y}_{n-1}$ and $\dot{y}_{n-2}$	$y_n' = y_{n-1/2} + \dfrac{7}{8}\dot{y}_{n-1} - \dfrac{3}{8}y_{n-2}$
4. Estimate based on $y_{n-1/2}$	$y_n' = y_{n-1/2}$

The estimate for y_n in the first method is simply based on averaging $y_{n+1/2}$ and $y_{n-1/2}$. This is equivalent to utilizing trapezoidal integration for the damping term. Although this means that $y_{n+1/2}$ now appears on both sides of the difference equation in (13), for the linear system considered here it is possible to solve explicitly for $y_{n+1/2}$, as we will see in the next section. In the second method the estimate for y_n is based on a linear extrapolation from $y_{n-1/2}$ and $y_{n-3/2}$. This is equivalent to using trapezoidal integration for the damping term. Since $y_{n+1/2}$ now appears only on the left side of the difference equation in (13), this method can be used in the simulation of equations where dy/dt is a nonlinear function of y. This is also true for the third and fourth methods. The third is based on a second-order predictor integration over the interval $h/2$, starting with $y_{n-1/2}$ and using dy/dt at the $n-1$ and $n-2$ intervals. This is equivalent to estimating y_n with quadratic extrapolation based on $y_{n-1/2}$, $y_{n-3/2}$ and $y_{n-5/2}$. In the fourth method we simply use $y_{n-1/2}$ as our estimate for y_n. This is equivalent to Euler integration for the damping term.

4. Modified Euler Integration with Trapezoidal Damping

We have seen in Table 2 that the velocity estimate y'_n for the modified Euler difference equations in (13) can be based on the average of $y_{n+1/2}$ and $y_{n-1/2}$. Thus

$$y'_n = \frac{y_{n+1/2} + y_{n-1/2}}{2} \tag{14}$$

As noted earlier, this is equivalent to utilizing trapezoidal integration for the damping term. Although this means that $y_{n+1/2}$ now appears on both sides of the difference equation, for the linear system considered here it is possible to solve explicitly for $y_{n+1/2}$. In this way we obtain the following equations:

$$y_{n+1/2} = C_1 y_{n-1/2} + C_2(\phi_n - x_n), \quad x_{n+1} = x_n + \omega_n h y_{n-1/2} \tag{15}$$

where

$$C_1 = \frac{1 - \zeta \omega_n h}{1 + \zeta \omega_n h}, \quad C_2 = \frac{\omega_n h}{1 + \zeta \omega_n h} \tag{16}$$

From the method of z transforms applied to Eqs. (15) and (16) we obtain the following asymptotic formulas for the frequency and damping ratio errors of the digital simulation [4]:

$$e_\omega = \frac{\omega_d^* - \omega_d}{\omega_d} \cong \frac{1 + 4\zeta^2 - 8\zeta^4}{24(1 - \zeta^2)} \omega_n^2 h^2, \quad e_\zeta = \zeta^* - \zeta \cong \frac{\zeta}{24}(4\zeta^2 - 1), \quad \omega_n h \ll 1 \tag{17}$$

The transfer function gain and phase errors are given approximately by

$$\text{Fractional gain error} = \frac{|H^*|}{|H|} - 1 = e_M \cong \frac{-\dfrac{\zeta\omega}{\omega_n}\left[1 - \dfrac{\omega^2}{\omega_n^2}\right]}{\left[1 - \dfrac{\omega^2}{\omega_n^2}\right]^2 + \left[\dfrac{2\zeta\omega}{\omega_n}\right]^2}(\omega h), \quad \omega h \ll 1 \tag{18}$$

$$\text{Phase error} = e_A \cong \frac{\dfrac{2\zeta^2\omega^2}{\omega_n^2}}{\left[1 - \dfrac{\omega^2}{\omega_n^2}\right]^2 + \left[\dfrac{2\zeta\omega}{\omega_n}\right]^2}(\omega h), \quad \omega h \ll 1 \tag{19}$$

The characteristic root errors in Eq. (17) and the transfer function gain and phase errors in Eqs. (18) and (19) are comparable with those for AM-2 integration for the same step size h [3]. Yet AM-2 is a two-pass method whereas the modified Euler with trapezoidal damping, as used here, is a single-pass method. Thus it will take only half as long to execute as AM-2 while producing comparable accuracy. Its accuracy is approximately 5 times better than the accuracy of AB-2 integration when applied to the same second-order system.

The accuracy of modified Euler integration when applied to a linear second-order system can be further improved by the technique of root matching, which was originally employed by Fowler to improve the performance of conventional Euler integration [5]. By taking the z transform of Eqs. (15) and (16) we can obtain exact analytic formulas for the undamped natural frequency ω_n^* and damping ratio ζ^* in terms of ω_n and ζ. From these formulas we can solve for ω_n and ζ in terms of ω_n^* and ζ^*. If in these formulas we then replace ω_n and ζ by ω_n' and ζ', respectively, and ω_n^* and ζ^* by ω_n and ζ, respectively, we obtain the following [4]:

$$\omega_n' = \frac{1}{h}\sqrt{2 - \frac{2\cos(\omega_n h\sqrt{1 - \zeta^2})}{\cosh(\zeta\omega_n h)}} \tag{20}$$

$$\zeta' = \frac{\tanh(\zeta\omega_n h)}{\omega_n' h} \tag{21}$$

If ω_n' and ζ' from these formulas are used instead of ω_n and ζ in Eqs. (15) and (16), the resulting digital simulation will exhibit ω_n^* and ζ^* values which exactly match the ω_n and ζ of the continuous system being simulated. For a given step size h the ω_n', ζ', C_1 and C_2 can be precomputed, so that each integration step in simulating the second-order system only requires 3 multiplies and 2 adds, as before. Now the charcteristic roots of the digital simulation will be exactly equal to those of the continuous system, regardless of the integration step size h. The approximate formulas for the transfer function gain and phase errors are given by [4]:

$$e_M \cong \frac{1}{24}(\omega h)^2 \ , \quad e_A \cong -\frac{\zeta \omega_n}{6\omega}(\omega_n h)^2 \ , \quad \omega h \ll 1 \tag{22}$$

Note that the fractional error in gain, e_M, is completely independent of the damping ratio ζ, and the phase error e_A approaches zero as ζ approaches zero. Thus our modified Euler algorithm with root matching will be especially effective in simulating lightly-damped second-order systems, as will be the case in structural modes. This is illustrated in Figure 1, where gain and phase versus frequency for a second-order system with $\zeta = 0.01$ are plotted. Because of the sharp resonant peak in gain and the extremely rapid change in phase as ω passes through ω_n, it is very critical that both the natural frequency and damping ratio of the digital simulation match that of the continuous system. The table at the bottom of the figure shows the transfer function errors for input frequencies in the vicinity of ω_n for the specific case of $\omega_n h = 0.5$, which corresponds to only 2 integration steps per radian or 12.57 steps per cycle. Shown in the table are the gain and phase errors based on both an exact calculation from the system z transform, $H^*(e^{j\omega h})$, as well as the approximate formulas of Eq. (31). Note how closely the approximate caculations agree with the exact, even for the example here for which $\omega h \approx 0.5$.

Until now we have only analyzed the dynamic performance of the modified Euler method in the frequency domain. This has been accomplished by examining the gain and phase errors of the transfer function for sinusoidal inputs. We now consider the errors in computed response of the second-order system to a unit-step input. Figure 2 shows the errors which result when using RK-2 integration (Heun's method); modified Euler with trapezoidal integration for the damping term, i.e., Eqs. (15) and (16); and modified Euler with root matching, i.e., ω_n' and ζ' from Eqs. (20) and (21) substituted for ω_n and ζ in Eqs. (15) and (16). For the example in the figure the damping ratio $\zeta = 0.707$ and the integration step size is given by $\omega_n h = 0.5$. The results show that the RK-2 errors are 4 to 10 times larger than the modified Euler errors. It should also be noted that RK-2 is a two-pass method, that is, it requires two evaluations of the state-variable derivatives per integration step. It follows that RK-2 will take approximately twice as long to execute per integration step as the single-pass modified Euler methods. To provide the same output integration frame rate in real time the RK-2 method will therefore require twice the mathematical step size h in comparison with the modified Euler methods considered here. This will further increase by a factor of 4 the RK-2 errors relative to the modified Euler errors in Figure 2.

The modified Euler results shown in Figure 2 were obtained using an initial step of $h/2$ in integrating dy/dt to obtain y. After one integration step this provides the calculation of $y_{1/2}$ starting with the initial condition y_0. The step size is taken as h for all subsequent dy/dt integration steps. This results in successive velocity values representing y at half-integer step times, consistant with the concept introduced in the beginning of this section.
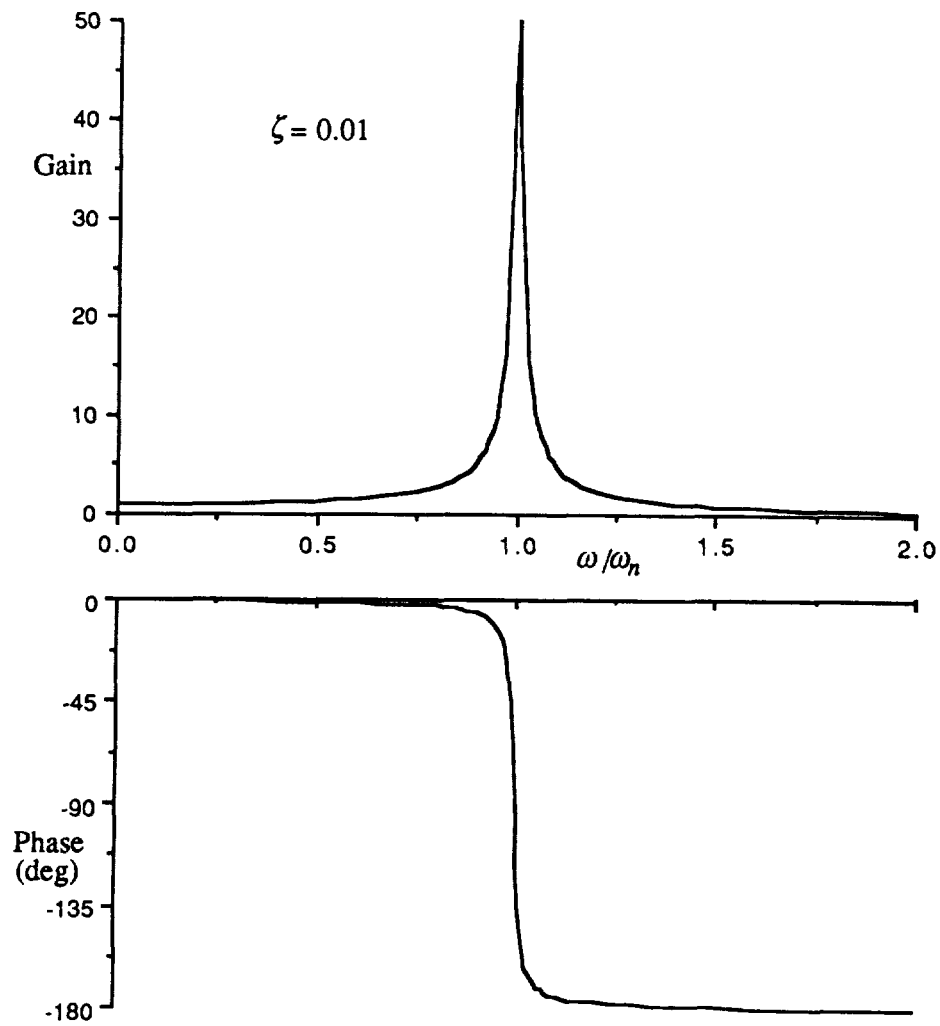
	ω/ω_n	Fractional Gain Error		Phase Error (radians)	
		Exact	Eq. (31)	Exact	Eq. (31)
$\omega_n h = 0.5$	0.7	0.01040	0.01021	-0.000296	-0.000292
	0.9	0.01727	0.01888	-0.000381	-0.000375
12.57 steps	1.0	0.02137	0.02083	-0.000424	-0.000317
per cycle)	1.1	0.02592	0.02521	-0.000467	-0.000458
	1.4	0.04240	0.04083	-0.000595	-0.000583

Figure 1. Frequency response of lightly-damped second-order system using modified Euler integration with root matching, $\omega_n h = 0.5$.
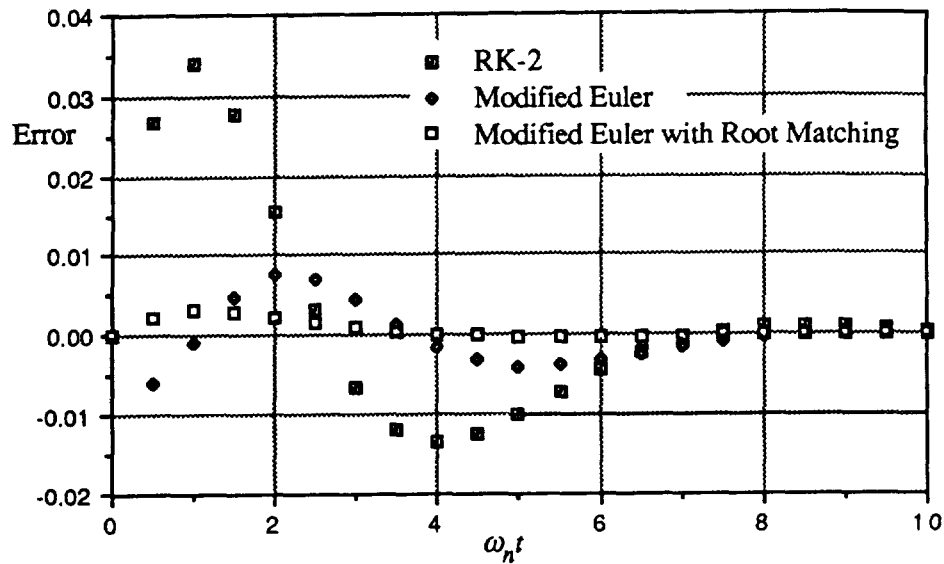
Figure 2. Unit step response errors in simulating a second-order system with damping ratio $\zeta = 0.707$, integration step size given by $\omega_n h = 0.5$.

5. Performance of Other Versions of Modified Euler Integration

In this section we present the asymptotic formulas for characteristic root and transfer function errors when modified Euler integration is used to simulate a second-order system with methods 2, 3, or 4 in Table 2 utilized to calculate the velocity estimate y_n' in Eq. (13). For method 2, which is equivalent to AB-2 integration for the damping term, the following results are obtained for e_ω, the fractional error in root frequency, and e_ζ, the damping ratio error [6]:

$$e_\omega \cong \frac{1 - 32\zeta^2 + 40\zeta^4}{24(1 - \zeta^2)}(\omega_n h)^2 , \qquad e_\zeta \cong \frac{11\zeta - 20\zeta^3}{24}(\omega_n h)^2 , \qquad \omega_n h << 1 \qquad (23)$$

These errors are significantly less than the errors when AB-2 is used for all integrations. For method 3 in Table 2, which uses a second-order predictor integration algorithm to compute y_n', the following asymptotic formulas are obtained for the root frequency and damping errors:

$$e_\omega \cong \frac{(1 - 4\zeta^2)}{24}(\omega_n h)^2 , \qquad e_\zeta \cong \frac{\zeta - \zeta^3}{12}(\omega_n h)^2 , \qquad \omega_n h << 1 \qquad (24)$$

The transfer function gain and phase errors are given by

$$e_M \cong -\frac{2\frac{\omega^2}{\omega_n^2}\left[1 - 2\zeta^2 - \frac{\omega^2}{\omega_n^2}\right]}{\left[1 - \frac{\omega^2}{\omega_n^2}\right]^2 + \left[2\zeta\frac{\omega}{\omega_n}\right]^2}\frac{(\omega_n h)^2}{24} \quad , \quad e_A \cong \frac{2\zeta\frac{\omega}{\omega_n}\left[1 + \frac{\omega^2}{\omega_n^2}\right]}{\left[1 - \frac{\omega^2}{\omega_n^2}\right]^2 + \left[2\zeta\frac{\omega}{\omega_n}\right]^2}\frac{(\omega_n h)^2}{24} \quad , \quad \omega h \gg 1$$

$$(25)$$

In both Eqs. (24) and (25) the errors are a factor of two smaller than the corresponding errors when AM-2 is used for all integrations. In addition, the AM-2 algorithm is a two-pass method which will therefore take twice as long to execute on a given computer. For method 4 in Table 2, which is equivalent to using Euler integration for the damping term, the following formulas are obtained for the characteristic root and transfer function errors [4]:

$$e_\omega \cong \frac{\frac{1}{2}\zeta - \zeta^3}{1 - \zeta^2}\omega_n h \quad , \quad e_\zeta \cong \frac{1}{2}\zeta^2\omega_n h \quad , \quad \omega_n h \ll 1 \tag{26}$$

$$e_M \cong \frac{-\zeta\frac{\omega}{\omega_n}\left[1 - \frac{\omega^2}{\omega_n^2}\right]}{\left[1 - \frac{\omega^2}{\omega_n^2}\right]^2 + \left[2\zeta\frac{\omega}{\omega_n}\right]^2}(\omega h) \quad , \quad e_A \cong \frac{2\zeta^2\frac{\omega^2}{\omega_n^2}}{\left[1 - \frac{\omega^2}{\omega_n^2}\right]^2 + \left[2\zeta\frac{\omega}{\omega_n}\right]^2}(\omega h) \quad , \quad \omega h \gg 1 \quad (26)$$

Note that the errors are all proportional to the first power of the step size h. This is because of the first-order Euler algorithm used for integration of the damping term. For $\zeta = 0$, however, the first-order errors in Eqs. 25) and (26) vanish, meaning that the errors become second-order in h. This is to be expected, since the conventional Euler integration plays no role when $\zeta = 0$. In fact it can be shown that when $\zeta = 0$, the digital solution will have zero damping regardless of the step size h.

When method 2,3, or 4 in Table 1 (or any other explicit method) is used to provide the estimate y_n' for the velocity state, the modified Euler method can be used as the algorithm for integrating the nonlinear state equations represented by (1). The vector difference equations become the following:

$$\dot{q}_{n+1/2} = \dot{q}_{n-1/2} + h\{F_n - [M(q_n)]^{-1}C(q_n,\dot{q}_n') - K(q_n)\} \quad , \quad q_{n+1} = q_n + h\dot{q}_{n+1/2} \tag{28}$$

We now turn to a consideration of integration algorithm stability.

6. Stability of Integration Methods

It has already been pointed out that the stability of numerical integration algorithms becomes an important consideration when the flexible structure is modeled by discretization. This is because the discretized model will contain high frequency modes which are unimportant in the simulation but can cause numerical instabilities for reasonable integration step sizes. For a given integration method the stability boundary in the λh plane can be obtained by considering a simulation of the linear system with transfer function $H(s) = 1/(s-\lambda)$. From the difference equation the z transform, $H^*(z)$, is obtained. The stability boundary is defined by the λh values for which the denominator of $H^*(z)$ vanishes when $|z| = 1$. These λh values can be obtained by letting $z = e^{j\theta}$ in the denominator of $H^*(z)$ and solving for λh for θ values ranging between 0 and π. When this is done for the AB predictor methods, the stability regions plotted in Figure 3 are obtained. The regions are symmetric with respect to the real axis so that only the upper half plane is shown. For any values of λh lying outside the boundaries the digital simulation will be unstable. In the case of both AB-3 and AB-4 the boundary crosses over into the right half plane. This means that a continuous system with roots on the imaginary axis which correspond to undamped transients can exhibit stable transients in the digital solution. Put another way, it means that AB-3 and AB-4 solutions will exhibit more damping than the continuous system being simulated. This is actually desirable in the case of the high frequency modes which are not of interest in a given simulation. On the other hand the AB predictor methods do not have particularly large stability regions and therefore do not permit very large integration step sizes h compared with the reciprocal magnitude, $1/|\lambda|$, of the largest eigenvalues in the simulation.

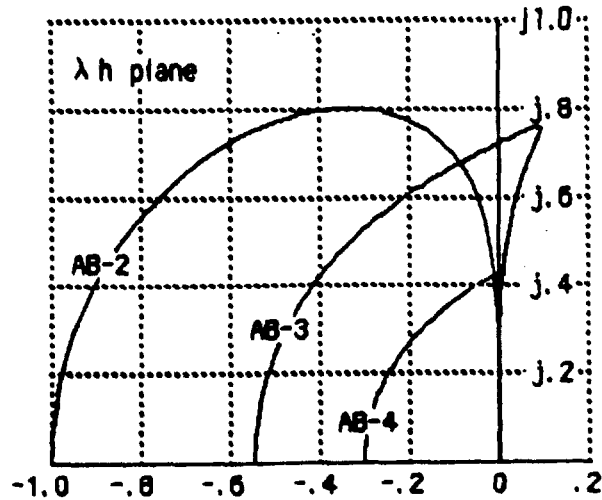Figure 3. Stability boundaries for AB predictor integration.

In Figure 4 the stability boundaries are shown for the the two-pass AM predictor-corrector methods. Although the boundaries are considerably larger than those for the AB methods, it must be remembered that the AM algorithms will take twice as long to execute. Thus the boundaries should be reduced by a factor of two for a valid comparison with AB-2. When this

is done, the AM-2 and 3 boundaries actually fall inside the AB-2 and 3 boundaries, although the AM-4 boundary still lies outside the AB-4 boundary. In all cases the higher-order algorithms exhibit less stability and are therefore unlikely to be candidates for simulating flexible structures.
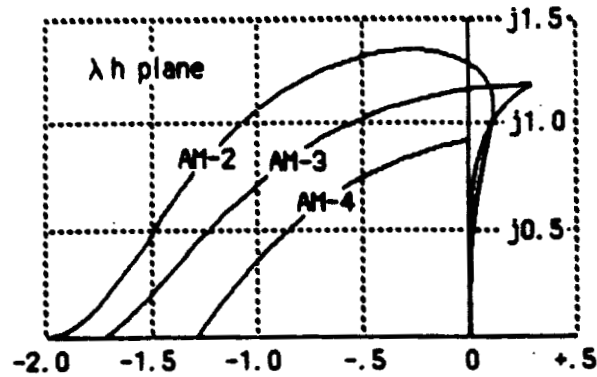
Figure 4. Stability boundaries for two-pass AM predictor-corrector integration.

For comparison purposes the stability boundaries for RK-2, 3 and 4 are shown in Figure 5. We recall that these algorithms require 2, 3 and 4 passes, respectively, through the state equations per integration step. Thus for proper comparison with single-pass methods the boundaries shown should be reduced by factors of 2, 3 and 4, respectively. When this is done, the RK-2 boundary roughly matches the AB-2 boundary, while the RK-3 and RK-4 boundaries still fall outside the AB-3 and 4 boundaries, respectively.

Figure 5. Stability boundaries for Runge-Kutta integration methods.

Finally, in Figure 6 are shown the stability boundaries for various modified Euler methods, as described in Sections 4 and 5. The trapezoidal damping case corresponds to method 1 in Table 2, the Euler damping case to method 4, the AB-2 damping case to method 2, and the predictor damping case to method 3. Also shown for comparison purposes in Figure 6 are the stability boundaries for AB-2, AM-2 and RK-2, as presented earlier in Figure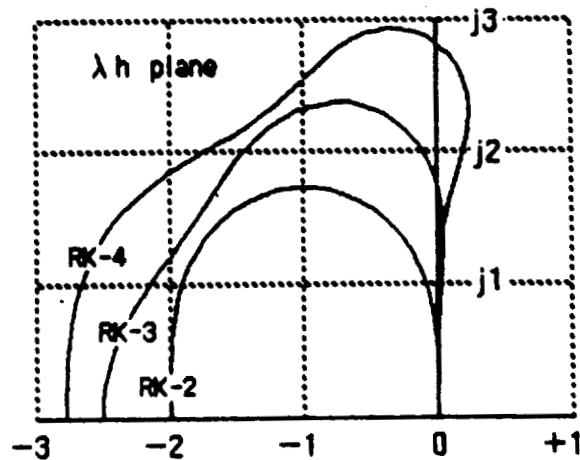s 3, 4 and 5, respectively. The AM-2 and RK-2 stability boundaries have been reduced by a factor of two to reflect the two passes per integration step required in the implementation of these methods. Note that all four of the Modified Euler methods in Figure 6 have stability regions which permit values of $|\lambda h|$ up to 2 for lightly damped transients, e.g., eigenvalues near the imaginary axis. In this regard the methods are considerably superior to the AB-2, AM-2 and RK-2 algorithms and should perform especially well in the simulation of flexible structures.

It should also be noted that the modified Euler methods are ideally suited for real-time simulation in that they do not require inputs prior to their occurence in real time. For example, if $F(t)$ in Eq. (1) is a real time input, then the single-pass modified Euler algorithm of Eq. (28) only requires F_n at the beginning of the nth integration step. On the other hand, the AM predictor-corrector algorithms require F_{n+1} at the start of the second pass for the nth integration step, and F_{n+1} is not yet available in real time. There is, however, a modified version of the AM-2 predictor method which is compatible with real-time inputs [6]. The AB predictor methods are also compatible with real time inputs, and there are versions of RK-2 and RK-3 which permit real-time inputs [3]. RK-4 is not compatible with real-time inputs, since it requires $F_{n+1/2}$ at the beginning of the second pass and F_{n+1} at the start of the fourth pass, in both cases prior to their availability in real time.
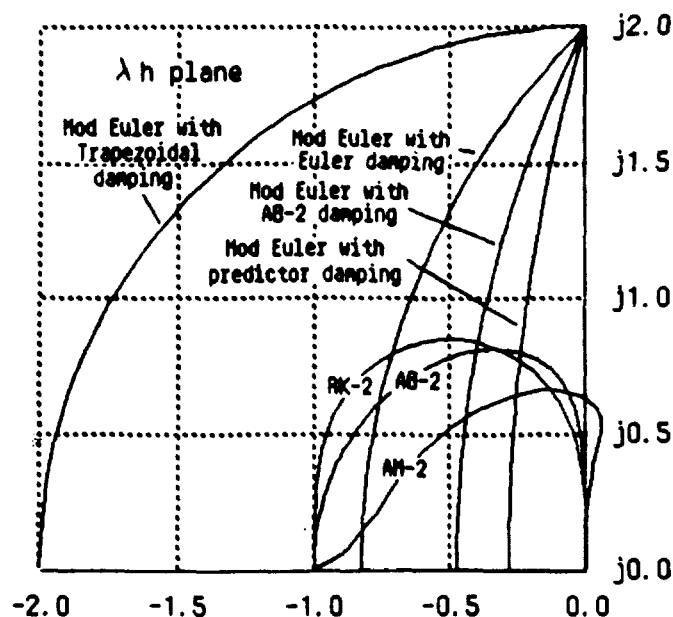

Figure 6. Stability boundaries for modified Euler integration methods.

7. Conclusions

In this paper we have considered the dynamic performance of integration methods in the context of simulating flexible structures. In terms of both characteristic root errors and transfer function errors, both important in such simulations, we have compared the performance of traditional integration methods with various versions of modified Euler integration . We have shown that modified Euler integration is especially effective in simulating lightly-damped structural modes. We have also shown that the modified Euler methods have very favorable stabilty boundaries in the λh plane with respect to requirements in the simulation of lightly-damped modes. This is especially significant when a flexible structure is modeled by discretization as opposed to normal coordinates, since it will allow larger integration step sizes before the solution goes unstable due to the presence of higher modes which are unimportant to the simulation.

References

1. Gilbert, E.G., "Dynamic Error Analysis of Digital and Combined Digital-Analog Systems," *Simulation*, vol. 6, no. 4, April 1966, pp 241-257.

2. Benyon, P.K., "A Review of Numerical Methods for Digital Simulation," *Simulation*, vol. 11, no. 4, Nov. 1968.

3. Howe, R.M., "Transfer Function and Characteristic Root Errors for Fixed-Step Integration Algorithms," *Transactions of the Society for Computer Simulation*, vol. 2, no. 4, Dec. 1985, pp 293-320.

4. Howe, R.M., "Simulation of Linear Systems Using Modified Euler Integration Methods," to appear in *Transactions of the Society for Computer Simulation*.

5. Fowler, M.E., "A New Numerical Method for Simulation," *Simulation*, vol. 4, no. 5, May 1965, pp 324-330.

6. Howe, R.M., "The Role of Modified Euler Integration in Real-Time Simulation," *Proceedings of the Conference on Aerospace Simulation II*, San Diego, 1986; pp 263-275. The Society for Computer Simulation, P.O. Box 17900, San Diego, CA 92117.

THIS PAGE INTENTIONALLY LEFT BLANK

OPTICAL PROCESSING FOR DISTRIBUTED SENSORS IN CONTROL OF FLEXIBLE SPACECRAFT

BY

SHARON S. WELCH, RAYMOND C. MONTGOMERY,
MICHAEL F. BARSKY, AND IAN T. GALLIMORE

NASA LANGLEY RESEARCH CENTER
SPACECRAFT CONTROL BRANCH
HAMPTON, VA 23665-5225

PRESENTATION FOR THE
WORKSHOP ON COMPUTATIONAL ASPECTS IN THE
CONTROL OF FLEXIBLE SYSTEMS
WILLIAMSBURG, VA

OPTICAL PROCESSING FOR DISTRIBUTED SENSORS IN
CONTROL OF FLEXIBLE SPACECRAFT

by

Raymond C. Montgomery, Sharon S. Welch
Michael F. Barsky, and Ian T. Gallimore

NASA Langley Research Center
Hampton, VA 23665-5225

ABSTRACT

This presentation will deal with a recent potential of distributed image processing. Applications in the control of flexible spacecraft will be emphasized. Devices are currently being developed at NASA and in universities and industries that allow the real-time processing of holographic images. Within 5 years, it is expected that, in real-time, one may add or subtract holographic images at optical accuracy. Images are stored and processed in crystal mediums. The accuracy of their storage and processing is dictated by the grating level of laser holograms. It is far greater than that achievable using current analog-to-digital, pixel oriented, image digitizing and computing techniques.

Processors using image processing algebra can conceptually be designed to mechanize Fourier transforms, least square lattice filters, and other complex control system operations. Thus, actuator command inputs derived from complex control laws involving distributed holographic images can be generated by such an image processor. The presentation will reveal plans for the development of a Conjugate Optics Processor for control of a flexible object.

PRESENTATION OUTLINE

- OBJECTIVES OF RESEARCH

- OPTICAL COMPUTING

- HOLOGRAPHIC INFORMATION STORAGE

- A PROPOSED EXPERIMENT IN THE CONTROL OF
    FLEXIBLE STRUCTURES

- CONCLUDING REMARKS

PRESENTATION OUTLINE

This talk deals with plans to develop optical distributed sensors and computation techniques for the control of flexible structures. The plan is to develop the technology relative to active vibration damping of structural dynamics systems and, then, to demonstrate it with a closed-loop control system experiment in the laboratory. First, objectives of the research will be presented. Next, fundamentals of optical computing will be briefly overviewed and new capabilities in real-time holography and holographic information storage will be discussed. An experiment being developed at Langley will be presented with emphasis on the sensor concept and the operations unique to optical distributed processing. The talk concludes with a status summary of both the analytical and laboratory work in this area.

517

OBJECTIVES OF RESEARCH

CONTROL OF FLEXIBLE STRUCTURES

- DEVELOP DISTRIBUTED SENSORS
- DEVELOP DISTRIBUTED COMPUTATION TECHNIQUES
- USE NEW TECHNOLOGY FOR OPTICAL PROCESSING

518

OBJECTIVES OF RESEARCH

Objectives of this research center around the interest in active control of flexible spacecraft structures. The intention is to develop distributed sensors for this application and complimentary distributed computing techniques. This has been enabled by recent advances in real-time holography using photorefractive crystals. Hence, some research objectives will be directed to laboratory development of optical processing using the new real-time holography techniques.

ADVANTAGES OF OPTICAL COMPUTING

- PARALLEL MULTIPLICATION

- PARALLEL ADDITION

- INTEGRATED DISTRIBUTED SENSING AND PROCESSING

ADVANTAGES OF OPTICAL COMPUTING

Here we list some advantages of optical computing. Parallel multiplication and addition are possible via optical computing but, more importantly, the potential of distributed sensing and signal processing exists.

OPTICAL COMPUTING TECHNIQUES

- ANALOGUE

  COHERENT
  INCOHERENT

- DIGITAL

  COHERENT
  INCOHERENT

522

OPTICAL COMPUTING TECHNIQUES

Both analogue and digital computing are possible via optics. These can be accomplished via both coherent and incoherent light. Our work will, however, be directed at analogue and coherent processing.

COHERENT ANALOGUE OPTICAL COMPUTING

Plane wave

Input image plane

Fourier transform plane

Output image plane

524

Input image	Input mask	Output image	Operation
g	H	g*h	convolution
g	H*	g○h	correlation
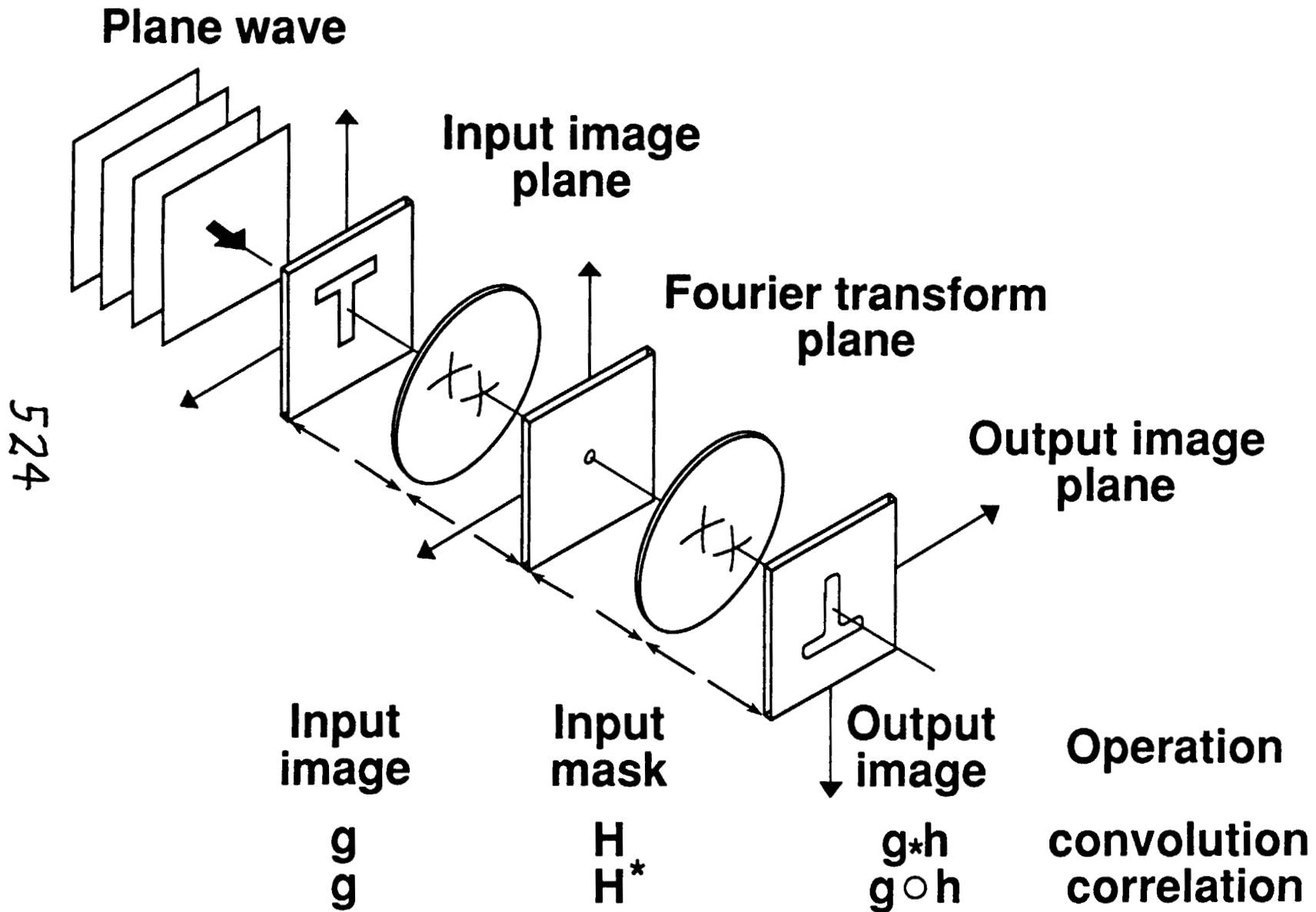
COHERENT ANALOGUE OPTICAL COMPUTING

An example of a coherent analogue optical processor, a plane wave correlator, is illustrated in this slide. The plane wave correlator performs two-dimensional correlation or convolution operations on two input functions $g(x,y)$, and $h(x,y)$. The operations of correlation and convolution are performed in the following way. An image transparency (mask) with transmittance $g(x,y)$, shown here as an image of the letter "T", is placed in the front focal plane of lens L1 (Input Image Plane). An undistorted plane wave of coherent light is passed through the image mask which introduces phase and amplitude changes in the light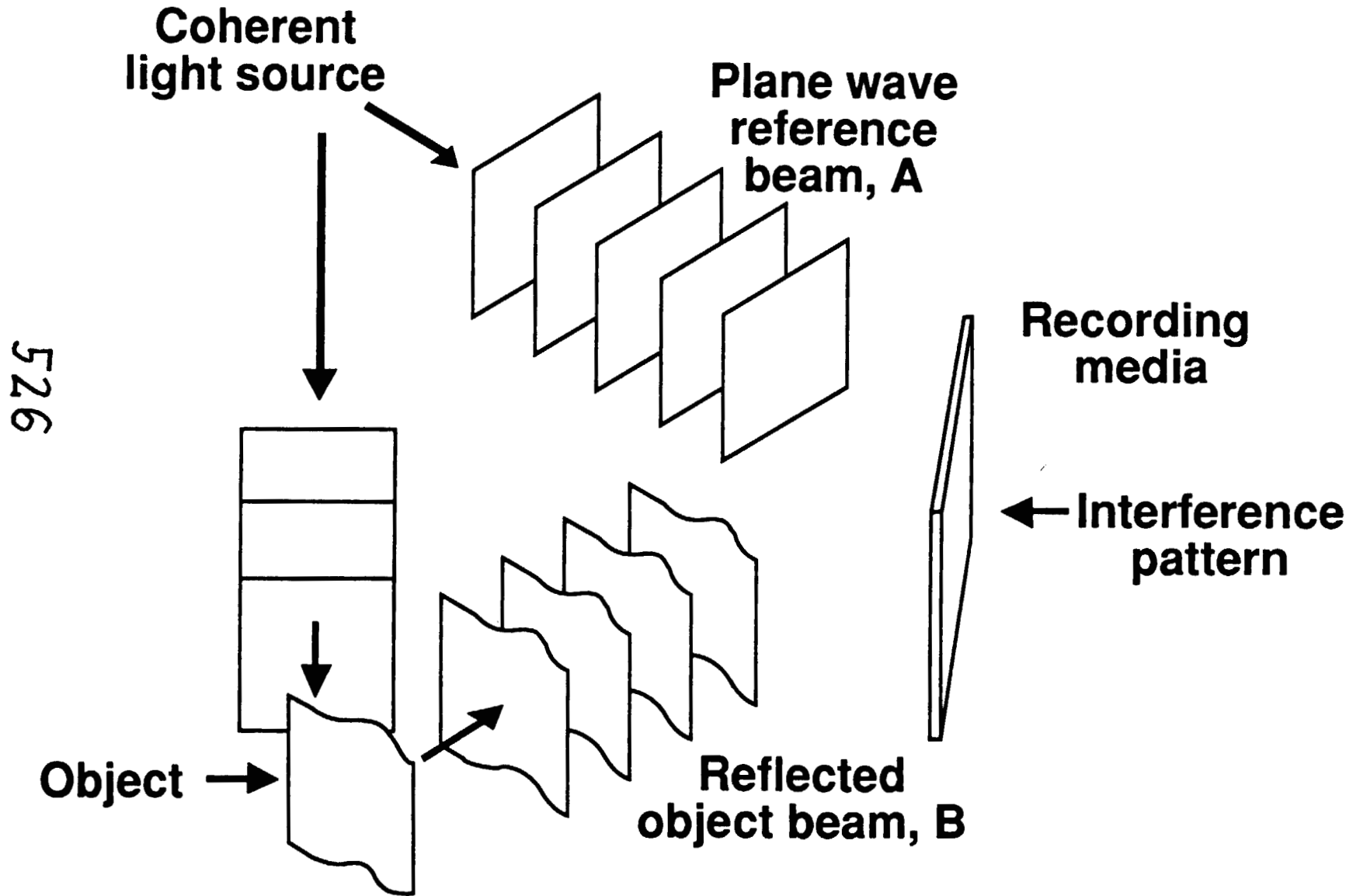. In the back focal plane of lens L1 the light amplitude distribution is proportional to the Fourier Transform of the transmittance of the input mask, $g(x,y)$. Thus, the back focal plane of L1, the Fourier Transform Plane, has coordinates which coorespond to spatial frequencies, here denoted by k_x and k_y. If a second mask is inserted in the Fourier Transform Plane with transmittance $H(k_x,k_y)$, then the light propagated through this mask is GH, where G is the Fourier transform of g. The second lens L2 performs a Fourier transform of the product GH, and thus, the image which is formed in the Output Image Plane is the intensity of the convolution of g with h, where h is the inverse Fourier transform of the second mask H. Similarly, if the mask inserted in the Fourier transform plane is H*, the image formed in the Output Image Plane is the correlation of g with h.

In the example shown in this slide, the input function is an image of the letter "T", and $H(k_x,k_y)$ is a circular aperature. The image formed in the Output Image Plane is the convolution of the image "T" with the inverse Fourier transform of the circular aperature function $H(k_x,k_y)$. The image which appears in the Output Image Plane is an image of the letter "T" with the high spatial frequencies removed.

Although a simple function was used in this example for H, in general the mask or filter may be complex and contain both amplitude and phase variations. To construct a mask (filter) which contains both amplitude and phase information, interferometric techniques are used. The mask thus formed is a holographic filter.

CONSTRUCTION OF A HOLOGRAPHIC FILTER

Coherent light source

Plane wave reference beam, A

Recording media

Interference pattern

Object

Reflected object beam, B

526

CONSTRUCTION OF A HOLOGRAPHIC FILTER

Now consider the construction of a holographic filter (mask). A holographic filter (or hologram) is simply the recorded intensity distribution corresponding to the interference of two (or more) light waves. The basic geometry for constructing a hologram of a three dimensional diffusely reflecting object is shown in this slide. An incident coherent plane wave is divided into two plane waves. One, the reference beam, travels directly toward the recording media (A). The second, travels toward the object. The portion of the wave reflected from the object (B) is changed in both amplitude and phase. This reflected object beam interferes with the reference beam at the plane of the recording media. The recording media (usually film) records the light intensity distribution of the interference of A with B.

The interference pattern need not be constructed as described above, but may be generated artificially using a computer. Computer generated holograms can be recorded on film or on a spatial light modulator. The recorded hologram provides a means of modulating (i.e., filtering) a light beam.

HOLOGRAM FILTER TRANSMITTANCE

Light intensity at the filter plane (film)

$$I(x,y) = |A(x,y)|^2 + |B(x,y)|^2$$

$$+ 2A(x,y)B(x,y) \cos (P(x,y) - Q(x,y))$$

Filter transmittance

$$Tf(x,y) = Tb + C(|B|^2 + A^*B + AB^*)$$

HOLOGRAPHIC FILTER TRANSMITTANCE

In this slide is the mathematical expression which describes the light intensity distribution at the recording media (previous slide). The expression for the intensity I(x,y) is composed of three terms. The first two terms are proportional to the intensities of the reference wave and object wave respectively. The third term is the interference term and depends upon the relative phases of the reference and object beams.

The second expression is the transmittance of the developed film. The first term in this expression is a constant which results from uniform exposure over the recording media, that is, the A^2 term in the expression for the intensity. The remaining three terms are, respectively, the intensity of the object beam, the product of the complex conjugate of the field amplitude of the reference and the field amplitude of the object and the product of the field amplitude of the reference and the complex conjugate of the object.

When the recorded hologram is illuminated with a beam which is the exact duplicate of the reference wave, the filter reconstructs the original object wave, B. If the hologram is illuminated with a beam which is the conjugate of the reference wave, the filter reconstructs the conjugate of the object beam, or B*.

529

REAL-TIME HOLOGRAPHY WITH A VARIABLE MASK

- REQUIRED FOR MANY OPTICAL COMPUTING APPLICATIONS

- TWO TECHNIQUES AVAILABLE

    - SPATIAL LIGHT MODULATORS

    - DFWM IN PHOTOREFRACTIVE CRYSTALS

530

REAL-TIME HOLOGRAPHY WITH A VARIABLE MASK

For many optical processing applications it is necessary to change or update the holographic filter real-time. Two methods exist for recording a holographic filter real-time, using devices known as spatial light modulators, or using a nonlinear optical technique known as degenerate four wave mixing (DFWM) in a photorefractive crystal.

A spatial light modulator is a device composed of a matrix of individual pixels of variable optical transmittance. The transmittance of an individual pixel is adjusted by varying a voltage to the pixel. The pixel voltages are addressable in x and y. One example of a spatial light modulator is a liquid crystal display.

SPATIAL LIGHT MODULATORS

ADVANTAGES

- **PROVEN TECHNOLOGY**

- **INTERFACE WITH DIGITAL SYSTEMS**

DISADVANTAGES

- **LOW RESOLUTION**

SPATIAL LIGHT MODULATORS

Advantages of spatial light modulators is that they are a proven technology. They can also be driven conveniently with signals from digital computers. Also, their outputs can be conveniently interfaced via photodiodes to digital computers.

The disadvantages of these devices is that they have relatively low spatial resolution.

DFWM IN PHOTOREFRACTIVE CRYSTALS

ADVANTAGES

- HIGH RESOLUTION

- ALL OPTICAL

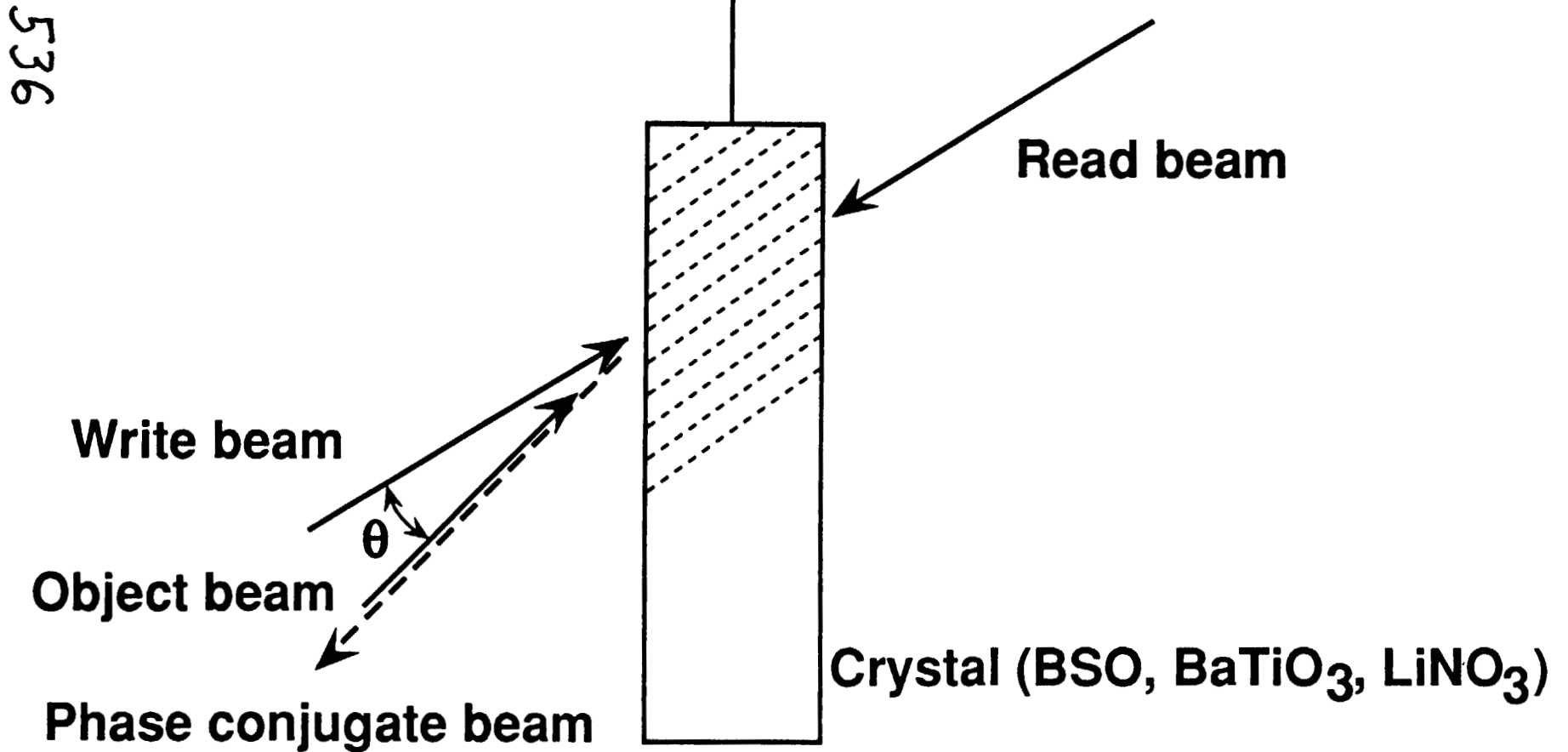- SIGNAL AMPLIFICATION POSSIBLE

DISADVANTAGES

- DIFFICULT TO WORK WITH (MANY VARIABLES)

534

DFWM IN PHOTOREFRACTIVE CRYSTALS

Advantages of DFWM are: extremely high spatial resolution (at the level of atomic particles), all optical systems, and the potential for optical signal amplification. Signal amplification is perhaps one of the most important advantages. Typically, in an optical system about 4% of the light is lost as the beam passes through each optical element due to scattering, internal reflections, and so on. This can be reduced somewhat with optical coatings, but even with optical coatings some light is lost. The result is that the light beam will be significantly attenuated after passing through a relatively few number of components (i.e., lenses, masks, etc.). With DFWM, signal amplification on the order of 100 is possible, and thus, the problem of signal attenuation can be overcome.

The greatest disadvantage of DFWM is that the devices are still in the development stage. There are many variables such as beam alignment, temperature, light intensity, etc. that must be carefully adjusted, and as a result, this technique of doing real-time holography is difficult to implement.

DFWM IN PHOTOREFRACTIVE MEDIA

C axis

536

Read beam

Write beam

θ

Object beam

Phase conjugate beam

Crystal (BSO, BaTiO$_3$, LiNO$_3$)

DFWM IN PHOTOREFRACTIVE MEDIA

In this slide is a diagram illustrating the typical geometry for performing degenerate four wave mixing (DFWM) in a photorefractive crystal. Some examples of photorefractive crystals are Bismuth Silicon Oxide (BSO), Barium Titanate ($BaTiO_3$), and Lithium Niobate ($LiNO_3$). DFWM is a nonlinear optical process whereby a phase conjugate beam is produced by mixing (or causing to interfere) three coherent beams of light of the same wavelength within a medium.

The medium in this case is a photorefractive crystal. The photorefractive crystal provides a unique way of recording light intensity which allows multiple beams to be mixed. Light incident upon a photorefractive crystal causes trapped charges within the crystal to migrate. The charges migrate to regions of low light intensity and become retrapped. If the light intensity distribution over the crystal is not uniform, then a non-uniform distribution of charge will be established within the crystal. This distribution of charge will in turn give rise to a spatially varying electric field. The induced electric field causes a change in the index of refraction through the electro-optic effect. In this way, the distribution of light intensity over the volume of the crystal is recorded as a change in index of refraction.

In the diagram, three coherent beams of light, the write beam, the object beam, and the read beam are incident upon a photorefractive crystal simultaneously. The interference pattern generated by the interference of these three beams, produces (as described above) a spatially varying index of refraction within the crystal. The variation in index of refraction within the crystal is similar to a recorded hologram, and causes diffraction of the beams. If the read beam is identical to the write beam and counterpropagating with respect to the write beam, the diffracted beam is the phase conjugate of the object beam.

DFWM in a photorefractive crystal is similar to conventional holography where the hologram is illuminated with the phase conjugate of the reference, producing a phase conjugate of the object. The difference between DFWM and conventional holography is that in the DFWM process, the hologram is written and read simultaneously. Because of the fast response times of many photorefractive crystals, holography may be performed real-time using DFWM in a photorefractive crystal.
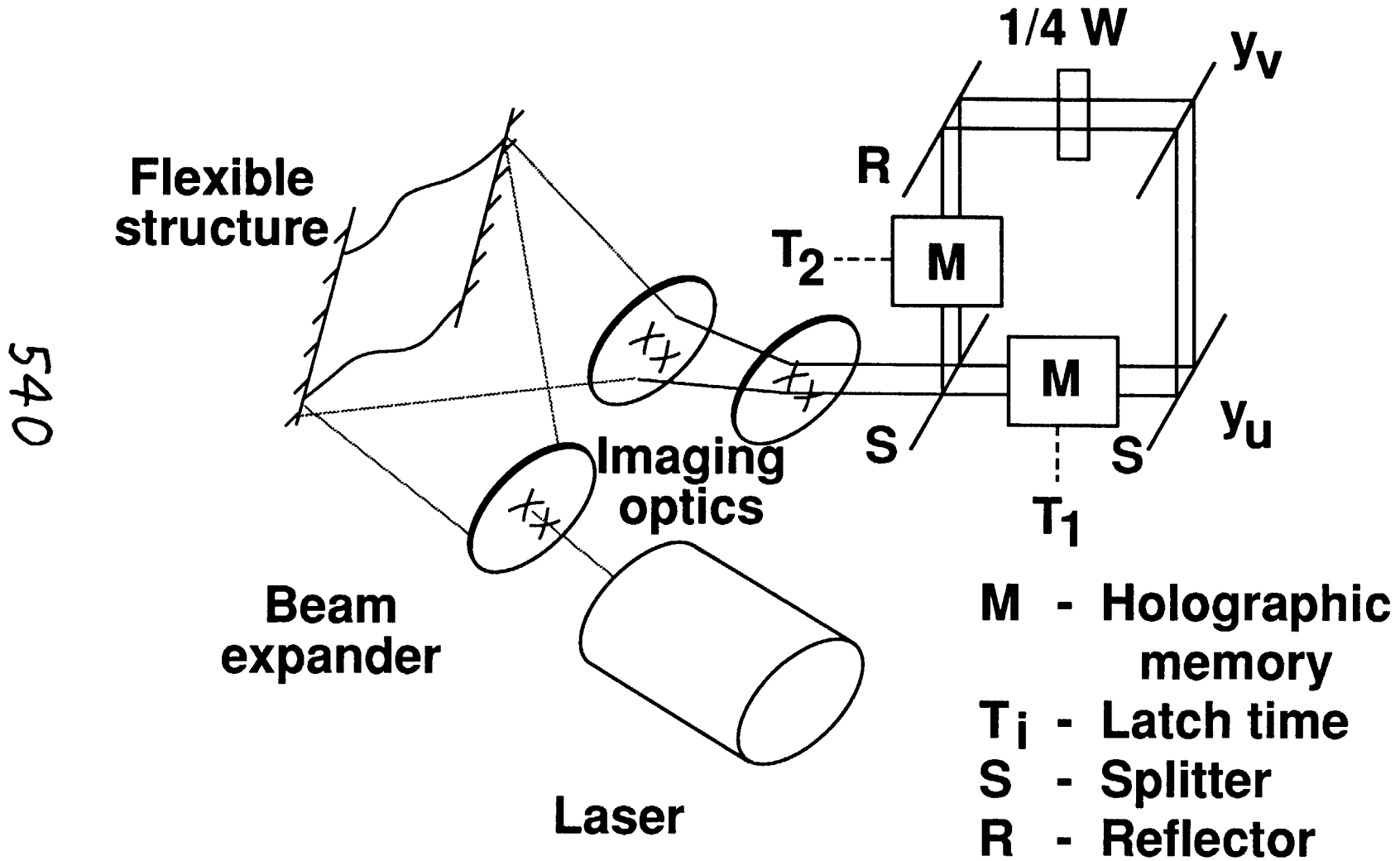
537

TWO-CRYSTAL
OSCILLATOR FOR STORAGE AND COMPUTING

Write beam 1 → Photorefractive crystal ← Read beam 1

Object beam 1 →

Object beam 2 →

Write beam 2 → Photorefractive crystal ← Read beam 2
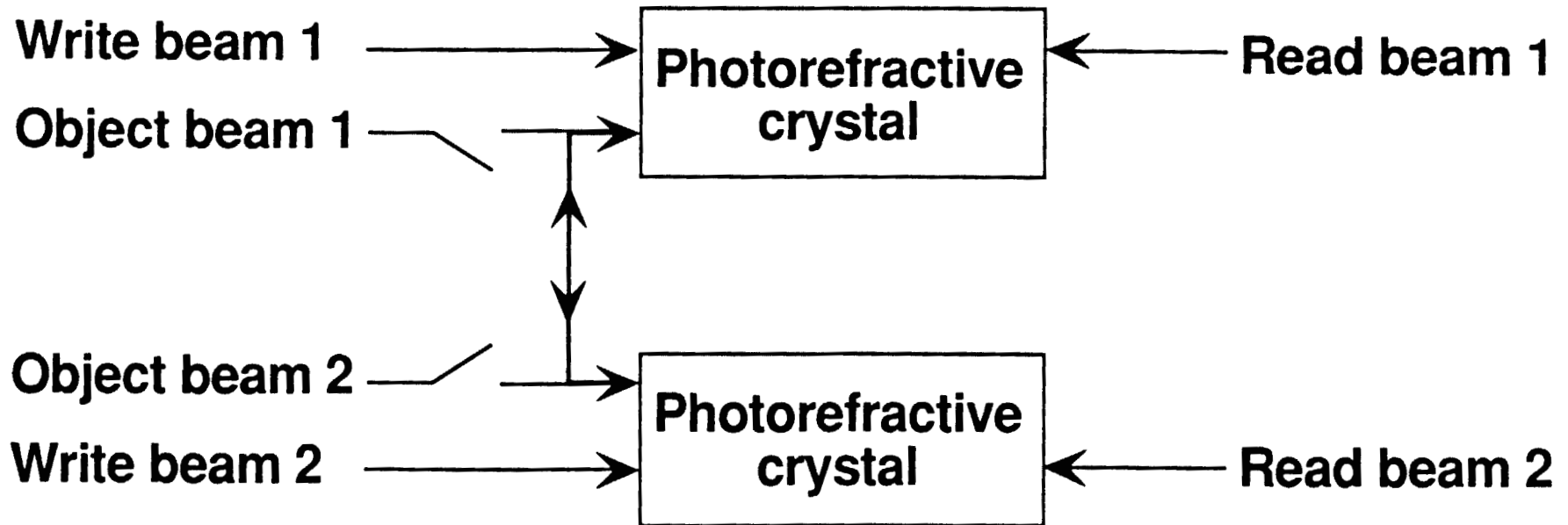
TWO-CRYSTAL OSCILLATOR FOR STORAGE AND COMPUTING

Illustrated in this slide is a schematic diagram of a new optical memory being developed at NASA Langley. The memory will allow more efficient computational use of the crystals used in real-time holography. The memory consists of a two crystal oscillator. Each crystal has independent write, object, and read beams. Here, the top crystal may be used to write a hologram and (simultaneously) produce a phase conjugate beam. The phase conjugate output of this crystal is routed as the object beam to the bottom crystal. The phase conjugate output of the lower beam is thus the original object beam. The input to the top crystal may be switched to the phase conjugate output of the bottom crystal. The images will then oscillate in the optical path between the crystals and, with proper gain stabilization, retain the original object image operating like a conventional MOS dynamic RAM.

539

SENSOR SYSTEM

1/4 W

y_v

R

T_2 --- M

M

S

S

y_u

T_1

Flexible
structure

Imaging
optics

Beam
expander

Laser

540

M - Holographic
memory
T_i - Latch time
S - Splitter
R - Reflector

SENSOR SYSTEM

Our intention is to develop a optical control system, based on the concepts presented, and to demonstrate it in a closed-loop laboratory test. The test structure will most probably be a beam because of the simplicity in representing the dynamics as a partial differential equation. This slide illustrates the sensor concept for the experiment. A Q Switched Laser source through a beam expander is used to illuminate the Flexible Structure. The object beam from the structure is focused to a small beam and is then defocused to a straight beam. This beam contains in phase information the image of the object. It is passed through a beam splitter to two holographic memory devices where the images are retained. These are latched at different times and the inteference between these images is used to obtain a rate image. One image is phase shifted by one quarter wave length to produce the correct intensity variation in the interference beam.

CONTROLLER STRUCTURE

- **PDE MODEL** $-$ $U_{tt} + U_{xxxx} = 0$

- **A PRIORI MODEL** $-$ $U(X,K) = U(X,K-1) + V(X,K-1)T$
$$-F^{-1}\{K^4 F[U(X,K-1)]\}T /2$$
$$V(X,K) = V(X,K-1) -F^{-1}\{K^4 F[U(X,K-1)]\}T$$

- **UPDATE EQUATION** $-$ $U(X,K) = U(X,K) + \int_\Omega G_Y(X,Z)E(Z,K)DZ$
$$E(Z,K) = Y_M(Z,K) - Y(Z,K)$$
$$Y(Z,K) = H_U(Z)U(Z,K) + H_V(Z)V(Z,K)$$

- **CONTROL LAW** $-$ $F(K) = \int_\Omega G_U(Z)U(Z,K)DZ$

  NOTE: IF $G_Y(X,Z) = G_Y(X-Z)$ AND $G_U(X,Z) = G_U(X-Z)$ THEN
    THE INTEGRAL CAN BE COMPUTED VIA FOURIER OPTICS.

CONTROLLER STRUCTURE

The controller will process state type stored images. In this case the state images are position and velocity images. A conventional estimator structure will be first attempted. In this case the a priori model is obtained from the partial differential equation model of the system by an Euler integration scheme. Hence, the a priori estimate of the state at sample k is obtained from the position and velocity information at sample k-1. The true value of the scheme is the fact that the Fourier transform operation shown on the slide can be accomplished using Fourier optics as previously described. This type of distributed model of the system completely eliminated spillover caused by modal representations of the system dynamics.

The estimator update equation would usually appear as shown on the slide. It involves an integral of estimation errors taken over the space of the state image. If the gain operator is shift invariant, depending only on x-z, then the integral in the update equation can be accomplished also via Fourier optics. finally, the control law takes the form of the integral shown. Again, if the gain operator is shift invariant, the integral can be accomplished via Fourier optics. It the gain operator is not shift invariant a more general holographic technique would have to be developed. The output of the control law to drive an actuator will probably be accomplished using a photodiode. The signal would be amplified appropriately and used to drive an actuator such as a torque wheel attached to the beam.

543

CONCLUDING REMARKS

- EXPERIMENT PROPOSED FOR DEVELOPMENT AND
  DEMONSTRATION OF DISTRIBUTED SENSING

- POTENTIAL TO ELIMINATE SPILLOVER CAUSED
  BY MODAL MODELLING

- STATUS OF HARDWARE DEVELOPMENT

  - HOLOGRAPHY WITH PHOTOGRAPHIC PLATES DONE
  - CRYSTALS HAVE BEEN DELIVERED TO LaRC
  - REAL-TIME HOLOGRAPHY DEMONSTRATION IS
    IN PROGRESS

- STATUS OF DISTRIBUTED SENSING RESEARCH

  - SYSTEM HAS BEEN CONCEPTUALLY DESIGNED
  - SIMULATION OF OPTICAL COMPONENTS IN PROGRESS

CONCLUDING REMARKS

An experiment has been proposed to facilitate development of an optical processor that processes distributed signals. The input to the processor is a distributed image of a structure and the output of the optical processor will probably be used to drive a torque wheel. Processing using distributed images will eliminate spillover caused by modal representations of controllers.

To this time we have accomplished conventional holography via photographic plates in our laboratory. This was done to create holograms of structures which are digitized and input to a Sun workstation computer system thus enabling simulation developement to proceed in pace with simultaneous laboratory development. Photorefractive crystals have been delivered to Langley and we are now in the process of developing an in-house real-time holography capability.

At this time the control system has ben conceptually designed. Detail design awaits the real-time holographic and dynamic memory developments and demonstrations. Analytical simulations of the optical components involved are currently being programmed for the Sun workstation.

NASA
National Aeronautics and Space Administration

Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TM-101578, Part 1		

4. Title and Subtitle		5. Report Date
Proceedings of the Workshop on Computational Aspects in the Control of Flexible Systems		May 1989
		6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Lawrence W. Taylor, Jr. (Compiler)	
	10. Work Unit No.
	506-46-11-01

9. Performing Organization Name and Address	11. Contract or Grant No.
NASA Langley Research Center Hampton, VA 23665-5225	
	13. Type of Report and Period Covered

12. Sponsoring Agency Name and Address	Technical Memorandum
National Aeronautics and Space Administration Washington, DC 20546-0001	14. Sponsoring Agency Code

15. Supplementary Notes

This report is part 1 of two parts published as NASA TM-101578.

16. Abstract

This publication is a collection of papers presented at the Workshop on Computational Aspects in the Control of Flexible Systems held at the Royce Hotel, Williamsburg, Virginia, July 12-14, 1988. The papers address the formulation, modeling, computation, software and control for flexible spacecraft, aircraft and robotic manipulators.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Large Flexible Spacecraft Control, Strucutal Dynamics	Unclassified-Unlimited Subject Category - 18

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassifed	Unclassified	553	A24

NASA FORM 1626 OCT 86