

# **ADAPT**

**STRUCTURAL CONCRETE SOFTWARE SYSTEM**

## **ABI-GEN** for **Segmentally Constructed Bridges and Frames**

Copyright 1997; 2005





## LIST OF CONTENTS

<b>1.1</b>	<b>INTRODUCTION .....</b>	<b>1-1</b>
<b>1.2</b>	<b>PRE-, AND POST-CONSTRUCTION PHASES .....</b>	<b>1-1</b>
	1.2.1 OUTLINE .....	1-1
	1.2.2 PRE-CAPTURE CONDITION (CONSTRUCTION PHASE) .....	1-3
	1.2.3 POST-CAPTURE (IN-SERVICE CONDITION) .....	1-3
<b>1.3</b>	<b>STRUCTURE .....</b>	<b>1-4</b>
<b>2.1</b>	<b>GENERAL .....</b>	<b>2-1</b>
<b>2.2</b>	<b>DEFINITION OF STRUCTURE .....</b>	<b>2-1</b>
<b>2.3</b>	<b>LOAD PATTERN; ENVELOPE .....</b>	<b>2-3</b>
<b>2.4</b>	<b>LOAD COMBINATION .....</b>	<b>2-3</b>
<b>2.5</b>	<b>HYPERSTATIC ACTIONS DUE TO PRESTRESSING .....</b>	<b>2-5</b>
<b>2.6</b>	<b>PRESTRESSING ACTIONS .....</b>	<b>2-6</b>
<b>2.7</b>	<b>LONG-TERM LOSSES .....</b>	<b>2-6</b>
<b>3.1</b>	<b>OVERVIEW .....</b>	<b>3-1</b>
	3.1.1 CAPABILITIES .....	3-1
	3.1.2 PREPARATION OF INPUT DATA .....	3-1
<b>3.2</b>	<b>COMMAND SUMMARY .....</b>	<b>3-3</b>
	3.2.1 NATURAL SEQUENCE OF COMMANDS .....	3-3
	3.2.2 SUMMARY OF COMMANDS .....	3-5
<b>3.3</b>	<b>SAMPLE INPUT FOR A NON-PRESTRESSED FRAME .....</b>	<b>3-8</b>
	3.3.1 DESCRIPTION OF THE FRAME .....	3-8
	3.3.2 INPUT FILE .....	3-11
<b>3.4</b>	<b>DETAILED DESCRIPTION OF COMMAND SYNTAX .....</b>	<b>3-13</b>
	CAPTURE STATUS .....	3-13
	COLUMN .....	3-14
	COMBINATION .....	3-15
	COMBINATIONS .....	3-17
	GENERIC .....	3-19
	GRID .....	3-20
<b>4.1</b>	<b>BACKGROUND .....</b>	<b>4-1</b>
<b>4.2</b>	<b>FEATURES OF ADAPT-GEN .....</b>	<b>4-1</b>
	4.2.1 PRE-CAPTURE FEATURES .....	4-1
	4.2.2 POST-CAPTURE FEATURES .....	4-1
<b>4.3</b>	<b>STRESSES IN CONCRETE; PRE-, AND POST-CAPTURE ALLOCATION OF ACTIONS AT A NODE .....</b>	<b>4-3</b>
	4.3.1 DESCRIPTION .....	4-3
	4.3.2 ILLUSTRATIVE EXAMPLE .....	4-5
<b>4.4</b>	<b>HYPERSTATIC ACTIONS: SECONDARY ACTIONS .....</b>	<b>4-9</b>
<b>4.5</b>	<b>ENVELOPES AND COMBINATIONS .....</b>	<b>4-10</b>
<b>5.1</b>	<b>GENERAL .....</b>	<b>5-1</b>

<b>5.2</b>	<b>CONSTRUCTION PHASE (PRE-CAPTURE)</b> .....	<b>5-1</b>
<b>5.3</b>	<b>REGULAR LOADING CASES</b> .....	<b>5-2</b>
<b>5.4</b>	<b>PATTERNED (SKIPPED) LOADING CASES; ENVELOPES</b> .....	<b>5-2</b>
<b>5.5</b>	<b>PRESTRESSING</b> .....	<b>5-4</b>
<b>5.6</b>	<b>COMBINATIONS</b> .....	<b>5-5</b>
<b>6.1</b>	<b>GENERAL</b> .....	<b>6-1</b>
<b>6.2</b>	<b>EXAMPLE 1: EXTRACTION OF TIME-DEPENDENT EFFECTS</b> .....	<b>6-1</b>
	6.2.1 DESCRIPTION .....	6-1
	6.2.2 INPUT DATA .....	6-5
<b>6.3</b>	<b>EXAMPLE 1: TWO-STORY POST-TENSIONED FRAME</b> .....	<b>6-7</b>
	6.3.1 DESCRIPTION .....	6-7
	A. Cross-sectional properties: .....	6-7
	B. Material Properties: .....	6-7
	Concrete: .....	6-7
	Steel: .....	6-8
	Springs: .....	6-8
	C. Prestressing: .....	6-8
	D. Loading: .....	6-11
	6.3.2 INPUT DATA .....	6-16
<b>6.4</b>	<b>EXAMPLE 3: THREE STORY PARKING STRUCTURE SPLIT FRAME</b> .....	<b>6-22</b>
	6.4.1 DESCRIPTION .....	6-22
	6.4.2 INPUT DATA .....	6-26

## 1.1 INTRODUCTION

ADAPT-GEN is a computer program for the analysis and design of prestressed or nonprestressed concrete frames, during the construction phase, or when complete. It is based on a general finite element engine (ADAPT-ABI) with specific provisions for concrete material, prestressing, and composite construction.

ADAPT-GEN adds ease of modeling, and post-processing for analysis and design of real structures to its basic module (engine) ABI. This software is tailored to be a powerful production tool for analysis and design of frames, either in building construction or in bridges.

The principal features of ADAPT-ABI are:

1. Addition of an enhanced post-construction analysis capability, with options for load combination and envelope extraction;
2. Ability to re-define and refer to the structural system in terms of the familiar concepts of frame subdivision into “levels”, “spans”, “columns”, as opposed to nodes and elements;
3. Provisions for skipping (patterning) of loading to maximize the effects of load;
4. Extraction of hyperstatic (secondary) actions due to prestressing, and the option to combine these with other actions to arrive at design actions; and
5. Extraction of actions and deformations due to long-term effects of creep and shrinkage; extraction of actions due to prestressing.

## 1.2 PRE-. AND POST-CONSTRUCTION PHASES

### 1.2.1 Outline

Consider **Figure 1.2-1**. It shows the construction sequence of a typical frame of a parking structures with split levels. To demonstrate the concept, only two intermediate construction conditions, stages 1 and 2, (**Figure 1.2-1(a)** and **(b)**) are illustrated. If required, the construction can be broken down into additional intermediary stages. The completed frame is shown in **Figure 1.2-1(c)**.

During the construction phase, the support condition, the shoring, and the construction loading apply. After the frame is placed in service, other loadings, such as the in-service live loads would also apply. The long-term design for serviceability and strength of the completed frame is typically calculated at a time when the frame is aged for two or more years. At this time, the time-dependent long-term effects are deemed to have essentially settled.

Based on the foregoing, the software conducts the analysis in two stages; namely:

- A. construction phase, or initial phase, and
- B. post-construction, or in-service condition.

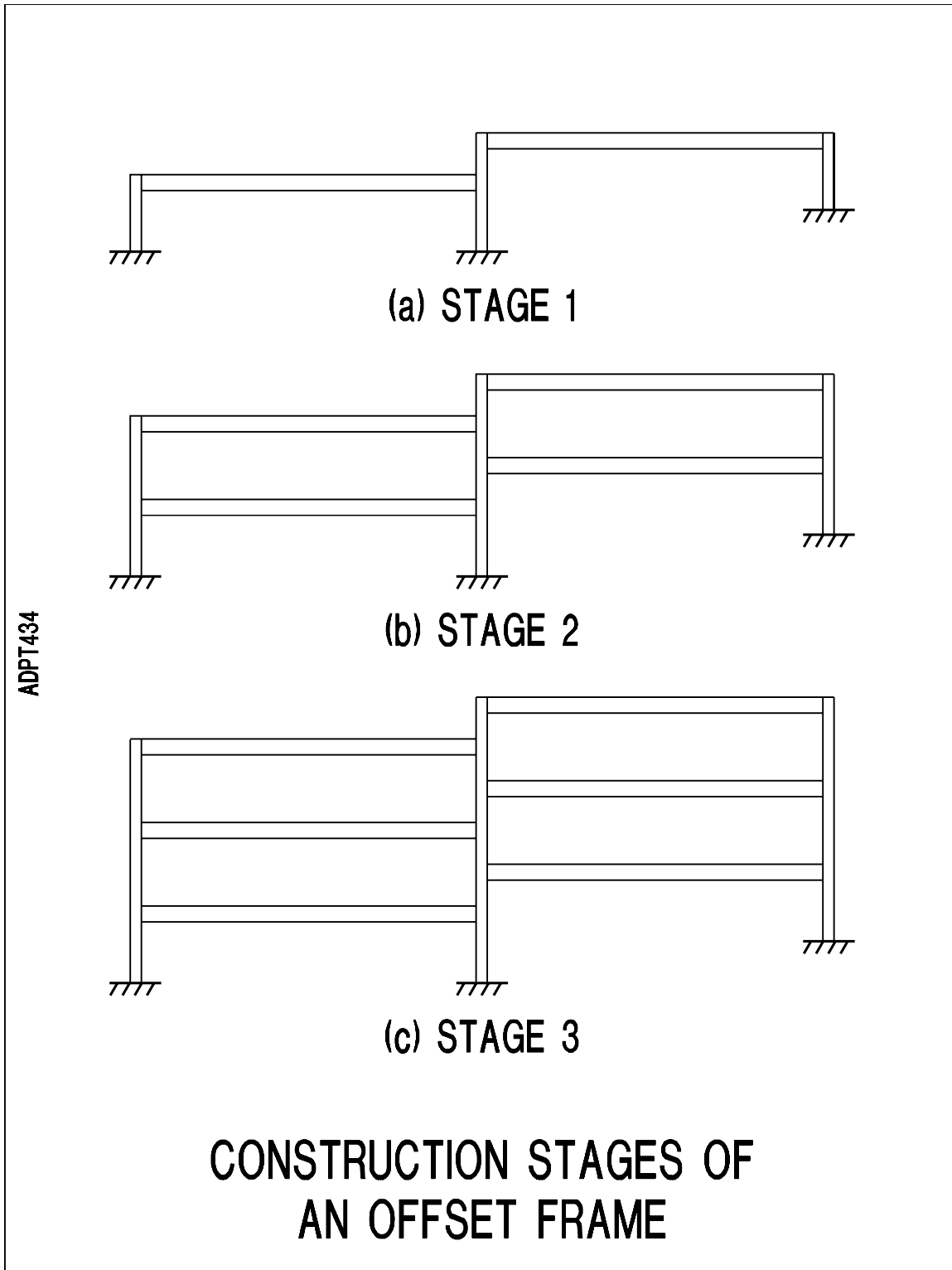


FIGURE 1.2-1

The strength limit state condition is a scenario of the post-construction stage. The separation between the two stages is made by the user. In the input data, the user enters a command (CAPTURE STATUS) to effect demarcation between the two stages

of pre- and post-construction. Details of this operation are given in Chapter 3 of the manual.

### 1.2.2 Pre-Capture Condition (Construction Phase)

At pre-capture phase, the structure is constructed, assembled, or dismantled, with essentially all the features of the basic ABI engine. For details refer to ADAPT-ABI Program Manual. At this stage, frame parts can be added or deleted. Shoring, or other types of support can be introduced or removed. Tendons can be stressed or de-stressed. The behavior of the in-construction frame is time-dependent, with due allowance to creep and shrinkage of concrete, aging of concrete, relaxation in pre-stressing. The frame response is also dependent on the load and stress history of the frame.

No specific time lapse is associated with the construction (pre-capture) phase. The construction stage can be extended in time, until, in the opinion of the user, the in-service condition of the frame is reached. This time lapse can be several years. The actual construction period itself can be short - expressed in terms of days. But a lapse of time of the order of years may be called for, to allow for the remaining changes in the long-term effects to become negligible.

ADAPT-GEN has the capability to faithfully model and assemble a frame during the pre-capture stage. For example, the tendon layout shown in **Figure 1.2-2** for the beam at split level of a parking structure can be specified exactly as is illustrated. Tendon layout and stressing will be as required for construction. The stressing can be phased with construction of the beam. Other levels will be stressed independently at their respective due times.

### 1.2.3 Post-Capture (In-Service Condition)

At in-service stage, the time element is disregarded. For example, the live loading, wind or seismic loading, or any other loading specified at this stage, is considered to be applied instantaneously.

Many loading cases can be specified for the post-construction phase. For each loading, the software obtains an independent solution, as if that loading were applied alone. Unlike the preconstruction phase, the load and stress history of the structure is not retained. Hence, each solution is independent. As a result, pattern loading for the post-construction phase becomes possible. If requested by the user, the software obtains several solutions for a given loading, each with the loading placed at a different location. The results of the individual loading cases are then combined into an envelope of maxima of actions and displacements.

Solutions from different loading cases can be combined, each with a factor specified by the user.

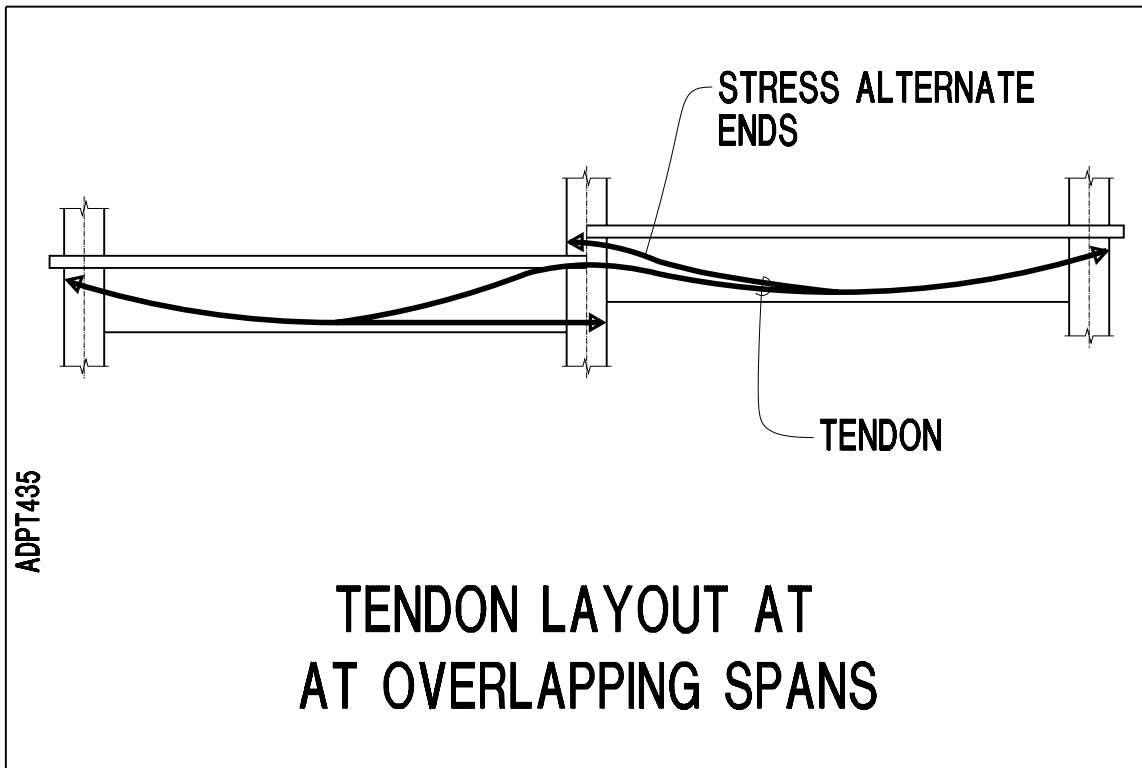


FIGURE 1.2-2

### 1.3 STRUCTURE

ADAPT-GEN has the ability to view the structure as a system of levels, consisting of spans, and column grid lines, consisting of individual columns. Loading can be specified in relation to spans and columns, as opposed to nodes and elements. The results of the analysis can be organized for the display and the tabulated hard-copy in terms of spans and columns. These features bring the analysis and the output close to the user's intuitive orientation to modeling and computational procedure for the frame.

A frame prototype, such as shown in **Figure 1.2-1(c)** is first modeled into a structural system (**Figure 1.3-1(a)**), in which centerline dimensions and boundary conditions, as well as assumptions relating to the nature of connection of members are fully defined. For example in the structural modeling illustrated, the connection between the upper end of outer columns to the beams is assumed hinged. This structural model is discretized into nodes and elements for a finite element analysis (ABI) in the conventional manner, as described in ABI.

In addition and parallel to the finite element discretization, in ADAPT-GEN the frame model can be redefined into columns lines, levels and spans, such as SPAN1 in level BEAM1 in **Figure 1.3-1(b)**. Loading and output can be expressed in terms of the redefined structure. Details of the definition and its features are given in Chapter 2.



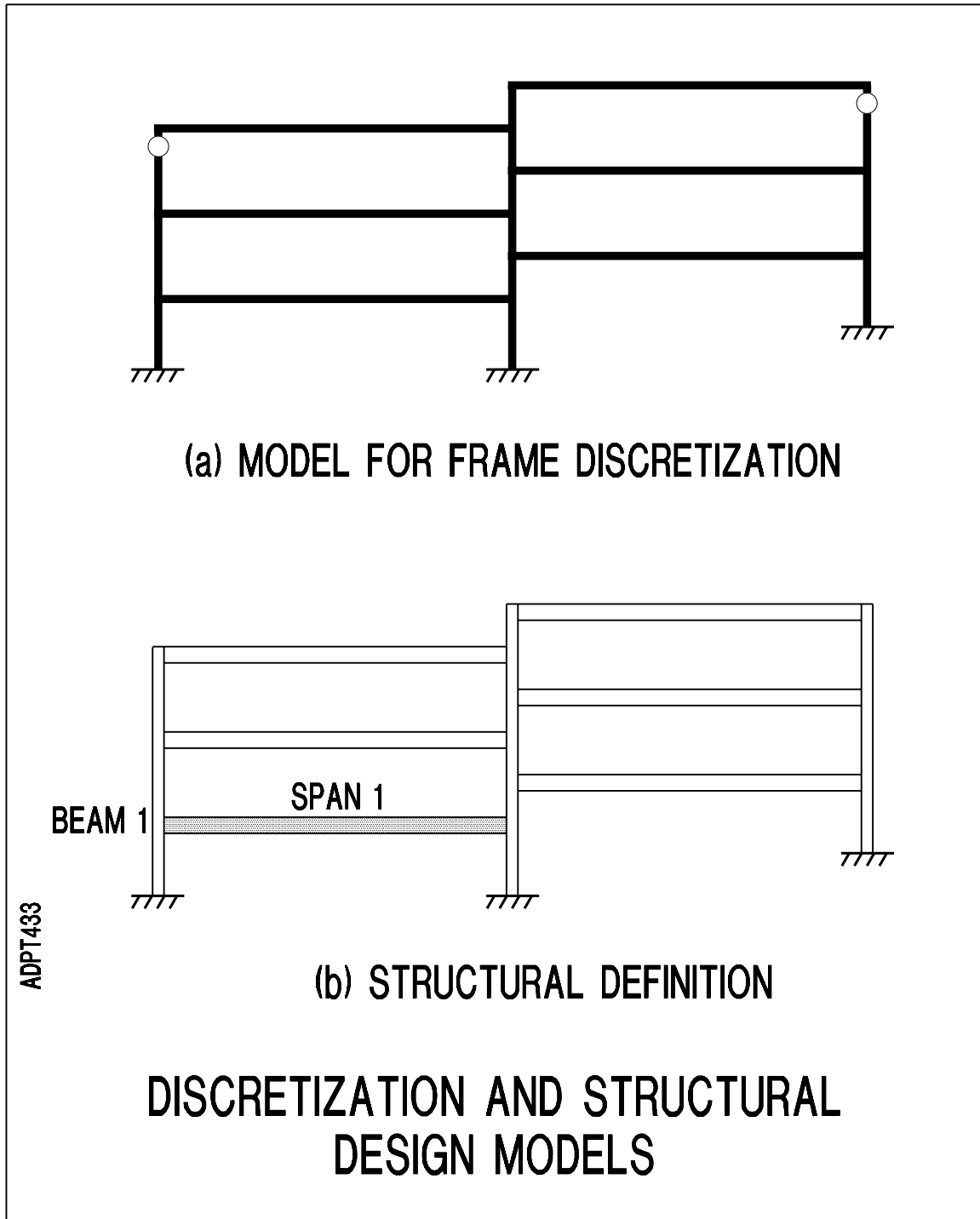


FIGURE 1.3-1

This Page Left Intentionally

BLANK

## 2.1 GENERAL

This chapter describes the three primary features of ADAPT-GEN. These are:

1. Definition of a frame as a structure in terms of spans and columns;
2. Loading pattern, skipping of loads;
3. Load combinations and envelope extraction;
4. Determination of hyperstatic (secondary) actions and displacements due to prestressing.

## 2.2 DEFINITION OF STRUCTURE

A part, or the entire, frame being analyzed can be defined as a structure concurrent with frame's discretization for the finite element analysis. The concurrent definition of the frame as a structure does neither change the solution procedure, nor does it affect the results. Rather, it groups the input and output data in a more recognizable and manageable format for the user. Also, it can reduce the volume of unwanted output. A finite element analysis, of which ADAPT-GEN is one, generally requires a lengthy input and results in a voluminous output. A pre- and post-processor, such as ADAPT-GEN, can significantly reduce the input generation, and tailor the output to critical information of interest to the user.

Internally, the frame is treated in terms of nodes and elements of the finite element discretization. Externally, at the level of user interface, the frame can be viewed consisting of column grids, levels, columns, and spans. These are defined and selected by the user.

A general hypothetical plane frame is selected for the illustration of the concept (**Figure 2.2-1**). The frame is viewed as a matrix of intersecting lines. The lines with vertical or near vertical orientation are referred to as GRID lines. The frame shown has five grid lines, marked: A, B, C, D, and E. The selection of identification letter or name for the grid lines is arbitrary. The somewhat horizontal lines are called LEVELS. The frame has a total of seven levels. The levels are identified as: ABL\_26, LEV\_B, LEV\_3, 23L, MID\_LEV, DECK, ROOF. Again, the identification is arbitrary, within the limits of the name selection outlined in Chapter 3.

The grids are considered to consist of columns. A column is a region generally intercepted by two adjacent levels. For example, GRID C has 6 natural columns. The first column is between the support and level ABL\_26. The last column is the region above the ROOF level. If need be, a column can be broken into several parts, each regarded as an individual column, in which case the sub-columns would be not intercepted by two adjacent levels.

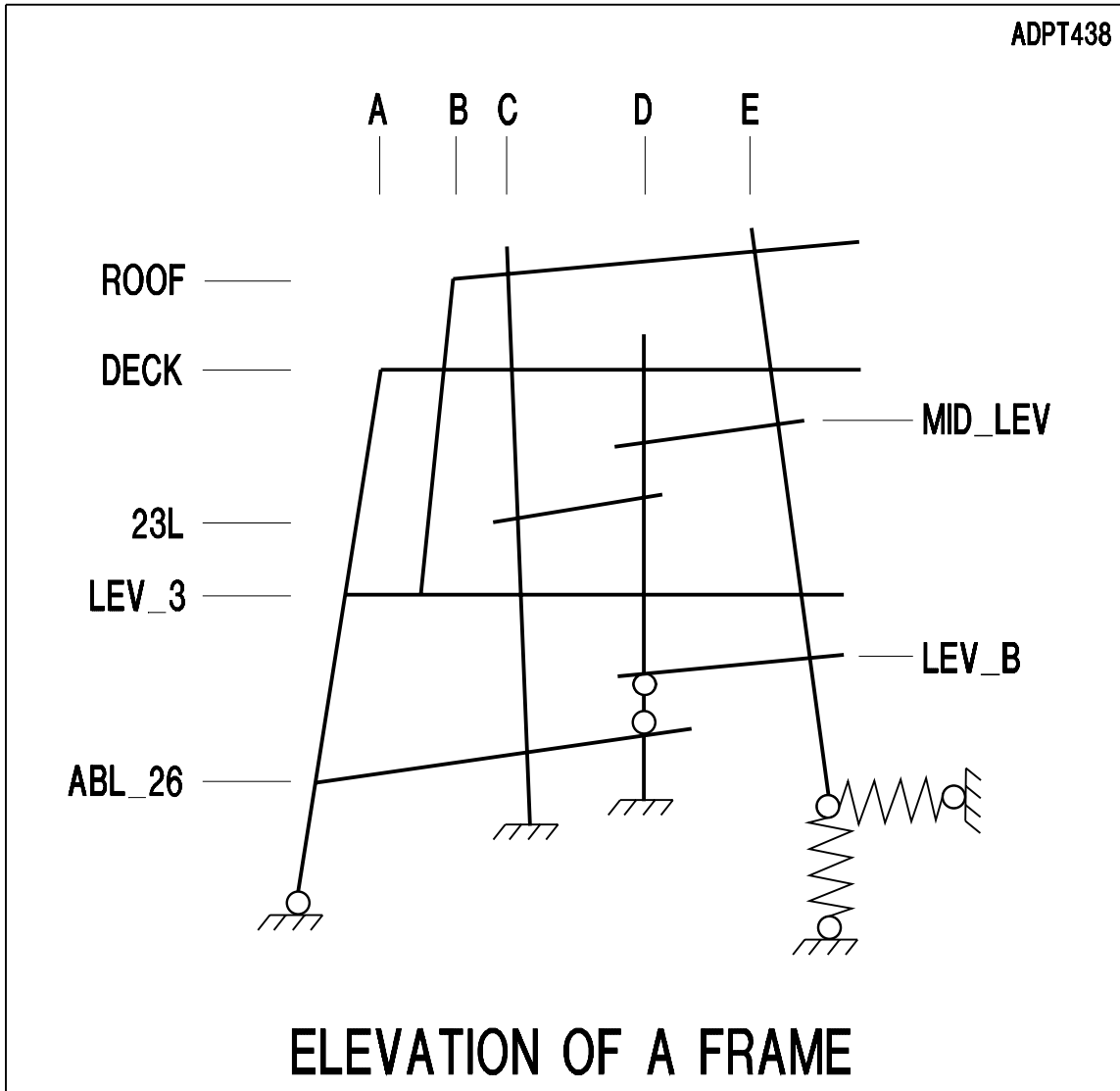


FIGURE 2.2-1

Spans are regions of levels generally bounded by two adjacent grid lines. Cantilevers are also considered as spans. For example, level LEV\_3 has five spans, including the cantilever beyond grid line E.

In summary, a breakdown option for the frame shown in **Figure 2.2-1** is:

- 5 Grids
- 7 Levels
- Level ABL\_26 has 3 spans
- Level LEV\_B has 3 spans
- .....
- .....

In the preceding example, the entire frame is subdivided into grids and levels. If the focus of the analysis is on a particular region of the frame, only that region of the frame need be defined as a *structure*. By limiting the definition to only one region of the frame, the input of loading and the output can be trimmed, with focus on the region of interest.

Commonly, a span, or a column, is regarded as a region between two supports, or a cantilever. But, in ADAPT-GEN this limitation is not adhered to. A span is a region in the frame which is of particular interest to the user. For example, the single span simple support beam of **Figure 2.2-2(a)** is regarded as consisting of two levels, namely: LEV\_1 and LEV\_2. Level LEV\_1 is viewed to have two regions, region SPAN 1 and region SPAN 2. The subdivision of a frame into levels and spans is not related to its discretization into elements and nodes for the finite element analysis. The only constraint is that, spans, or columns must start and terminate at a node (**Figure 2.2-2(b)**). For example, SPAN 2 of LEV\_1 starts at node 5 and terminates at node 9.

### 2.3 LOAD PATTERN; ENVELOPE

The loading can be patterned (skipped) to obtain maxima and minima of displacements and actions. In skipping the load, each span is considered either *loaded*, or *unloaded*. The example shown in **Figure 2.3-1** illustrates the concept. The example shows two beams, a three-span beam (**Figure 2.3-1(a)**) and a single span beam (**Figure 2.3-1(b)**). In terminology of ADAPT-GEN both beams may be defined to consist of three SPANS, or three regions. A possible pattern loading is defined in **Figures 2.3-1(d)** through, **(f)**, where each SPAN is loaded while the others are unloaded. This can simulate a moving load. The three pattern loading components result in a bending moment envelope in the continuous beam such as shown in **Figure 2.3-1(g)** or, **(h)**, for the single beam and the three-span beams respectively.

Note that in the context of pattern loading, the primary significance of defining a SPAN in ADAPT-GEN is to delineate the region over which loading either is active, or is absent.

The outcome of a pattern loading is an envelope, such as shown in **Figure 2.3-1(g)** and **(h)**. Individual loading cases are computed, saved and can be accessed by the interested user, but are not included as part of the regular output.

For determination of actions due to moving loads in bridge structures, it is recommended to use the “moving load” module of ADAPT-ABI.

### 2.4 LOAD COMBINATION

The software has the capability of combining the results of individual load cases, each with a factor defined by the user.

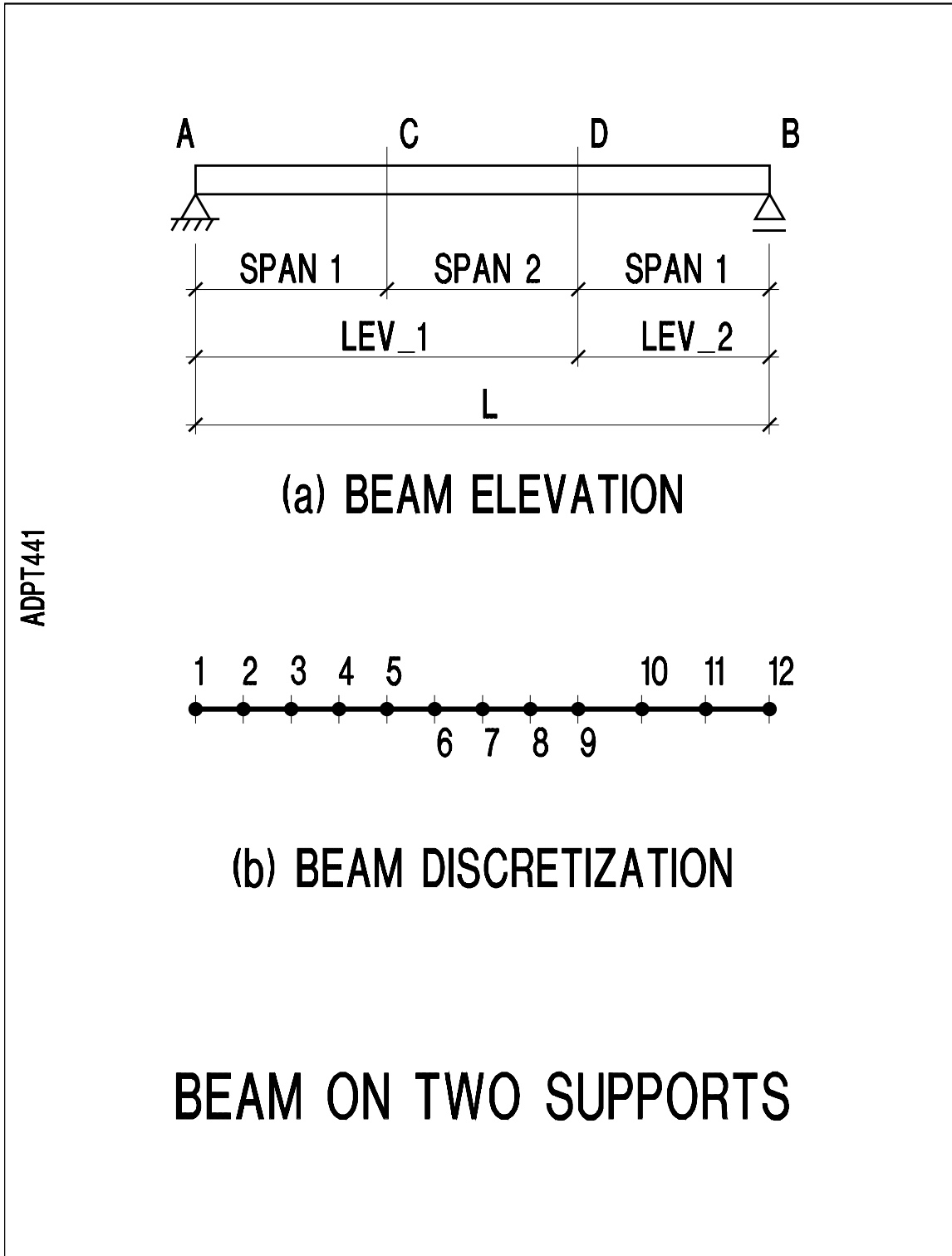


FIGURE 2.2-2

2.5 HYPERSTATIC ACTIONS DUE TO PRESTRESSING

ADAPT-GEN extracts the hyperstatic (secondary) actions and displacements due to prestressing. The hyperstatic actions are treated as a loading case. They can be combined with other loading cases with a multiplying factor defined by the user.

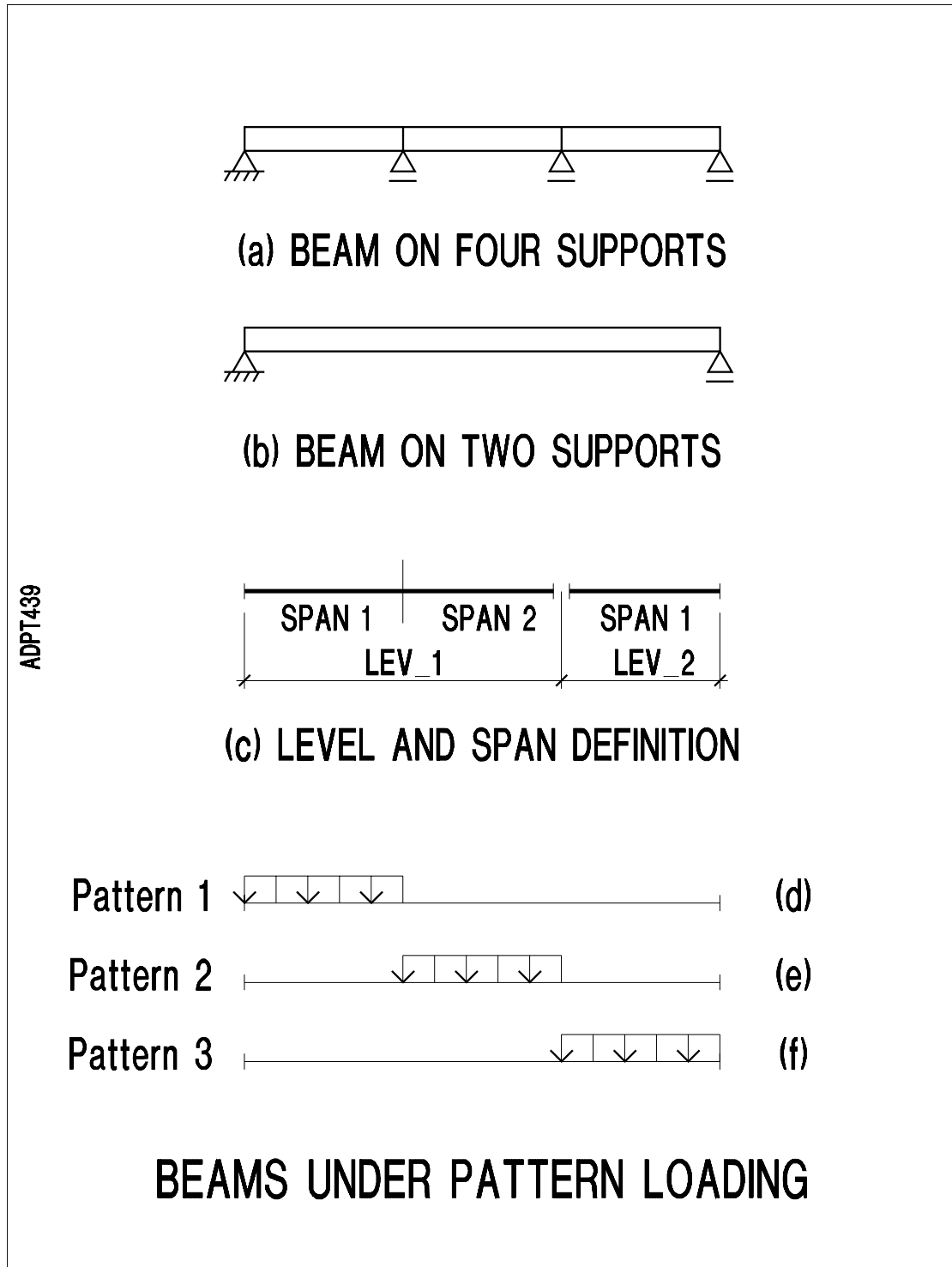


FIGURE 2.3-1

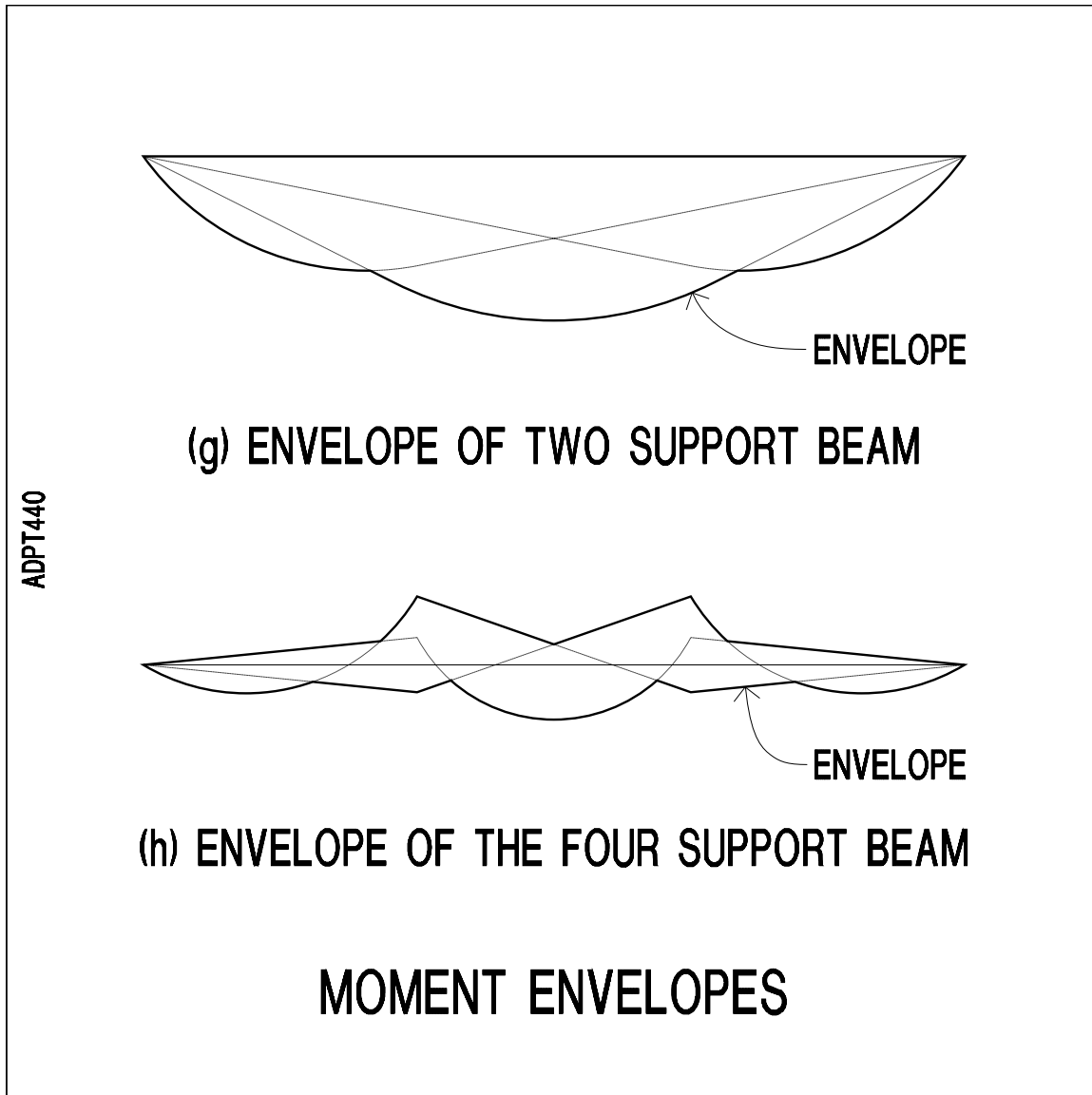


FIGURE 2.3-1

## 2.6 PRESTRESSING ACTIONS

The program determines the actions and displacements in the structure due to the prestressing forces. In doing so, the program accounts for the fact that prestressing may have been applied at different times and at different stages of construction. Further, there may have been stress losses, due to other reasons apart from stress relaxation. In other words, the actions due to prestressing are determined with full recognition and implementation of the loading and construction history of the structure.

## 2.7 LONG-TERM LOSSES

The solutions from the basic module of the program ABI (Basic) allow for and include the influence of the long-term effects using a rigorous formulation. As a result, in principle, there is no need to



calculate the long-term effects separately. However, some bridge design codes, such as AASHTO require that these effects, also can be estimated independently and combined with other load effects using a different load factor for each. For this reason, and also to serve the academia, ADAPT-Gen extracts the actions and displacements due to long term effects of creep and shrinkage as an independent set of results. Losses in prestressing are lumped into the prestressing solution. Influence of aging of concrete is included in the actions due to dead load of the structure.

This Page Left Intentionally

BLANK

## 3.1 OVERVIEW

ADAPT-GEN software is the extension of the special purpose software ADAPT-ABI. It is tailored to structural design of frames of both simple and complex geometry. Details of input generation of ADAPT-ABI are given in ABI's manual. This manual is a companion to ADAPT-ABI manual. In this manual, the input description is limited to commands which are specific to data generation and ADAPT-GEN, with minimal comments on other aspects of the software. For a complete analysis, the two manuals must be used together.

### 3.1.1 Capabilities

This module features three principal capabilities beyond the basic module, ADAPT-ABI. One is the visualization and treatment of the frame in terms of a *structure*, consisting of levels, spans, column lines and column. Likewise, it offers the option of defining the loading in terms of spans and columns, as opposed to nodes and elements. The second feature is the capability of *load combination* and extraction of envelopes for maxima of displacement and actions. The third feature is the extraction of prestressing, hyperstatic actions due to prestressing, and long-term effects each as a distinct solution, either to be used alone or in combination with other solutions.

Commands related to the definition of the structure are grouped together under the command `STRUCTURE DEFINITION`. The commands for loading associated to load combination and extraction of envelopes are grouped together after `CAPTURE STATUS` command.

A sample input in Section 3.3 illustrates the use of the commands specific to this module. Where elucidating, the commands of this module are printed in bold font, in order to differentiate them from the commands of the basic module.

### 3.1.2 Preparation of Input Data

As in the basic module, input data is fed to the program through an input (ASCII) file. All input takes the form of commands consisting of one or more keywords, followed by optional numerical data required for the execution of the command. The actions and the data requirements of the commands are summarized in the following pages. All keywords must be *typed in full* as they appear in this manual. Mis-spelled words result in error. In most cases the errors are detected by the software's data verification module.

All numerical data are entered in the following free field form:

$n_1, n_2, n_3, \dots \quad A=a_1, a_2, a_3, \dots \quad B=b_1, b_2, b_3, \dots$

where,  $n_i$ ,  $a_i$ , and,  $b_i$ , represent input data and the character pairs, A=, and, B=, are identifiers, specified in the input manual, for the data list which follows. Items in a data list must be separated by a single comma or by one or more blanks. If a numerical data list without identification, such as  $n_1, n_2, n_3, \dots$  above, is required it must

be located as the first data list on the line. A data list of the form  $B=b_1, b_2, b_3, \dots$  may be in any order or location on the line.

Simple arithmetical statements are possible when entering floating point real numbers. This can clarify the meanings of some data. For example, the following forms of data can be entered:

$$E=29600*144 \quad C=200*12+3.5, \quad 400/12, \quad 10+20/5-2$$

These arithmetical statements are evaluated from left to right without operator hierarchy. The statement  $10+20/5-2$  is evaluated as  $((10+20)/5)-2$ . Blank spaces within arithmetical statements are not allowed.

When a loading case must be multiplied by a coefficient, it is essential to use the multiplication symbol,  $*$ .

$$1.4*SDL+1.7*LL+1*PH$$

This example means combine a loading case consisting of 1.4 times the,  $SDL$ , loading, 1.7 times the,  $LL$ , loading and 1 times the loading designated by,  $PH$  (hyperstatic actions due to prestressing).

The command interpreter in the program recognizes several special characters used to delimit multiple command lines on one physical line, continuation of a command line on the next physical line, and comments to be ignored by the command interpreter.

- ! The exclamation mark is a special character used to delimit several command lines provided on one physical line. The data to the right of an exclamation point are not considered to be part of the current command line, and are instead considered as the next command line. Any number of command lines may be input on one physical input line by separating them with this character.
- \ The backslash is a special character used for continuation of a command on the next physical line. All data to the right of the backslash are ignored, and the following input line is interpreted as a continuation of the first line. This option allows a maximum of 160 characters to be entered as one line of data.
- ; The semicolon is a special character used to delimit comments in the input stream. All characters to the right of the semicolon up to the end of the line are ignored by the program. If the semicolon is located in the first column in a command line, the entire command line is ignored.

## 3.2 COMMAND SUMMARY

### 3.2.1 Natural Sequence of Commands

The following represents the sequence of commands for most common problems. The commands highlighted are specific to ADAPT-GEN, and will be expounded in this Chapter. The remainder of the commands are explained in ADAPT-ABI manual.

```
START
TITLE
UNITS
CONCRETE PARAMETERS

MESH INPUT
  NODES

  SEQUENCE

  CONCRETE PROPERTIES
  MILD STEEL PROPERTIES
  SECTION PROPERTIES
  OFFSET DATA
  ELEMENTS
    FRAME
    SPRINGS

  PRESTRESSING STEEL
    TENDON GEOMETRY

  TRAVELERS
MESH COMPLETE

STRUCTURE DEFINITION
  GRIDS n LEVELS m
    GRID #####
      COLUMN
    LEVEL #####
      SPAN
STRUCTURE COMPLETE

SET

CHANGE STRUCTURE
  BUILD (Frame Element)
  RESTRAINTS

  REMOVE (Element)
  STRESS (Tendon)
  DE-STRESS (Tendon)
  MOVE (Traveler)
CHANGE COMPLETE
```

```
CAMBER INITIAL
LOADING

SOLVE
OUTPUT
CAMBER DESIGN

CAPTURE STATUS
  LOADING #####
    DEAD LOAD
  SOLVE
  OUTPUT

  LOADING #####
    PRESTRESSING
  SOLVE
  OUTPUT

  LOADING #####
    LONG TERM
  SOLVE
  OUTPUT

  LOADING #####
    GENERIC
    STRUCTURE
    LEVEL
    GRID
  SOLVE
  OUTPUT

  LOADING #####
    PATTERN
    LEVEL
    GRID
    SKIP
    LEVEL
    GRID
    SKIP
    .....
    .....
  SOLVE
  OUTPUT

COMBINATIONS
  COMBINATION #####
OUTPUT
  ENVELOPE #####
  ?, ?, ?
OUTPUT

STOP
```

### 3.2.2 Summary of Commands

The following brief descriptions of each command and its input data requirements are intended as a summary of the software's command syntax and as a quick reference guide for the experienced user. The highlighted commands are specific to ADAPT-GEN. Detailed descriptions of the highlighted commands and the meaning of all data are provided further on, in this manual. Sample input for a simple frame is included in this chapter for illustration.

COMMAND SUMMARY	DESCRIPTION
<b>Problem Initialization Input:</b>	
START	Start interpreting new input data
START R=1	Start interpreting data for a program re-run
TITLE N=?	Print N lines of title
UNITS U=?	Set input/output units
ACTIVATE EXTRACT	Extract dead load, prestressing, and long-term effects.
CONCRETE PARAMETERS N=?	Number of concrete materials used followed by the modeling scheme used for concrete's creep and shrinkage
<b>Frame Definition Input:</b>	
MESH INPUT	Start definition of structural system
NODES	Input node coordinates
SEQUENCE	Optional input for sequence of degrees of freedom for solution optimization
CONCRETE PROPERTIES	Input concrete material properties; strength; time dependent parameters; material model type
MILD STEEL PROPERTIES	Input mild steel properties and its amount in concrete used
SECTION PROPERTIES OFFSET DATA	Input cross section properties or geometry Input distance between natural frame nodes and analysis nodes.
ELEMENTS	Input total number of elements in the complete structure
FRAME	Input elements of the structural concrete frame
SPRINGS	Input springs and their stiffnesses
PRESTRESSING STEEL	Input prestressing steel properties
TENDON GEOMETRY	Define tendon numbers and geometry of each in the structure

TRAVELERS	Define traveling formwork
MESH COMPLETE	Signals end of structure definition
<b>Structure Definition Input:</b>	
STRUCTURE DEFINITION	Signals start of structure definition
GRIDS n LEVELS m	There are, n, column lines and, m, levels
GRID #####	Start of definition of column line #####
COLUMN n	Start of definition of column, n
LEVEL #####	Start of definition of level #####
SPAN n	Start of definition of span, n
STRUCTURE COMPLETE	End of structure definition
<b>Set Ambient Conditions:</b>	
SET	Define ambient conditions for the solution; such as day, temperature and solution technique parameters
<b>Construction Operations:</b>	
CHANGE FRAME	Start construction operations for this step
BUILD	Install frame and spring elements
RESTRAINTS	Change or define boundary conditions
(blank line in input)	
REMOVE	Remove frame, spring, slave and hinge elements
STRESS	Stress tendons
DE-STRESS	Stress again (remove) previously stressed tendons
MOVE	Move traveler to new destination nodes
CHANGE COMPLETE	End construction operations for this step
<b>External Loading Command:</b>	
LOADING	Specify load increment which can consist of nodal loads, nodal displacements, and element temperatures
<b>Initialize Camber Computation:</b>	
CAMBER INITIAL	Assume current displacements as datum for camber computations
<b>Solution Command:</b>	
SOLVE	Solve the current structure



**Post Construction Command:**

CAPTURE STATUS

Indicates completion of construction and termination of time dependency

**Post Construction Loading:**

LOADING #####

Header for Loading case #####

DEAD LOAD

Obtains solution due to dead loading only

PRESTRESSING

Obtains solution due to prestressing and hyperstatic actions

LONG TERM

Obtains solution due to long-term effects of creep and shrinkage

GENERIC

Loading specified in terms of nodes and elements

STRUCTURE

Loading specified in terms of spans and columns

LEVEL #####

Header for loading specification on level #####

GRID #####

Header for loading specification on column line #####

PATTERN

Indicates that following load is to be skipped

SKIP

Indicates start of skip pattern table

SOLVE

Solve for current loading

OUTPUT

Output the solution for graphical view and tabulated report

**Load Combination Commands:**

COMBINATIONS

Start of combination commands

COMBINATION n

Header for combination, n

ENVELOPE #####

Header and identification for envelope subcommand

**Output Commands:**

OUTPUT

Output total structure response

**Problem Termination Command:**

STOP

Stop execution, save for restart

**Output Commands:**

OUTPUT

Output total structure response

CAMBER DESIGN

Output displacements to be adjusted for camber control

### 3.3 SAMPLE INPUT FOR A NON-PRESTRESSED FRAME

#### 3.3.1 Description of the Frame

The following is a sample input for a simple non-prestressed frame (**Figure 3.3-1**) with uniform rectangular cross-section subject to the following loading.

- A. Selfweight
- B. Uniform live load on the spans  
 $q_1 = 10 \text{ lb/in}$   
 $q_2 = 8 \text{ lb/in}$
- C. Concentrated load at midpoint of level 3rd\_lv  
 $P_1 = 10,000 \text{ lb}$
- D. Horizontal loads at column/span joints as shown  
 $P_2 = 10,000 \text{ lb}$   
 $P_3 = 5,000 \text{ lb}$

To further simplify the example, creep, shrinkage, and aging of concrete are suppressed. Also, prestressing is excluded. The purpose of the example is to illustrate the definition of the *structure* and *load combinations*. Comprehensive examples, illustrating other features, are given in Chapter 5 of this manual.

The example defines the frame as having two levels, and two column grids, each consisting of two columns.

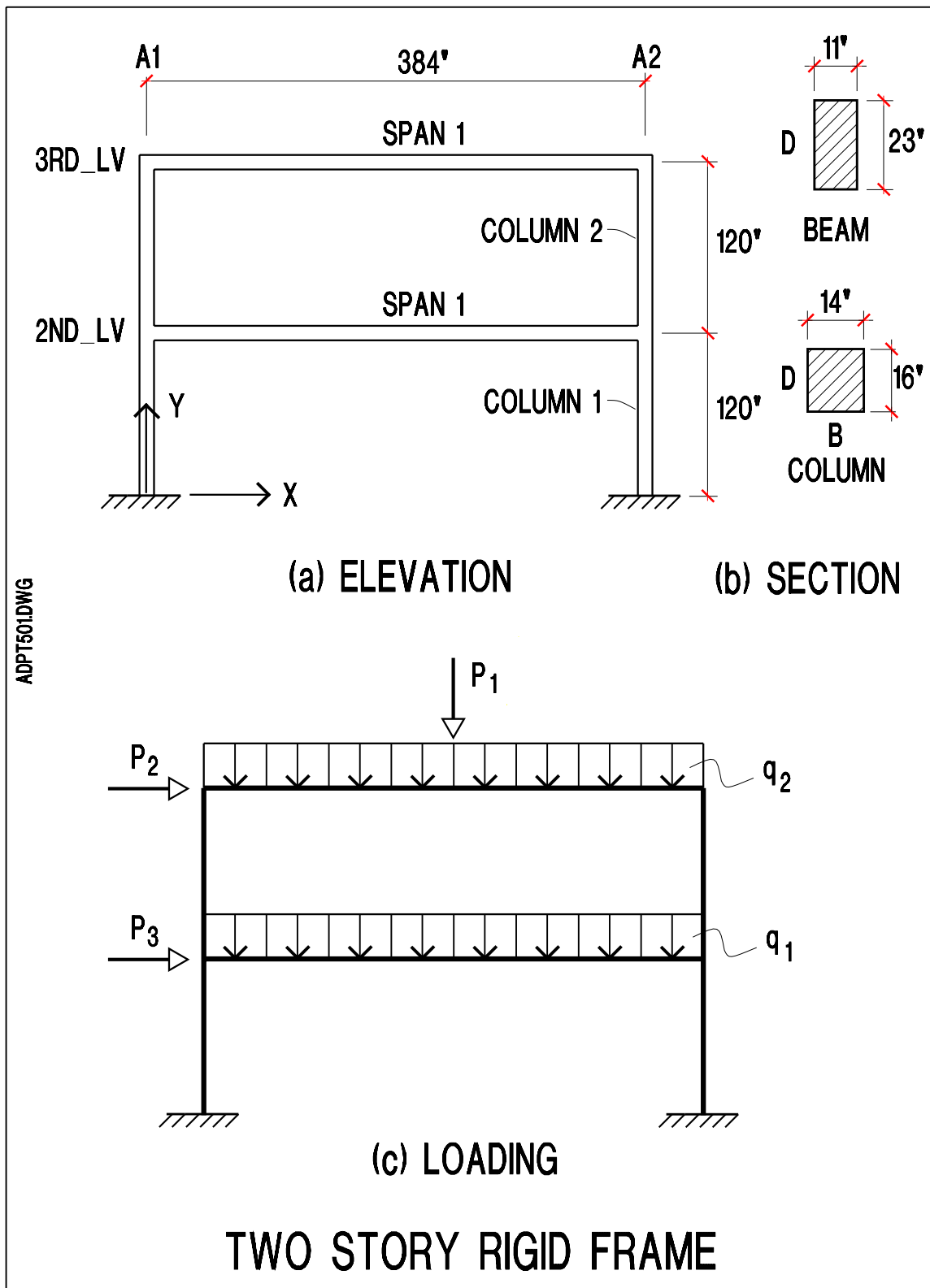


FIGURE 3.3-1

The loading cases are:

- A. Selfweight (designated as Dead)
- B. Live loading (designated as Live) consisting of the uniform loads on spans and the concentrated loading  $P_1$ , skipped to maximize actions.
- B. Lateral loads  $P_2$ , and  $P_3$ , (designated as Lat)

The Load combinations are:

- A. (dead load) + 1.7\*(maxima of live loading)
- B. (dead load) + 1.43\*(lateral loading)
- C. (dead load) + 1.43\*(maxima of live load) + 1.43\*(lateral)

The finite element discretization of the frame is shown in **Figure 3.3-2**.

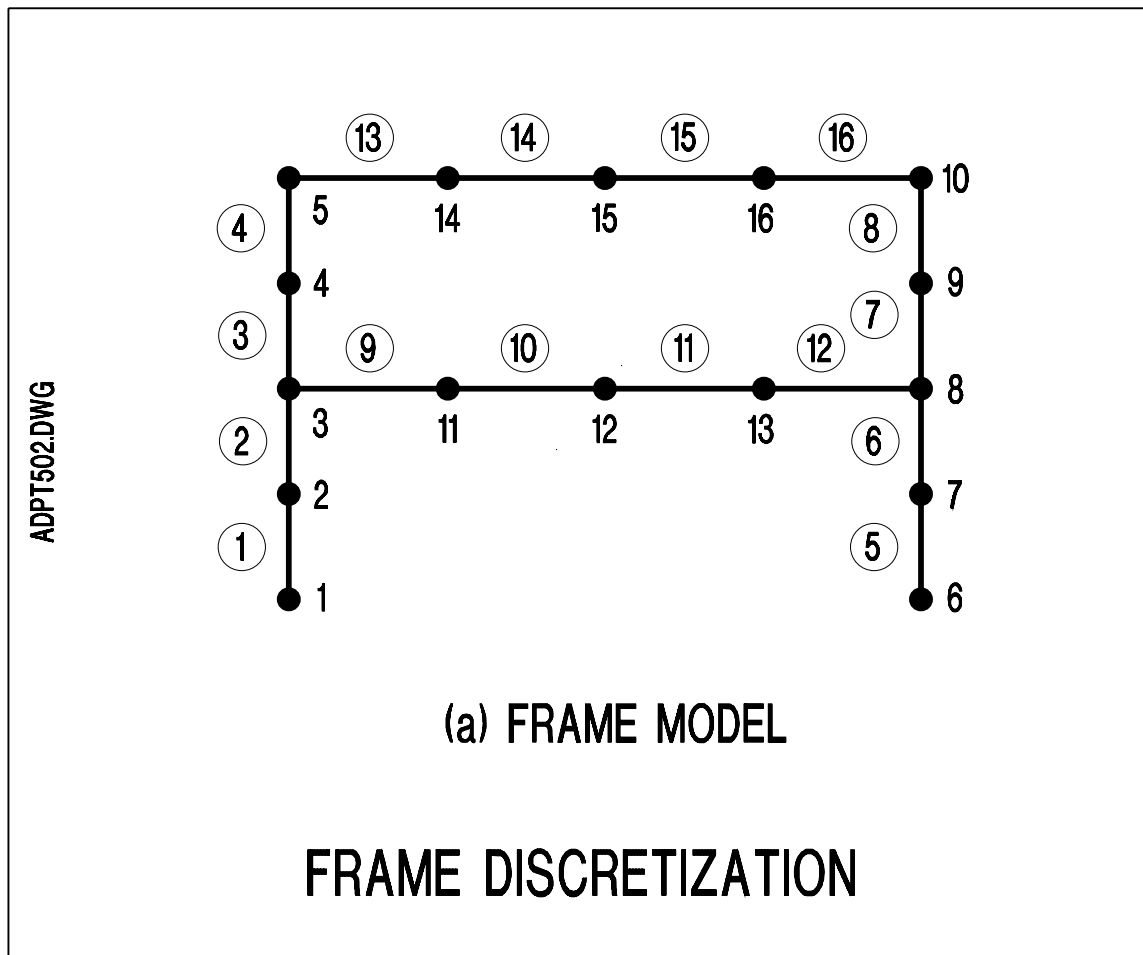


FIGURE 3.3-2

### 3.3.2 Input File

```

;=====
;   ADAPT-BRIDGE-INCREMENTAL  ABI  SOFTWARE MANUAL EXAMPLES
;=====
;   name of this file: Gen-Ex31.inp

START
TITLE N=2
      TWO STORY FRAME WITH CONCENTRATED
      AND LATERAL LOADS APPLIED

UNITS U=USA
ACTIVATE EXTRACT
CONCRETE PARAMETERS N=1           ;SUPPRESS AGING EFFECT
      1 M=ACI  a=0  b=1

MESH INPUT
  NODES N=16
    1 X=0  Y=0  ! 5 X=0  Y=240 G=1,5
    6 X=384 Y=0  ! 10 X=384 Y=240 G=6,10
    11 X=96  Y=120 ! 13 X=288 Y=120 G=11,13
    14 X=96  Y=240 ! 16 X=288 Y=240 G=14,16

CONCRETE PROPERTIES N=1           ; NO CREEP, NO SHRINKAGE
  1 Fpc=5000 Cr=0.0 Sh=0.0 W=150/1728
MILD STEEL PROPERTIES N=1
  1 Es=29000000 P=0.00
SECTION PROPERTIES N=2
  1 D=16 B=14                       ;COLUMN PROPERTIES
  2 D=23 B=11                       ;BEAM PROPERTIES
ELEMENTS N=16
  FRAME N=16
    1, 1, 2 C=1 X=1 St=1 Day=1 G=1,4,1,1,1
    5, 6, 7 C=1 X=1 St=1 Day=1 G=5,8,1,1,1
    9, 3,11 C=1 X=2 St=1 Day=1
    12,13, 8 C=1 X=2 St=1 Day=1
    13, 5,14 C=1 X=2 St=1 Day=1
    16,16,10 C=1 X=2 St=1 Day=1
    10,11,12 C=1 X=2 St=1 Day=1 G=10,11,1,1,1
    14,14,15 C=1 X=2 St=1 Day=1 G=14,15,1,1,1

MESH COMPLETE

;Defining the frame structure, specifying columns and spans
STRUCTURE DEFINITION
  GRIDS 2 LEVELS 2
    GRID A1
      COLUMN 1 LIST=1,2
      COLUMN 2 E=3,4,1
    GRID A2
      COLUMN 1 E=5,6,1
      COLUMN 2 LIST=7,8
  LEVEL 2ndLV
    SPAN 1 E=9,12,1

```

```

    LEVEL 3rdLV
      SPAN 1 LIST=13,14,15,16
STRUCTURE COMPLETE

;Start building the frame structure elements
CHANGE STRUCTURE
  BUILD N=1,16,1
  RESTRAINTS
    1 R=1,1,1
    6 R=1,1,1

CHANGE COMPLETE
SOLVE Day=28 !OUTPUT
SOLVE Day=100 !OUTPUT

;Start of structure loading
CAPTURE STATUS
  LOADING Dead ;selfweight
  DEAD LOAD

SOLVE
OUTPUT

LOADING Live ;concentrated and uniform live load
PATTERN
  LEVEL 2ndLV
    SPAN 1
    UNIFORM F=0,-10
  LEVEL 3rdLV
    SPAN 1
    UNIFORM F=0,-8
    CONCENTRATED F=0,-10000,0 AT N=15
SKIP
  LEVEL 2ndLV SPANS 1
  LEVEL 3rdLV SPANS 0
SKIP
  LEVEL 2ndLV SPANS 0
  LEVEL 3rdLV SPANS 1
SKIP
  LEVEL 2ndLV SPANS 1
  LEVEL 3rdLV SPANS 1

SOLVE
OUTPUT

LOADING Lat ;concentrated lateral live load
STRUCTURE
  GRID A1
    COLUMN 2
    CONCENTRATED F=10000,0,0 AT N=5
  GRID A1
    COLUMN 1
    CONCENTRATED F= 5000,0,0 AT N=3

SOLVE
OUTPUT

```

```

COMBINATIONS
  COMBINATION A ;dead + live load
    1.4*Dead+1.7*Live
OUTPUT          ;output here is optional

  COMBINATION B ;dead + lateral
    0.9*Dead+1.43*Lat
  COMBINATION C ;dead + live + lateral
    1.0*Dead+1.43*Live+1.43*Lat
OUTPUT
  ENVELOPE D
    A,B,C ;envelop load combinations A,B, & C
OUTPUT

STOP

```

### 3.4 DETAILED DESCRIPTION OF COMMAND SYNTAX

The detailed syntax of each command and a description of its use, actions and output are described below. The commands are organized alphabetically. An approximate order in which the commands normally will be found in a typical input file is given in Section 3.2.1. Most commands are optional, and must not be used unless required for the structure under analysis (i.e. a command such as, `GENERIC`, must not be used if no loading is defined using the `GENERIC` format).

The optional parts of the *syntax line* are enclosed in square brackets [ . . . ]. For example, [ALL] when used would refer to all the relevant spans or columns; if left out, the command refers to the spans mentioned on the command line only.

In the *Explanation* part of the command, default values for input quantities, where implemented in the program, are indicated by [ ? ], with the value of the default entered within the square brackets. [pv] indicates a default value assumed to be equal to the previous value entered.

It is recommended that the first time user follows the order of commands as described in the natural sequence (Section 3.2.1), starting with the description of command `STRUCTURE`, in conjunction with **Figure 3.3-2**. A familiar user would use this section as a reference.

#### CAPTURE STATUS

**Syntax:**        `CAPTURE STATUS`

**Explanation:** This command has no arguments. It signifies the termination of the construction stage and the time dependent analysis. After this command, the frame behaves as a linear elastic structure. Loading is no longer incremental (accumulative). Displacements and actions due to each loading case will be computed based on the geometry and stiffness at the time `CAPTURE STATUS` command is specified. Solutions

of different loading cases can be combined to obtain envelopes or factored actions.

## COLUMN

**Syntax:** COLUMN i E=l1,l2,inc  
 or  
 COLUMN i LIST=l1,l2,l3,...,ln

where,

i	= Column number
l1	= Frame element number of first element in a series of elements which define the column
l2	= Element number of last element in series
inc	= Element number increment
l1,l2,l3,..,ln	= Frame element number of first element in a series of elements which define the column, followed by a sequential list of other elements, l2, l3, until the last element of series, ln.

**Explanation:** Each column in a GRID is defined by one subcommand line starting with COLUMN i. The column number, i, is a sequential integer designating the column being defined from the list of columns in a GRID. For example, if a grid has 5 columns, there must be five COLUMN subcommands, each defining one of the 5 columns of the grid.

**Example:** (i) Refer to Figure 3.3-3(d)

```
GRID C3
  COLUMN 1 ....
  COLUMN 2 ....
  COLUMN 3 ....
```

Note that the column not highlighted as part of the structure is not listed.

(ii) Refer to Figure 3.3-3(c)

```
GRID C2
  COLUMN 1 List=....
  COLUMN 2 List=....
```



(iii) Refer to **Figure 3.3-3(b)**

```

GRID C2
      COLUMN 1  E=....
      COLUMN 2  E=....
      COLUMN 3  List=....

```

## COMBINATION

**Syntax:** COMBINATION #####

```

[k1*load1] + [k2*load2] + [k3*combination1]
+...[ki*PH]

```

where,

##### = An up to six character alphanumeric string identifying the combination

k1,k2,... = Coefficients to be multiplied by the loading cases and combinations load1, load2,..., combination1,...

PH = Hyperstatic actions due to prestressing (secondary actions)

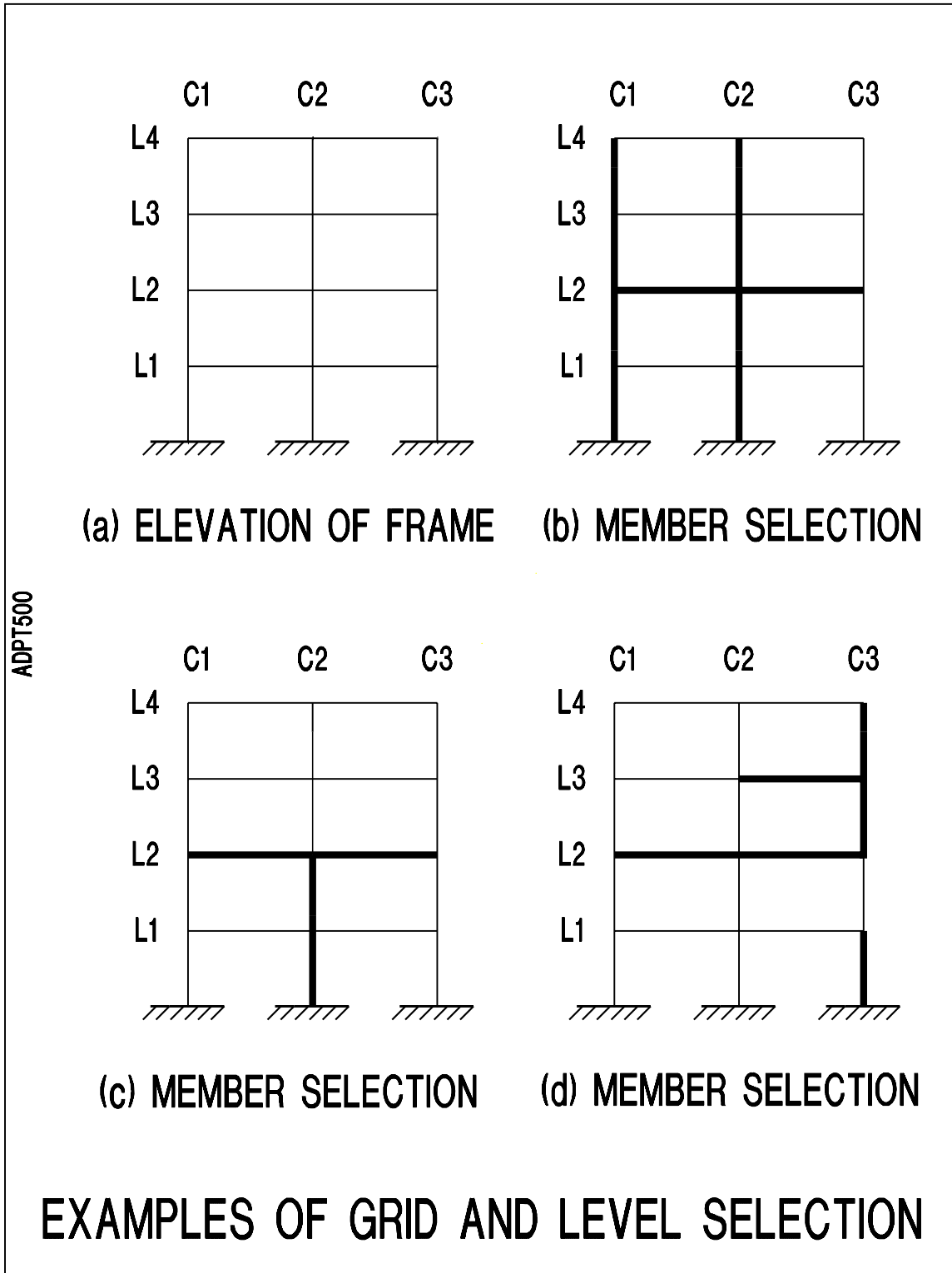
**Explanation:** The combination identification, #####, will be printed at the top of its relating output.

The loading cases must all have been previously defined. The addition sign, +, can be either a positive sign, +, or a negative sign, -, for subtraction.

The optional loading [PH] refers to the hyperstatic (secondary) action of prestressing.

The prestressing load will be referred to by its user defined loading identification.

If any of the components of the combination requested has an envelope, that is to say it is obtained through a SKIP command, the combination outcome would also be an envelope.



ADPT500

FIGURE 3.3-3

## COMBINATIONS

**Syntax:** COMBINATIONS

**Explanation:** This subcommand has no argument of its own. It signifies the end of loading conditions and the start of combination of previously obtained solutions under LOADING.

The COMBINATIONS command is followed by a series of COMBINATION and ENVELOPE subcommands, each of which describe one loading combination or envelope. Refer to COMBINATION and ENVELOPE command for their syntax. An OUTPUT command following a COMBINATION or ENVELOPE subcommand will result in a print out of that combination in the user's file, as well as a display of its results. The OUTPUT subcommand is optional. If the OUTPUT subcommand is used, the software generates a set of internal output data which would allow the combination/envelope to be displayed on the screen, and to be integrated as part of subsequent COMBINATION or ENVELOPE commands.

**Example:**

```

COMBINATIONS
    COMBINATION #####
                .....
OUTPUT
    COMBINATION #####
                .....
;no output will be generated
    ENVELOPE #####
                ?,?,?
OUTPUT

```

## DEAD LOAD

**Syntax:** DEAD LOAD

**Explanation:** This subcommand has no argument. It generates a solution for the entire loading on the frame at the time CAPTURE STATUS was called, minus PRESTRESSING, if any, and minus LONG TERM effects. The loading prior to capture status on the frame can be due to selfweight and other loading, such as superimposed loading. The rationale behind this subcommand rests on the following scenario. A frame is built; prestressing is added optionally; superimposed loading, such as topping is added; lapse of time allows for creep and shrinkage to take place; the frame enters the phase of its service life - the steady state. At this time, through CAPTURE STATUS command the time is frozen. Subsequently, the frame stresses and deformation are investigated due to a combination of loading cases. One such loading case is the steady state itself, which is dead load plus prestressing, and plus long-

term effects if any. The investigation proceeds by obtaining separate solutions for *dead loading*, *prestressing* and *long-term* effects, and then combining them. Solutions for the loading components are extracted separately, since they will enter the combinations with different factors.

## ENVELOPE

**Syntax:** ENVELOPE #####  
 [LOADING<sub>i</sub>], [ENVELOPE<sub>i</sub>], [COMBINATION<sub>i</sub>]

where,

##### = Any up-to-6-character alphanumeric string for identification of the current envelope  
 LOADING<sub>i</sub> = The identification string of a previously defined loading case  
 ENVELOPE<sub>i</sub> = The identification string of a previously defined envelope  
 COMBINATION<sub>i</sub> = The identification string of a previously defined combination

**Explanation:** The identification string, #####, used for each envelope must be unique, that is to say, it must not have been used elsewhere in data input. The software initially stores the outcome of the envelope under a file name which incorporates the string, #####. The output, if requested, uses the same string as its header.

The ENVELOPE command scans the results of the components indicated in the syntax, for the nodes and elements of the structure, and selects both the maximum and minimum values of displacements and actions. The command groups the maxima and minima selected and stores them in a file for future use by the user. If the ENVELOPE command is followed by an OUTPUT command, the results will be printed as part of the output of the problem, and will also be available for graphical display. Otherwise, the solution remains in the background in an internal file until it is called to be included in a future COMBINATION, or ENVELOPE command.

It is imperative that all components listed after the ENVELOPE command have been previously defined.

The ENVELOPE command compares the results of the components listed and selects the largest and smallest of each item. Unlike the COMBINATIONS command, ENVELOPE command does not scale, nor does it add the arguments listed.

```

Example:      COMBINATIONS
                  COMBINATION A
                   1.4*DEAD+1.7*LIVE
                  COMBINATION B
                   0.9*DEAD+1.7*EQ
                  COMBINATION C
                   1.1*DEAD+1.43*LIVE+1.43*EQ
                  ENVELOPE DESIGN
                   A,B,C

```

In this example, it is implicit that prior to the `COMBINATIONS` command, solutions for three loading cases must have been obtained. These are `SDL`, `LIVE` and `EQ`. Combinations A, B, and C, each scale and combine the respective loading according to the relationship given for each. The `ENVELOPE` command reviews the three combinations A, B, and C, and selects the maximum and minimum value of each for `DESIGN`.

## GENERIC

**Syntax:** `GENERIC`

**Explanation:** This subcommand has no arguments. It signifies that in the lines following, the loading is specified in terms of its position with respect to a node, or element. The loading specification follows a syntax identical to `ADAPT-ABI` Basic `LOADING` command, except that no imposed nodal displacements are allowed at this stage.

The load specification under this subcommand is similar to the pre-capture stage. Note that there is a greater flexibility in some types of load definition if `STRUCTURE` subcommand is used. For example concentrated loads can be applied at given distances, as opposed to nodal locations (see `STRUCTURE` subcommand).

A summary of the load specification syntaxes applicable under `GENERIC` subcommand are given below:

```

N=n1 , n2 , inc   F=fx , fy , fzz (concentrated load)
L=l1 , l2 , inc   F=fx , fy      (uniform load)
L=l1 , l2 , inc   T=ttop , tbot  (temperature load)

```

where,

```

n1   = Node number of first node in a series of nodes with identical
      loading or displacements
n2   = Node number of last nodes in the series [n1]

```

`l1` = Frame element number of first element in a series of elements with the identical applied loads or identical temperatures  
`inc` = Element or node number increment  
`fx` = X-force, or X-force per unit Y-projected length  
`fy` = Y-force, or Y-force per unit X-projected  
`fzz` = Z-moment  
`ttop` = Temperature at top fiber of cross section  
`tbot` = Temperature at bottom fiber of cross section

For additional details, refer to ABI manual.

## GRID

**Syntax:** `GRID #####`

where,

`#####` = Is any alphanumeric string up to six characters, such as **LINE4**.

**Explanation:** Each grid line is identified by a user selected identification name consisting of up to six alphanumeric characters, such as, 5/A. Each grid line and level should have its own unique identification. The same name may not be used for two, or more, grid or column lines. Solutions of columns within a grid are grouped together for graphical display, or hard-copy printout.

## GRIDS n LEVELS

**Syntax:** `GRIDS n LEVELS m`

**Explanation:** It indicates that the structure envisaged is assigned to have, `n`, column grid lines and, `m`, levels. The designation is subjective. It depends on the extent of the members a user intends to investigate in detail. Either the entire frame, or portion thereof can be designated as `GRID` and `LEVEL`. Refer to **Figure 3.3-3(a)**. It shows the elevation of a frame with three column lines, designated `C1`, through, `C3`. and four levels, `L1`, through, `L4`. The following provides several valid examples for defining part of the frame as a structure.

**Example:** (i) For **Figure 3.3-3(a)**

`GRIDS 3 LEVELS 4`

(ii) For **Figure 3.3-3(b)**

GRIDS 2 LEVELS 1

(iii) For **Figure 3.3-3(c)**

GRIDS 1 LEVELS 1

(iv) For **Figure 3.3-3(d)**

GRIDS 1 LEVELS 2

In all the previous examples, a complete solution based on the entire frame and its loading will be obtained. The difference between the various options lies in the amount and organization of output. In each case, the primary graphical and hard-copy output is limited to the grid and levels defined for the structure. This will allow the user to trim the unwanted output and focus on the members of interest. However, should a question regarding the actions and displacements in members not defined as part of the structure arise, the user can refer to a set of secondary files generated by the software. In the secondary files, the entire solution is documented for detailed review. In other words, by defining a *structure* within the entire *frame*, the user does not lose the option to access the entire solution.

## LEVEL

**Syntax:** LEVEL #####

where,

##### = Is any alphanumeric string up to six characters, such as **2LEVEL**

**Explanation:** Each level line is identified by a user selected identification name consisting of up to six alphanumeric characters, such as, 3rdLev. Each grid line and level should have its own unique identification. The same name may not be used for two, or more, grid or column lines. Solutions of spans within a grid are grouped together for graphical display, or hard-copy printout.

## LOADING

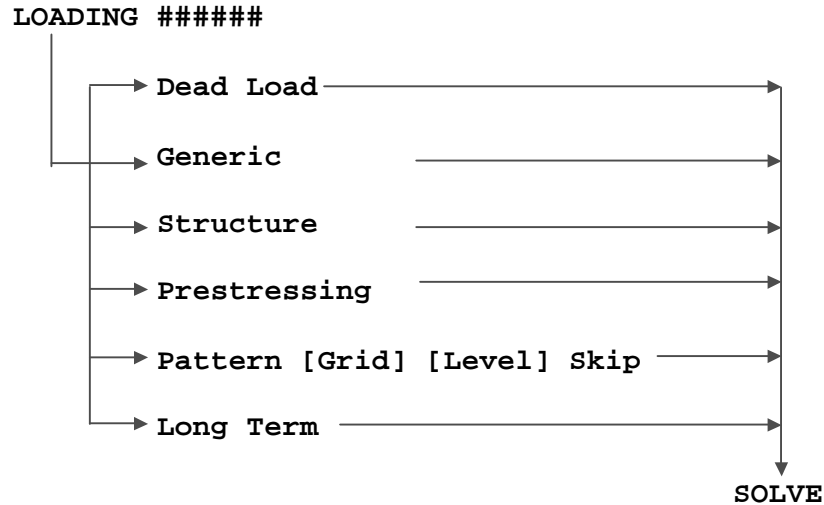
**Syntax:** LOADING #####

where,

##### = Any alphanumeric string of six or more characters used by the user to identify the loading case that is described following this command.

**Explanation:** The loading identification serves two purposes. (i) It will be printed at top of the respective solution in the output for ease of reference. (ii) When using load combinations, the contribution of each loading in the combination is effected by a user defined factor times the load. The load identification string used must be unique among the loading cases of each problem. Two or more loading cases can not be assigned the same identification string.

An overview of the subcommands following `LOADING` command is given below:



A `LOADING` command is terminated by a `SOLVE` command. Note that only one of the types of loading indicated in the preceding diagram, can be included in each `LOADING` command. The `LOADING` command, followed by `SOLVE` command must be repeated as many times, as it is necessary to cover all the loading intended by the user.

[`GENERIC`] [`STRUCTURE`] shown in the first category indicates the options. Either one, or both, of the loading types shown in each of the square brackets, [ ], can be specified under the same loading identifications. Each loading type, such as [`STRUCTURE`] is described in detail in this chapter under its designated heading. A short account for definition of each loading type is given next:

**GENERIC :**

Loading is defined in exactly the same manner as the pre-capture-status, that is to say, like in ADAPT-ABI basic module. The special feature is that loading is expressed in terms of nodes and elements. No imposed displacements are allowed at this stage.

**STRUCTURE :**

The loading is specified in terms of spans and columns of the structure previously defined.



**PRESTRESSING :**

Determines the displacements and actions due to prestressing only, including hyperstatic (secondary) actions.

**PATTERN :**

Applies to live loading. Solutions for a number of loading combinations will be computed, each with a different arrangement of live loading, in order to obtain the maxima of actions and displacements at each location. For details refer to **PATTERN** command.

**LONG TERM**

**Syntax:** LONG TERM

**Explanation:** The long-term loading case when invoked extracts the changes in actions and displacements of the structure due to the delayed effects of shrinkage and creep, and the effects of temperature change. The effects are the accumulated sum of such changes over the life of the structure until the call of **CAPTURE STATUS** command. The changes due to relaxation in prestressing are lumped with the **PRESTRESSING** loading case. The impact of “aging” on the solution is grouped with **DEAD LOAD** loading condition.

Note that each of these load cases must be invoked on its own, separate from the other loads. Once the values are obtained, they can be enveloped or combined with other loads.

**Example:**

```

CAPTURE STATUS

LOADING DL
  DEAD LOAD           ; reserved loading case
  SOLVE
  OUPUT

LOADING PT
  PRESTRESSING       ; reserved loading case
  SOLVE
  OUPUT

LOADING LT
  LONG TERM          ; new reserved loading case
  SOLVE
  OUPUT

LOADING POINT1
  GENERIC

```

```
N=3  F=0,-100,0
SOLVE
OUPUT
```

```
COMBINATIONS
COMBINATION SELWGT ; gives a solution identical
      1*DL+1*PT+1*LT ; to the last stage of
OUTPUT ; precapture
```

## OUTPUT

**Syntax:** OUTPUT [ALL][SAME][DISPLACEMENTS][ACTIONS][REACTIONS]  
[STRESSES]

**Explanation:** Other options within [ ] are identical in function to the pre-capture OUTPUT command, except PRESTRESSING. The prestressing command would function only if the preceding loading is, prestressing, otherwise it would be ignored. The reason the output for prestressing is turned off for other loading conditions is that prestressing would not change for the post-capture solutions. Hence, its printout would be repetitious.

If the loading preceding the output command includes SKIP subcommand, the output would automatically generate an *envelope*.

OUTPUT command in conjunction with COMBINATION and ENVELOPE commands has the same syntax, and is mandatory to obtain printout of such results.

## PATTERN

**Syntax :** PATTERN

**Explanation:** The PATTERN subcommand indicates that the loading followed will be placed selectively on different locations, in order to obtain the maximum and minimum values of displacements and actions at each location in the *structure*.

The general form of Pattern subcommand is as follows:

```
PATTERN
      Applied Loading
SKIP
      Skip table
SOLVE
```

The pattern subcommand must be followed by `SKIP` and `SOLVE` subcommands as shown in the preceding. After the subcommand `PATTERN` and before the `SKIP` subcommand, the loading which will be selectively applied is described. After the subcommand `SKIP`, the sequence in which the defined loading must be applied to various spans and columns will be specified in a user defined skip table. In the *applied loading* section, the entire loading which is envisaged to be considered for skipping, will be defined. It is the *skip table* which determines which loading must be applied where and together with which other loading.

The general sequence of loading and skip table is as follows:

```

PATTERN
  LEVEL #####
    SPAN
      CONCENTRATED
      UNIFORM
      TEMPERATURE
  GRID #####
    COLUMN
      CONCENTRATED
      UNIFORM
      TEMPERATURE
SKIP
  GRID ##### COLUMNS ? ? ? ?
  LEVEL ##### SPANS ? ? ? ?
SKIP
  GRID ##### COLUMNS ? ? ? ?
  LEVEL ##### SPANS ? ? ? ?
.....
SOLVE

```

The applied loading is described in exactly the same manner which is outlined under `STRUCTURE` subcommand of loading. That is to say, the loading is defined in terms of spans and columns of levels and grids. The loading can not be specified in the generic format. `PRESTRESSING`, `DEAD LOAD`, and `LONG TERM` can not be specified as part of skip loading. The effect of these three types of loading on the structure can best be investigated through load `COMBINATIONS` command.

The skip tables describe where the loading must be applied. Each skip table starts with the subcommand `SKIP`. It is terminated by either the `SOLVE` command, or start of another `SKIP` table.

The general format of a skip table is as follows:

```

SKIP
    LEVEL ##### SPANS 1 0 1 0 1 0 ....
    LEVEL ##### SPANS 0 1 0 1 0 1 0 ....
;..... more levels, if needed .....
    GRID ##### COLUMNS 1 1 1 1 ....
    GRID ##### COLUMNS 1 1 1 1 ....
;..... more columns, if needed .....
SKIP
    LEVEL ##### SPANS 0 1 0 1 0 1 ...
.....

```

A skip table is nested between two SKIP subcommands, or a SKIP subcommand and a SOLVE command. After the token SPANS, or COLUMNS, a series of 1, and, 0, are specified. Each digit refers to one of the spans or columns, starting from span/column 1 and continuing successively until the last span. Hence, the number of digits following a SPANS, or COLUMNS token equals the number of respective spans/columns of the associated LEVEL/GRID. Only the levels and spans which are called in the table and are given an identification of, 1, are loaded. A, 0, means that the associated span, or column is not loaded. For example, SPANS 1 0 1 1, signifies that there are 4 spans on the associated level. Spans 1, 3, and 4 are loaded, span, 2, is not loaded.

## PH

**Syntax:** PH

**Explanation:** Refers to hyperstatic (secondary) actions. It is described under subcommand PRESTRESSING

## PRESTRESSING

**Syntax:** PRESTRESSING

**Explanation:** This subcommand has no argument. When invoked, it generates a solution to the prestressing available at time of capture. At time of CAPTURE STATUS, the time-dependent parameters are frozen, hence, there will be no change in prestressing due to creep, shrinkage, aging of concrete, and relaxation of steel. However, due to post-capture loading there will be changes in displacement of the structure. Strictly speaking, the strains due to displacement cause a change in prestressing forces. This subcommand gives a solution which corresponds to the condition of the frame at CAPTURE STATUS time. Once invoked, it

generates a detailed set of information on prestressing forces, losses, displacements due to prestressing and hyperstatic actions (secondary actions). Changes in prestressing due to applied loading must be investigated at the pre-capture condition. At the post-capture state, regardless of when the *prestressing* subcommand is called, it always yields the same solution, since time-dependency and impact of displacement of structure on prestressing are suppressed.

From the preceding, the `PRESTRESSING` subcommand generates two sets of solutions. Both sets are printed in the user's output and are available for graphical viewing. The first set is named `PT`. It contains the displacements and actions (moments, shears, and axial loading) associated with the prestressing forces at the time of the capture (freezing of time-dependent parameters). The second set is named `PH`. It is the hyperstatic (secondary) actions due to prestressing at the time of capture. The hyperstatic actions are defined as actions generated due to post-tensioning forces in the structure, due to the structure's resistance to displacement at the supports. The relationship between the two actions at any given location is as follows, for additional details refer to Appendix A.

$$PH = PT - P \cdot e$$

where,

`PH` = Refers to hyperstatic actions  
`PT` = Refers to the total of actions due to post-tensioning forces  
`P` = Prestressing force  
`e` = eccentricity of the prestressing force (`P`)

It is important to note that the hyperstatic actions are computed incrementally with the progress of construction. These actions are duly adjusted as stress losses take place, and as the boundary conditions of the structure change.

For example, pre-stressing in a simply supported single span girder will not cause hyperstatic action by itself. But if the girder is then locked to other parts of the structure (made continuous), subsequent changes in prestressing, such as due to creep, will result in hyperstatic actions. Because of its incremental solution strategy, ADAPT-Gen correctly captures and reports the associated hyperstatic actions.

## SPAN

### Syntax:

```
SPAN i E=l1,l2,inc
```

or

```
SPAN i LIST=l1,l2,l3,...,ln
```

where,

*i* = Span number  
*l1* = Frame element number of first element in a series of elements which define the column  
*l2* = Element number of last element in series  
*inc* = Element number increment  
*l1, l2, l3, . . . , ln* = Frame element number of first element in a series of elements which define the span, followed by a sequential list of other elements, *l2, l3*, until the last element of series, *ln*.

**Explanation:** Each span in a LEVEL is defined by one subcommand line starting with SPAN *i*. The span number, *i*, is a sequential integer designating the span being defined from the list of spans in a LEVEL. For example, if a level has 5 spans, there must be five SPAN subcommands, each defining one of the 5 spans of the level.

**Example:** (i) Refer to **Figure 3.3-3(d)**

```
LEVEL L2
    SPAN 1
    SPAN 2
LEVEL L3
    SPAN 1
```

Note that the span not highlighted as part of the structure is not listed.

(ii) Refer to **Figure 3.3-1** and **Figure 3.3-2**

```
LEVEL 2nd_lv
    SPAN 1 List=9,10,11,12
LEVEL 3rd_lv
    SPAN 1 E=13,16,1
```

(iii) Refer to **Figure 3.3-3(c)**

```
LEVEL L2
    SPAN 1 E=....
    SPAN 2 E=....
```

**STOP**

**Syntax:** STOP

**Explanation:** The STOP is the last command in the input data that terminates the program. Its inclusion is mandatory.

**STRUCTURE**

**Syntax:** STRUCTURE

**Explanation:** Signifies that the **loading** is specified in reference to the structural members (columns and span) as previously defined under the STRUCTURE DEFINITION command. Defining the load under STRUCTURE subcommand has the advantage that the user can specify the loads in term of spans, such as uniform load on an entire span, as opposed to loads on nodes and elements. Likewise, concentrated loads can be given at distances from the span supports compared to specification at the nodes.

The common sequence and syntax for the structural member definition for this subcommand is:

```

STRUCTURE
  LEVEL #####
    SPAN [n][l1,l2,inc][ALL]
    CONCENTRATED N=n1,n2,inc F=fx,fy,fzz \
                  [at N=?][at x=?]
    UNIFORM      [L=l1,l2,inc] F=fx,fy \
                  [xB=?][xE=?][NB=?][NE=?]
    TEMPERATURE  [L=l1,l2,inc] T=ttop,tbot
  GRID #####
    COLUMN [n][l1,l2,inc][ALL]
    CONCENTRATED N=n1,n2,inc F=fx,fy,fzz \
                  [at N=?][at x=?]
    UNIFORM      [L=l1,l2,inc] F=fx,fy \
                  [xB=?][xE=?][NB=?][NE=?]
    TEMPERATURE  [L=l1,l2,inc] T=ttop,tbot

```

In the following the syntax of each of the subcommands is described in more details.

**Syntax:** CONCENTRATED N=n1,n2,inc F=fx,fy,fzz \  
[at N=?][at x=?]

where,

$n1$  = Node number of first node in a series of nodes with identical applied loading  
 $n2$  = Node number of last node in the series [ $n1$ ]  
 $inc$  = Node number increment  
 $f_x$  = X-force  
 $f_y$  = Y-force  
 $f_{zz}$  = Z-moment  
 $At N$  = Node number where loads are applied  
 $AT x$  = Distance,  $x$ , measured in local coordinate along the member, where loads are applied

The *local coordinate* of a member are defined as the line joining the first node of the member to its last node.

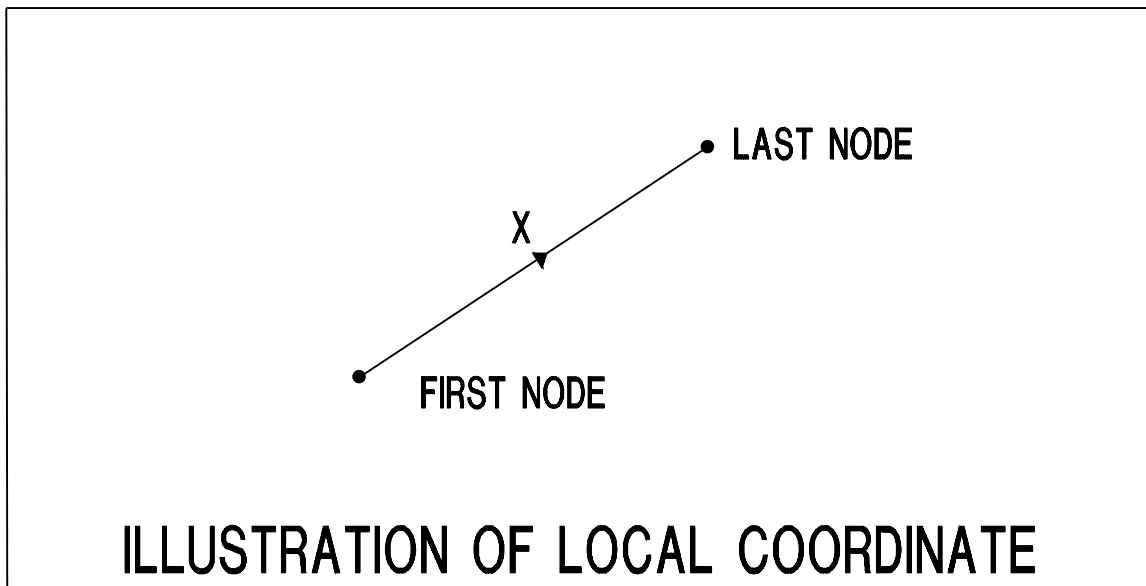
**Example:** Refer to **Figure 3.3-3**

```

SPAN 2
CONCENTRATED F=-200,0,0 AT x=120
  
```

Note: There is a concentrated loading in the global  $y$ -direction with magnitude of -200 units acting at a distance 120 units along the length of span 2 from its beginning.

The subcommand described must be repeated as many times as necessary to describe the entire loading. In the general case, it is not practical to specify all the loading in one line.



**FIGURE 3.3-4**



**Syntax:** GRID #####  
 COLUMN [n][n1,n2,inc][ALL]

**Note:** The syntax and load description is identical to that of span described in the preceding:

```
CONCENTRATED N=n1,n2,inc F=fx,fy,fzz \
              [at N=?][at x=?]
UNIFORM      [L=l1,l2,inc] F=Fx,Fy [xB=?][xE=?]\
              [NB=?][NE=?]
TEMPERATURE  [L=l1,l2,inc] T=Ttop,Tbot
```

**Explanation:** The syntax and load specification of subcommands Concentrated, Uniform and Temperature are identical to the description given under Span subcommand.

**Syntax:** LEVEL #####

**Explanation:** Indicates that the following loading are on spans of level previously identified by the alphanumeric string, #####. The level must have been previously defined in the STRUCTURE DEFINITION command, before its spans can be loaded.

**Syntax:** SPAN [n][l1,l2,inc][ALL]

where,

n = Span number  
 l1 = Frame element number of first element in a series of nodes  
 l2 = Element number of last element in the series  
 inc = Node number increment  
 ALL = All spans

**Examples:** SPAN 2  
 SPAN 2,5,1  
 SPAN ALL

**Syntax:** TEMPERATURE [L=l1,l2,inc] T=ttop,tbot

where,

l1 = Frame element number of first element in a series of elements with identical applied loads  
 l2 = Element number of last element in the series

`inc` = Element number increment  
`ttop` = Temperature at the top fiber of the element  
`tbot` = Temperature at the bottom fiber of the element

**Explanation:** If `[L=l1,l2,inc]` is omitted, the temperature specified is applied to the entire member (span, or column)

**Syntax:** `UNIFORM [L=l1,l2,inc] F=fx,fy [xB=?][xE=?]\`  
`[NB=?][NE=?]`

where,

`l1` = Frame element number of first element in a series of elements with identical applied loads  
`l2` = Element number of last element in the series  
`inc` = Element number increment  
`fx` = X-force per unit Y-projected length  
`fy` = Y-force per unit X-projected length  
`NB` = Node number of the beginning point of the pressure loading  
`NE` = Node number of the end point of pressure loading  
`xB` = Coordinate, x, of the beginning point of the pressure loading in local coordinates [0]  
`xE` = Coordinate, x, of the end point of the pressure loading in local coordinates, [length of member]

**Example:** Refer to **Figure 3.3-3**

```

LEVEL L1
      SPAN ALL
          UNIFORM F=-10,0
  
```

Note: All spans of level L1 are subject to a uniform load in the global y-direction with intensity of, -10, per unit of length

The subcommand described must be repeated as many times as necessary to describe the entire loading. In the general case, it is not practical to specify all the loading in one line.

## STRUCTURE COMPLETE

**Syntax:** STRUCTURE COMPLETE

**Explanation:** The STRUCTURE COMPLETE command has no argument of its own. It signals the termination of subcommands used to define either a region, or the entire frame previously defined as a STRUCTURE DEFINITION.

## STRUCTURE DEFINITION

**Syntax:** STRUCTURE DEFINITION

**Explanation:** The STRUCTURE DEFINITION command has no argument of its own, but is followed by a series of subcommands which define as a *structure*, either a region, or the entire, frame generated through the MESH command.

The structure definition command includes one or more subcommands nested between the two commands STRUCTURE DEFINITION and STRUCTURE COMPLETE. These commands are:

STRUCTURE DEFINITION	:	starts the structure definition;
GRIDS n LEVELS m	:	gives total of column and level lines;
GRID #####	:	header for column line #####;
COLUMN i	:	header for column, i;
LEVEL #####	:	header for level #####;
SPAN i	:	header for span, i;
STRUCTURE COMPLETE	:	terminates the STRUCTURE DEFINITION command.

This Page Left Intentionally

BLANK

## 4.1 BACKGROUND

ADAPT-GEN is a software for the analysis and design of concrete or composite plane frames with or without prestressing. It accounts for the time-dependent behavior of concrete material in shrinkage, creep, aging, as well as the time-dependent response of prestressing in stress relaxation. In addition, the software has the capability of incremental construction, or dismantling of a frame (reverse construction).

The basic formulation of the problem and the procedure for obtaining solutions are detailed in Chapter 4 of ADAPT-ABI program manual. ADAPT-ABI is the primary program module, to which ADAPT-GEN is an augmentation. In this chapter, only the features which are specific to ADAPT-GEN are presented.

## 4.2 FEATURES OF ADAPT-GEN

For each problem, ADAPT-GEN performs two types of analysis. The first is referred to as pre-capture and the second the post-capture. The word *capture* is used in relation to the command `CAPTURE STATUS` in input data. When invoked, this command will freeze the lapse of time for the frame being analyzed.

### 4.2.1 Pre-Capture Features

The program behaves identically to ABI. The frame can be constructed, or dismantled incrementally, with or without allowance for time-dependent parameters: creep, shrinkage, aging of concrete and relaxation of prestressing. Further, the load and stress history of the structure is retained and implemented in the analysis as time and construction proceed.

### 4.2.2 Post-Capture Features

Once the `CAPTURE STATUS` command is encountered, the software assumes the post-capture mode. In this mode, the time element is assumed frozen at the date the capture status command was called. All subsequent computations are performed without inclusion of creep, shrinkage, aging of concrete and relaxation of prestressing - if any. Existing stress and load history of the frame are abandoned. Further:

- The structure is assumed to have the configuration and the support condition of the `CAPTURE STATUS` day. No change in configuration of the structure is allowed, neither for adding, nor for deleting of elements. Supports may not be added or removed.
- The stiffness of the structure is assumed to remain unchanged throughout the post-capture analysis. It is assumed to be equal to stiffness associated with the day the capture status command was called, but excluding the stiffness of prestressing tendons. The stiffness of cable stays and the stiffness of nonprestressed steel is retained. Note that in precapture condition, the stiffness

due to the extension of tendons under applied loading is included in the analysis.

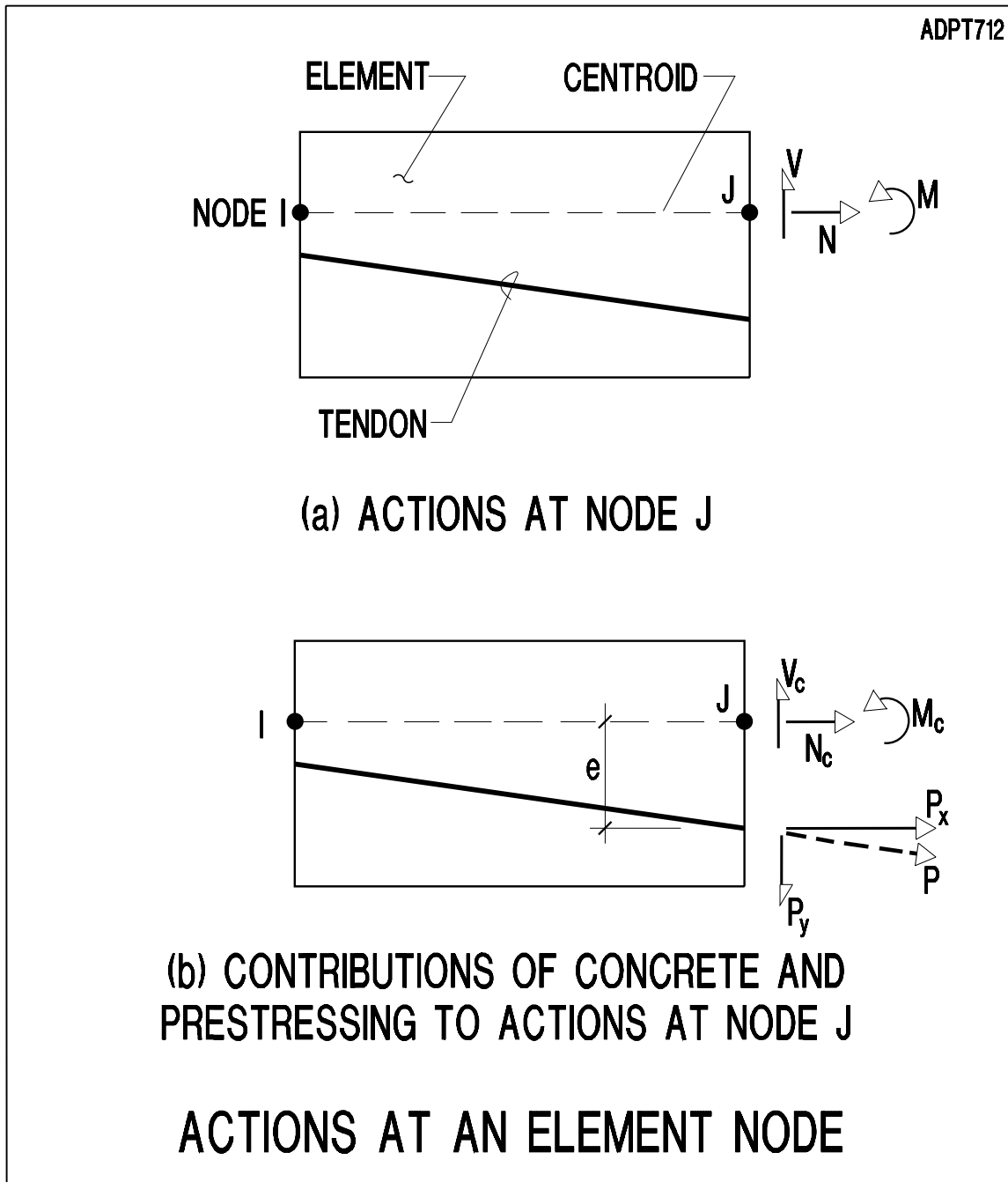


FIGURE. 4.3-1

- The structure is assumed to be under no loading. Selfweight is also removed.
- A linear elastic analysis is used to determine the deformation and stresses due to each added loading. Hence, the solution of each loading is independent from the prior loading on the structure. Each solution is stored separately. The

stresses and strains associated with the current loading are flushed out from the structural data prior to the analysis of a new loading case.

### 4.3 STRESSES IN CONCRETE; PRE-, AND POST-CAPTURE ALLOCATION OF ACTIONS AT A NODE

In the interpretation of the ABI solutions, it is important to be clear on the method used in allocating actions at a section among concrete, prestressing, and the non-prestressing reinforcement. Also, the computation of concrete stresses requires special attention.

The resultant moment, shear, and axial loading are defined as actions at a node.

#### 4.3.1 Description

Refer to **Figure 4.3-1**. It shows an element, IJ, with the actions entered at node, J. In **Figure 4.3-1(a)** the resultant of external actions at node, J, are the moment, shear and axial loading, M, V, and N, as shown.

For pre-capture computations, the internal actions at a node are made up from contributions of the prestressing tendons, if any, and the concrete. Refer to **Figure 4.3-1(b)**. The force, P, represents the resultant force of all tendons at node, J. For clarity, only one tendon is shown. The relationships between the resultant actions at node, J, and their constituents are:

$$\begin{aligned} M &= M_c + P_x * e \\ V &= V_c + P_y \\ N &= N_c + P_x \end{aligned}$$

Where,  $M_c$ ,  $V_c$ , and  $N_c$ , are moment, shear and axial loading contribution of concrete (plus nonprestressed reinforcement);  $P_x$ , and  $P_y$  are contributions from prestressing. The contribution from pre-stressing is due to initial prestressing force with allowance for creep, shrinkage, aging of concrete, relaxation of stress in tendon, and strain at location of strand due to deformation of the structure.

At the pre-capture status stage, the actions listed for the element at a node are those associated with the concrete constituent, namely,  $M_c$ ,  $N_c$ , and  $V_c$ . Likewise, the stresses are those due to these actions acting on the section. Hence, stress due to bending is:

$$f_{top} = M_c * y_{top} / I$$

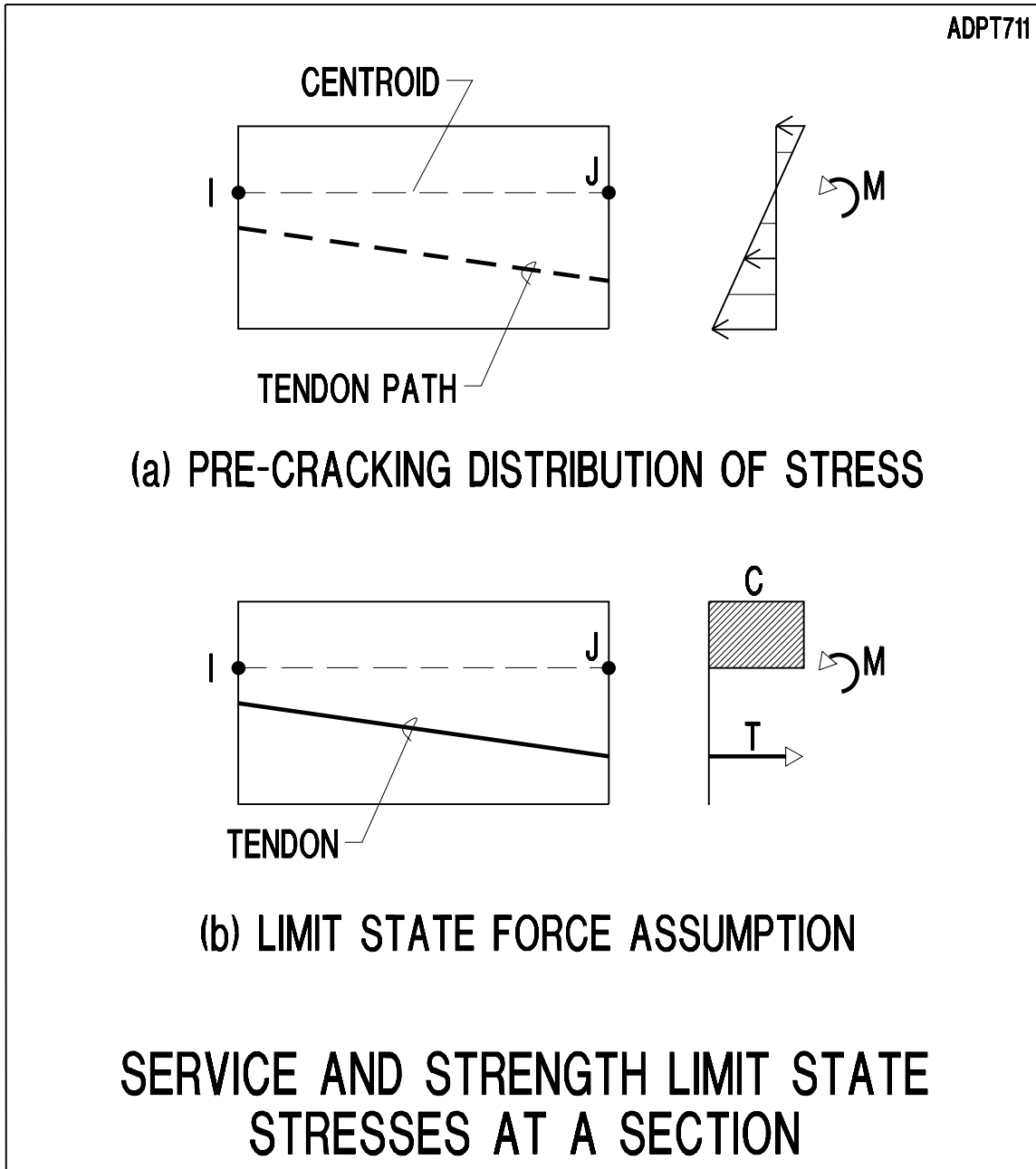


FIGURE 4.3-2



Where,  $f_{top}$  is stress at top fiber;  $y_{top}$  is distance of centroid to top fiber; and  $I$  is the moment of inertia of the section. Stress at bottom fiber is obtained in a similar manner. Refer to **Figure 4.3-2(a)** for distribution of stress at pre-capture condition.

At post-capture stage, the actions listed for node,  $J$ , are due to forces on the entire concrete cross-section, that is to say,  $M$ ,  $N$ , and  $V$  in **Figure 4.3-1**. The actions are not broken between the tendons and the concrete cross-section. Hence for the same loading and boundary conditions, the stress obtained for concrete can be different for pre- and post-capture conditions. The commonly performed engineering hand calculations correspond to the post-capture scheme of the software. At limit state, the stress distribution on the section is as shown in **Figure 4.3-2(b)**.

In simple hand calculations, the contributions of tendon in resisting the action is generally disregarded. The sum of actions at this node, namely,  $M_c$ ,  $N_c$ , and  $V_c$ , as shown in **Figure 4.3-1(b)**.

Hence, for the post-capture condition, the stress due to moment at top fiber is:

$$f_{top} = M * y_{top} / I$$

### 4.3.2 Illustrative Example

The cantilever beam shown in **Figure 4.3-3** is loaded at its tip and prestressed with a sloping tendon as indicated. To simplify the problem, selfweight, creep, shrinkage, aging of concrete and relaxation in prestressing are all suppressed. No stress loss in post-tensioning is included. The objective is to demonstrate the computation and interpretation of moments and stresses at pre- and post-capture cases.

Consider section a-a, 45-in. from the support. The section is at the  $J$ , node of element, 3, (**Figure 4.3-3(d)**). The actions and stresses for this section at pre- and post-capture stages will be reviewed next. The input data for this example is also appended at the end of this review. Units are all in lb-in.

The tendon makes an angle,  $\theta = 30.964^\circ$ , with the centroidal axis.

(i) External actions at section a-a:

$$\begin{aligned} M &= -30,000 * 15 = -450,000 \\ N &= -40,000 \\ V &= 30,000 \end{aligned}$$

Components of the prestressing force ( $P$ ) at section a-a:

$$\begin{aligned} P &= 50,000 \text{ lb.} \\ P_x &= 50,000(\cos 30.964) = 42,874 \text{ lb.} \\ P_y &= 50,000(\sin 30.964) = 25,725 \text{ lb.} \end{aligned}$$

(ii) Pre-capture internal actions at section a-a:

$$e = 9$$

$$P = 50,000$$

$$P_x = 50,000 * \cos\theta = 42,874$$

$$P_y = 50,000 * \sin\theta = 25,725$$

$$M_c = M - P_x * e$$

$$= -450,000 - 42,874 * 9 = -835,866$$

*(ADAPT-GEN output, data block 107.1 gives -835,870, OK)*

$$V_c = V - P_y$$

$$= 30,000 - 25,725 = 4,275$$

*(ADAPT-GEN output, data block 107.1 gives 4275.2, OK)*

$$N_c = N - P_x$$

$$= -40,000 - 42,874 = -82,874$$

*(ADAPT-GEN output, data block 107.1 gives 82,875, OK)*

$$\text{Section modulus, } S = 10 * 40^2 / 6 = 2666.7$$

$$\text{Area} = 10 * 40 = 400$$

Tension stress

$$\text{in concrete} = M_c / S - N_c / A$$

$$= 835,866 / 2666.7 - 82,874 / 400$$

$$= 106.26$$

*(ADAPT-GEN output, data block 108 gives 106.27, OK)*

(iii) Post-capture internal actions at section a-a:

$$M = -450,000$$

*(ADAPT-GEN output, data block 107.1 gives -450,000)*

$$N = -40,000$$

*(ADAPT-GEN output, data block 107.1 gives -40,000)*

$$V = 30,000$$

*(ADAPT-GEN output, data block 107.1 gives 30,000)*

$$\text{Tension stress} = 450,000/2,666.7 - 40,000/400 = 68.75$$

*(ADAPT-GEN output, data block 108 gives 68.75)*

Note that for the post-capture condition, the stresses in concrete do not include pre-stressing.

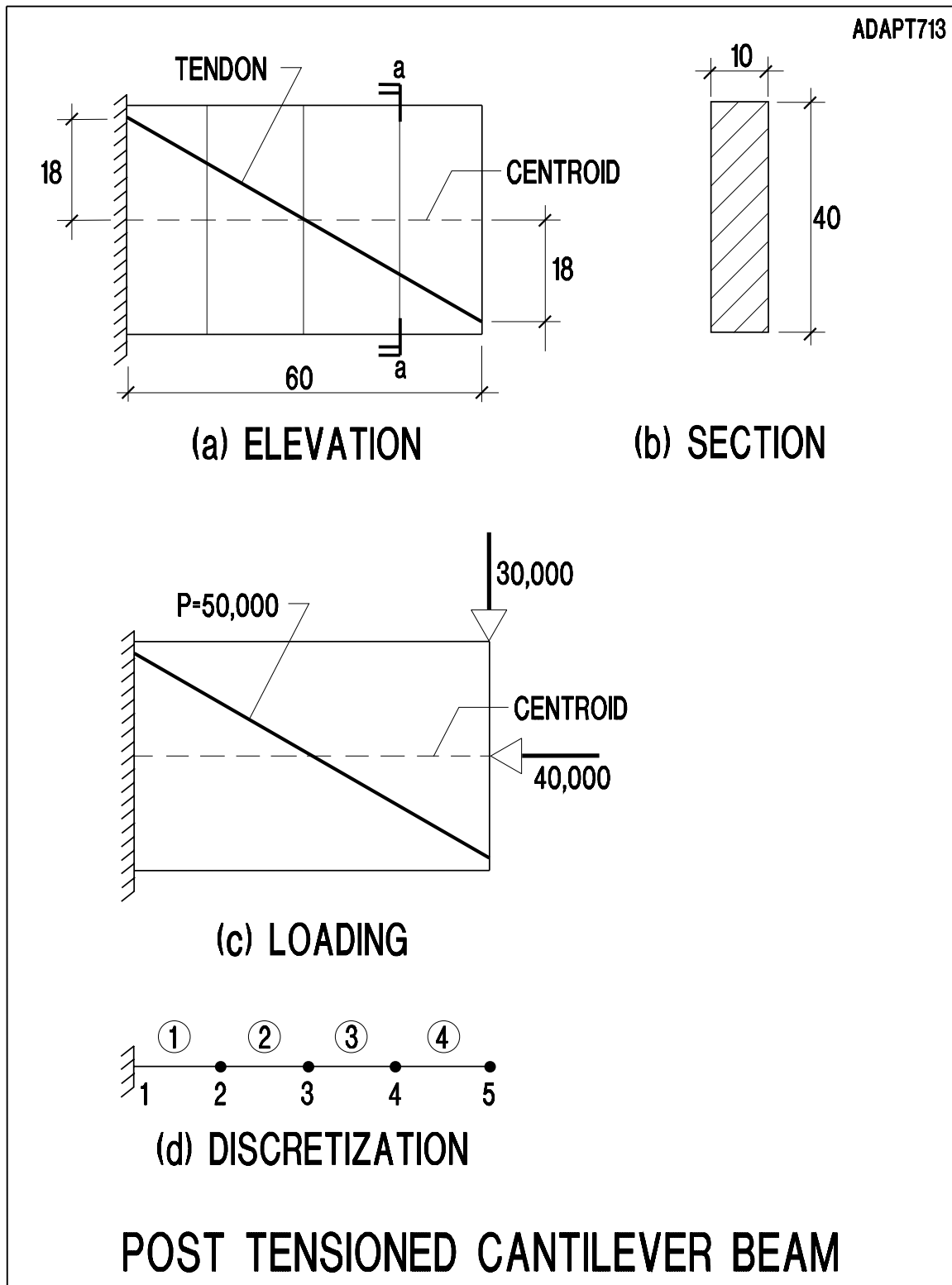


FIGURE 4.3-3

## (iv) Input File

```

;=====
; ADAPT-BRIDGE-INCREMENTAL   ABI   SOFTWARE MANUAL EXAMPLES
;=====
; Name of this file: GEN-EX01.INP
; Units are in lb-in

START
TITLE N=1
      CANTILEVER BEAM WITH PRESTRESSED TENDON

CONCRETE PARAMETERS N=1
      1 M=ACI a=0 b=1 ;no aging

MESH INPUT
      NODES N=5
          1 X=0 Y=0.0 ! 5 X=60 Y=0.0 G=1,5,1

      CONCRETE PROPERTIES N=1
          1 Fpc=4000 Cr=0.0 Sh=0.0 W=0.0 ;no creep and shrinkage
      SECTION PROPERTIES N=1
          1 D=40 B=10
      ELEMENTS N=4
          FRAME N=4
              1,1,2 C=1 X=1 Day=1 G=1,4,1,1,1

      PRESTRESSING STEEL N=1
          1 Ep=2.87E7 Meu=0.0 K=0.0 Fpu=270000 ;no friction
      TENDON GEOMETRY N=1
          1 Spans=1 M=1 Area=0.306
              1 N=5
                  LIST=1,2,3,4,5
                      1 r=0 s=18
                      2 r=15 s=9
                      3 r=30 s=0
                      4 r=45 s=-9
                      5 r=60 s=-18

MESH COMPLETE

CHANGE STRUCTURE
      BUILD N=1,4,1
      RESTRAINTS
          1 R=1,1,1

      STRESS N=1 Force=0,50000
CHANGE COMPLETE

```

```

LOADING
  N=5 F=-40000,-30000,0

SOLVE
OUTPUT

CAPTURE STATUS

LOADING CASE1
  GENERIC
    N = 5,5,0 F = -40000,-30000,0

SOLVE
OUTPUT

STOP

```

#### 4.4 HYPERSTATIC ACTIONS: SECONDARY ACTIONS

The hyperstatic (secondary) actions are moments, shears, and axial loading, caused by the restraint of the supports to free deformation of the structure under prestressing load. At post-capture condition, the hyperstatic actions (secondary) are computed by the software. Hyperstatic actions are calculated incrementally accounting for load and stress history of the structure. At each increment of time or loading, the associated increment of hyperstatic actions are give by:

$$M_{hp} = M_p - P * e$$

$$N_{hp} = N - P$$

$$V_{hp} = V - V_p$$

where,

$M_{hp}$  = Hyperstatic (secondary) moment

$M_p$  = Moment due prestressing forces

$P$  = Combined force of prestressing tendons in a member

$e$  = Combined eccentricity of prestressing tendons

$N_{hp}$  = Hyperstatic axial loading

$N$  = Axial force in a member due to application of the prestressing forces on the structure

$V_{hp}$  = Hyperstatic shear

$V$  = Shear, including contribution of tendons, in a section caused by the action of prestressing forces on the structure

$V_p$  = Combined component of prestressing force in plane of section

#### 4.5 ENVELOPES AND COMBINATIONS

The load combinations are linear addition of solutions obtained from loading components, each multiplied by a coefficient called for in the load combination. The envelopes give the maxima of the actions and displacements at each location.

## 5.1 GENERAL

The output of ADAPT-GEN is identical to ADAPT-ABI basic, using the same data block numbering and organization, except as detailed in this chapter.

## 5.2 CONSTRUCTION PHASE (PRE-CAPTURE)

The only addition to pre-construction phase is the definition of *structure*- if the option is used. The structure defined is printed out in data block 6, as shown in the following, extracted from example GEN-EX62.

```

6. STRUCTURAL DEFINITION AND INPUT DATA
=====
6.1 LEVELS OF THE STRUCTURE
-----
LEVEL = LEV1
  Span =          1
    NODES      44  45  46  47  48  49  50
    ELEMENTS   41  42  43  44  45  46
  Span =          2
    NODES      51  52  53  54  55  56  57
    ELEMENTS   49  50  51  52  53  54
  Span =          3
    NODES      58  59  60  61  62  63  64
    ELEMENTS   57  58  59  60  61  62
  Span =          4
    NODES      65  66  67
    ELEMENTS   65  66

LEVEL = LEV2
  Span =          1
    NODES      68  69  70  71  72  73  74
    ELEMENTS   68  69  70  71  72  73
  Span =          2
    NODES      75  76  77  78  79  80  81
    ELEMENTS   76  77  78  79  80  81

6.1 COLUMN GRID LINES
-----
GRID = AA
  Column =        1
    NODES        1   2   3   4   5
    ELEMENTS     4   5   6   7
  Column =        2
    NODES        7   8   9  10  11
    ELEMENTS    10  11  12  13

GRID = BB
  Column =        1
    NODES       14  15  16  17  18
    ELEMENTS    15  16  17  18
  Column =        2
    NODES       20  21  22  23  24
    ELEMENTS    21  22  23  24
  Column =        3
    NODES       26  27  28
    ELEMENTS    27  28

GRID = CC
  Column =        1
    NODES       30  31  32  33  34
    ELEMENTS    30  31  32  33

```

```

GRID = DD
Column =
      1
      38 39 40 41 42
      35 36 37 38

```

### 5.3 REGULAR LOADING CASES

A regular loading case is one which does not include skipping of loads, combinations, envelopes, nor does it include prestressing. The output for a regular loading case is the same as ADAPT-ABI, with the exception that no prestressing is output.

### 5.4 PATTERNED (SKIPPED) LOADING CASES; ENVELOPES

The output for skipped load cases and envelopes use the same format. The format is differentiated from the regular output printout, in that, for the skipped and enveloped cases the maxima and minima of the actions are each listed together with the associated values of other actions or displacements. For example, the maximum moment at a section is printed with its associated shear and axial loading. Likewise, the maximum shear at the same location is printed with the moment and axial loading which corresponds with that shear.

Examples of the printout are given next:

```

=====
===  NON-TIME-DEPENDENT SOLUTIONS : ENVELOPE LIVE  ===
=====

104 LOADING
=====
104.1 EXTERNALLY APPLIED LOADING
-----
NODE      X-FORC      Y-FORC      Z-MOMENT      X-DISP      Y-DISP      Z-ROTN
1      0.0000E+00  -1.2500E+02  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
2      0.0000E+00  -2.2500E+02  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
3      0.0000E+00  -1.0000E+02  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
6      0.0000E+00  -1.0000E+02  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
7      0.0000E+00  -1.0000E+02  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00

104.2 ELEMENT TEMPERATURES
-----
FRAME      TOP FIBER      BOTTOM FIBER
ELEMENT  TEMPERATURE  TEMPERATURE
7      0.0000E+00  0.0000E+00

105 - DISPLACEMENTS AT THE NODES
=====
LEVEL LEV1  SPAN      1
-----
NODE  U-max      V-max      ROT-max      U-min      V-min      ROT-min
-----
1      0.0000E+00  0.0000E+00  -6.4838E-08  0.0000E+00  0.0000E+00  -1.5886E-07
2     -1.2451E-11  -6.0423E-07  -5.1594E-08  -2.3346E-11  -1.5071E-06  -1.3441E-07

```



```

LEVEL LEV1  SPAN  2
-----
NODE  U-max      V-max      ROT-max      U-min      V-min      ROT-min
-----
  2 -1.2451E-11 -6.0423E-07 -5.1594E-08 -2.3346E-11 -1.5071E-06 -1.3441E-07
  3 -7.8160E-11 -9.8029E-07 -2.2870E-08 -1.6759E-10 -2.5620E-06 -7.2075E-08

106.1 - REACTIONS AT FIXED NODES; MAXIMA AND MINIMA VALUES
NODE  X-FORCE-max Y-FORCE      Z-MOMENT      R-FORCE
-----
  1 -1.7903E-02 -6.2499E+02  0.0000E+00  6.2499E+02
  7  0.0000E+00 -2.5416E+02  0.0000E+00  2.5416E+02

106.2 - REACTIONS AT FIXED NODES; MAXIMA AND MINIMA VALUES
NODE  X-FORCE      Y-FORCE-max Z-MOMENT      R-FORCE
-----
  1 -1.7903E-02 -6.2499E+02  0.0000E+00  6.2499E+02
  7  0.0000E+00 -5.0832E+02  0.0000E+00  5.0832E+02

106.3 - REACTIONS AT FIXED NODES; MAXIMA AND MINIMA VALUES
NODE  X-FORCE      Y-FORCE      Z-MOMENT-max R-FORCE
-----
  1 -9.6975E-03 -3.9583E+02  0.0000E+00  3.9583E+02
  7  0.0000E+00 -2.5416E+02  0.0000E+00  2.5416E+02

106.4 - REACTIONS AT FIXED NODES; MAXIMA AND MINIMA VALUES
NODE  X-FORCE      Y-FORCE      Z-MOMENT      R-FORCE-max
-----
  1 -1.7903E-02 -6.2499E+02  0.0000E+00  6.2499E+02
  7  0.0000E+00 -5.0832E+02  0.0000E+00  5.0832E+02

LEVEL LEV1  SPAN  1
-----
107.1m; MOMENTS: MAXIMA AND MINIMA VALUES
NO  MOMENT-max  SHEAR      AXIAL      MOMENT-min  SHEAR      AXIAL
-----
  1 I  1.1829E-11  4.9999E+02 -1.7903E-02  3.9160E-12  2.7083E+02 -9.6975E-03
     J  4.9999E+03  4.9999E+02 -1.7903E-02  2.6666E+03  2.6666E+02 -9.5483E-03

LEVEL LEV1  SPAN  1
-----
107.1s; SHEARS: MAXIMA AND MINIMA VALUES
NO  MOMENT      SHEAR-max  AXIAL      MOMENT      SHEAR-min  AXIAL
-----
  1 I  1.1829E-11  4.9999E+02 -1.7903E-02  1.1319E-11  2.6666E+02 -9.5483E-03
     J  4.9999E+03  4.9999E+02 -1.7903E-02  2.6666E+03  2.6666E+02 -9.5483E-03

LEVEL LEV1  SPAN  1
-----
107.1a; AXIAL LOADING: MAXIMA AND MINIMA VALUES
NO  MOMENT      SHEAR      AXIAL-max  MOMENT      SHEAR      AXIAL-min
-----
  1 I  1.1319E-11  2.6666E+02 -9.5483E-03  1.1829E-11  4.9999E+02 -1.7903E-02
     J  2.6666E+03  2.6666E+02 -9.5483E-03  4.9999E+03  4.9999E+02 -1.7903E-02

108 - EXTREME FIBER STRESSES IN CONCRETE
=====

LEVEL LEV1  SPAN  1
ELEMENT  TOP-max  TOP-min  BOT-max  BOT-min
-----
  1 I -4.7742E-06 -8.9516E-06 -4.7742E-06 -8.9516E-06
     J -2.0000E-01 -3.7500E-01  3.7498E-01  1.9999E-01

LEVEL LEV1  SPAN  2
ELEMENT  TOP-max  TOP-min  BOT-max  BOT-min
-----
  2 I -1.9999E-01 -3.7499E-01  3.7488E-01  1.9994E-01
     J -2.3747E-01 -5.8118E-01  5.8107E-01  2.3742E-01

```

## 5.5 PRESTRESSING

The output for prestressing lists both the actions and displacements due to prestressing, as well as the hyperstatic actions precipitated by prestressing forces. Examples of the associated data blocks are given in the following.

```

=====
===  NON-TIME-DEPENDENT SOLUTIONS : PRESTRESSING  ===
=====

105 - DISPLACEMENTS AT THE NODES
=====

LEVEL LEV1  SPAN   1
  NODE      U          V          ROT
    1  0.0000E+00  0.0000E+00  2.7310E-06
    2 -7.6142E-06  2.6549E-05  2.5026E-06

LEVEL LEV1  SPAN   2
  NODE      U          V          ROT
    2 -7.6142E-06  2.6549E-05  2.5026E-06
    3 -1.6025E-05  4.8263E-05  1.7614E-06

106 - TOTAL HYPERSTATIC (SECONDARY) REACTIONS AT FIXED NODES
=====

  NODE  X-FORCE  Y-FORCE  Z-MOMENT
    1  1.2129E+00 -9.7029E-01  0.0000E+00
    7  0.0000E+00 -1.7169E+00  0.0000E+00

TOTAL  1.2129E+00 -2.6872E+00  0.0000E+00

107 - FRAME ELEMENT ACTIONS
=====
107.1 PRIMARY ELEMENTS
-----

LEVEL LEV1  SPAN   1
  ELEMENT    NO          BENDING          BENDING          SHEAR          AXIAL
                   (END I)          (END J)          FORCE          FORCE
    1          1          -9.5497E-12 -4.6712E+04 -4.6712E+03 -5.8389E+03

LEVEL LEV1  SPAN   2
  ELEMENT    NO          BENDING          BENDING          SHEAR          AXIAL
                   (END I)          (END J)          FORCE          FORCE
    2          2          -5.1600E+04 -9.9975E+04 -4.8375E+03 -6.4500E+03

108 - EXTREME FIBER STRESSES IN CONCRETE
=====

LEVEL LEV1  SPAN   1
ELEMENT  I-TOP  I-BOT  J-TOP  J-BOT  AXIAL
    1  -2.9195E+00 -2.9195E+00  5.8389E-01 -6.4228E+00 -2.9195E+00

LEVEL LEV1  SPAN   2
ELEMENT  I-TOP  I-BOT  J-TOP  J-BOT  AXIAL
    2   6.4500E-01 -7.0950E+00  4.2731E+00 -1.0723E+01 -3.2250E+00

```

```

109 - TENDON FORCES
=====

109.1 INDIVIDUAL TENDON ACTIONS
-----

TENDON      1 TOTAL RESULTS
=====

  SEG  TOTAL      PERCENT  FRAME  BENDING  BENDING  SHEAR  AXIAL
  NO   FORCE          LOSS   EL NO  (END I)  (END J)  FORCE   FORCE

    1  7.479E+03    2.71     1    0.000E+00  4.672E+04  4.672E+03  5.840E+03
    2  8.066E+03    2.56     2    5.162E+04  1.000E+05  4.839E+03  6.453E+03
    3  9.480E+03    3.02     3    1.426E+05  1.655E+05  2.299E+03  9.197E+03

TENDON      1 TOTAL FORCES
=====

SEGMENT NO  - - - - - SEGMENT FORCES - - - - -
  1 - 6  7.479E+03  8.066E+03  9.480E+03  1.136E+04  1.277E+04  1.323E+04

109.2 COMBINED ACTIONS AND FORCES OF ALL TENDONS
-----

  Element End   Moment      Shear      Axial      Eccentricity
    1     I   0.0000E+00  4.6721E+03  5.8402E+03  0.0000E+00
        J   4.6721E+04  4.6721E+03  5.8402E+03  8.0000E+00
    2     I   5.1620E+04  4.8394E+03  6.4525E+03  8.0000E+00
        J   1.0001E+05  4.8394E+03  6.4525E+03  1.5500E+01
    3     I   1.4255E+05  2.2992E+03  9.1969E+03  1.5500E+01

109.3 HYPERSTATIC ACTIONS OF ALL TENDONS COMBINED
-----

LEVEL LEV1  SPAN      1

  ELEMENT      BENDING      BENDING      SHEAR      AXIAL
  NO           (END I)      (END J)      FORCE      FORCE
    1          2.7342E-11  1.4340E+05  1.4340E+04 -4.0000E+04

LEVEL LEV1  SPAN      2

  ELEMENT      BENDING      BENDING      SHEAR      AXIAL
  NO           (END I)      (END J)      FORCE      FORCE
    2          1.4340E+05  2.6944E+05  1.2604E+04 -4.0000E+04

```

## 5.6 COMBINATIONS

A loading combination can result in two possible types of printout.

- (i) If the components of the combination are all regular loading cases (without skipping), the combination output will also have the format of a regular printout as described in the preceding.
- (ii) When one of the components of the load combination is an envelope, the printout format of the combination will also be an envelope.

This Page Left Intentionally

BLANK

## 6.1 GENERAL

ADAPT-GEN is capable of treating frames in building, bridges and special structures. These can be pre-stressed or non-prestressed concrete; of composite construction; incrementally constructed, or be a complete construction. They can have time-dependent, or time-free response. In this chapter, three examples are selected to illustrate some of the capabilities of ADAPT-Gen. Example 1, is a simple post-tensioned frame, viewed as if constructed in one stage. Example 1 is a frame common to parking structures with split level. It is constructed in stages.

The focus of the examples is the features particular to ADAPT-GEN, such as handling of the frames as *structures*, load combinations and envelopes. In particular, in example 1 an attempt is made to illustrate all alternatives of load specification included in ADAPT-GEN. As a result, some of the loading conditions selected for the example may appear unrealistic, from a practical point of view.

## 6.2 EXAMPLE 1: EXTRACTION OF TIME-DEPENDENT EFFECTS

### 6.2.1 Description

From the total integrated solution obtained by ADAPT-ABI, the program ADAPT-GEN has the capability to separate the solution to its components due to “applied loading”, “prestressing,” and “long-term effects” of creep and shrinkage. The objective of this example is to illustrate the extraction of the actions and displacements due to creep and shrinkage.

The structure considered is a two span beam shown in **Figure 6.2.1-1**. The beam consists of two single span girders which are installed next to one another over the central support and made continuous through a cast-in-place splice. Loading is limited to self weight. Creep and shrinkage over time will result in a change in the distribution of actions in the structure.

The distribution of moment due to selfweight after 7300 days is shown in **Figure 6.2.1-2**. The moment due to selfweight at the time the girders were made continuous was as shown in **Figure 6.2.1-4**. The moments generated due to the creep and shrinkage of the girders are shown in **Figure 6.2.1-5**. Note that the moment due to creep and shrinkage are comparable in magnitude to those due to selfweight. In this example due to roller end supports shrinkage does not cause moments. Hence **Figure 6.2.1-5** reflects the effects of creep only.

**Figure 6.2.1-3** shows the combination of moments due to selfweight and long-term effects obtained from the last stage of ADAPT-ABI and shown in **Figure 6.2.1-2**.

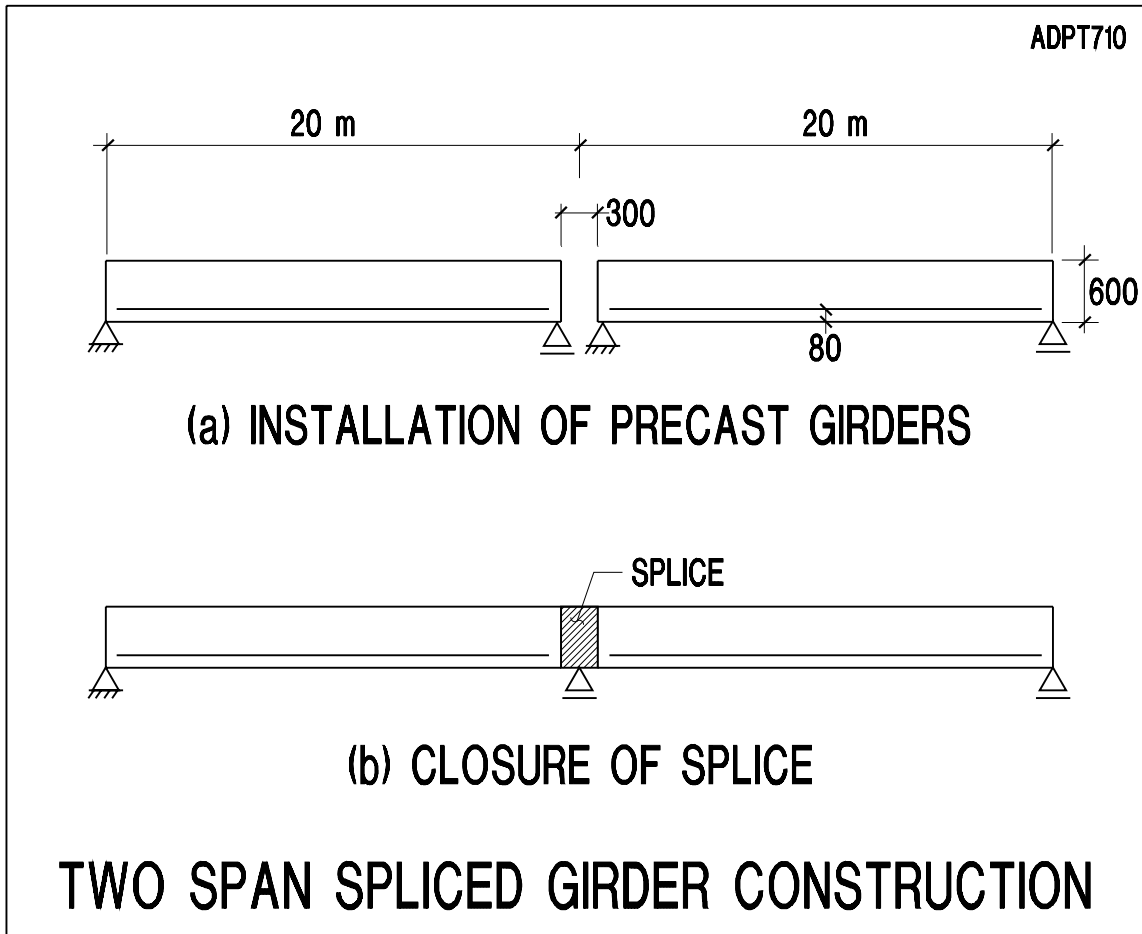


FIGURE 6.2.1-1

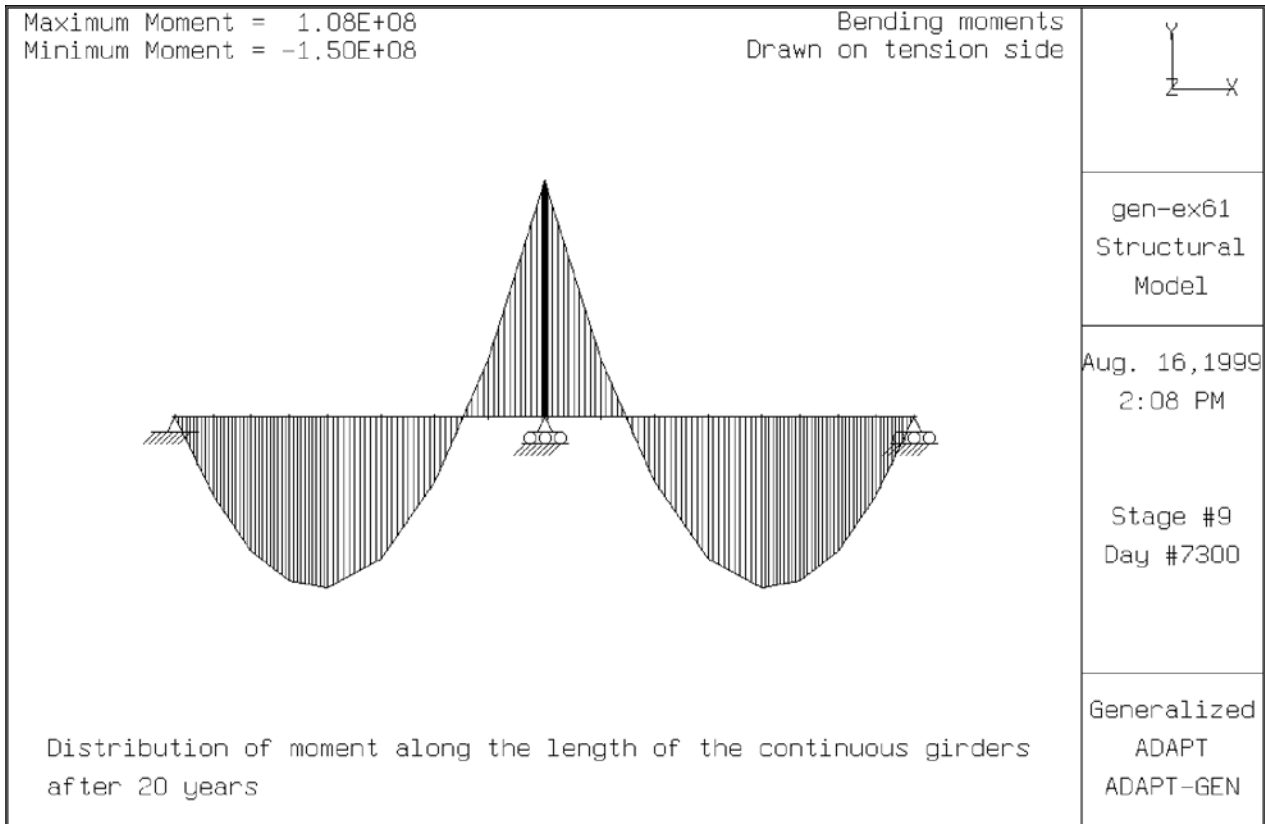


FIGURE 6.2.1-2

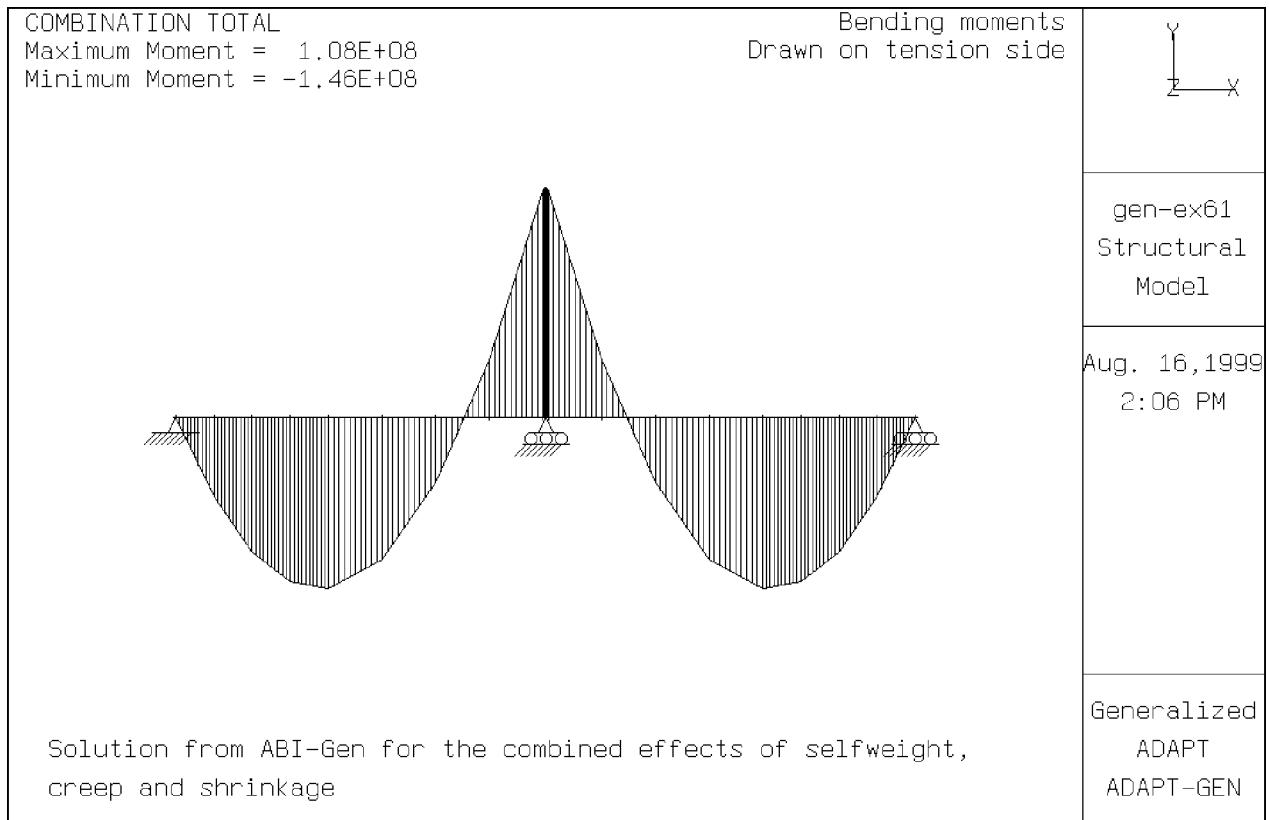


FIGURE 6.2.1-3

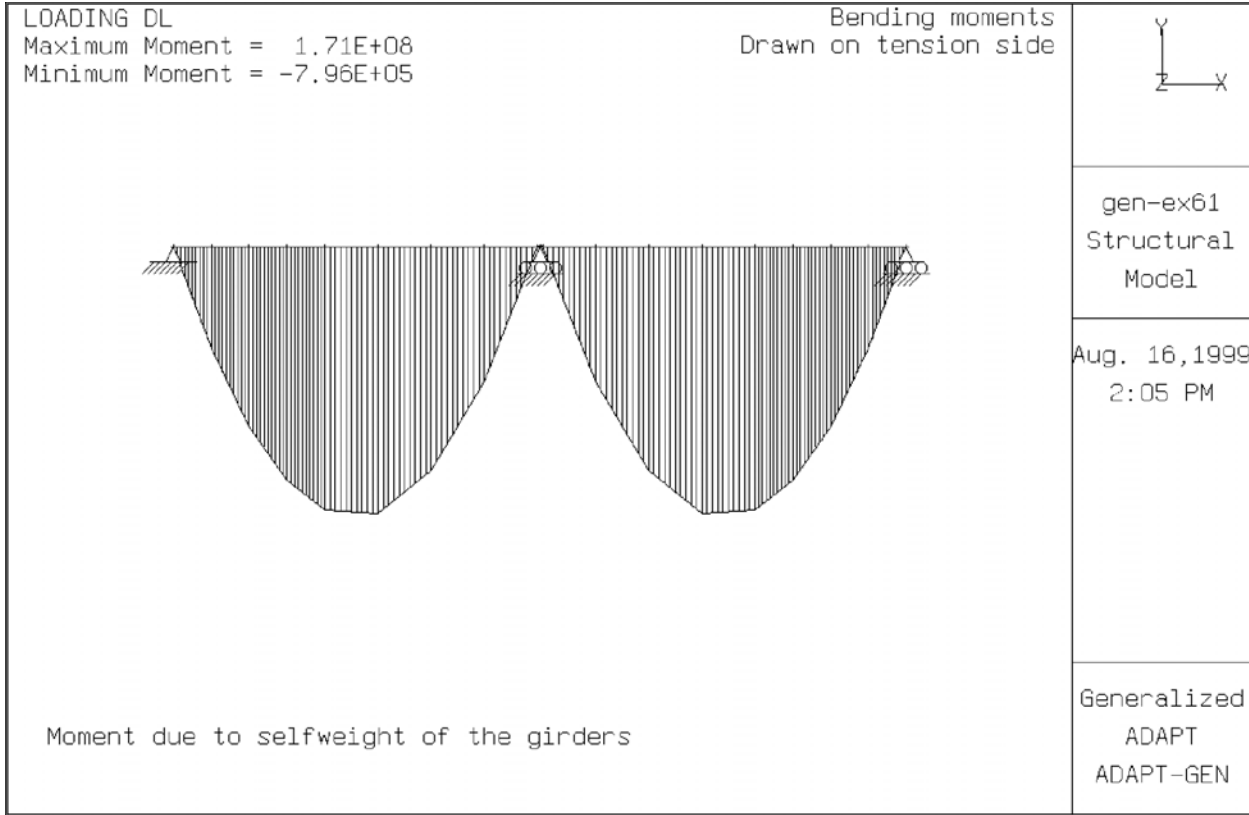


FIGURE 6.2.1-4

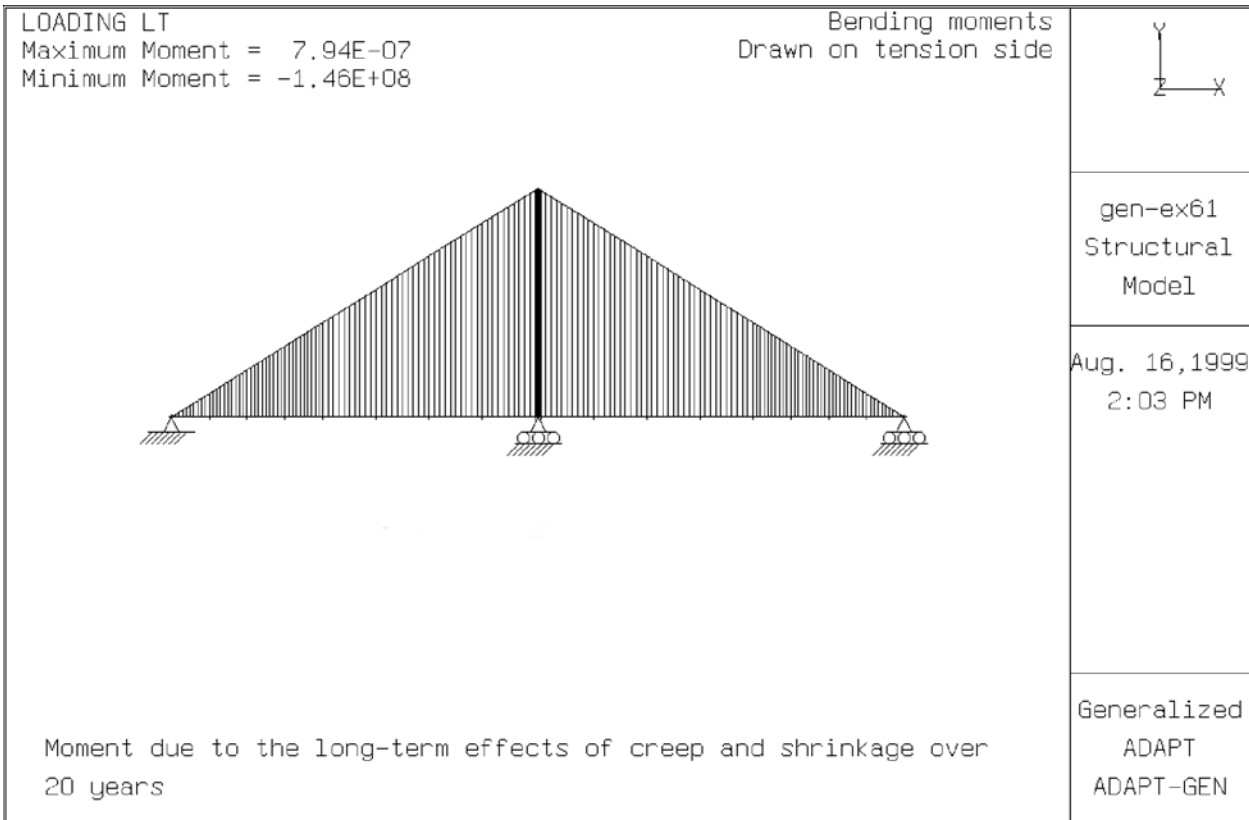


FIGURE 6.2.1-5



## 6.2.2 Input Data

```

;=====
;                               ADAPT ABI-GEN
;       For incrementally constructed bridges and frames
;=====
;   File Name   : GEN-EX61.inp
;   Last Update : June 25, 1999
;   Units in    : N-mm

; This example illustrates the modeling of precast girders spliced
; for continuity. It is also used. It also illustrates the extraction
; of effects of creep and shrinkage from the integrated solution.

START
TITLE N=12

      This is a simplified example of a two-span bridge
      constructed with precast girders. Prestressing is not
      included. The girders are made continuous over the central
      support through a splice joint. Then, a post-tensioned cable
      passing through both spans completes the continuity.
      The focus of the example is the sequence of construction,
      extraction of effects due to shrinkage and creep, and the
      verification that the solution components computed by ABI-Gen
      add up to the incrementally constructed scenario.

UNITS U=SI

ACTIVATE EXTRACT

CONCRETE PARAMETERS ! 1 M=ACI

MESH INPUT

NODES N=19
  1 X=0.0           Y=0.0           ; Left span, left support
  5 X=8.28*1000     Y=0.0 G=1,5,1   ; Left support to pt vertex
  9 X=20.0-0.15*1000 Y=0.0 G=5,9,1   ; Pt vertex to right support
 10 X=20.0*1000     Y=0.0           ; Mid support
 11 X=20.0+0.15*1000 Y=0.0         ; Right span, left support
 15 X=40.0-8.28*1000 Y=0.0 G=11,15,1 ; Left support to pt vertex
 19 X=40.0*1000     Y=0.0 G=15,19,1 ; Pt vertex to right support

SECTION PROPERTIES N=1
  1 B=250.0 D=600.0           ; Rectangular

CONCRETE PROPERTIES N=2
  1 FPC=45.00 CR=2.5 SH=0.0004 W=0.0000024 Ac=0.0 ; Precast
  2 FPC=35.00 CR=2.5 SH=0.0004 W=0.0000024 Ac=0.0 ; Splice

MILD STEEL PROPERTIES N=1
  1 Es=200000 P=0.0           ; no passive steel

ELEMENTS N=18
FRAME N=18
  1, 1, 2 C=1 X=1 Day=0 ST=1 G= 1, 8,1,1,1,0,0,0,0,0,0 ; Left girder
  9, 9,10 C=2 X=1 Day=0 ST=1 G= 9,10,1,1,1,0,0,0,0,0,0 ; Splice
 11,11,12 C=1 X=1 Day=0 ST=1 G=11,18,1,1,1,0,0,0,0,0,0 ; Right girder

```

MESH COMPLETE

```

; .....Construction phase .....
SET Day=0 G=0,-1 ; Cast girders in bed

; ..... cast concrete .....
CHANGE STRUCTURE
  BUILD N=1,8 Day=0 ; Cast span 1 at day=0
  BUILD N=11,18 Day=0 ; Cast span 2 at day=0
  RESTRAINTS
    1 R=1,1,0 ! 9 R=0,1,0
    11 R=1,1,0 ! 19 R=0,1,0

CHANGE COMPLETE
SOLVE Day=3 ! OUTPUT ; Update stress status in

SOLVE DAY=14 ! OUTPUT

SOLVE Day=32 ; update stresses to Day 32
CHANGE STRUCTURE
  BUILD N=9,10 ; build splice segment
  RESTRAINTS
    9 R=0,0,0 ; adjust support over central
    11 R=0,0,0 ; pier
    10 R=0,1,0

CHANGE COMPLETE
SOLVE ! OUTPUT

SOLVE Day=50 ! OUTPUT
SOLVE Day=100 ! OUTPUT
SOLVE Day=200 ! OUTPUT
SOLVE Day=365 ! OUTPUT
SOLVE Day=365*10 ! OUTPUT
SOLVE Day=365*20 ! OUTPUT

; ..... extract components of actions for load combination
; ..... and service stress checks

CAPTURE STATUS

LOADING DL
  DEAD LOAD ; Dead load actions
SOLVE
OUTPUT

LOADING LT
  LONG TERM
SOLVE
OUTPUT

COMBINATIONS
  COMBINATION TOTAL ; DL+LT Service
  1*DL+1*LT

STOP

```

### 6.3 EXAMPLE 1: TWO-STORY POST-TENSIONED FRAME

#### 6.3.1 Description

The elevation of the two-story plane frame is given in **Figure 6.3-1**. The support conditions are fixed (Grid A), one fixed hinge (Grid B); one on elastic springs (Grid D) (**Figure 6.3.2(a)**).

The cross sectional properties of the frame and the material used are given in the following tables. These are identified in **Figure 6.3-2(b)**.

#### A. Cross-sectional properties:

Type	Area (in <sup>2</sup> )	Moment of Inertia (in <sup>4</sup> )	C <sub>top</sub>	C <sub>bottom</sub>
X1	400	14,000	9	11
X2	300	10,000	7	13
X3	600	50,000	11	29
X4	500	30,000	7	19

#### B. Material Properties:

##### Concrete:

Designation	f <sup>'</sup> c (psi)	Creep coefficient	Shrinkage coefficient
C1	6000	3.0	0.0004
C2	5000	2.5	0.0003
C3	4000	2.0	0.06

Unit weight of concrete:  $W = 155/1728 \text{ lb/ft}^3$

Thermal expansion coefficient:  $A_c = 5.5E-06$

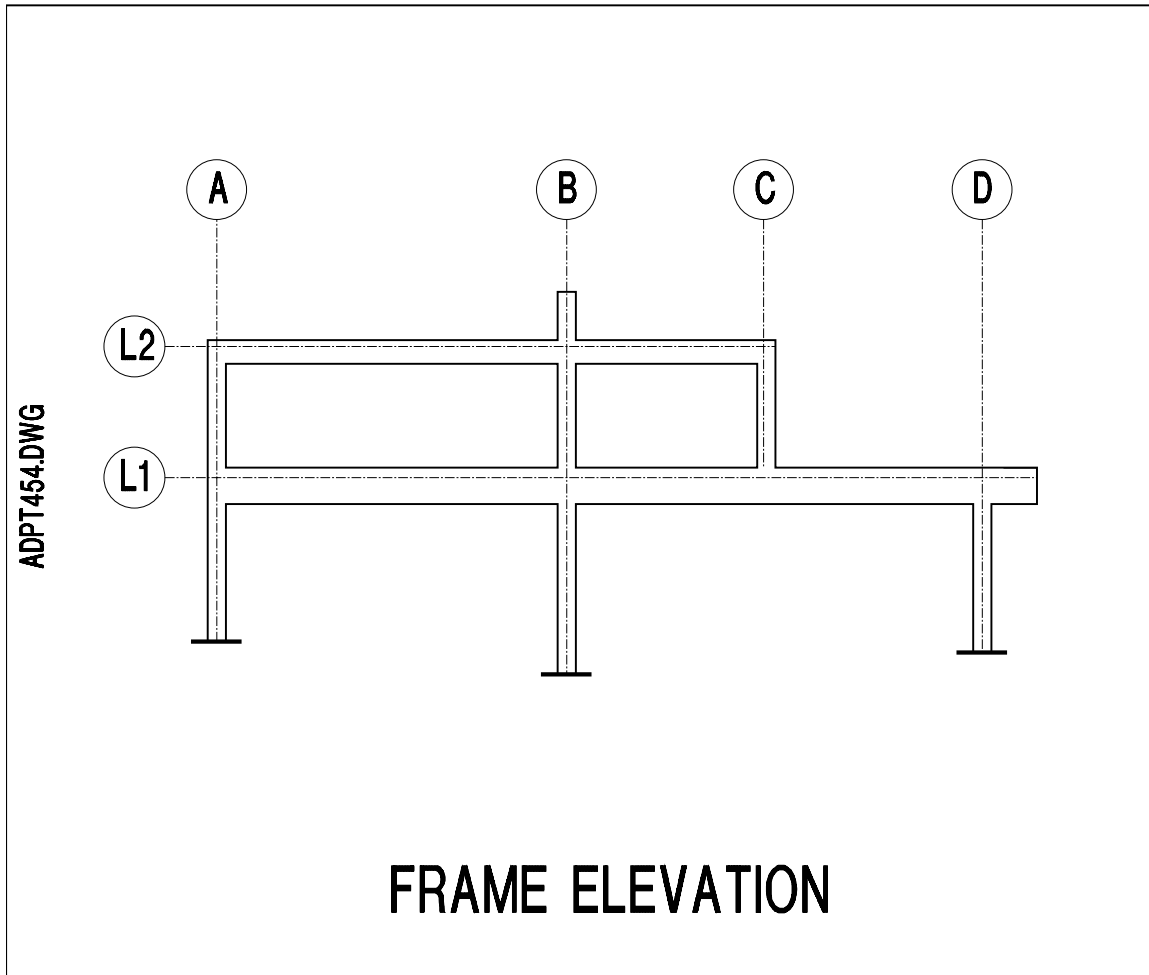


FIGURE 6.3-1

**Steel:**

Modulus Elasticity:  $E_s = 29,000 \text{ ksi}$   
 Thermal expansion coefficient:  $A_s = 0.00006$

**Springs:**

Horizontal spring:  $K = 100,000 \text{ lb/in}$   
 Vertical spring:  $K = 100,000 \text{ lb/in}$

**C. Prestressing:**

The beams are prestressed each with a continuous profiled tendon. In addition the column at line C is prestressed with a single tendon located at centroid of the column. An overview of the tendon layout is given in **Figure 6.3-3**. The tendons are identified as tendons T1 through T3. The parameters defining the geometry of each tendon are also entered in the diagram. For additional details on the geometry parameters, R, and, S, refer to ADAPT-ABI program manual.

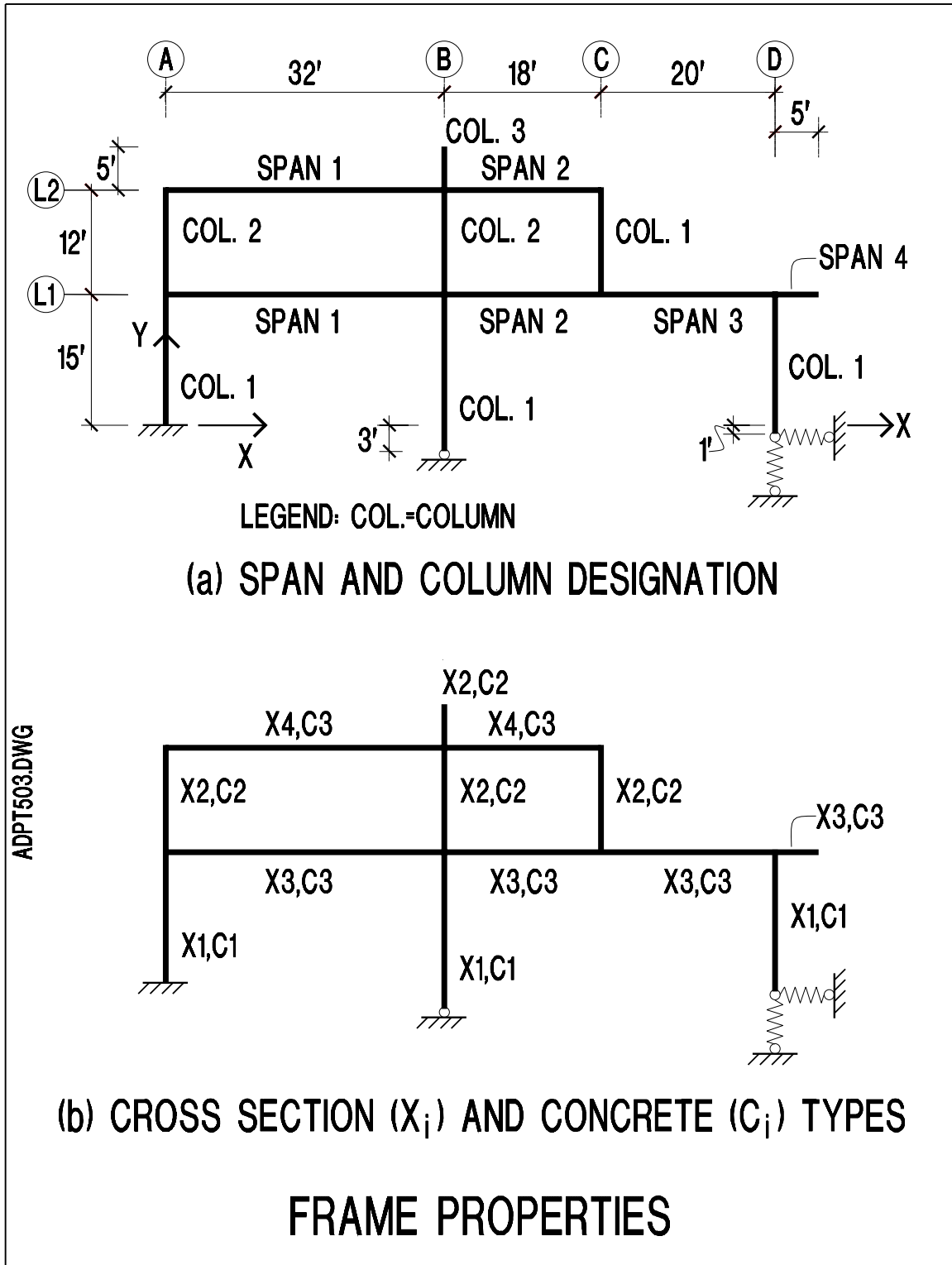


FIGURE 6.3-2

PT Angular friction coefficient = 0.2  
 PT Wobble friction coefficient = 0.0002  
 PT Ultimate stress = 270,000 lb/in

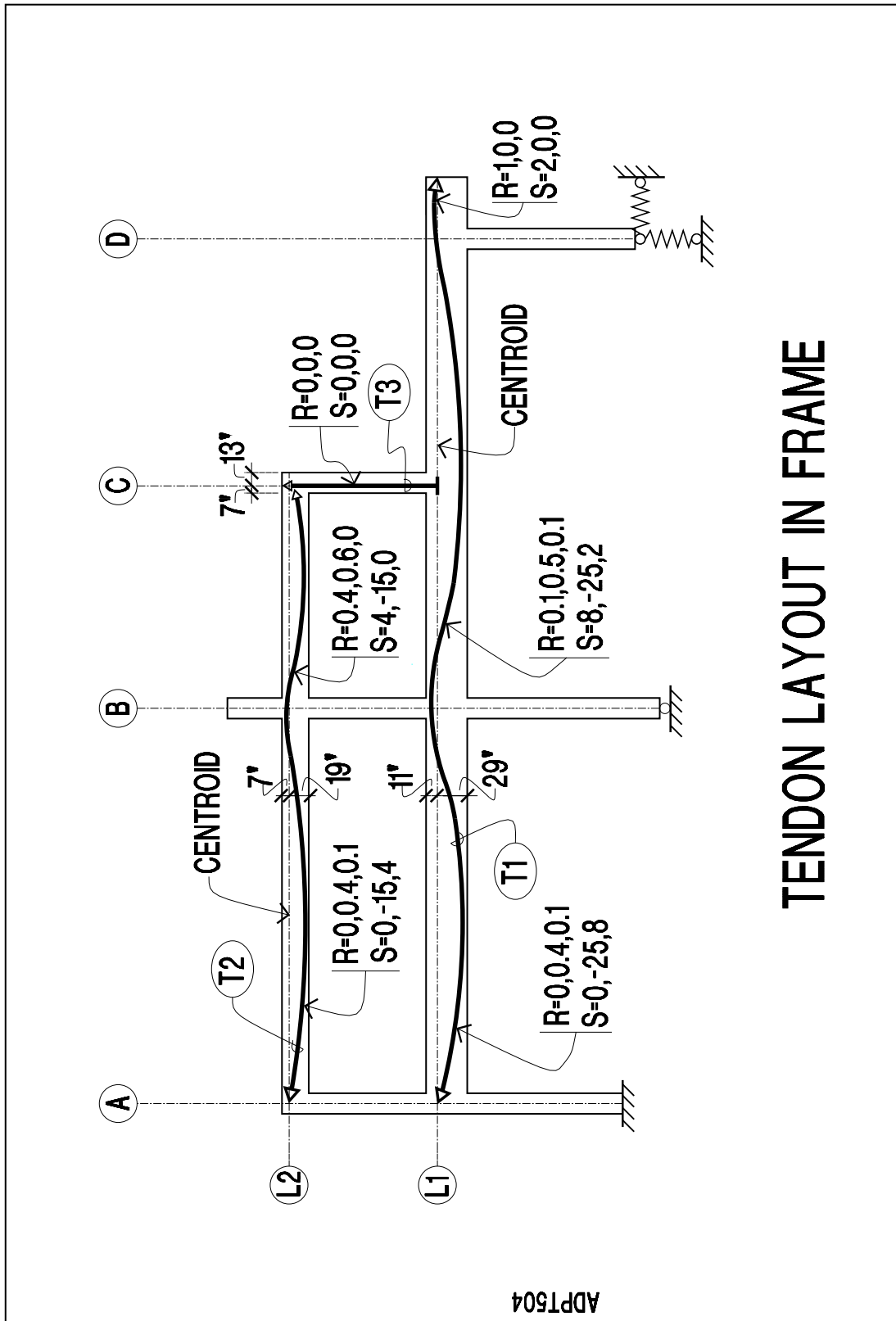


FIGURE 6.3-3

Tendon type	Area (in <sup>2</sup> )
T1	2.659
T2	1.839
T3	1.169

**D. Loading:**

## 1 - Dead loading

Designation	Loading
Level 1, Span 1	2.00 kips/ft
Level 1, Span 2	0.75 kips/ft
Level 1, Span 3	0.75 kips/ft
Level 1, Span 4	0.70 kips/ft
Level 2, Span 1	0.90 kips/ft
Level 2, Span 2	0.80 kips/ft

## 2 - Prestressing

Tendon type	Jacking stress ratio	Anchorage Slip
T1	0.85	0.25
T2	0.85	0.25
T3	0.85	0.25

## 3 - Live loading

## Uniform live load (LIVE1)

Designation	Loading
Level 1, Span 1	1.00 kips/ft
Level 1, Span 2	0.85 kips/ft
Level 1, Span 3	0.85 kips/ft
Level 1, Span 4	0.85 kips/ft
Level 2, Span 1	1.20 kips/ft
Level 2, Span 2	1.30 kips/ft

## Alternative live loading (LIVE2)

Designation	Loading
Level 1, at Node 46	8.00 kips
Level 1, at Node 48	4.00 kips
Level 1, Span 2 at XB=6' XE=12'	0.30 kips/ft
Level 1 at Node 61	4.00 kips
Level 2, at Node 71	1.20 kips/ft
Level 2, at Node 78	1.30 kips/ft

Live loading is skipped to maximize values. A number of envelope and combination loading cases are selected, as shown in the input data, to illustrate the load specification.

4 - Lateral loading:

Earthquake load

Designation	Loading
At Node 6	6 kips
At Node 12	12 kips
At Node 28	1 kips

Wind load

Designation	Loading
Grid AA, Column 1 & 2	0.30 kips/ft
Grid CC, Column 1	0.20 kips/ft
Grid BB, at Node 28	1.00 kips

A summary of frame loading described above is shown in **Figure 6.3-4**. A finite element discretization of the frame is modeled as the one shown in **Figure 6.3-5**.



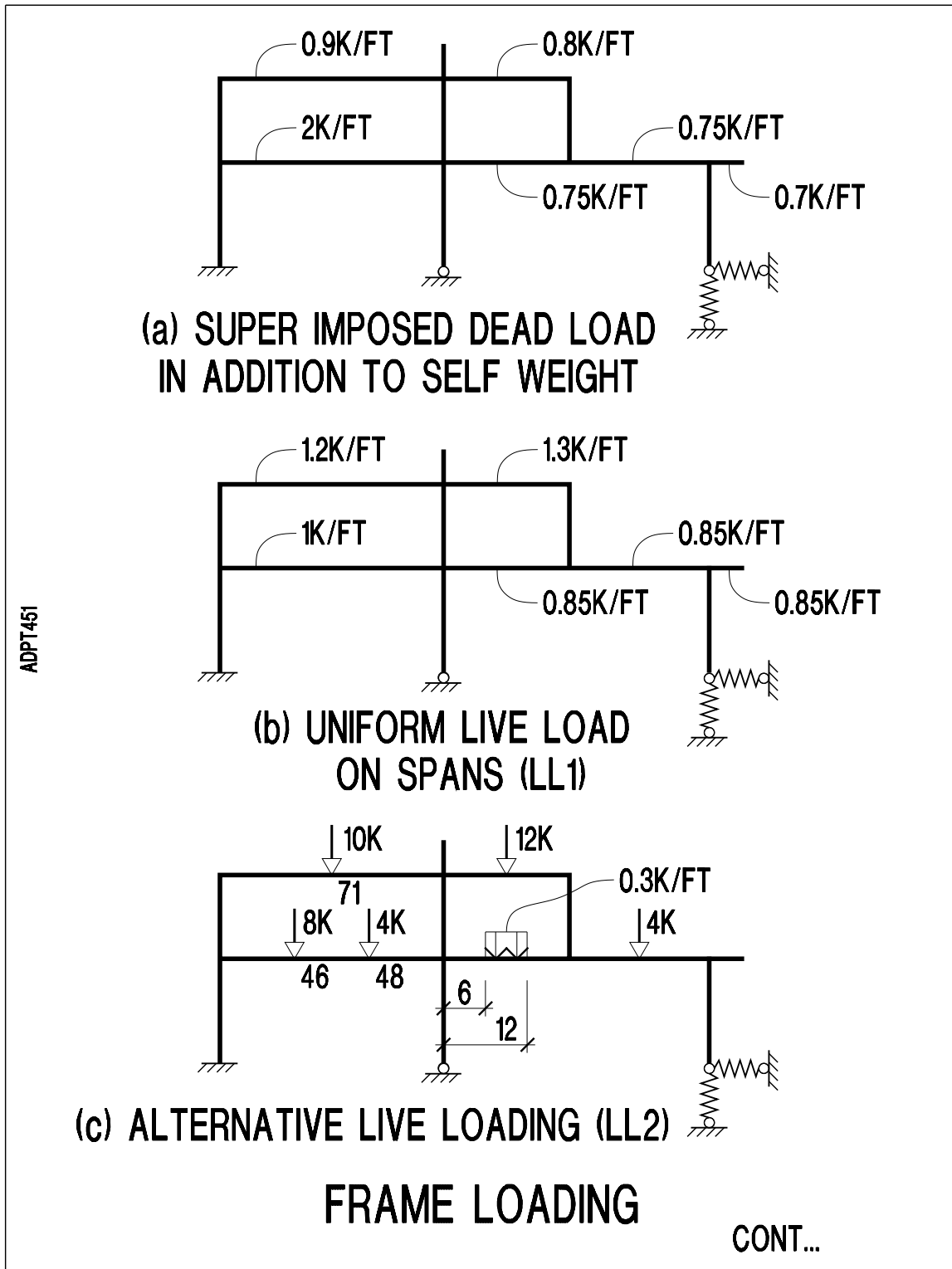


FIGURE 6.3-4

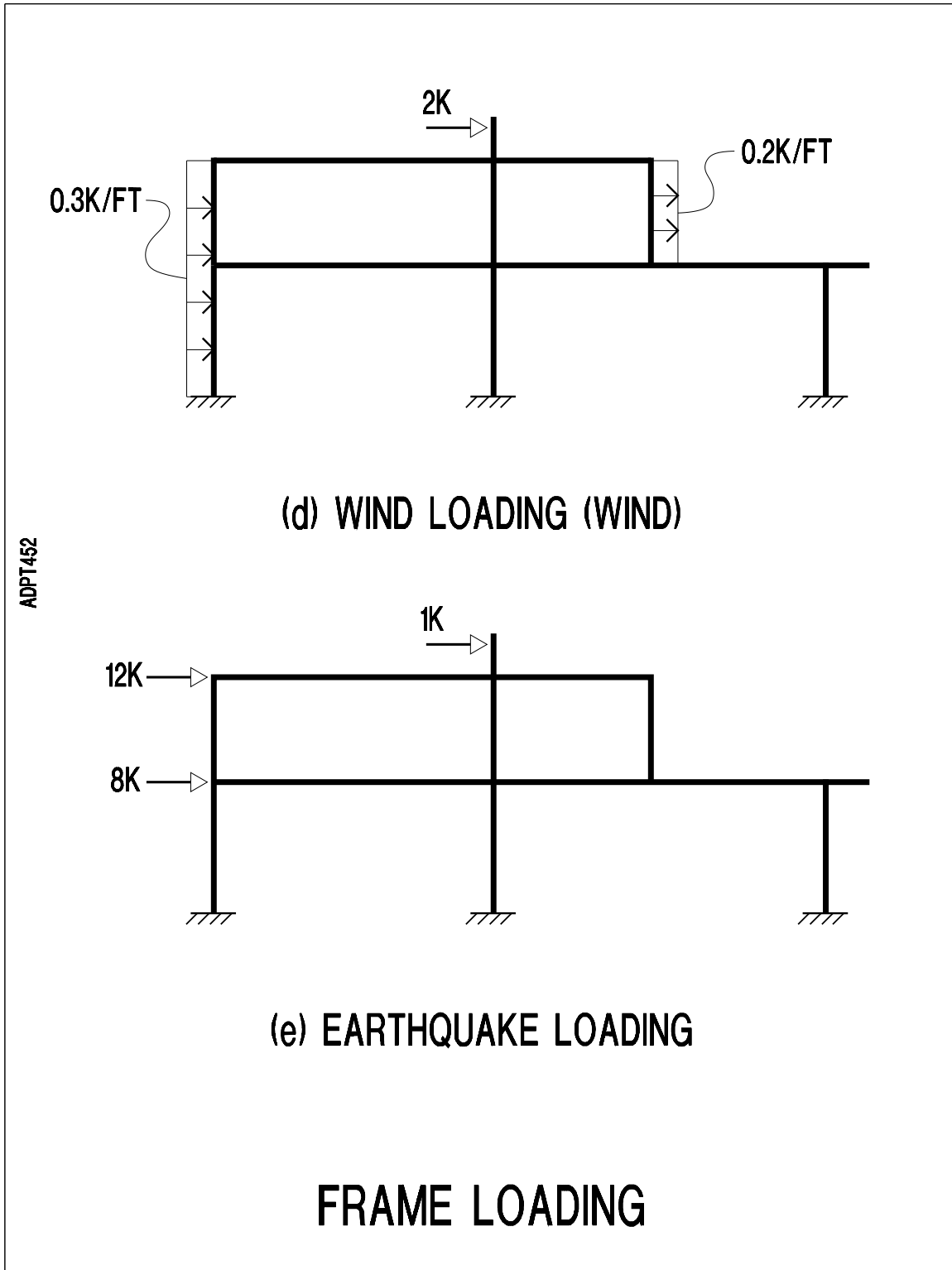
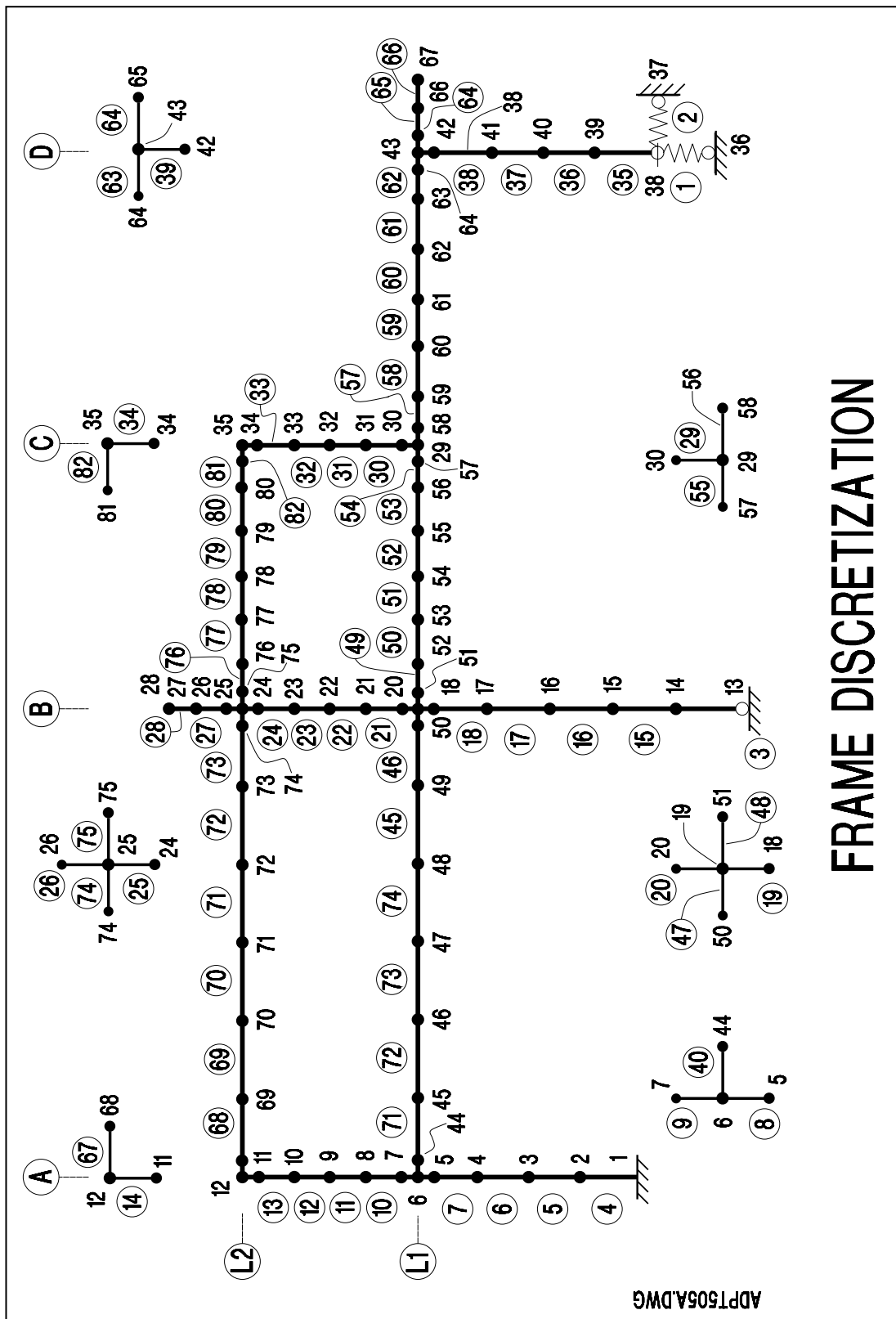


FIGURE 6.3-4



FRAME DISCRETIZATION

FIGURE 6.3-5

### 6.3.2 Input Data

```

;=====
; ADAPT-BRIDGE-INCREMENTAL  ABI  SOFTWARE MANUAL EXAMPLES
;=====
; name of this file: GEN-ex62.INP

START
TITLE N=1
      TWO-STORY POST-TENSIONED FRAME

UNITS  U=USA

CONCRETE PARAMETERS N=1
      1 M=ACI a=0 b=1 W=155/1728 ; suppress concrete aging

MESH INPUT
      NODES N=81
          1 X=0      Y=0
          5 X=0      Y=151  G=1,5,1
          6 X=0      Y=180
          7 X=0      Y=191
         11 X=0      Y=305  G=7,11,1
         12 X=0      Y=324
         13 X=384   Y=-36.18
         14 X=384   Y=-36
         18 X=384   Y=151  G=14,18,1
         19 X=384   Y=180
         20 X=384   Y=191
         24 X=384   Y=305  G=20,24,1
         25 X=384   Y=324
         26 X=384   Y=331
         28 X=384   Y=384  G=26,28,1
         29 X=600   Y=180
         30 X=600   Y=191
         34 X=600   Y=305  G=30,34,1
         35 X=600   Y=324
         36 X=840   Y=-72
         37 X=900   Y=-12
         38 X=840   Y=-12
         42 X=840   Y=151  G=38,42,1
         43 X=840   Y=180
         44 X=11    Y=180
         50 X=375   Y=180  G=44,50,1
         51 X=395   Y=180
         57 X=593   Y=180  G=51,57,1
         58 X=613   Y=180
         64 X=831   Y=180  G=58,64,1
         65 X=851   Y=180
         67 X=900   Y=180  G=65,67,1
         68 X=13    Y=324
         74 X=377   Y=324  G=68,74,1
         75 X=397   Y=324
         81 X=593   Y=324  G=75,81,1

```

```

CONCRETE PROPERTIES N=3
  1 Fpc=6000 Cr=3.0 Sh=0.0004 W=155/1728 Ac=0.0000055
  2 Fpc=5000 Cr=2.5 Sh=0.0003 W=155/1728 Ac=0.0000055
  3 Fpc=4000 Cr=2.0 Sh=0.006 W=155/1728 Ac=0.0000055
MILD STEEL PROPERTIES N=1
  1 Es=29000000 P=.02 As=0.00006
SECTION PROPERTIES N=4
  1 Area=400 I=14000 C=9,11 ;level 1 column property
  2 Area=300 I=10000 C=7,13 ;level 2 column property
  3 Area=600 I=50000 C=11,29 ;level 1 span property
  4 Area=500 I=30000 C=7,19 ;level 2 span property
ELEMENTS N=82
FRAME N=80
  3,13,14 C=1 X=1 St=1 Day=1
  4, 1 ,2 C=1 X=1 St=1 Day=1 G=4,8,1,1,1
  9, 6, 7 C=2 X=2 St=1 Day=1 G=9,14,1,1,1
 15,14,15 C=1 X=1 St=1 Day=1 G=15,19,1,1,1
 20,19,20 C=2 X=2 St=1 Day=1 G=20,28,1,1,1
 29,29,30 C=2 X=2 St=1 Day=1 G=29,34,1,1,1
 35,38,39 C=1 X=1 St=1 Day=1 G=35,39,1,1,1
 40,6,44 C=3 X=3 St=1 Day=1
 41,44,45 C=3 X=3 St=1 Day=1 G=41,46,1,1,1
 47,50,19 C=3 X=3 St=1 Day=1
 48,19,51 C=3 X=3 St=1 Day=1
 49,51,52 C=3 X=3 St=1 Day=1 G=49,54,1,1,1
 55,57,29 C=3 X=3 St=1 Day=1
 56,29,58 C=3 X=3 St=1 Day=1
 57,58,59 C=3 X=3 St=1 Day=1 G=57,62,1,1,1
 63,64,43 C=3 X=3 St=1 Day=1
 64,43,65 C=3 X=3 St=1 Day=1
 65,65,66 C=3 X=3 St=1 Day=1 G=65,66,1,1,1
 67,12,68 C=3 X=4 St=1 Day=1
 68,68,69 C=3 X=4 St=1 Day=1 G=68,73,1,1,1
 74,74,25 C=3 X=4 St=1 Day=1
 75,25,75 C=3 X=4 St=1 Day=1
 76,75,76 C=3 X=4 St=1 Day=1 G=76,81,1,1,1
 82,81,35 C=3 X=4 St=1 Day=1
SPRINGS N=2
  1,36,38 K=100000
  2,37,38 K=100000
PRESTRESSING STEEL N=1
  1 Ep=2.8E7 Meu=0.2 K=0.0002 Fpu=270000 Fpy=256000 \
    R=45 Ap=0.0
TENDON GEOMETRY N=3
  1 Spans=3 M=1 Area=2.659
    1 N=9 B=0,180 E=384,180
      List=6,44,45,46,47,48,49,50,19
      R=0.0,0.4,0.1 S=0.0,-25,8
    2 N=17 B=384,180 E=840,180
      List=19,51,52,53,54,55,56,57,29,58,59,60,61,\
        62,63,64,43
      R=0.1,0.5,0.1 S=8,-25,2
    3 N=4 B=840,180 E=900,180
      List=43,65,66,67

```

```

      R=0.0,1.0,0.0 S=2,0.0,0.0
2 Spans=2 M=1 Area=1.839
  1 N=9 B=0,324 E=384,324
    List=12,68,69,70,71,72,73,74,25
    R=0.0,0.4,0.1 S=0.0,-15,4
  2 N=9 B=384,324 E=600,324
    List=25,75,76,77,78,79,80,81,35
    R=0.1,0.6,0.0 S=4,-15,0.0
3 Spans=1 M=1 Area=1.169
  1 N=7 G=29,35,1 B=600,180 E=600,324
    R=0.0,0.0,0.0 S=0.0,0.0,0.0

```

MESH COMPLETE

STRUCTURE DEFINITION

```

GRIDS 4 LEVELS 2
  GRID AA
    COLUMN 1 E=4,7,1
    COLUMN 2 E=10,13,1
  GRID BB
    COLUMN 1 E=15,18,1
    COLUMN 2 E=21,24,1
    COLUMN 3 E=27,28,1
  GRID CC
    COLUMN 1 LIST=30,31,32,33
  GRID DD
    COLUMN 1 E=35,38,1
  LEVEL LEV1
    SPAN 1 E=41,46,1
    SPAN 2 E=49,54,1
    SPAN 3 E=57,62,1
    SPAN 4 LIST=65,66
  LEVEL LEV2
    SPAN 1 E=68,73,1
    SPAN 2 E=76,81,1

```

STRUCTURE COMPLETE

CHANGE STRUCTURE

```

BUILD N=4,8,1 Day=7
BUILD N=3 Day=7
BUILD N=15,19,1 Day=7
BUILD N=1,2,1 Day=7
BUILD N=35,39,1 Day=7

```

RESTRAINTS

```

  1 R=1,1,1
 13 R=1,1,0
 14 R=1,1,0
 36 R=1,1,0
 37 R=1,1,0

```

CHANGE COMPLETE

SOLVE Day=7

OUTPUT

```

CHANGE STRUCTURE
  BUILD N=40,47,1 Day=14
  BUILD N=48,55,1 Day=14
  BUILD N=56,63,1 Day=14
  BUILD N=66,64,-1 Day=14
  STRESS N=1 Ratio=0.85 Anchor=0.25,0.25

CHANGE COMPLETE
SOLVE Day=14 Steps=8
OUTPUT

CHANGE STRUCTURE
  BUILD N=10,14,1 Day=18
  BUILD N=21,28,1 Day=18
  BUILD N=30,34,1 Day=18
  BUILD N=67,74,1 Day=25
  BUILD N=75,82,1 Day=25
  BUILD N=9 Day=18
  BUILD N=20 Day=18
  BUILD N=29 Day=18
  STRESS N=2 Ratio=0.85 Anchor=0.25,0.25
  STRESS N=3 Ratio=0.85 Anchor=0.25,0.25

CHANGE COMPLETE
SOLVE Day=18
OUTPUT

SOLVE Day=45
OUTPUT

SOLVE Day=180 Steps=2
OUTPUT

SOLVE Day=365 Steps=2
OUTPUT

SOLVE Day=730 Steps=5
OUTPUT

CAPTURE STATUS
LOADING DL ; Superimposed Dead load
  STRUCTURE
    LEVEL LEV1
      SPAN 1
        UNIFORM F=0,-2000/12
      SPAN 2
        UNIFORM F=0,-750/12
      SPAN 3
        UNIFORM F=0,-750/12
      SPAN 4
        UNIFORM F=0,-700/12
    LEVEL LEV2
      SPAN 1

```

```

        UNIFORM  F=0,-900/12
    SPAN 2
        UNIFORM  F=0,-800/12

```

```

SOLVE
OUTPUT

```

```

LOADING LIVE1                ; Liveload 1

```

```

    PATTERN

```

```

        LEVEL LEVEL1

```

```

            SPAN 1

```

```

                UNIFORM  F=0,-1000/12

```

```

            SPAN 2

```

```

                UNIFORM  F=0,-850/12

```

```

            SPAN 3

```

```

                UNIFORM  F=0,-850/12

```

```

            SPAN 4

```

```

                UNIFORM  F=0,-850/12

```

```

        LEVEL LEVEL2

```

```

            SPAN 1

```

```

                UNIFORM  F=0,-1200/12

```

```

            SPAN 2

```

```

                UNIFORM  F=0,-1300/12

```

```

    SKIP

```

```

        LEVEL LEVEL1 SPANS 1 0 1 0

```

```

        LEVEL LEVEL2 SPANS 0 1

```

```

    SKIP

```

```

        LEVEL LEVEL1 SPANS 0 1 0 1

```

```

        LEVEL LEVEL2 SPANS 1 0

```

```

    SKIP

```

```

        LEVEL LEVEL1 SPANS 1 1 1 1

```

```

        LEVEL LEVEL2 SPANS 1 1

```

```

    SKIP

```

```

        LEVEL LEVEL1 SPANS 1 1 0 1

```

```

        LEVEL LEVEL2 SPANS 1 1

```

```

SOLVE
OUTPUT

```

```

LOADING LIVE2                ; Liveload 2

```

```

    STRUCTURE

```

```

        LEVEL LEVEL1

```

```

            SPAN 1

```

```

                CONCENTRATED F=0,-8000,0  AT N=46

```

```

                CONCENTRATED F=0,-4000,0  AT N=48

```

```

            SPAN 2

```

```

                UNIFORM      F=0,-300/12  XB=72 XE=144

```

```

            SPAN 3

```

```

                CONCENTRATED F=0,-4000,0  AT N=61

```

```

        LEVEL LEVEL2

```

```

            SPAN 1

```

```

                CONCENTRATED F=0,-10000,0 AT N=71

```

```

            SPAN 2

```

```

                CONCENTRATED F=0,-12000,0 AT N=78

```



SOLVE  
OUTPUT

LOADING EQKE ; Earthquake  
GENERIC  
N=6 F=8000,0,0  
N=12 F=12000,0,0  
N=28 F=1000,0,0

SOLVE  
OUTPUT

LOADING WIND ; Wind  
GENERIC  
N=28 F=2000,0,0  
STRUCTURE  
GRID AA  
COLUMN ALL  
UNIFORM F=300/12,0  
GRID CC  
COLUMN 1  
UNIFORM F=200/12,0

SOLVE  
OUTPUT

LOADING PT ; Prestressing load  
PRESTRESSING

SOLVE  
OUTPUT

LOADING DEAD ; Selfweight/Dead loading  
DEAD LOAD

SOLVE  
OUTPUT

LOAD LT  
LONG TERM

SOLVE  
OUTPUT

LOADING TEMP ; Temperature loading  
STRUCTURE  
LEVEL LEV2  
SPAN 1  
TEMPERATURE L=68,73,1 T=110,85  
SPAN 2  
TEMPERATURE L=76,81,1 T=110,85

SOLVE  
OUTPUT

```

COMBINATIONS
  COMBINATION ONE           ; DL+LL1
    1.4*DEAD+1.4*PT+1.7*LIVE1
OUTPUT

  COMBINATION TWO           ; DL+LL1+WIND
    1.4*DEAD+1.4*PT+1.4*WIND+1.7*LIVE1
  COMBINATION THREE        ; DL+LL2+WIND
    1.4*DEAD+1.4*PT+1.4*WIND+1.7*LIVE2
    ENVELOPE A
      ENVELOPE TWO,THREE
OUTPUT

  COMBINATION FOUR          ; EQKE+DL
    0.9*DEAD+0.9*PH+1.7*EQKE
    ENVELOPE B
      ENVELOPE ONE,THREE

OUTPUT

STOP

```

## 6.4 EXAMPLE 3: THREE STORY PARKING STRUCTURE SPLIT FRAME

### 6.4.1 Description

The elevation of the three-story split frame, which is constructed in three stages, is given in **Figure 6.4-1**. The three supports of the frame are modeled as fixed.

The beams in each level are prestressed with a simple parabola tendon. **Figure 6.4-2** shows the tendon arrangement of a typical level. The discretization and structural design models of the frame is shown in **Figure 6.4-3**.

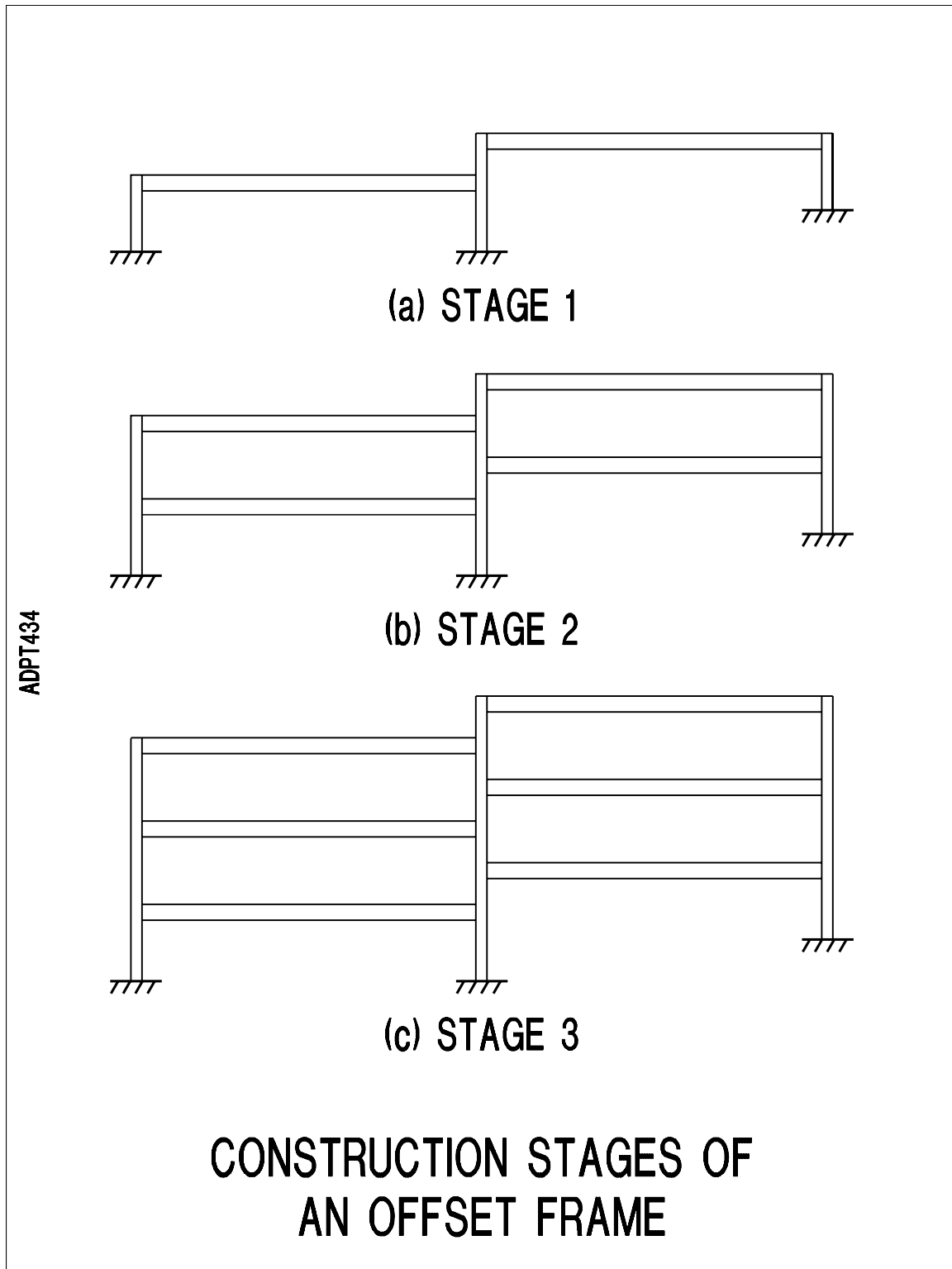


FIGURE 6.4-1

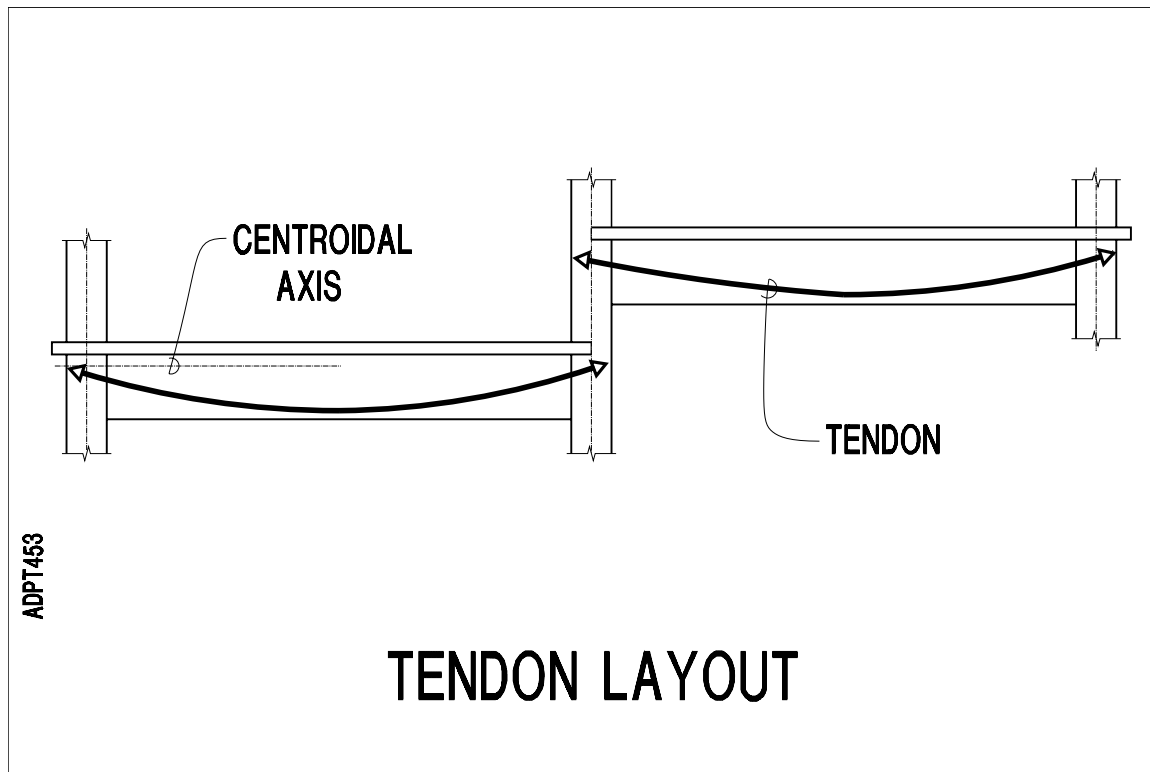


FIGURE 6.4-2

Live loading is skipped to maximize values. A number of envelope and combination loading cases are selected, as indicated in the input data.

The structural model selected is shown in **Figure 6.4-3**. The specifics of the model are:

- (i) The middle column at short intervals between the levels is modeled with essentially zero moment of inertia to simulate hinge supports at connection to the beams. This would necessitate either construction of a hinge detail at connection to the beam, or tightly placed ties in the column to increase confinement and accommodate the ensuing rotations.
- (ii) The upper level exterior columns are modeled with an effective moment of inertia equal to one-quarter of the actual value. This would result in additional reinforcement in the beam span. Ties, tightly spaced at the top of the column should be provided to retain the integrity of the column while allowing joint rotation at this location.

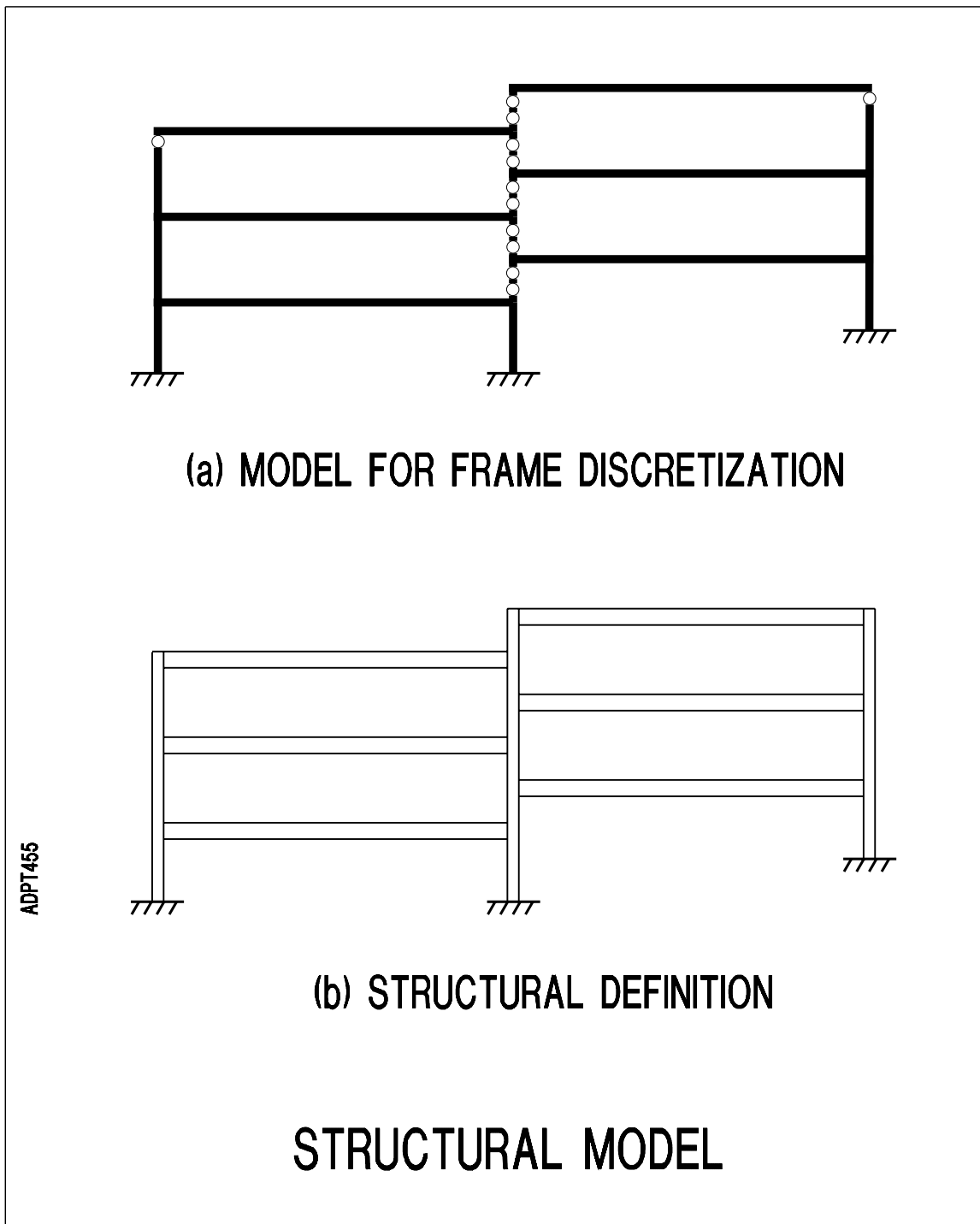


FIGURE 6.4-3

## 6.4.2 Input Data

```

;=====
; ADAPT-BRIDGE-INCREMENTAL  ABI  SOFTWARE MANUAL EXAMPLES
;=====
; name of this file: GEN-EX63.INP
;
; The parking structure frame is three split levels. Each T-
; beam is a 17' wide x 30 inches deep, with 5.5 thick flange,
; and a 16" wide beam stem. Live load is 50 psf.
;
;

START
TITLE N=1
      THREE STORY PARKING STRUCTURE SPLIT FRAME

ACTIVATE EXTRACT
UNITS U=USA

CONCRETE PARAMETERS N=1
  1  M=ACI W=150/1728

MESH INPUT
  NODES N=97
    1 X=0      Y=0      !  3 X=0      Y=84   G=1,3
    4 X=0      Y=105.15
    5 X=0      Y=114    !  7 X=0      Y=198  G=5,7
    8 X=0      Y=219.15
    9 X=0      Y=228    ! 11 X=0      Y=312  G=9,11
   12 X=0      Y=333.15
   13 X=756    Y=0      ! 15 X=756   Y=84   G=13,15
   16 X=756    Y=105.15
   17 X=756    Y=114
   18 X=756    Y=132
   19 X=756    Y=153.15
   20 X=756    Y=162
   21 X=756    Y=198
   22 X=756    Y=219.15
   23 X=756    Y=228
   24 X=756    Y=246
   25 X=756    Y=267.15
   26 X=756    Y=276
   27 X=756    Y=312
   28 X=756    Y=333.15
   29 X=756    Y=342
   30 X=756    Y=360
   31 X=756    Y=381.15
   32 X=1512   Y=48     ! 34 X=1512  Y=132  G=32,34
   35 X=1512   Y=153.15
   36 X=1512   Y=162    ! 38 X=1512  Y=246  G=36,38
   39 X=1512   Y=267.15
   40 X=1512   Y=276    ! 42 X=1512  Y=360  G=40,42
   43 X=1512   Y=381.15
   44 X=-10    Y=105.15

```

```

45 X=10      Y=105.15 ! 51 X=746   Y=105.15 G=45,51
52 X=766     Y=105.15
53 X=-10     Y=219.15
54 X=10      Y=219.15 ! 60 X=746   Y=219.15 G=54,60
61 X=766     Y=219.15
62 X=-10     Y=333.15
63 X=10      Y=333.15 ! 69 X=746   Y=333.15 G=63,69
70 X=766     Y=333.15
71 X=746     Y=153.15
72 X=766     Y=153.15 ! 78 X=1502  Y=153.15 G=72,78
79 X=1522    Y=153.15
80 X=746     Y=267.15
81 X=766     Y=267.15 ! 87 X=1502  Y=267.15 G=81,87
88 X=1522    Y=267.15
89 X=746     Y=381.15
90 X=766     Y=381.15 ! 96 X=1502  Y=381.15 G=90,96
97 X=1522    Y=381.15

```

## CONCRETE PROPERTIES N=1

```
1 Fpc=4000 Cr=2.5 Sh=0.0005 W=150/1728
```

## MILD STEEL PROPERTIES N=1

```
1 Es=29000000 P=0.05
```

## SECTION PROPERTIES N=4

```
1 D=5.5,24.5 B=104,16           ;SLAB PROPERTY
2 D=20 B=20                     ;COLUMN PROPERTY
```

## OFFSET DATA N=3

```
1 OI=0,0      OJ=0,-21.15
2 OI=0,8.85  OJ=0,0
3 OI=0,8.85  OJ=0,-21.15
```

## ELEMENTS N=79

## FRAME N=79

```

1,1,2      C=1 X=2 St=1 Day=1
2,2,3      C=1 X=2 St=1 Day=1 Off=1
3,5,6      C=1 X=2 St=1 Day=1 Off=2
4,6,7      C=1 X=2 St=1 Day=1 Off=1
5,9,10     C=1 X=2 St=1 Day=1 Off=2
6,10,11    C=1 X=2 St=1 Day=1 Off=1
7,13,14    C=1 X=2 St=1 Day=1
8,14,15    C=1 X=2 St=1 Day=1 Off=1
9,17,18    C=1 X=2 St=1 Day=1 Off=3
10,20,21   C=1 X=2 St=1 Day=1 Off=3
11,23,24   C=1 X=2 St=1 Day=1 Off=3
12,26,27   C=1 X=2 St=1 Day=1 Off=3
13,29,30   C=1 X=2 St=1 Day=1 Off=3
14,32,33   C=1 X=2 St=1 Day=1
15,33,34   C=1 X=2 St=1 Day=1 Off=1
16,36,37   C=1 X=2 St=1 Day=1 Off=2
17,37,38   C=1 X=2 St=1 Day=1 Off=1
18,40,41   C=1 X=2 St=1 Day=1 Off=2
19,41,42   C=1 X=2 St=1 Day=1 Off=1
20,44,4    C=1 X=1 St=1 Day=1
21,4,45    C=1 X=1 St=1 Day=1
22,45,46   C=1 X=1 St=1 Day=1 G=22,27,1,1,1
28,51,16   C=1 X=1 St=1 Day=1

```

```

29,16,52 C=1 X=1 St=1 Day=1
30,53,8 C=1 X=1 St=1 Day=1
31,8,54 C=1 X=1 St=1 Day=1
32,54,55 C=1 X=1 St=1 Day=1 G=32,37,1,1,1
38,60,22 C=1 X=1 St=1 Day=1
39,22,61 C=1 X=1 St=1 Day=1
40,62,12 C=1 X=1 St=1 Day=1
41,12,63 C=1 X=1 St=1 Day=1
42,63,64 C=1 X=1 St=1 Day=1 G=42,47,1,1,1
48,69,28 C=1 X=1 St=1 Day=1
49,28,70 C=1 X=1 St=1 Day=1
50,71,19 C=1 X=1 St=1 Day=1
51,19,72 C=1 X=1 St=1 Day=1
52,72,73 C=1 X=1 St=1 Day=1 G=52,57,1,1,1
58,78,35 C=1 X=1 St=1 Day=1
59,35,79 C=1 X=1 St=1 Day=1
60,80,25 C=1 X=1 St=1 Day=1
61,25,81 C=1 X=1 St=1 Day=1
62,81,82 C=1 X=1 St=1 Day=1 G=62,67,1,1,1
68,87,39 C=1 X=1 St=1 Day=1
69,39,88 C=1 X=1 St=1 Day=1
70,89,31 C=1 X=1 St=1 Day=1
71,31,90 C=1 X=1 St=1 Day=1
72,90,91 C=1 X=1 St=1 Day=1 G=72,77,1,1,1
78,96,43 C=1 X=1 St=1 Day=1
79,43,97 C=1 X=1 St=1 Day=1

```

## PRESTRESSING STEEL N=1

```
1 Ep=28000000 Meu=0.07 K=0.001/12 Fpu=270000 Fpy=255000 R=45
```

## TENDON GEOMETRY N=6

```

1 Spans=1 M=1 Area=1.377
  1 N=9 B=0,105.15 E=756,105.15
    LIST=4,45,46,47,48,49,50,51,16
    R=0.05,0.5,0.05 S=0,-18.15,0
2 Spans=1 M=1 Area=1.377
  1 N=9 B=0,219.15 E=756,219.15
    LIST=8,54,55,56,57,58,59,60,22
    R=0.05,0.5,0.05 S=0,-18.15,0
3 Spans=1 M=1 Area=1.377
  1 N=9 B=0,333.15 E=756,333.15
    LIST=12,63,64,65,66,67,68,69,28
    R=0.05,0.5,0.05 S=0,-18.15,0
4 Spans=1 M=1 Area=1.377
  1 N=9 B=756,153.15 E=1512,153.15
    LIST=19,72,73,74,75,76,77,78,35
    R=0.05,0.5,0.05 S=0,-18.15,0
5 Spans=1 M=1 Area=1.377
  1 N=9 B=756,267.15 E=1512,267.15
    LIST=25,81,82,83,84,85,86,87,39
    R=0.05,0.5,0.05 S=0,-18.15,0
6 Spans=1 M=1 Area=1.377
  1 N=9 B=756,381.15 E=1512,381.15
    LIST=31,90,91,92,93,94,95,96,43
    R=0.05,0.5,0.05 S=0,-18.15,0

```



```
MESH COMPLETE

STRUCTURE DEFINITION
  GRIDS 3 LEVELS 6
    GRID AA
      COLUMN 1 E=1,2,1
      COLUMN 2 E=3,4,1
      COLUMN 3 E=5,60,1
    GRID BB
      COLUMN 1 E=7,8,1
      COLUMN 2 LIST=9
      COLUMN 3 LIST=10
      COLUMN 4 LIST=11
      COLUMN 5 LIST=12
      COLUMN 6 LIST=13
    GRID CC
      COLUMN 1 E=14,15,1
      COLUMN 2 E=16,17,1
      COLUMN 3 E=18,19,1
  LEVEL LEV1
    SPAN 1 E=22,27,1
  LEVEL LEV2
    SPAN 1 E=52,57,1
  LEVEL LEV3
    SPAN 1 E=32,37,1
  LEVEL LEV4
    SPAN 1 E=62,67,1
  LEVEL LEV5
    SPAN 1 E=42,47,1
  LEVEL LEV6
    SPAN 1 E=72,77,1
STRUCTURE COMPLETE

CHANGE STRUCTURE
  BUILD N=1,2,1   Day=10
  BUILD N=7,8,1   Day=10
  BUILD N=14,15,1 Day=10
  RESTRAINTS
    1 R=1,1,1
    13 R=1,1,1
    32 R=1,1,1

CHANGE COMPLETE

SOLVE Day=10 ! OUTPUT

CHANGE STRUCTURE
  BUILD N=20,29,1 Day=17
  RESTRAINTS
    4 R=3,3,3   M=3
    16 R=3,3,3  M=15
    35 R=3,3,3  M=34

STRESS N=1 StressTo=216000,216000 Anchor=0.25,0.25
```

```

CHANGE COMPLETE

; apply selfweight - difference between tributary and effective
  flange width
LOADING
  L=20,29,1 F=0,-100*5.5*150/1728

SOLVE Day=17
OUTPUT

CHANGE STRUCTURE
  BUILD N=3,4,1      Day=24
  BUILD N=9          Day=24
  RESTRAINTS
    5 R=3,3,3      M=3
    17 R=3,3,3     M=15

CHANGE COMPLETE

SOLVE DAY=24 ! OUTPUT

CHANGE STRUCTURE
  BUILD N=50,59,1   Day=31
  RESTRAINTS
    17 R=3,3,0     M=15
    19 R=3,3,0     M=18
    35 R=3,3,3     M=34

  STRESS N=4 StressTo=216000,216000 Anchor=0.25,0.25

CHANGE COMPLETE

; apply selfweight - difference between tributary and effective
  flange width
LOADING
  L=50,59,1 F=0,-100*5.5*150/1728

SOLVE Day=31 ! OUTPUT

CHANGE STRUCTURE
  BUILD N=10        Day=38
  BUILD N=16,17,1   Day=38
  RESTRAINTS
    20 R=3,3,3     M=18
    36 R=3,3,3     M=34

CHANGE COMPLETE

SOLVE DAY=38 ! OUTPUT

CHANGE STRUCTURE
  BUILD N=30,39,1   Day=45
  RESTRAINTS
    8 R=3,3,3      M=7
    20 R=3,3,0     M=18
    22 R=3,3,0     M=21

```

```
STRESS N=2 StressTo=216000,216000 Anchor=0.25,0.25

CHANGE COMPLETE

; apply selfweight - difference between tributary and effective
  flange width
LOADING
  L=30,39,1 F=0,-100*5.5*150/1728

SOLVE Day=45 ! OUTPUT

CHANGE STRUCTURE
  BUILD N=5,6,1      Day=52
  BUILD N=11         Day=52
  RESTRAINTS
    9 R=3,3,3      M=7
    23 R=3,3,3     M=21

CHANGE COMPLETE

SOLVE DAY=52 ! OUTPUT

CHANGE STRUCTURE
  BUILD N=60,69,1   Day=59
  RESTRAINTS
    23 R=3,3,0     M=21
    25 R=3,3,0     M=24
    39 R=3,3,3     M=38

STRESS N=5 StressTo=216000,216000 Anchor=0.25,0.25

CHANGE COMPLETE

; apply selfweight - difference between tributary and effective
  flange width
LOADING
  L=60,69,1 F=0,-100*5.5*150/1728

SOLVE Day=59 ! OUTPUT

CHANGE STRUCTURE
  BUILD N=12         Day=66
  BUILD N=18,19,1   Day=66
  RESTRAINTS
    26 R=3,3,3     M=24
    40 R=3,3,3     M=38

CHANGE COMPLETE

SOLVE DAY=66 ! OUTPUT

CHANGE STRUCTURE
  BUILD N=40,49,1   Day=73
  BUILD N=13        Day=73
  RESTRAINTS
    12 R=3,3,0     M=11
```

```

26 R=3,3,0 M=24
28 R=3,3,0 M=27
29 R=3,3,3 M=27

```

```
STRESS N=3 StressTo=216000,216000 Anchor=0.25,0.25
```

```
CHANGE COMPLETE
```

```
; apply selfweight - difference between tributary and effective
flange width
```

```
LOADING
```

```
L=40,49,1 F=0,-100*5.5*150/1728
```

```
SOLVE Day=73 ! OUTPUT
```

```
CHANGE STRUCTURE
```

```
BUILD N=70,79,1 Day=80
```

```
RESTRAINTS
```

```
29 R=3,3,0 M=27
```

```
31 R=3,3,0 M=30
```

```
43 R=3,3,0 M=42
```

```
STRESS N=6 StressTo=216000,216000 Anchor=0.25,0.25
```

```
CHANGE COMPLETE
```

```
; apply selfweight - difference between tributary and effective
flange width
```

```
LOADING
```

```
L=70,79,1 F=0,-100*5.5*150/1728
```

```
SOLVE Day=80
```

```
OUTPUT
```

```
SOLVE Day=100
```

```
OUTPUT
```

```
SOLVE Day=180
```

```
OUTPUT
```

```
SOLVE Day=365
```

```
OUTPUT
```

```
SOLVE Day=365*20
```

```
OUTPUT
```

```
CAPTURE STATUS
```

```
LOADING DL
```

```
; extract dead loading
```

```
DEAD LOAD
```

```
SOLVE ! OUPUT
```

```
LOADING PT
```

```
; extract prestressing component
```

```
PRESTRESSING
```

```
SOLVE ! OUTPUT
```

```
LOADING LT ; extract long-term effects
  LONG TERM
SOLVE ! OUTPUT
```

```
LOADING LIVE1
  STRUCTURE
    LEVEL LEV1
      SPAN 1
      UNIFORM F=0,-88.33
    LEVEL LEV2
      SPAN 1
      UNIFORM F=0,-88.33
    LEVEL LEV3
      SPAN 1
      UNIFORM F=0,-88.33
    LEVEL LEV4
      SPAN 1
      UNIFORM F=0,-88.33
    LEVEL LEV5
      SPAN 1
      UNIFORM F=0,-88.33
    LEVEL LEV6
      SPAN 1
      UNIFORM F=0,-88.33
```

```
SOLVE ! OUTPUT
```

```
LOADING LIVE2
  PATTERN
    LEVEL LEV1
      SPAN 1
      UNIFORM F=0,-66.25
    LEVEL LEV2
      SPAN 1
      UNIFORM F=0,-66.25
    LEVEL LEV3
      SPAN 1
      UNIFORM F=0,-66.25
    LEVEL LEV4
      SPAN 1
      UNIFORM F=0,-66.25
    LEVEL LEV5
      SPAN 1
      UNIFORM F=0,-66.25
    LEVEL LEV6
      SPAN 1
      UNIFORM F=0,-66.25
  SKIP
    LEVEL LEV1 SPANS 1
    LEVEL LEV2 SPANS 0
    LEVEL LEV3 SPANS 1
    LEVEL LEV4 SPANS 0
    LEVEL LEV5 SPANS 1
    LEVEL LEV6 SPANS 0
  SKIP
```

```
LEVEL LEV1 SPANS 0
LEVEL LEV2 SPANS 1
LEVEL LEV3 SPANS 0
LEVEL LEV4 SPANS 1
LEVEL LEV5 SPANS 0
LEVEL LEV6 SPANS 1
```

```
SOLVE ! OUPUT
```

```
;;..... combine solution
```

```
COMBINATIONS
```

```
COMBINATION DLPTLT
```

```
1*DL+1*PT+1*LT
```

```
STOP
```