K A R D E S T U N C E R - O V E R V I E W

What is Kardestuncer?

Kardestuncer is structural analysis software programmed for the HP 48 G+/GX. Like many famous PC programs, Kardestuncer has powerful features. They include:

- Static analysis of plane structures including grid structures (it can solve any plane structure with any load condition).
- Load cases (it supports infinite load cases).
- Complete and accurate results (it reports all data analysis).
- Practical data input (it has a simple way for codifying the structure).
- Customizable (it has adjustable options).
- Solution by matrix method (it uses the matrix stiffness method)
- Small code size (it needs almost 6240 bytes).

Using Kardestuncer

Although Kardestuncer can solve any plane structure, here will be demonstrated only the procedure for solving a frame structures. The procedure for solving other structures is identical. The procedure that will be showed is intrinsic to the structural analysis and its philosophy had been adopted in many famous PC programs during many years.

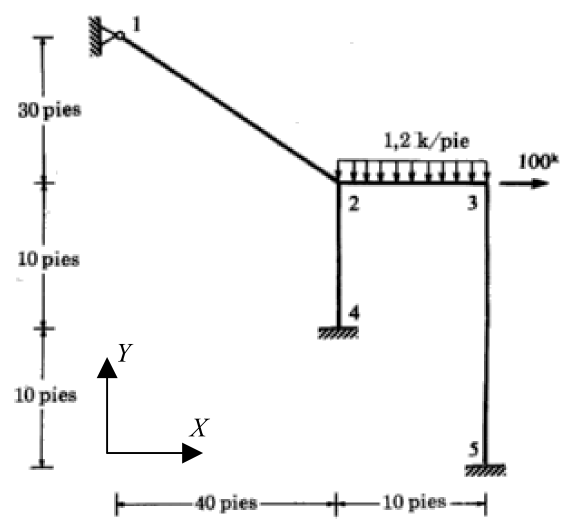
Example

Source

KARDESTUNCER HAYRETTIN (1974). "Introducción al Análisis Estructural con Matrices". Ed. McGraw-Hill. México. pp. 267-268, 410.

Statement 6.6

Determine the internal forces of the elements of the simple frame structure shown at the right.



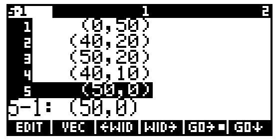
E	1000	k/pie^2
I	100	pie^4
A	10	pie^3

Solution

The first thing that we have to do is create a directory for this problem. Any frame structure must be saved inside of { HOME KAR MAR }. If this new directory is named FRA1 we must prepare the root as shown in the next screenshot:

Now will be entered the structure geometry which consist basically of nodes and elements.

For enter the nodes, we have to suppose an origin in any convenient place like the bottom left position shown in the above figure. Then, the coordinates of the nodes must be ordered and entered in a matrix form as shown in the following screenshot:

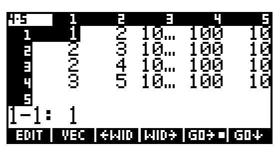


Row 1 has the coordinates of the node 1; row 2 has the coordinates of the node 2 and so forth. Note that the VEC option is off. Subsequently, this information must be saved in a variable named NU, which can be accessed pressing the CST key. To here we have the following screen:

Before to introduce the frame elements we should have written a preliminary table like this:

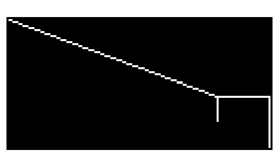
Element	Initial Node i	Final Node j	E	I	A
1	1	2	1000	100	10
2	2	3	1000	100	10
3	2	4	1000	100	10
4	3	5	1000	100	10

Note that i is always less than j. And the units must be maintained compatible no matter which system of units is being used. After that, copy the last table in a matrix form as shown in the following screenshot:



Notice that the matrix is created starting from the second column of the preliminary table. This matrix should be saved in a variable named ELE, which can be accessed pressing CST key. To here we have the next screen:

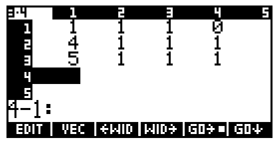
Once the nodes and elements have been entered, it is good idea to verify the structure geometry pressing CST, NXT and GRA in order to obtain:



Prior to enter the boundary conditions, each structure support should be codified according to the next reference table:

	Displacement *(X)*	Displacement *(Y)*	Rotation *(Z)*
Fixed	1	1	1
Released	0	0	0

Then, the structure supports of this frame must be entered as shown in the next screenshot:



In this matrix, note that the first column has the supported node number. Finally, the matrix must be saved in a variable named BOR. To here we have the following screen:

Now, we are ready to enter the applied loads. First, we must create a directory for the unique load case that we have in this frame structure. Let name this directory as C1 as shown in the screenshot:

To begin with the applied loads, we can write a preliminary table related to the joint loads as follow:

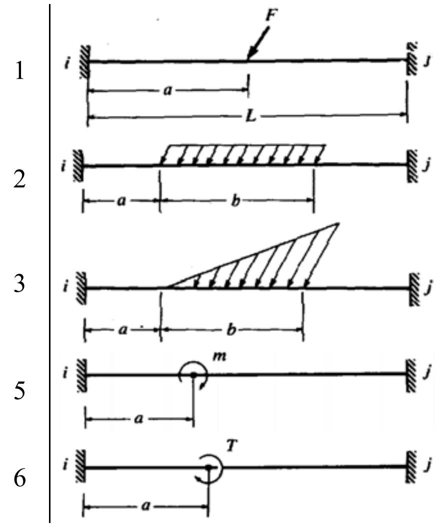
Load Case 1 – Joint Loads

Node	Load *(X)*	Load *(Y)*	Moment Load *(Z)*
3	100	0	0

Notice that the join load in *Z* axis direction could be filled with a concentrated moment when it exists acting on the node. Then, we must copy the above table in a matrix form as shown in the screenshot:

Remember that the VEC option is off. Finally, this matrix must be saved in a variable named ↓N accessible pressing the CST key. To here we should have:

The second step is related to the element loads. Before to enter the element loads, we have to check the type of load verifying the following reference table:



According to the last reference, we have a distributed element load codified as 2. With this information in hand, we should prepare a preliminary table like this:

Load Case 1 – Element Loads

Element	Load Code	Start *(a)*	Application Distance *(b)*	Load *(X)*	Load *(Y)*	Moment Load *(Z)*
2	2	0	10	0	-1.2	0

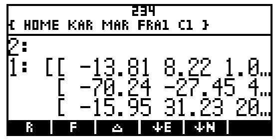
Next, we must copy the above table in a matrix form as show in the following screenshot:

Remember that the VEC option is off. Then, this matrix must be saved in a variable named ↓E accessible pressing the CST and NXT keys. To here we have:

As soon as the information about the structure geometry and applied loads are completed, we are ready to run the structural analysis that will give us important results.

In order to execute the analysis, we have to press CST, NXT and CAL. After that, the calculator will delay some seconds while performing the analysis. Once finished this step, the VAR menu should show:

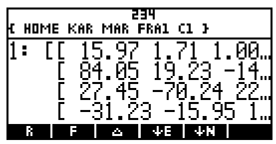
The results appear saved in R, F and Δ. The reactions acting on the supports are saved in R, the internal element forces are saved in F and the node displacements are represented by Δ. When R is put on the stack we have:



Each row represents each supported node entered during the boundary conditions step. Then, this reaction matrix could be interpreted as follow:

$$R_1 = \begin{bmatrix} -13.81 \\ 8.22 \\ 0 \end{bmatrix} \qquad R_4 = \begin{bmatrix} -70.24 \\ -27.45 \\ 474.98 \end{bmatrix} \qquad R_5 = \begin{bmatrix} -15.95 \\ 31.23 \\ 208.60 \end{bmatrix}$$

When F is put on the stack, we can see:



Each row represents the internal forces acting on the corresponding element. If we expand this matrix then we have:

$$\begin{bmatrix} 15.97 & 1.71 & 0 & 15.97 & 1.71 & -85.57 \\ 84.05 & 19.23 & -141.84 & 84.05 & 31.23 & -110.49 \\ 27.45 & -70.24 & 227.42 & 27.45 & -70.24 & 474.98 \\ -31.23 & -15.95 & 110.49 & -31.23 & -15.95 & 208.60 \end{bmatrix}$$

Columns 1 to 3 represent the *ij* force element and columns 4 to 6 represent the *ji* force element. Now we can interpret these results as follow:

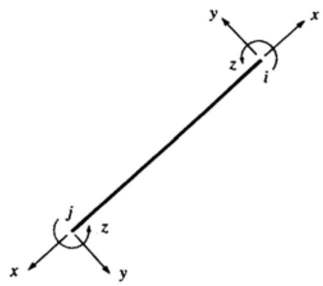
$$F_{12} = \begin{bmatrix} 15.97 \\ 1.71 \\ 0 \end{bmatrix} \qquad F_{21} = \begin{bmatrix} 15.97 \\ 1.71 \\ -85.57 \end{bmatrix}$$

$$F_{23} = \begin{bmatrix} 84.05 \\ 19.23 \\ -141.84 \end{bmatrix} \qquad F_{32} = \begin{bmatrix} 84.05 \\ 31.23 \\ -110.49 \end{bmatrix}$$

$$F_{24} = \begin{bmatrix} 27.45 \\ -70.24 \\ 227.42 \end{bmatrix} \qquad F_{42} = \begin{bmatrix} 27.45 \\ -70.24 \\ 474.98 \end{bmatrix}$$

$$F_{35} = \begin{bmatrix} -31.23 \\ -15.95 \\ 110.49 \end{bmatrix} \qquad F_{53} = \begin{bmatrix} -31.23 \\ -15.95 \\ 208.60 \end{bmatrix}$$

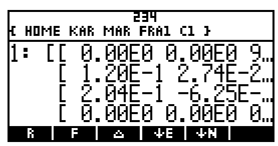
Remember that each vector is associated with the local coordinate system of element. The local coordinate system of any element is represented by the following figure:



According to the "scanned" answer of the book, we have:

$$\mathbf{p}_{32} = \begin{bmatrix} 84{,}0^{k} \\ 31{,}23^{k} \\ -\,110{,}5 \text{ k-pies} \end{bmatrix} \qquad \mathbf{p}_{35} = \begin{bmatrix} -\,31{,}23^{k} \\ -\,15{,}95^{k} \\ 110{,}5 \text{ k-pies} \end{bmatrix}$$

Finally, we will see the displacement results. If we put Δ on the stack then we have:



This displacement matrix could be expanded as follow:

$$\Delta_1 = \begin{bmatrix} 0\times 10^{0} \\ 0\times 10^{0} \\ 9.02\times 10^{-3} \end{bmatrix} \quad \Delta_2 = \begin{bmatrix} 1.20\times 10^{-1} \\ 2.74\times 10^{-2} \\ -1.24\times 10^{-2} \end{bmatrix} \quad \Delta_3 = \begin{bmatrix} 2.04\times 10^{-1} \\ -6.25\times 10^{-2} \\ -9.81\times 10^{-3} \end{bmatrix} \quad \Delta_4 = \begin{bmatrix} 0.00\times 10^{0} \\ 0.00\times 10^{0} \\ 0.00\times 10^{0} \end{bmatrix} \quad \Delta_5 = \begin{bmatrix} 0.00\times 10^{0} \\ 0.00\times 10^{0} \\ 0.00\times 10^{0} \end{bmatrix}$$

The displacements for two components of node 1 and all components of nodes 4 and 5 are as expected because they do not have displacement in the fixed direction.

Ing. Roger Saravia A.
http://www.geocities.com/gstvsrv/kar
gstvsrv@lycos.com
La Paz – Bolivia

Oct/03