**Example 13.1 : Page 400**

Reinforced Concrete - Design Theory and Examples
Second Edition
T.J. Macginley & B.S. Choo

appVersion(−4) = "0.99.7822.147"

appVersion(4) = "0.99.7822.147"

$t_0 := \text{time}(0)$

https://www.academia.edu/34894173/Reinforced_Concrete_Design_Theory_and_Examples

## Example 13.1 Simplified analysis of concrete framed building—vertical load

The application of the various simplified methods of analysis given in section 3.2 of the code is shown in the following example.

### (a) Specification

The cross-section of a reinforced concrete building is shown in Fig. 13.5(a). The frames are at 4.5 m centres, the length of the building is 36 m and the column bases are fixed. Preliminary sections for the beams and columns are shown in Fig. 13.5(b). The floor and roof slabs are **designed to span one way** between the frames. Longitudinal beams are provided between external columns of the roof and floor levels only.:

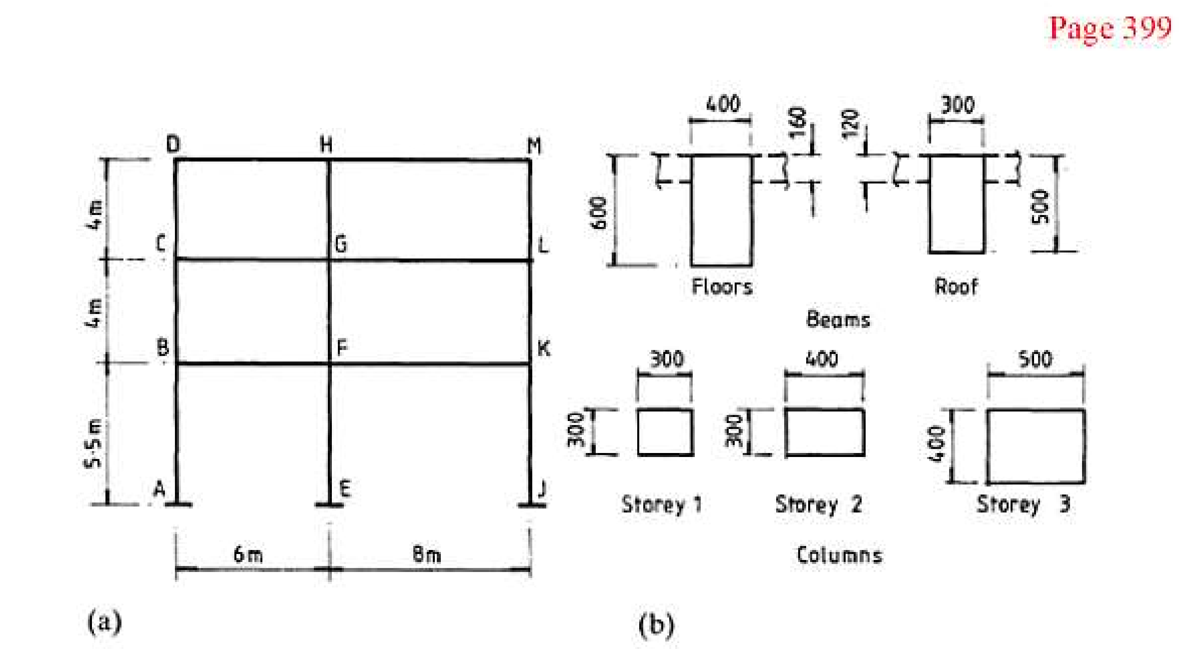



Fig. 13.5 (a) Cross-section; (b) assumed member sections.

Frames are spaced at 4.5 m

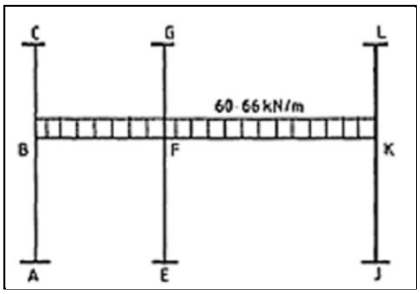
$S_{frame} := 4.5 \text{ m}$

$L_{building} := 36 \text{ m}$

$\gamma_{cu} := 24 \frac{\text{kN}}{\text{m}^3}$

**This worked example shows only the analysis of the sub-frame shown. Hence, only Dead & Imposed loads on Beams BF & FK are considered.**

**Iterations -----> While LOOP based on a given threshold value, to be defined at MAIN PROGRAM level**




For column **GF**

$L_{BF} := 6 \text{ m}$

$L_{FK} := 8 \text{ m}$

Define Number of Member Moments to evaluate

$N_{members} := 16$

$col_{FE\_b} := 400 \text{ mm}$ $col_{FE\_h} := 500 \text{ mm}$

$col_{GF\_b} := 300 \text{ mm}$ $col_{GF\_h} := 400 \text{ mm}$

$beam_{FK\_b} := 400 \text{ mm}$ $beam_{FK\_h} := 600 \text{ mm}$

## Loadings

**For the floors**

Floors D/L

$$g_{k\_floor} := 6.2\ \frac{kN}{m^2}$$

Floors I/L

$$q_{k\_floor} := 3.0\ \frac{kN}{m^2}$$

**Following load cases are required for the beam BFK for the braced frame.**
**But, only Case 1 is shown in this example.**

Case 1 $1.4 \cdot G_k + 1.6 \cdot Q_k$ on the whole beam

Case 2 $1.4 \cdot G_k + 1.6 \cdot Q_k$ on BF and $1.6 \cdot G_k$ on FK

Case 3 $1.0 \cdot G_k$ on BF and $1.4 \cdot G_k + 1.6 \cdot Q_k$ on FK

## Loads on Beams

Dead loads on Beams

$$G_{k\_floor} := g_{k\_floor} \cdot S_{frame} = 27.9\ \frac{kN}{m}$$

Imposed loads on Beams

$$Q_{k\_floor} := q_{k\_floor} \cdot S_{frame} = 13.5\ \frac{kN}{m}$$

## Calculation of Moments of Inertia

$$I_{Col\_FE} := \frac{1}{12} \cdot col_{FE\_b} \cdot \left(col_{FE\_h}\right)^3 = 4.1667 \cdot 10^9\ mm^4$$

$$I_{Col\_FG} := \frac{1}{12} \cdot col_{GF\_b} \cdot \left(col_{GF\_h}\right)^3 = 1.6 \cdot 10^9\ mm^4$$

$$I_{beam\_FK} := \frac{1}{12} \cdot beam_{FK\_b} \cdot \left(beam_{FK\_h}\right)^3 = 7.2 \cdot 10^9\ mm^4$$

$$I_{beam\_FB} := I_{beam\_FK} = 7.2 \cdot 10^9\ mm^4$$

**Beam BF and FK : same cross section**

Moment of Inertia

$$I := \begin{bmatrix} I_{Col\_FE} \\ I_{Col\_FG} \\ I_{beam\_FK} \\ I_{beam\_FB} \end{bmatrix} = \begin{bmatrix} 4.1667 \cdot 10^9 \\ 1.6 \cdot 10^9 \\ 7.2 \cdot 10^9 \\ 7.2 \cdot 10^9 \end{bmatrix} mm^4$$

Length
Cols & Beams

$$L := \begin{bmatrix} 5500 \\ 4000 \\ 8000 \\ 6000 \end{bmatrix} mm$$

Stiffness

$$\begin{bmatrix} K_{FE} \\ K_{FG} \\ K_{FK} \\ K_{FB} \end{bmatrix} := \overrightarrow{\frac{I}{L}} = \begin{bmatrix} 7.5758 \\ 4 \\ 9 \\ 12 \end{bmatrix} 10^5\ mm^3$$

### (d) : Subframe analysis for braced frames

The subframe consists of the beams at first floor level and the columns above and below that level with ends fixed.
The frame is analysed for the dead and imposed load cases given..
Member Stiffness of the subframe at **B, F, K are as follows**

**Stiffnesses** (**Arrangement of K is very important**)

**Joint B**

$$K_B := \begin{bmatrix} K_{FE} \\ K_{FB} \\ K_{FG} \end{bmatrix}$$

**Joint F**

$$K_F := \begin{bmatrix} K_{FE} \\ K_{FB} \\ K_{FG} \\ K_{FK} \end{bmatrix}$$

**Joint K**

$$K_K := \begin{bmatrix} K_{FE} \\ K_{FK} \\ K_{FG} \end{bmatrix}$$

→

**Group A, B, C, D**

$$K := \begin{bmatrix} K_B \\ K_F \\ K_K \end{bmatrix}$$

## Fixed End Moments

**Case I**

**Case II** (Not considered in this example)

**The FEMs are for case1 (F1) fon both beams**

$F1 := 1.4 \cdot G_{k\_floor} + 1.6 \cdot Q_{k\_floor} = 60.66 \ \frac{kN}{m}$

$F2 := 1.0 \cdot G_{k\_floor} = 27.9 \ \frac{kN}{m}$

FEM for span BF: Case I

$FEM_{BF} := - F1 \cdot \frac{L_{BF}^{2}}{12} = - 181.98 \ kN\ m$

FEM for span FK: Case I

$FEM_{FK} := - F1 \cdot \frac{L_{FK}^{2}}{12} = - 323.52 \ kN\ m$

FEM for FB: Case I

$FEM_{FB} := - FEM_{BF} = 181.98 \ kN\ m$

FEM for span KF: Case I

$FEM_{KF} := - FEM_{FK} = 323.52 \ kN\ m$

## These programs (Yellow Color) are applicable to any Method

PROGRAM 1 : Find Nested DF for ALL JOINTS.
Using vectorize function

```
Find_DF(k) := | df := vectorize( k / Σ k )
```

PROGRAM 2 : To replace zeros with blank spaces.
(To improve readability in TABLE).
Lots of zeros will clutter the appearance.

```
format(M#) := | j := [1..rows(M#)]
              | k := [1..cols(M#)]
              | BB[j k] := { ""       if  M#[j k] = 0
              |            { M#[j k]  otherwise
              | BB
```

PROGRAM 3 : To find column totals.
(Last row in Mom. Dist Table)

```
tot_BMM(M#) := | j := [1..cols(M#)]
               | A#[j] := Σ col(M#, j)
```

PROGRAM 4 : To stack a nested matrix

```
Stack_M(M) := | A := M[1]
              | for j ∈ [2..rows(M)]
              |   | A := stack(A, M[j])
              | eval(A)
```

PROGRAM 5: To Group LOAD, DF, FEM, BMM & BMM from MAIN PROGRAM
PROGRAM 4 stacks nested 'd'. PROGRAM 3 calc all Col Totals (A3).
A1 removes the last CO row of output from MAIN PROGRAM.

```
Group(ld, d, fem#, df) := | d := Stack_M(d) / (kN m)
                          | A1 := d[[1..(rows(d) - 1)][1..cols(d)]]
                          | A2 := stack(fem# / (kN m), A1)
                          | A3 := tot_BMM(A2)^T
                          | eval(stack(ld, df^T, A2, A3))
```

Define variables for use in PROGRAM 6

```
P1 := [ "Bal"
        "CO" ]
P2 := [ "DF"
        "FEM" ]
P3 := [ "Final BM" ]
```

PROGRAM 6 : Create Left Stub for Table

```
Create_LS(n) := | B := P1
                | for j ∈ [1..(n - 1)]
                |   | B := stack(B, P1)
                | B := B[[1..(rows(B) - 1)][1..cols(B)]]
                | stack("Load", P2, B, P3)
```

# Following PROGRAMS apply ONLY to the type of Example used

Header for Results Table

$Hd := [\text{"AB" "BA" "BF" "BC" "CB" "EF" "FE" "FB" "FG" "FK" "GF" "JK" "KJ" "KF" "KL" "LK"}]$

$\text{cols}(Hd) = 16$ Should be equal to ➡ $N_{members} = 16$

## Dist. Factors

PROGRAM 7:
Call PROGRAM 1
to find DF Array (Nested)
K = Stiffness

$$DF := Find\_DF(K) = \begin{bmatrix} \begin{bmatrix} 0.32 \\ 0.51 \\ 0.17 \end{bmatrix} \\ \begin{bmatrix} 0.23 \\ 0.37 \\ 0.12 \\ 0.28 \end{bmatrix} \\ \begin{bmatrix} 0.37 \\ 0.44 \\ 0.19 \end{bmatrix} \end{bmatrix}$$

PROGRAM 8
DFV vector; ONLY to display in the
Final Result & Table

$DFV := \text{stack}\left(0, DF_1, 0, 0, DF_2, 0, 0, DF_3, 0\right)$ $\text{cols}\left(DFV^T\right) = 16$

$DFV^T = [0\ 0.32\ 0.51\ 0.17\ 0\ 0\ 0.23\ 0.37\ 0.12\ 0.28\ 0\ 0\ 0.37\ 0.44\ 0.19\ 0]$

PROGRAM 9 : Balancing Moments

$$Bal\_M(M\#, n\#) := \left| \begin{array}{l} Z_{1\,n\#} := 0 \\ \left[Z_2\ Z_3\ Z_4\right] := -M\#_1 \cdot {DF_1}^T \\ \left[Z_7\ Z_8\ Z_9\ Z_{10}\right] := -\left(M\#_2 + M\#_3\right) \cdot {DF_2}^T \\ \left[Z_{13}\ Z_{14}\ Z_{15}\right] := -M\#_4 \cdot {DF_3}^T \\ \text{eval}(Z) \end{array} \right.$$

PROGRAM 10 : Carry Over Moments

$$CO\_M(M, BL) := \left| \begin{array}{l} M_1 := 0.5 \cdot BL_2 \\ M_3 := 0.5 \cdot BL_8 \\ M_5 := 0.5 \cdot BL_4 \\ M_6 := 0.5 \cdot BL_7 \\ M_8 := 0.5 \cdot BL_3 \\ M_{10} := 0.5 \cdot BL_{14} \\ M_{11} := 0.5 \cdot BL_9 \\ M_{12} := 0.5 \cdot BL_{13} \\ M_{14} := 0.5 \cdot BL_{10} \\ M_{16} := 0.5 \cdot BL_{15} \\ \text{eval}(M) \end{array} \right.$$

PROGRAM 11 :
Load arrangement

$$Find\_LD(M) := \left| \begin{array}{l} M_3 := F1 \\ M_8 := F1 \\ M_{10} := F1 \\ M_{14} := F1 \\ \text{eval}\left(\dfrac{M}{\frac{\text{kN}}{\text{m}}}\right) \end{array} \right.$$

```
MAIN PROGRAM : Moment Distribution.
mem = Number of members;  Δba = Threshold value
```

```
BMM(mem, Δba) :=
    | fem[1 mem := 0
    | co[1 mem := 0
    | BA[1 mem := 0
    | LD[1 mem := 0
    LDD := Find_LD(LD)
    fem[3 := FEM_BF
    fem[8 := − fem[3
    fem[10 := FEM_FK
    fem[14 := − fem[10
    ΔBal := Δba kN m
    "----------------"
    j := 1
    BA[j := Bal_M([fem[3; fem[8; fem[10; fem[14], mem)
    CO[j := CO_M(co, BA[j)
    D[j := stack(BA[j, CO[j)
    tmp := BA[j
    while Max(|→tmp|) ≥ ΔBal
        "------------------"
        j := j + 1
        BA[j := Bal_M([co[3; co[8; co[10; co[14], mem)
        CO[j := CO_M(co, BA[j)
        tmp := BA[j
        D[j := stack(BA[j, CO[j)
    A := Group(LDD, D, fem, DFV)
    B := Create_LS(j)
    eval([A; B; j])
```

## MAIN PROGRAM

Initialize input **arrays**

**Load** arrangement : **Program 11**

**FEM** Assign

Assign Unit to **Δba** **(Threshold Value)**

**j = 1** Iteration starts

**Bal Program 9**

**CO Program 10**

Stack **Bal** & **CO**

Assign **BA** array to **tmp.**

**Can't vectorize Max(BA$_j$)** directly.

**Vectorize abs (tmp)** to find **NUMERICALLY LARGEST value of each Bal array**

**j** : Iteration continue

**Bal Program 9**

**CO Program 10**

Assign **BA** array to **tmp. Can't use BA$_j$** directly.

Group **Loads, DF, FEM, Bal, CO, Totals** **Program 5**

**Left Stub** for TABLE **Program 6**

**j =** Number of Iterations

**Ex: Largest Numerical Value of an Array**

$XX := [-3\ 5\ -1\ -10]$

$\text{Max}(XX) = 5$

$|\overrightarrow{XX}| = [3\ 5\ 1\ 10]$

$\text{Max}(|\overrightarrow{XX}|) = 10$

| Already defined | Define Threshold value | Results from MAIN PROGRAM | |
|---|---|---|---|
| $N_{members} = 16$ | $\Delta Val := 0.1$ | $Ans := BMM(N_{members}, \Delta Val)$ | |
| **Moment Distrb.** | **Left Stub for Table** | **Iterations** | **Format "MD" for Table** |
| $MD := Ans_1$ | $LS := Ans_2$ | $Iter := Ans_3 = 6$ | $CC := format(MD)$ |

$$MD = \begin{bmatrix} 0 & 0 & 60.66 & 0 & 0 & 0 & 0 & 60.66 & 0 & 60.66 & 0 & 0 & 0 & 60.66 & 0 & 0 \\ 0 & 0.32 & 0.51 & 0.17 & 0 & 0 & 0.23 & 0.37 & 0.12 & 0.28 & 0 & 0 & 0.37 & 0.44 & 0.19 & 0 \\ 0 & 0 & -181.98 & 0 & 0 & 0 & 0 & 181.98 & 0 & -323.52 & 0 & 0 & 0 & 323.52 & 0 & 0 \\ 0 & 58.48 & 92.63 & 30.88 & 0 & 0 & 32.92 & 52.14 & 17.38 & 39.1 & 0 & 0 & -119.12 & -141.51 & -62.89 & 0 \\ 29.24 & 0 & 26.07 & 0 & 15.44 & 16.46 & 0 & 46.31 & 0 & -70.76 & 8.69 & -59.56 & 0 & 19.55 & 0 & -31.45 \\ 0 & -8.38 & -13.27 & -4.42 & 0 & 0 & 5.68 & 9 & 3 & 6.75 & 0 & 0 & -7.2 & -8.55 & -3.8 & 0 \\ -4.19 & 0 & 4.5 & 0 & -2.21 & 2.84 & 0 & -6.63 & 0 & -4.28 & 1.5 & -3.6 & 0 & 3.38 & 0 & -1.9 \\ 0 & -1.45 & -2.29 & -0.76 & 0 & 0 & 2.54 & 4.02 & 1.34 & 3.01 & 0 & 0 & -1.24 & -1.48 & -0.66 & 0 \\ -0.72 & 0 & 2.01 & 0 & -0.38 & 1.27 & 0 & -1.15 & 0 & -0.74 & 0.67 & -0.62 & 0 & 1.51 & 0 & -0.33 \\ 0 & -0.65 & -1.02 & -0.34 & 0 & 0 & 0.44 & 0.69 & 0.23 & 0.52 & 0 & 0 & -0.55 & -0.66 & -0.29 & 0 \\ -0.32 & 0 & 0.35 & 0 & -0.17 & 0.22 & 0 & -0.51 & 0 & -0.33 & 0.12 & -0.28 & 0 & 0.26 & 0 & -0.15 \\ 0 & -0.11 & -0.18 & -0.06 & 0 & 0 & 0.2 & 0.31 & 0.1 & 0.23 & 0 & 0 & -0.1 & -0.11 & -0.05 & 0 \\ -0.06 & 0 & 0.15 & 0 & -0.03 & 0.1 & 0 & -0.09 & 0 & -0.06 & 0.05 & -0.05 & 0 & 0.12 & 0 & -0.03 \\ 0 & -0.05 & -0.08 & -0.03 & 0 & 0 & 0.03 & 0.05 & 0.02 & 0.04 & 0 & 0 & -0.04 & -0.05 & -0.02 & 0 \\ 23.95 & 47.85 & -73.11 & 25.26 & 12.64 & 20.89 & 41.81 & 286.13 & 22.07 & -350.01 & 11.03 & -64.1 & -128.25 & 195.97 & -67.72 & -33.85 \end{bmatrix}$$

$time(0) - t_0 = 0.14\ s$

| A | B | | | C | E | F | | | | G | J | K | | | L |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AB | BA | BF | BC | CB | EF | FE | FB | FG | FK | GF | JK | KJ | KF | KL | LK |
| | 0·32 | 0·51 | 0·17 | | | 0·23 | 0·37 | 0·12 | 0·28 | | | 0·37 | 0·44 | 0·19 | |
| | 58·2 | -181·9<br>92·8 | 30·9 | | | 32·5 | 181·9<br>52·3 | 16·9 | -323·2<br>39·6 | | | -119·6 | 323·2<br>-142·2 | -61·6 | |
| 29·1 | -8·3 | 26·2<br>-13·6 | -4·5 | 15·5 | 16·3 | 5·7 | 46·4<br>9·1 | 3·0 | -71·1<br>6·9 | 8·5 | -59·8 | -7·3 | 19·8<br>-8·7 | -3·8 | -30·7 |
| -4·2 | -1·5 | 4·6<br>-2·3 | -0·8 | -2·3 | 2·9 | 2·6 | -6·7<br>4·1 | 1·3 | -4·4<br>3·1 | 1·5 | -3·7 | -1·3 | 3·5<br>-1·5 | -0·7 | -1·9 |
| 24·9 | 48·4 | -74·0 | 25·6 | 13·3 | 19·2 | 40·8 | 287·1 | 21·2 | -349·1 | 10·0 | -63·5 | -128·2 | 194·1 | -65·9 | -32·6 |

**Text Book Result : Page 400**

**Fig. 13.6**

**Reinforced Concrete - Design Theory and Examples**
**Second Edition - T.J. Macginley & B.S. Choo**
**Example 13.1 : Page 398**

| | AB | BA | BF | BC | CB | EF | FE | FB | FG | FK | GF | JK | KJ | KF | KL | LK |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Load** | | | 60.66 | | | | | 60.66 | | 60.66 | | | | 60.66 | | |
| **DF** | | 0.32 | 0.51 | 0.17 | | | 0.23 | 0.37 | 0.12 | 0.28 | | | 0.37 | 0.44 | 0.19 | |
| **FEM** | | | -181.98 | | | | | 181.98 | | -323.52 | | | | 323.52 | | |
| **Bal** | | 58.48 | 92.63 | 30.88 | | | 32.92 | 52.14 | 17.38 | 39.1 | | | -119.12 | -141.51 | -62.89 | |
| **CO** | 29.24 | | 26.07 | | 15.44 | 16.46 | | 46.31 | | -70.76 | 8.69 | -59.56 | | 19.55 | | -31.45 |
| **Bal** | | -8.38 | -13.27 | -4.42 | | | 5.68 | 9 | 3 | 6.75 | | | -7.2 | -8.55 | -3.8 | |
| **CO** | -4.19 | | 4.5 | | -2.21 | 2.84 | | -6.63 | | -4.28 | 1.5 | -3.6 | | 3.38 | | -1.9 |
| **Bal** | | -1.45 | -2.29 | -0.76 | | | 2.54 | 4.02 | 1.34 | 3.01 | | | -1.24 | -1.48 | -0.66 | |
| **CO** | -0.72 | | 2.01 | | -0.38 | 1.27 | | -1.15 | | -0.74 | 0.67 | -0.62 | | 1.51 | | -0.33 |
| **Bal** | | -0.65 | -1.02 | -0.34 | | | 0.44 | 0.69 | 0.23 | 0.52 | | | -0.55 | -0.66 | -0.29 | |
| **CO** | -0.32 | | 0.35 | | -0.17 | 0.22 | | -0.51 | | -0.33 | 0.12 | -0.28 | | 0.26 | | -0.15 |
| **Bal** | | -0.11 | -0.18 | -0.06 | | | 0.2 | 0.31 | 0.1 | 0.23 | | | -0.1 | -0.11 | -0.05 | |
| **CO** | -0.06 | | 0.15 | | -0.03 | 0.1 | | -0.09 | | -0.06 | 0.05 | -0.05 | | 0.12 | | -0.03 |
| **Bal** | | -0.05 | -0.08 | -0.03 | | | 0.03 | 0.05 | 0.02 | 0.04 | | | -0.04 | -0.05 | -0.02 | |
| **Final BM** | 23.95 | 47.85 | -73.11 | 25.26 | 12.64 | 20.89 | 41.81 | 286.13 | 22.07 | -350.01 | 11.03 | -64.1 | -128.25 | 195.97 | -67.72 | -33.85 |

CC

$Iter = 6$ $time(0) - t_0 = 0.16\ s$

## Checking Balacing Moments of While Loop Calculations

Balancing Rows in MD : **4, 6, 8, 10, 12, 14**. $\text{rows}(MD) - 1 = 14$ Last row of **MD** is **Final BM**

NUMERICALLY LARGEST VALUE of each BAL row of "MD"

```
Check_Max_Bal(M#) := | A := M#[4,6..(rows(M#)-1)][1..cols(M#)]
                     | for k ∈ [1..rows(A)]
                     |   | B := A_k[1..cols(A)]
                     |   | C_k := Max(|B→|)
                     | C
```

Extract **Bal** rows from **MD** matrix

**Largest** Value of each row of **Bal** matrix

$$M\_Bal := Check\_Max\_Bal(MD) = \begin{bmatrix} 141.51 \\ 13.27 \\ 4.02 \\ 1.02 \\ 0.31 \\ 0.08 \end{bmatrix}$$

$\le$ $\Delta Val = 0.1$ $Iter = 6$ $\text{time}(0) - t_0 = 0.16\ s$

---

## Check for a Different Threshold value

New Threshold value

$\Delta Val2 := 0.05$

Results from MAIN PROGRAM

$Ans2 := BMM(N_{members}, \Delta Val2)$

Moment Distrb.

$MD2 := Ans2_1$

Iterations

$Iter2 := Ans2_3 = 7$

$$M\_Bal2 := Check\_Max\_Bal(MD2) = \begin{bmatrix} 141.51 \\ 13.27 \\ 4.02 \\ 1.02 \\ 0.31 \\ 0.08 \\ 0.02 \end{bmatrix}$$

$\le$ $\Delta Val2 = 0.05$ $Iter2 = 7$ $\text{time}(0) - t_0 = 0.27\ s$