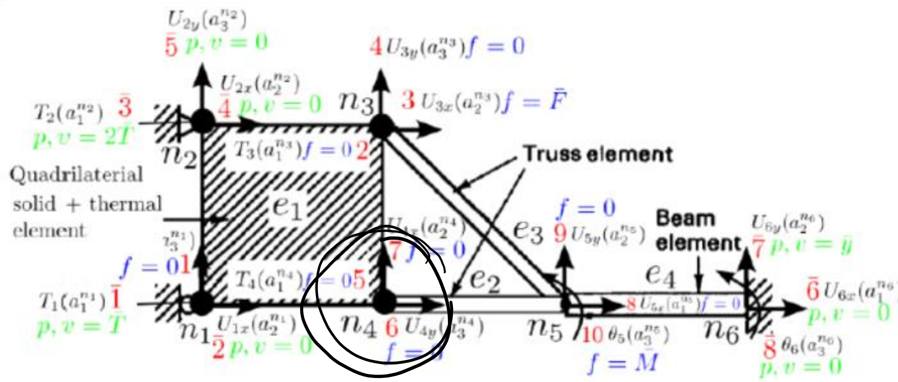
FEM2021/11/11

Thursday, November 11, 2021 2:50 PM

FEM Solver Objects: 4. Node: Data



Node data

ID 4

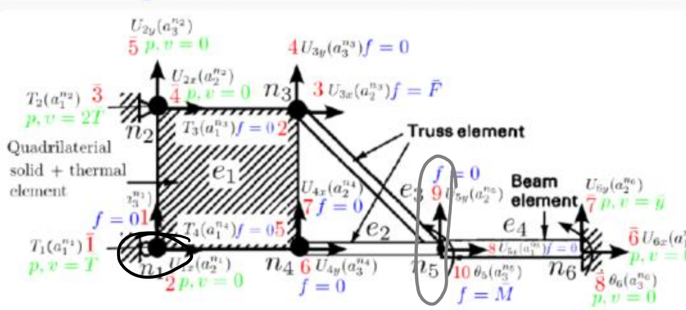
coordinate [1 0]

nndof 3

{PhyDof} list, vector, ... of dofs

```
public:
    void set_nndof(int nndofIn);
    void UpdateNodePrescribedDofForces(VECTOR& Fp);
    ID id;
    VECTOR coordinate;
    vector <PhyDof> ndof;
    int nndof;// number of dofs
};
```

FEM Solver Objects: 5. Dof: Data



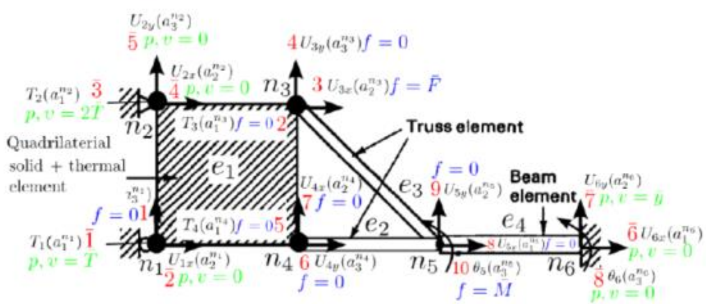
dof 1 @ node 1 : temperature

	p (boolean: prescribed)	value	pos	f (force)	field	component
	1	T	1	?	T	—
dof (3) of node (5)	0	?	9	0	U	2

dof (3) of node (5) 0 | ? | 9 | 0 | U | 2

not needed for your implementation

FEM Solver Objects: 5. Dof



Examples of dof for the structure shown are:

dof	p	pos	v	f	field	index
1 of n_1	true	$\bar{1}$	$\bar{T}$	unknown	T	-
3 of n_1	false	1	unknown	0	U	2
3 of n_5	false	10	unknown	$\bar{M}$	θ	- (a vector in 3D)
2 of n_6	true	$\bar{7}$	$\bar{y}$	unknown	U	2

CFEM\PhyDof.h

```
class PhyDof
{
public:

    PhyDof();

    bool p;    // boolean: whether the dof is prescribed
    int pos;   // position in the global system (for free and prescribed)
    double v; // value of dof
    double f;  // force corresponding to dof

//    F can be stress i can be (0, 1) sigma_{01}
//    FieldF;
//    INDEX i;
};
```

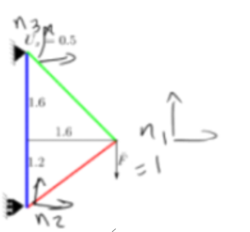
Solution steps

The steps for FEM solution are:

1. Set Element nodal dofs.
2. Set global dofs using element dofs.
3. Compute n_f from n_{dof} and n_p and resize and zero stiffness matrix and force vector.
4. Set global prescribed dofs.
5. Set global free dofs.
6. Set dof (free + prescribed) positions.
7. Set $\mathbf{F}(\mathbf{F}_f)$.
8. Set element dof maps $\mathbf{M}_t^e$.
9. Set element (prescribed) dofs.
10. Compute element stiffness matrix and force vectors.
11. Assemble element stiffnesses and forces to global system.
12. Solve for (free) dofs $\mathbf{a}$ from $\mathbf{Ka} = \mathbf{F}$.
13. Assign $\mathbf{a}$ to nodes and elements.
14. Compute prescribed dof forces: $\mathbf{F}_p$ (if needed).
15. Compute (if needed) output nodes and elements.

402 / 456

These are all class members that you'd be storing in FEM solver



$n_f = 3 \quad n_p = 3 \quad n = 6$

node	P	pos	v	f
1				
2				
3				

$U_f = \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} \begin{bmatrix} \\ \\ \\ \end{bmatrix}$ size n_f

$K_{n_f \times n_f}$

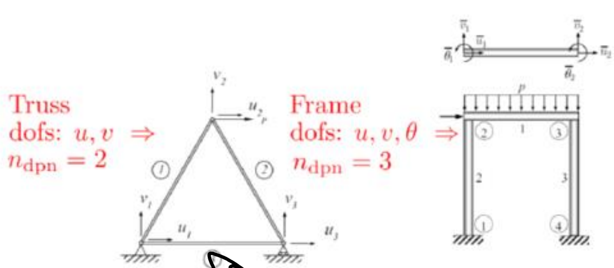
$F_n = \begin{bmatrix} \\ \\ \end{bmatrix}$

$F_p = \begin{matrix} \bar{1} \\ \bar{2} \\ \bar{3} \end{matrix} \begin{bmatrix} \\ \\ \\ \end{bmatrix}$

e	LEM	dof Map	dofs	fe

For the term project, we really don't have the complex steps 1 and 2 since we deal with these structures

Steps 1 & 2: Simplified limited case



- FEM implementation become considerably simpler for problems where all elements are of the same type (regardless of number of physics per element).
- In this case, we define:

$$\boxed{n_{dpn} := \text{Number of dof. per node denoted by ndofpn}} \qquad (448)$$

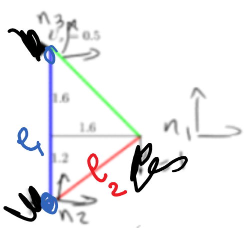
- There would be identity map between element nodal dof and global nodal dofs. That is, there is the same set of dofs used for both.

$$\boxed{n_{\text{dpn}} := \text{Number of dof. per node denoted by ndofpn}} \qquad (448)$$

- There would be identity map between element nodal dof and global nodal dofs. That is, there is the same set of dofs used for both.
- Figure above shows two of such examples:

CFEM\TrussTest.txt this is the INPUT file for the truss example in Truss section.
dim 2 → 2D
ndofpn 2 → 2Dof per node

Nodes
nNodes 3
id crd
1 1.6 1.2
2 0 0
3 0 2.8



Elements
ne 3
id elementType matID neNodes eNodes
1 3 1 2 2 3
2 3 1 2 2 1
3 3 1 2 3 1

LEM nodal element map

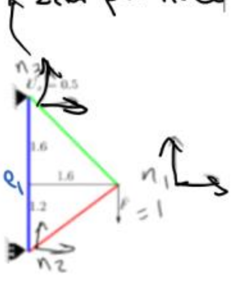
e_3 what material is made of

#3 means Truss elements

1 bar, 2 beam, 4 frame

This would be the end of steps 1 and 2:

2 dof per node



$n_f = 3 \quad n_p = 3 \quad n = 6$

node	p	pos	v	f
1				
2				
3				

$U_p = \begin{array}{c} 1 \\ 2 \\ 3 \end{array}\begin{bmatrix} \\ \\ \\ \end{bmatrix}$ size n_p

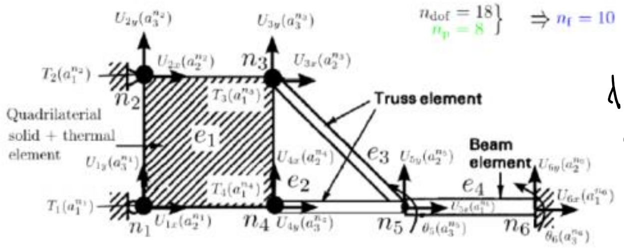
$K_{n_f \times n_f}$

$F_n = \begin{bmatrix} \\ \\ \end{bmatrix}$

$F_p = \begin{array}{c} 1 \\ 2 \\ 3 \end{array}\begin{bmatrix} \\ \\ \\ \end{bmatrix}$

e	LEM	dof Map	dofs	fe
1	[2 3]			
2	[2 1]			
3	[3 1]			

Step 3: Set global number of dofs, stiffness, and force.



total dof from problem $\leftarrow n_{dof} = nnodes \times ndofpn$

$= 3 \times 2 = 6$

$ndof = n_f + n_p$

- n_p (number of prescribed dofs) is an input to FEM analysis.
- n_{dof} is computed in step 2.
- $n_f = n_{dof} - n_p$ is the number of free dofs.
- We size and zero $\mathbf{K}(\mathbf{K}_{ff}) : n_f \times n_f$ and $\mathbf{F}(\mathbf{F}_f) : n_f$ (member of FEM Solver).
- While not necessary, for simplicity we also size and zero $\mathbf{F}_p : n_p$ (member of FEM Solver).

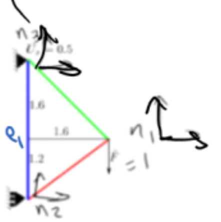
Support forces

411 / 456

PrescribedDOF
np 3
node node_dof_index value
2 1 0.0
3 1 0.5
3 2 0.0

Step 4: Set global prescribed nodal dof

2dof per node



$n_f = 3$ $n_p = 3$ $n = 6$

$ndof = 3 \times 2 = 6$ $n_p = 3 \rightarrow n_f = ndof - n_p = 3$

node	p	pos	v	f
1	0			0
	0			0
2	1		0.0	0
	0			0
3	1		0.5	0
	1		0.6	0

U_p = [ ] 1 2 3, size n_f $n_f \times 1$

$K_{n_f \times n_f}$

F_n = [ ] $n_f \times 1$

F_p = [ ] 1 2 3, $n_p \times 1$

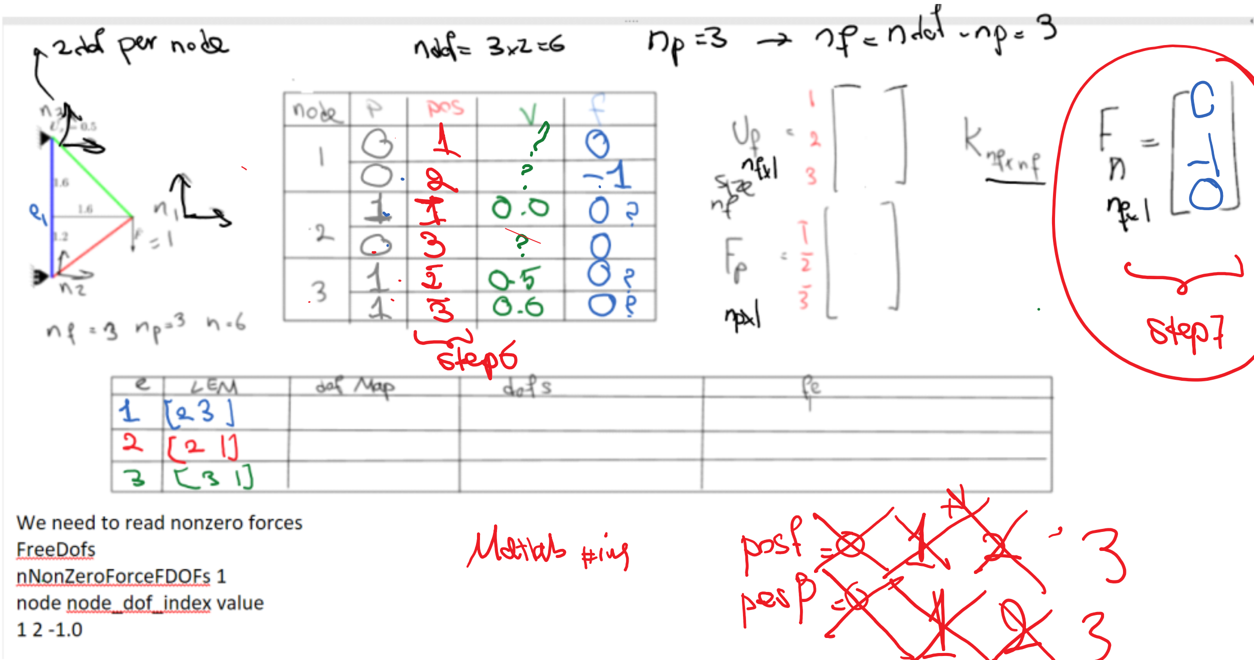
e	LEM	dof Map	dofs	fe
1	[2 3]			
2	[2 1]			
3	[3 1]			

PrescribedDOF
np 3
node node_dof_index value
2 1 0.0
3 1 0.5
3 2 0.0

Step 5: Set global free nodal dof

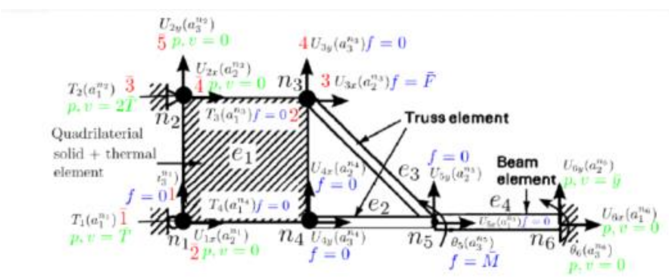
We need to read nonzero forces

FreeDofs
nNonZeroForceFDOFs 1
node node_dof_index value
1 2 -1.0

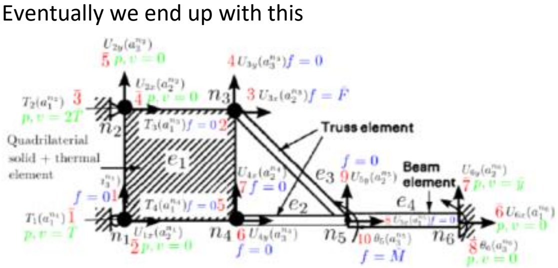


We need to read nonzero forces
FreeDofs
nNonZeroForceFDOFs 1
node node_dof_index value
1 2 -1.0

Step 6: dof positions; Step 7: Set $\mathbf{F}(\mathbf{F}_f)$



Eventually we end up with this

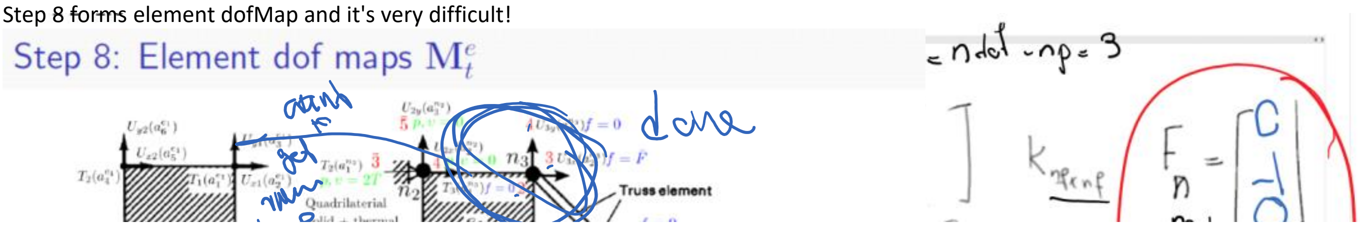


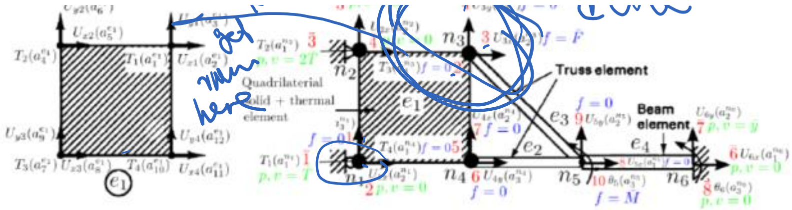
A bit complicated in general

We go with simple structures that are discussed in this project:

Step 8 ~~forms~~ element dofMap and it's very difficult!

Step 8: Element dof maps $\mathbf{M}_t^e$





$K_{n_f \times n_f}$

$F_n = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}$ $n_f \times 1$

step7

- As mentioned, $\mathbf{M}_t^e$ is a vector of size n_{dof}^e that maps elements dofs to global positions.
- For element 1, dofs are ordered as (loop over nodes, then loop over dofs for the node):

$$a_1^e = [a_1^{e1} \quad a_2^{e1} \quad \cdots \quad a_{12}^{e1}]$$
$$= [\ T_1 \quad U_{x1} \quad U_{y1} \mid T_2 \quad U_{x2} \quad U_{y2} \mid T_3 \quad U_{x3} \quad U_{y3} \mid T_4 \quad U_{x4} \quad U_{y4}\ ]$$

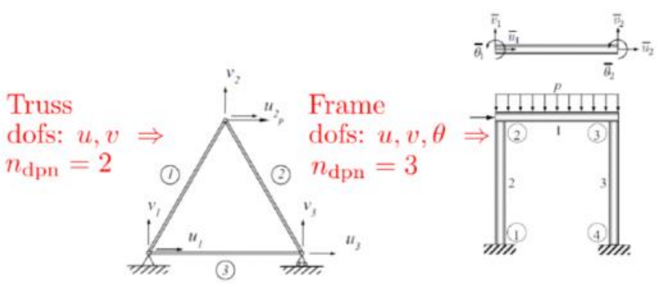
and form

- We need to map these dofs to global dofs and have their position in $\mathbf{M}_t^e$ vector. For example, 1st dof of node 1 ($a_1^{e1} = T_1$) is mapped to first dof of n_3 which has position 2.
- 2nd dof of node 3 ($a_8^{e1} = U_{y2}$) is mapped to 2nd dof of n_1 which has position $\bar{2}(-2)$.
- The map for element e_1 is:

$$\mathbf{M}^{e_1}{}_t = [\ 2 \quad 3 \quad 4 \quad \bar{3} \quad \bar{4} \quad \bar{5} \quad \bar{1} \quad \bar{2} \quad 1 \quad 5 \quad 6 \quad 7\ ]$$

Again, for step 8, we use a simple implementation for the problem in hand:

Step 8: Element dof maps $\mathbf{M}_t^e$: Simplified limited case

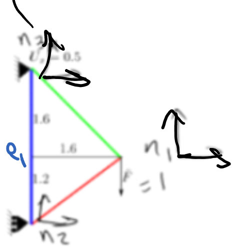


- in Steps 1 and 2 we mentioned that FEM implementation becomes considerably simpler if we assume all nodes share exactly the same set of dofs.
- Examples are bars, beams, trusses, and frames.
- In (448) we defined n_{dpn} (ndofpn) as,

$$n_{\text{dpn}} := \text{Number of dof. per node}$$

- In this limited scenario ith dof of node in element is mapped to ith dof of its corresponding global node.

2 dof per node



$n_f = 3 \quad n_p = 3 \quad n = 6$

ndof = 3x2 = 6

$n_p = 3 \rightarrow n_f = n_{dof} - n_p = 3$

node	P	pos	V	f
1	0	1		0
	0	2		−1
2	1	$\bar{1}$	0	0 ?
	0	3		0
3	1	$\bar{2}$	0.5	0 ?
	1	$\bar{3}$	0.6	0 ?

step6

$U_f = \begin{array}{c} 1 \\ 2 \\ 3 \end{array} \begin{bmatrix} \ \\ \ \\ \ \end{bmatrix}$ $n_f \times 1$, size n_f

$F_p = \begin{array}{c} \bar{1} \\ \bar{2} \\ \bar{3} \end{array} \begin{bmatrix} \ \\ \ \\ \ \end{bmatrix}$ $n_p \times 1$

$\underline{K_{n_f \times n_f}}$

$F_n = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}$ $n_f \times 1$

$\underbrace{\qquad}_{\text{step7}}$

e	LEM	dof Map	dofs	fe

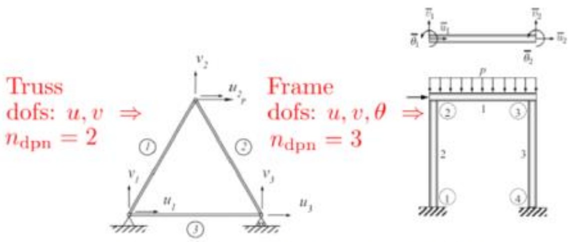
$n_f = 3 \quad n_p = 3 \quad n = 6$

step 6

e	LEM	dof Map	dofs	fe
1	[2 3]	$[\bar{1}\ 3\ \bar{2}\ \bar{3}]_{4\times1}$	$[0\ 0\ 0\ 0]_{4\times1}$ 0.5	
2	[2 1]	$[\bar{1}\ 3\ 1\ 2]_{4\times1}$	$[0\ 0\ 0\ 0]_{4\times1}$	
3	[3 1]	$[\bar{2}\ \bar{3}\ 1\ 2]_{4\times1}$	$[0.5\ 0\ 0\ 0]_{4\times1}$	

step 8 step 9

Step 9: Set element dofs $\mathbf{a}^e$: Simplified limited case



- Similar to steps 1, 2, and 8, step 9 can be greatly simplified if we assume all nodes share exactly the same set of dofs.
- Noting n_{dpn} (ndofpn) = Number of dof. per node, simplified merged steps 8 & 9 are:

dofs = zeros(nedof) element dofs (edof) resized to number of element dofs and zeroed
ecdof = 1 dof counter for element
for en = 1: neNodes number of element nodes
  gn = LEM(en) global node number for element node en
  for endof = 1: ndofpn This number is fixed now, *e.g.*, 2 for 2D trusses
    if (node(gn).dof(endof).p == true)   gndof = endof, we bypass some steps here
      dofs(ecdof) = node(gn).dof(endof).value; e dof val = corresponding global val
    end
    dofMap(ecdof) = node(gn).dof(endof).pos
    ecdof = ecdof + 1   increment counter
  end
end