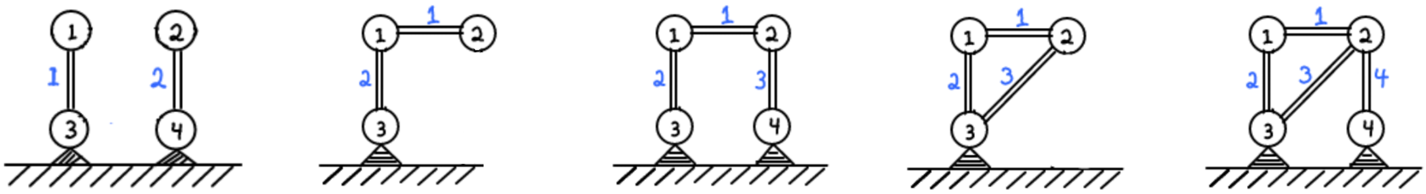
Truss Systems (Linear Approximation)

Force-Balance Matrix and Mechanisms

For each truss system below,

(a) Calculate the force-balance matrix B,
(b) Determine the number of independent mechanisms,
(c) Identify the mechanisms corresponding to each nullspace vector of B^{T},
(d) Give at least two force vectors which will be balanced by internal tension/compression (no collapse).
(e) Write the force vector $f_1^x = f_1^y = 0,\ f_2^x = f_2^y = 1$ as a balanced force plus a force causing motion $\mathbf{f} = \mathbf{f_b} + \mathbf{f_m}$.



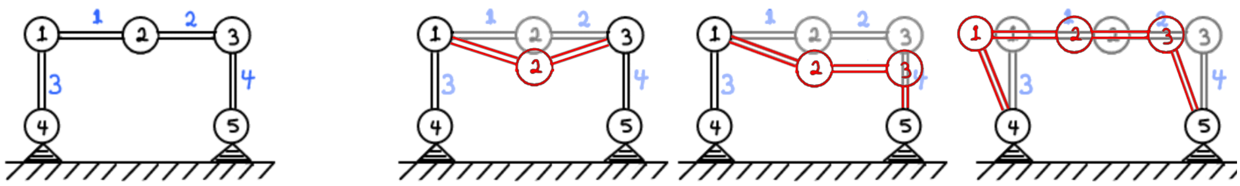
(I) **(II)** **(III)** **(IV)** **(V)**

Identify the mechanisms.

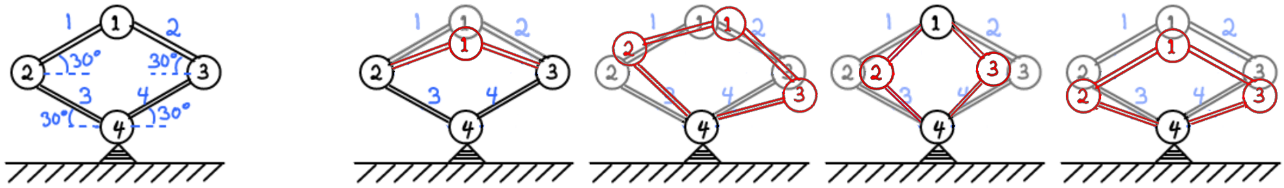
Several possible mechanisms (red bars) of the truss system on the left are given.

(a) Determine the transposed force-balance matrix B^T for the systems on the far left.
(b) Identify (approximately) a motion vector $\mathbf{f_m}$ indicated by the change of nodes from grey to red position.
(c) Check whether the motion vectors from (b) satisfy $B^T\mathbf{f_m} = 0$. Is $\mathbf{f_m}$ a mechanism?

(I)



(II)



Row Space, Column Space, Nullspace

For the matrices below give bases for the **(a)** row space, **(b)** column space, and **(c)** nullspace.

(I) $\begin{bmatrix} 1 & 2 & 1 & 2 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ **(II)** $\begin{bmatrix} 1 & 2 & 1 & 2 \\ 2 & 4 & 3 & 7 \\ -1 & -2 & -3 & -8 \end{bmatrix}$ **(III)** $\begin{bmatrix} 1 & 2 & 1 & 2 & 1 \\ 0 & 0 & 1 & 3 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$ **(IV)** $\begin{bmatrix} 1 & 2 & 1 & 2 & 1 \\ 2 & 4 & 3 & 7 & 3 \\ -1 & -2 & -3 & -8 & -2 \end{bmatrix}$

(V) $\begin{bmatrix} 0 & 1 & 3 & 2 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ **(VI)** $\begin{bmatrix} 1 & 2 & 1 & 2 \\ -2 & -4 & -1 & -1 \\ -3 & -6 & -1 & 0 \\ 2 & 4 & 3 & 7 \end{bmatrix}$ **(VII)** $\begin{bmatrix} 0 & 1 & 3 & 2 & 1 \\ 0 & 0 & 1 & 3 & 1 \\ 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$ **(VIII)** $\begin{bmatrix} 1 & 2 & 1 & 2 & 1 \\ -2 & -4 & -1 & -1 & -1 \\ 1 & 2 & -1 & -4 & 1 \\ 2 & 4 & 1 & 1 & -1 \end{bmatrix}$

- Use the command `null(<matrix>)` to compute nullspaces with MatLab. The default output of `null` is an orthogonal basis for the nullspace – i.e. a minimal set of **perpendicular** vectors with **length 1** that span the nullspace. Including the switch `'r'` will give instead **integer** vectors (when possible) calculated using the LU decomposition.

  For example you could compute the nullspace of the matrix $A = \begin{bmatrix} 1 & 2 & 3 & -1 \\ 0 & 0 & 1 & 2 \end{bmatrix}$ as follows.

```
>> A = [1 2 3 -1 ; 0 0 1 2 ] ;
>> null(A)          % orthogonal nullspace
ans =
    -0.9504   -0.1709
    -0.0114    0.8542
     0.2781   -0.4393
    -0.1390    0.2196
>> A * ans          %  check A*null = 0
ans =
    1.0e-14 *
    -0.1110   -0.0278
    -0.0222         0
```

```
>> null(A, 'r')  % integer nullspace
ans =
    -2     7
     1     0
     0    -2
     0     1
>> A * ans          %  check A*null = 0
ans =
     0     0
     0     0
```

- To compute the row space of a matrix you can use the MatLab commands `ref(<matrix>)` and `rref(<matrix>)`. The command `ref` computes the "*row echelon form*" of a matrix; this is a fancy name for the matrix U from the LU decomposition. Recall that the nonzero rows of U are a basis for the rowspace.
  The command `rref` computes the "*reduced row echelon form*" of the matrix; this is like U but with extra row operations dividing rows so that pivots are 1, and applying back substitution to remove elements above pivots.

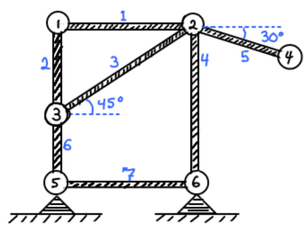
  For example you could compute the row space of the matrix $A = \begin{bmatrix} 4 & 2 & 6 & 2 \\ -2 & -1 & -1 & -3 \\ 2 & 1 & 5 & -1 \end{bmatrix}$ as follows.

```
>> A = [ 4 2 6 2 ; -2 -1 -1 -3 ; 2 1 5 -1 ] ;
>> ref(A)           % row echelon form
ans =
     4     2     6     2
     0     0     2    -2
     0     0     0     0
```

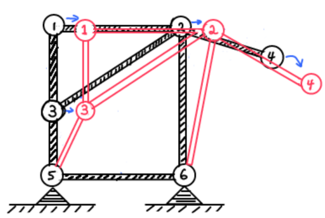
```
>> rref(A)          % reduced row echelon form
ans =
    1.0000    0.5000         0    2.0000
         0         0    1.0000   -1.0000
         0         0         0         0
```

  The nonzero rows in the results above are a basis for the rowspace.

- Since force-balance matrices for truss systems are mostly 0, when entering them into MatLab it is easier to begin with a matrix of all 0 and then add the nonzero entries rather than typing in the entire matrix. To create a matrix of zeros, use the command `zeros(<#rows>,<#cols>)`. For example you could enter the truss system to the right (4 moving nodes and 6 bars) as follows.



```
>> B = zeros(4*2, 6);       % 4 moving joints, 6 bars
>> B(1:2,1) = [-1, 0];                       B(3:4,1) = [ 1, 0];                        % bar 1
>> B(1:2,2) = [ 0, 1];                       B(5:6,2) = [ 0,-1];                        % bar 2
>> B(3:4,3) = [ cos(pi/4), sin(pi/4)];       B(5:6,3) = [-cos(pi/4),-sin(pi/4)];        % bar 3
>> B(3:4,4) = [ 0, 1];                       % other end does not move                  % bar 4
>> B(3:4,5) = [-cos(pi/6), sin(pi/6)];       B(7:8,5) = [ cos(pi/6),-sin(pi/6)];        % bar 5
>> B(5:6,6) = [ 0, 1];                       % other end does not move                  % bar 6
>> null(B')
ans =
     0.4472   -0.2582
    -0.0000    0.0000
     0.4472   -0.2582
    -0.0000    0.0000
     0.4472   -0.2582
    -0.0000    0.0000
    -0.5854   -0.2394
    -0.2394   -0.8618
```



Mechanism corresponding to the first nullspace vector