Dynamics of Machines and Mechanisms

Introduction SPACAR

Ronald Aarts
Mechanical Automation (WA)
Room: HR Z 2.29
Phone: (053) 489 2557
Email: `R.G.K.M.Aarts@utwente.nl`

Info: `http://www.wa.ctw.utwente.nl/lectures/113173/`
`http://www.wa.ctw.utwente.nl/software/spacar/`

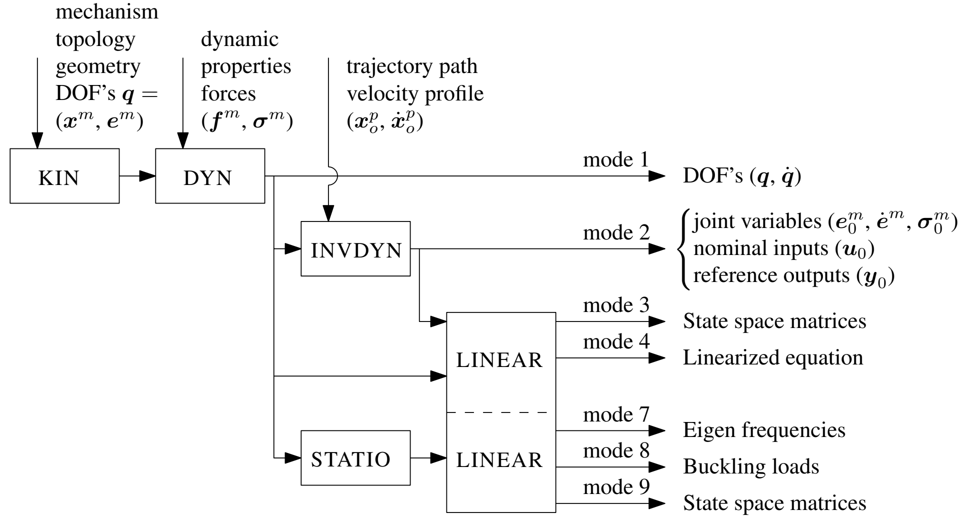
Today's program

- Overview SPACAR software package
  → Download & install
  → Basic usage
- Some examples of advanced usage
- Two exercises for "guided self-education"

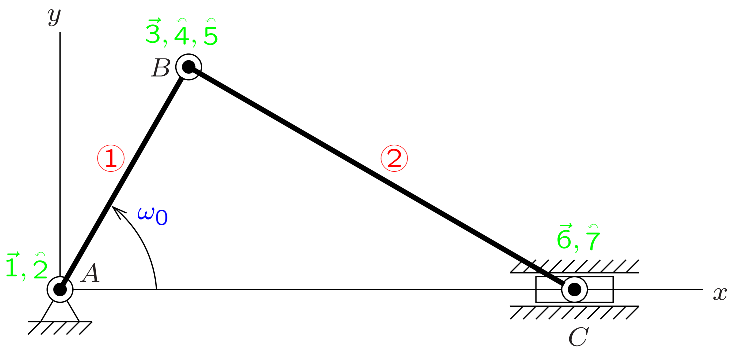
Overview SPACAR software package

- A GUI to create/modify systems
  → link is provided via BlackBoard.
  Current version is 2.0.23.
- The SPACAR toolbox for MATLAB/SIMULINK
  → download from WA site.
  Current version is 2010.
  The 2011 version will be released on Monday March 21, 2011, and includes an new element.
- Visualisation with the MATLAB script `spavisual` (included):
  Mode shapes, buckling modes, stress distribution in beams, analysis of over and under constraint systems.
  The 2011 version will offer an improved user interface.

Modules in SPACAR/MATLAB



Slider-crank mechanism



Time series: two crank rotations.

Input data file `crank.dat`:

(created with the GUI or your favourite text editor)

```
PLBEAM  1  1  2  3  4
PLBEAM  2  3  5  6  7

X       1     0.00    0.
X       3     0.15    0.
X       6     0.45    0.

FIX     1  1
FIX     1  2
FIX     6  2
INPUTX  2  1

END
HALT
```

```
XM      6     0.033380
EM      2     0.222500

INPUTX  2  1  0.    150.    0.
TIMESTEP      0.1   100

END
END
```


Output of forward dynamics computation `spacar(1,'crank')` are:

- a log file with e.g. error messages (`crank.log`) and
- a SPACAR Binary Data file (`crank.sbd`).

The data is also available in MATLAB arrays:

`ndof`: Computed number of DOF's.

`lnp`: Nodes location matrix: location of j^{th} coordinate of node i is `lnp(i,j)`.

`le`: Elements location matrix: location of j^{th} generalized deformation of element i is `le(i,j)`.

`time`: Array with time steps.

`x, xd, xdd`: Nodal coordinates, velocities and accelerations.

`fx`: Forces/moments.

`e, ed, edd`: Generalized deformations, velocities and accelerations.

`sig`: Generalized stress resultants.

Input data file `crankfl.dat`:

```
PLBEAM     1   1   2   3   4
PLBEAM     2   3   5   6   7
PLBEAM     3   6   7   8   9

X          1       0.000     0.000
X          3       0.150     0.000
X          6       0.300     0.000
X          8       0.450     0.000

FIX        1
FIX        8   2
INPUTX     2   1
DYNE       2   2   3
DYNE       3   2   3

END
HALT
```

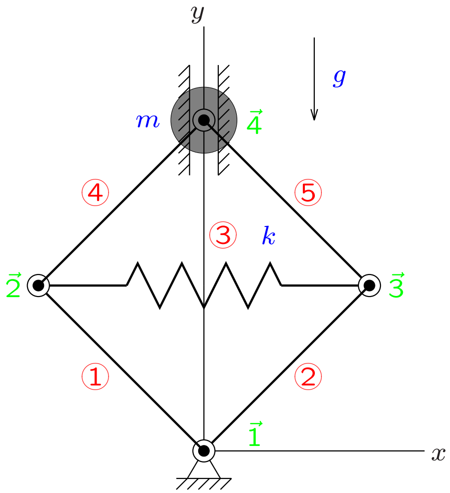
```
XM         8     0.033380
EM         2     0.222500
EM         3     0.222500

ESTIFF     2     0.000000   13.359623
ESTIFF     3     0.000000   13.359623

INPUTX     2  1    0.0    150.0     0.0
TIMESTEP           0.1    100
STARTDE    2  2    0.0      0.0
STARTDE    2  3    0.0      0.0
STARTDE    3  2    0.0      0.0
STARTDE    3  3    0.0      0.0

END
END
```

Four-bar mechanism



Equation of motion:

$$m\ddot{e}_3 + \sqrt{2}m(\dot{e}_3)^2 + ke_3 = mg$$

Linearized equation of motion:

$$m\delta\ddot{e}_3 + 2\sqrt{2}m\dot{e}_3\delta\dot{e}_3 + (k - \sqrt{2}mg + 2\sqrt{2}m\ddot{e}_3 + 5m(\dot{e}_3)^2)\delta e_3 = 0$$

Input data file `fourbar.dat`:

```
PLTRUSS 1  1  2
PLTRUSS 2  1  3
PLTRUSS 3  2  3
PLTRUSS 4  2  4
PLTRUSS 5  3  4
X       1     0.       0.
X       2    -0.7071   0.7071
X       3     0.7071   0.7071
X       4     0.       1.4142
FIX     1
FIX     4  1
DYNE    3  1
END
HALT
```

```
XM      4     1.
XF      4     0.     -10.
ESTIFF  3     1.4142
STARTDE 3  1  0.       1.
END
END
```


Output of forward dynamics computation `spacar(1,'fourbar')` are:

- a log file with e.g. error messages (`fourbar.log`) and
- a SPACAR Binary Data file (`fourbar.sbd`).

Extra output of linearization computation `spacar(4,'fourbar')` is:

- a SPACAR Binary Matrix data file (`fourbar.sbm`).

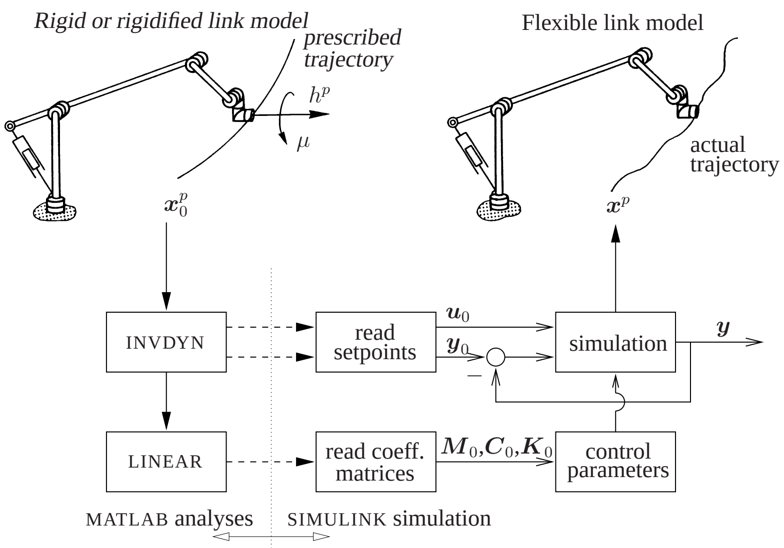
The data is also available in additional MATLAB arrays:

`m0`, `c0`, `k0`, `g0`: Matrices of the linearized equation of motion. Note that there are differences with the definitions in the lecture notes: `k0` is the sum of the structural stiffness and the dynamic stiffening terms.

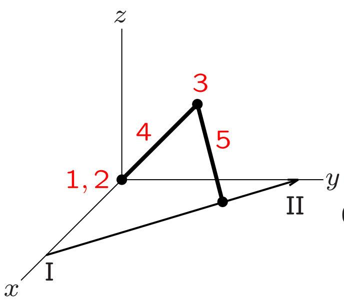
The binary files can be loaded, checked, modified with utilities:

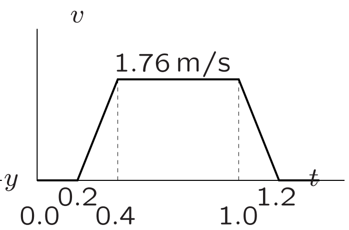
- `loadsbd`, `loadsbm`.
- `checksbf`.
- `repinsbf`.

SPACAR in MATLAB and SIMULINK



Rigid spatial manipulator mechanism





Spatial manipulator with two (rigid) links (`BEAM`) and three rotational joints (`HINGE`).

Prescribed trajectory: Tip motion along straight line with desired velocity profile.

Procedure:

`invdyn` computation `spacar(2,'robotinv')`:

- Define the correct degrees of freedom and deformation modes for the inverse dynamics in the kinematics part.
- Define the trajectory.
- Define the nominal inputs and reference outputs to be used in a future simulation.

`linear` computation `spacar(3,'robotinvlin')` using the data from the previous `invdyn` run:

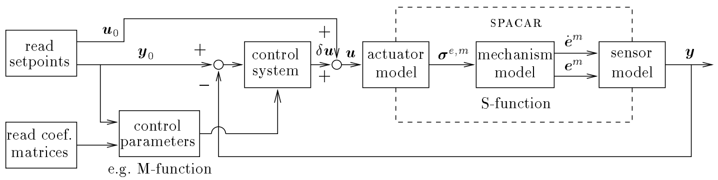
- Define the degrees of freedom for the forward dynamics.
- Define the inputs and outputs to be used in a future simulation.

Simulation (SIMULINK):

Open-loop:

- Read nominal inputs u_0 and connect to mechanism inputs.
- Compare mechanism outputs to reference outputs y_0.

Closed-loop:



- Nominal inputs u_0 can be used as a feed-forward signal.
- Compare mechanism outputs y to reference outputs y_0 and use the difference for feed-back control.
- Read coefficient matrices and other data to compute control parameters.

Flexible spatial manipulator mechanism

- Redo `invdyn` and `linear` analyzes with modified input files including deformations representing the flexibility of the beams.
- Redo the SIMULINK simulation with this mechanism. A non-linear simulation with SPASIM can be very time consuming!
- Alternatively, a linearized set of equations can be solved in SIMULINK using the LTV block.
- Further reduction in CPU time is possible by applying modal techniques.