Control System in LS-DYNA

Charlotte Keisser[1], Isheng Yeh[2]

[1]DYNAmore France, Versailles, France
[2]Livermore Software Technology Corporation, Detroit, MI, USA

1 Introduction

Control System is a very old concept aimed at improving or changing the behavior of any dynamic system.
In our daily lives, there are many control applications like in a toaster or in room temperature regulators but also in our cars when using cruise control or ABS. More recently, Control System appears to be also very useful and powerful in pre-crash safety or in modeling Human Body response.

Since the study and design of automatic control has become more and more popular, it has been part of LS-DYNA research and development over the last 3 years.

2 Definition

Control is the process of making a system of design variables conforms to some desired values.

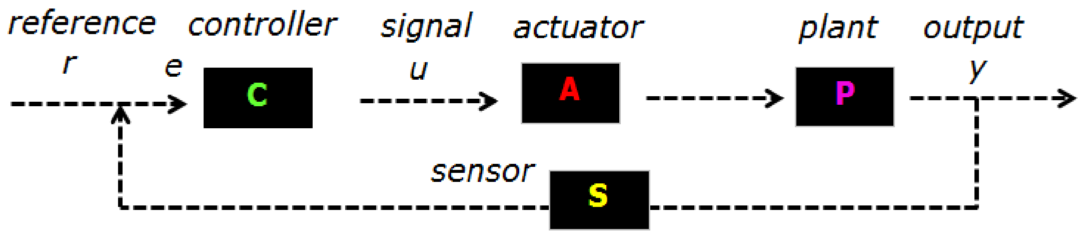


Fig.1: Control System loop

In a general control system loop, the **plant** is the physical part to be controlled and has the form of a linear system or a FE model. The **sensor** provides measurement of the plant output when the **actuator** converts the control signal to power signal.

3 Control Capabilities in LS-DYNA

Nowadays, several control system toolbox in MATLAB/Simulink, Scilab, Octave … already exist.
Most of the time, FEM software uses Control System capabilities with MATLAB/Simulink Control System toolbox.

In LS-DYNA, we will propose two approaches. The first approach will be the classical one, meaning that you will be able to extract a MATLAB file from a LS-DYNA run and then run MATLAB for control purposes. The second approach will be fully included in LS-DYNA, meaning that you will be able to run LS- DYNA for control as well as for FEM.

The control capabilities in LS-DYNA are the following:

- Already available:
  - The `PIDCTL` and `DELAY` functions defined inside the `*DEFINE_CURVE_FUNCTION` keyword;

  - The **`*SENSOR`** keyword;

- On development:
  - An internal Control System Toolbox (**`*CTRLLER_`** keywords) with control functions similar to those proposed by MATLAB;
  - The **`*CTRLLER_PLANT`** keyword that creates the interface between the FE simulation and the control;
  - The piezo-electric material (**`*MAT_PZELECTRIC`** keyword), a smart material including control;

- Just started:
  - A Graphical User Interface for the internal control toolbox.

3.1 PIDCTL and DELAY

`PIDCTL`*(lmeas, ref, lref, kp, lkp, ki, lki, kd, lkd, tf, ltf, ei0)*

The **`PIDCTL`** function evaluates the signal of a PID controller:

$u(t) = kp \text{ x } e(t) + ki \text{ x } \int e(\sigma)d\sigma + kd \text{ x } de(t)/dt$.

Where *e(t)* is the control error defined as the difference between the reference value *ref* and the measured value, the value of the curve *lmeas* :

e(t) = ref – f(lmeas).

The control parameters are proportional gain *kp*, integral gain *ki*, derivative gain *kd* and low-pass filter *tf* for the derivative calculation. The initial integral value at time 0 is *ei0*.

lref, lkp, lki, lkd and ltf are the optional curves for control parameters. When a curve is defined, the input parameter serves as a scale factor for the curve.

`DELAY`*(n, tdly, ydef)*

Delays curve *n* by *tdly* and sets the delayed curve value to *ydef* when time is lower than *tdly*:

f_delay(time) = f(time-tdly)   if time ≥ tdly,

fydef   else.

<u>Example:</u>
SAFER A-HBMs use feedback PID control of line muscles to stabilize the posture.

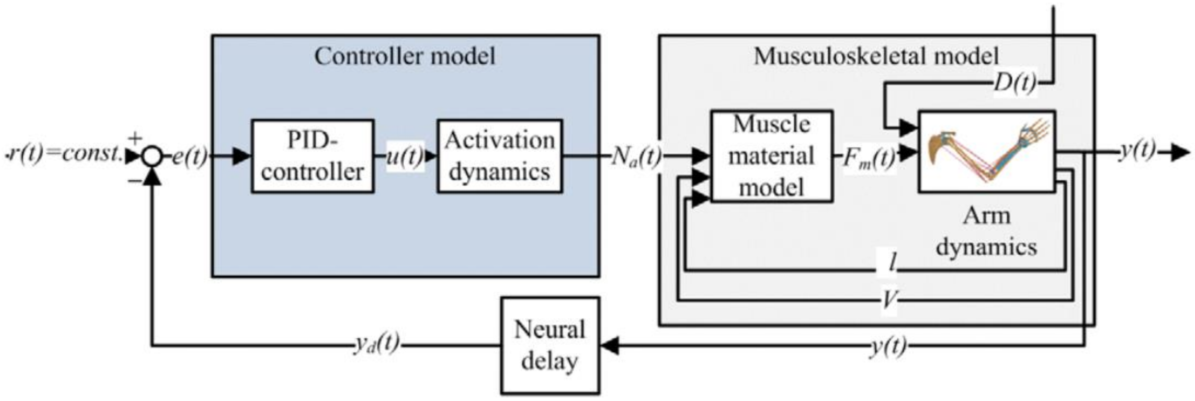


Fig.2: Illustration of the neuromuscular feedback control loop used for the elbow joint by Östh et Al (2012b) [1].

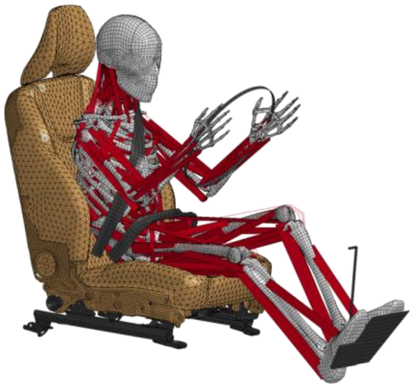
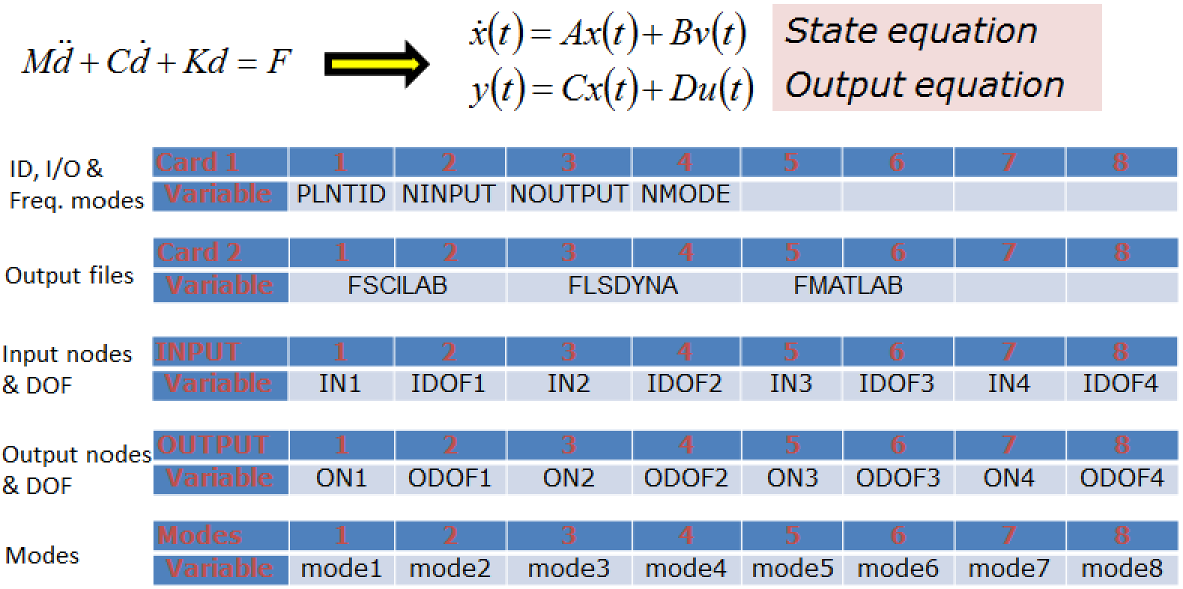


Fig.3: SAFER Active HBM in a driver braking simulation [2].

3.2 FE and Control connection

The **`*CTRLLER_PLANT`** keyword is the gate between LS-DYNA FE simulation and the Control. This keyword has several uses:

- Perform mode truncation on control data;
- Perform LQR control;
- Perform the extraction of a "control file", converting FE data into Control format data (state equation) and written in a MATLAB, or SCILAB or DYNA format.



$$M\ddot{d}+C\dot{d}+Kd=F \Longrightarrow \begin{aligned} \dot{x}(t)&=Ax(t)+Bv(t) \\ y(t)&=Cx(t)+Du(t) \end{aligned}$$

State equation
Output equation

ID, I/O & Freq. modes	Card 1	1	2	3	4	5	6	7	8
	Variable	PLNTID	NINPUT	NOUTPUT	NMODE				

Output files	Card 2	1	2	3	4	5	6	7	8
	Variable	FSCILAB		FLSDYNA		FMATLAB			

Input nodes & DOF	INPUT	1	2	3	4	5	6	7	8
	Variable	IN1	IDOF1	IN2	IDOF2	IN3	IDOF3	IN4	IDOF4

Output nodes & DOF	OUTPUT	1	2	3	4	5	6	7	8
	Variable	ON1	ODOF1	ON2	ODOF2	ON3	ODOF3	ON4	ODOF4

Modes	Modes	1	2	3	4	5	6	7	8
	Variable	mode1	mode2	mode3	mode4	mode5	mode6	mode7	mode8

*Fig.4: **`*CTRLLER_PLANT`** keyword card*

3.3 Internal Control Toolbox

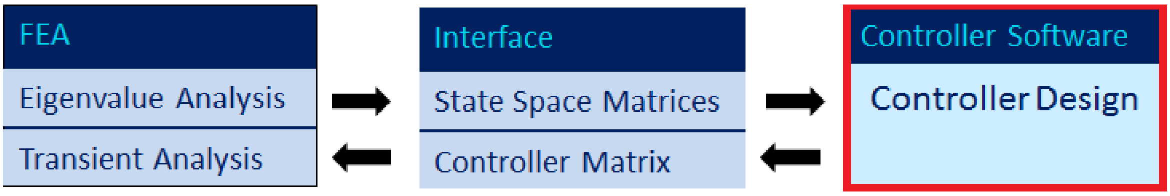
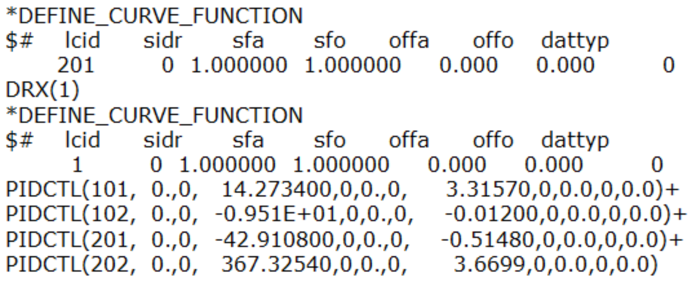


Fig.5: Interaction between LS-DYNA FE simulation and Control

When applying control, the standard process is to first identify and model the system in the appropriate control format. Then the second step is the analysis of the system dynamic property (the stability for example). The last step, the control design, is used when the dynamic is not satisfying.

The LS-DYNA internal control system toolbox will have the main control tools:
- System definition in the time or frequency domain;
- System analysis;
- Solvers;
- Control tools;
- Model reduction;
- System connections.

Example:
A very simple example of control application is the control of a vertical vibration. In this example, we run one implicit simulation of LS-DYNA with the ***CTRLLER_PLANT** keyword to obtain the model in control format, we play with the control tools (PID and LQR control). Once the right controller is found, we use the control coefficients and add it in a final explicit run of LS-DYNA using the **PIDCTL** function of ***DEFINE_CURVE_FUNCTION** and have the vibrations stop.



```
*DEFINE_CURVE_FUNCTION
$#    lcid      sidr        sfa         sfo      offa      offo    dattyp
       201         0  1.000000  1.000000      0.000      0.000          0
DRX(1)
*DEFINE_CURVE_FUNCTION
$#    lcid      sidr        sfa         sfo      offa      offo    dattyp
         1         0  1.000000  1.000000      0.000      0.000          0
PIDCTL(101,  0.,0,   14.273400,0,0.,0,     3.31570,0,0.0,0,0.0)+
PIDCTL(102,  0.,0,  -0.951E+01,0,0.,0,    -0.01200,0,0.0,0,0.0)+
PIDCTL(201,  0.,0,  -42.910800,0,0.,0,    -0.51480,0,0.0,0,0.0)+
PIDCTL(202,  0.,0,   367.32540,0,0.,0,      3.6699,0,0.0,0,0.0)
```

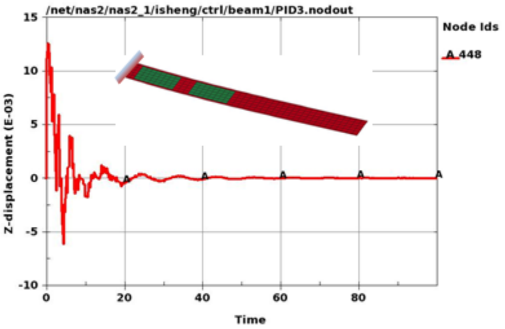
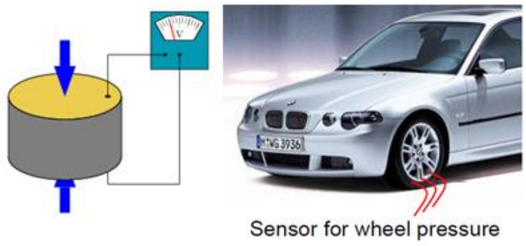


Fig.6: Control used to stop vertical vibration

3.4 Piezo-electric material

The piezo-electric is a material including control. It works in two ways:
- As a sensor by generating electric potential when mechanical stress occurs;
- As an actuator by applying an electric field to obtain mechanical strain.



$$[\sigma] = [c^E][\varepsilon] - [e]^T E$$
$$D = [e][\varepsilon] + [d^S]E$$

Fig.7: Piezo-electric material

The ***MAT_ADD_PZELECTRIC** and ***BOUNDARY_PZEPOT** keywords are currently on development in LS-DYNA.

4 Summary

The control system in LS-DYNA is not yet available (except for the **PIDCTL** and **DELAY** functions). We are currently looking for applications and feedbacks.
For now, the control system toolbox is restricted to linear and time invariant systems and the ***CTRLLER_PLANT** keyword is limited to implicit SMP simulations. But we are planning to expand and add more capabilities regarding to customer needs.

5 Literature

[1] Östh, J. ; Brolin, K. ; Olafsdottir, J. ; et al: "Muscle Activation Strategies in Human Body Models for the development of integrated safety", the 24th ESV Conference Proceeding, 2015.

[2] Östh, J. ; Brolin, K. ; Olafsdottir, J. : "Modeling Active Human Muscle responses during Driver and Autonomous Avoidance Maneuvers.", Chalmers University library.