INTERCONNECTED SYSTEMS

TEARING and ZOOMING CONTROL

Chaire Francqui, Lecture V, May 19, 2004

Université catholique de Louvain

Road Map

What is a dynamical system ?

Road Map

What is a dynamical system ?

Historically:

- **'Closed' systems $\rightsquigarrow$** $\frac{d}{dt}x = f(x)$ **very limited**
- **Input/output map?**
- **Input/state/output structure ?**

Assumes I/O partition. Possible? Obtainable? How? Needed?

- **Other possibilities? CS? Graph theory? Object oriented modeling?**

Road Map

What is a dynamical system ?

What is a mathematical model, really? $\rightsquigarrow$

Dynamical system := $\boxed{(\mathbb{T}, \mathbb{W}, \mathfrak{B})}$ **with** $\mathfrak{B} \subseteq (\mathbb{W})^{\mathbb{T}}$ **the 'behavior'.**

Road Map

What is a dynamical system ?

Dynamical system := $\boxed{(\mathbb{T}, \mathbb{W}, \mathfrak{B})}$ with $\mathfrak{B} \subseteq (\mathbb{W})^{\mathbb{T}}$ the 'behavior'.

Behavioral eq'ns contain latent variables

$\rightsquigarrow$ elimination thms, algorithms.

Road Map

What is a dynamical system ?

Dynamical system := $\boxed{(\mathbb{T}, \mathbb{W}, \mathfrak{B})}$ **with** $\mathfrak{B} \subseteq (\mathbb{W})^{\mathbb{T}}$ **the 'behavior'.**

Basic notions

- **Controllability** $\rightsquigarrow$ **image representation**
- **Observability**
- **(Dissipative systems)**
- **(Stability)**
- **State** $\rightsquigarrow$ **state representation algorithms**

Road Map

What is a dynamical system ?

Dynamical system := $(\mathbb{T}, \mathbb{W}, \mathfrak{B})$ with $\mathfrak{B} \subseteq (\mathbb{W})^{\mathbb{T}}$ the 'behavior'.

Basic problems

- Modeling from data (system ID)
- Modeling by interconnecting components
- Control (= interconnection) $\rightsquigarrow$ LQ, $\mathcal{H}_2$, $\mathcal{H}_\infty$
- ...

Road Map

What is a dynamical system ?

Dynamical system := $\boxed{(\mathbb{T}, \mathbb{W}, \mathfrak{B})}$ **with** $\mathfrak{B} \subseteq (\mathbb{W})^{\mathbb{T}}$ **the 'behavior'.**

Projects

- **N-d systems and PDE's (Rocha, Shankar, Pillai, Zerz, Oberst)**
- **Software**
- **Stochastic systems**
- **...**

THEME

1. **Modeling by tearing and zooming**
   - General ideas
   - Terminals
   - Modules
   - Interconnection architecture
   - Examples
   - RTCT circuits

THEME

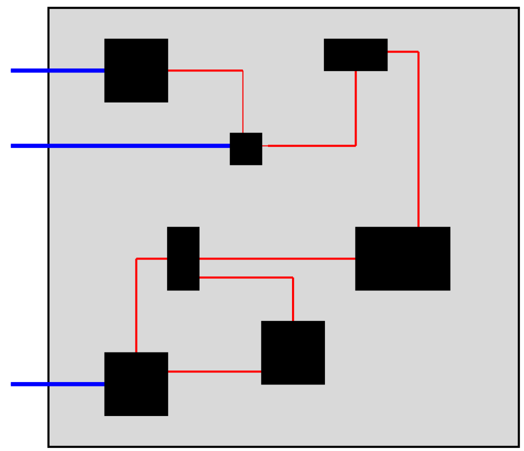
1. **Modeling by tearing and zooming**
   - General ideas
   - Terminals
   - Modules
   - Interconnection architecture
   - Examples
   - RTCT circuits
2. **Control**
   - Control by interconnection
   - Implementability

THEME

1. **Modeling by tearing and zooming**
   - General ideas
   - Terminals
   - Modules
   - Interconnection architecture
   - Examples
   - RTCT circuits
2. **Control**
   - Control by interconnection
   - Implementability
3. **Conclusions**

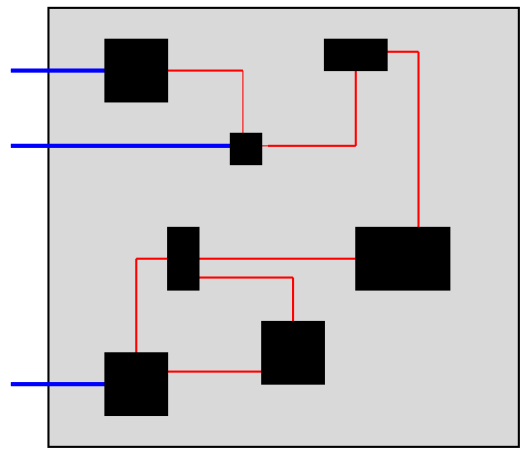
TEARING and ZOOMING

How do we model a complex interconnected system?



TEARING and ZOOMING

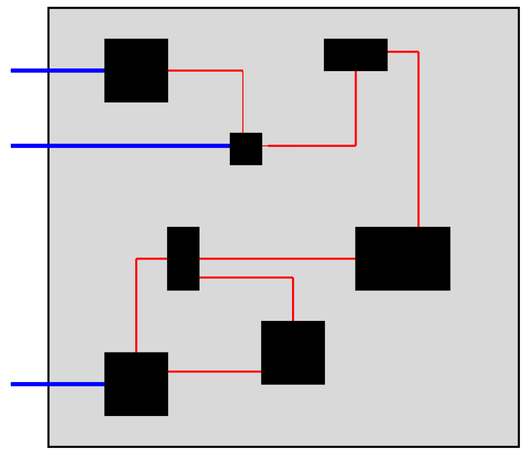
How do we model a complex interconnected system?



When systems are interconnected, what really happens?
How do we obtain a model from
the components and the interconnections?

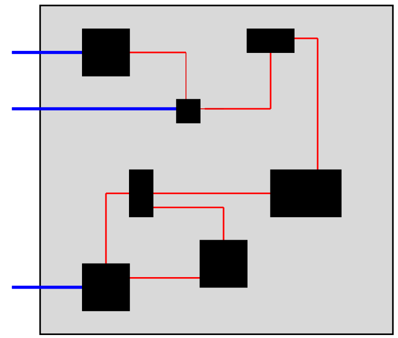
TEARING and ZOOMING

How do we model a complex interconnected system?



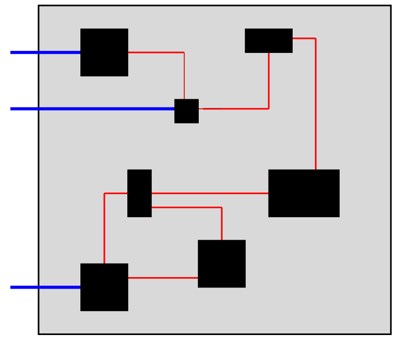
'Tearing' the system into subsystems,
and, in order to model, 'zooming' on the individual subsystems

TEARING and ZOOMING



The ingredients of the language and methodology that we propose:

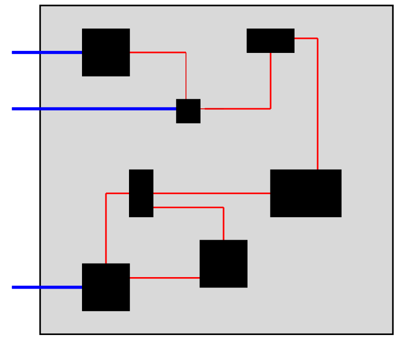
TEARING and ZOOMING



The ingredients of the language and methodology that we propose:

1. ***Modules*** **: the subsystems**

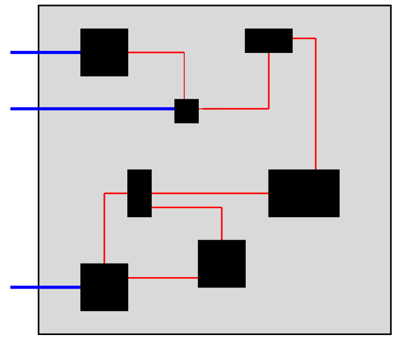
TEARING and ZOOMING



The ingredients of the language and methodology that we propose:

1. ***Modules*** **: the subsystems**
2. ***Terminals*** **: the physical links between subsystems**

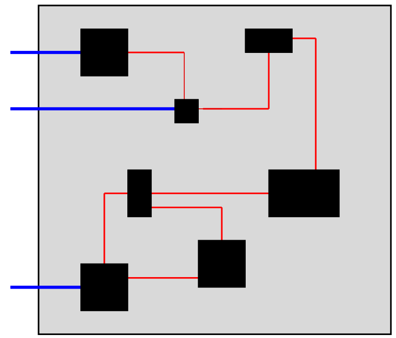
TEARING and ZOOMING



The ingredients of the language and methodology that we propose:

1. ***Modules*** **: the subsystems**
2. ***Terminals*** **: the physical links between subsystems**
3. **The *interconnection architecture* :**
   the layout of the modules and their interconnection

TEARING and ZOOMING



The ingredients of the language and methodology that we propose:

1. ***Modules*** **: the subsystems**
2. ***Terminals*** **: the physical links between subsystems**
3. **The *interconnection architecture* :**
   the layout of the modules and their interconnection
4. **The *manifest variable assignment* :**
   which variables does the model aim at?

TEARING and ZOOMING

Features:

- Reality — 'physics' — based
- Uses behavioral systems concepts
  more akin to bond-graphs and across/through variables,
  than to input/output thinking.
- Hierarchical: allows new systems to be build from old
- Models are reusable, generalizable & extendable
- Assumes that accurate and detailed modelling is the aim

TEARING and ZOOMING

Features:

- **Reality — 'physics' — based**
- **Uses behavioral systems concepts**
  more akin to bond-graphs and across/through variables, than to input/output thinking.
- **Hierarchical: allows new systems to be build from old**
- **Models are reusable, generalizable & extendable**
- **Assumes that accurate and detailed modelling is the aim**

System theory with its inputs and outputs and signal flow graphs, as implemented e.g. in SIMULINK© is hopelessly inadequate. MODELICA© is much better.

TERMINALS

A ***terminal*** is specified by its ***type***.

TERMINALS

A *terminal* is specified by its *type*.

The *type* implies an ordered set of *terminal variables.*

TERMINALS

A *terminal* is specified by its *type*.

The *type* implies an ordered set of *terminal variables.*

Example:

**An electrical terminal (type)
implies a (voltage, current) pair of real terminal variables.**

Examples

Type of terminal	Variables	Signal space
electrical	(voltage, current)	$\mathbb{R}^2$
mechanical (1-D)	(force, position)	$\mathbb{R}^2$
mechanical (2-D)	((position, attitude), (force, torque))	$(\mathbb{R}^2 \times S^1) \times (\mathbb{R}^2 \times T^*S^1)$
mechanical (3-D)	((position, attitude), (force, torque))	$(\mathbb{R}^2 \times S^2) \times (\mathbb{R}^2 \times T^*S^2)$
thermal	(temp., heat flow)	$\mathbb{R}^2$
fluidic	(pressure, flow)	$\mathbb{R}^2$
thermal - fluidic	(pressure, temp., mass flow, heat flow)	$\mathbb{R}^4$

Examples

Type of terminal	Variables	Signal space
chemical		
input	u	$\mathbb{U} \subseteq \mathbb{R}$
output	y	$\mathbb{Y} \subseteq \mathbb{R}$
m-dim input	$(u_1, u_2, \ldots, u_m)$	$\mathbb{U} \subseteq \mathbb{R}^{\mathrm{m}}$
p-dim output	$(y_1, y_2, \ldots, y_p)$	$\mathbb{Y} \subseteq \mathbb{R}^{\mathrm{p}}$
etc.	etc.	etc.

MODULES

A ***module*** is specified by

- its ***type***,
- its ***parametrization***,
- and its ***parameter values***.

MODULES

A *module* is specified by

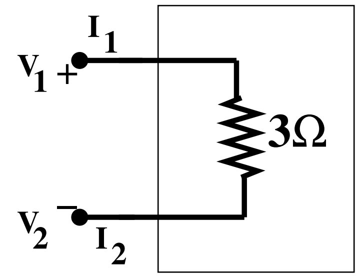
- its *type*,
- its *parametrization*,
- and its *parameter values*.

The idea is the following.

By specifying the module type, we give the variables living on its terminals. We want a fully automated way of specifying the behavior of these variables. This typically happens by specifying some parameters, and a map, the parametrization, which maps these parameters into the correct behavior.

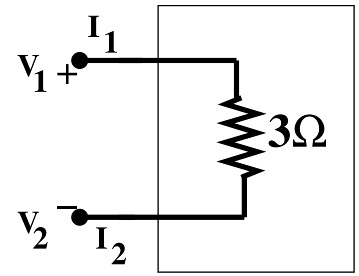
Example

Example: The module is a 3 Ohm resistor:



Example

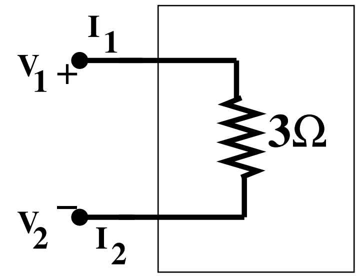
Example: The module is a 3 Ohm resistor:



The module type is 'Ohmic resistor'.

Example

<u>Example</u>: **The module is a 3 Ohm resistor:**



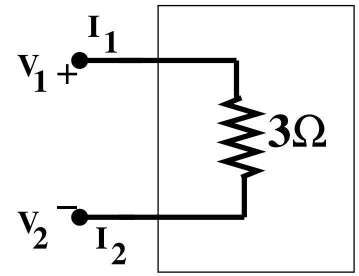
The module type is 'Ohmic resistor'.
This means that it has two electrical terminals

$\rightsquigarrow$ **terminal variables** $((V_1, I_1), (V_2, I_2))$.

The possible behaviors form a family of two-dimensional linear subspaces of $\mathbb{R}^4$.

Example

Example: The module is a 3 Ohm resistor:



The module type is 'Ohmic resistor'.
This means that it has two electrical terminals

$\rightsquigarrow$ terminal variables $((V_1, I_1), (V_2, I_2))$.

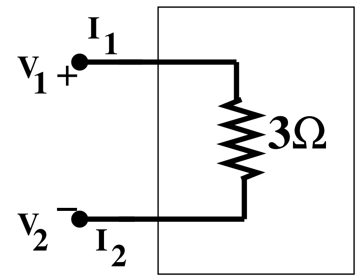
The possible behaviors form a family of two-dimensional linear subspaces of $\mathbb{R}^4$.

The resistance parametrization is the map from $R \in [0, \infty)$ into the behavioral eq'ns

$$V_1 - V_2 = R\, I_1, \quad I_1 + I_2 = 0.$$

Example

<u>Example</u>: The module is a 3 Ohm resistor:



The **module type** is **'Ohmic resistor'**.
This means that it has **two electrical terminals**

$\rightsquigarrow$ **terminal variables** $((V_1, I_1), (V_2, I_2))$.

The possible behaviors form a family of two-dimensional linear subspaces of $\mathbb{R}^4$.

The **resistance parametrization** is the map from $R \in [0, \infty)$ into the behavioral eq'ns

$$V_1 - V_2 = R\, I_1, \quad I_1 + I_2 = 0.$$

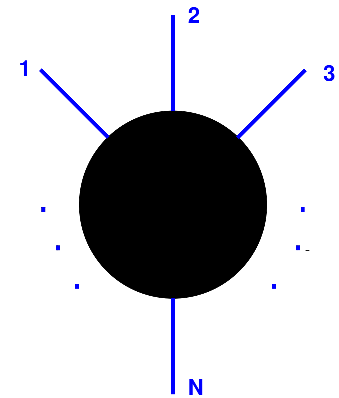
The **parameter value** equals **3**, via the parametrization $\rightsquigarrow$

$$V_1 - V_2 = 3 I_1, \quad I_1 + I_2 = 0.$$

Module type

The module type specifies an ordered set of terminals

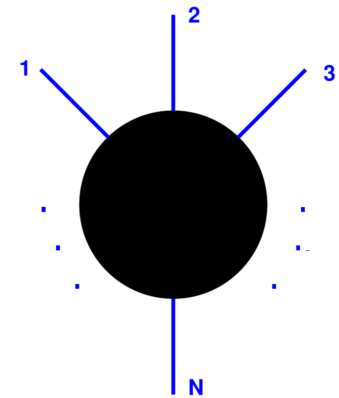
$$(t_1, t_2, \ldots, t_N).$$



Module type

The **module type** specifies an **ordered set of terminals**

$$(t_1, t_2, \ldots, t_N).$$



Together with the **terminal types,**

$\rightsquigarrow$ an ordered set of **terminal variables**

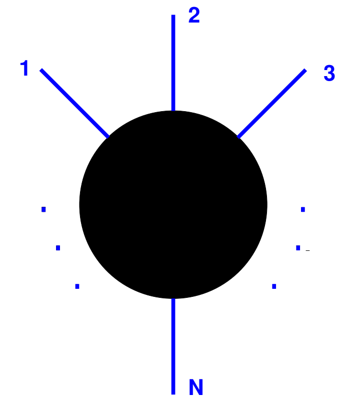
$$((w_{t_1,1}, w_{t_1,2}, \ldots), \ldots, (w_{t_N,1}, w_{t_N,2}, \ldots))$$

taking values in the product space of the terminal signal spaces.

Module type

The **module type** specifies an ordered set of terminals

$(t_1, t_2, \ldots, t_{\mathrm{N}})$.



Together with the terminal types,

$\rightsquigarrow$ an ordered set of terminal variables

$$((w_{t_1,1}, w_{t_1,2}, \ldots), \ldots, (w_{t_{\mathrm{N}},1}, w_{t_{\mathrm{N}},2}, \ldots))$$

taking values in the product space of the terminal signal spaces.

The **module type** also specifies a set $\mathbb{B}$

of possible behaviors of the terminal variables of the module.

Parametrization

We assume that the module is further specified by

a parametrization of $\mathbb{B}$,

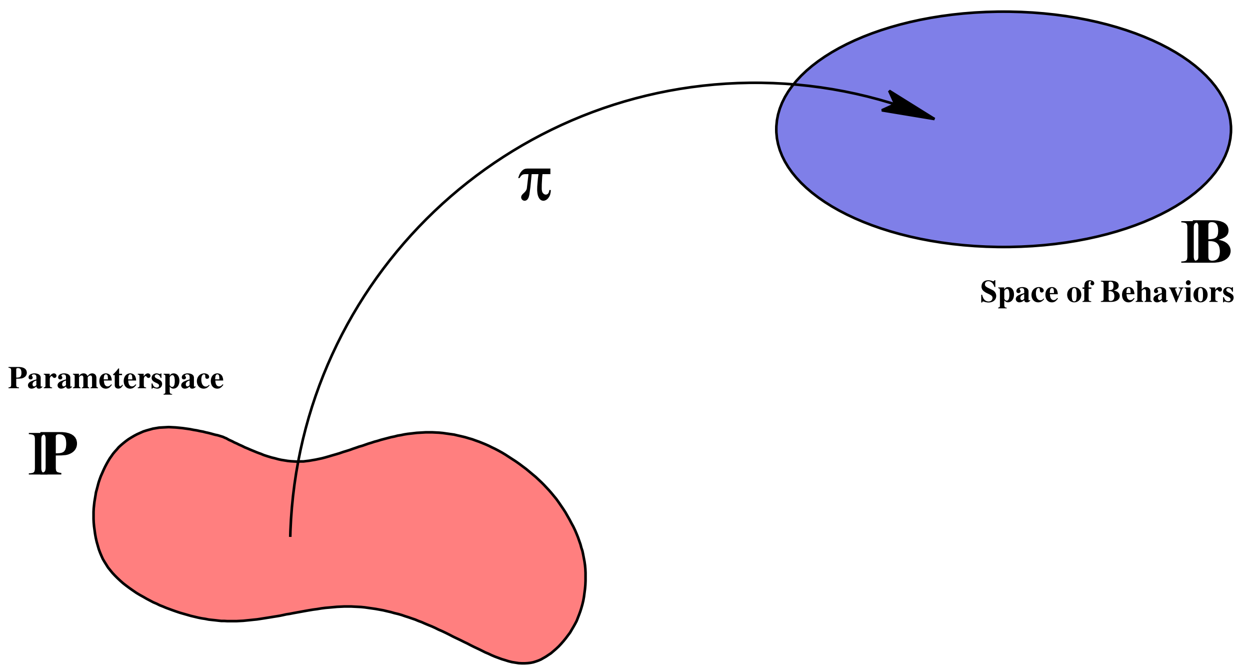
that is, a surjective map π from a parameterspace $\mathbb{P}$ into the space of behaviors $\mathbb{B}$.

Parametrization

We assume that the **module** is further specified by

a **parametrization** of $\mathbb{B}$,

that is, a surjective map π from a **parameterspace** $\mathbb{P}$ into the **space of behaviors** $\mathbb{B}$.



$\mathbb{P}$ is typically a combination of a set of integers and real numbers.

Parametrization

We assume that the module is further specified by

a parametrization of $\mathbb{B}$,

that is, a surjective map π from a parameterspace $\mathbb{P}$ into the space of behaviors $\mathbb{B}$.

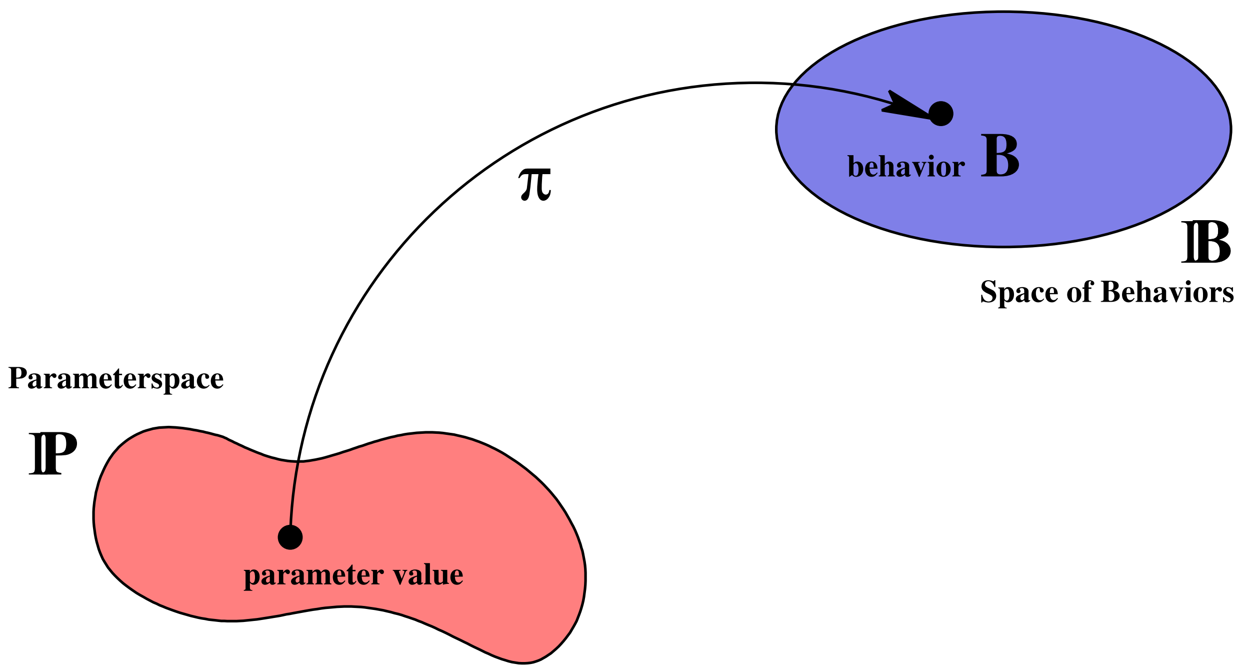
A module is further specified by giving the value of the parameters.

Parametrization

We assume that the **module** is further specified by

a **parametrization** of $\mathbb{B}$,

that is, a surjective map π from a parameterspace $\mathbb{P}$ into the space of behaviors $\mathbb{B}$.



MODULES

By specifying a module, we thus obtain the *behavior* of the variables

$$(w_1, w_2, \ldots, w_{\mathrm{n}})$$

on the *terminals of the module.*

MODULES

By specifying a module, we thus obtain the ***behavior*** of the variables

$$(w_1, w_2, \ldots, w_n)$$

on the ***terminals of the module.***

This way we obtain a dynamic model of the interaction of the module with its environment.

Examples

ELECTRICAL MODULES

Module type	Parametrization	Parameter value
2-terminal Ohmic resistor	resistance $\pi : \mathbb{R}_+ \rightarrow \cdots$	R in ohms
2- terminal Ohmic resistor	conductance $\pi : \mathbb{R}_+ \rightarrow \cdots$	G in mhos
2- terminal current driven resistor	all maps: $\mathbb{R} \longrightarrow \mathbb{R}$	$\rho : \mathbb{R} \rightarrow \mathbb{R}$
capacitor	capacitance $\pi : \mathbb{R}_+ \rightarrow \cdots$	C in farads
inductor	inductance $\pi : \mathbb{R}_+ \rightarrow \cdots$	L in henrys

Examples

Module type	Parametrization domain	Parameter value
linear impedances	$\mathbb{N}$ (number of ports) $\times\ \mathbb{R}^{n\times n}(\xi)$	$Z \in \mathbb{R}^{n\times n}[\xi]$
resistive $\triangle$	$\mathbb{R}$	R in ohms
Y with linear diff. systems	$(\mathbb{R}^2[\xi])^3$	$(R_1, R_2, R_3) \in \mathbb{R}^{1\times 2}[\xi]$
transformer	$\mathbb{R}$	$n \in \mathbb{R}$
transmission line	$(\mathbb{R}_+)^5$	L, ℓ, c, r_s, r_p
transistor		
etc.	etc.	etc.

Examples

MECHANICAL MODULES

Module type	Parametrization	Parameters
mass	$\pi : \mathbb{R}_+ \to \cdots$	m in kg
solid bar	length, mass/unit length $\pi : \mathbb{R}_+ \times \mathbb{R}_+ \to \cdots$	L, m
spring		
damper		
multi-terminal mass		geometry
flexible bar		
etc.	etc.	etc.

Examples

OTHER DOMAINS

Module type	Parametrization	Parameters
servo joint		$m_r, m_s, J_r, J_s,$ L, R, K
2 inlet tank		geometry
etc.	etc.	etc.

Examples

LINEAR SYSTEMS

Module type	Parametrization	Parameters
$\Sigma \in \mathfrak{L}^\bullet$	$\mathbb{N} \times \{\text{ker}, \text{im}, \text{etc.}\}$ $\times \mathbb{R}^{\bullet \times \bullet}[\xi]$, or $\cdots$	$(\mathtt{w}, \text{ker}, R \in \mathbb{R}^{\bullet \times \mathtt{w}}[\xi])$ $\cdots$
$\Sigma \in \mathfrak{L}^\bullet_{\text{cont}}$	$\mathbb{N} \times \{\text{im}, \ldots\}$	$(\mathtt{w}, M \in \mathbb{R}^{\mathtt{w} \times \bullet}[\xi])$, $\cdots$
$\Sigma \in \mathfrak{L}^{\text{i/o}}_{\text{cont}}$	$\mathbb{N} \times \mathbb{N} \times \{\text{tf. f'n.},$ $\ldots\} \times \mathbb{R}^{\bullet \times \bullet}(\xi), \ldots$	$\mathtt{m}, \mathtt{p}, G \in \mathbb{R}^{\mathtt{p} \times \mathtt{m}}[\xi]$ $\cdots$
$\Sigma \in \mathfrak{L}^{\text{i/s/o}}$	$\mathbb{N}^3, \ldots$	$\mathtt{m}, \mathtt{n}, \mathtt{p}, (A, B, C, D)$
etc.	etc.	etc.

INTERONNECTION ARCHITECTURE

Let

$$T = \{t_1, t_2, \ldots, t_{|T|}\}$$

be a set of terminals.

INTERONNECTION ARCHITECTURE

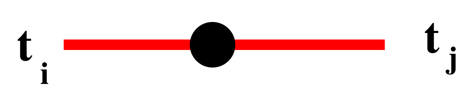
Let

$$T = \{t_1, t_2, \ldots, t_{|T|}\}$$

be a set of terminals.

The *interconnection architecture* is a set of *terminal pairs* (unordered, disjoint, and with distinct elements), denoted by $\mathbb{I}$.

If $\{t_{\mathtt{i}}, t_{\mathtt{j}}\} \in \mathbb{I}$, then we say that these terminals are connected.



INTERONNECTION ARCHITECTURE

The interconnection architecture is a set of *terminal pairs* (unordered, disjoint, and with distinct elements), denoted by $\mathbb{I}$.

If $\{t_{\mathtt{i}}, t_{\mathtt{j}}\} \in \mathbb{I}$, means that these terminals are connected.

We impose that connected terminals must be adapted.

INTERONNECTION ARCHITECTURE

The interconnection architecture is a set of *terminal pairs* (unordered, disjoint, and with distinct elements), denoted by $\mathbb{I}$.

If $\{t_{\mathtt{i}}, t_{\mathtt{j}}\} \in \mathbb{I}$, means that these terminals are connected.

We impose that connected terminals must be adapted.

In the case of physical terminals, this means that they must be of the same type (both electrical, 2-D mechanical, thermal, etc.).

In the case of logical terminals (input or output terminals), this means that if one of the connected terminals is an $\mathtt{m}$-dimensional input terminal, the other must be an $\mathtt{m}$-dimensional output terminal.

Interconnection constraints

Pairing of adapted terminals imposes an *interconnection law* .

Interconnection constraints

Pairing of adapted terminals imposes an *interconnection law* .

Example: pairing 2 electrical terminals $\rightsquigarrow$

$$V_1 = V_2, \quad I_1 + I_2 = 0.$$

Interconnection constraints

Pairing of adapted terminals imposes an *interconnection law* .

Pair of terminals	Terminal 1	Terminal 2	Interconnection law
electrical	(V_1, I_1)	(V_2, I_2)	$V_1 = V_2, I_1 + I_2 = 0$
1-D mech.	(F_1, q_1)	(F_2, q_2)	$F_1 + F_2 = 0, q_1 = q_2$
2-D mech.			
thermal	(Q_1, T_1)	(Q_2, T_2)	$Q_1 + Q_2 = 0, T_1 = T_2$
fluidic	(p_1, f_1)	(p_2, f_2)	$p_1 = p_2, f_1 + f_2 = 0$
info processing	$\mathbb{m}$-input u	$\mathbb{m}$-output y	$u = y$
etc.	etc.	etc.	etc.

MANIFEST VARIABLE ASSIGNMENT

We finally assume that the modeler assigns the variables at which the model aims. These are the manifest variables .

MANIFEST VARIABLE ASSIGNMENT

We finally assume that the modeler assigns the variables at which the model aims. These are the manifest variables .

The model unavoidably contains many other variables. These latent variables could be
either

interconnection variables,

or

latent variables used to describe the behavior of the modules.

MODEL GENERATION

So, in order to obtain a model of an interconnected system, specify:

MODEL GENERATION

So, in order to obtain a model of an interconnected system, specify:

- **A set of modules** $M_1, M_2, \cdots, M_{\mathtt{m}}$

  so, for each module,

  the type, the parametrization, and parameter value.

This yields a list of terminals $T = \{t_1, t_2, \ldots, t_{|T|}\}$

and the behavior $\mathfrak{B}_{\mathtt{i}}, \mathtt{i} = 1, \ldots, \mathtt{m}$**, for the terminal variables.**

Denote $\mathfrak{B}' = \mathfrak{B}_1 \times \cdots \times \mathfrak{B}_{\mathtt{m}}$.

MODEL GENERATION

So, in order to obtain a model of an interconnected system, specify:

- **A set of modules** $M_1, M_2, \cdots, M_{\mathtt{m}}$
  Denote $\mathfrak{B}' = \mathfrak{B}_1 \times \cdots \times \mathfrak{B}_{\mathtt{m}}$.
- **Interconnection architecture** $\mathbb{I}$ **on** $T = \{t_1, t_2, \ldots, t_{|T|}\}$
  $\rightsquigarrow$ **interconnection laws,**
  and a behavior $\mathfrak{B}''$ **for the terminal variables**

MODEL GENERATION

So, in order to obtain a model of an interconnected system, specify:

- **A set of modules** $M_1, M_2, \cdots, M_{\mathtt{m}}$
  Denote $\mathfrak{B}' = \mathfrak{B}_1 \times \cdots \times \mathfrak{B}_{\mathtt{m}}$.
- **Interconnection architecture** $\mathbb{I}$ **on** $T = \{t_1, t_2, \ldots, t_{|T|}\}$
  $\leadsto$ **interconnection laws,**
  and a behavior $\mathfrak{B}''$ **for the terminal variables**
- **The manifest variable assignment.**

MODEL GENERATION

So, in order to obtain a model of an interconnected system, specify:

- A set of modules $M_1, M_2, \cdots, M_{\mathtt{m}}$
  Denote $\mathfrak{B}' = \mathfrak{B}_1 \times \cdots \times \mathfrak{B}_{\mathtt{m}}$.
- Interconnection architecture $\mathbb{I}$ on $T = \{t_1, t_2, \ldots, t_{|T|}\}$
  $\rightsquigarrow$ interconnection laws,
  and a behavior $\mathfrak{B}''$ for the terminal variables
- The manifest variable assignment.
- The yields $\mathfrak{B}' \cap \mathfrak{B}'' =$ the full behavior
  contains both latent variables and manifest variables.

MODEL GENERATION

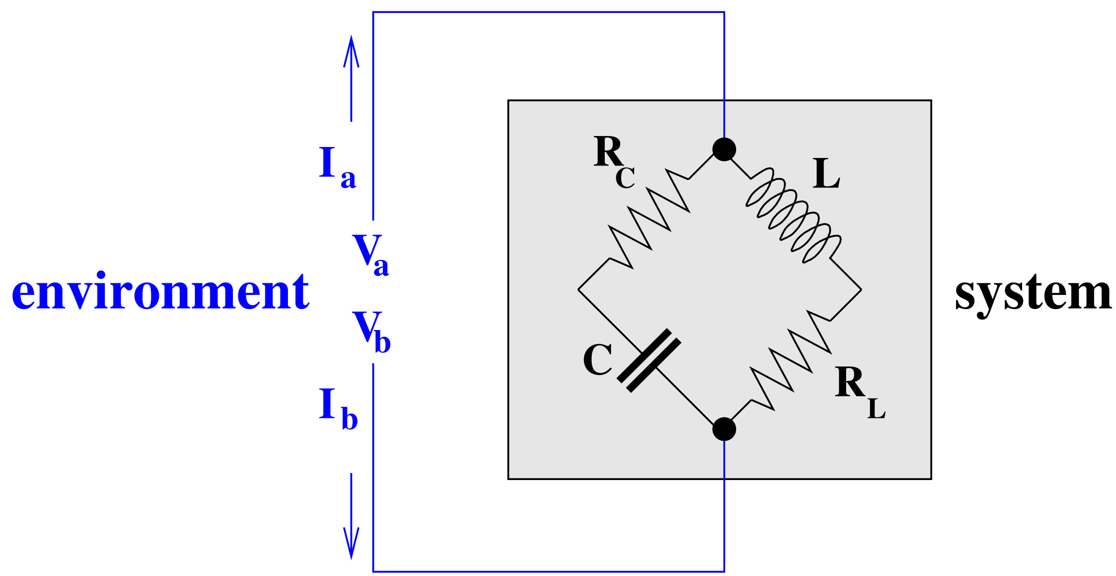
So, in order to obtain a model of an interconnected system, specify:

- A set of modules $M_1, M_2, \cdots, M_{\mathtt{m}}$
  Denote $\mathfrak{B}' = \mathfrak{B}_1 \times \cdots \times \mathfrak{B}_{\mathtt{m}}$.
- Interconnection architecture $\mathbb{I}$ on $T = \{t_1, t_2, \ldots, t_{|T|}\}$
  $\rightsquigarrow$ interconnection laws,
  and a behavior $\mathfrak{B}''$ for the terminal variables
- The manifest variable assignment.
- The yields $\mathfrak{B}' \cap \mathfrak{B}'' =$ the full behavior
  contains both latent variables and manifest variables.

- Elimination of latent variables $\longrightarrow$ the manifest behavior $\mathfrak{B}$.

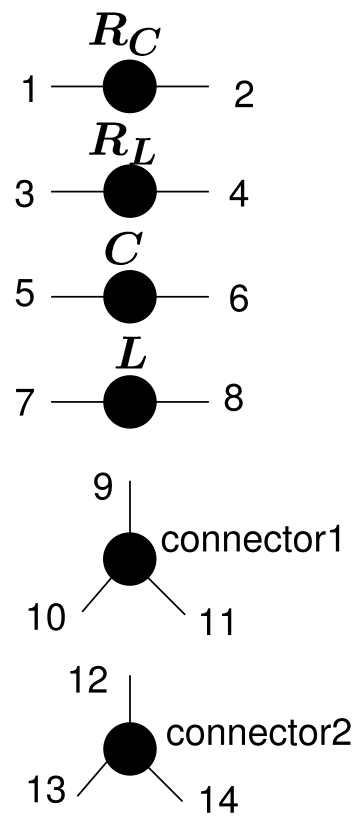
Examples

RLC circuit



RLC circuit

TEARING



RLC circuit

ZOOMING

The list of the modules & the associated terminals:

Module	Type	Terminals	Parameter
R_C	resistor	(1, 2)	R in ohms
R_L	resistor	(3, 4)	R in ohms
C	capacitor	(5, 6)	C in farad
L	inductor	(7, 8)	L in henry
connector1	3-terminal connector	(9, 10, 11)	
connector2	3-terminal connector	(12, 13, 14)	

The interconnection architecture:

Pairing
$\{10, 1\}$
$\{11, 7\}$
$\{2, 5\}$
$\{8, 3\}$
$\{6, 13\}$
$\{4, 14\}$

RLC circuit

Manifest variable assignment:

the variables

$$V_9, I_9, V_{12}, I_{12}$$

on the external terminals $\{9, 12\}$, i.e,

$$V_a = V_9, I_a = I_9, V_b = V_{12}, I_b = I_{12}.$$

RLC circuit

Manifest variable assignment:

the variables

$$V_9, I_9, V_{12}, I_{12}$$

on the external terminals {9, 12}, i.e,

$$V_a = V_9, I_a = I_9, V_b = V_{12}, I_b = I_{12}.$$

The internal terminals are

$$\{1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 13, 14\}$$

The variables on these terminals are latent variables.

RLC circuit

Equations for the full behavior:

Modules	Constitutive equations	
R_C	$I_1 + I_2 = 0$	$V_1 - V_2 = R_C I_1$
R_L	$I_7 + I_8 = 0$	$V_7 - V_8 = R_L I_7$
C	$I_5 + I_6 = 0$	$C \frac{d}{dt}(V_5 - V_6) = I_5$
L	$I_7 + I_8 = 0$	$V_7 - V_8 = L \frac{d}{dt} I_7$
connector1	$I_9 + I_{10} + I_{11} = 0$	$V_9 = V_{10} = V_{11}$
connector2	$I_{12} + I_{13} + I_{14} = 0$	$V_{12} = V_{13} = V_{14}$

RLC circuit

Interconnection pair	Interconnection equations	
$\{10, 1\}$	$V_{10} = V_1$	$I_{10} + I_1 = 0$
$\{11, 7\}$	$V_{11} = V_7$	$I_{11} + I_7 = 0$
$\{2, 5\}$	$V_2 = V_5$	$I_2 + I_5 = 0$
$\{8, 3\}$	$V_8 = V_3$	$I_8 + I_3 = 0$
$\{6, 13\}$	$V_6 = V_{13}$	$I_6 + I_{13} = 0$
$\{4, 14\}$	$V_4 = V_{14}$	$I_4 + I_{14} = 0$

RLC circuit

All these eq'ns combined define a latent variable system in the manifest variables

$$w = (V_a, I_a, V_b, I_b)$$

with latent variables

$$\ell = (V_1, I_1, V_2, I_2, V_3, I_3, V_4, I_4, V_5, I_5, V_6, I_6, V_7, I_7, \\ V_8, I_8, V_{10}, I_{10}, V_{11}, I_{11}, V_{13}, I_{13}, V_{14}, I_{14}).$$

RLC circuit

All these eq'ns combined define a latent variable system in the **manifest variables**

$$w = (V_a, I_a, V_b, I_b)$$

with **latent variables**

$$\ell = (V_1, I_1, V_2, I_2, V_3, I_3, V_4, I_4, V_5, I_5, V_6, I_6, V_7, I_7, \\ V_8, I_8, V_{10}, I_{10}, V_{11}, I_{11}, V_{13}, I_{13}, V_{14}, I_{14}).$$

The manifest behavior $\mathfrak{B}$ is given by

$$\mathfrak{B} = \{(V_a, I_a, V_b, I_b) : \mathbb{R} \to \mathbb{R}^4 \mid \exists\ \ell : \mathbb{R} \to \mathbb{R}^{24} \ldots\}$$

RLC circuit

Elimination: for example, using Gröbner bases.

RLC circuit

Elimination: for example, using Gröbner bases.

Case 1: $CR_C \neq \frac{L}{R_L}$.

$$\left(\frac{R_C}{R_L} + (1 + \frac{R_C}{R_L})CR_C\frac{d}{dt} + CR_C\frac{L}{R_L}\frac{d^2}{dt^2}\right)(V_a - V_b)$$
$$= (1 + CR_C\frac{d}{dt})(1 + \frac{L}{R_L}\frac{d}{dt})R_C I_a.$$

$$I_a + I_b = 0$$

RLC circuit

Elimination: for example, using Gröbner bases.

Case 1: $CR_C \neq \frac{L}{R_L}$.

$$\left(\frac{R_C}{R_L} + \left(1+\frac{R_C}{R_L}\right)CR_C\frac{d}{dt} + CR_C\frac{L}{R_L}\frac{d^2}{dt^2}\right)(V_a - V_b) = \left(1+CR_C\frac{d}{dt}\right)\left(1+\frac{L}{R_L}\frac{d}{dt}\right)R_C I_a.$$

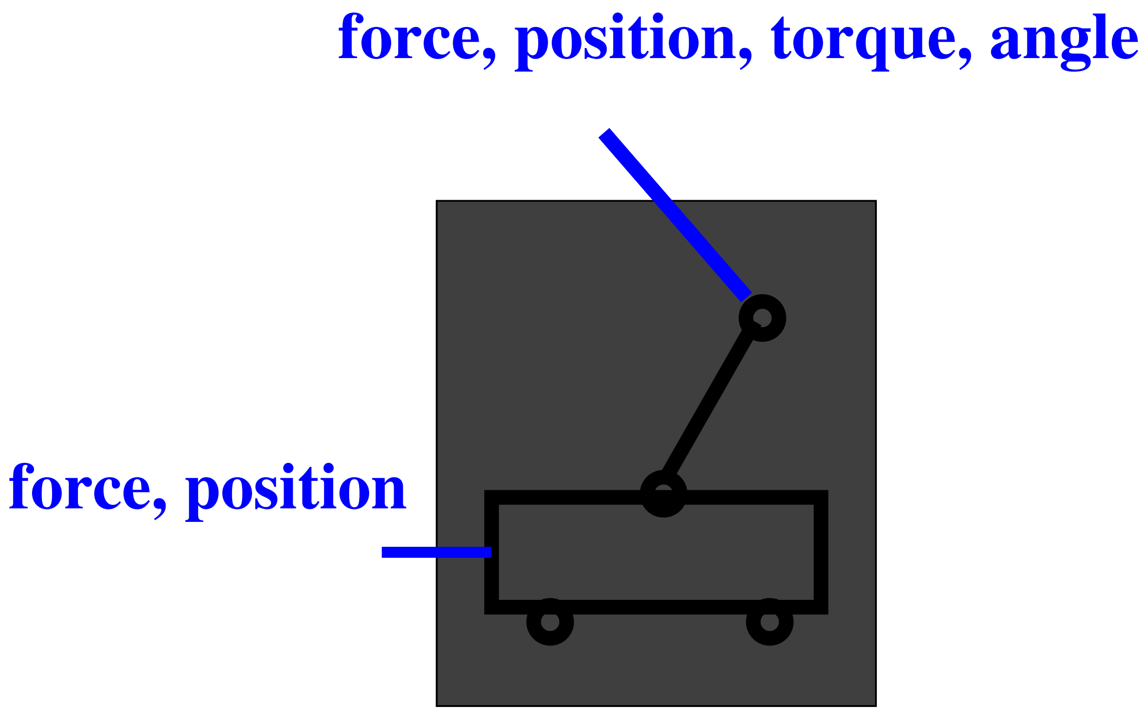
$$I_a + I_b = 0$$

Case 2: $CR_C = \frac{L}{R_L}$.

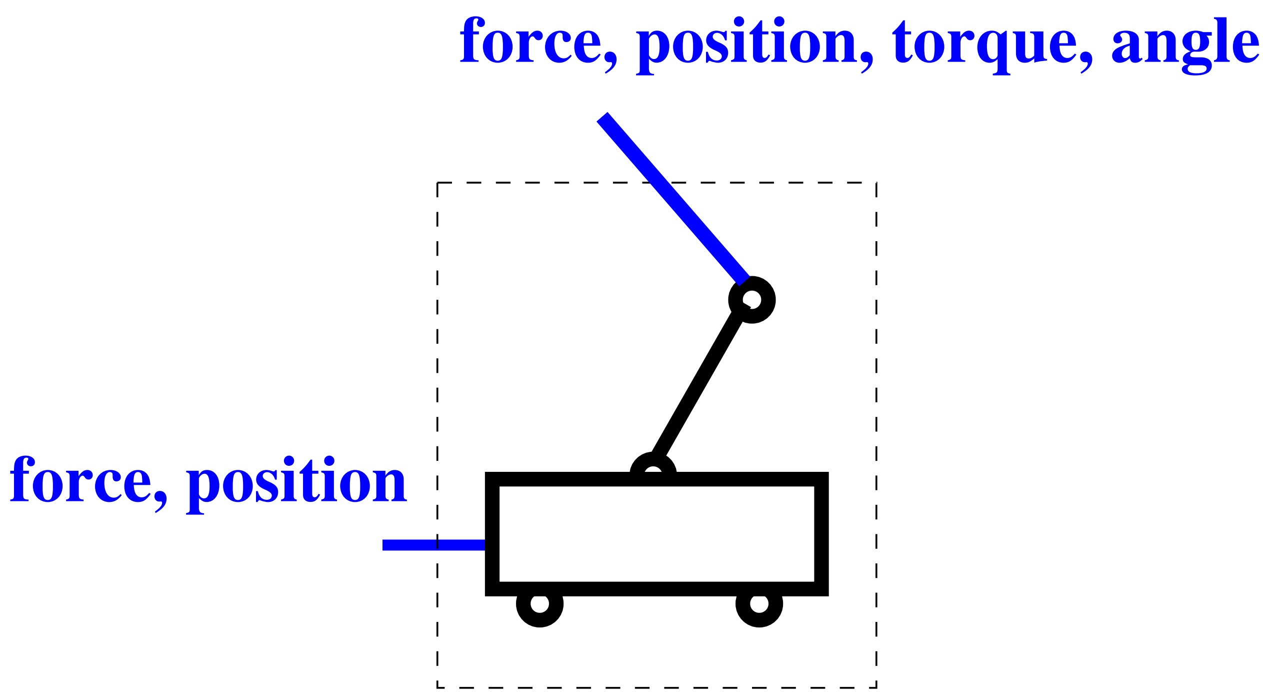
$$\left(\frac{R_C}{R_L} + CR_C\frac{d}{dt}\right)(V_a - V_b = \left(1+CR_C\frac{d}{dt}\right)R_C I_a$$

$$I_a + I_b = 0$$

CART

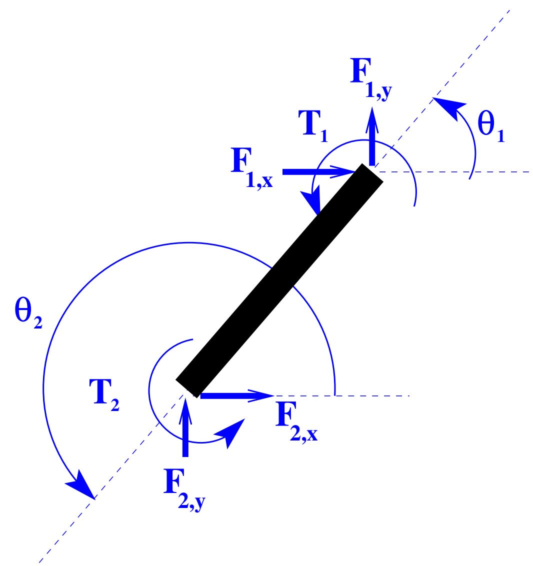


CART



Required modules: **Solid bars, cart, servo's.**

CART



Solid bar

Terminals: 2 mechanical 2-D terminals.

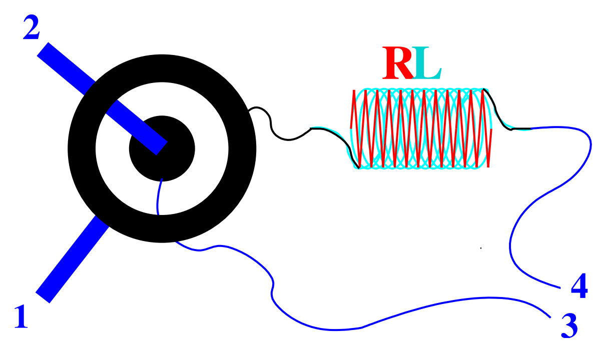
Parameters:

$L \in \mathbb{R}_+$ (length), $m \in \mathbb{R}_+$ (mass per unit length).

Behavioral equations:

$$
\begin{aligned}
mL\tfrac{d^2}{dt^2}x_c &= F_{x_1} + F_{x_2},\\
mL\tfrac{d^2}{dt^2}y_c &= F_{y_1} + F_{y_2} - mLg,\\
m\tfrac{L^3}{12}\tfrac{d^2}{dt^2}\theta_c &= T_1 + T_2 - \tfrac{L}{2}F_{x_1}\sin(\theta_1)\\
&\quad + \tfrac{L}{2}F_{y_1}\cos(\theta_1) - \tfrac{L}{2}F_{x_2}\sin(\theta_2) + \tfrac{L}{2}F_{y_2}\cos(\theta_2),\\
\theta_1 &= \theta_c, \theta_2 = \theta_1 + \pi,\\
x_1 &= x_c + \tfrac{L}{2}\cos(\theta_c), x_2 = x_c - \tfrac{L}{2}\cos(\theta_c),\\
y_1 &= y_c + \tfrac{L}{2}\sin(\theta_c), y_2 = y_c - \tfrac{L}{2}\sin(\theta_c).
\end{aligned}
$$

Note: Contains latent variables x_c, θ_c.



Hinge with servo

Terminals: 2 mechanical 2-D terminals, 2 electrical.

Parameters:

rotor mass m_r, the stator mass m_s, the rotor inertia J_r, the stator inertia J_s, the inductance L, the resistance R of the motor circuit, the motor torque constant K.

CART

Behavioral equations:

$$(m_r + m_s)\frac{d^2}{dt^2}x_1 = F_{x_1} + F_{x_2}$$

$$(m_r + m_s)\frac{d^2}{dt^2}y_1 = F_{x_1} + F_{x_2}$$

$$J_r\frac{d^2}{dt^2}\theta_1 = T_1 + T_m$$

$$J_s\frac{d^2}{dt^2}\theta_2 = T_2 - T_m$$

$$V_3 - V_4 = L\frac{d}{dt}I_3 + RI_3 + K\frac{d}{dt}(\theta_1 - \theta_2)$$

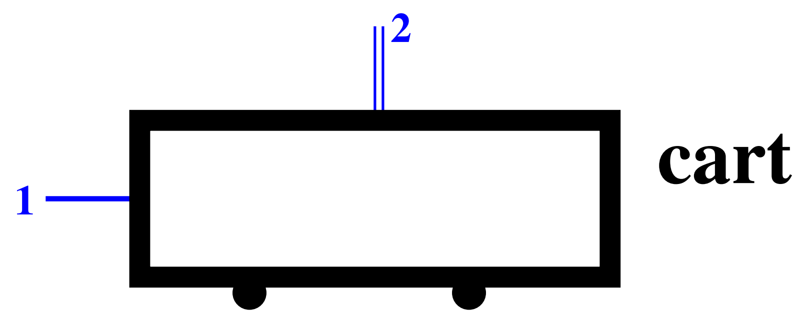
$$KI_3 = T_m, I_3 = -I_4$$

$$x_1 = x_2, y_1 = y_2$$

Terminal variables: $(x_1, y_1, \theta_1, F_{x_1}, F_{y_1}, T_1,$
$x_2, y_2, \theta_2, F_{x_2}, F_{y_2}, T_2, V_3, I_4, V_4, I_4).$

The motor torque T_m is a latent variable.

CART



Terminals: 1 mechanical 1-D terminal, 1 mechanical 2-D terminal.

Parameters: mass M.

CART

Behavioral equations:

$$M\frac{d^2}{dt^2}x_1 = F_1 + F_{x_2}$$

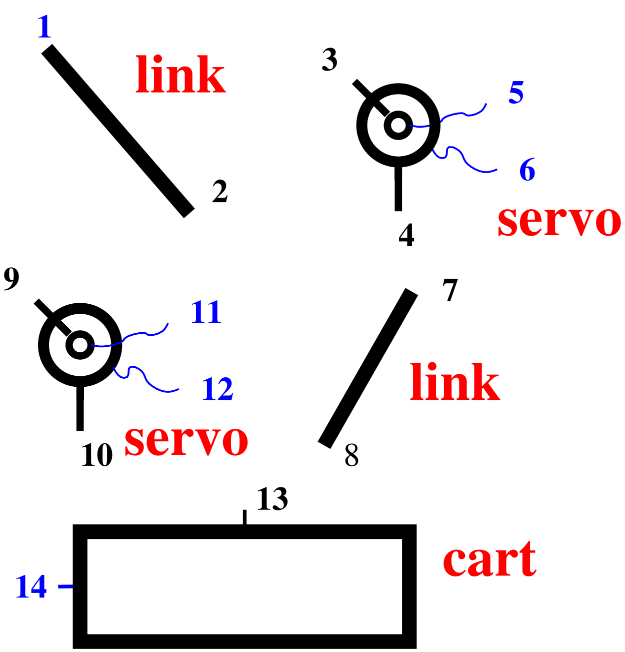
$$x_2 = x,$$

$$y_2 = 0,$$

$$\theta_2 = \pi/2$$

CART

TEARING



CART

ZOOMING

The list of the modules & the associated terminals:

Module	Type	Terminals	Parameter
Link 1	bar	(7,8)	L_1, m_1
Link 2	bar	(1,2)	L_2, m_2
Cart	cart	(13,14)	M
Hinge 1	servo	(9,10,11,12)	$m_{r_1}, m_{s_1}, J_{r_1}, J_{r_1}, L_1, R_1, K_1$
Hinge 2	servo	(3,4,5,6)	$m_{r_2}, m_{s_2}, J_{r_2}, J_{r_2}, L_2, R_2, K_2$

CART

The interconnection architecture:

Pairing
$\{2, 3\}$
$\{4, 7\}$
$\{8, 9\}$
$\{10, 13\}$

CART

The interconnection architecture:

Pairing
$\{2, 3\}$
$\{4, 7\}$
$\{8, 9\}$
$\{10, 13\}$

Manifest variable assignment:

the variables on the external terminals $\{1, 5, 6, 11, 12, 14\}$.

All other variables are latent variables.

CART

Equations for the full behavior:

$$
\begin{aligned}
m_1 L_1 \frac{d^2}{dt^2} x_{c_1} &= F_{x_1} + F_{x_2}, \\
m_1 L_1 \frac{d^2}{dt^2} y_{c_1} &= F_{y_1} + F_{y_2} - m_1 L_1 g, \\
m_1 \frac{L_1^3}{12} \frac{d^2}{dt^2} \theta_{c_1} &= T_1 + T_2 - \\
& \quad \frac{L_1}{2} F_{x_1} \sin(\theta_1) + \frac{L_1}{2} F_{y_1} \cos(\theta_1) - \frac{L_1}{2} F_{x_2} \sin(\theta_2) + \frac{L_1}{2} F_{y_2} \cos(\theta_2), \\
\theta_1 &= \theta_{c_1}, \\
\theta_2 &= \theta_1 + \pi, \\
x_1 &= x_{c_1} + \frac{L_1}{2} \cos(\theta_{c_1}), \\
x_2 &= x_{c_1} - \frac{L_1}{2} \cos(\theta_{c_1}), \\
y_1 &= y_{c_1} + \frac{L_1}{2} \sin(\theta_{c_1}), \\
y_2 &= y_{c_1} - \frac{L_1}{2} \sin(\theta_{c_1}),
\end{aligned}
$$

CART

$$
\begin{aligned}
&m_2 L_2 \tfrac{d^2}{dt^2} x_{c_2} = F_{x_7} + F_{x_8}, \\
&m_2 L_2 \tfrac{d^2}{dt^2} y_{c_2} = F_{y_7} + F_{y_8} - m_2 L_2 g, \\
&m_2 \tfrac{L_2^3}{12} \tfrac{d^2}{dt^2} \theta_{c_2} = T_7 + T_8 - \tfrac{L_2}{2} F_{x_7} \sin(\theta_7) + \tfrac{L_2}{2} F_{y_7} \cos(\theta_7), \\
&\qquad - \tfrac{L_2}{2} F_{x_8} \sin(\theta_8) + \tfrac{L_2}{2} F_{y_8} \cos(\theta_8), \\
&\theta_7 = \theta_{c_2}, \\
&\theta_8 = \theta_7 + \pi, \\
&x_7 = x_{c_2} + \tfrac{L_1}{2} \cos(\theta_{c_2}), \\
&x_8 = x_{c_2} - \tfrac{L_1}{2} \cos(\theta_{c_2}), \\
&y_7 = y_{c_2} + \tfrac{L_1}{2} \sin(\theta_{c_2}), \\
&y_8 = y_{c_2} - \tfrac{L_1}{2} \sin(\theta_{c_2}),
\end{aligned}
$$

CART

$$M\frac{d^2}{dt^2}x_{14} = F_{14} + F_{x_{14}}$$

$$x_{14} = x_{13},$$

$$y_{13} = 0,$$

$$\theta_{13} = \pi/2,$$

CART

$$(m_{r_1} + m_{s_1})\frac{d^2}{dt^2}x_3 = F_{x_3} + F_{x_4},$$
$$(m_{r_1} + m_{s_1})\frac{d^2}{dt^2}y_3 = F_{y_3} + F_{y_4},$$
$$J_{r_1}\frac{d^2}{dt^2}\theta_3 = T_3 + T_m,$$
$$J_{s_1}\frac{d^2}{dt^2}\theta_4 = T_4 - T_m,$$
$$V_5 - V_6 = L_1\frac{d}{dt}I_5 + R_1 I_5 + K\frac{d}{dt}(\theta_3 - \theta_4),$$
$$K_1 I_5 = T_{m_1},$$
$$x_3 = x_4, y_3 = y_4,$$
$$I_5 = -I_6,$$

CART

$$(m_{r_2} + m_{s_2})\frac{d^2}{dt^2}x_9 = F_{x_9} + F_{x_{10}},$$

$$(m_{r_2} + m_{s_2})\frac{d^2}{dt^2}y_9 = F_{y_9} + F_{y_{10}},$$

$$J_{r_2}\frac{d^2}{dt^2}\theta_9 = T_9 + T_m,$$

$$J_{s_2}\frac{d^2}{dt^2}\theta_{10} = T_{10} - T_m,$$

$$V_{11} - V_{12} = L_2\frac{d}{dt}I_{11} + R_2 I_{11} + K\frac{d}{dt}(\theta_9 - \theta_{10}),$$

$$K_2 I_{11} = T_{m_2},$$

$$x_{10} = x_{11}, y_{10} = y_{11},$$

$$I_{11} = -I_{12},$$

CART

$$F_{x_2} + F_{x_3} = 0,\ F_{y_2} + F_{y_3} = 0,\ x_2 = x_3,\ y_2 = y_3,$$

$$\theta_2 = \theta_3 + \pi,\ T_2 + T_3 = 0,$$

$$F_{x_4} + F_{x_7} = 0,\ F_{y_4} + F_{y_7} = 0,\ x_4 = x_7,\ y_4 = y_7,$$

$$\theta_4 = \theta_7 + \pi,\ T_4 + T_7 = 0,$$

$$F_{x_8} + F_{x_9} = 0,\ F_{y_8} + F_{y_9} = 0,\ x_8 = x_9,\ y_8 = y_9,$$

$$\theta_8 = \theta_9 + \pi,\ T_8 + T_9 = 0,$$

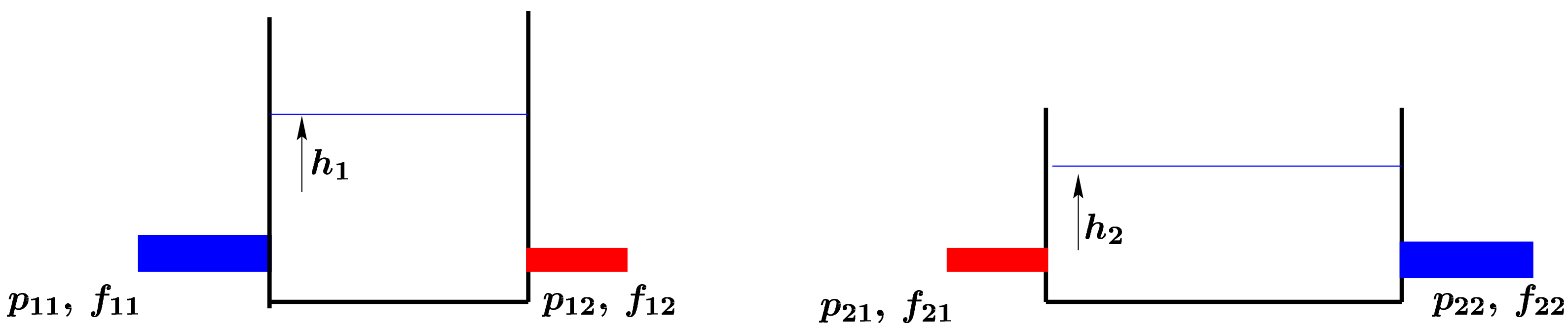
$$F_{x_{10}} + F_{x_{13}} = 0,\ F_{x_{10}} + F_{x_{13}} = 0,$$

$$x_{10} = x_{13},\ y_{10} = y_{13}.$$

$$\theta_{10} = \theta_{13} + \pi,\ T_{10} + T_{13} = 0.$$

INPUT - to - OUTPUT CONNECTIONS

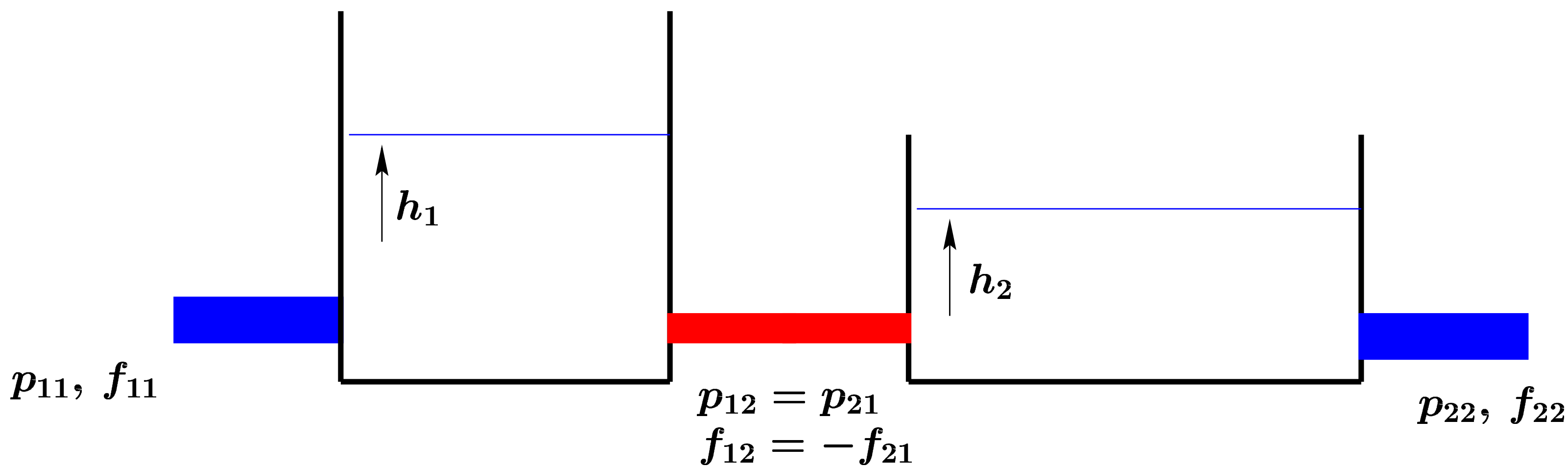
The inappropriateness of input - to - output connections is best illustrated by the following simple physical example:



Logical choice of inputs: the pressures $p_{11}, p_{12}, p_{21}, p_{22}$, and of the outputs $f_{11}, f_{12}, f_{21}, f_{22}$.

In any case, the choice should be 'symmetric'.

INPUT - to - OUTPUT CONNECTIONS



Interconnection constraints:

$$p_{12} = p_{21}, \quad f_{12} = f_{21}.$$

Equates two ‘inputs’ and two ‘outputs’.

LINEAR RLCT CIRCUITS

BUILDING BLOCKS

Module Types:

Resistors, Capacitors, Inductors, Transformers, Connectors.

All terminals are of the same type: electrical

There are 2 variables associated with each terminal, (V, I),

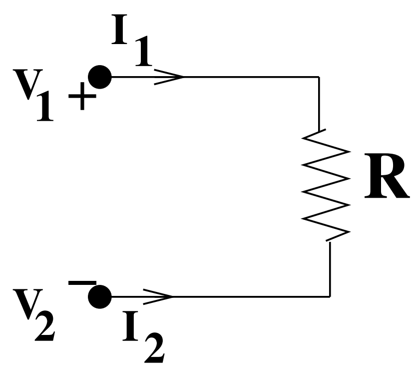
V the *potential*,

I the *current* (counted > 0 when it flows *into* the module).

$\rightsquigarrow$ terminal signal space $\mathbb{R}^2$.

LINEAR RLCT CIRCUITS

SPECIFICATION of the BEHAVIOR of the MODULES



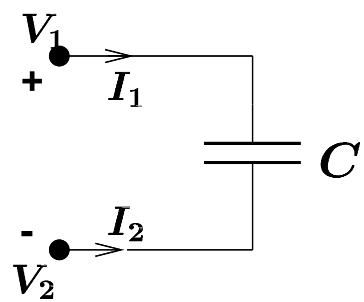
Resistor: 2-terminal module.

Parameter: R (resistance in ohms, say).

Device laws:

$$V_1 - V_2 = R\, I_1 \; ; \quad I_1 + I_2 = 0.$$

LINEAR RLCT CIRCUITS



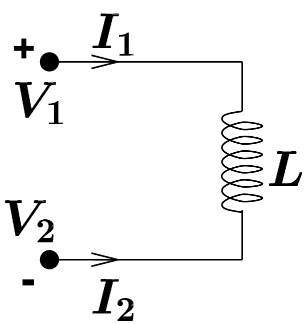
Capacitor: 2-terminal module.

Parameter: C (capacitance in farads, say).

Device laws:

$$C\,\frac{d}{dt}(V_1 - V_2) = I_1\,; \quad I_1 + I_2 = 0.$$

LINEAR RLCT CIRCUITS



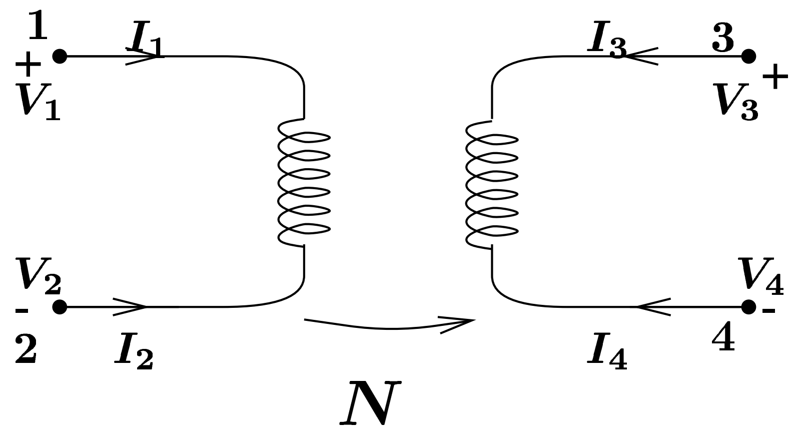
Inductor: 2-terminal module.

Parameter: L (inductance in henrys, say).

Device laws:

$$L\,\frac{d}{dt}I_1 = V_1 - V_2\,; \quad I_1 + I_2 = 0.$$

LINEAR RLCT CIRCUITS



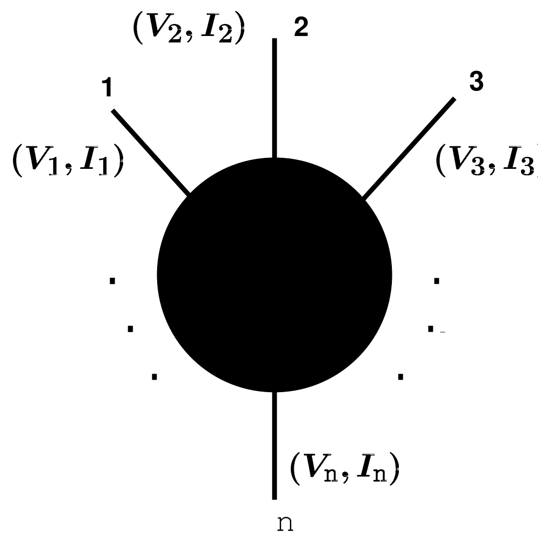
Transformer: **4-terminal module; terminals (1,2): primary;**
terminals (3,4): secondary.

Parameter: N **(the turns ratio,** $\in (0, \infty)$ **).**

Device laws:

$$V_3 - V_4 = N(V_1 - V_2)\,; \qquad I_1 = -NI_3\,;$$
$$I_1 + I_2 = 0\,; \qquad I_3 + I_4 = 0.$$

LINEAR RLCT CIRCUITS



Connector: many-terminal module.

Parameter: n (number of terminals, an integer).

Device laws:

$$V_1 = V_2 = \cdots = V_n\,; \quad I_1 + I_2 + \cdots + I_n = 0.$$

LINEAR RLCT CIRCUITS

MODULES and TERMINAL ASSIGNMENT

Modules

Resistors	$r_1, r_2, \ldots, r_{\mathrm{n}_r}$,	**parameters** $R_1, R_2, \ldots, R_{\mathrm{n}_r}$;
Capacitors	$c_1, c_2, \ldots, c_{\mathrm{n}_c}$,	**parameters** $C_1, C_2, \ldots, C_{\mathrm{n}_c}$;
Inductors	$\ell_1, \ell_2, \ldots, \ell_{\mathrm{n}_\ell}$,	**parameters** $L_1, L_2, \ldots, L_{\mathrm{n}_\ell}$;
Transformers	$T_1, T_2, \ldots, T_{\mathrm{n}_T}$,	**parameters** $N_1, N_2, \ldots, N_{\mathrm{n}_T}$;
Connectors	$\mathrm{k}_1, \mathrm{k}_2, \ldots, \mathrm{k}_{\mathrm{n}_\mathrm{k}}$,	**parameters** $\mathrm{n}_1, \mathrm{n}_2, \ldots, \mathrm{n}_{\mathrm{n}_\mathrm{k}}$.

This yields the set of terminals

$$\mathbb{T} = \{1, 2, \ldots, 2(\mathrm{n}_r + \mathrm{n}_c + \mathrm{n}_\ell) + 4\mathrm{n}_T + \mathrm{n}_1 + \mathrm{n}_2 + \ldots + \mathrm{n}_{\mathrm{n}_\mathrm{k}}\}.$$

INTERCONNECTION ARCHITECTURE

Interconnection architecture :

$\mathbb{I}$ $=$ **a set of disjoint (unordered) pairs of different elements (i.e., doubletons) from** $\mathbb{T}$**.**

LINEAR RLCT CIRCUITS

MANIFEST VARIABLE ASSIGNMENT

External terminals = $\mathbb{E} := \mathbb{T} - \cup_{\mathbb{I}} \{a, b\}$.

Manifest variables = external terminal voltages and currents $= \Pi_{k \in \mathbb{I}} (V_k, I_k)$. Denote the manifest variables by $\Pi_{k \in \mathbb{I}} (V_k, I_k)$ as $(V, I) \in \mathbb{R}^{2\mathbb{E}}$.

Manifest behavior: $\subseteq (\mathbb{R}^{2\mathbb{E}})^{\mathbb{R}}$.
Denote further the full behavior (the behavior of all the terminal voltages and currents) by $\mathfrak{B}_{\mathbb{T}} \subseteq (\mathbb{R}^{2\mathbb{T}})^{\mathbb{T}}$.

LINEAR RLCT CIRCUITS

FULL BEHAVIORAL EQUATIONS

1. Module Laws:

1.1 **Resistors:** for each resistor r_n, terminals $(t_1^{r_n}, t_2^{r_n})$,

$$V_{t_1^{r_n}} - V_{t_2^{r_n}} = R_n\, I_{t_1^{r_n}}\ ; \quad I_{t_1^{r_n}} + I_{t_2^{r_n}} = 0.$$

1.2 **Capacitors:** for each capacitor c_n, terminals $(t_1^{c_n}, t_2^{c_n})$,

$$\frac{d}{dt} C_n\, (V_{t_1^{c_n}} - V_{t_2^{c_n}}) = I_{t_1^{c_n}}\ ; \quad I_{t_1^{c_n}} + I_{t_2^{c_n}} = 0.$$

1.3 **Inductors:** for each inductor ℓ_n, terminals $(t_1^{\ell_n}, t_2^{\ell_n})$,

$$\frac{d}{dt} L_n\, I_{t_1^{\ell_n}} - V_{t_2^{\ell_n}}\ ; \quad I_{t_1^{\ell_n}} + I_{t_2^{\ell_n}} = 0.$$

1.4 **Transformers:** for each transformer T_n, terminals $(t_1^{T_n}, t_2^{T_n}, t_3^{T_n}, t_4^{T_n})$,

$$V_{t_1^{T_n}} - V_{t_2^{T_n}} = N_n (V_{t_3^{T_n}} - V_{t_4^{T_n}})\ ; \qquad I_{t_3^{T_n}} = -N_n I_{t_1^{T_n}}$$

$$I_{t_1^{T_n}} + I_{t_2^{T_n}} = 0\ ; \qquad I_{t_3^{T_n}} + I_{t_4^{T_n}} = 0.$$

1.5 **Connectors:** for each connector k_n, terminals $(t_1^{k_n}, \ldots, t_{n_{k_n}}^{k_n})$,

$$V_{t_1^{k_n}} = \cdots = V_{t_{n_{k_n}}^{k_n}}\ ; \quad I_{t_1^{k_n}} + \cdots + I_{t_{n_{k_n}}^{k_n}}\ .$$

LINEAR RLCT CIRCUITS

2. **Interconnection Laws:**

For each 'connected' terminal pair $\{a, b\} \in \mathbb{I}$:

$$V_a = V_b; \quad I_a + I_b = 0.$$

Solution of behavioral equations $\rightsquigarrow \mathfrak{B}_{\mathbb{T}}$.

After elimination of internal variables $\rightsquigarrow \mathfrak{B}_{\mathbb{E}}$.

LINEAR RLCT CIRCUITS

PROPERTIES of $\mathfrak{B}_{\mathbb{E}}$

When is $\mathfrak{B}_{\mathbb{E}} \subseteq (\mathbb{R}^{2\mathbb{E}})^{\mathbb{R}}$
the external terminal behavior of a circuit
containing a finite number of positive
R's, L's, C'c, T's, and connectors?

It is possible to derive necessary & sufficient conditions!

LINEAR RLCT CIRCUITS

1. $\mathfrak{B}_{\mathbb{E}} \in \mathfrak{L}^{2\mathbb{E}}.$

2. **KVL:**

$((V, I) \in \mathfrak{B}_{\mathbb{E}})$ and $(\alpha \in \mathfrak{C}^{\infty}(\mathbb{R}, \mathbb{R})) \Rightarrow ((V + \alpha e) \in \mathfrak{B}_{\mathbb{E}})$
with

$$e = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

3. **KCL:**

$$((V, I) \in \mathfrak{B}_{\mathbb{E}}) \Rightarrow (e^{\top} I = 0)$$

LINEAR RLCT CIRCUITS

4. **Input cardinality:** $\mathtt{m}(\mathfrak{B}_{\mathbb{E}}) = \mathbb{E}$

5. **Hybridicity:**

   There exists an input/output choice such that the input variables $(u_1, u_2, \ldots, u_{\mathbb{E}})$ and output variables $(y_1, y_2, \cdots, y_{\mathbb{E}})$ pair as follows:
   $$\{u_{\mathtt{i}}, y_{\mathtt{i}}\} = \{V_{\mathtt{i}}, I_{\mathtt{i}}\}$$

   Each terminal is either current controlled or voltage controlled.

6. **Passivity:**

Assume for simplicity $\mathfrak{B}_{\mathbb{E}} \in \mathbb{L}^{2\mathbb{E}}_{\text{controllable}}$. There holds

$$\boxed{\int_0^{+\infty} V^\top(t) I(t)\, dt \geq 0}$$

for all $(V, I) \in \mathfrak{B}_{\mathbb{E}}$ of compact support.

This states that the net electrical energy flows <u>into</u> the circuit.

7. Reciprocity:

Assume again for simplicity $\mathfrak{B}_{\mathbb{E}} \in \mathbb{L}^{2\mathbb{E}}_{\text{controllable}}$. There holds

$$\int_{-\infty}^{+\infty} V_1^\top(t) I_2(-t)\, dt = \int_{-\infty}^{+\infty} I_1^\top(t) V_2(-t)\, dt$$

for all $(V_1, I_1), (V_2, I_2) \in \mathfrak{B}_{\mathbb{E}}$ of compact support.

Equivalently: $\mathfrak{B}_{\mathbb{E}} =$ rev$(\mathfrak{B}_{\mathbb{E}}^{\perp_\Sigma})$,
where rev denotes time-reversal, and $\Sigma = \left[\begin{smallmatrix} O & I \\ -I & O \end{smallmatrix}\right]$.

This curious properties may be translated into:

The influence of terminal `i` on terminal `j` is equal to the influence of terminal `j` on terminal `i`.

Proof of necessity:

Show that the modules satisfy properties (1) to (7).
Show that these properties remain valid after one interconnection.
The difficult part here is (4).

Proof of necessity:

'Synthesis'.

TERMINALS or PORTS?

Note that (for instance for electrical circuits) we have used the terminal description. It is simply more appropriate and more general than the port description (even when using only ‘port’ devices).

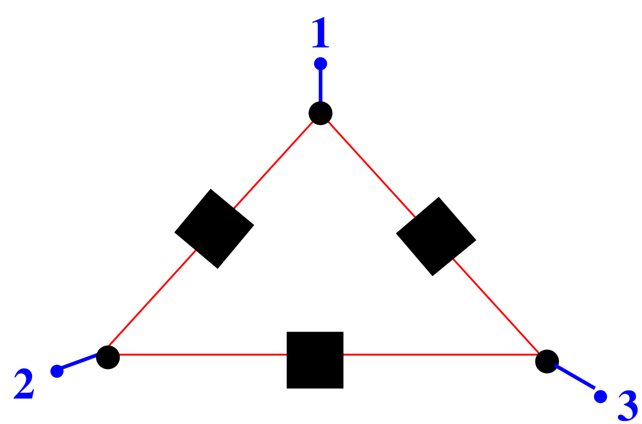
The port description is not ‘closed under interconnection’.

TERMINALS or PORTS?

Note that (for instance for electrical circuits) we have used the **terminal description**. It is simply more appropriate and more general than the **port description** (even when using only 'port' devices).

The port description is not **'closed under interconnection'**.

Example:



TERMINALS or PORTS?

Note that (for instance for electrical circuits) we have used the terminal description. It is simply more appropriate and more general than the port description (even when using only ‘port’ devices).

The port description is not ‘closed under interconnection’.

However, port descriptions are more parsimomious in the choice of variables (it halves their number). It is important to incorporate this parsimony.

RECAPITULATION

- **Modelling interconnected systems** $\cong$ **Interplay of**

RECAPITULATION

- **Modelling interconnected systems** $\cong$ **Interplay of**
  - **modules and their behavior**
  - **terminals and their type**
  - **the interconnection architecture**
  - **interconnection laws**
  - **manifest variable assignment**

RECAPITULATION

- **Modelling interconnected systems** $\cong$ **Interplay of**
  - **modules and their behavior**
  - **terminals and their type**
  - **the interconnection architecture**
  - **interconnection laws**
  - **manifest variable assignment**
- **Adapted to computer assisted modelling**
- **Hierarchical, reusable, extendable**

RECAPITULATION

- **Modelling interconnected systems** $\cong$ **Interplay of**
  - **modules and their behavior**
  - **terminals and their type**
  - **the interconnection architecture**
  - **interconnection laws**
  - **manifest variable assignment**
- **Adapted to computer assisted modelling**
- **Hierarchical, reusable, extendable**
- **Many latent variables, many equations (many static relations, i.e., algebraic equations). Far distance from i/o, i/s/o, tf. f'ns.**
- **Importance of elimination algorithms**

CONCLUSION

$*$ **for physical systems ($\Rightarrow\Leftarrow$ signal processors)** $*$

CONCLUSION

∗ **for physical systems (⇒⇐ signal processors)** ∗

- **External variables are basic, but what ‘drives’ what, is not.**

CONCLUSION

∗ for physical systems ($\Rightarrow\!\Leftarrow$ signal processors) ∗

- External variables are basic, but what 'drives' what, is not.
- Interconnection, variable sharing, rather that input selection, is the basic mechanism by which a system interacts with its environment.

BONDGRAPHS

Views interconnected systems indeed in terms of

ports, modules, and interconnections.

BONDGRAPHS

Views interconnected systems indeed in terms of

ports, modules, and interconnections.

It is assumed that for each of the terminals the interconnection variables come in pairs:

an effort variable and a flow variable

their (inner) product must be power.

BONDGRAPHS

Views interconnected systems indeed in terms of

ports, modules, and interconnections.

It is assumed that for each of the terminals the interconnection variables come in pairs:

an effort variable and a flow variable

their (inner) product must be power.

Examples:

- **Electrical ports: effort: voltage, flow: current**
- **Mechanical ports: effort: force, flow: velocity**
- **Thermal ports: effort: T, flow: Q/T**
- **etc. etc.**

BONDGRAPHS

- **Bondgraphs ideas very good, brilliant**
- **certainly superior to SIMULINK©**

BONDGRAPHS

- **Bondgraphs ideas very good, brilliant**
- **certainly superior to SIMULINK©**
- **notation very awkward, mathematical notions primitive**
- **terminal variable structure seems limited to linearity**
- **some interconnections fail their assumptions:**
  mechanical terminals equate positions, NOT velocities

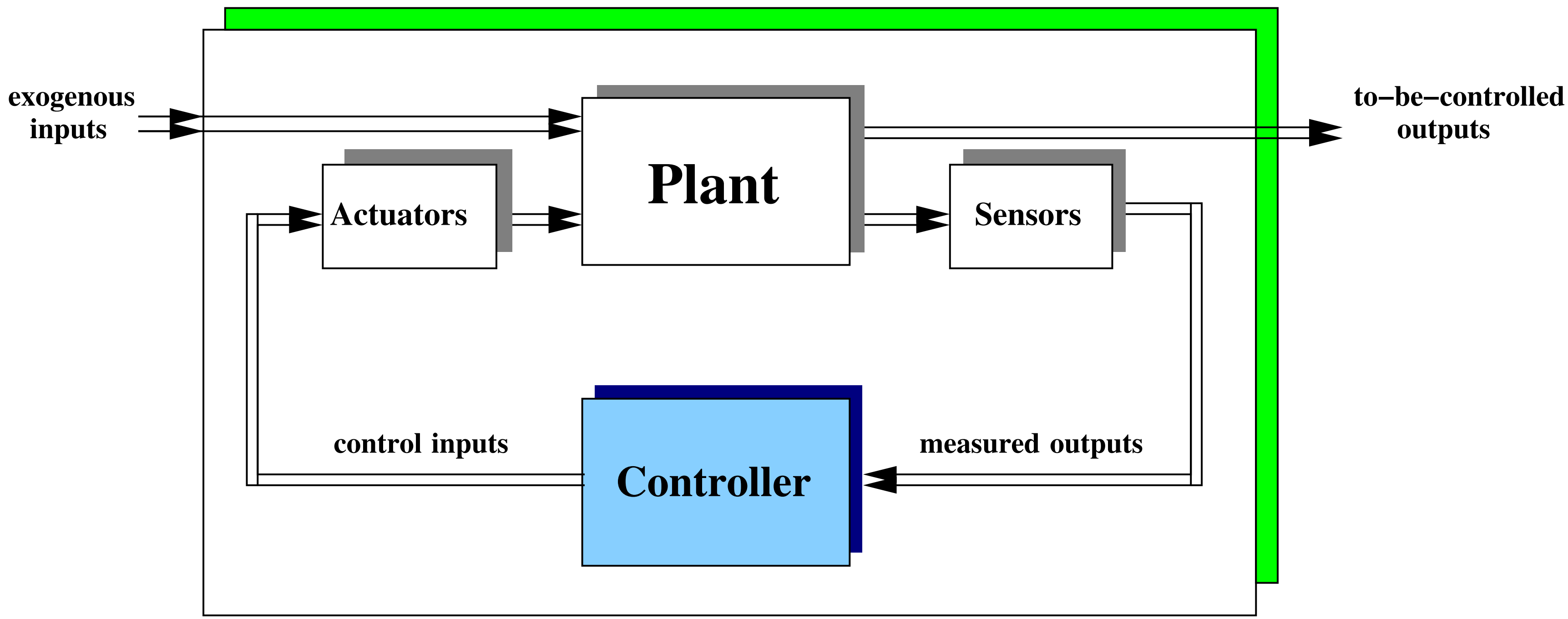
BONDGRAPHS

- **Bondgraphs ideas very good, brilliant**
- **certainly superior to SIMULINK©**
- **notation very awkward, mathematical notions primitive**
- **terminal variable structure seems limited to linearity**
- **some interconnections fail their assumptions: mechanical terminals equate positions, NOT velocities**
- **effort/flow, while apparently deep, remains unexplored**
- **interconnections happen via terminals, not ports.**
- **there is more structure to interconnection variables than effort/flow.**

CONTROL in a BEHAVIORAL SETTING

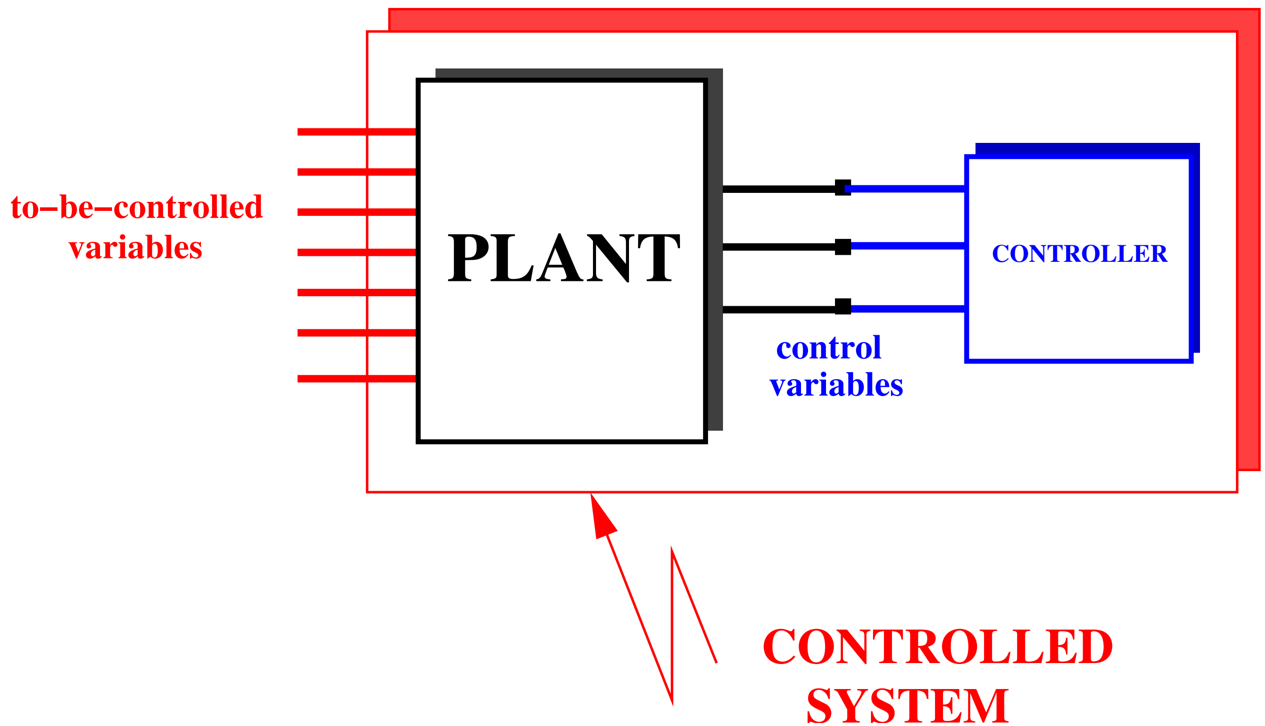
FEEDBACK CONTROL

The usual paradigm for control:



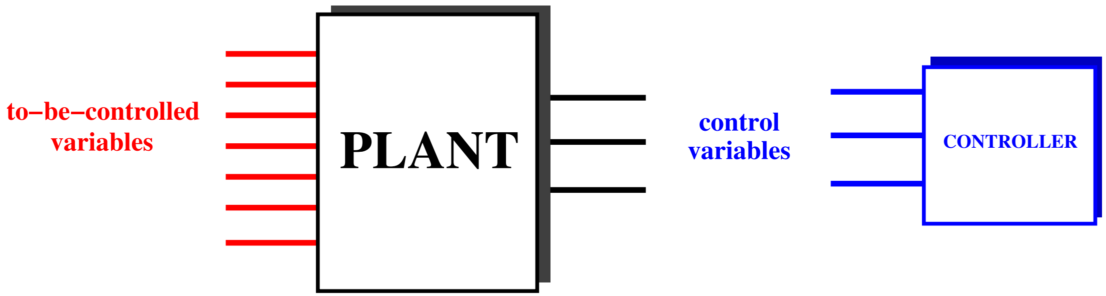
'Intelligent' Control

BEHAVIORAL CONTROL



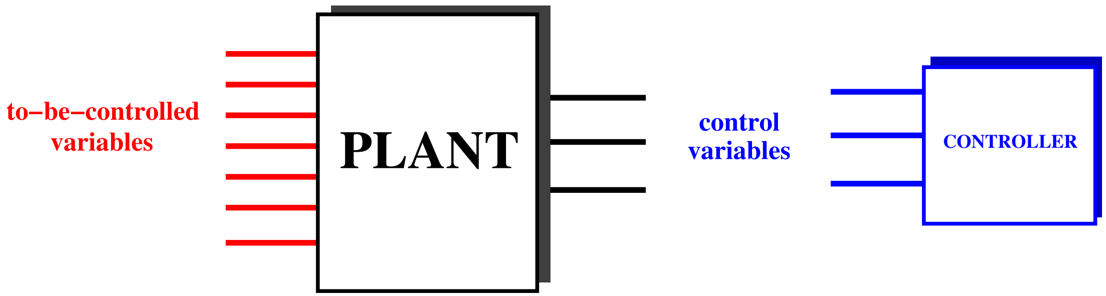
Control as interconnection

BEHAVIORAL CONTROL

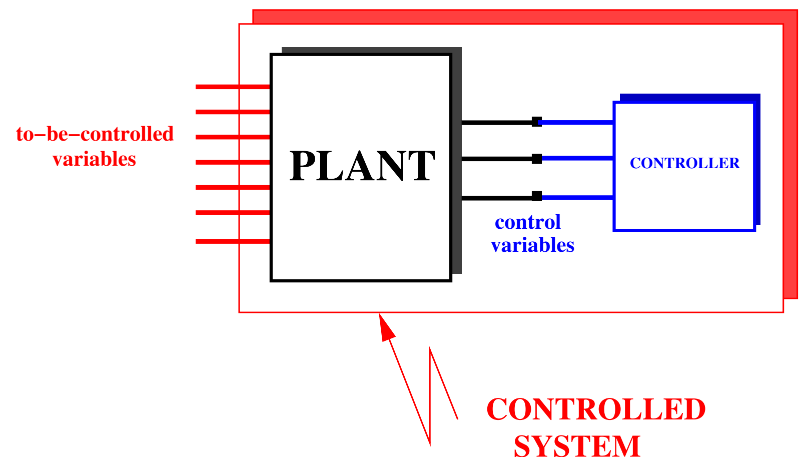


Before interconnection

BEHAVIORAL CONTROL



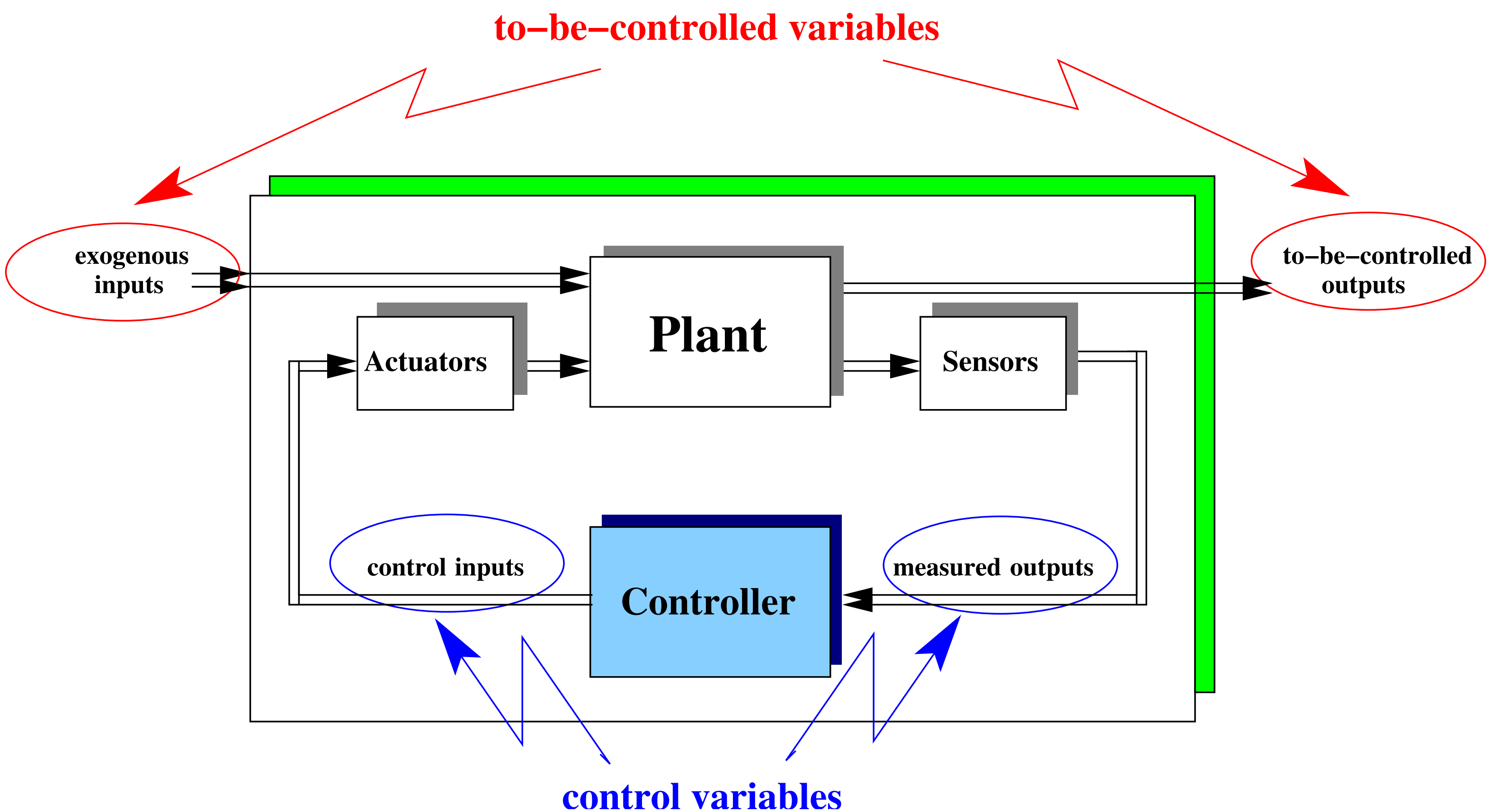
Before interconnection



After interconnection

Control = designing a subsystem

Feedback control as an example

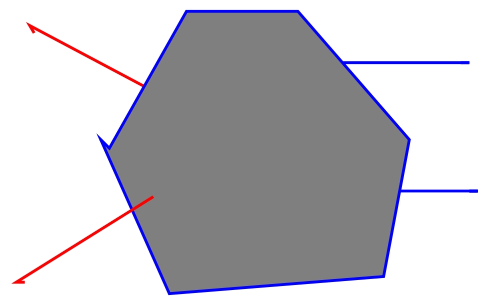


‘Example’

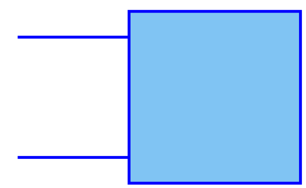
Many practical control devices do not function as feedback controllers! Dampers, heat fins, pressure valves, overflows, turbulence control strips, characteristic impedances, etc. etc.

‘Example’

Many practical control devices do not function as feedback controllers! Dampers, heat fins, pressure valves, overflows, turbulence control strips, characteristic impedances, etc. etc.



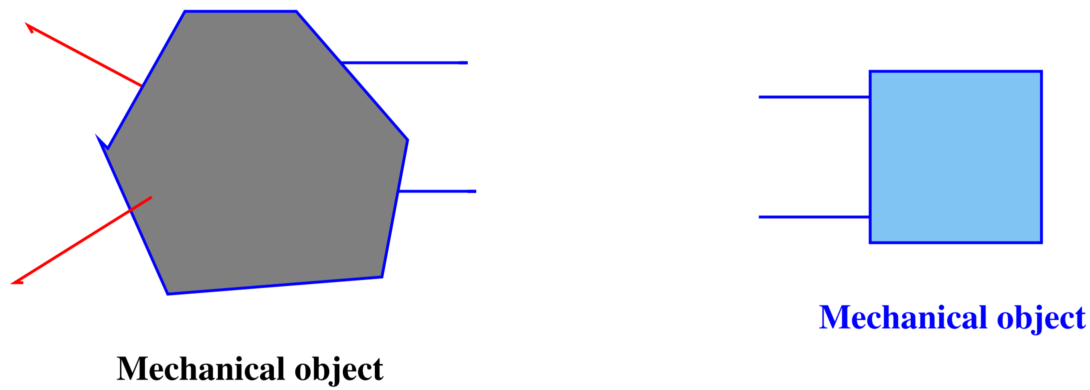
Mechanical object

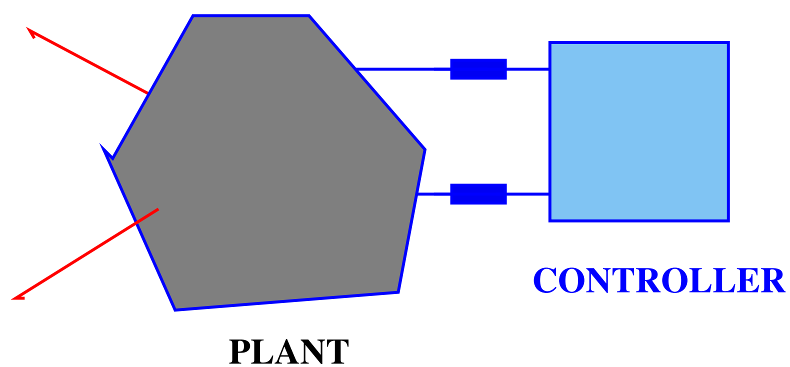


Mechanical object

‘Example’

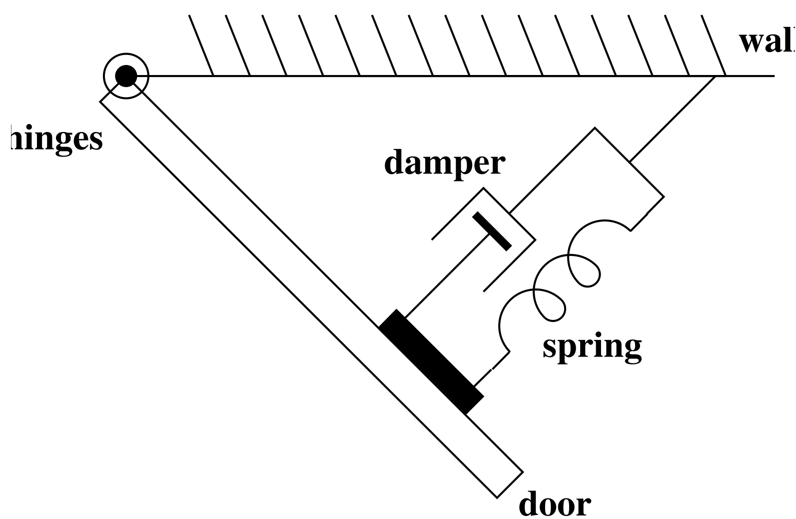
Many practical control devices do not function as feedback controllers! Dampers, heat fins, pressure valves, overflows, turbulence control strips, characteristic impedances, etc. etc.





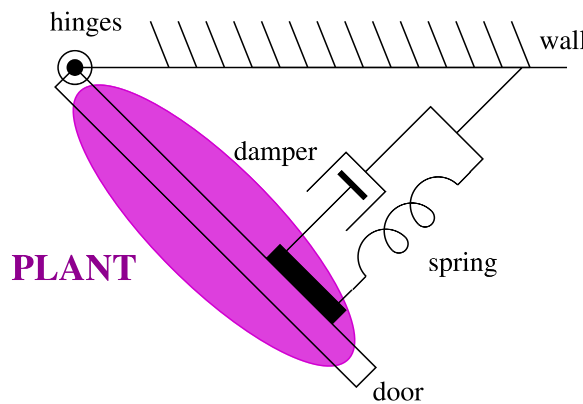
‘Example’

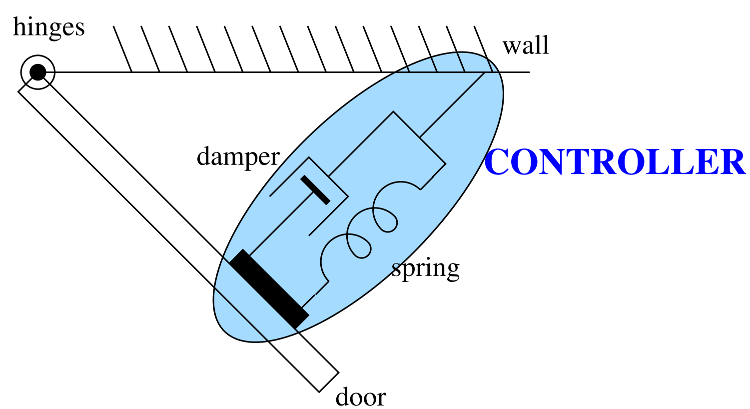
Many practical control devices do not function as feedback controllers! Dampers, heat fins, pressure valves, overflows, turbulence control strips, characteristic impedances, etc. etc.



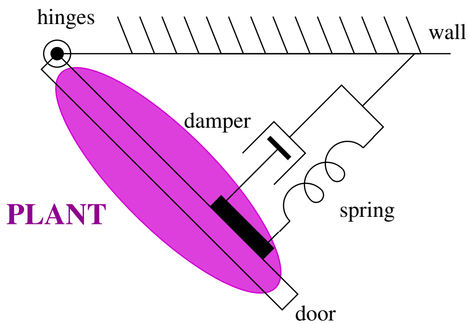
‘Example’

Many practical control devices do not function as feedback controllers! Dampers, heat fins, pressure valves, overflows, turbulence control strips, characteristic impedances, etc. etc.





'Example'

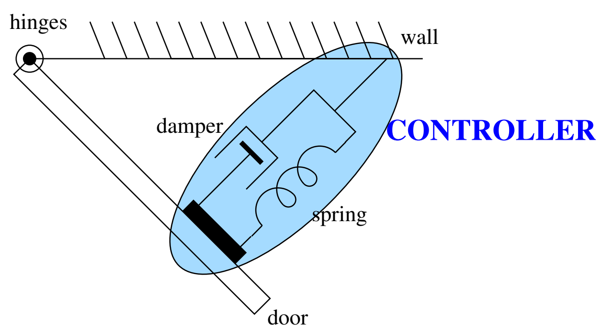


Equation of motion of the door (the plant):

$$M' \frac{d^2\theta}{dt^2} = F_c + F_e$$

θ: opening angle,

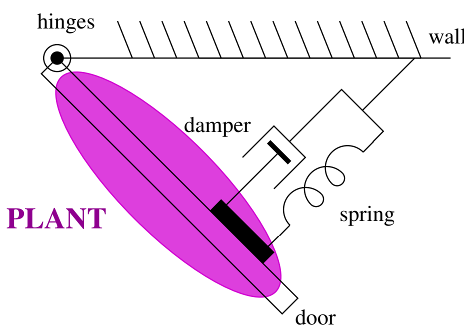
F_c force device, F_e exogenous force.



Door closing mechanism (the controller):

$$M'' \frac{d^2\theta}{dt^2} + D\frac{d\theta}{dt} + K\theta = -F_c.$$

‘Example’

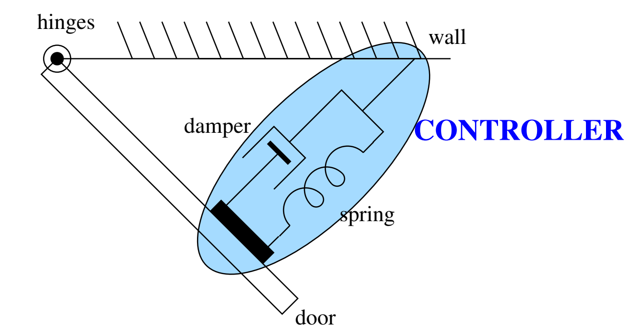


Equation of motion of the door (the plant):

$$M'\frac{d^2\theta}{dt^2} = F_c + F_e$$

θ: opening angle,

F_c force device, F_e exogenous force.



Door closing mechanism (the controller):

$$M''\frac{d^2\theta}{dt^2} + D\frac{d\theta}{dt} + K\theta = -F_c.$$

Controlled behavior:

$$(M' + M'')\frac{d^2\theta}{dt^2} + D\frac{d\theta}{dt} + K\theta = F_e$$

Specs: small overshoot, fast settling, not-to-high gain from $F_e \mapsto \theta$. Controller $\rightsquigarrow$ M', K and D.

Note: Plant: second order; Controller: second order; Controlled plant: second (not fourth) order.

Note: PDD controller, but no noise problems

MATHEMATIZATION

Domain of the to-be-controlled variables: $\mathbb{W}$

Domain of the control variables: $\mathbb{C}$

Typically: families of time-signals

MATHEMATIZATION

Full plant behavior:

$$\mathcal{P}_{\text{full}} = \{(w, c) \in \mathbb{W} \times \mathbb{C} \mid \textbf{allowed by plant laws}\}$$

Controller:

$$\mathcal{C} = \{c \in \mathbb{C} \mid \textbf{allowed by controller laws}\}$$

MATHEMATIZATION

Full plant behavior:

$$\mathcal{P}_{\text{full}} = \{(w, c) \in \mathbb{W} \times \mathbb{C} \mid \textbf{allowed by plant laws}\}$$

Controller:

$$\mathcal{C} = \{c \in \mathbb{C} \mid \textbf{allowed by controller laws}\}$$

Controlled behavior:

$$\mathcal{K} := \{w \in \mathbb{W} \mid \exists\, c \in \mathbb{C} \;\; \textbf{such that } (w, c) \in \mathcal{P}_{\text{full}} \textbf{ and } c \in \mathcal{C}\}.$$

MATHEMATIZATION

Controlled behavior:

$$\mathcal{K} := \{w \in \mathbb{W} \mid \exists\, c \in \mathbb{C} \text{ such that } (w, c) \in \mathcal{P}_{\text{full}} \text{ and } c \in \mathcal{C}\}.$$

We say that $\mathcal{C}$ *implements* $\mathcal{K}$, and that $\mathcal{K}$ is implementable

MATHEMATICAL MODEL

Controlled behavior:

$$\mathcal{K} := \{w \in \mathbb{W} \mid \exists\, c \in \mathbb{C} \text{ such that } (w, c) \in \mathcal{P}_{\text{full}} \text{ and } c \in \mathcal{C}\}.$$

We say that $\mathcal{C}$ *implements* $\mathcal{K}$, and that $\mathcal{K}$ is implementable

Questions:

- Which $\mathcal{C}$ implements the *desired controlled behavior* $\mathcal{D}$?
- Given $\mathcal{P}_{\text{full}}$, which $\mathcal{K} \subseteq \mathbb{W}$ are implementable?

We henceforth restrict attention to

linear time-invariant differential systems.

$\mathfrak{L}^\bullet$

We henceforth restrict attention to

linear time-invariant differential systems.

The *behavior* $\mathfrak{B}$ belongs to $\mathfrak{L}^{\mathtt{w}}$

:⇔

∃ a polynomial matrix $R \in \mathbb{R}^{\bullet \times \mathtt{w}}[\xi]$ such that

$$\mathfrak{B} = \{w \in \mathfrak{C}^\infty(\mathbb{R}, \mathbb{R}^{\mathtt{w}}) \mid R(\tfrac{d}{dt})w = 0\}.$$

$\mathfrak{L}^\bullet$

Plant:

$$\mathcal{P}_{\text{full}} \in \mathfrak{L}^{\mathtt{w}+\mathtt{c}}.$$

Controller:

$$\mathcal{C} \in \mathfrak{L}^{\mathtt{c}}.$$

Controlled system:

$$\mathcal{K} = \{w \in \mathfrak{C}^\infty(\mathbb{R}, \mathbb{R}^{\mathtt{w}}) \mid \exists\, c \in \mathcal{C} : (w, c) \in \mathcal{P}_{\text{full}}\}.$$

Plant:

$$\mathcal{P}_{\text{full}} \in \mathfrak{L}^{\mathtt{w}+\mathtt{c}}.$$

Controller:

$$\mathcal{C} \in \mathfrak{L}^{\mathtt{c}}.$$

Controlled system:

$$\mathcal{K} = \{w \in \mathfrak{C}^\infty(\mathbb{R}, \mathbb{R}^{\mathtt{w}}) \mid \exists\, c \in \mathcal{C} : (w, c) \in \mathcal{P}_{\text{full}}\}.$$

By the ‘elimination theorem’

$$\mathcal{K} \in \mathfrak{L}^{\mathtt{w}}$$

IMPLEMENTABILITY

Which behaviors $\mathcal{K} \in \mathfrak{L}^{\mathtt{w}}$ can be implemented by attaching a controller $\mathcal{C} \in \mathfrak{L}^{\mathtt{c}}$ to a given plant $\mathcal{P}_{\mathrm{full}} \in \mathfrak{L}^{\mathtt{w}+\mathtt{c}}$?

IMPLEMENTABILITY

Which behaviors $\mathcal{K} \in \mathfrak{L}^{\mathtt{w}}$ can be implemented by attaching a controller $\mathcal{C} \in \mathfrak{L}^{\mathtt{c}}$ to a given plant $\mathcal{P}_{\text{full}} \in \mathfrak{L}^{\mathtt{w}+\mathtt{c}}$?

This question has a very concrete and intuitive answer.

Theorem: Let $\mathcal{P}_{\text{full}} \in \mathfrak{L}^{\mathtt{w}+\mathtt{c}}$ be given.

The behavior $\mathcal{K} \in \mathfrak{L}^{\mathtt{w}}$ is implementable if and only if

$$\mathcal{N} \subseteq \mathcal{K} \subseteq \mathcal{P}$$

IMPLEMENTABILITY

The behavior $\mathcal{K} \in \mathfrak{L}^{\mathtt{w}}$ is implementable if and only if

$$\mathcal{N} \subseteq \mathcal{K} \subseteq \mathcal{P}$$

where $\mathcal{N} \in \mathfrak{L}^{\mathtt{w}}$ is the *hidden behavior* defined by

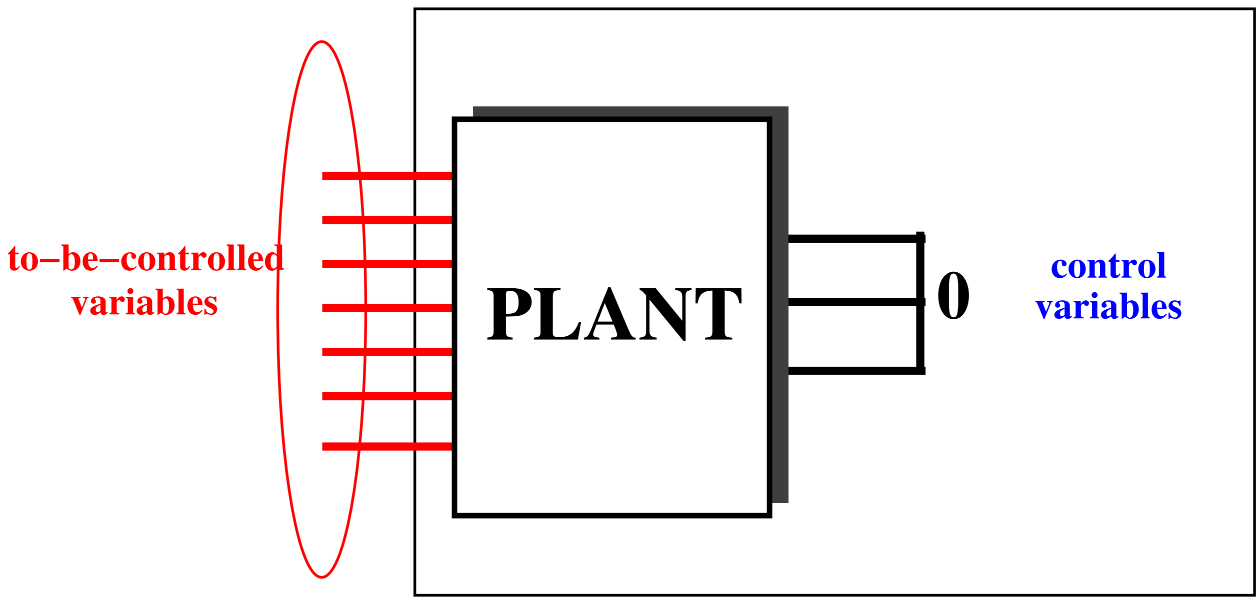
$$\mathcal{N} := \{w \in \mathfrak{C}^{\infty}(\mathbb{R}, \mathbb{R}^{\mathtt{w}}) \mid (w, 0) \in \mathcal{P}_{\text{full}}\},$$

and $\mathcal{P}$ is the *manifest plant behavior* defined by

$$\mathcal{P} := \{w \in \mathfrak{C}^{\infty}(\mathbb{R}, \mathbb{R}^{\mathtt{w}}) \mid \exists\, c : (w, c) \in \mathcal{P}_{\text{full}}\}.$$

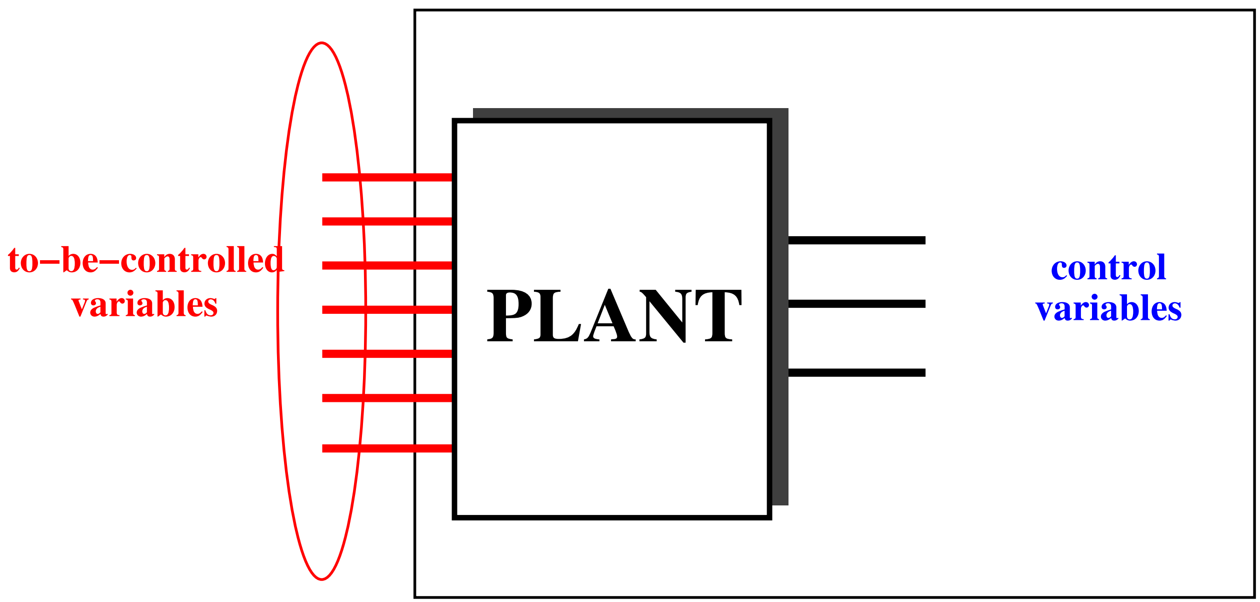
IMPLEMENTABILITY

$\mathcal{N} \in \mathfrak{L}^{\mathtt{w}}$, the ***hidden behavior***



IMPLEMENTABILITY

$\mathcal{P} \in \mathfrak{L}^{\mathtt{w}}$, the ***manifest plant behavior***



IMPLEMENTABILITY

The behavior $\mathcal{K} \in \mathfrak{L}^{\mathtt{w}}$ is implementable if and only if

$$\mathcal{N} \subseteq \mathcal{K} \subseteq \mathcal{P}$$

This theorem reduces control to linear algebra / functional analysis: finding suitable subspaces wedged between given subspaces.

Example:

Assume observability of the to-be-controlled variables w from the control variables $c \Leftrightarrow \mathcal{N} = \{0\}$. Assume $\mathcal{P} \neq \{0\}$, controllable.

$\Rightarrow$ pole assignability $\Rightarrow$ stabilizability

e.g., $\frac{d}{dt}x = Ax + Bu, y = Cx + Du, c = (u, y), w = x.$

IMPLEMENTABILITY

The behavior $\mathcal{K} \in \mathfrak{L}^{\mathtt{w}}$ is implementable if and only if

$$\mathcal{N} \subseteq \mathcal{K} \subseteq \mathcal{P}$$

This theorem reduces control to linear algebra / functional analysis: finding suitable subspaces wedged between given subspaces.

LQ-control and $\mathcal{H}_\infty$ control are very neatly worked out from this point of view/

Regularity

The *full controlled behavior* $\mathcal{K}_{\mathrm{full}} \subseteq \mathcal{P}_{\mathrm{full}}$ is defined by

$$\mathcal{K}_{\mathrm{full}} := \{(w, c) \in \mathcal{P}_{\mathrm{full}} \mid c \in \mathcal{C}\}.$$

Regularity

The ***full controlled behavior*** $\mathcal{K}_{\text{full}} \subseteq \mathcal{P}_{\text{full}}$ **is defined by**

$$\mathcal{K}_{\text{full}} := \{(w, c) \in \mathcal{P}_{\text{full}} \mid c \in \mathcal{C}\}.$$

Consider the maps $\mathtt{m}, \mathtt{p} : \mathfrak{L}^{\mathtt{w}} \to \{0, 1, \ldots, \mathtt{w}\}$
with $\mathtt{m}(\mathfrak{B})$ **the number of input variables,**
and $\mathtt{p}(\mathfrak{B})$ **the number of output variables in** $\mathfrak{B}$**.**

Regularity

The *full controlled behavior* $\mathcal{K}_{\text{full}} \subseteq \mathcal{P}_{\text{full}}$ is defined by

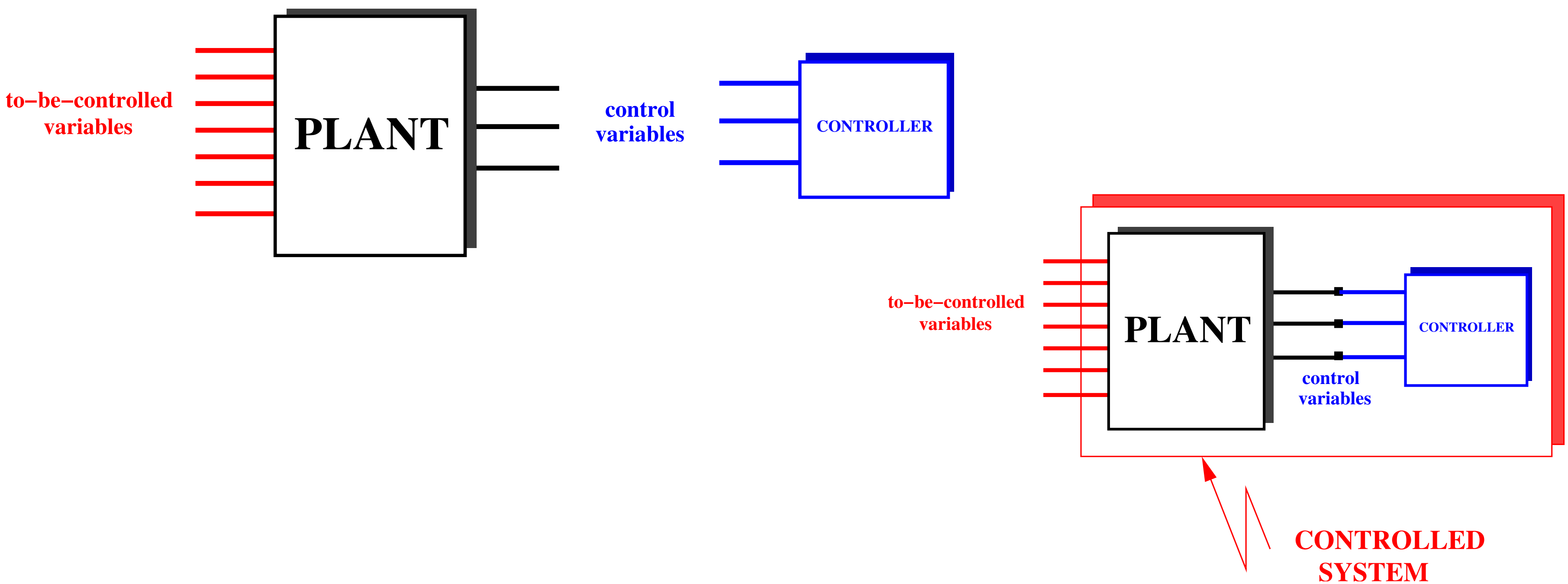
$$\mathcal{K}_{\text{full}} := \{(w, c) \in \mathcal{P}_{\text{full}} \mid c \in \mathcal{C}\}.$$

Consider the maps $\mathtt{m}, \mathtt{p} : \mathfrak{L}^{\mathtt{w}} \rightarrow \{0, 1, \ldots, \mathtt{w}\}$
with $\mathtt{m}(\mathfrak{B})$ the number of input variables,
and $\mathtt{p}(\mathfrak{B})$ the number of output variables in $\mathfrak{B}$.

The controller $\mathcal{C} \in \mathfrak{L}^{\mathtt{c}}$ is said to be regular if

$$\mathtt{p}(\mathcal{K}_{\text{full}}) = \mathtt{p}(\mathcal{P}_{\text{full}}) + \mathtt{p}(\mathcal{C}).$$

Regularity

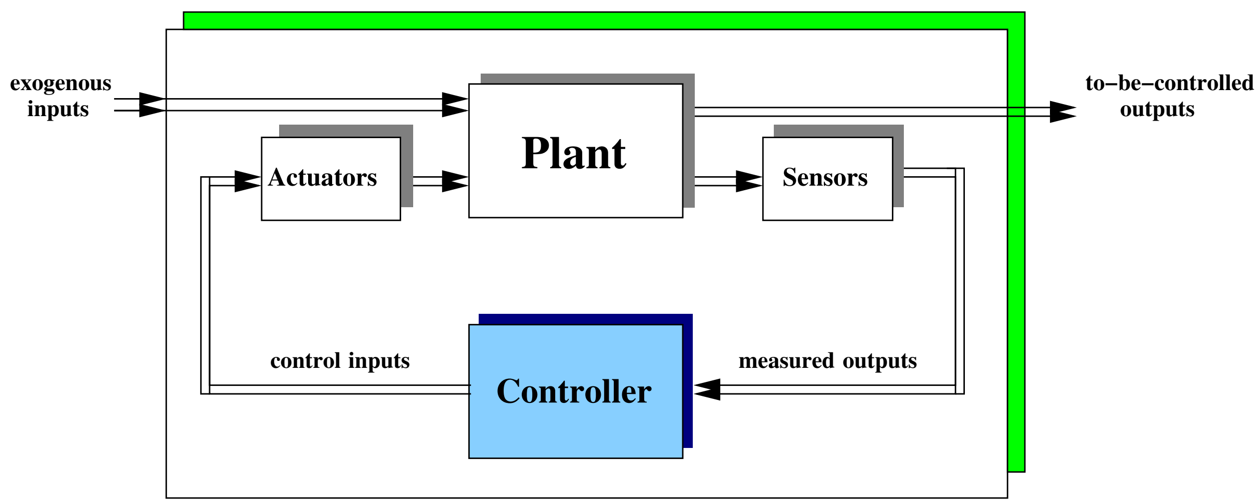


Regularity :=

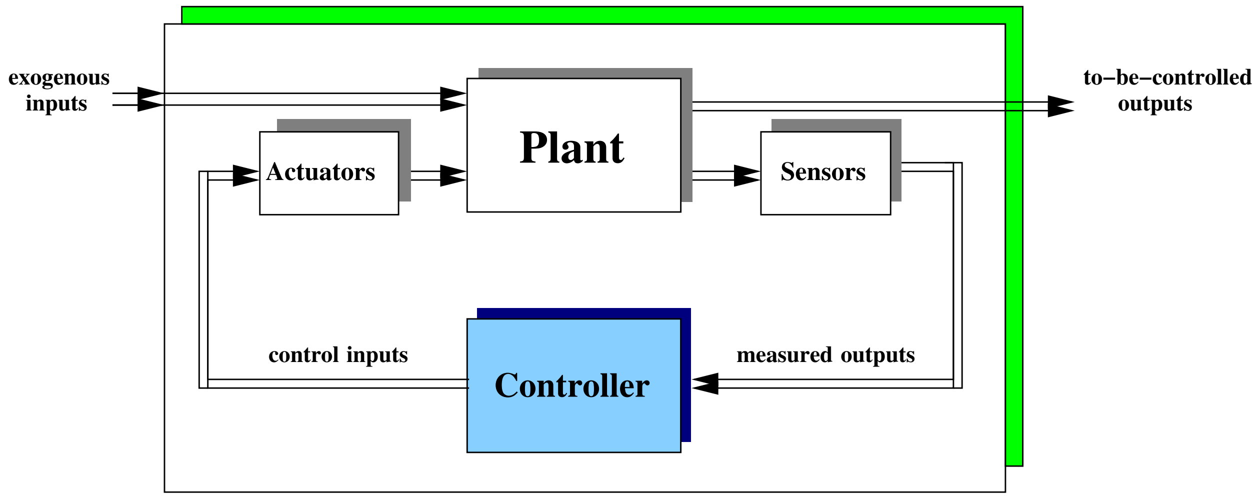
if the controller has p bound (i.e. output) variables, then the plant looses p free variables after interconnection.

Regularity

A controller is regular if and only if it can be realized as a feedback controller with a (possibly non-proper) transfer function from an output to an input in $\mathcal{P}_{\text{full}}$ for an input/output partition of $\mathcal{C}$.



Regularity



⇒ **A controller is regular if and only if it can be viewed as an ‘intelligent controller’ that processes sensor inputs outputs into actuator inputs.**

Regularity

$\Rightarrow$ **A controller is regular if and only if it can be viewed as an 'intelligent controller' that processes sensor inputs outputs into actuator inputs.**

If $\mathcal{P}$ is controllable, then every implementable controlled behavior $\mathcal{K}$ is actually regularly implementable.

Regularity

$\Rightarrow$ **A controller is regular if and only if it can be viewed as an 'intelligent controller' that processes sensor inputs outputs into actuator inputs.**

If $\mathcal{P}$ is controllable, then every implementable controlled behavior $\mathcal{K}$ is actually regularly implementable.

In feedback control, we have the additional property that the controller can be (de)coupled at any time. No state perparatiuon is required in attaching the controller.

A LOOK BACK

What have we been trying to do, really ?

A LOOK BACK

What have we been trying to do, really ?

Set up a ‘correct’ mathematical framework for discussing dynamical systems.

Usable in control, signal processing, econometrics, and, especially, incorporating in an honest way the classical models of physical systems.

A LOOK BACK

What have we been trying to do, really ?

Basic observations:

- **Any reasonable theory takes open systems as the basic paradigm.**
- **Most dynamical models will be described by (differential or difference) equations, but we need a basic notion of equivalence of models.**
- **First principles models invariably contain auxiliary variables**
- $\exists$ **a complete theory for linear time-invariant systems.**

A LOOK BACK

What have we been trying to do, really ?

Basic observations:

- **Any reasonable theory takes open systems as the basic paradigm. $\rightsquigarrow$ the predominance in mathematical research of closed systems is very hard to comprehend.**
- **Most dynamical models will be described by (differential or difference) equations, but we need a basic notion of equivalence of models.**
- **First principles models invariably contain auxiliary variables**
- **$\exists$ a complete theory for linear time-invariant systems.**

A LOOK BACK

What have we been trying to do, really ?

Basic observations:

- **Any reasonable theory takes open systems as the basic paradigm.**
- **Most dynamical models will be described by (differential or difference) equations, but we need a basic notion of equivalence of models.** $\rightsquigarrow$ **the behavior.**
- **First principles models invariably contain auxiliary variables**
- $\exists$ **a complete theory for linear time-invariant systems.**

A LOOK BACK

What have we been trying to do, really ?

Basic observations:

- **Any reasonable theory takes open systems as the basic paradigm.**
- **Most dynamical models will be described by (differential or difference) equations, but we need a basic notion of equivalence of models.**
- **First principles models invariably contain auxiliary variables $\leadsto$ manifest and latent variables.**
- **$\exists$ a complete theory for linear time-invariant systems.**

A LOOK BACK

What have we been trying to do, really ?

Basic observations:

- **Any reasonable theory takes open systems as the basic paradigm.**
- **Most dynamical models will be described by (differential or difference) equations, but we need a basic notion of equivalence of models.**
- **First principles models invariably contain auxiliary variables**
- **$\exists$ a complete theory for linear time-invariant systems.**
  $\rightsquigarrow$polyomial matrix based models, with as highlights the elimination thm., controllability, and image repr.

A LOOK BACK

What have we been trying to do, really ?

A LOOK BACK

What have we been trying to do, really ?

Basic observations:

- **The manifest variables of systems do not come as input/output pairs. On a physical terminal, many variables live simulaneously. I/O structures give the wrong suggestion. An I/O partition, if possible at all, is usually not unique, and if needed, depends on the purpose of the model.**
- **An input/output model is simply not a ‘map’.**
- **The state is a construct, and so are the input and output.**
- **Many technologically very relevant controllers are not sensor-output-to-actuator-input signal processors.**

A LOOK BACK

What have we been trying to do, really ?

Basic observations:

- **The manifest variables of systems do not come as input/output pairs. On a physical terminal, many variables live simulaneously. I/O structures give the wrong suggestion. An I/O partition, if possible at all, is usually not unique, and if needed, depends on the purpose of the model.**
  $\rightsquigarrow$ The classical I/O framework fails in the first and most elementary examples.
- **An input/output model is simply not a ‘map’.**
- **The state is a construct, and so are the input and output.**
- **Many technologically very relevant controllers are not sensor-output-to-actuator-input signal processors.**

A LOOK BACK

What have we been trying to do, really ?

Basic observations:

- **The manifest variables of systems do not come as input/output pairs. On a physical terminal, many variables live simulaneously. I/O structures give the wrong suggestion. An I/O partition, if possible at all, is usually not unique, and if needed, depends on the purpose of the model.**
- **An input/output model is simply not a ‘map’.**
  $\rightsquigarrow$ This is the historical raison d’être of state models .
- **The state is a construct, and so are the input and output.**
- **Many technologically very relevant controllers are not sensor-output-to-actuator-input signal processors.**

A LOOK BACK

What have we been trying to do, really ?

Basic observations:

- **The manifest variables of systems do not come as input/output pairs. On a physical terminal, many variables live simulaneously. I/O structures give the wrong suggestion. An I/O partition, if possible at all, is usually not unique, and if needed, depends on the purpose of the model.**
- **An input/output model is simply not a ‘map’.**
- **The state is a construct, and so are the input and output. $\rightsquigarrow$ Algorithmically well worked out for linear time-invariant systems.**
- **Many technologically very relevant controllers are not sensor-output-to-actuator-input signal processors.**

A LOOK BACK

What have we been trying to do, really ?

Basic observations:

- **The manifest variables of systems do not come as input/output pairs. On a physical terminal, many variables live simulaneously. I/O structures give the wrong suggestion. An I/O partition, if possible at all, is usually not unique, and if needed, depends on the purpose of the model.**
- **An input/output model is simply not a ‘map’.**
- **The state is a construct, and so are the input and output.**
- **Many technologically very relevant controllers are not sensor-output-to-actuator-input signal processors.**
- **The behavioral approach is consistent, pedagogically attractive, pragmatic, and practical.**

Thank you

Thank you

Thank you

Thank you

Thank you

Thank you

Thank you

Thank you

End of the Lecture V