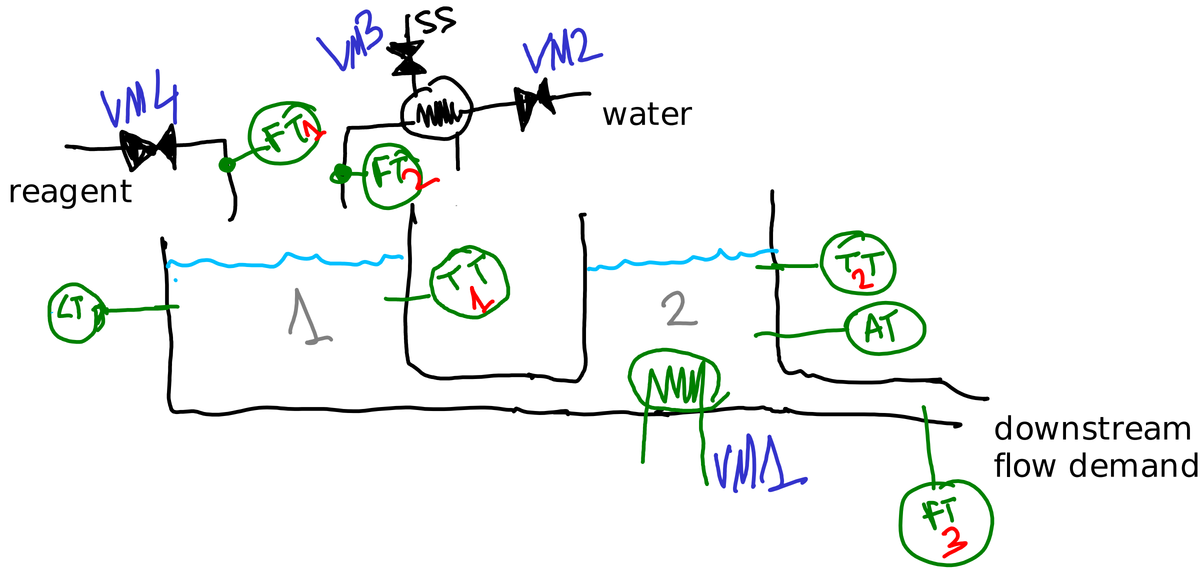
Control structures for a 2 tank process

Basic description and motivation



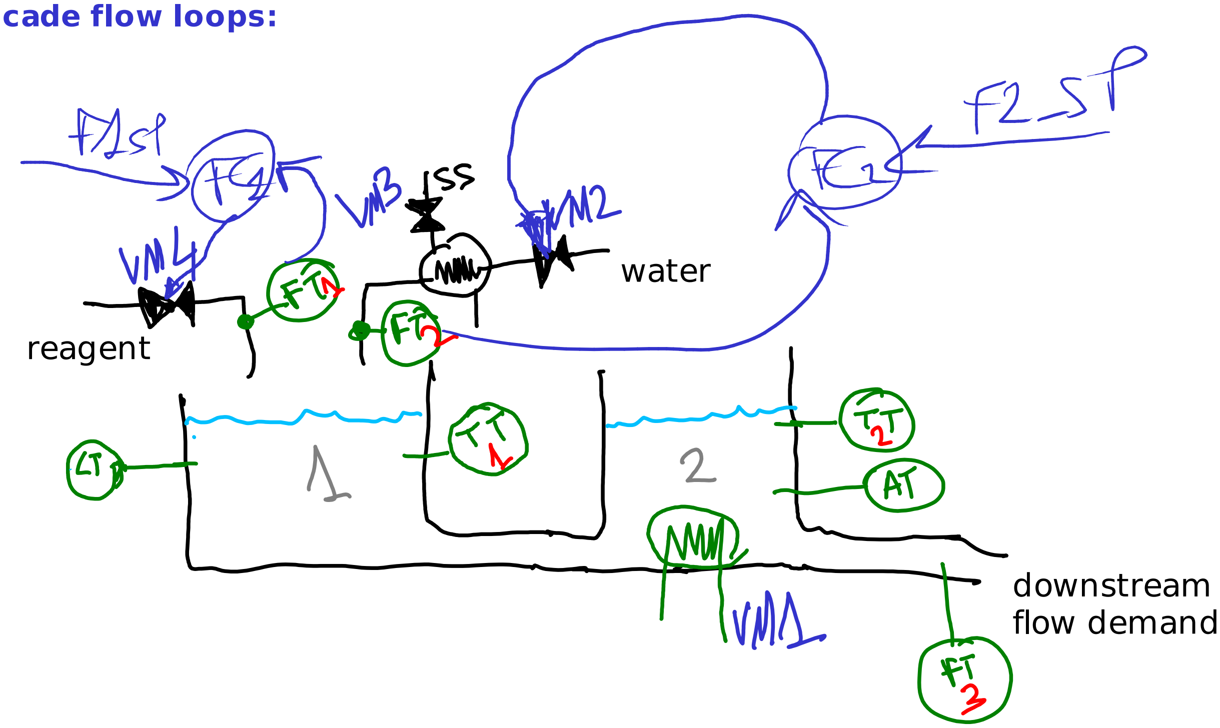
Presentations in Video:

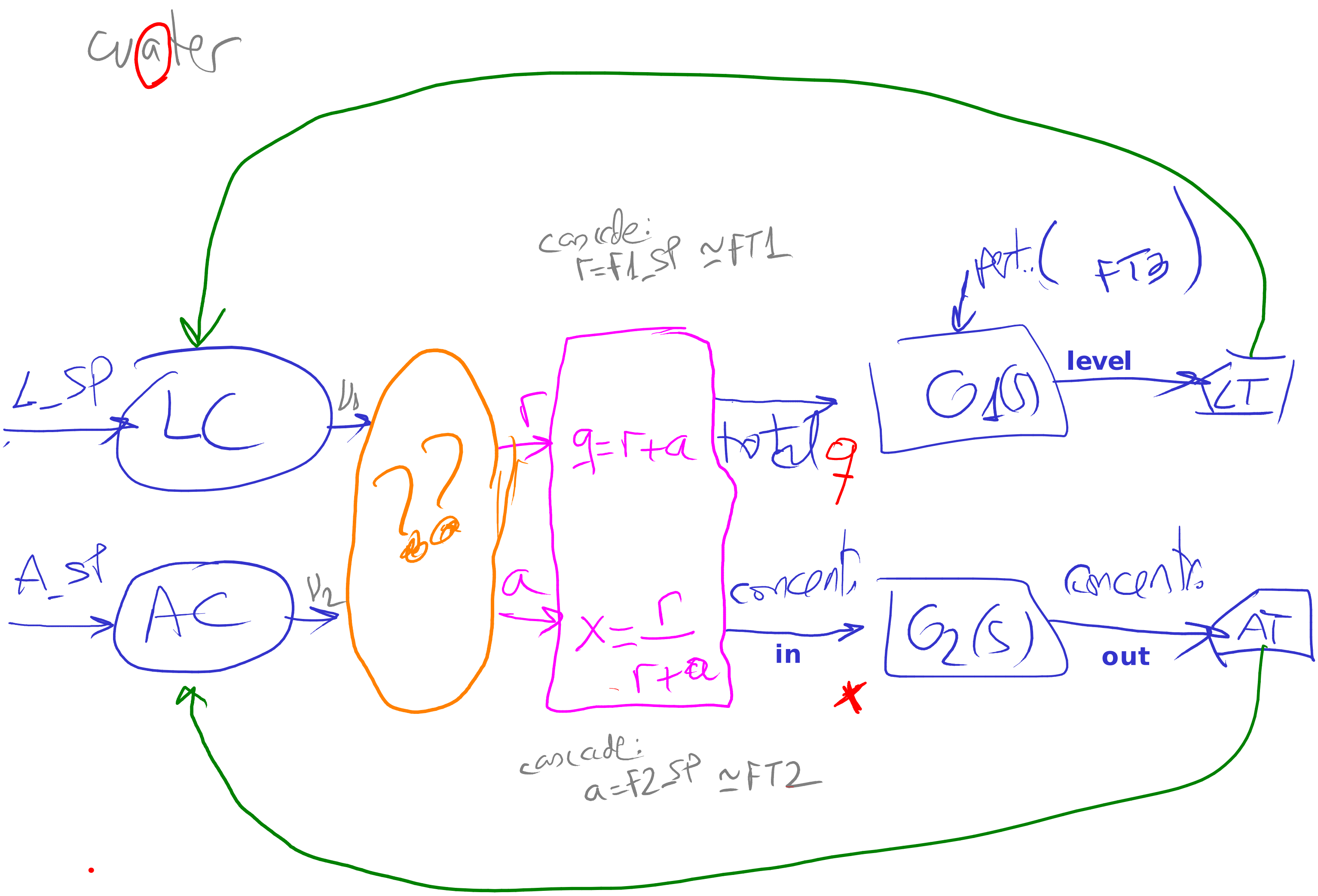
http://personales.upv.es/asala/YT/V/mzrgaEN.html

http://personales.upv.es/asala/YT/V/mzratdc1EN.html

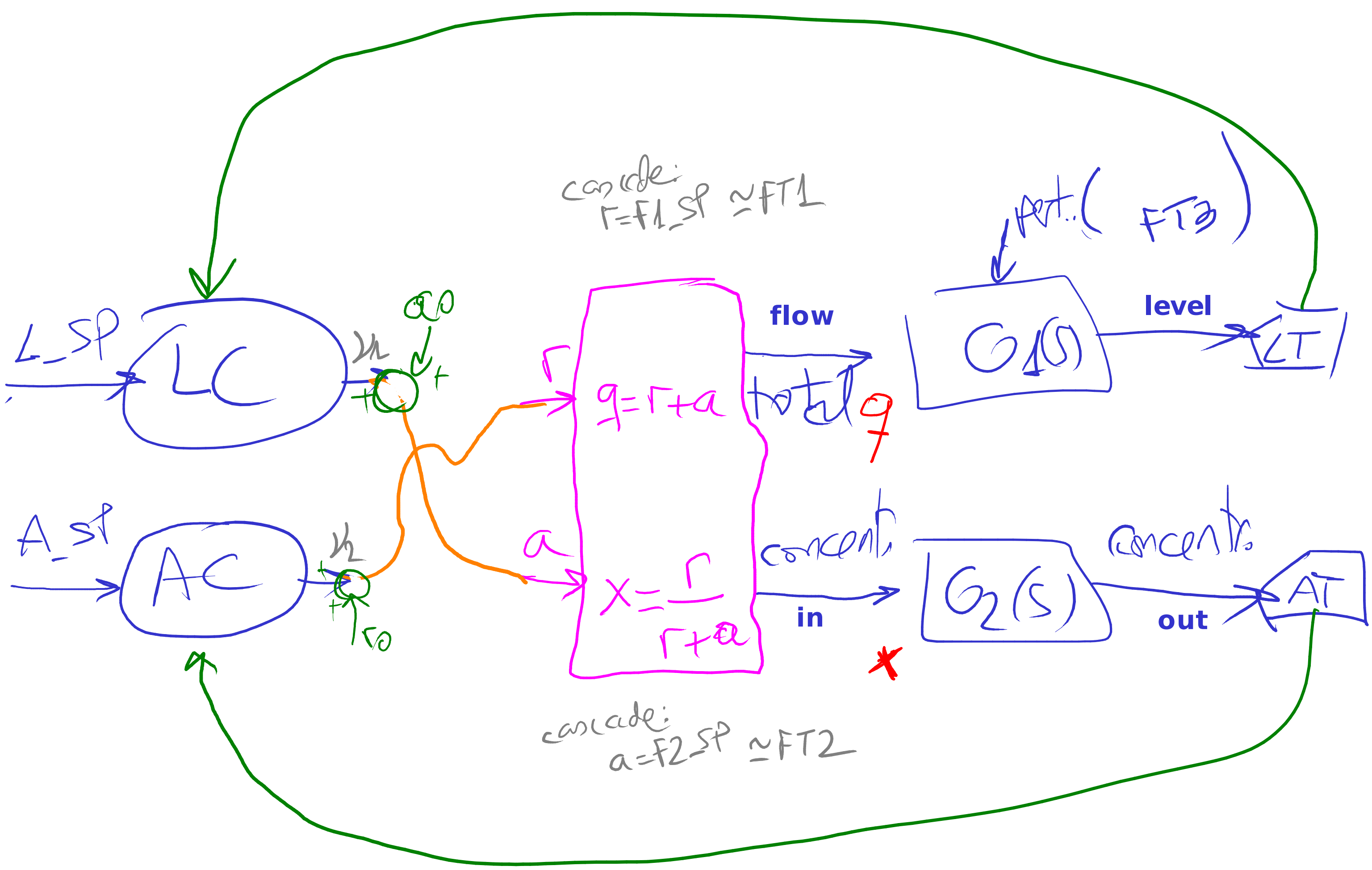
http://personales.upv.es/asala/YT/V/mzratdc2EN.html

Cascade flow loops:





MULTILOOP (nominal a=5, r=1; q=6, x=1/6):



RATIO control:

$r = \rho a$

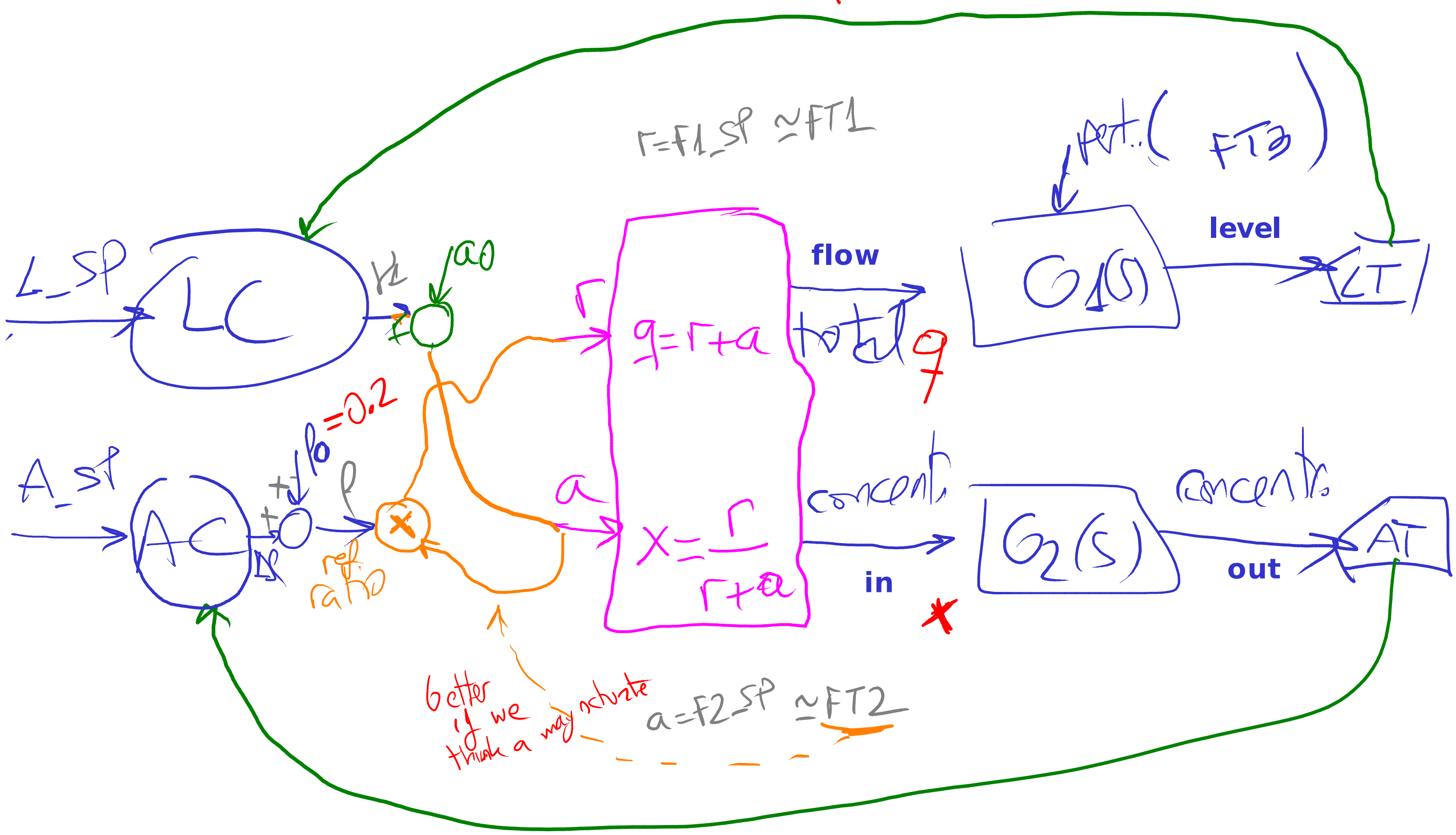
$a = y_1 + a_0$

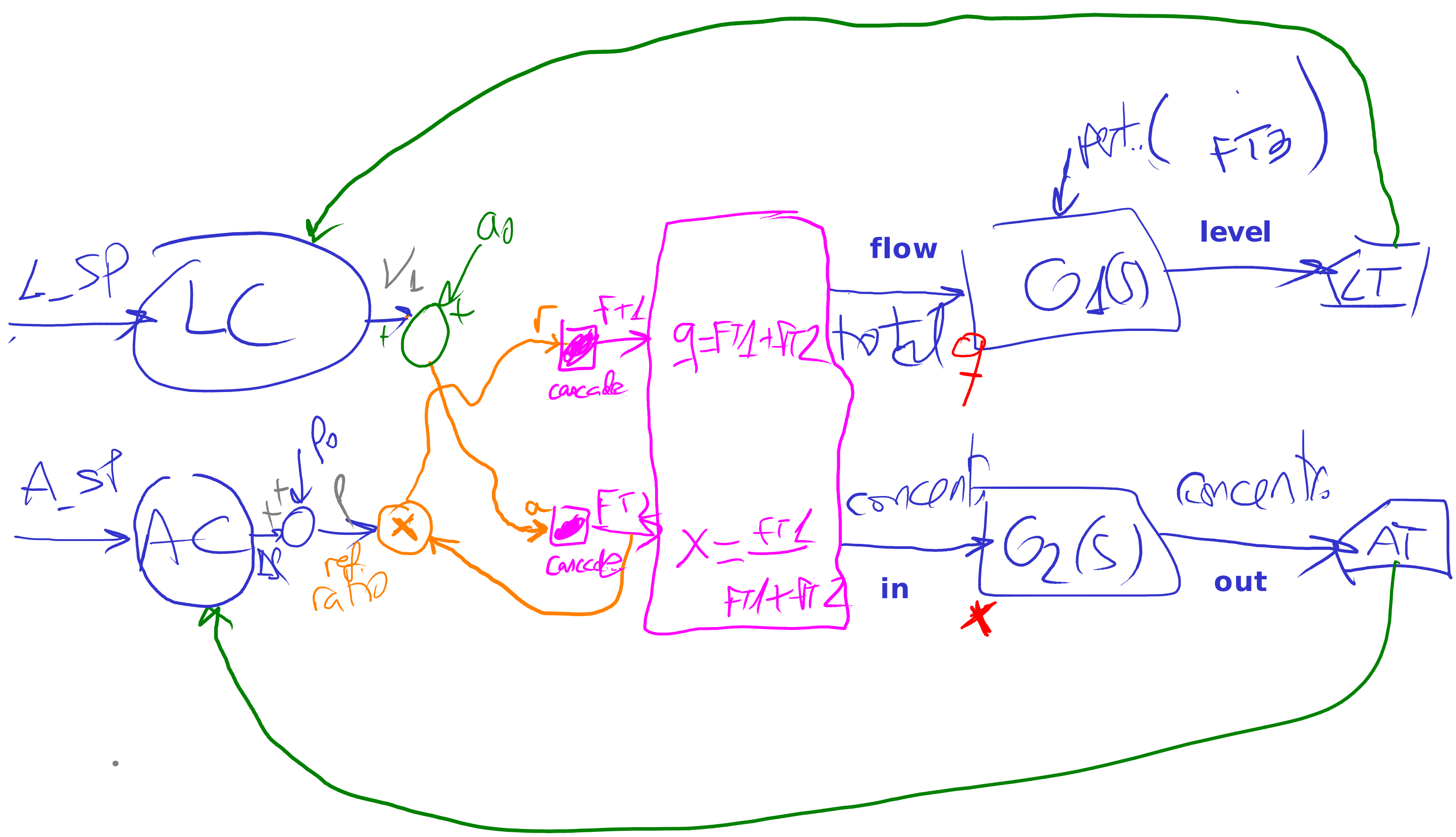
$r = (\rho_0 + \Delta\rho)(\Delta a + a_0)$

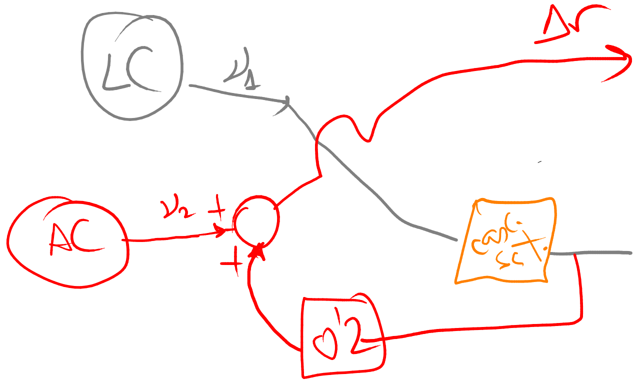
$r_0 = \rho_0 a_0$

$\Delta r = \rho_0 \Delta a + a_0 \Delta\rho + \Delta a \Delta\rho \to 0$

$\Delta r = \rho_0 y_1 + y_2$



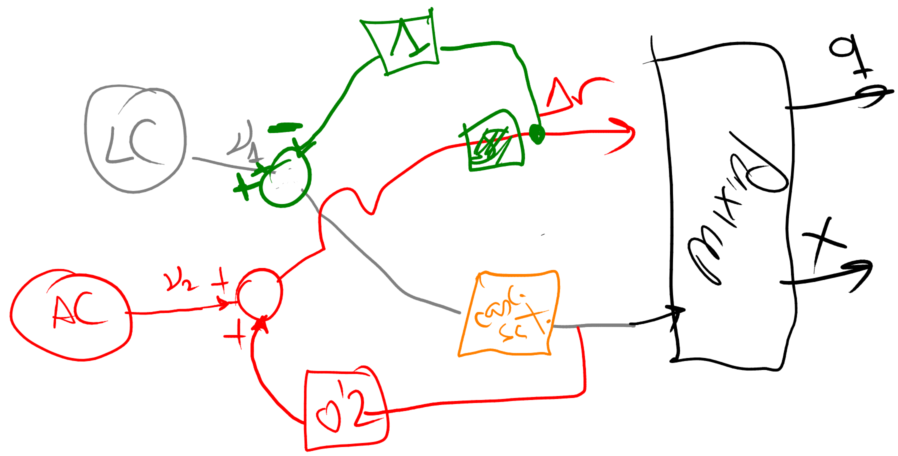




$$\Delta r = 0{,}2\,\nu_1 + \nu_2$$

decoupling term

multiloop AC action



Decoupling in INVERSE form:

no lin.:

$r = \rho \cdot a$

$a = \nu_1 - r + a_0$

$\rightarrow a + r = \nu_1 + a_0$

$(1+\rho)a = \nu_1 + a_0$

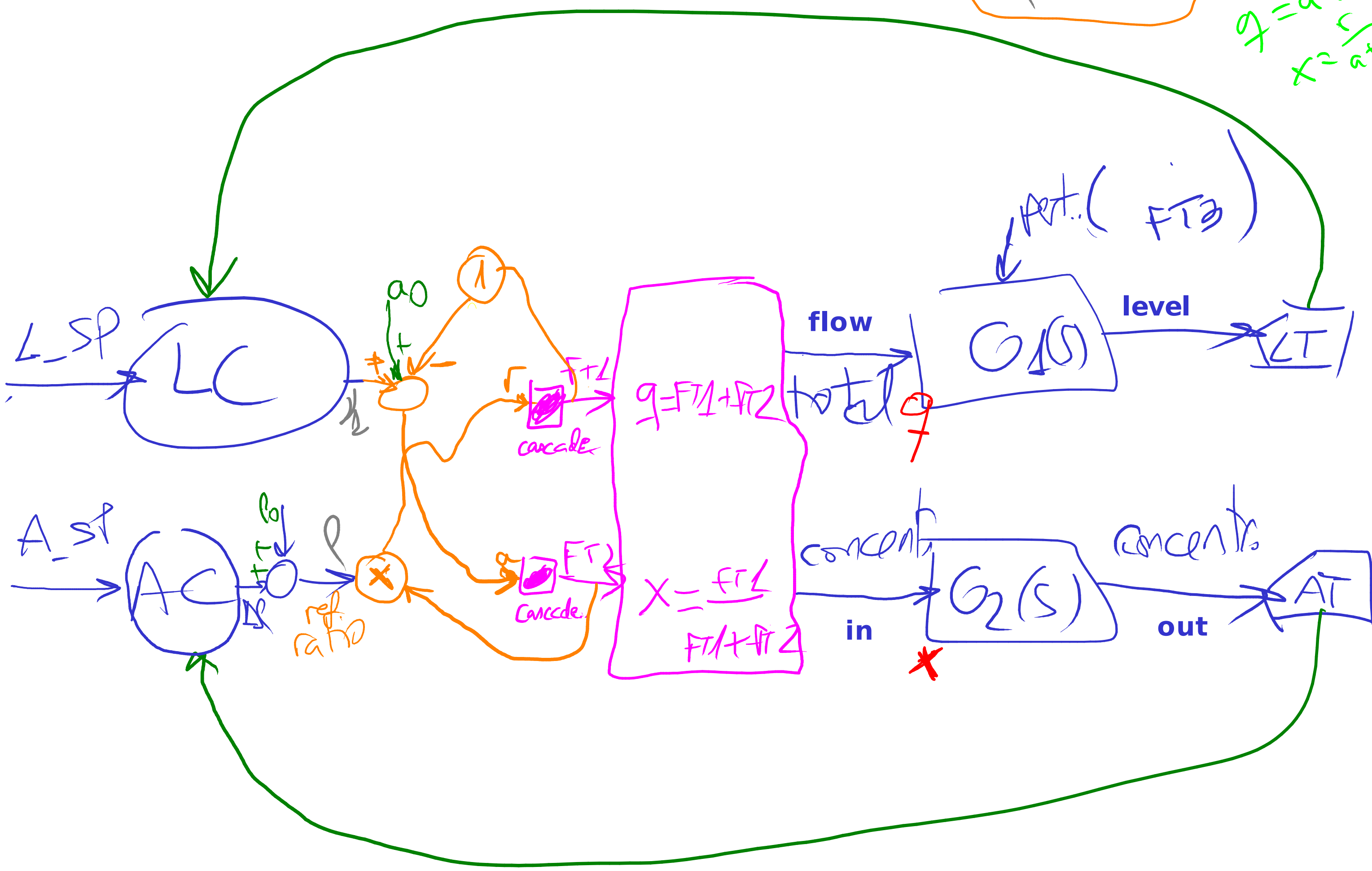
$a = \left(\frac{1}{1+\rho}\right)(\nu_1 + a_0)$

$r = \frac{\rho}{1+\rho}(\nu_1 + a_0)$

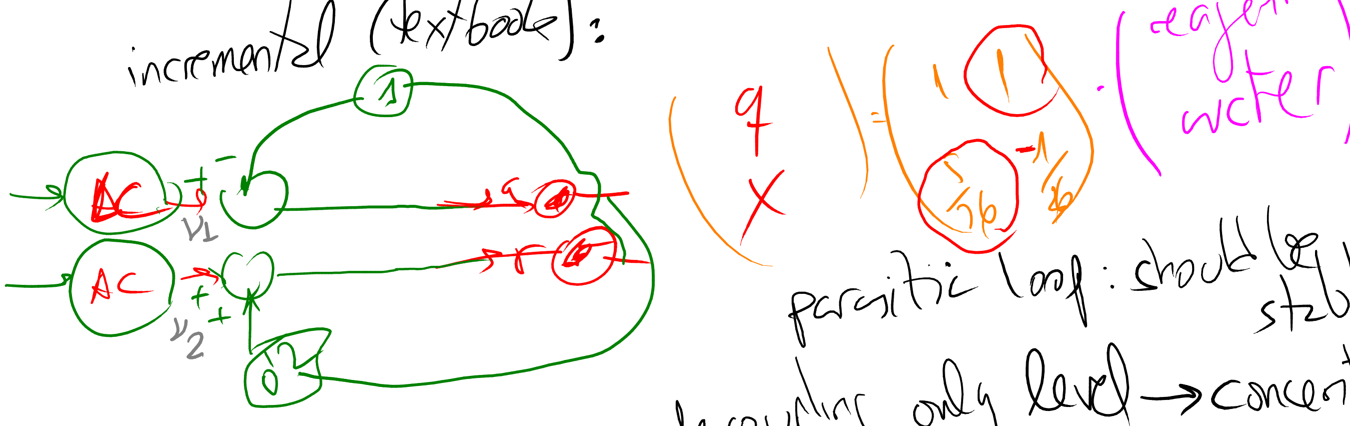
$q = a + r$

$x = \frac{r}{a+r}$

$$\begin{cases} x = \frac{\rho}{1+\rho} \\ q = \nu_1 + a_0 \end{cases}$$



incremental (textbook):



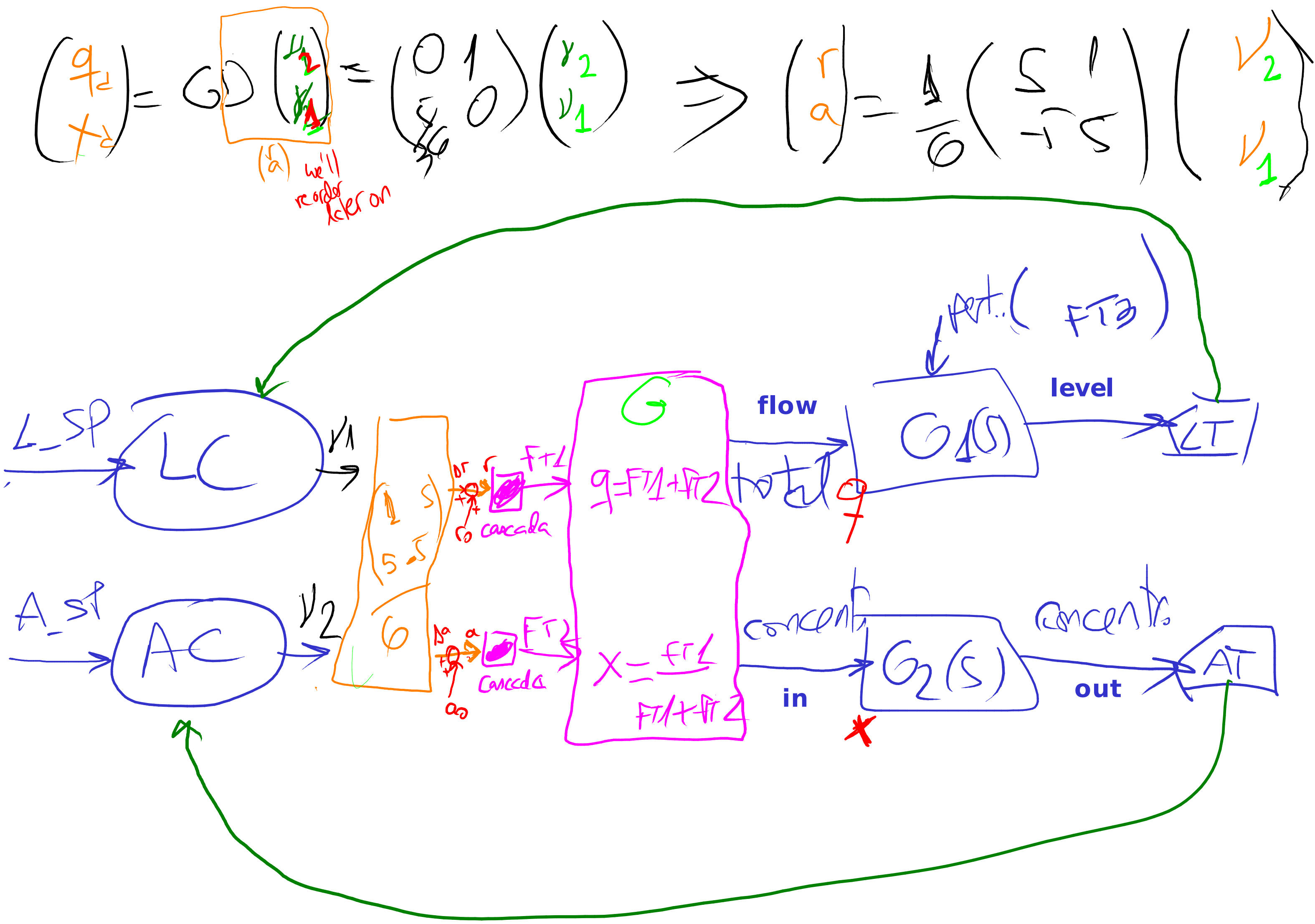
parasitic loop: should be stable!

decoupling only level → concentration

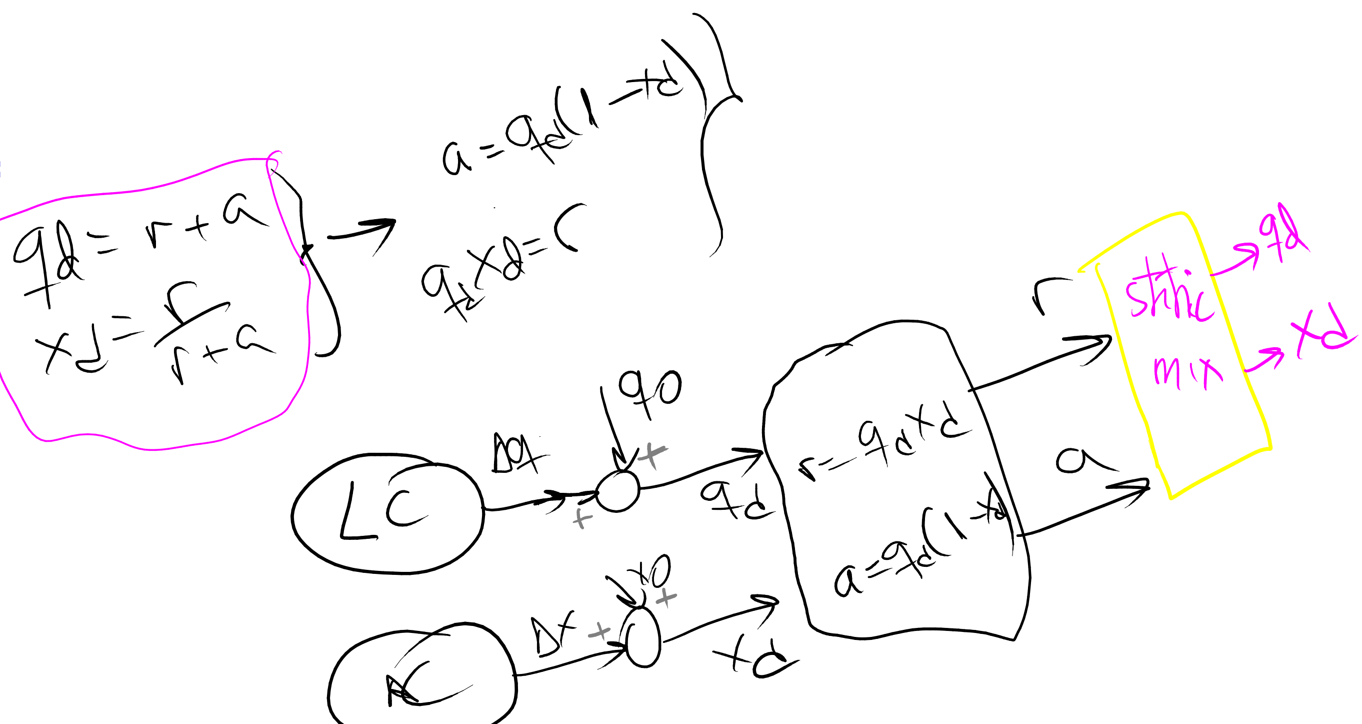
results in: $r = \nu_2 + \rho_0 \nu_1$

Decoupling in DIRECT form:

LINEAR



NON-LINEAR:



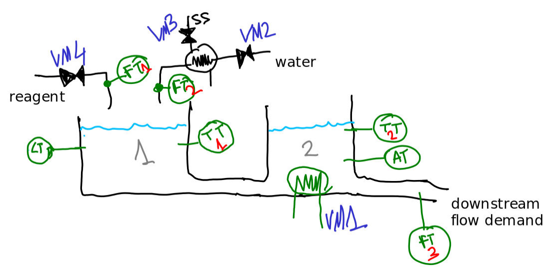
**We invert everything,*
we don't leave 5/36 linearised gain,
contrarily to other decoupling strategies trying to keep the multiloop tuning.

Multiloop, ratio control, decoupling: control of flow and reagent dilution (static mixing process)

Table of Contents

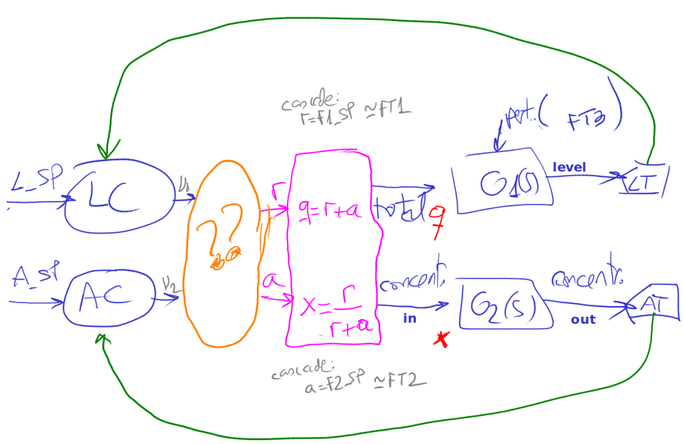
Objectives and motivation

Consider this mixing process (heating not considered), where nominally we must introduce 1 liter of reagent for every 5 liters of water to dilute it:



Multiloop: we must decide if we control level either with water flow or with reagent and, therefore, we control concentration with the other flow (VM4 or VM2). Temperature control not considered here.

More sophisticated structures: we could also think of ratio control, decoupling, or similar, to improve the performance of the basic "multiloop" strategy. Our objective is to analyze all this and the relationships between the various options. In short, the goal is to decide what to put in the "??" block below:



Non-linear model of static mixing process

```
syms r w q x real
```

Variable r represents reagent flow (input); Variable a in drawings represents water input flow to dilute it, here I named it "w".

```
Model=[q==r+w; x==r/(r+w)]
```

Model =

$$\begin{pmatrix} q = r + w \\ x = \dfrac{r}{r + w} \end{pmatrix}$$

The first equation of the model describes the total flow (which will influence flows/levels in tanks downstream); the second row is the reagent concentration entering the tanks, resulting from the mixture.

Linearization

```
J=simplify(jacobian(rhs(Model),[r,w])); %partial derivatives
```

- If you wish symbolic output for "theory", uncomment this:

```
%syms r_0 w_0 real %symbolic operating point
```

- If you wish a numerical example with operating point $q_0 = 6$, $x_0 = 1/6$, input flows will be 5 liters of water, 1 liter of reagent (per unit time):

```
r_0=1; w_0=5; %Numerical example
```

Computations yield

```
G=subs(J,{r,w},{r_0,w_0}) %Linearization at operating point
```

G =

$$\begin{pmatrix} 1 & 1 \\ \dfrac{5}{36} & -\dfrac{1}{36} \end{pmatrix}$$

Outputs of G are ***[increments of total inflow to tanks; increment of inflow concentration]***. Inputs to G are ***[increment of reagent inflow; increment of water inflow]***.

Multiloop Control : RGA pairing rule

```
rga = simplify(G.*inv(G')) %Relative gain array
```

rga =

$$\begin{pmatrix} \frac{1}{6} & \frac{5}{6} \\ \frac{5}{6} & \frac{1}{6} \end{pmatrix}$$

Recommended pairing is

- "incr. flow ← water" e "incr. concentration in ← reagent" if $w_0 > r_0$;
- "incr. flow ← reagent" e "incr. concentration in ← water" if $w_0 < r_0$;

That is, concentration with the "nominally small inflow" and "total flow" with "nominally large inflow".

```
MultiloopPairing=[2;1];
```

Decoupling

Inverted (reverse) Form (linear)

If we assume $w > r$ at nominal operating point, inverted-form decoupling yields:

```
D1=-G(1,1)/G(1,2)
```

D1 = -1

```
D2=simplify(-G(2,2)/G(2,1))
```

D2 =

$$\frac{1}{5}$$

Direct Form (linear)

We insert a matrix so that virtual inputs "increase q" and "increase x" translate to flow rates of reagent and water... The direct inversion of the linearised model will be:

```
Decoupler1=inv(G)
```

Decoupler1 =

$$\begin{pmatrix} \frac{1}{6} & 6 \\ \frac{5}{6} & -6 \end{pmatrix}$$

The inputs to this decoupler would be *"desired total flow rate increment"* and *"desired inflow mix concentration increment"*. The outputs, *reagent* flow and *water* flow.

```
G*Decoupler1
```

ans =

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

If we simply seek to insert a matrix such that the apparent matrix is the "chosen" pairing of G (to improve a pre-existing RGA-based controller performance, without redesigning it), we would have:

```
Decoupler2=simplify(inv(G)*[0 G(1,2);G(2,1) 0])
```

```
Decoupler2 =
```

$$\begin{pmatrix} \frac{5}{6} & \frac{1}{6} \\ -\frac{5}{6} & \frac{5}{6} \end{pmatrix}$$

Inputs to `Decoupler2` would be "multiloop concentration control action ν_2" y "multiloop level control action ν_1"; outputs would be "reagent flow" and "water flow".

```
G*Decoupler2
```

```
ans =
```

$$\begin{pmatrix} 0 & 1 \\ \frac{5}{36} & 0 \end{pmatrix}$$

The forward direct decoupling equivalent to the block diagram of decoupling in inverted form is:

```
Decoupler3=inv([1 -D2;-D1 1]) %coincides with Decoupler2
```

```
Decoupler3 =
```

$$\begin{pmatrix} \frac{5}{6} & \frac{1}{6} \\ -\frac{5}{6} & \frac{5}{6} \end{pmatrix}$$

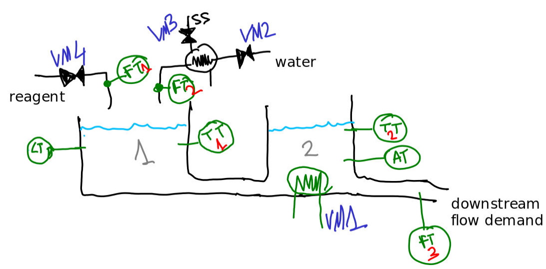
Multiloop, ratio control, decoupling: control of flow and reagent dilution (static mixing process)

PERMUTED FROM THAT IN VIDEO SO CHOSEN PAIRINGS IN DIAGONAL

Table of Contents

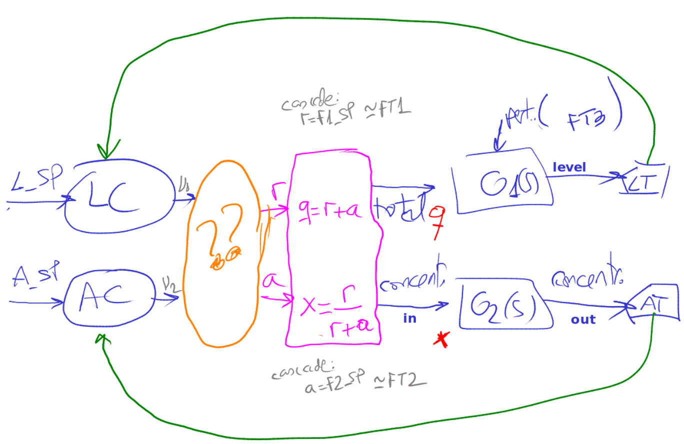
Objectives and motivation

Consider this mixing process (heating not considered), where nominally we must introduce 1 liter of reagent for every 5 liters of water to dilute it:



Multiloop: we must decide if we control level either with water flow or with reagent and, therefore, we control concentration with the other flow (VM4 or VM2). Temperature control not considered here.

More sophisticated structures: we could also think of ratio control, decoupling, or similar, to improve the performance of the basic "multiloop" strategy. Our objective is to analyze all this and the relationships between the various options. In short, the goal is to decide what to put in the "??" block below:



Non-linear model of static mixing process

```
syms r w q x real
```

Variable r represents reagent flow (input); Variable a in drawings represents water input flow to dilute it, here I named it "w".

```
Model=[q==r+w; x==r/(r+w)]
```

```
Model =
```

$$\begin{pmatrix} q = r + w \\ x = \frac{r}{r+w} \end{pmatrix}$$

The first equation of the model describes the total flow (which will influence flows/levels in tanks downstream); the second row is the reagent concentration entering the tanks, resulting from the mixture.

Linearization

```
J=simplify(jacobian(rhs(Model),[w,r])); %partial derivatives
```

- If you wish symbolic output for "theory", uncomment this:

```
%syms r_0 w_0 real %symbolic operating point
```

- If you wish a numerical example with operating point $q_0 = 6$, $x_0 = 1/6$, input flows will be 5 liters of water, 1 liter of reagent (per unit time):

```
r_0=1; w_0=5; %Numerical example
```

Computations yield

```
G=subs(J,{r,w},{r_0,w_0}) %Linearization at operating point
```

```
G =
```

$$\begin{pmatrix} 1 & 1 \\ -\frac{1}{36} & \frac{5}{36} \end{pmatrix}$$

Outputs of G are ***[increments of total inflow to tanks; increment of inflow concentration]***. Inputs to G are ***[increment of WATER inflow; increment of REAGENT inflow]***.

Multiloop Control : RGA pairing rule

```
rga = simplify(G.*inv(G')) %Relative gain array
```

```
rga =
```

$$\begin{pmatrix} \frac{5}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{5}{6} \end{pmatrix}$$

Recommended pairing is

- "incr. flow ← water" e "incr. concentration in ← reagent" if $w_0 > r_0$;
- "incr. flow ← reagent" e "incr. concentration in ← water" if $w_0 < r_0$;

That is, concentration with the "nominally small inflow" and "total flow" with "nominally large inflow".

Decoupling

Inverted (reverse) Form (linear)

If we assume $w > r$ at nominal operating point, inverted-form decoupling yields:

```
D1=-G(1,2)/G(1,1)
```

D1 = -1

```
D2=simplify(-G(2,1)/G(2,2))
```

D2 =

$$\frac{1}{5}$$

Direct Form (linear)

We insert a matrix so that virtual inputs "increase q" and "increase x" translate to flow rates of reagent and water... The direct inversion of the linearised model will be:

```
Decoupler1=inv(G)
```

Decoupler1 =

$$\begin{pmatrix} \frac{5}{6} & -6 \\ \frac{1}{6} & 6 \end{pmatrix}$$

The inputs to this decoupler would be *"desired total flow rate increment"* and *"desired inflow mix concentration increment"*. The outputs, *WATER* flow and *REAGENT* flow.

```
G*Decoupler1
```

ans =

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

If we simply seek to insert a matrix such that the apparent matrix is the "chosen" pairing of G (to improve a pre-existing RGA-based controller performance, without redesigning it), we would have:

```
Decoupler2=simplify(inv(G)*diag(diag(G)))
```

```
Decoupler2 =
```

$$\begin{pmatrix} \frac{5}{6} & -\frac{5}{6} \\ \frac{1}{6} & \frac{5}{6} \end{pmatrix}$$

Inputs to `Decoupler2` would be "multiloop level control action ν_1" y "multiloop concentration control action ν_2"; outputs would be "water flow" and "reagent flow". ROWS need to be swapped to be in accordance to the video's block diagram.

```
G*Decoupler2
```

```
ans =
```

$$\begin{pmatrix} 1 & 0 \\ 0 & \frac{5}{36} \end{pmatrix}$$

The forward direct decoupling equivalent to the block diagram of decoupling in inverted form is:

```
Decoupler3=inv([1 -D1;-D2 1]) %coincides with Decoupler2
```

```
Decoupler3 =
```

$$\begin{pmatrix} \frac{5}{6} & -\frac{5}{6} \\ \frac{1}{6} & \frac{5}{6} \end{pmatrix}$$