Advanced process control part 2

Sigurd Skogestad

NTNU

History of automatic process control

- 1900-1930: On/off control
  - Problem: Oscillates
- 1920s: P-control. Problem:
  - Problem: Bias
- 1930s: I-action
  - 1935: First commercial PI-controller (Foxboro)
  - 1939: First commercial PID (Taylor)
- 1940s: ARC: Cascade control, split range control
- 1960s: «Optimal control» and Kalman Filter
  - 1970s: Model predictive control
    - 1980: Commerical (DMC, Setpoint Co.)
- 1990s: Artifical intelligence for control
  - 2020s: second attempt (Machine learning)

ARC = Advanced regulatory control

QUIZ
What are the three most important inventions of process control?

- Hint 1: According to Sigurd Skogestad
- Hint 2: All were in use around 1940

SOLUTION

1. PID controller, in particular, I-action
2. Cascade control
3. Ratio control

How design standard ARC elements?

- Industrial literature (e.g., Shinskey).
  - Many nice ideas. But not systematic. Difficult to understand reasoning
- Academia: Very little work
  - I feel a little alone

Annual Reviews in Control 56 (2023) 100903

Contents lists available at ScienceDirect

Annual Reviews in Control

journal homepage: www.elsevier.com/locate/arcontrol

ELSEVIER

Review article

Advanced control using decomposition and simple elements

Sigurd Skogestad

Department of Chemical Engineering, Norwegian University of Science and Technology (NTNU), Trondheim, Norway

ARC: Standard Advanced control elements

Each element links a subset of inputs with a subset of outputs. Results in simple local design and tuning

First, there are some elements that are used to improve control for cases where simple feedback control is not sufficient:

E1*. Cascade control[2]
E2*. Ratio control
E3*. Valve (input)[3] position control (VPC) on extra MV to improve dynamic response.

Next, there are some control elements used for cases when we reach constraints:

E4*. Selective (limit, override) control (for output switching)
E5*. Split range control (for input switching)
E6*. Separate controllers (with different setpoints) as an alternative to split range control (E5)
E7*. VPC as an alternative to split range control (E5)

All the above seven elements have feedback control as a main feature and are usually based on PID controllers. Ratio control seems to be an exception, but the desired ratio setpoint is usually set by an outer feedback controller. There are also several features that may be added to the standard PID controller, including

E8*. Anti-windup scheme for the integral mode
E9*. Two-degrees of freedom features (e.g., no derivative action on setpoint, setpoint filter)
E10. Gain scheduling (Controller tunings change as a given function of the scheduling variable, e.g., a disturbance, process input, process output, setpoint or control error)

In addition, the following more general model-based elements are in common use:

E11*. Feedforward control
E12*. Decoupling elements (usually designed using feedforward thinking)
E13. Linearization elements
E14*. Calculation blocks (including nonlinear feedforward and decoupling)
E15. Simple static estimators (also known as inferential elements or soft sensors)

Finally, there are a number of simpler standard elements that may be used independently or as part of other elements, such as

E16. Simple nonlinear static elements (like multiplication, division, square root, dead zone, dead band, limiter (saturation element), on/off)
E17*. Simple linear dynamic elements (like lead–lag filter, time delay, etc.)
E18. Standard logic elements

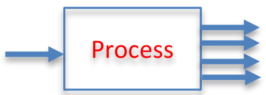
[2] The control elements with an asterisk * are discussed in more detail in this paper.

ARC = advanced reguklatory control

Sigurd Skogestad, "Advanced control using decomposition and simple elements". Annual Reviews in Control, vol. 56 (2023), Article 100903 (44 pages).

Constraint switching
(because it is optimal at steady state)

- CV-CV switching
  - Control one CV at a time



- MV-MV switching
  - Use one MV at a time

- MV-CV switching
  - MV saturates so must give up CV
  1. Simple («do nothing»)
  2. Complex (repairing of loops)

CV-CV switching: Use selectors (E4) (only* option!)

- Many CVs paired with one MV.
- But only one CV controlled at a time.
- Use: Max or Min selector

> = MAX = HS

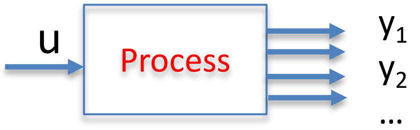
< = MIN = LS

Note: Selectors are logic blocks

- Sometimes called “override”
  - But this term may be misleading
- Selector is generally on MV (compare output from many controllers)

*Not quite true: Selectors may be implemented in other ways, for example, using «if-then»-logic.

Implementation selector



Alt. I (General). Several controllers (different CVs)

- Selector on MVs
  - Must have anti windup for C_1 and C_2 !

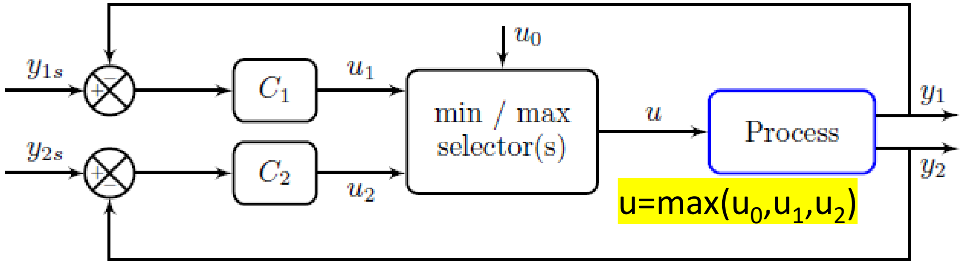
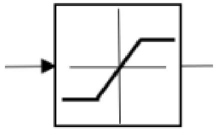


Figure 17: CV-CV switching with selector on MV (input u).

Alt. II (Less general) Controllers in cascade

- Selector on CV setpoint
- Good alternative if CVs (y_1 and y_2) are related so that cascade is good
- In this case: Selector may be replaced by saturation element (with y_{2s} as the max or min)



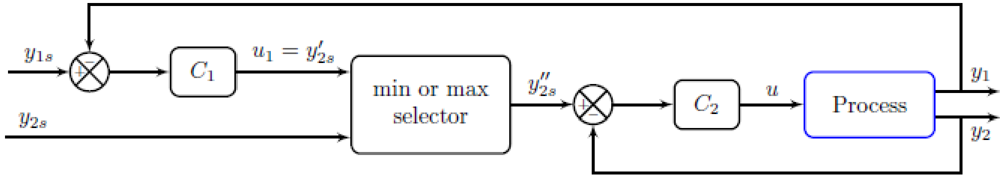
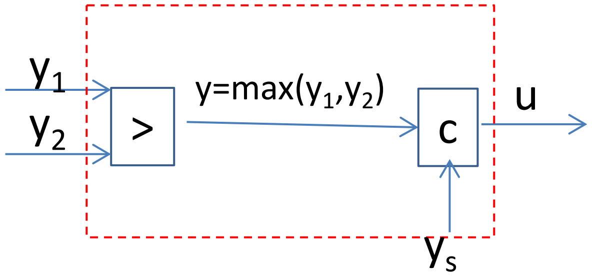


Figure 19: Alternative cascade CV-CV switching implementation with selector on the setpoint. In many cases, y_{1s} and y_{2s} are constraint limits.

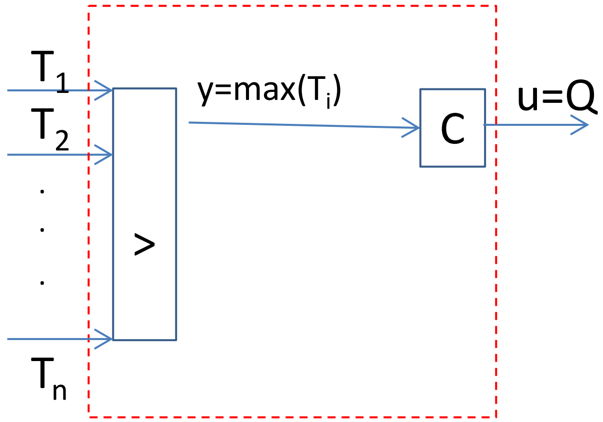
Alt. III (For special case where all CVs have same bound). One controller

- Selector is on CVs (Auctioneering)
- Also assumes that dynamics from u to y_1 and y_2 are similar; otherwise use Alt.I
- Example: Control hot-spot in reactor or furnace.



Example Alt. III

- Hot-spot control in reactor or furnace



- Comment: Could use General Alternative I (many controllers) for hot-spot control, with each temperature controller (c_1, c_2,…) computing the heat input (u_1=Q_1, u_2=Q_2, ….) and then select u = **min**(u_1, u_2, …), but it is more complicated.

Furnace control with safety constraint (Alt. I)

Input (MV)

u = Fuel gas flowrate

Output (CV)

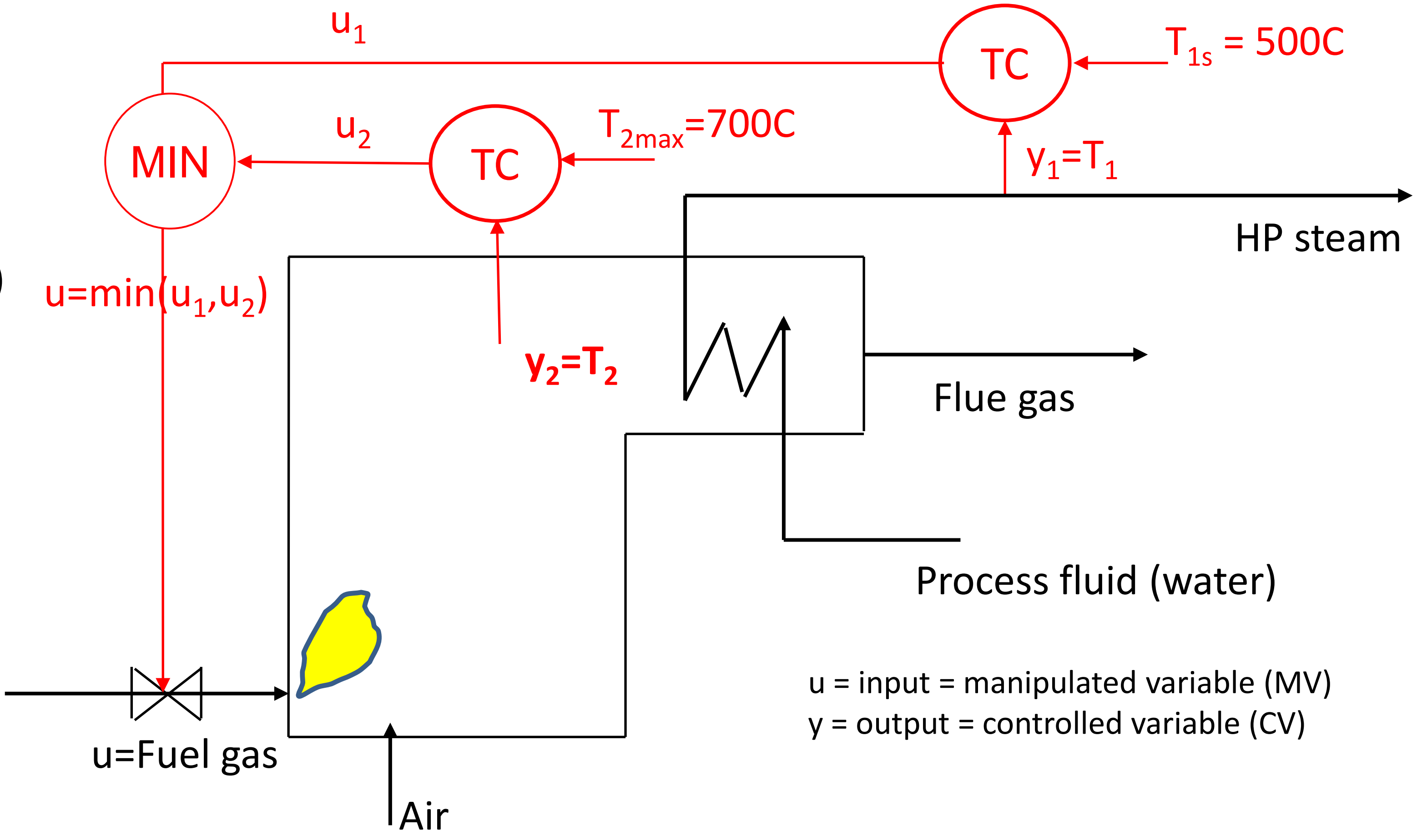
y_1 = process temperature T_1

(desired setpoint or max constraint)

y_2 = furnace temperature T_2

(max constraint)

Rule: Use min-selector for constraints that are satisfied with a small input

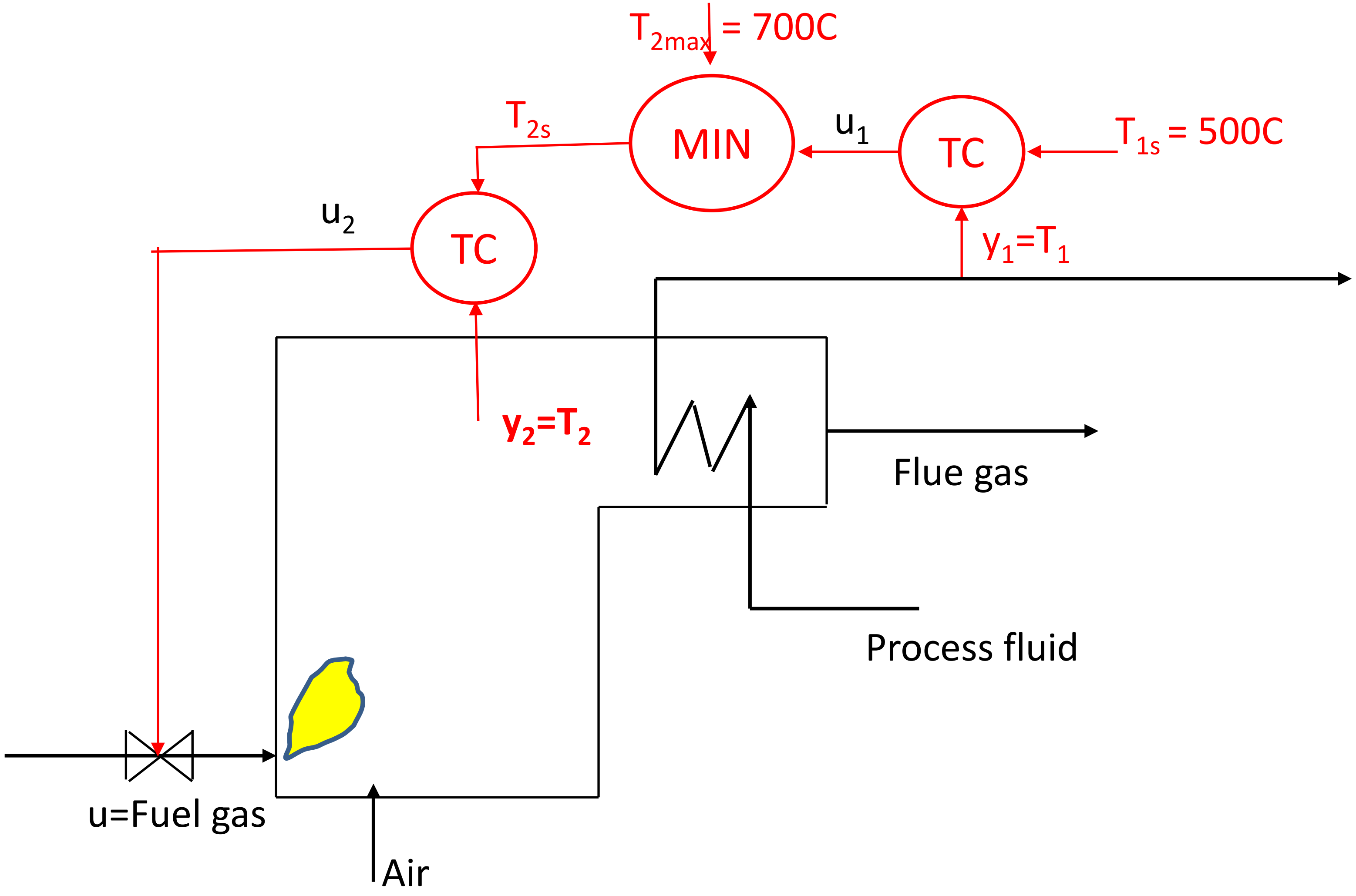


u = input = manipulated variable (MV)

y = output = controlled variable (CV)

Furnace control with cascade (Alt. II, selector on CV-sp)

Comparison
The cascade solution is less general but it may be better in this case.
Why better? Inner T2-loop is fast and always active and may improve control of T1.



Comment: For both Alt. I and II, we loose control of T1 (it drops below T1s=500C) when T2=T2max. If this is not acceptable then we need to something- More on this later!

Design of selector structure

Rule 1 (max or min selector)

- Use max-selector for constraints that are satisfied with a large input
- Use min-selector for constraints that are satisfied with a small input

Rule 2 (order of max and min selectors):

- If need both max and min selector: Potential infeasibility (conflict)
- Order does not matter if problem is feasible
- If infeasible: Put highest priority constraint at the end

"Systematic design of active constraint switching using selectors." Dinesh Krishnamoorthy , Sigurd Skogestad. Computers & Chemical Engineering, Volume 143, (2020)
"Advanced control using decomposition and simple elements". Sigurd Skogestad. Annual Reviews in Control, Volume 56, 100903 (2023)

Example. Maximize flow with pressure constraints

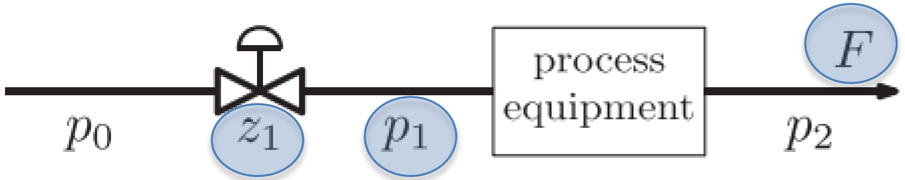


Fig. 6. Example 2: Flow through a pipe with one MV ($u = z_1$).

Input u = z_1
Want to maximize flow, J=-F:

Optimization problem is:

$$\max_{z_1} F$$

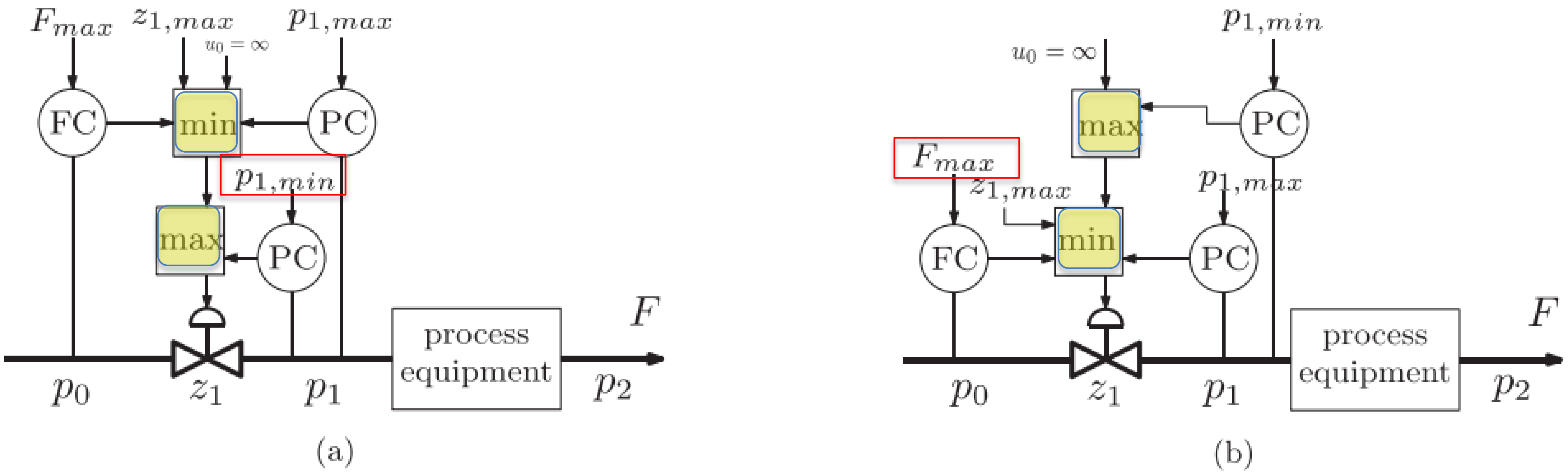
s.t.

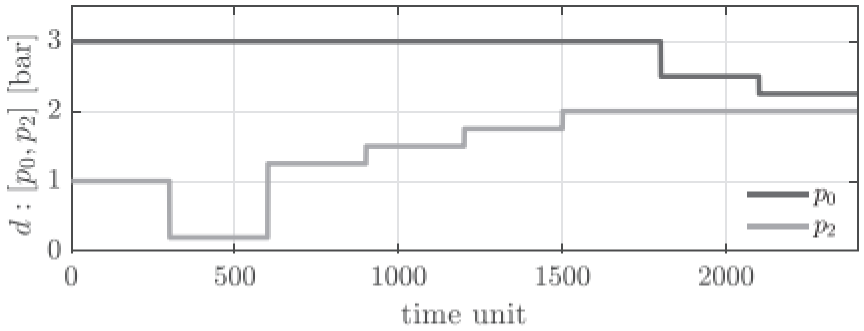
$$\boxed{\begin{array}{l} F \leqslant F_{max} \\ p_1 \leqslant p_{1,max} \\ p_1 \geqslant p_{1,min} \\ z_1 \leqslant z_{1,max} \end{array}} \qquad (15)$$

where F_{max} =10 kg/s, $z_{1,max} = 1$, $p_{1,max} = 2.5$ bar, and $p_{1,min} = 1.5$ bar. Note that there are both max and min- constraints on p_1. De-

The two p1-constraints are not conflicting, because they are on the same variable.
However the Fmax-constraint and p1min-constraint may be conflicting: Must choose which is most important.

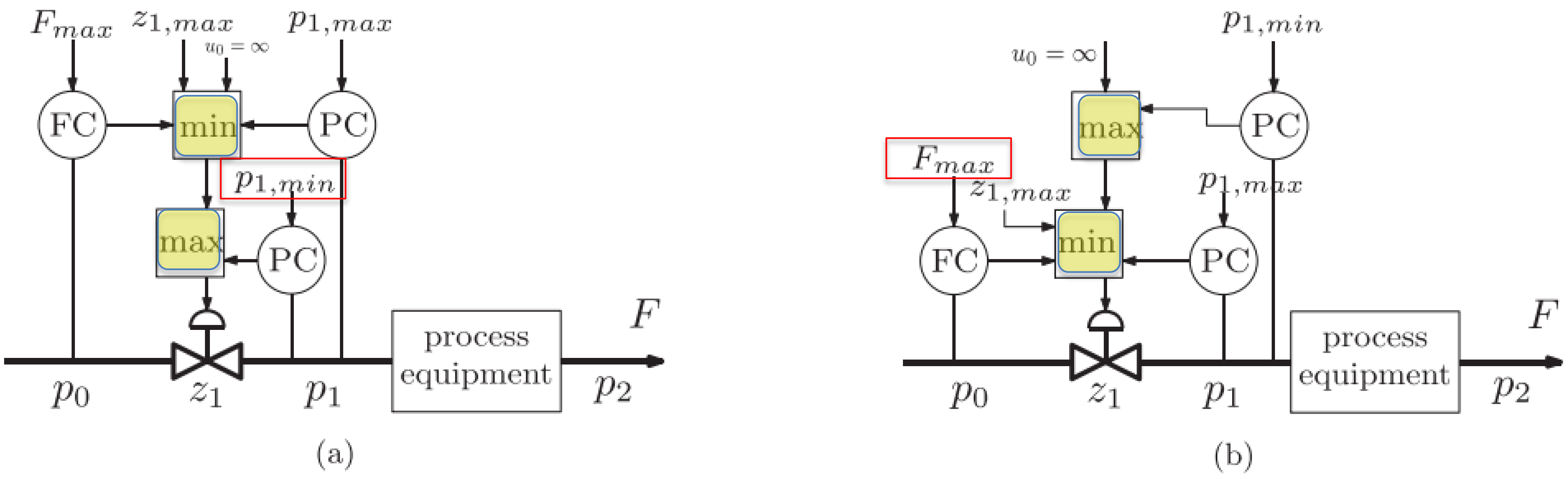
CV-CV switching

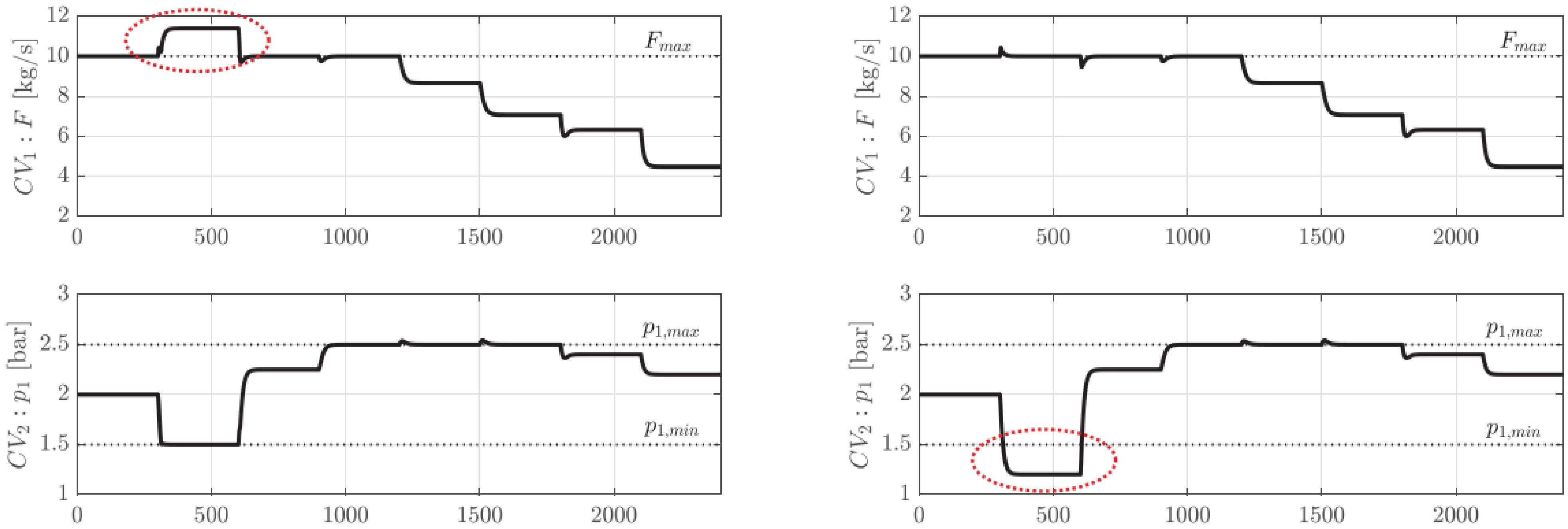




Disturbances in p_0 and p_2 (unmeasured)

CV-CV switching



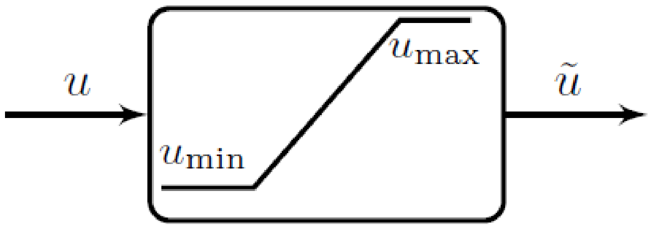


t>1800: u=zmax=1

Valves have "built-in" selectors

Rule 3 (a bit opposite of what you may guess)

- A closed valve (u_{min}=0) gives a "built-in" max-selector (to avoid negative flow)
- An open valve (u_{max}=1) gives a "built-in" min-selector
  - So: Not necessary to add these as selector blocks (but it will not be wrong).
  - The "built-in" selectors are never conflicting because cannot have closed and open at the same time
  - Another way to see this is to note that a valve works as a saturation element



Saturation element may be implemented in three other ways (equivalent because never conflict)

1. Min-selector followed by max-selector
2. Max-selector followed by min-selector
3. Mid-selector

$$\tilde{u} = \max(u_{min}, \min(u_{max}, u)) = \min(u_{max}, \max(u_{min}, u)) = \text{mid}(u_{min}, u, u_{max})$$

"Advanced control using decomposition and simple elements". Sigurd Skogestad. Annual Reviews in Control, Volume 56, 100903 (2023)

MV-MV switching

- Need several MVs to cover whole <u>steady-state</u> range (because primary MV may saturate)*
- Note that we only want to use one MV at the time.

Three solutions:

Alt.1 Split-range control (one controller) (E5)

Alt.2 Several controllers with different setpoints (E6)

Alt.3 Valve position control (E7)

Which is best? It depends on the case!

***Optimal Operation with Changing Active Constraint Regions using Classical Advanced Control, Adriana Reyes-Lua Cristina Zotica, Sigurd Skogestad, Adchem Conference, Shenyang, China. July 2018 ,**

Example MV-MV switching

- Break and gas pedal in a car
- Use only one at a time

E5. Split-range control (SRC) (for MV-MV switching)

[7]Note the blue saturation elements for the inputs in Figure 21 and other block diagrams. Saturation can occur for any physical input, but they are explicitly shown for cases where the saturation is either the reason for or part of the control logic. For example, in Figure 21, the reason for using u_2 is that u_1 may saturate.

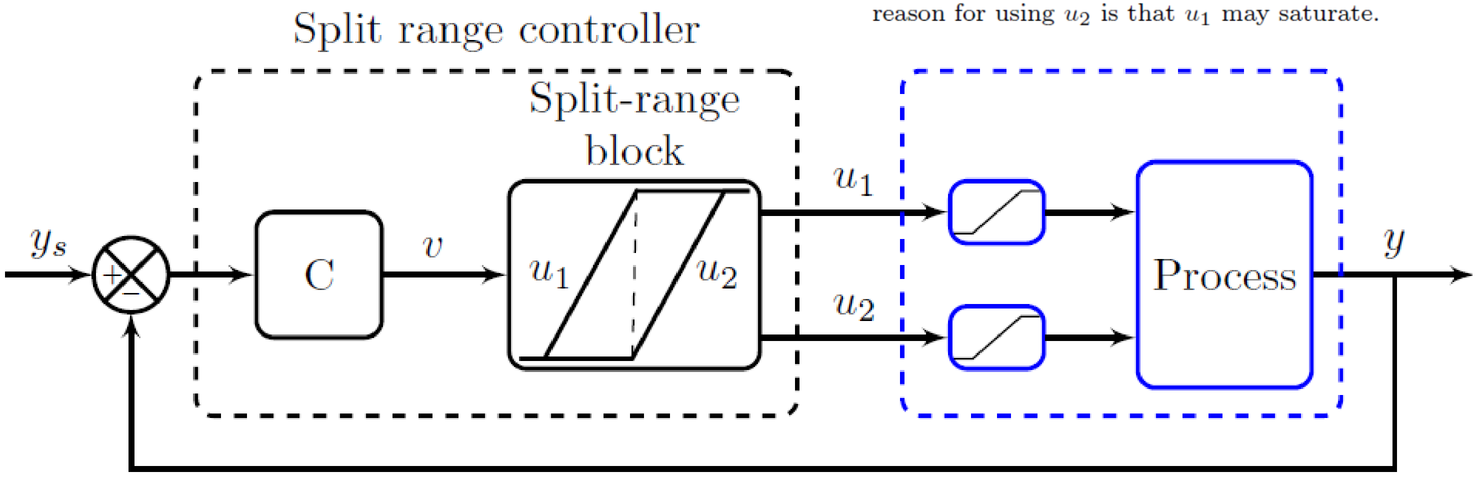


Figure 21: Split range control for MV-MV switching.

For MVs (u) that have same effect (same sign) on the output (y) (Fig. 21), we need to define the order in which the MVs will be used. This is done by the order in in the SR-block.

Example: With two heating sources, we need to decide which to use first (see next Example)

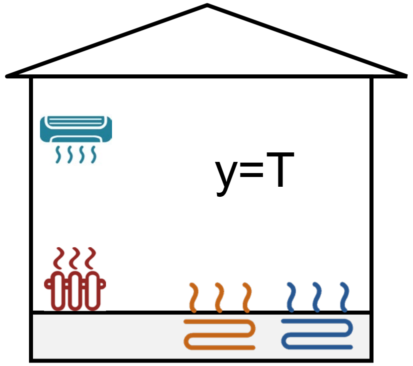
Advantage: SRC is easy to understand and implement!

Disadvantages:

1. Only one controller C ⇒ Same integral time for all inputs u_i (MVs)
   – Controller gains can be adjusted with slopes in SR-block!
2. Does not work well for cases where constraint values for u_i change

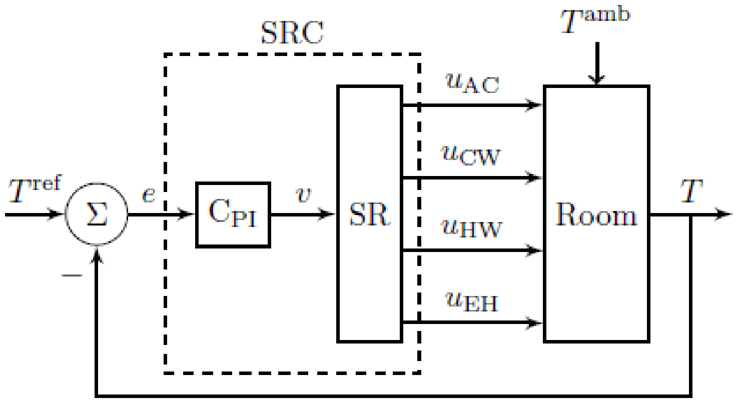
Alt. 1 Split-range control (SRC)

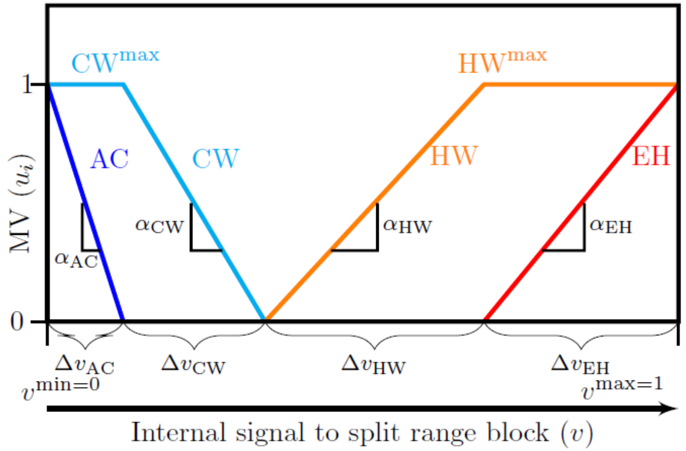
Example: Room heating with one CV (T) and 4 MVs



MVs (two for summer and two for winter):

1. AC (expensive cooling)
2. CW (cooling water, cheap)
3. HW (hot water, quite cheap)
4. Electric heat, EH (expensive)



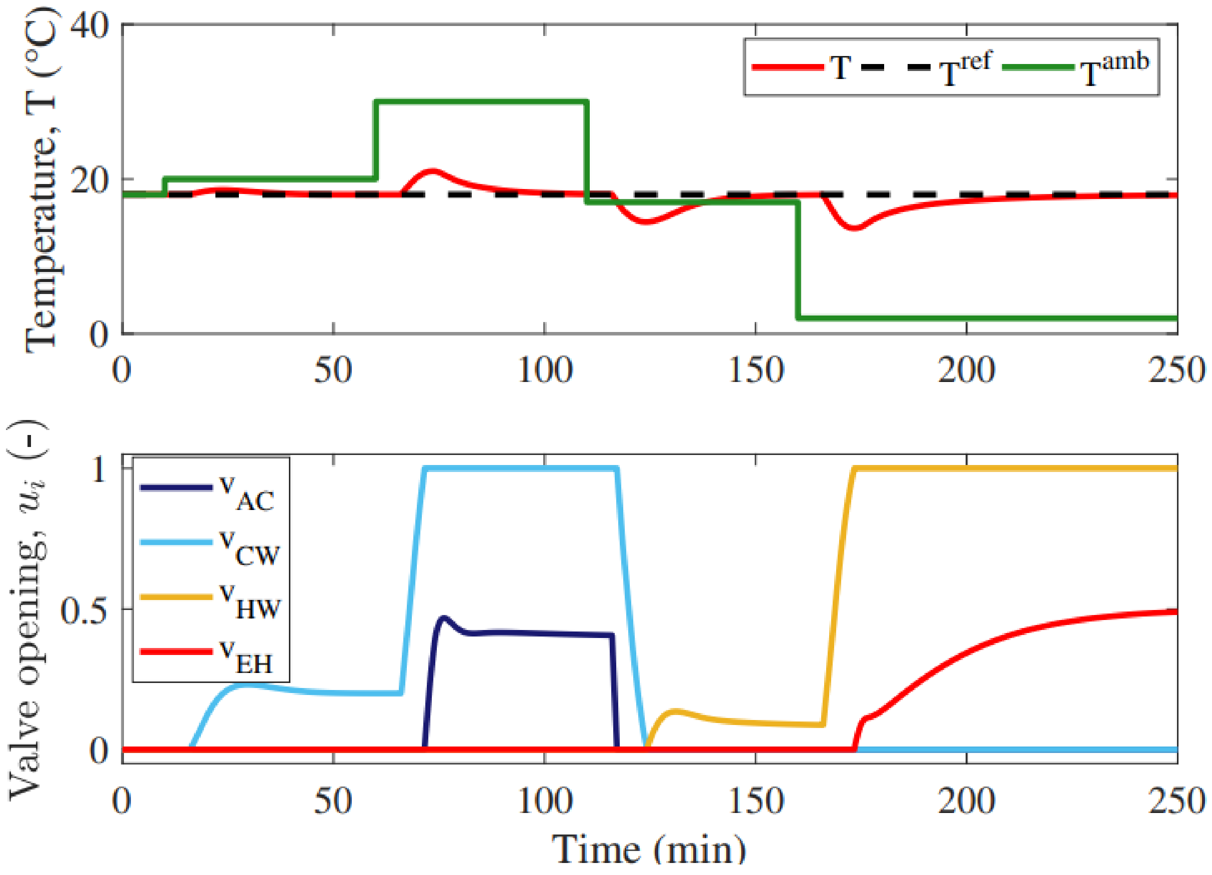


Note: may adjust the location of split (x-axis) to make loop gains equal.

Disadvantage SRC: 1. Must use same integral time for all MVs
2. Does not work well for cases where constraint values change

MV-MV switching

Alt. 1 Simulation Split-range control (SRC).



MV-MV switching- Alternative 2.

E6. Separate controllers with different setpoints

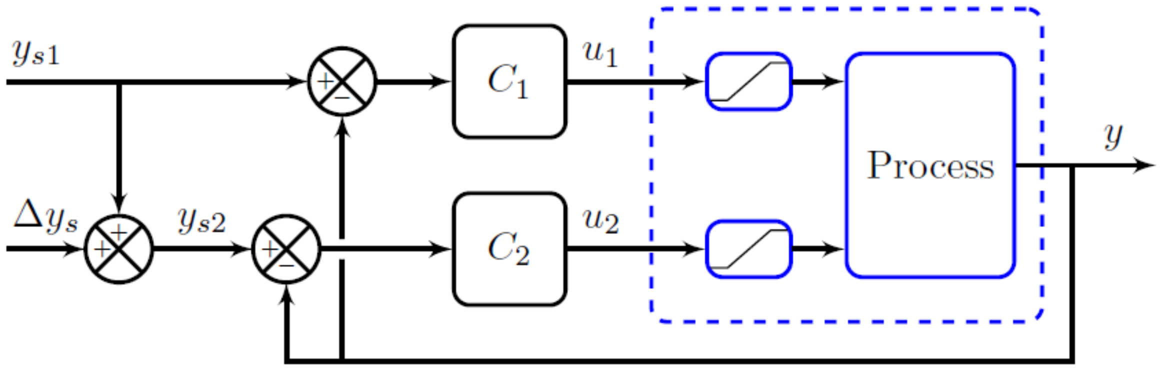


Figure 22: Separate controllers with different setpoints for MV-MV switching.

The setpoints $(y_{s1}, y_{s2}, \dots)$ should in the same order as we want to use the MVs. The setpoint differences (e.g., $\Delta y_s = y_{s2} - y_{s1}$ in Fig. 22) should be large enough so that, in spite of disturbances and measurement noise for y, only one controller (and its associated MV) is active at a given time (with the other MVs at their relevant limits).

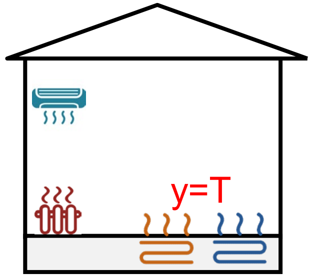
Advantages E6 (compared to split range control, E5):

1. Simple to implement (no logic)
2. Controllers can be tuned independently (different integral times)
3. Switching by feedback: Do not need to know constraint values
   - Big advantage when switching point varies (complex MV-CV switching)

Disadvantages:

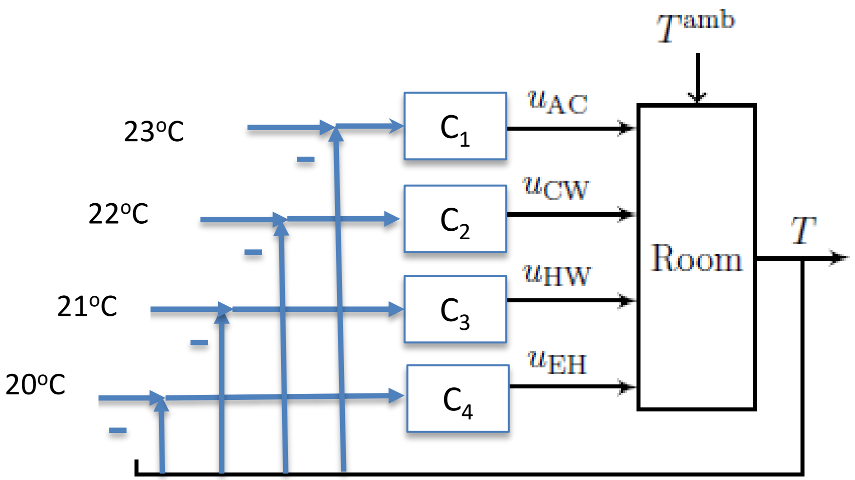
1. Temporary loose control during switching
2. Setpoint not constant
   - Can be an advantage (for example, may give energy savings for room heating)

Example: Room heating with one CV (T) and 4 MVs



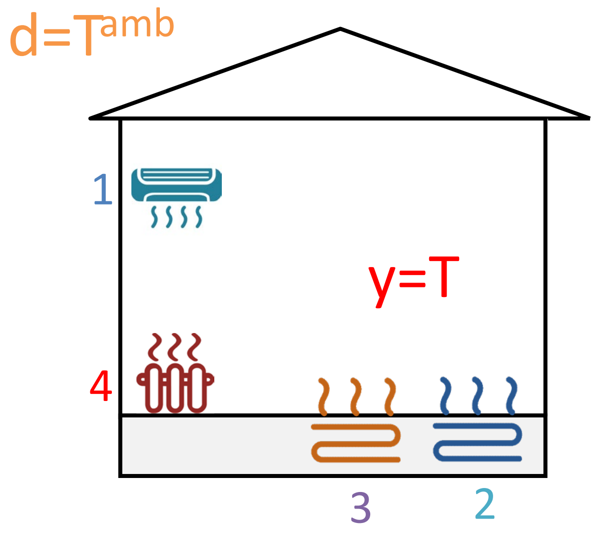
MVs (two for summer and two for winter):
1. AC (expensive cooling)
2. CW (cooling water, cheap)
3. HW (hot water, quite cheap)
4. Electric heat, EH (expensive)

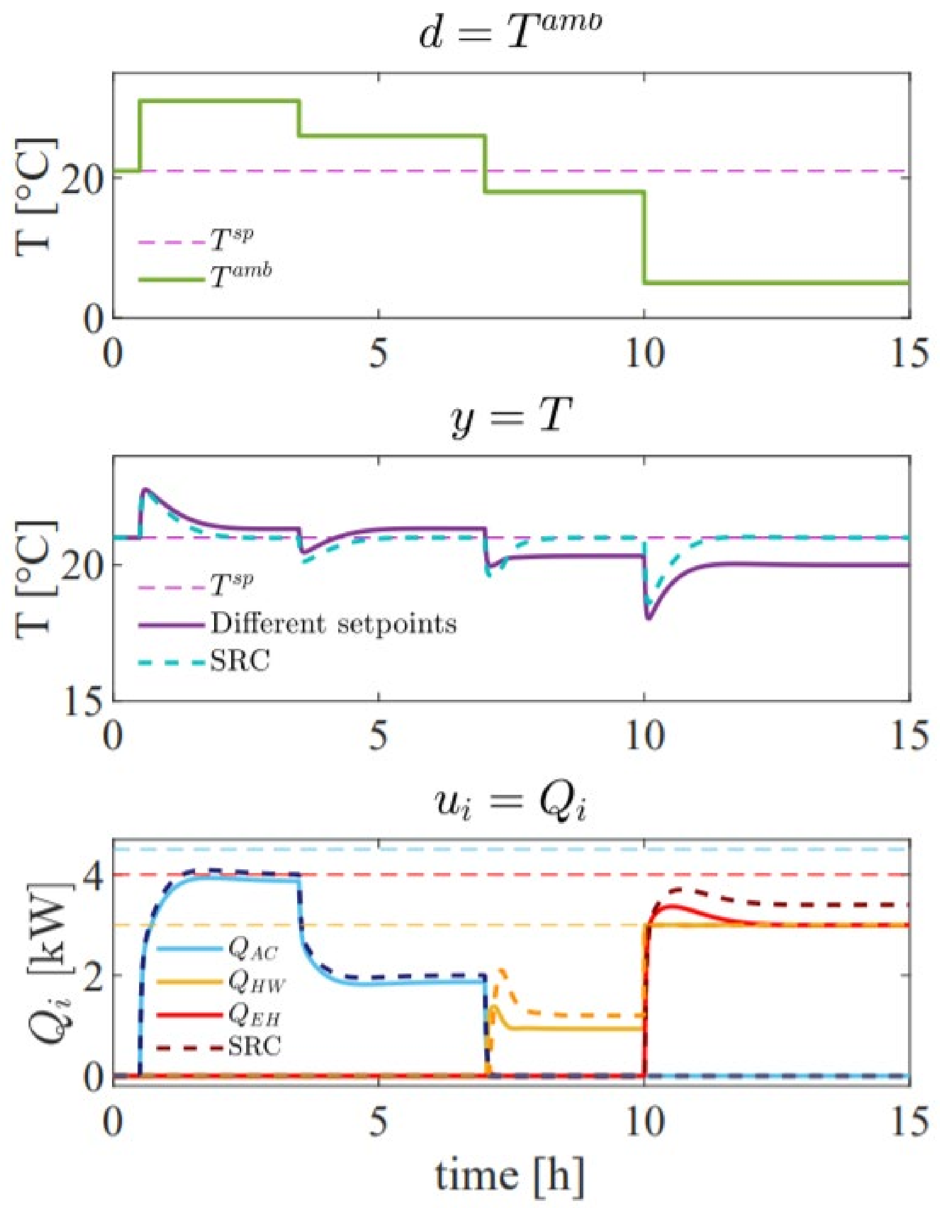
Alt. 2 for MV-MV switching. Multiple controllers with different setpoints



Disadvantage (comfort):
- Different setpoints
- Loose control during transition

Advantage (economics) :
- Different setpoints (energy savings)





A Reyes-Lua, S Skogestad. Multiple-Input Single-Output Control for Extending the Steady-State Operating Range - Use of Controllers with Different Setpoints. Processes 7 (12), 941

MV-MV switching- Alternative 3.

E7. VPC on main steady-state input

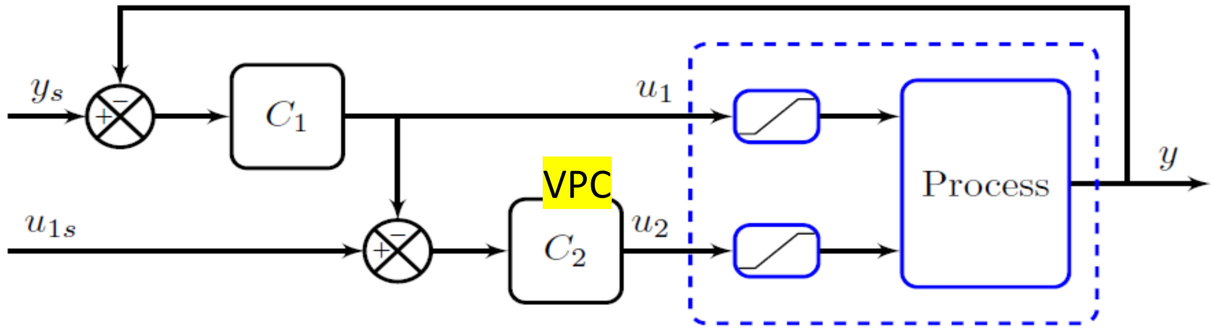


Figure 24: Valve (input) position control for MV-MV switching. A typical example is when u_2 is needed only in fairly rare cases to avoid that u_1 saturates.

Advantages E7 (for MV-MV switching): Always use u_1 to control y

- For example, u_2 may only allow discrete changes (e.g., u_2=0,1,2,3)
- or dynamics for u_2 may be very slow

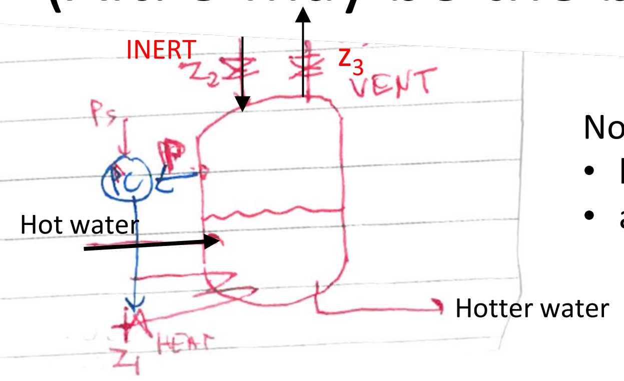
Disadvantages E7:

1. We cannot let u_1 become fully saturated because then control of y is lost
   - This means that we cannot use the full range for u_1 (potential economic loss)
2. Related: When u_2 is used, we need to keep using a ''little'' of u_1.
   - Example: May need to use both heating and cooling at the same time (when u1 normally should be off).

Example MV-MV switching: Pressure control

(Alt. 3 may be the best in this case)

CV=p
MV1=heat (Q)
MV2=inert
MV3=vent



Normal: Control CV=p using MV1=Q = z1
- but if Q=0 (because of too hot feed) we must use MV3=vent =z3
- and if Q=max (becase of too cold feed) we must use MV2=inert = z2

Example: Heating water to 213C = Control steam pressure at 20 bar*.
- «Inert» (z2) could be HP steam.

* Rule of thumb: p[bar] = [T[C]/100]^4

Example MV-MV switching: Pressure control

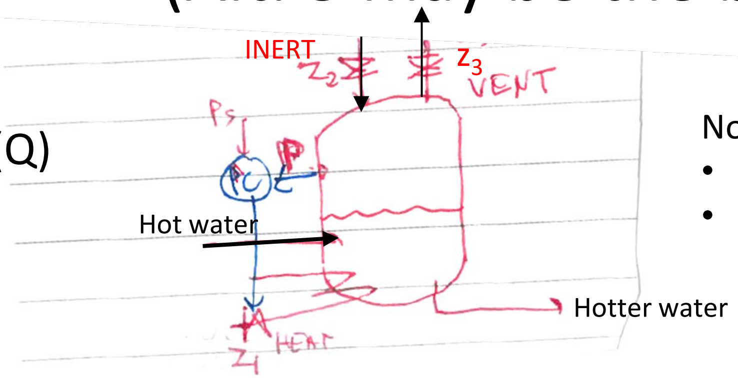
(Alt. 3 may be the best in this case)

CV=p

MV1=heat (Q)

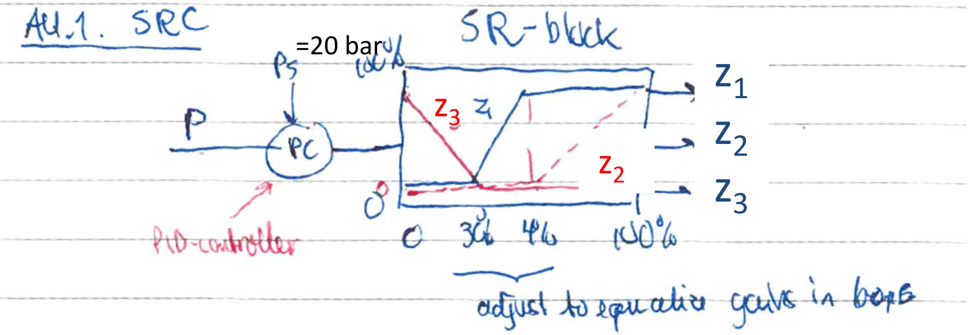
MV2=inert

MV3=vent



Normal: Control CV=p using MV1=Q

- but if Q=0 we must use MV3=vent
- and if Q=max we must use MV2=inert



Example MV-MV switching: Pressure control

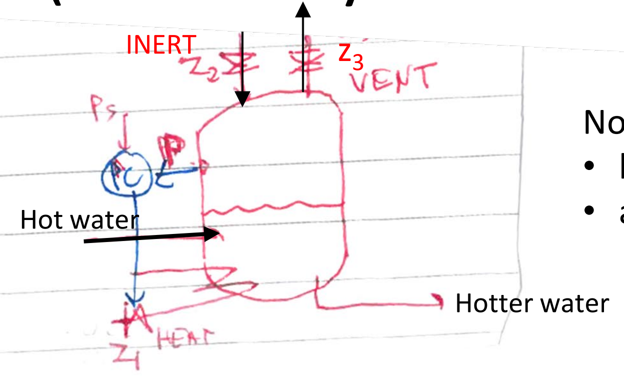
(Alt. 3 may be the best in this case)

CV=p

MV1=heat (Q)

MV2=inert

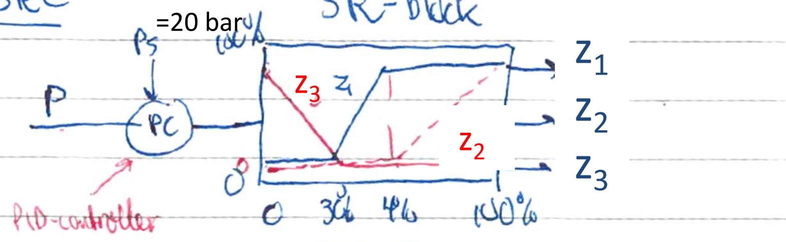
MV3=vent



Normal: Control CV=p using MV1=Q

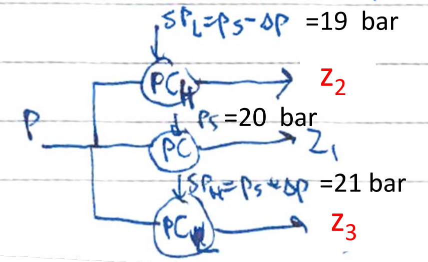
- but if Q=0 we must use MV3=vent
- and if Q=max we must use MV2=inert

Alt.1. SRC



adjust to equalize gains in loops

Alt.2. Three controllers with different setpoints.



Example MV-MV switching: Pressure control

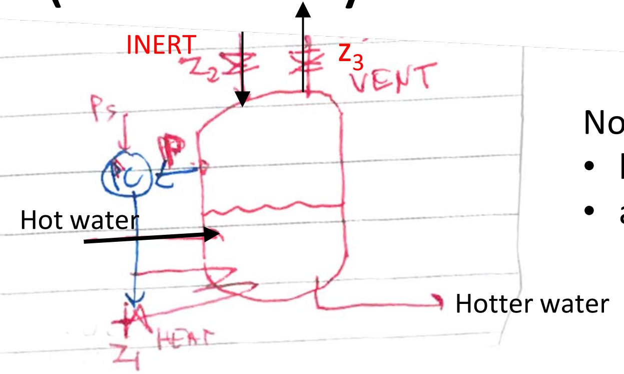
(Alt. 3 may be the best in this case)

CV=p

MV1=heat (Q)

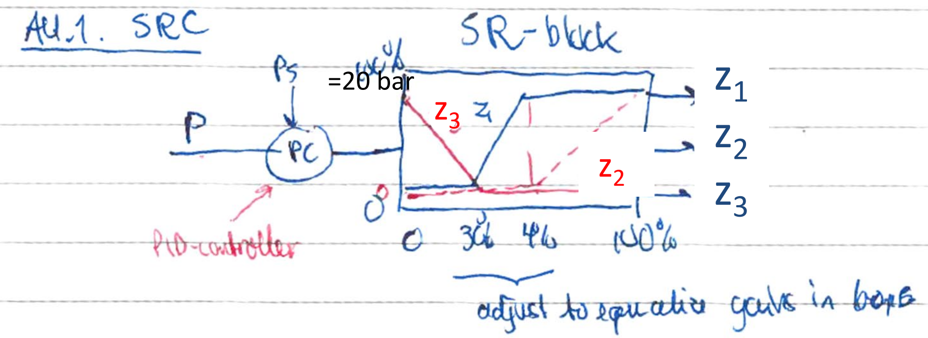
MV2=inert

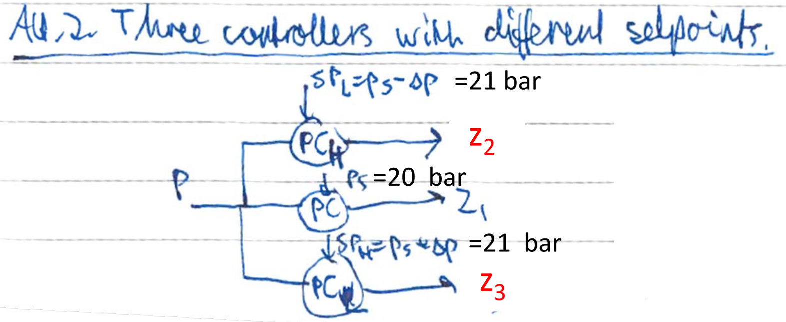
MV3=vent



Normal: Control CV=p using MV1=Q

- but if Q=0 we must use MV3=vent
- and if Q=max we must use MV2=inert



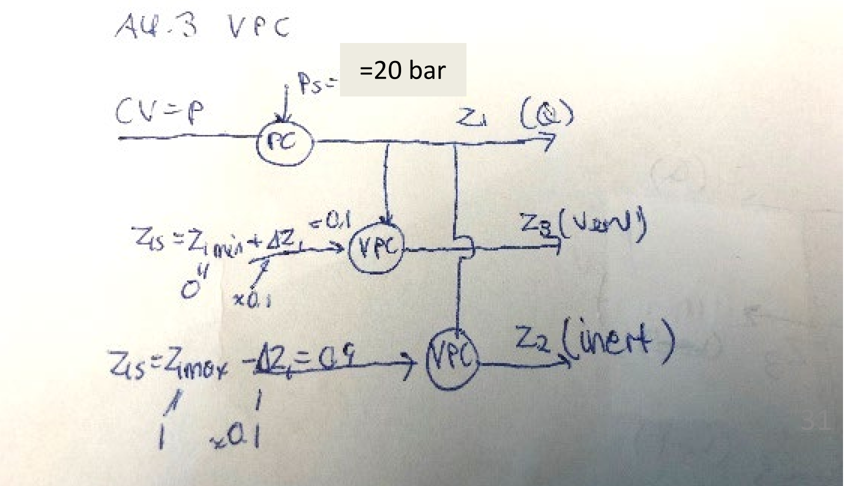


Alt.3: VPC (z2 and z3 could here even be on/off valves)

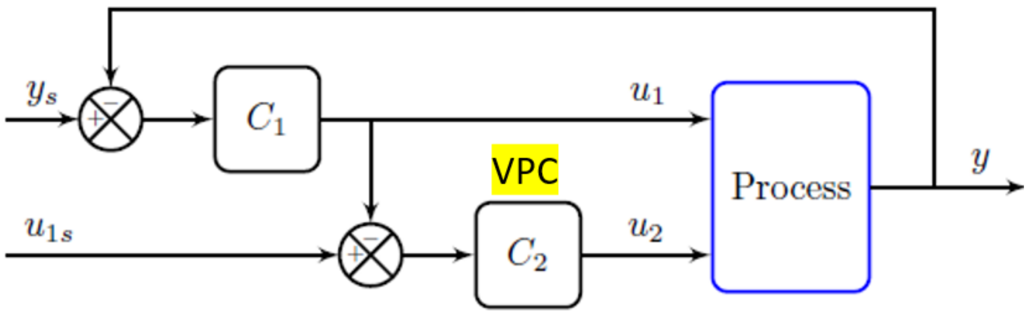
Always use Q (z1) to control p.

Need two VPC's:

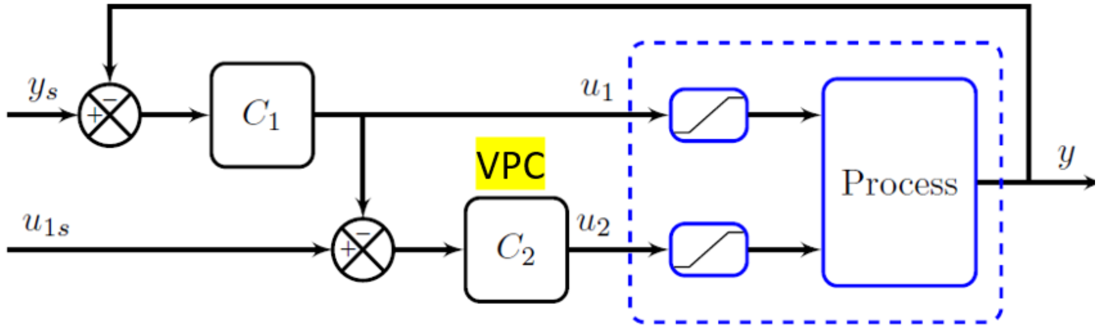
- Use vent (z3) to avoid Q small (z1=0.1)
- Use inert (z2) to avoid Q large (z1=0.9)
- z2=0 and z3=0 when 0.1<z1<0.9



Beware: Two different applications of VPC (E3 and E7)



E3 (Fig. 12)



E7 (Fig. 24)

The VPC schemes in Figure 12 (E3 - VPC on dynamic input) and Figure 24 (E7) seem to be the same
But their behavior is very different!

- In Figure 12 (E3) both inputs are used all the time
  - u_1 is used to improve the dynamic response
- In Figure 24 (E7)
  - u_1 is the main input (and used all the time)
  - u_2 is only used when u_1 approaches saturation (for MV-MV switching)

Summary MV-MV switching

- Need several MVs to cover whole steady-state range (because primary MV may saturate)*
- Note that we only want to use one MV at the time.

Alt.1 Split-range control (one controller) (E5)

- Advantage: Easy to understand because SR-block shows clearly sequence of MVs
- Disdvantages: (1) Need same tunings (integral time) for all MVs . (2) May not work well if MV-limits inside SR-block change with time, so: Not good for MV-CV switching

Alt.2 Several controllers with different setpoints (E6)

- Advantages: 1. Simple to implement, do not need to keep track of MVs. 2. Can have independent tunings. .
- Disadvantage: Setpoint varies (which can be turned into an advantage in some cases)

Alt.3 Valve position control (E7)

- Advantage: Always use “primary” MV for control of CV (avoids repairing of loops)
- Disadvantages: Gives some loss, because primary MV always must be used (cannot go to zero).

Which is best? It depends on the case!

***Optimal Operation with Changing Active Constraint Regions using Classical Advanced Control, Adriana Reyes-Lua Cristina Zotica, Sigurd Skogestad, Adchem Conference, Shenyang, China. July 2018 ,**

MV-CV switching (because reach constraint on MV)

- **Simple CV-MV switching**
  - Don't need to do anything if we followed the *Input saturation rule:*
  - *"Pair a MV that may saturate with a CV that can be given up (when the MV saturates)"*

Example: Avoid freezing in cabin

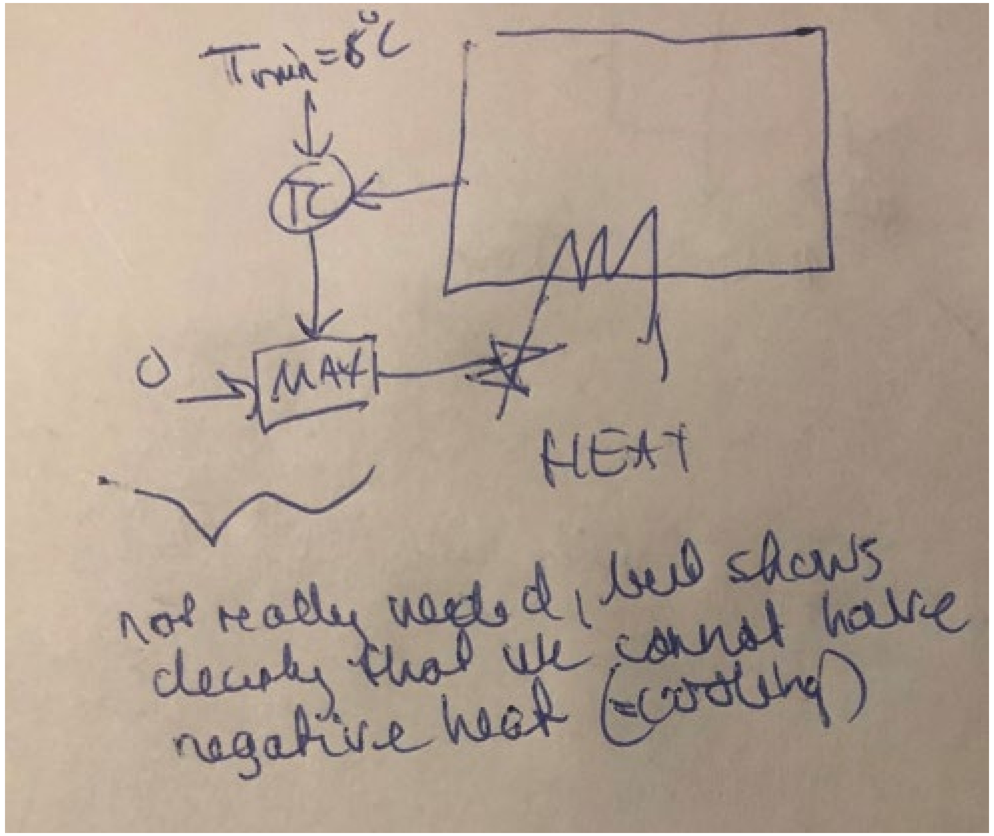
Minimize u (heating), subject to

$$T \geq T_{min}$$

$$u \geq 0$$

Keep CV=T>T_{min} = 8C in cabin in winter by using MV=heating

If it's hot outside (>8C), then the heat will go to zero (MV=Q=0), but this does not matter as the constraint is over-satisfied.



Example: Anti-surge control (= min-constraint on F)

Minimize u (recycle), subject to

$$F \geq F_{min}$$
$$u = z \geq 0$$

Keep minimum flow F_{min} for pump or compressor using recycle valve.

If the flow F_0 (and thus F) becomes large then the recycle valve will close (MV=0), but this does not matter as the constraint on F is over-satisfied.

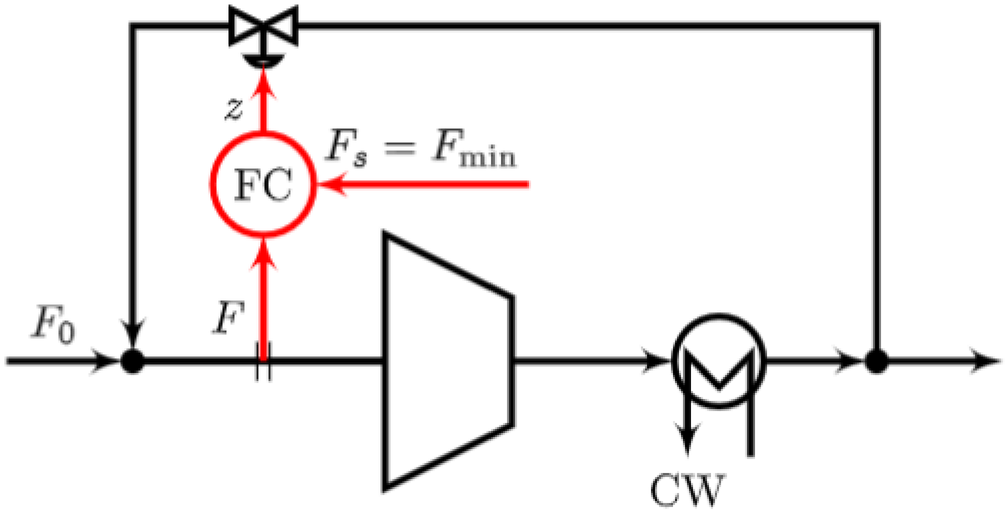
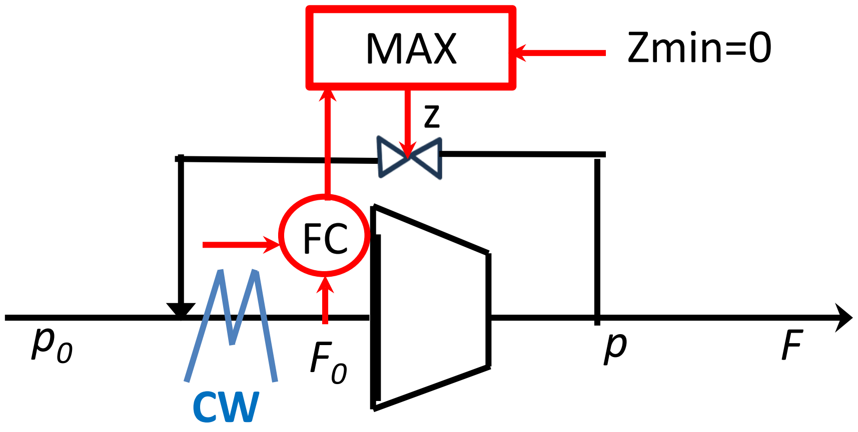


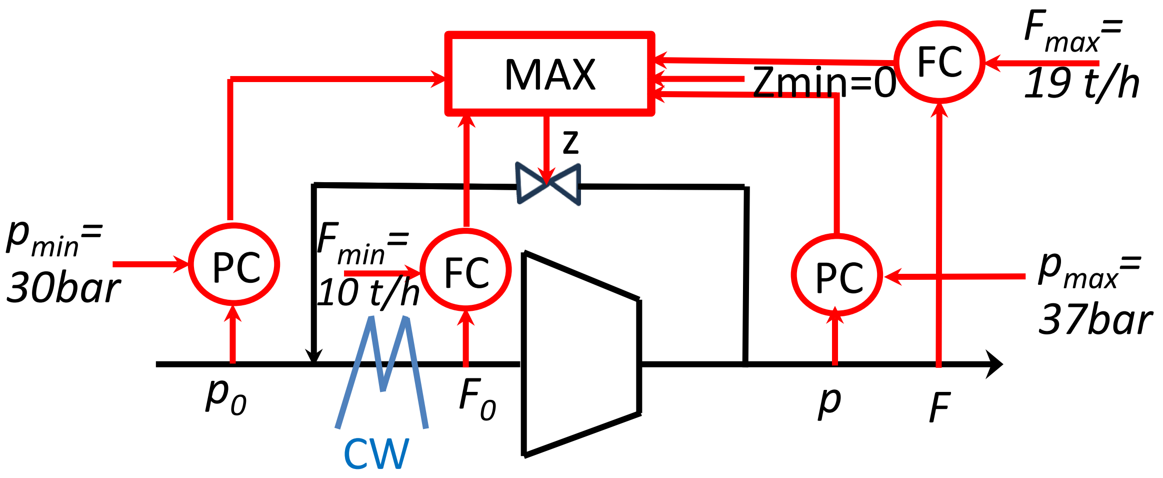
Fig. 32. Flowsheet of anti-surge control of compressor or pump (CW = cooling water). This is an example of simple MV-CV switching: When MV=z (valve position) reaches its minimum constraint ($z = 0$) we can stop controlling CV=F at $F_s = F_{min}$, that is, we do not need to do anything except for adding anti-windup to the controller. Note that the valve has a "built in" max selector.

We satisfy the input saturation rule:
«Pair a MV =z that may saturate with a CV =F that can be given up (when the MV saturates at z=0)"

QUIZ Compressor control

SOLUTION





Suggest a solution which achieves

- *$p < p_{max} = 37$ bar (max delivery pressure)*
- *$P_0 > p_{min} = 30$ bar (min. suction pressure)*
- *$F < F_{max} = 19$ t/h (max. production rate)*
- *$F_0 > F_{min} = 10$ t/h (min. through compressor to avoid surge)*

All these 4 constraints are satisfied by a large z
-> MAX-selector

MV-CV switching (because reach constraint on MV)

Simple CV-MV switching

- Don't need to do anything if we followed the *Input saturation rule:*
- *"Pair a MV that may saturate with a CV that can be given up (when the MV saturates)"*

Complex MV-CV switching

- Didn't follow input saturation rule
- This is a repairing of loops
- Need to combine MV-MV switching with CV-CV-switching
  - The CV-CV switching always uses a selector
  - As usual, there are three alternatives for the MV-MV switching:
    1. Split range control (block /\): Has problems because limits may change
    2. Several controllers with different setpoints (often the best for MV-CV switching)
    3. Valve position control (Gives «long loop» but avoids repairing).

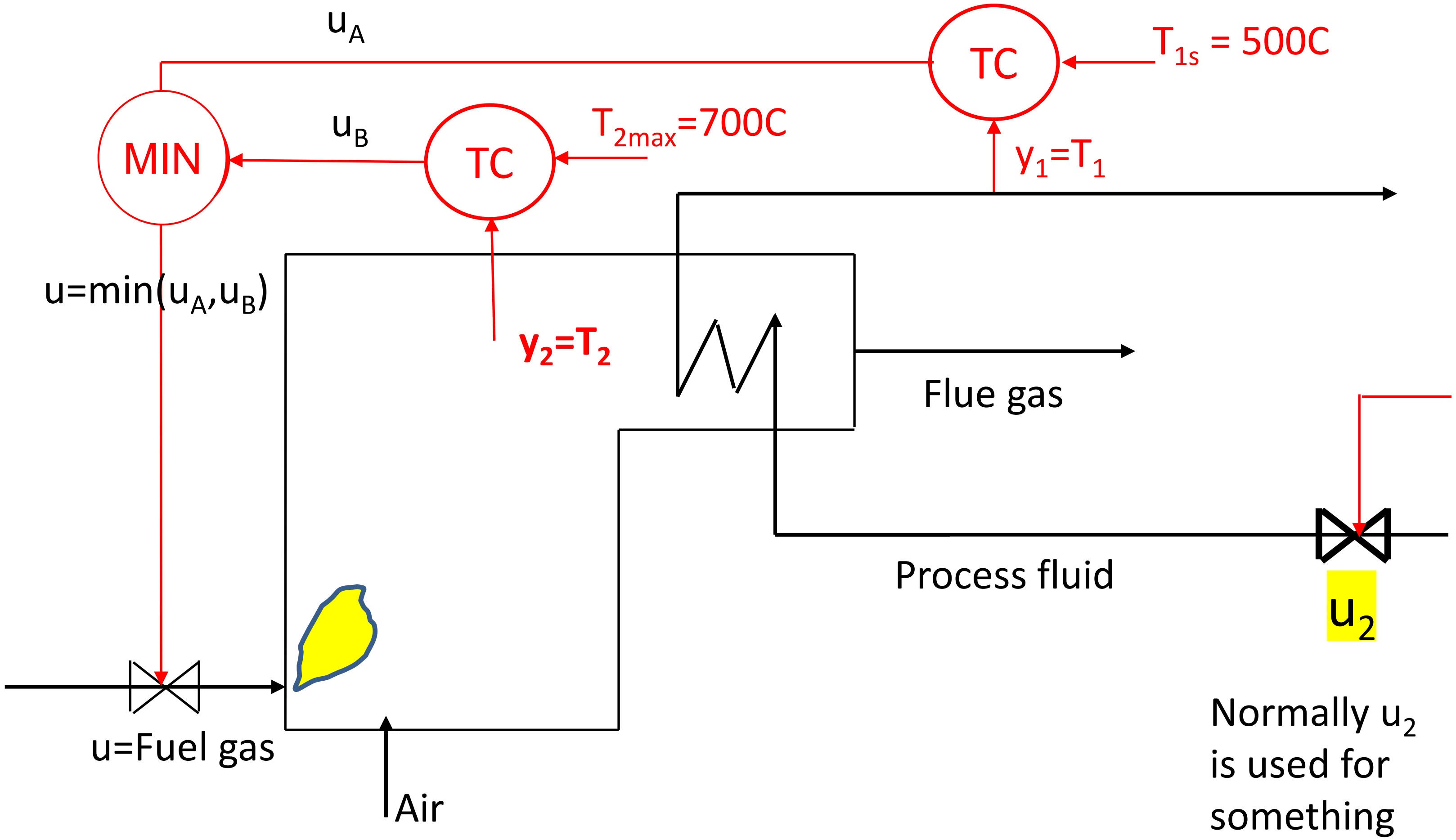
Furnace control : Cannot give up control of $y_1=T_1$. What to do?

Inputs (MV)

- u = Fuel gas flowrate
- u_2 = Process flowrate

Output (CV)

- y_1 = process temperature T_1 (with desired setpoint)



Normally u_2 is used for something else

Complex MV-CV switching

Cannot give up controlling T_1

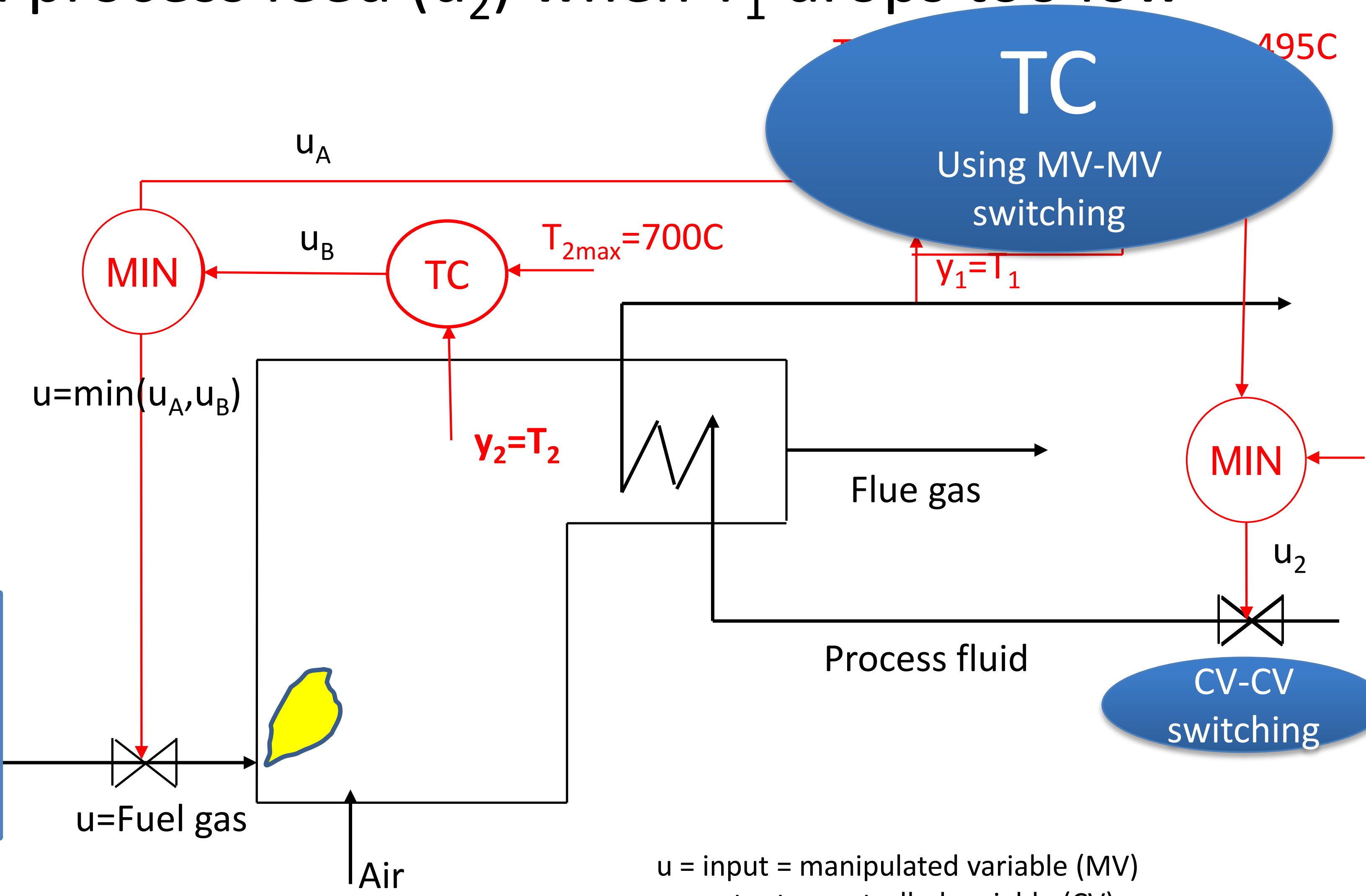
Solution: Cut back on process feed (u_2) when T_1 drops too low

Inputs (MV)

- u = Fuel gas flowrate
- u_2 = Process flowrate

Output (CV)

- y_1 = process temperature (with desired setpoint)



Note: Standard Split Range Control (Alt. 1) is not good here for MV-MV swiitching.
Could be two reasons for too little fuel

- Fuel is cut back by override (safety)
- Fuel at max,

So don't know limit for MV1 to use in SRC-block.

u = input = manipulated variable (MV)
y = output = controlled variable (CV)

Desired: $T_1 = T_{1s}, T_2 \leq T_{2\max}$

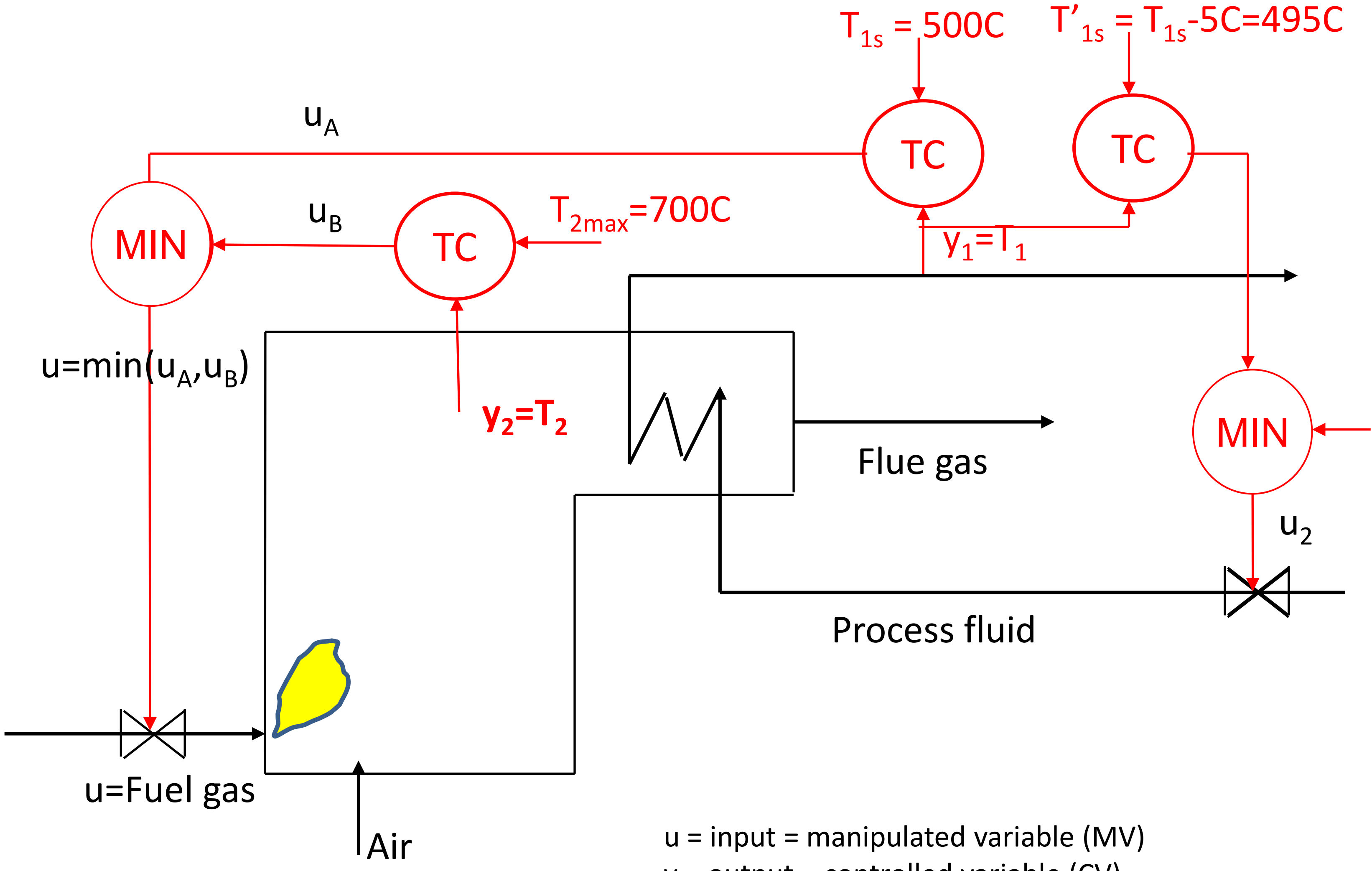
Use Alt. 2: Two controllers

Inputs (MV)

- u = Fuel gas flowrate
- u2 = Process flowrate

Output (CV)

- y_1 = process temperature (with desired setpoint)



u = input = manipulated variable (MV)
y = output = controlled variable (CV)

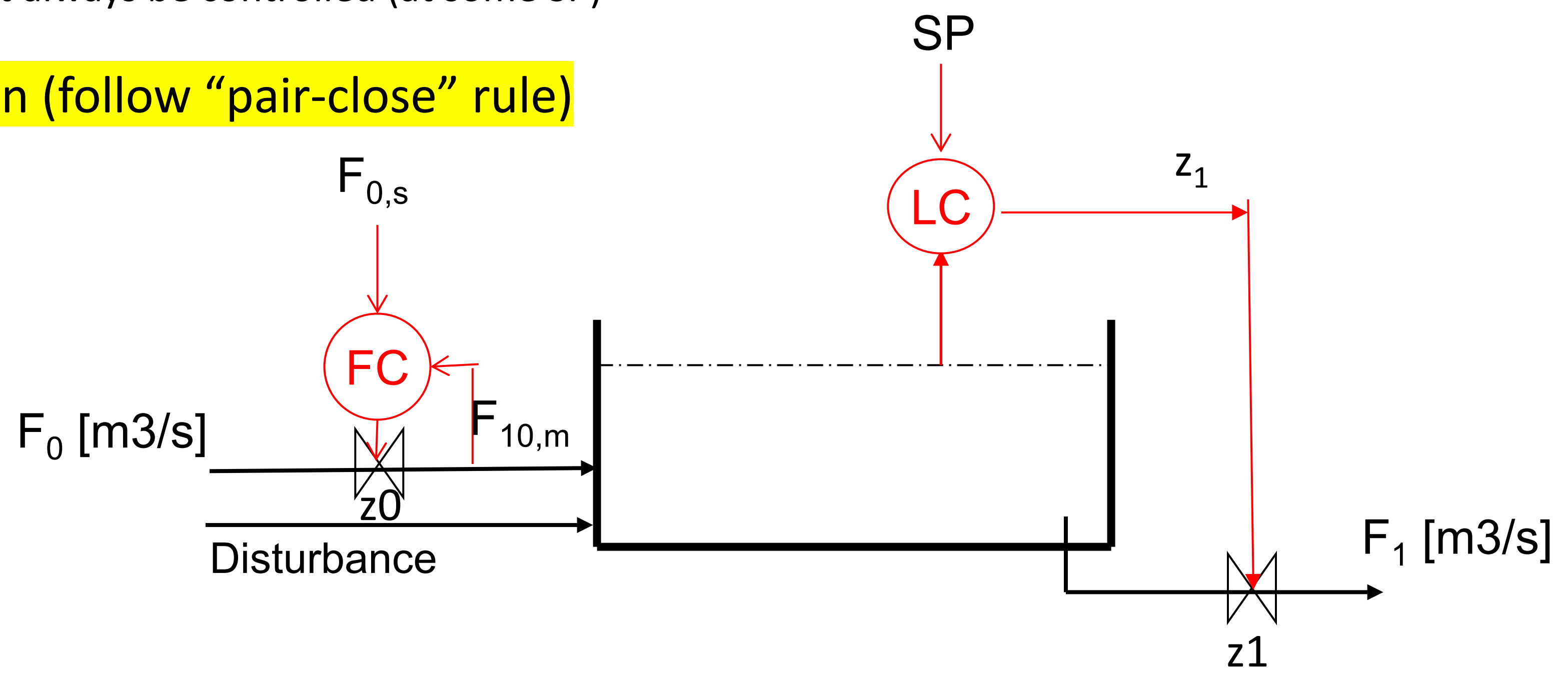
Systematic design of simple advanced controllers (APC)

- First design simple control system for nominal operation
  - With single-loop PID control we need to make pairing between inputs (MVs) and outputs (CVs):
  - Should try to follow two rules
    1. «Pair close rule» (for dynamics).
    2. «Input saturation rule»:

Example : Level control

MV1 = z0 (inflow valve position)
MV2 = z1 (outflow valve position) (likely to saturate)
CV1 = F0 (inflow): Should be controlled at setpoint $F_{0,s}$ (if possible)
CV2 = level: must always be controlled (at some SP)

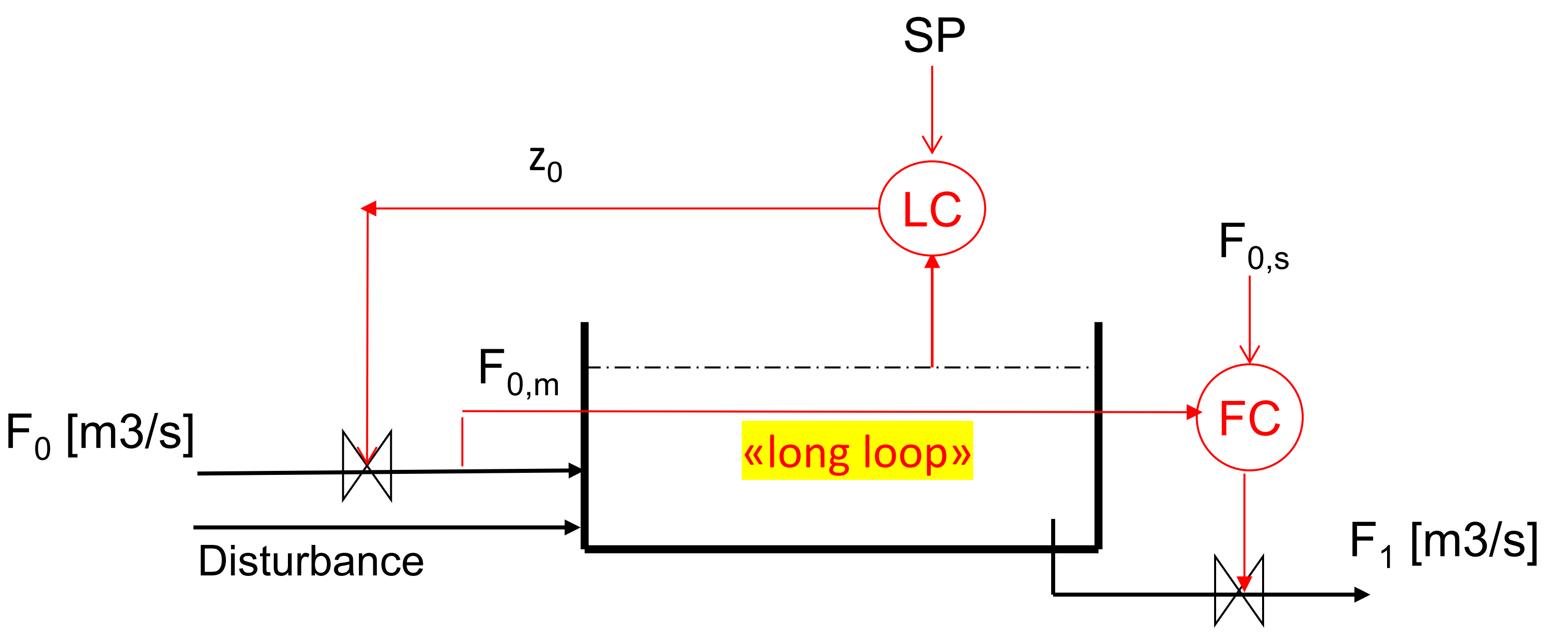
Nominal design (follow “pair-close” rule)



Problem: outflow-valve may saturate at fully open (z1=1) and then we lose level control
Note: We did not following the “input saturation rule” which says:
Pair MV that may saturate (z1) with CV that can be given up (F0)

This gives simple MV-CV switching (if z1 saturates at fully open)

Reverse pairing (follows “input saturation rule”):



BUT with Reverse pairing: Get “long loop” for F0
In addition: loose control of y2= level if z0 (F0-valve) saturates

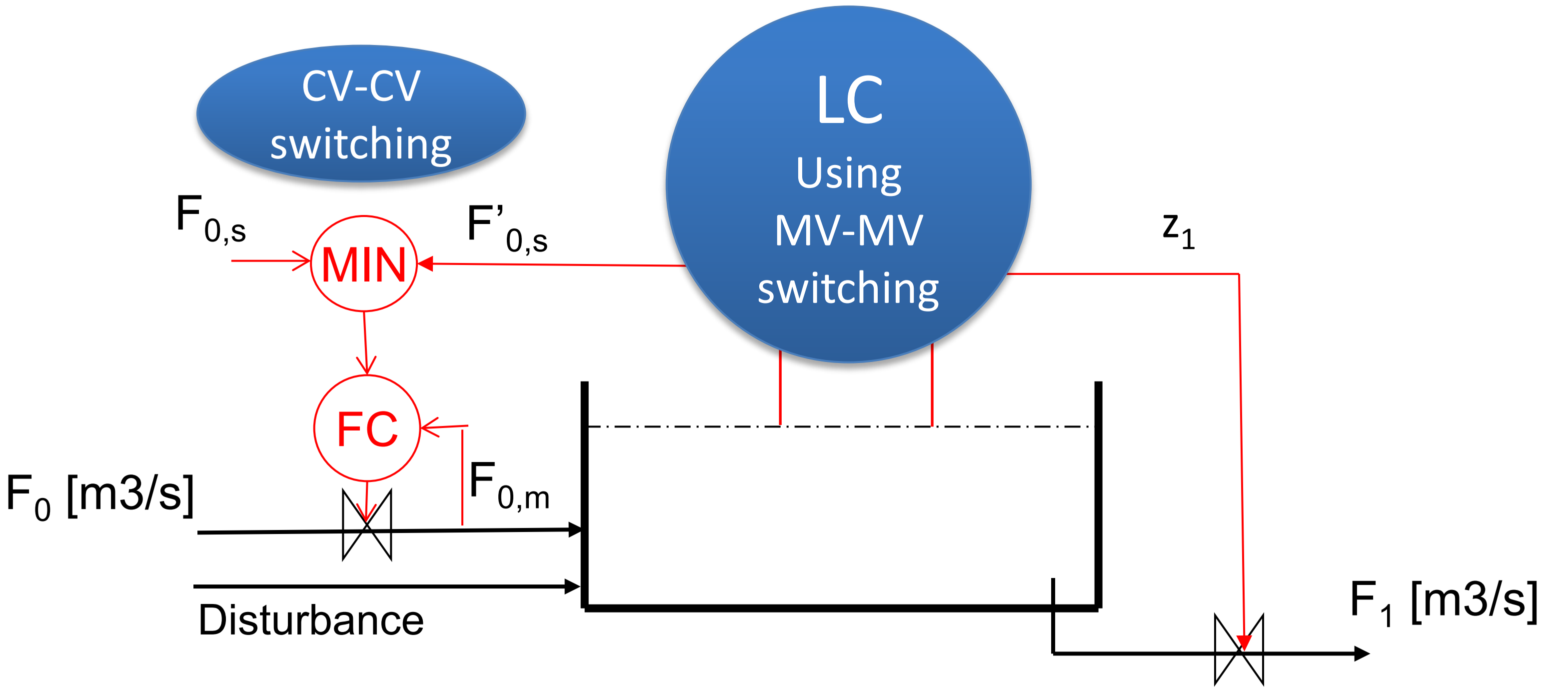
«Long loop» = Works through other loops

This is complex MV-CV switching

Alternative solution: Follow “Pair close”-rule and use Complex MV-CV switching.

When z1 saturates at max, use the other MV (z0) for level control and give up controlling F0

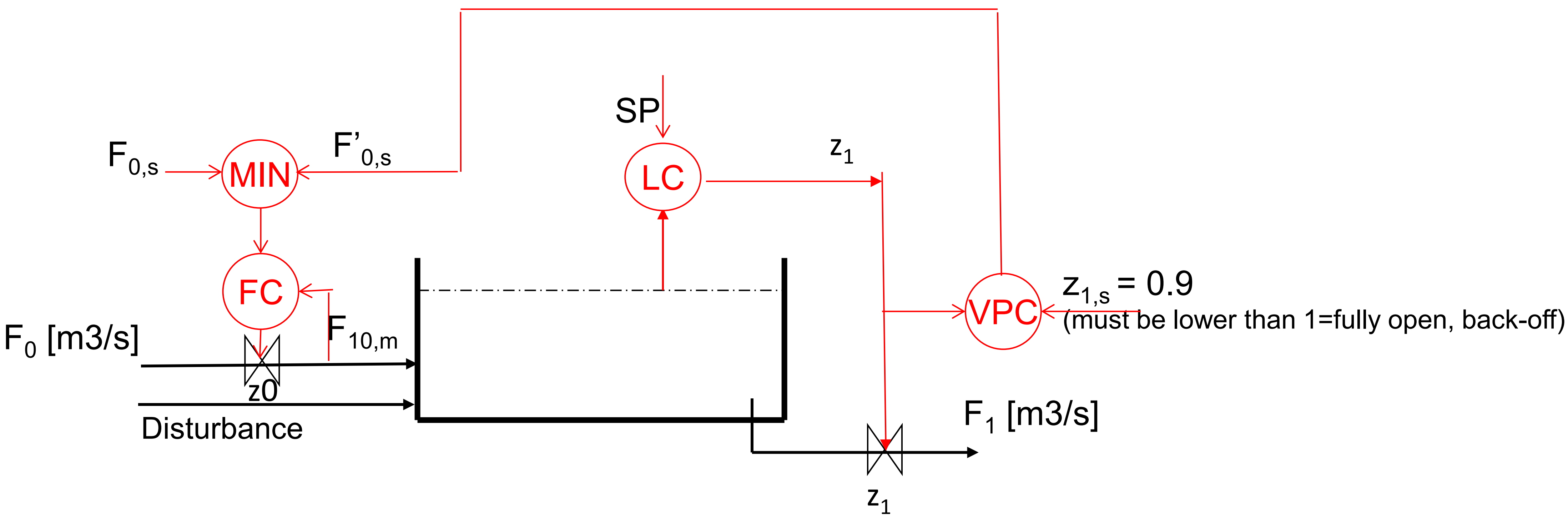
Get: “Bidirectional inventory control”



Three alternsatives for MV-MV switching

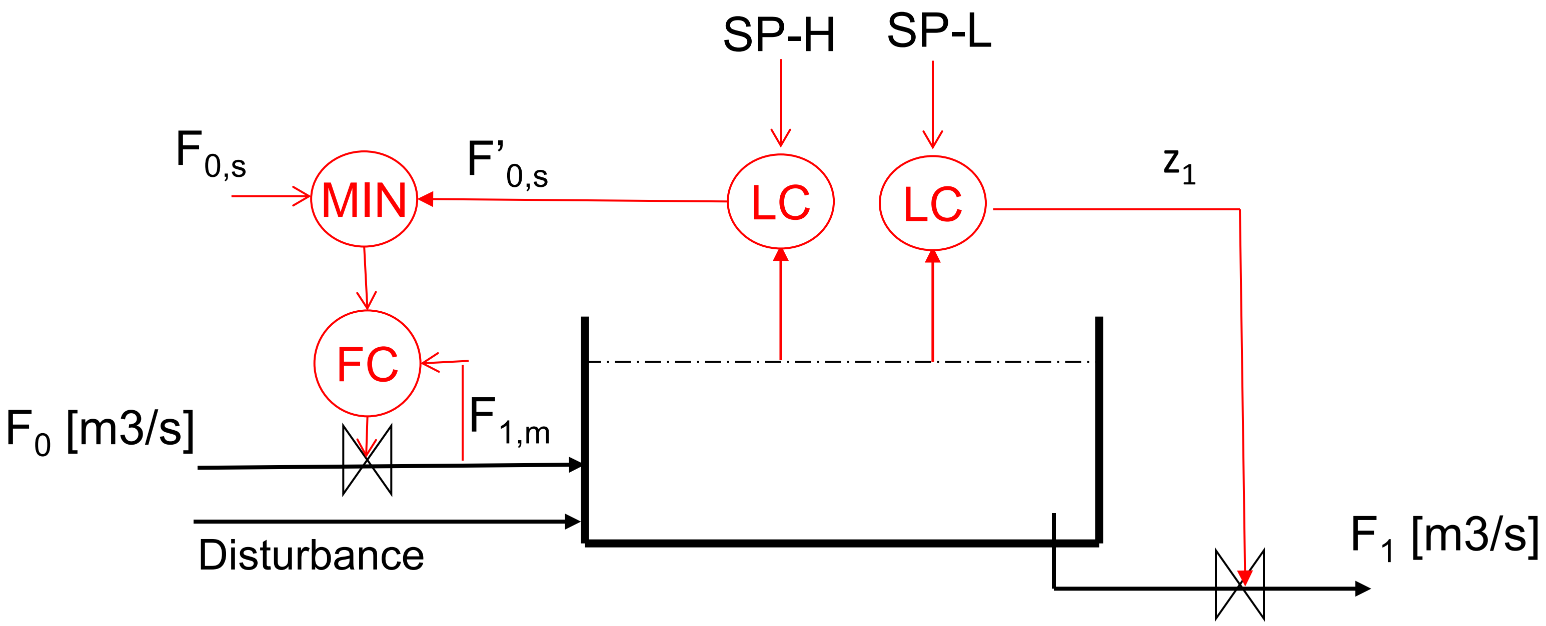
1. SRC (problem since F_{0s} varies)
2. Two controllers
3. VPC (“Long loop” for z1, backoff)

Alt. 3 MV-MV switching: VPC



VPC: “reduce inflow (F_0) if outflow valve (z_1) approaches fully open”

Alt. 2 MV-MV switching: Two controllers (recommended)



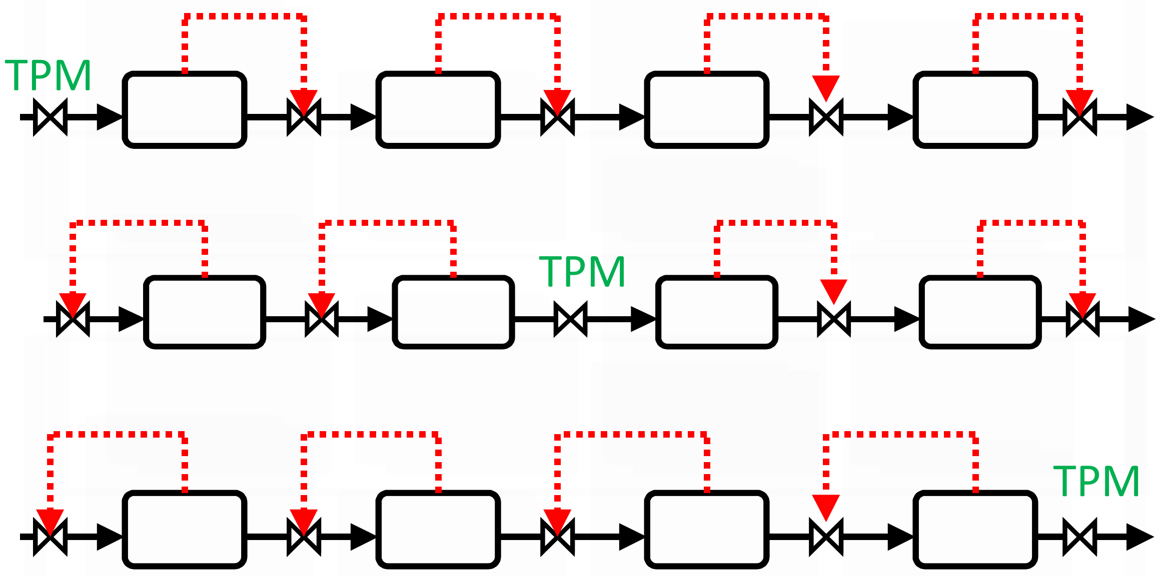
SP-L = low level setpoint
SP-H = high level setpoint

In addition: Use of two setpoints is good for using buffer dynamically!!

- Use low setpoint when level is controlled by product (outflow): Have room for feed if outflow stops temporarily.
- Use high setpoint when level is controlled by feed (inflow): Can keep producing if inflow stops temporarily.

Inventory control for units in series and TPM

- *TPM* ("gas pedal") = *Variable used for setting the throughput/production rate (for the entire process).*
- Where is the TPM located for the process?
  - Usually at the feed, but not always! To maximize production: Locate close to bottleneck
  - Important for dynamics
  - Determines the inventory control structure
- **Rule (Price et al., 1994): Inventory control (Level and pressure) must be radiating around TPM:**



TPM = Throughput manipulator

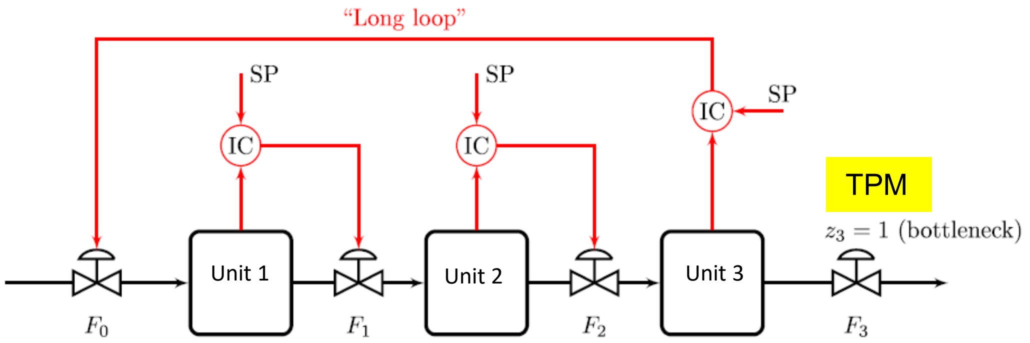
Rules for inventory control

Rule 1. Cannot control (set the flowrate) the same flow twice

Rule 2. Controlling inlet or outlet pressure indirectly sets the flow (indirectly makes it a TPM)

Rule 3. Follow the radiation rule whenever possible

Breaking the radiation rule leads to undesirable «long loops»*:



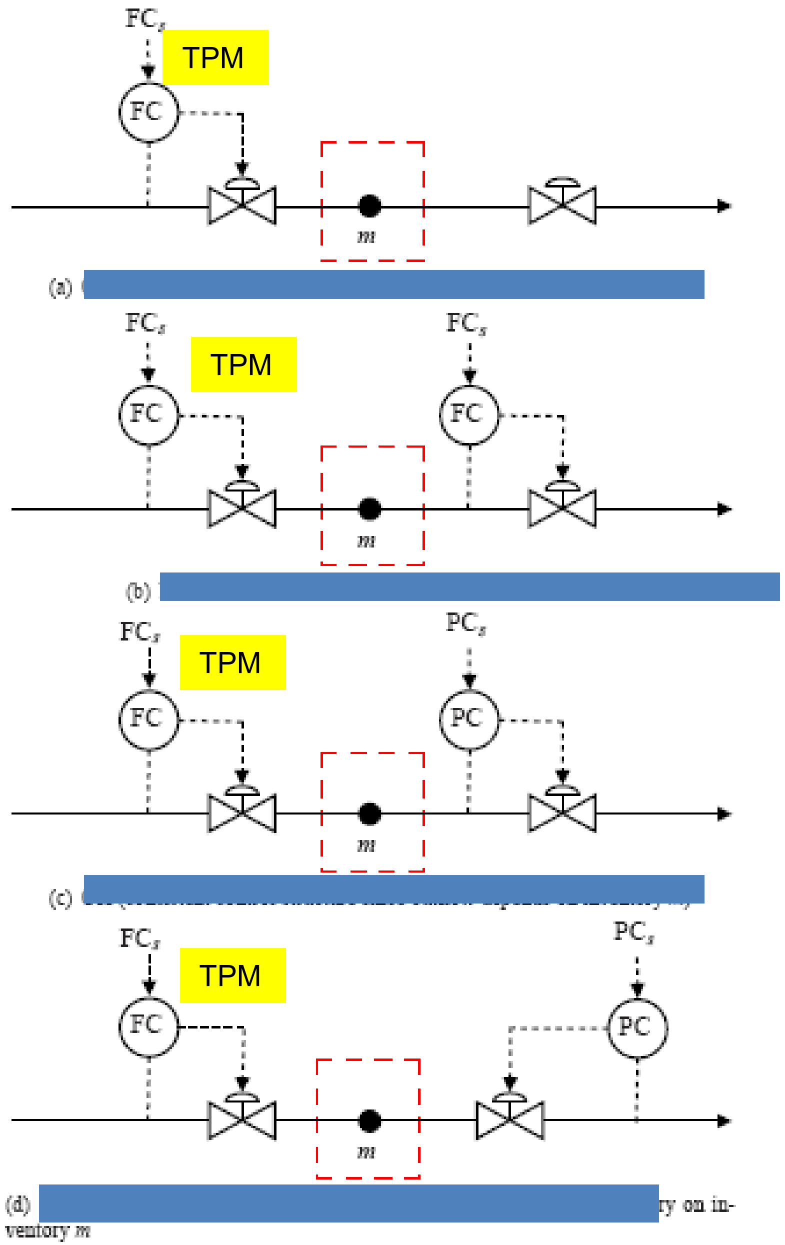
(d) Inventory control with undesired "long loop", not in accordance with the "radiation rule" (for given product flow, TPM= F_3)

Comment: Originally the TPM was at the feed (F_0) - but then the outflow (F_3) reached saturation (so this became the TPM) – and we let F_0 take over the inventory control in the last unit.
This may work OK if the inventory control in units 1 and 2 is very fast.

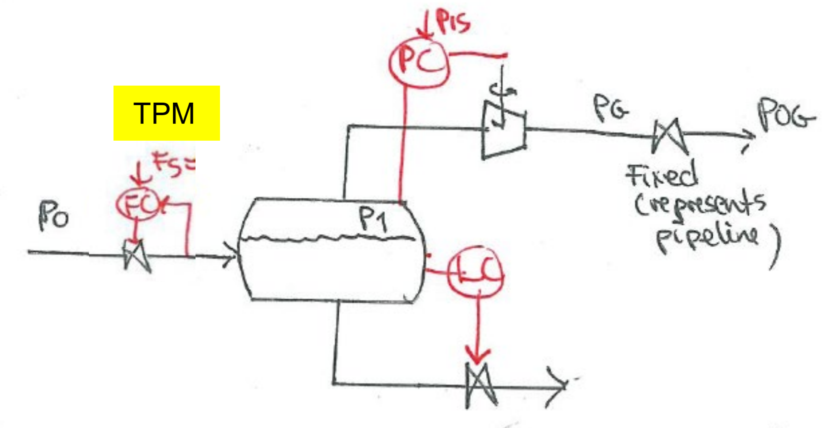
*A «Long loop» does not follow the «pair close» rule, and the functioning of a long loop depends on other loops being closed.

QUIZ. Are these structures workable (consistent)? Yes or No?

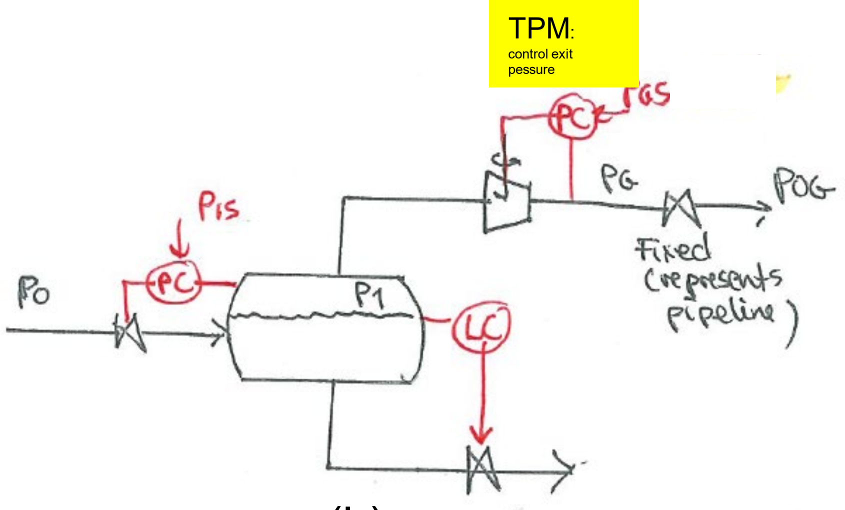
Hint: What happens to the mass holdup inside the red box? Is it self-regulated?



Quiz 2. Gas-liquid separator.
Where is TPM? Consistent (One is not)?

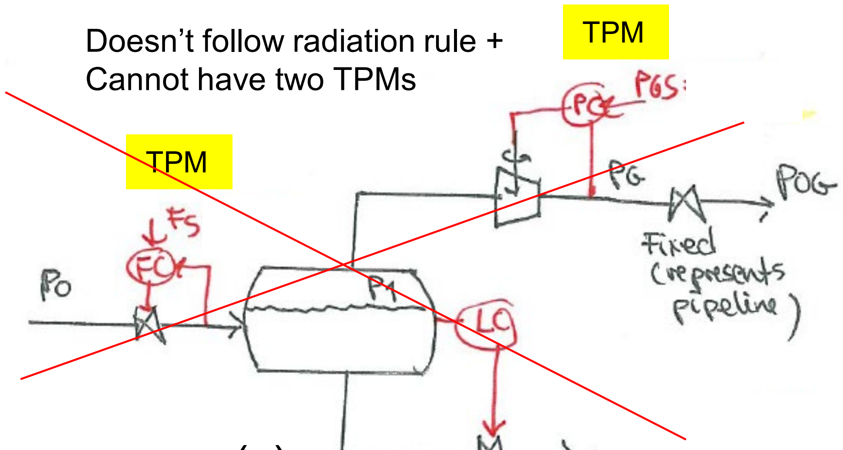


(a)

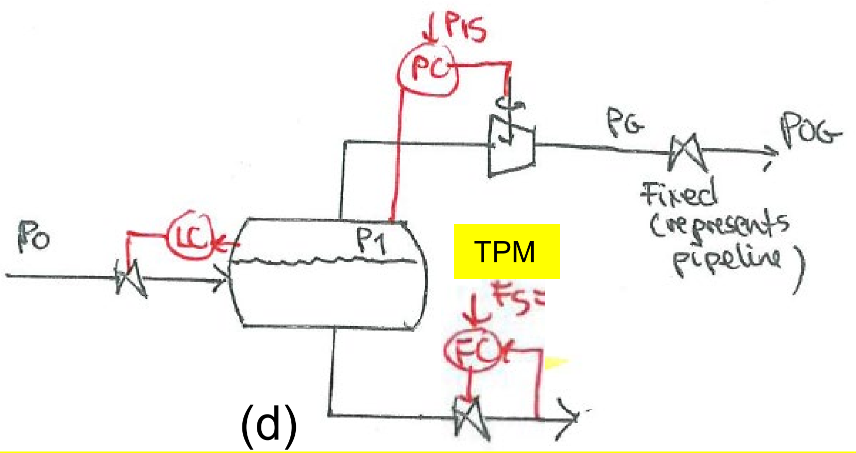


(b)

Doesn't follow radiation rule +
Cannot have two TPMs



(c)



(d)

Case (a): Given feedrate. Could alternatively set p_0
Cases (b) and (c): Gas production limiting
Case (d): Liquid production limiting

Rule: Setting in-pressure p_0 sets inflow = TPM at inlet or inlet direction (no cases above)
Setting out-pressure p_G sets outflow = TPM at outlet or outlket direction (offdiagonal two cases)

Generalization of bidirectional inventory control

Reconfigures automatically (to follow «radiation rule!) with optimal buffer management!!

Maximize throughput:

$F_s = \infty$

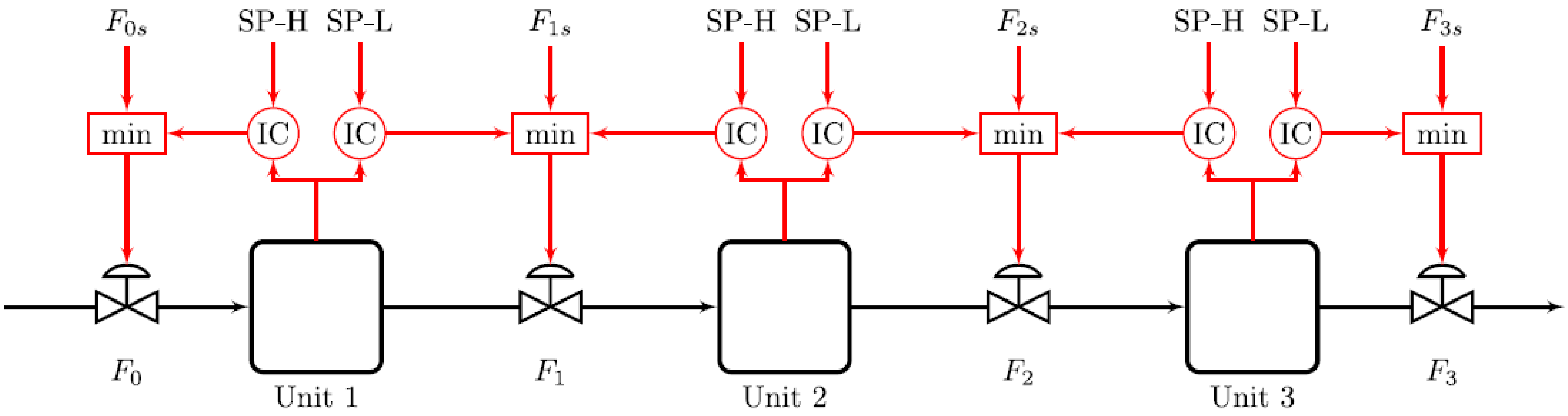


Fig. 36. Bidirectional inventory control scheme for automatic reconfiguration of loops (in accordance with the radiation rule) and maximizing throughput. Shinskey (1981) Zotică et al. (2022).
SP-H and SP-L are high and low inventory setpoints, with typical values 90% and 10%.
Strictly speaking, with setpoints on (maximum) flows ($F_{i,s}$), the four valves should have slave flow controllers (not shown). However, one may instead have setpoints on valve positions (replace $F_{i,s}$ by $z_{i,s}$), and then flow controllers are not needed.

F.G. Shinskey, «Controlling multivariable processes», ISA, 1981, Ch.3

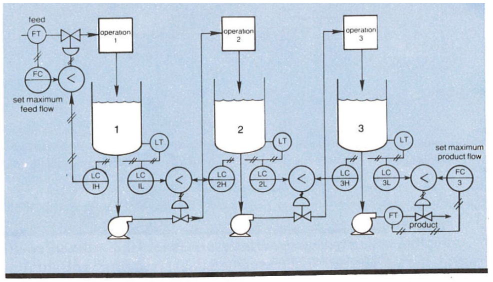
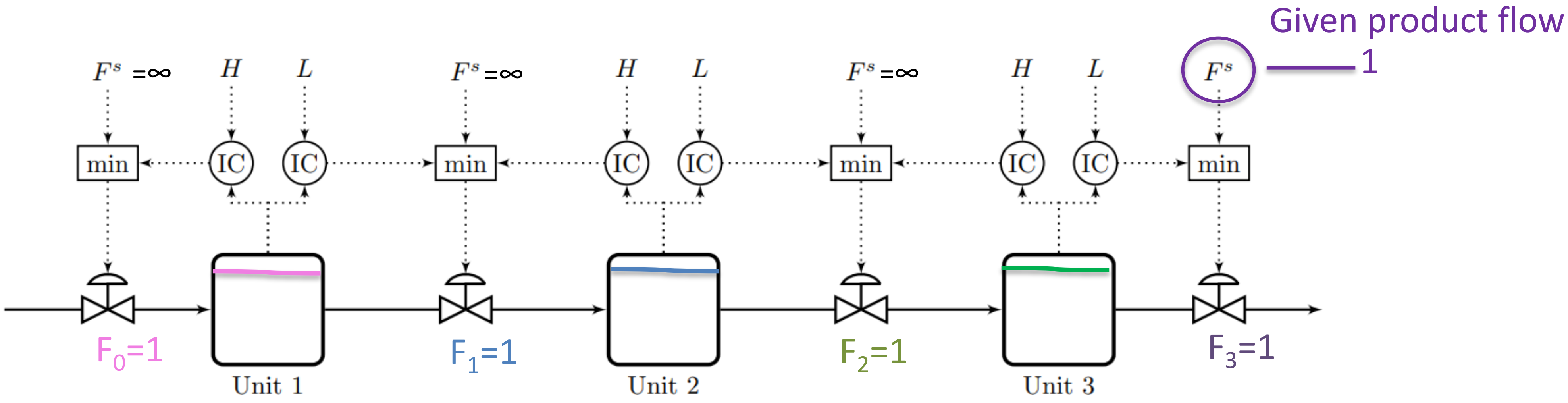
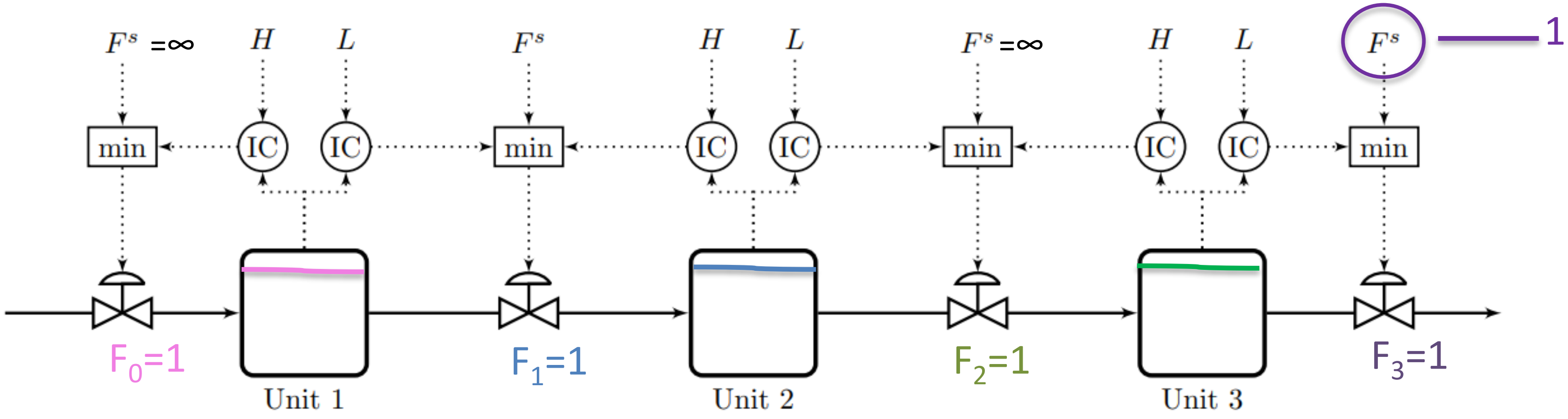


Fig. 3-7. Production rate can be set at either end of the process or constrained at any intermediate point without loss of inventory control.

Cristina Zotica, Krister Forsman, Sigurd Skogestad ,»Bidirectional inventory control with optimal use of intermediate storage», Computers and chemical engineering, 2022



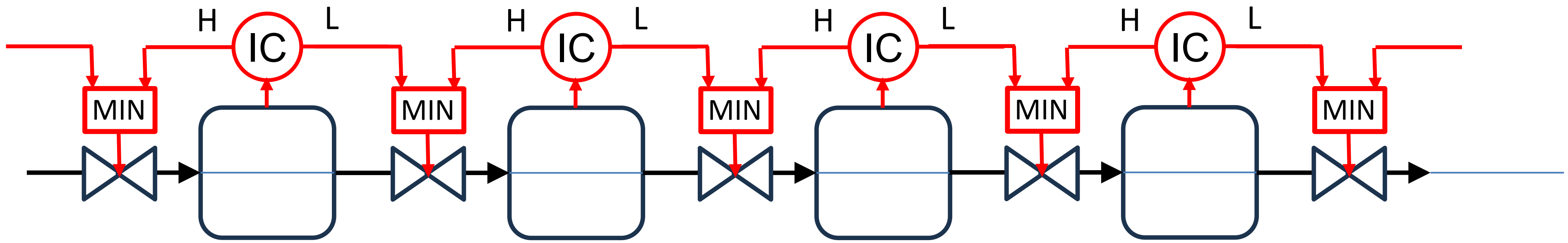
All levels are high (SP-H)

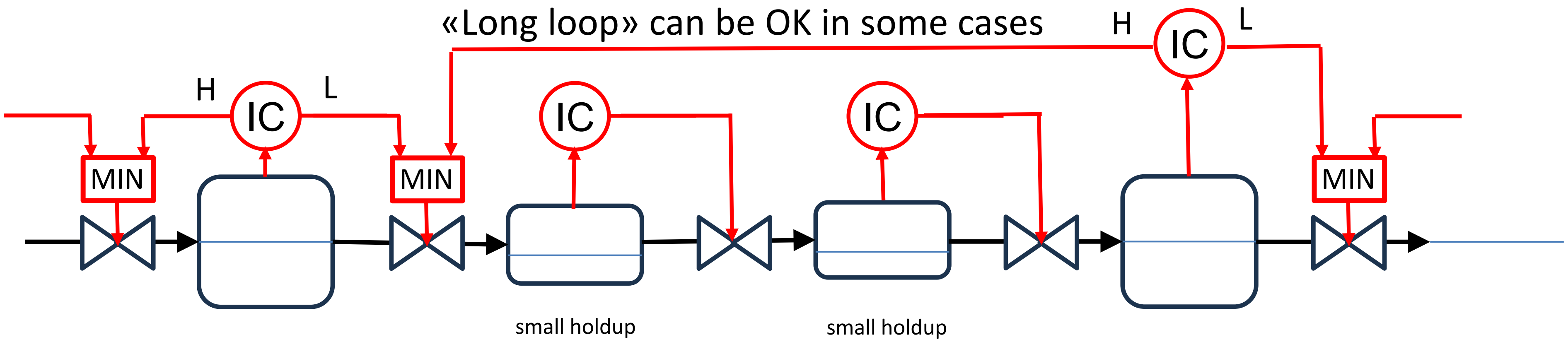


Challenge: Can MPC be made to do his? Optimally reconfigure loops and find optimal buffer?

YES. Use «trick»/insight of unachievable high setpoints on all flows

Don’t need bidirectional control on all units





Important insight

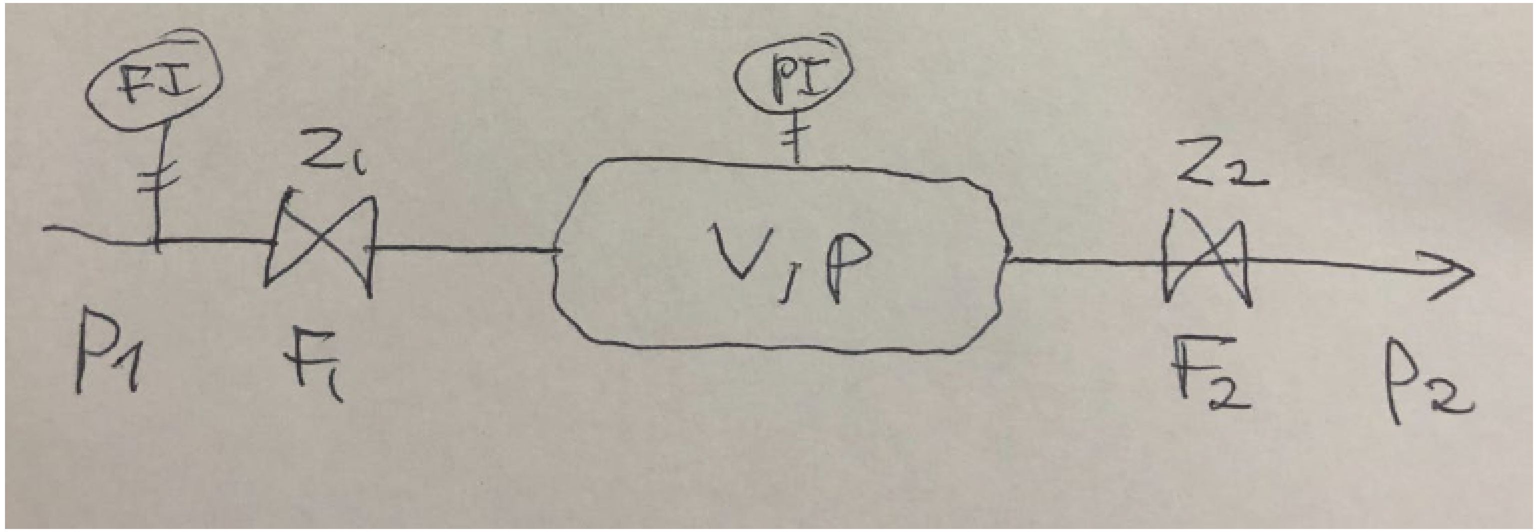
- Many problems: Optimal steady-state solution always at constraints
- In this case optimization layer may not be needed
  - if we can identify the active constraints and control them using selectors

QUIZ

What happens if we don't follow the radiation rule?

Answer: Go to saturation, so one loop fails.

Problem 5 (25%). Modelling and control of flow and pressure



Consider a gas pipeline with two valves. We have measurements of the inflow F_1 and the intermediate pressure p and these should be controlled. The volume of the pipeline can be represented as a tank with volume V as shown in the figure above.

Steady-state data: F_1=1 kg/s, $z_1=z_2$=0.5, p_1=2 bar, p=1.88 bar, p_2=1.8 bar, V=130 m^3, T=300 K, Parameters: R=8.31 J/K.mol, M_W=18e-3 kg/mol (so the gas is steam).

The following model equations are suggested to describe the system.

(1) $dm/dt = F_1 - F_2$

(2) $m = k_p\, p$ where $k_p = VM_w/(RT)$

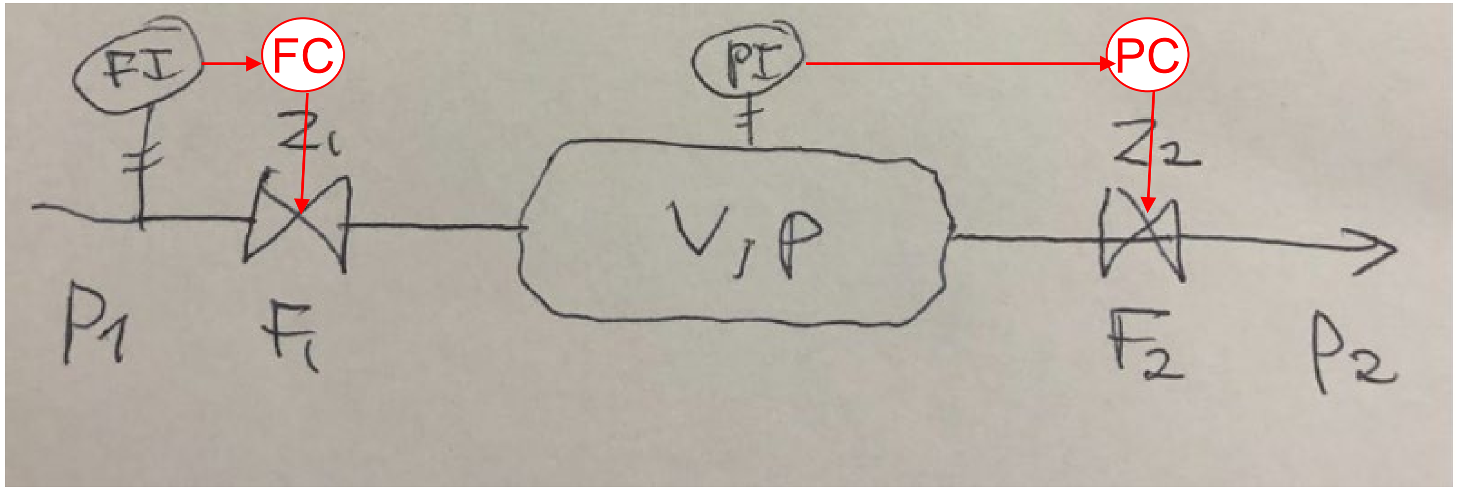
(3) $F_1 = C_1 z_1 \sqrt{p_1 - p}$

(4) $F_2 = C_2 z_2 \sqrt{p - p_2}$

(a) Suggest a control structure

(b) What if we want to control p2 instead of p?

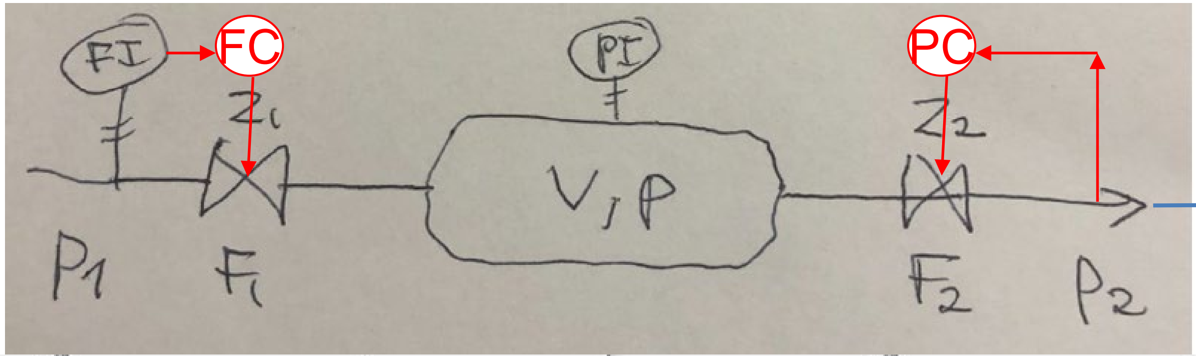
(a) The «obvious» pair-close pairing os OK. However, interactions between loops may be severe. Suggest tuning the FC first, and the PC about 5 times slower.



(b)

NO!

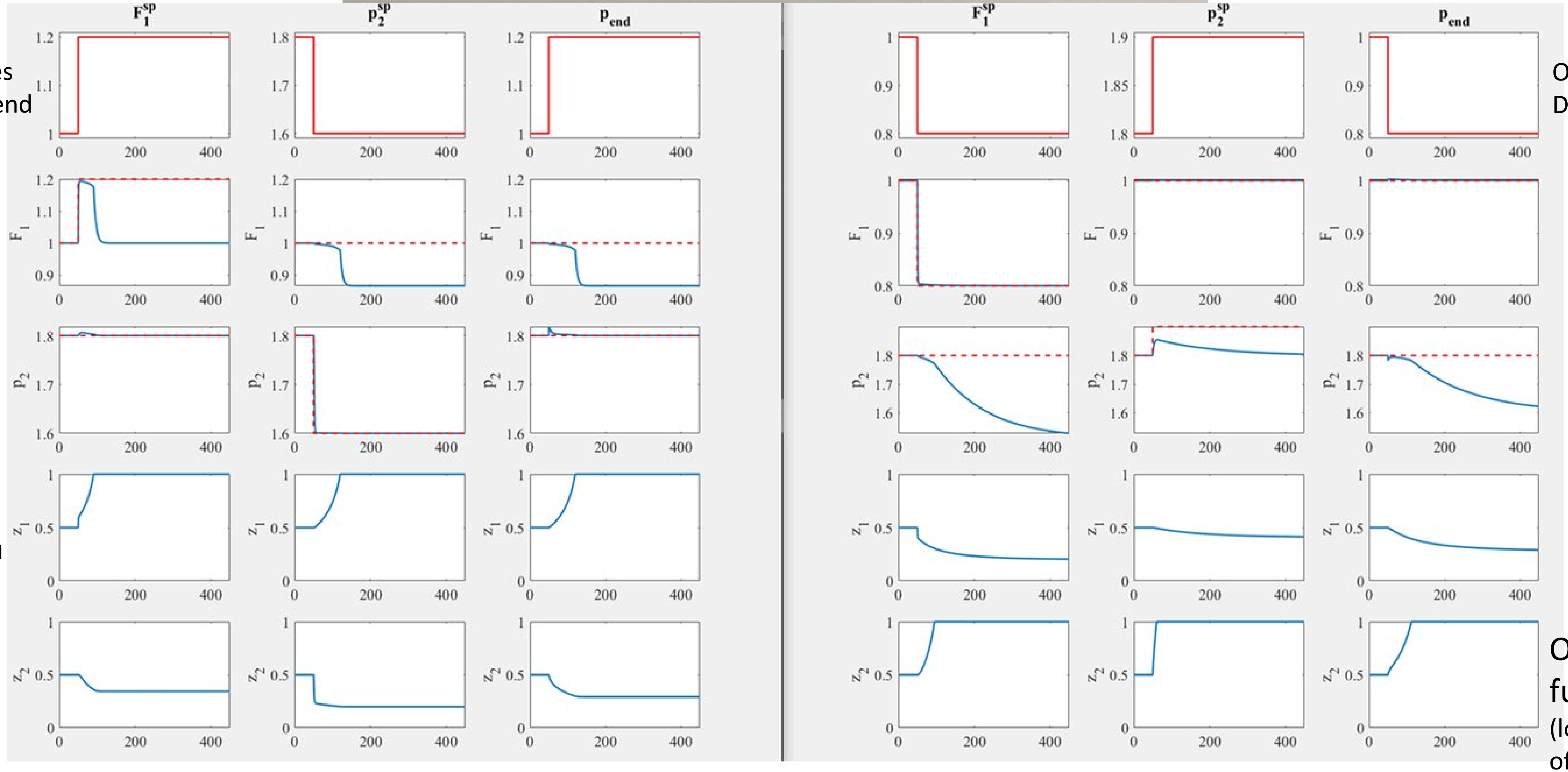
Not consistent



$F_3 = c_3(p_2 - p_{end})^{1/2}$

Disturbances in F1, p2, pend

Opposite Disturbances



z1 to fully open (lose control of F1)

Or: z2 to fully open (lose control of p2)

Systematic design of simple advanced controllers (APC)

- Then make a list of possible new contraints that may be encountered (because of disturbances, parameter changes, price changes)
- Reach constraint on new CV
  - Simplest: Find an unused input (simple CV-MV switching)
  - Otherwise: CV-CV switching using selector (may involve giving up a CV-constraint or a self-optimizing CV)
- Reach constraint on MV (which is used to control a CV)
  - Simplest (If we followed input saturation rule):
    - Can give ip controlling the CV (Simple CV-MV switching)
    - Don’t ned to do anything
  - Otherwise (if we cannot give up controlling CV)
    - Simplest: Find an unused input
      - MV-MV switching
    - Otherwise: Pair with a MV that already controls another CV
      - Complex CV-MV switching
      - Must combine MV-MV and CV-CV switching
- Is this always possible? No, pairing inputs and outputs may be impossible with many constraints.
- May then use MPC instead

Conclusion Advanced process control (APC)

- Classical APC, aka «Advanced regulatory control» (ARC) or «Advanced PID»:
  - Works very well in many cases
  - Optimization by feedback (active constraint switching)
  - Need to pair input and output.
    - Advantage: The engineer can specify directly the solution
    - Problem: Unique pairing may not be possible for complex cases
  - Need model only for parts of the process (for tuning)
  - Challenge: Need better teaching and design methods

- MPC may be better (and simpler) for more complex multivariable cases
  - But MPC may not work on all problems (Bidirectional inventorycontrol)
  - Main challenge: Need dynamic model for whole process
  - Other challenge: Tuning may be difficult