

**Part 2.**

**Plantwide process control  
«Control architectures»**

Sigurd Skogestad

# Plantwide control (Control architecture)

- Objective: Put controllers on flow sheet (make P&ID)
- Two main objectives for control: Longer-term economics (CV1) and shorter-term stability (CV2)
- Regulatory (basic) control layer for CV2 and supervisory (advanced) control layer for CV1

**How can we design a control system for a complete chemical plant?**

Where do we start?

What should we control? And why?



**Sigurd at Caltech (1984)**

## **How we design a control system for a complete chemical plant?**

- Where do we start?
- What should we control? and why?
- etc.
- etc.

# Control system structure\*

Alan Foss (“Critique of chemical process control theory”,  
AIChE Journal, 1973):

*The central issue to be resolved ... is the determination of control system structure\*.  
Which variables should be measured, which inputs should be manipulated  
and which links should be made between the two sets?*



\*Current terminology: Control system architecture

# Plantwide control = Control structure (**architecture**) design

- *Not* the tuning and behavior of each control loop...
- But rather the *control philosophy* of the overall plant with emphasis on the ***structural decisions***:
  - Selection of controlled variables (“outputs”)
  - Selection of manipulated variables (“inputs”)
  - Selection of (extra) measurements
  - Selection of control **configuration** (structure of overall controller that interconnects the controlled, manipulated and measured variables)
  - Selection of controller type (LQG, H-infinity, PID, decoupler, MPC etc.)

# QUIZ

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

- Hint 1: According to Sigurd Skogestad
- Hint 2: All became commonly used in the 1940s

## SOLUTION

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

Note: None of these are easily implemented using Model predictive control (MPC)

# Main objectives of a control system

1. Economics: Implementation of acceptable (near-optimal) operation
2. Regulation: Stable operation

ARE THESE OBJECTIVES CONFLICTING?

- Usually NOT
  - Different time scales
    - Stabilization → fast time scale
  - Stabilization doesn't "use up" any degrees of freedom
    - Reference value (setpoint) available for layer above
    - But it "uses up" part of the time window (frequency range)



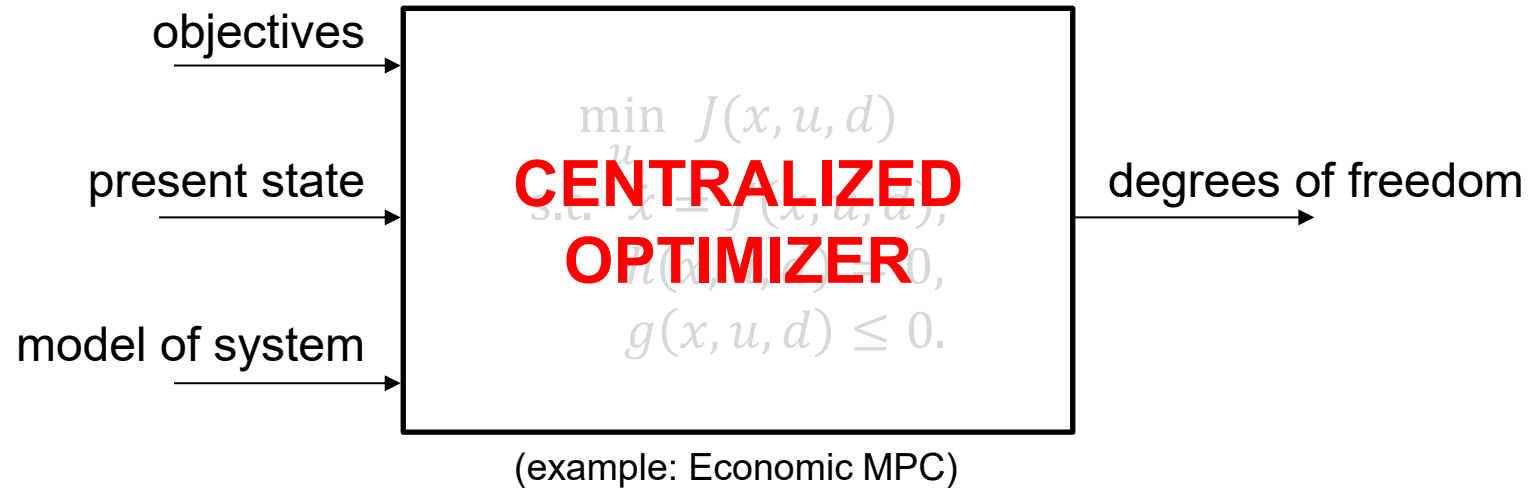
# Optimal operation

General approach: minimize cost / maximize profit, subject to satisfying constraints (product quality, environment, resources)

Mathematically,

$$\begin{aligned} \min_u \quad & J(x, u, d) \\ \text{s.t.} \quad & \dot{x} = f(x, u, d), \\ & h(x, u, d) = 0, \\ & g(x, u, d) \leq 0. \end{aligned}$$

# Optimal operation (in theory)



## Procedure:

- Obtain model of overall system
- Estimate present state
- Optimize all degrees of freedom

## Problems:

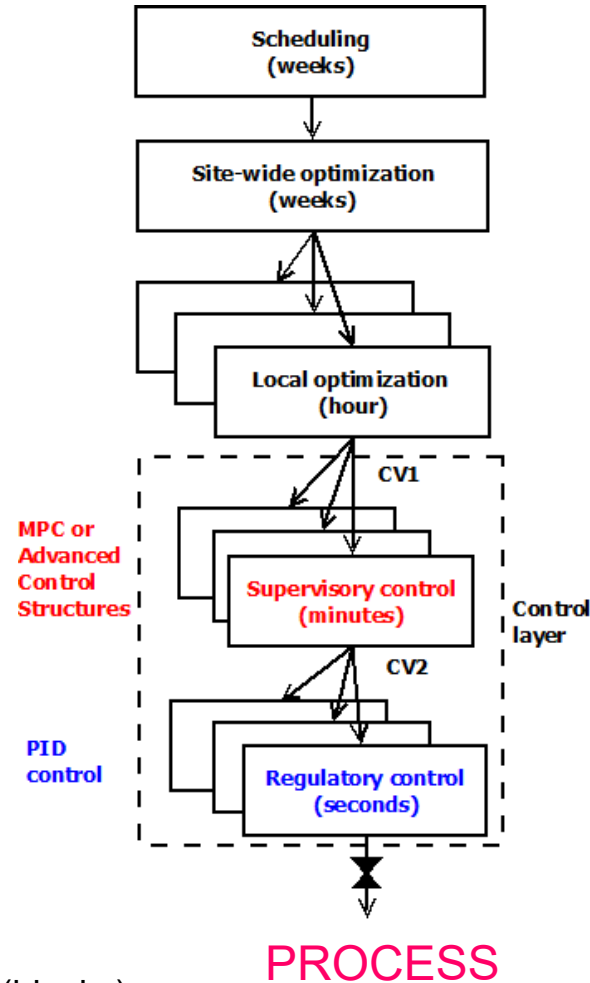
- Model not available
- Optimization is complex
- Not robust (difficult to handle uncertainty)
- Slow response time

# Engineering systems

- Most (all?) large-scale engineering systems are controlled using hierarchies of quite simple controllers
  - Large-scale chemical plant (refinery)
  - Commercial aircraft
- 100's of loops
- Simple components:
  - on-off + PI-control + nonlinear fixes + some feedforward

## Two fundamental ways of decomposing the controller

- Vertical (hierarchical; cascade)
- Based on time scale separation
- Decision: Selection of CVs that connect layers



- Horizontal  
(decentralized)
- Usually based on distance
- Decision: Pairing of MVs and CVs within layers

In addition: Decomposition of controller into smaller elements (blocks):  
Feedforward element, nonlinear element, estimators (soft sensors), switching elements

# Time scale separation: Control\* layers

Two objectives for control: Stabilization and economics

- **Supervisory (“advanced”) control layer**

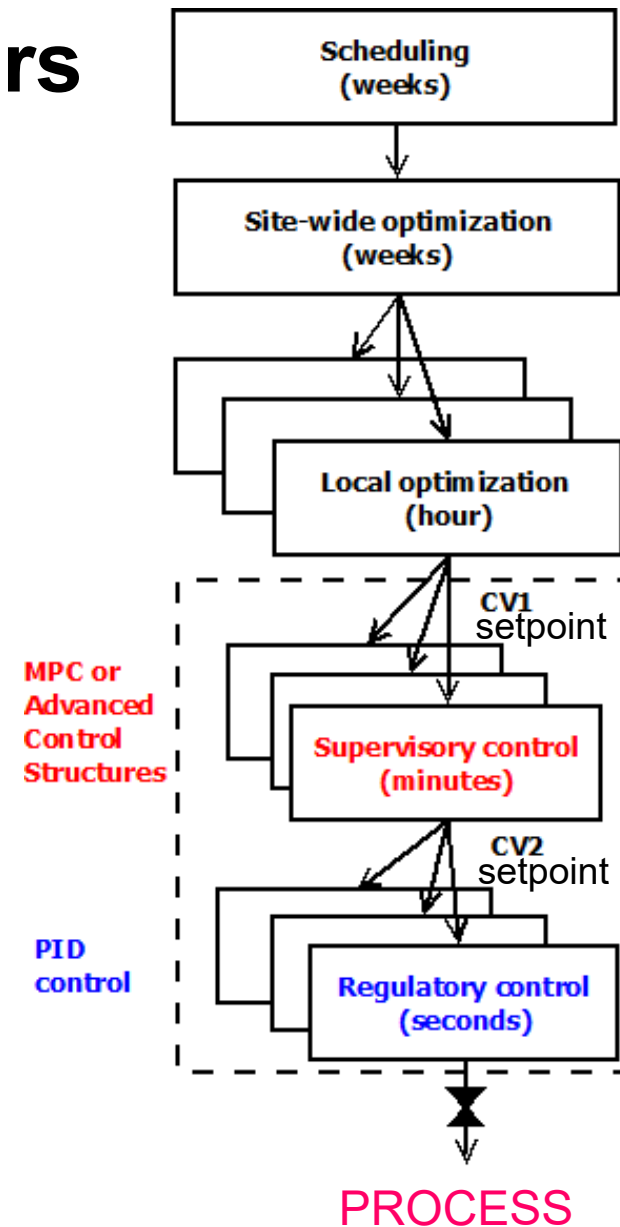
Tasks:

- Follow set points for CV1 from economic optimization layer
- Switch between active constraints (change CV1)
- Look after regulatory layer (avoid that MVs saturate, etc.)

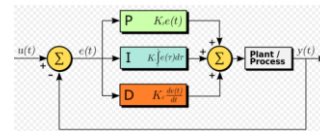
- **Regulatory control (PID layer):**

- Stable operation (CV2)

\*My definition of «control» is that the objective is to track setpoints

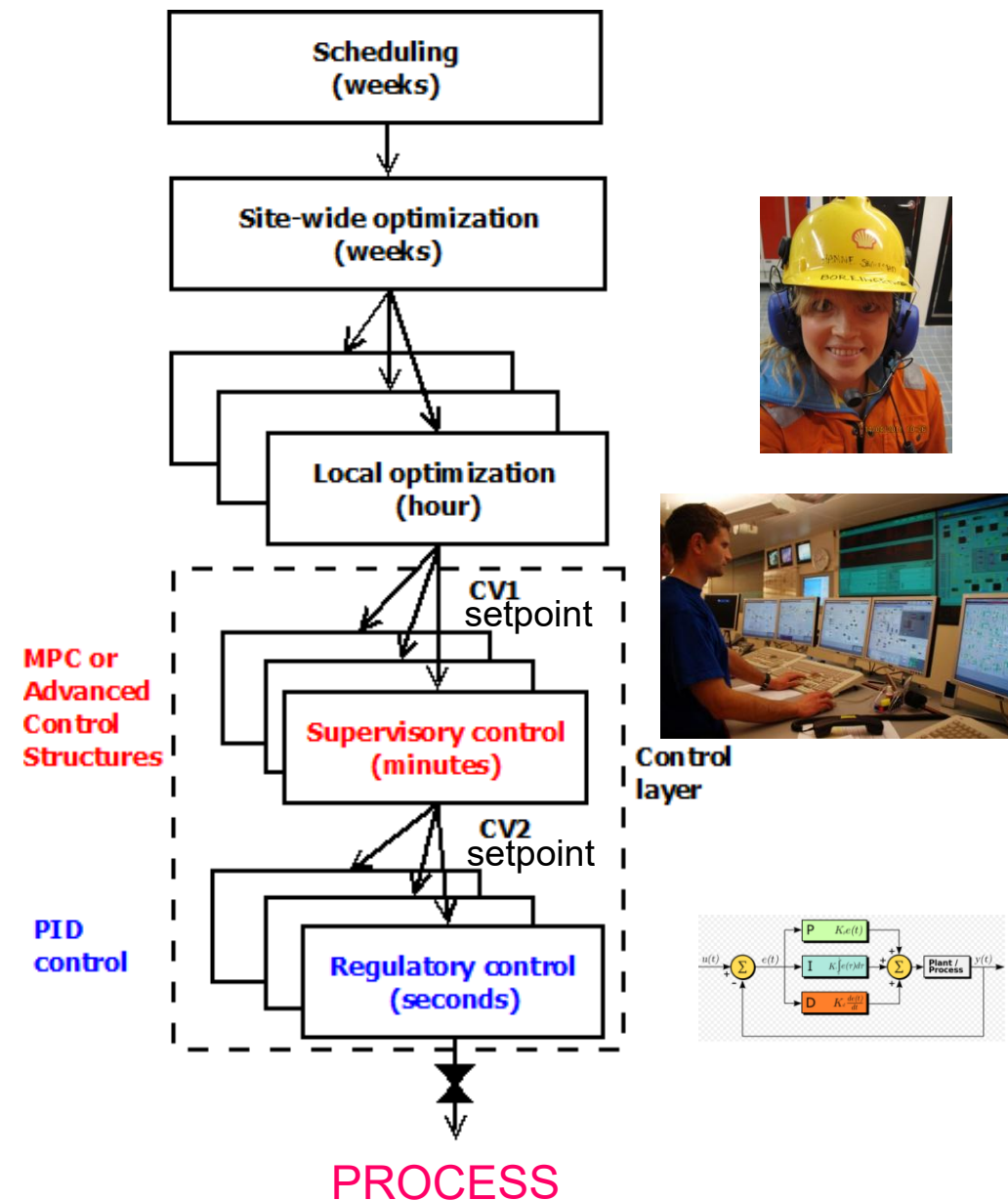


Control layer



# «Advanced» control

- Advanced: This is a relative term
- Usually used for anything than comes in addition to (or in top of) basic PID loops
- Mainly used in the «supervisory» control layer
- Two main options
  - **Standard «Advanced regulatory control» (ARC) elements**
    - Based on decomposing the control system
      - Cascade, feedforward, selectors, etc.
    - This option is preferred if it gives acceptable performance
  - **Model predictive control (MPC)**
    - Requires a lot more effort to implement and maintain
    - Use for interactive processes
    - Use with known information about future (use predictive capabilities)



# Combine control and optimization into one layer?

EMPC: Economic model predictive “control”

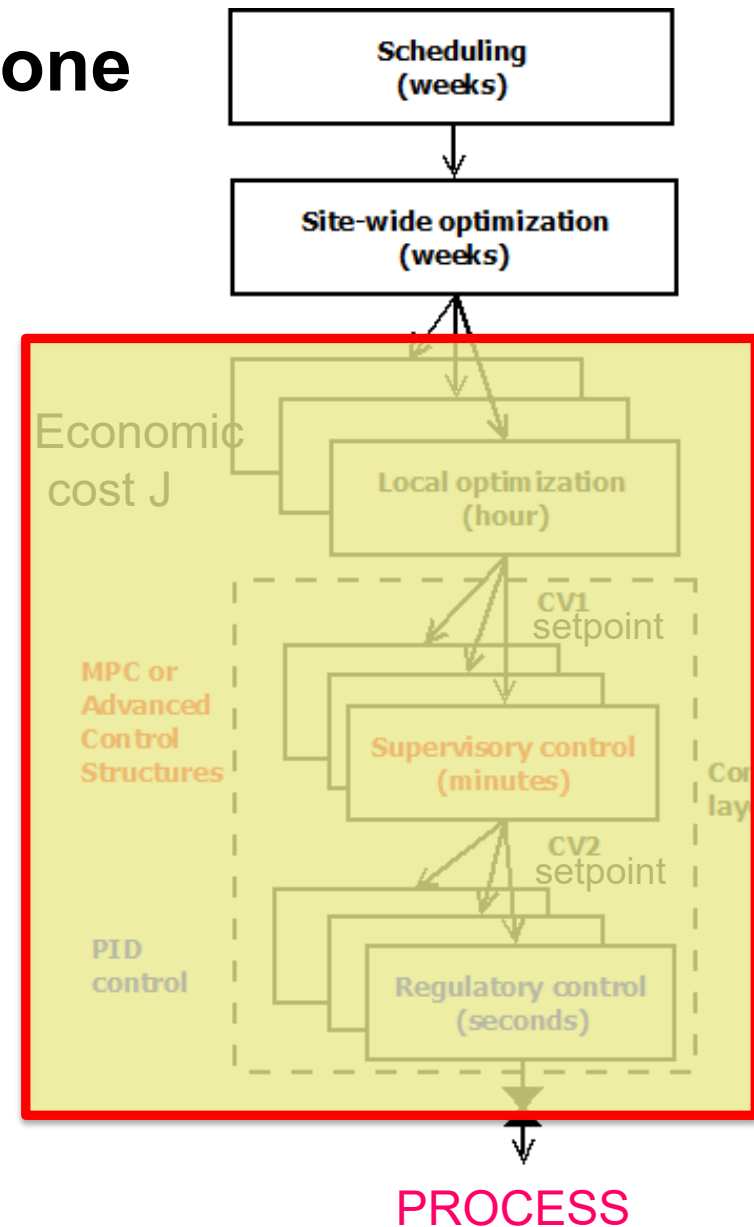
$$J_{EMPC} = J + J_{control}$$

- $J [\$/s] = J_{economic}$  = cost feed + cost energy – value products
- $J_{control} = \sum \Delta u_i^2$  (typical) - Penalize input usage

**NO, combining layers is generally not a good idea!**  
(the good idea is to separate them!)

One layer (EMPC) is optimal theoretically, but

- Need detailed dynamic model of everything
- Tuning difficult and indirect
- Slow! (or at least difficult to speed up parts of the control)
- Robustness poor
- Implementation and maintenance costly and time consuming



**EMPC**  
(no setpoints  
CV1, CV2)

# What about «conventional» RTO and MPC?

- Yes, it's OK
- Both has been around for more than 50 years (since 1970s)
  - but the expected growth never came
- MPC is still used mostly in large-scale plants (petrochemical and refineries).
- MPC is far from replacing PID as some expected in the 1990s.
- But plants need to be run optimally:
  - ⇒ Need something else than conventional RTO/MPC!

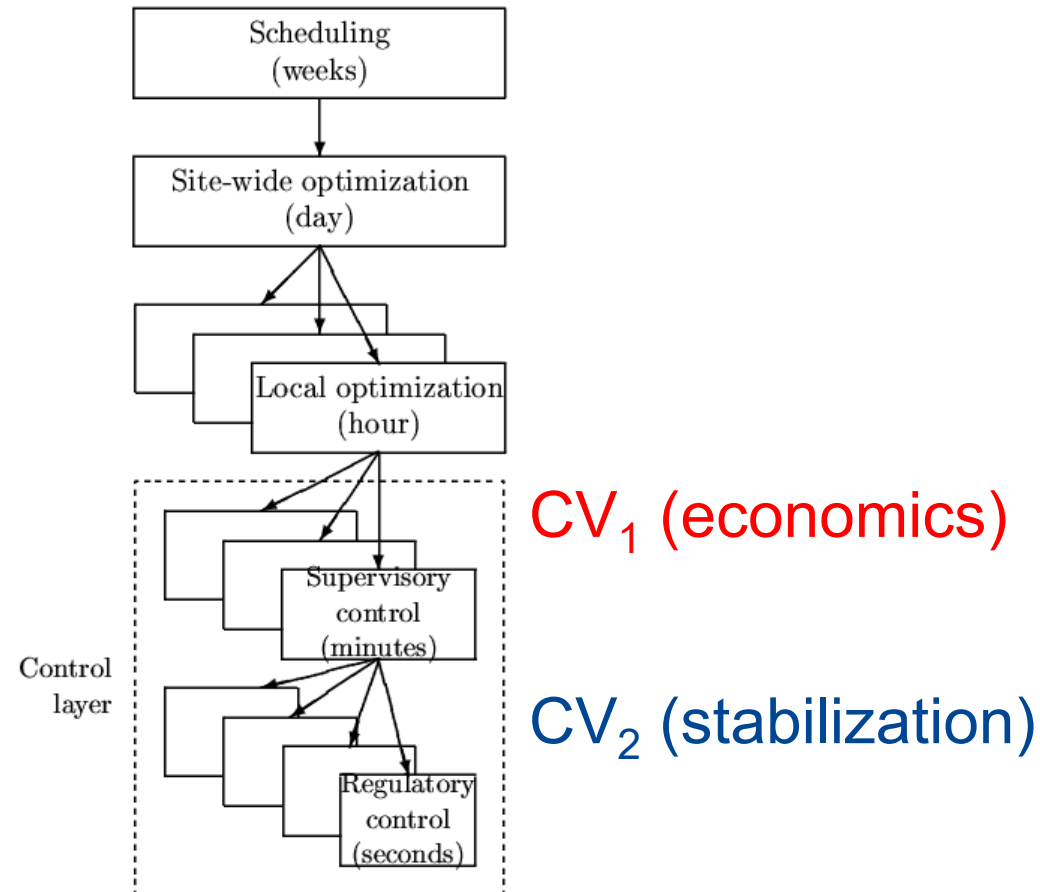


# Alternative solutions for advanced control

- Would like: Feedback solutions that can be implemented with minimum need for models
- **Machine learning?**
  - Requires a lot of data, not realistic for process control
  - And: Can only be implemented after the process has been in operation
- **“Classical advanced regulatory control“ (ARC) based on single-loop PIDs?**
  - **YES!**
  - Extensively used by industry
  - Problem for engineers: Lack of design methods
    - Has been around since 1930's
    - But almost completely neglected by academic researchers
  - Main fundamental limitation: Based on single-loop (need to choose pairing)

ARC = Advanced regulatory control

# Optimal operation and control objectives: What should we control?



# Skogestad procedure for control structure design:

- I. Top Down (analysis)
  - Step S1: Define operational objective (cost) and constraints
  - Step S2: Identify degrees of freedom and optimize operation for disturbances
  - Step S3: Implementation of optimal operation
    - What to control? (CV1) (self-optimizing control)
  - Step S4: Where set the production rate (TPM)? (Inventory control)
- II. Bottom Up (design)
  - Step S5: Regulatory control: What more to control (CV2)?
  - Step S6: Supervisory control
  - Step S7: Real-time optimization

TPM = Throughput manipulator

# **Step S1. Define optimal operation (economics)**

- Usually easy!
- What are the economic goals of the operation?
- Typical cost function\*:

$$J = \text{cost feed} + \text{cost energy} - \text{value products} \quad [\$/\text{s}]$$

\*No need to include fixed costs (capital costs, operators, maintainance) at "our" time scale (hours)

Note:  $J = -P$  where  $P =$  Operational profit

# Example: distillation column

- Distillation at steady state with given  $p$  and  $F$ :  $N=2$  DOFs, e.g.  $L$  and  $V$  (**u**)
- **Cost to be minimized (economics)**

cost energy (heating + cooling)

$$J = -P \text{ where } P = \underbrace{p_D D + p_B B}_{\text{value products}} - \underbrace{p_F F}_{\text{cost feed}} - p_V V$$

- **Constraints**

Purity D: For example,  $x_{D, \text{impurity}} \leq \max$

Purity B: For example,  $x_{B, \text{impurity}} \leq \max$

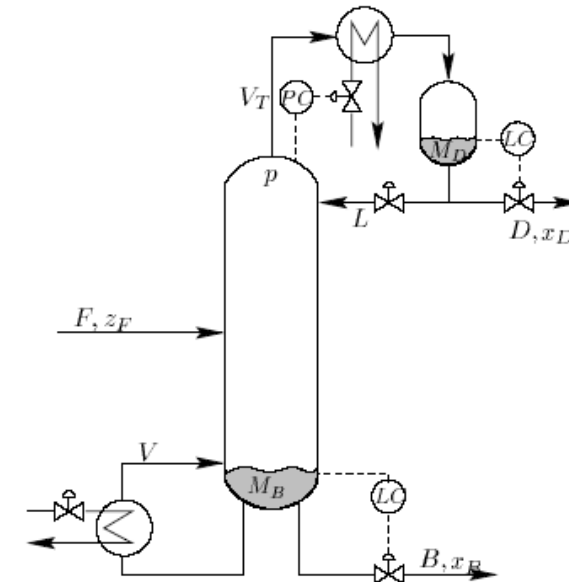
Flow constraints:  $\min \leq D, B, L \text{ etc.} \leq \max$

Column capacity (flooding):  $V \leq V_{\max}, \text{ etc.}$

Pressure: 1)  $p$  given (d) 2)  $p$  free (**u**):  $p_{\min} \leq p \leq p_{\max}$

Feed: 1)  $F$  given (d) 2)  $F$  free (**u**):  $F \leq F_{\max}$

- Optimal operation: Minimize  $J$  with respect to steady-state DOFs (**u**)



# Skogestad procedure for control structure design:

## I. Top Down

- Step S1: Define operational objective (cost  $J$ ) and constraints (easy!)
- Step S2: (a) Identify degrees of freedom and (b) optimize operation for disturbances
  - Usually not easy! So often based on process insight
- Step S3: Implementation of optimal operation
  - What to control? (primary CV's) (self-optimizing control)
- Step S4: Where set the production rate? (Inventory control)

## II. Bottom Up

- Step S5: Regulatory control: What more to control (secondary CV's)?
- Step S6: Supervisory control
- Step S7: Real-time optimization

## Step S2a: Degrees of freedom (DOFs) for operation

**IMPORTANT!**

**DETERMINES THE NUMBER OF VARIABLES TO CONTROL!**

- **No. of CV1 = No. of steady-state DOFs**

How many? NOT as simple as one may think!

To find all operational (**dynamic**) degrees of freedom:

- Count valves! ( $N_{\text{valves}}$ )
- “Valves” also includes adjustable compressor power, etc.  
Anything we can manipulate!

BUT: not all these have a (**steady-state**) effect on the economics

# How many **Steady-state** degrees of freedom (DOFs)?

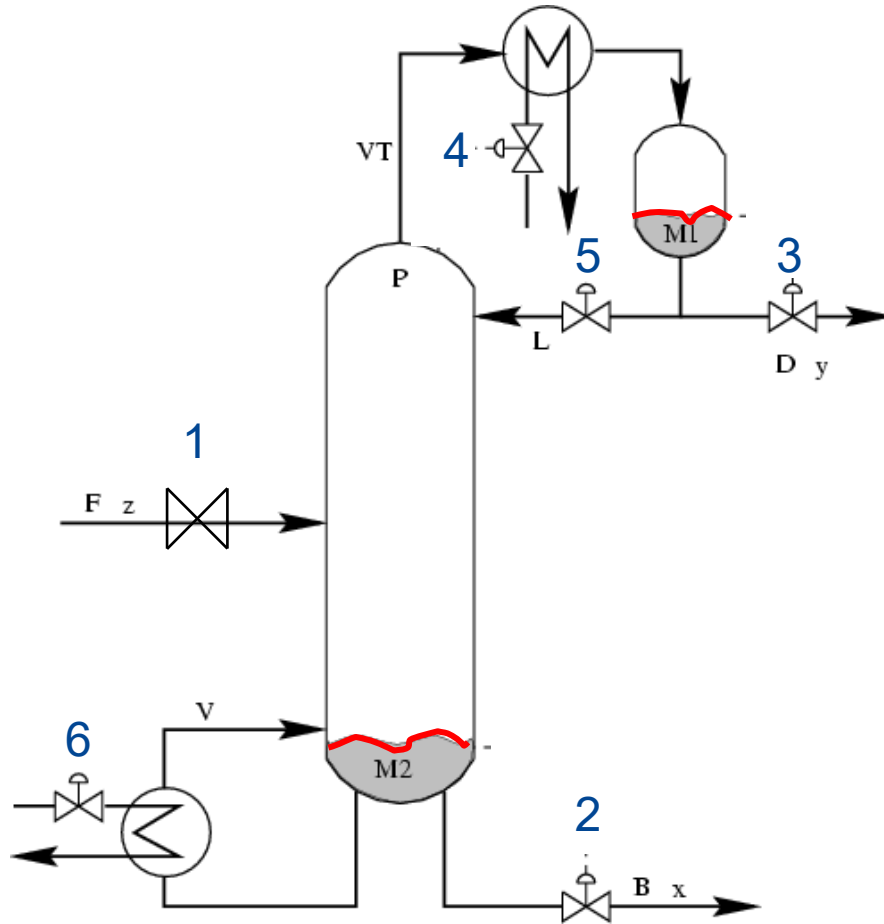
Methods to obtain no. of steady-state degrees of freedom ( $N_{ss}$ ):

1. Equation-counting
  - $N_{ss}$  = no. of variables – no. of equations/specifications
  - Very difficult in practice
2. **Valve-counting (easier!)**
  - $N_{ss} = N_{valves} - N_{0ss} - N_{specs}$
  - $N_{valves}$ : **include also variable speed for compressor/pump/turbine**
  - $N_{specs}$ : **Fixed variables (which are not later included in constraints)**
  - $N_{0ss}$  = **variables with no steady-state effect**
    - **Inputs/MVs with no steady-state effect (e.g. extra bypass)**
    - **Outputs/CVs with no steady-state effect that need to be controlled (e.g., liquid levels)**
3. Potential number for some units (useful for checking!)
4. Correct answer: Will eventually find it when we perform optimization



$$N_{ss} = N_{valves} - N_{0ss} - N_{specs}$$

## Example: Distillation column



$$N_{valves} = 6, \quad N_{0ss} = 2^*,$$

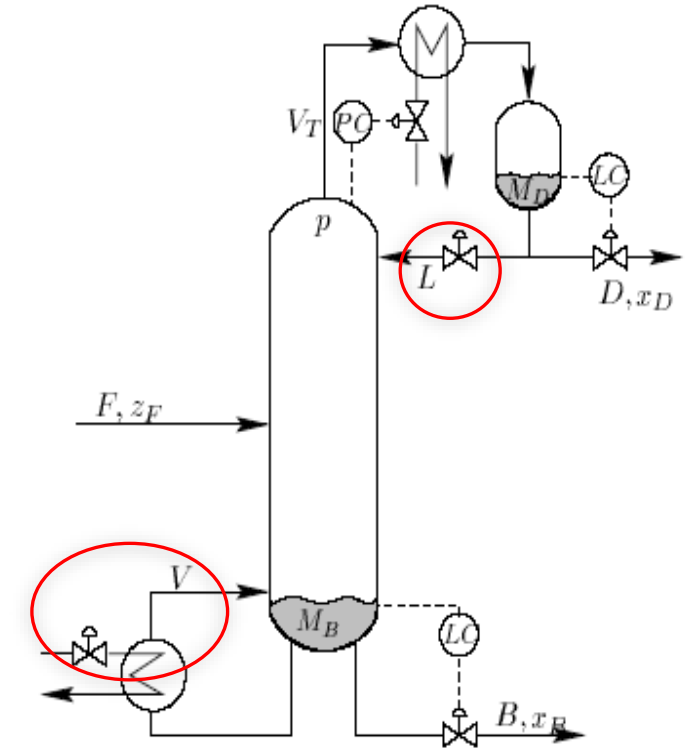
$$N_{DOF,SS} = 6 - 2 = 4 \text{ (including feed and pressure as DOFs.)}$$

If feed and pressure are fixed:  $N_{specs} = 2$  and  $N_{DOF,ss} = 4 - 2 = 2$

\* $N_{0ss}$  : no. controlled variables with no steady-state effect (here: levels M1 and M2)

## Steady-state DOFs

With levels and pressure controlled and given feed (LV-configuration):



NEED TO IDENTIFY 2 more CV's  
- Typical: Top and btm composition

# Step S2b: Optimize for expected disturbances

- What are the optimal values for our degrees of freedom  $u$  (MVs)?

$$J = \text{cost feed} + \text{cost energy} - \text{value products}$$

- Minimize  $J$  with respect to  $u$  for given disturbance  $d$  (usually steady-state):

$$\min_u J(x, u, d)$$

subject to:

- Model equations :  $\dot{x} = f(x, u, d) = 0$
- Operational constraints:  $g(x, u, d) \leq 0$

OFTEN VERY TIME CONSUMING

- Commercial simulators (Aspen, Unisim/Hysys) are set up in “design mode” and often work poorly in “operation (rating) mode”.
- Optimization methods in commercial simulators often poor
  - We can use Matlab or even Excel “on top”

# Step S2b: Optimize for expected disturbances

- Need good model, usually steady-state
- Optimization is time consuming! But it is offline
- **Main goal: Identify ACTIVE CONSTRAINTS (optimal to maintain)**
- A good engineer can often guess the active constraints:

Example:



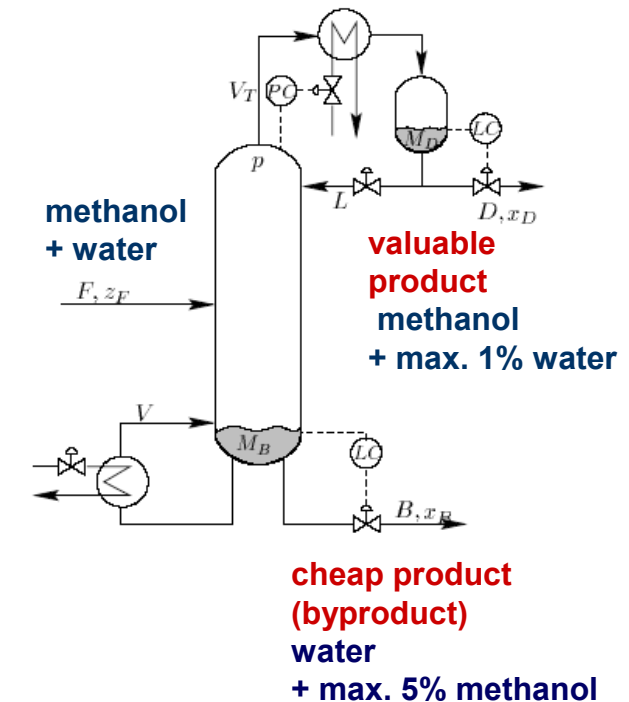
Cost  $J = T$  [h]

Constraint:  $v \leq 50$  km/h

**Control implementation:** Cruise control with setpoint 50 km/h (active constraint)

# Example Step S2b: Active constraints for distillation

- Both products (D, B) generally have purity specs
- **Rule 1: Purity spec. always active for valuable product**
  - Reason: 1. Maximize amount of valuable product (D or B)
    - Avoid product “give-away” (So “sell water as methanol”)
  - Reason 2: Save energy (because overpurification costs energy)
- **Rule 2: May overpurify (not control) cheap product**
  - Reason: Increase amount of valuable product (“reduce loss of methanol in bottom product”)
  - This typically results in an unconstrained optimum because overpurification costs energy (“optimal purity of cheap product”)



## **Step S2b: Optimize for expected disturbances**

$$\min J = \text{cost feed} + \text{cost energy} - \text{value products}$$

**Generally:** Two main cases (modes) depending on market conditions:

**Mode 1 (low product price). Given throughput (feed rate)**

**Mode 2 (high product price). Maximum production (more constrained)**

Comment: Depending on prices, Mode 1 may include many subcases (active constraints regions)

# Mode 1. Given feedrate

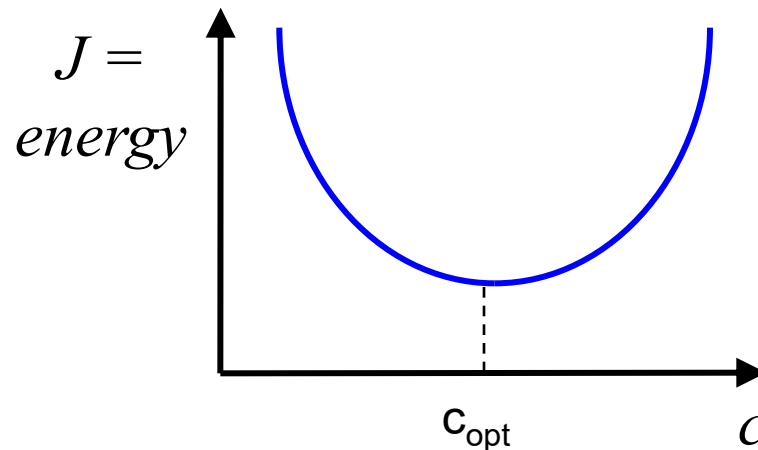
Amount of products is then usually indirectly given and

$$J = \underbrace{\text{cost feed} - \text{value products}}_{\text{Often constant}} + \text{cost energy}$$

Often constant

Optimal operation is then usually *unconstrained*

“maximize efficiency (energy)”



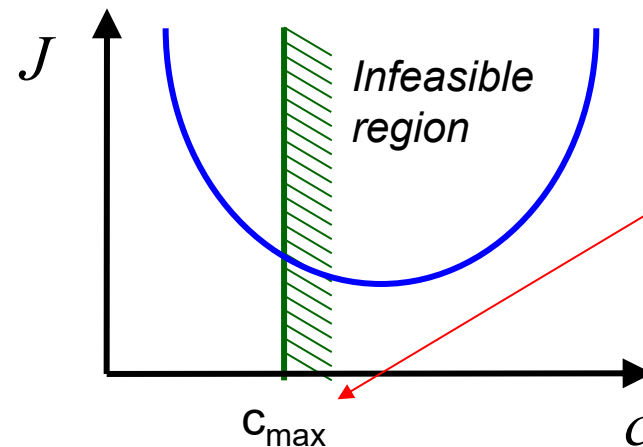
Control:

- Operate at optimal trade-off
- NOT obvious what to control
- CV = Self-optimizing variable

# Mode 2. Maximum production

$$J = \text{cost feed} + \text{cost energy} - \text{value products}$$

- Assume feed rate is degree of freedom
- Assume products much more valuable than feed
- Optimal operation is then to maximize product rate
- **“max. constrained”, prices do not matter**



Control:

- Focus on tight control of bottleneck
- “Obvious what to control”
- CV = ACTIVE CONSTRAINT
- $CV_s = c_{\max}$

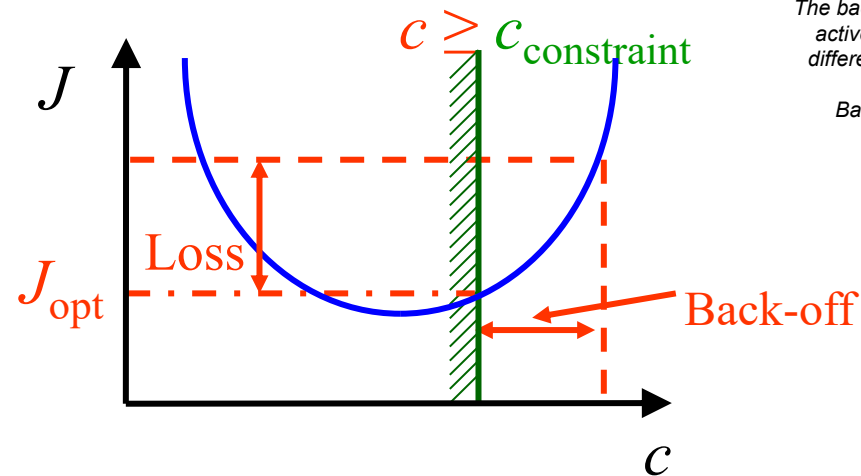
## Step S3. Implementation of optimal operation

- Assume we have analyzed the optimal way of operation. How should it be implemented?
- **What to control?** (primary CV's)
  1. Active constraints
  2. Self-optimizing variables (for unconstrained degrees of freedom)



# 1. Control of Active output constraints

Need back-off



- a) If constraint can be violated dynamically (only average matters)
  - **Required Back-off** = "measurement bias" (steady-state measurement error for  $c$ )
- b) If constraint cannot be violated dynamically ("hard constraint")
  - **Required Back-off** = "measurement bias" + maximum dynamic control error

Want tight control of hard output constraints to reduce the back-off. **"Squeeze and shift"-rule**

# Motivation for better control of active constraints: Squeeze and shift rule

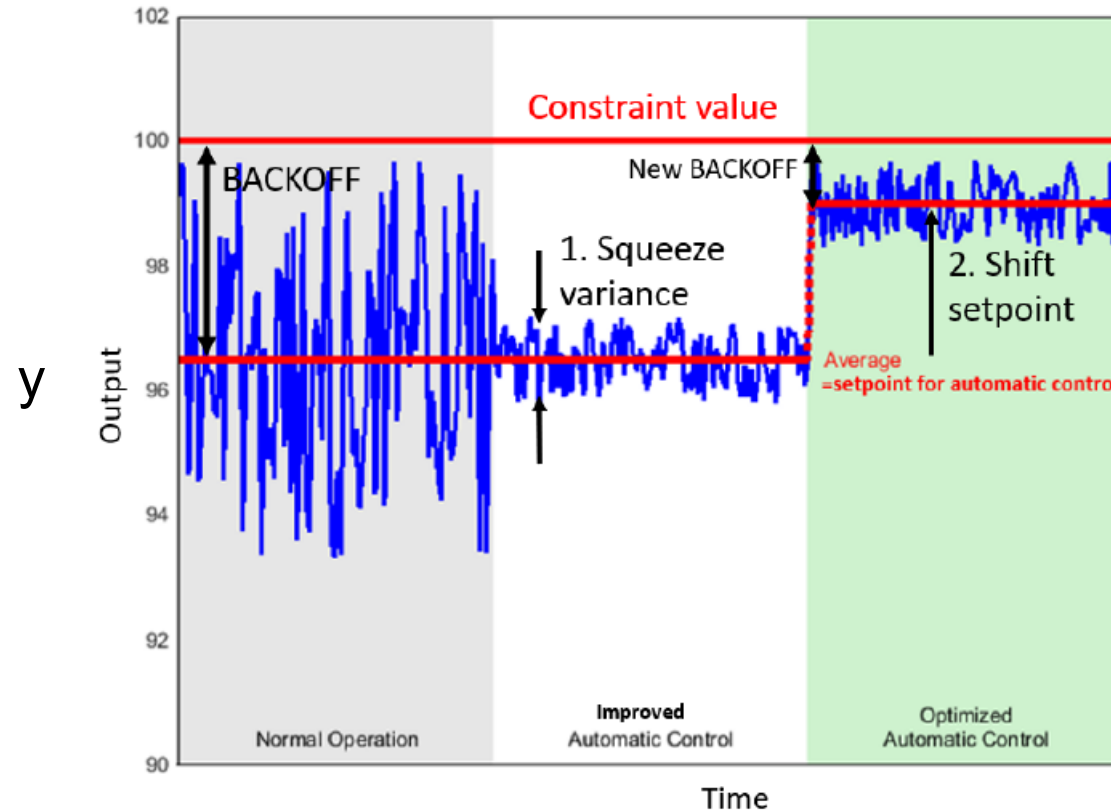
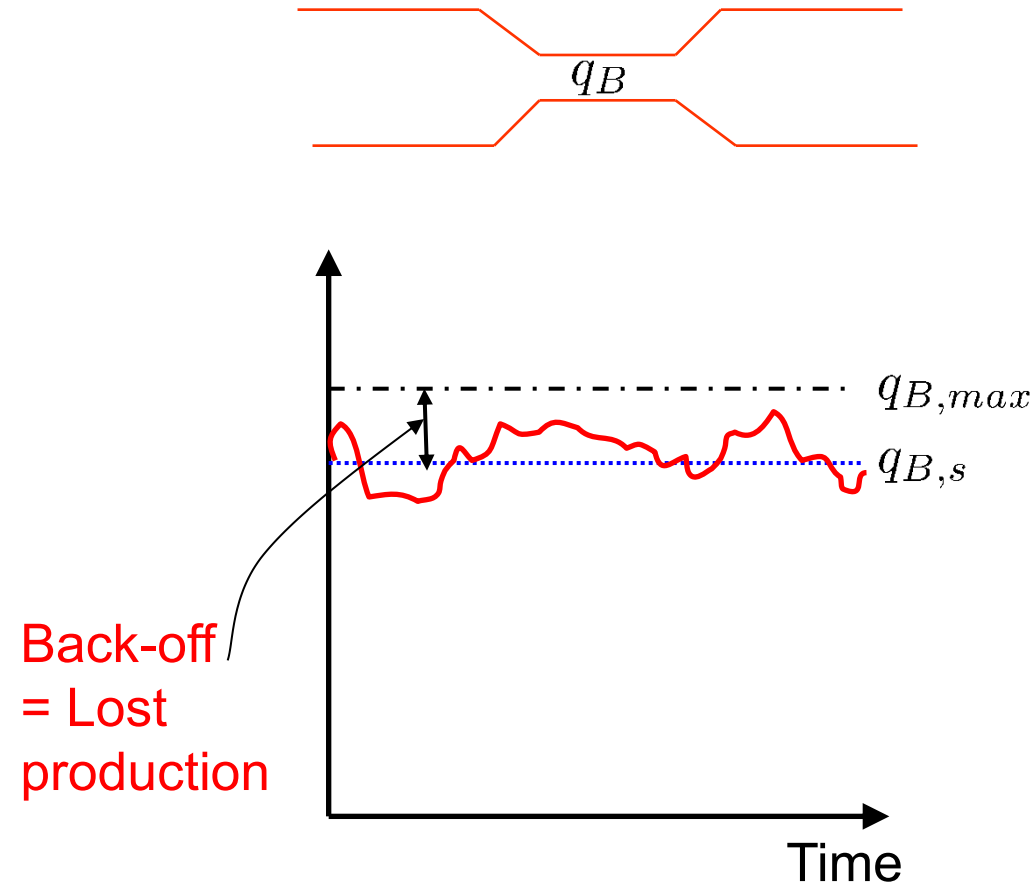


Figure 8: Squeeze and shift rule: Squeeze the variance by improving control and shift the setpoint closer to the constraint (i.e., reduce the backoff) to optimize the economics (Richalet et al., 1978).

# Example: max. throughput.

Want tight bottleneck control to reduce backoff!

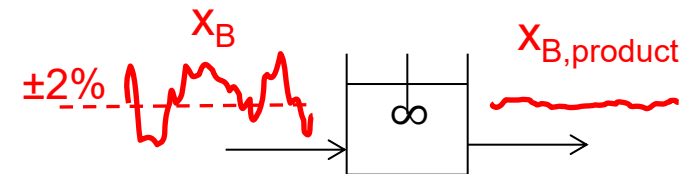


# Example active constraint: purity on distillate



$x_B$  = purity of product > 95% (min.)

- $D_2$  directly to customer (**hard** constraint)
  - Measurement error (bias): 1%
  - Control error (variation due to poor control): 2%
  - Backoff = 1% + 2% = 3%
  - Setpoint  $x_{Bs}$  = 95 + 3% = 98% (to be safe)
  - Can reduce backoff with better control (“squeeze and shift”)
- $D_2$  to large mixing tank (**soft** constraint)
  - Measurement error (bias): 1%
  - Backoff = 1%
  - Setpoint  $x_{Bs}$  = 95 + 1% = 96% (to be safe)
  - Do not need to include control error because it averages out in tank



## 2. Unconstrained optimum

Control “self-optimizing” variable! (More on this soon!)

- Which variable is best?
- Often not obvious

What are good self-optimizing variables?

1. Optimal value of CV is constant
2. CV is “sensitive” to MV (large gain)

Note: Tight control of the self-optimizing variable is usually not important because optimum should be flat.

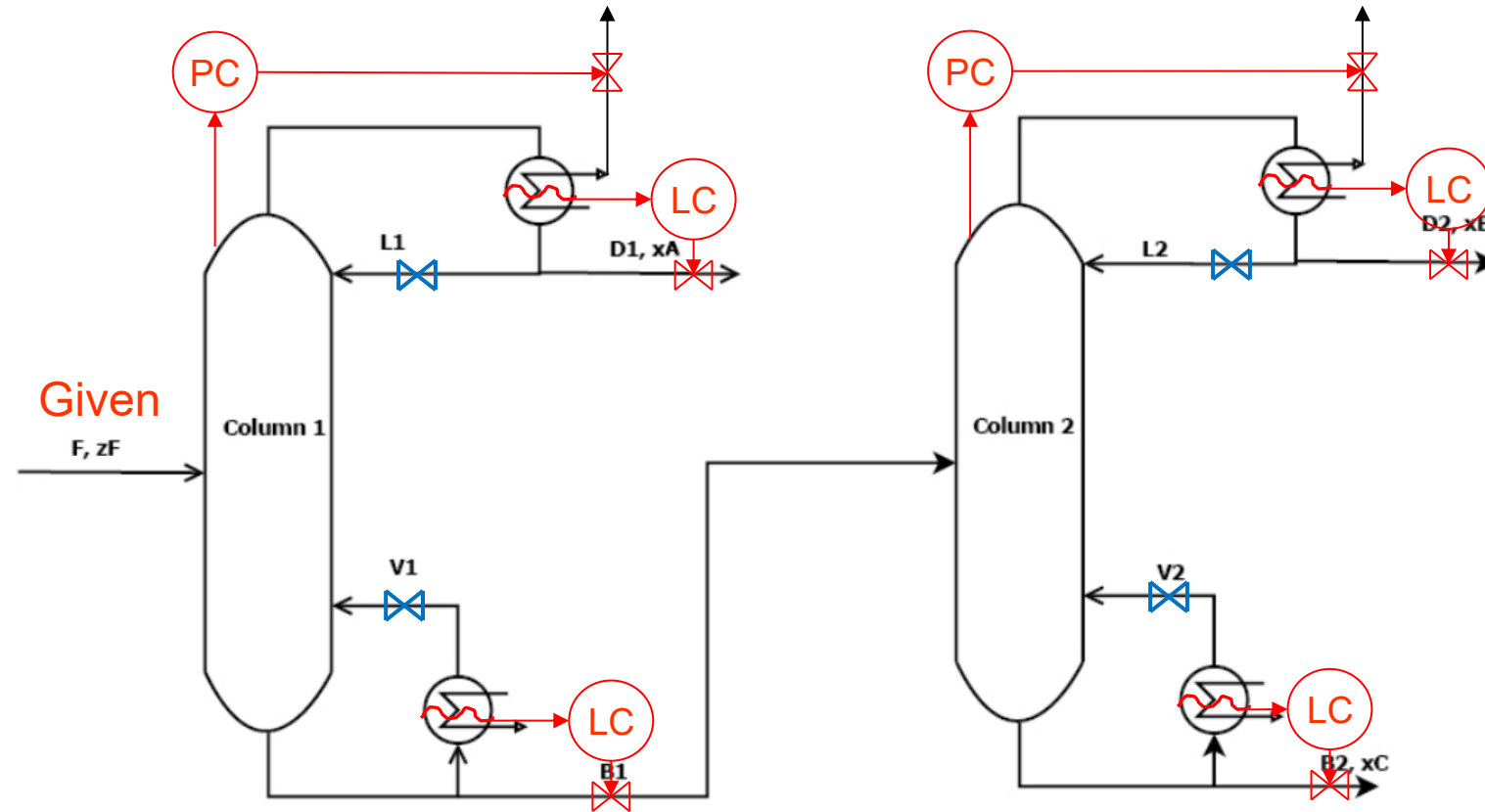
# Conclusion optimal operation

## ALWAYS:

1. Control active constraints and control them tightly!!
  - Good times: Maximize throughput → tight control of bottleneck
2. Identify “self-optimizing” CVs for remaining unconstrained degrees of freedom
  - Use offline analysis to find expected operating regions and prepare control system for this!
    - One control policy when prices are low (nominal, unconstrained optimum)
    - Another when prices are high (constrained optimum = bottleneck)

**ONLY** if necessary: consider RTO on top of this

# Example Steps 1, 2 & 3: Distillation columns in series

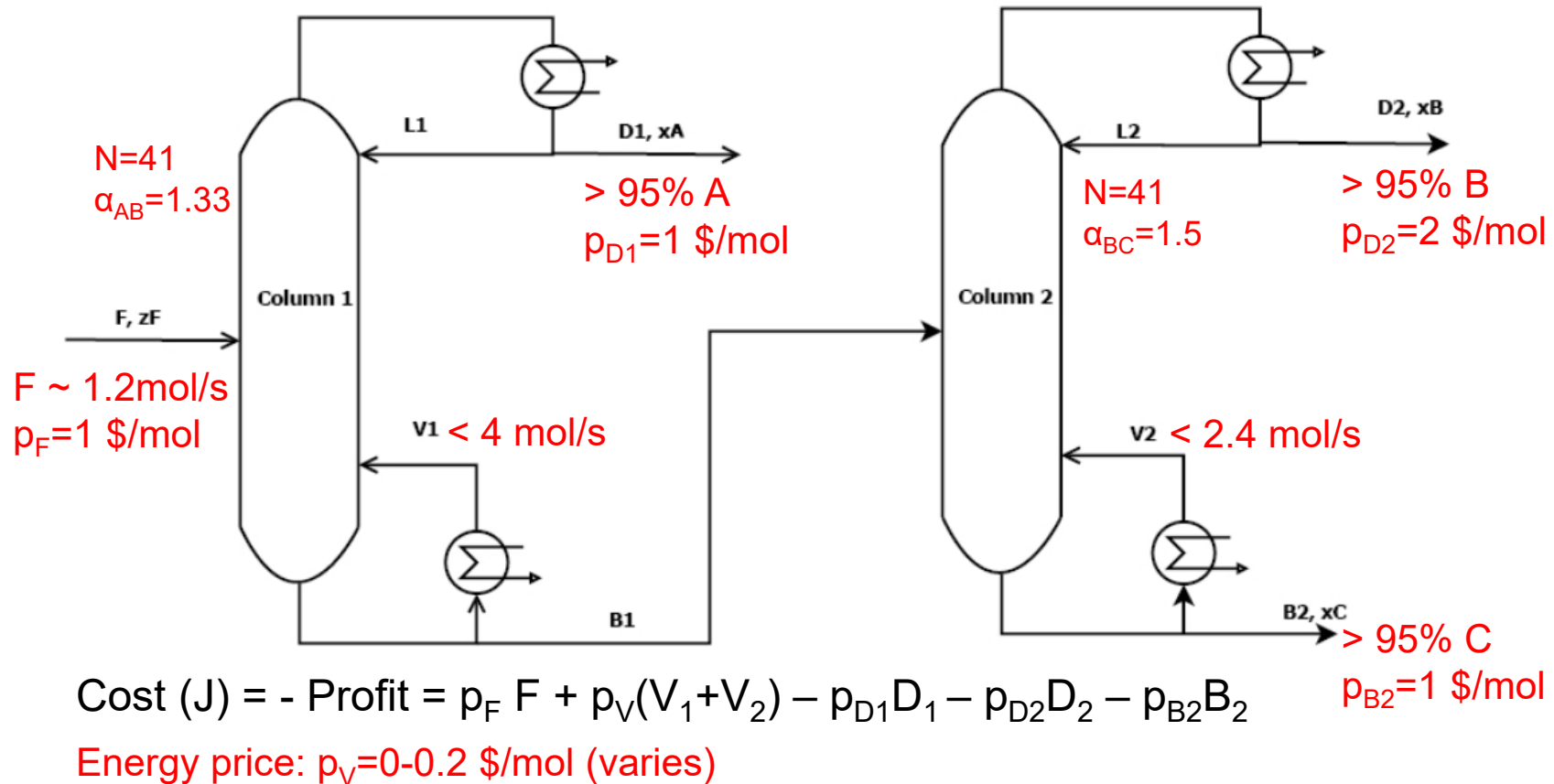


Given feed and pressures: We have 4 remaining steady-state MVs (L1, V1, L2, V2)  
What more should we control?  
HINT: CONTROL ACTIVE CONSTRAINTS

Red: Basic regulatory loops

# Step S1: Cost and constraints

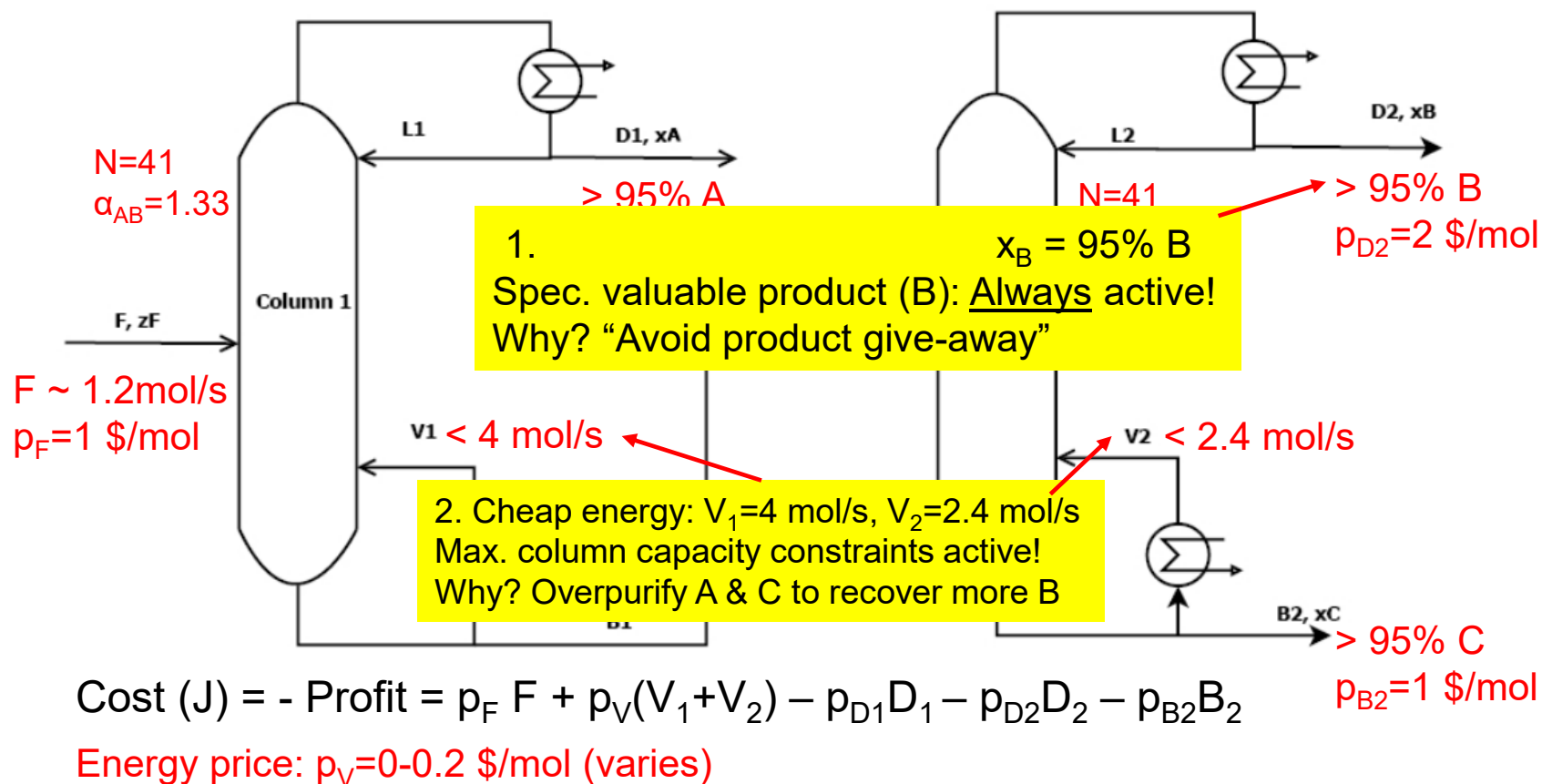
- 4 steady-state DOFs (e.g., L and V in each column)
- 5 (important) constraints: 3 product composition + 2 max. heat input



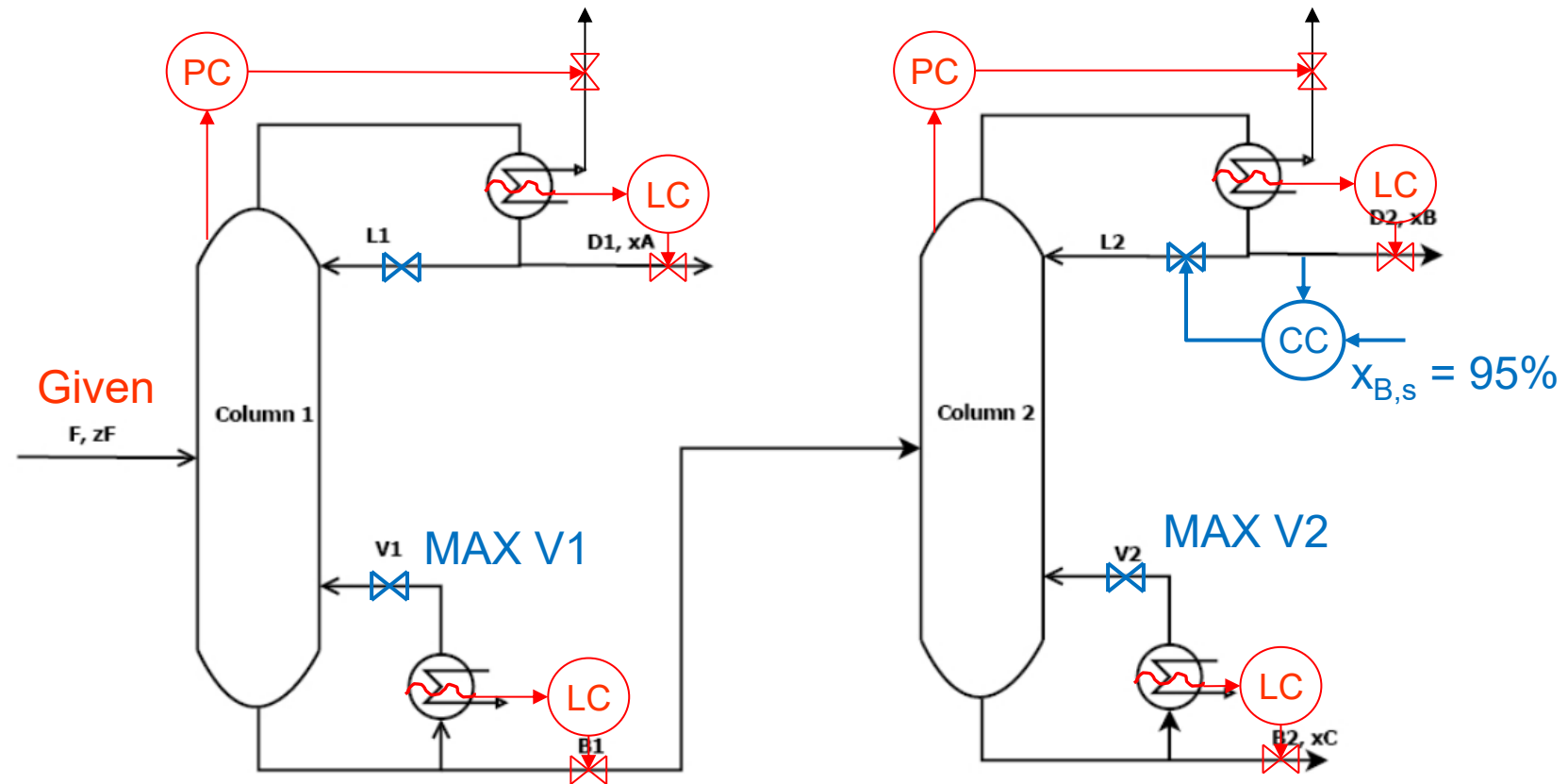


# Step S2. Optimal operation

With given feed and pressures (disturbances): 4 steady-state DOFs  
(e.g., L and V in each column)



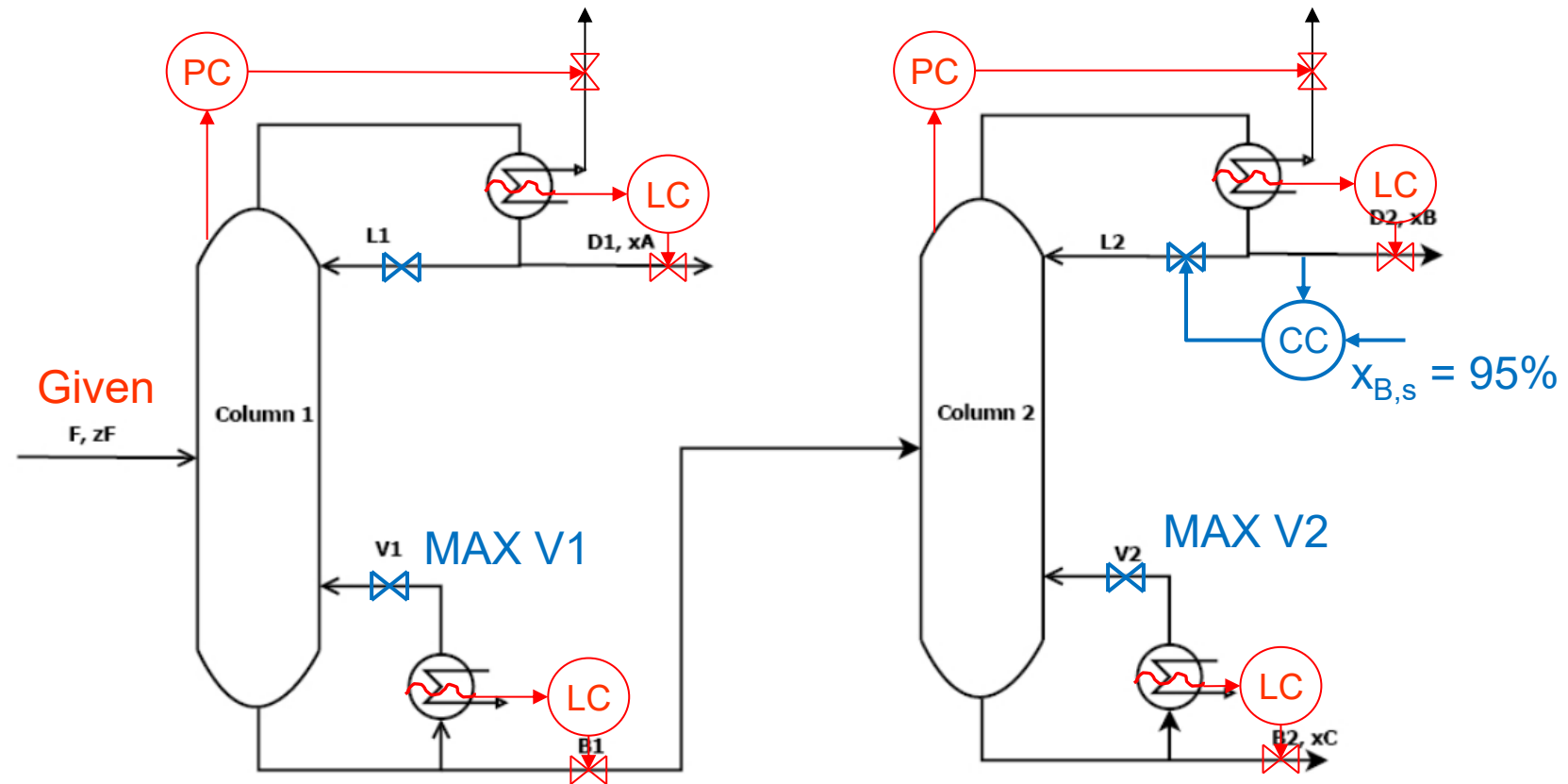
## Step S3: Control 3 Active constraints:



**L1 not used. What more should we control?**

Optimal to “overpurify” D1 - but optimal overpurification is **unconstrained** and varies with feedrate.  
LOOK FOR “SELF-OPTIMIZING” CVs = Variables we can keep constant

## Step S3: Control 3 Active constraints:

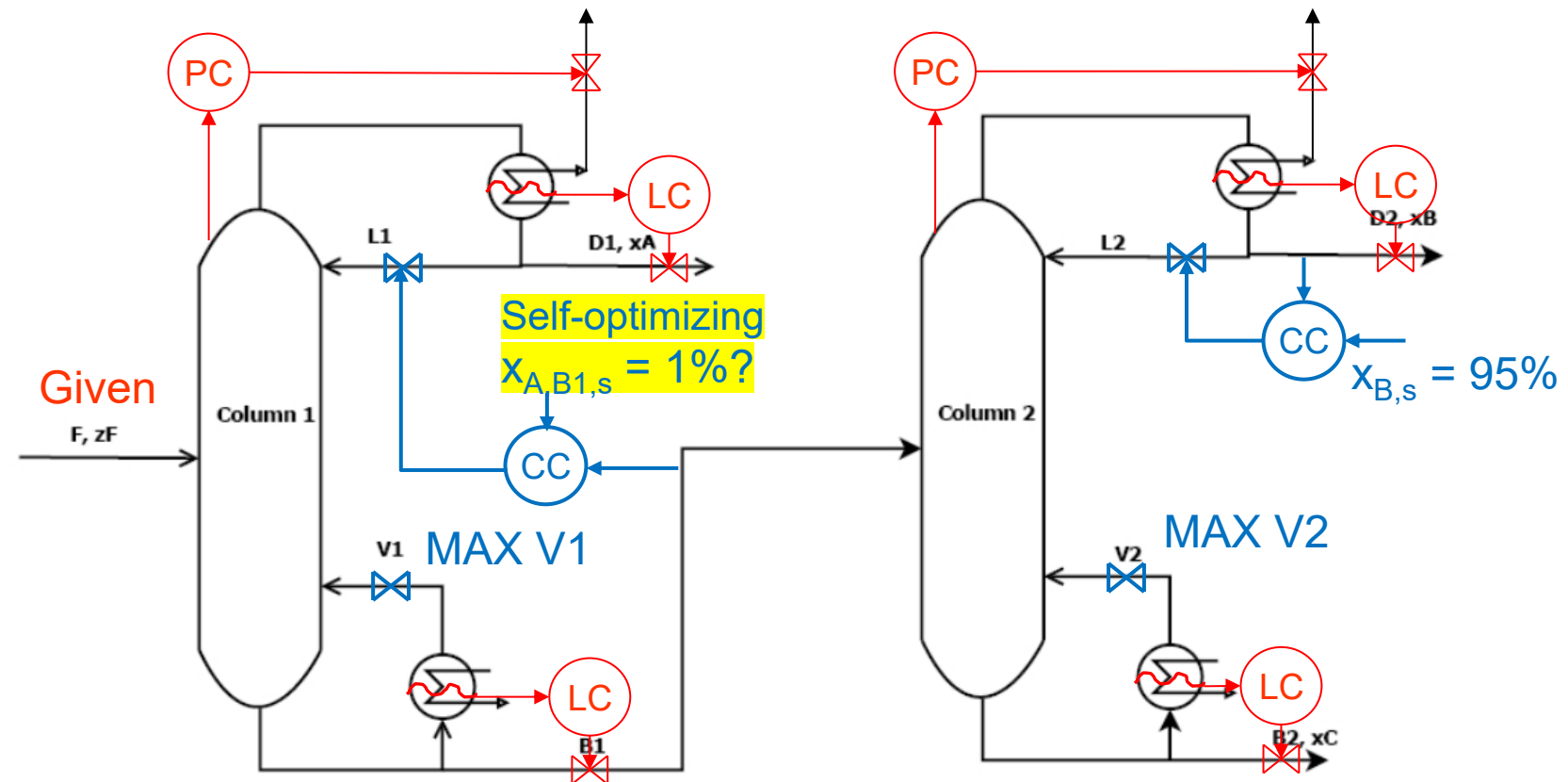


Red: Basic regulatory loops

What CV should L1 be paired with?

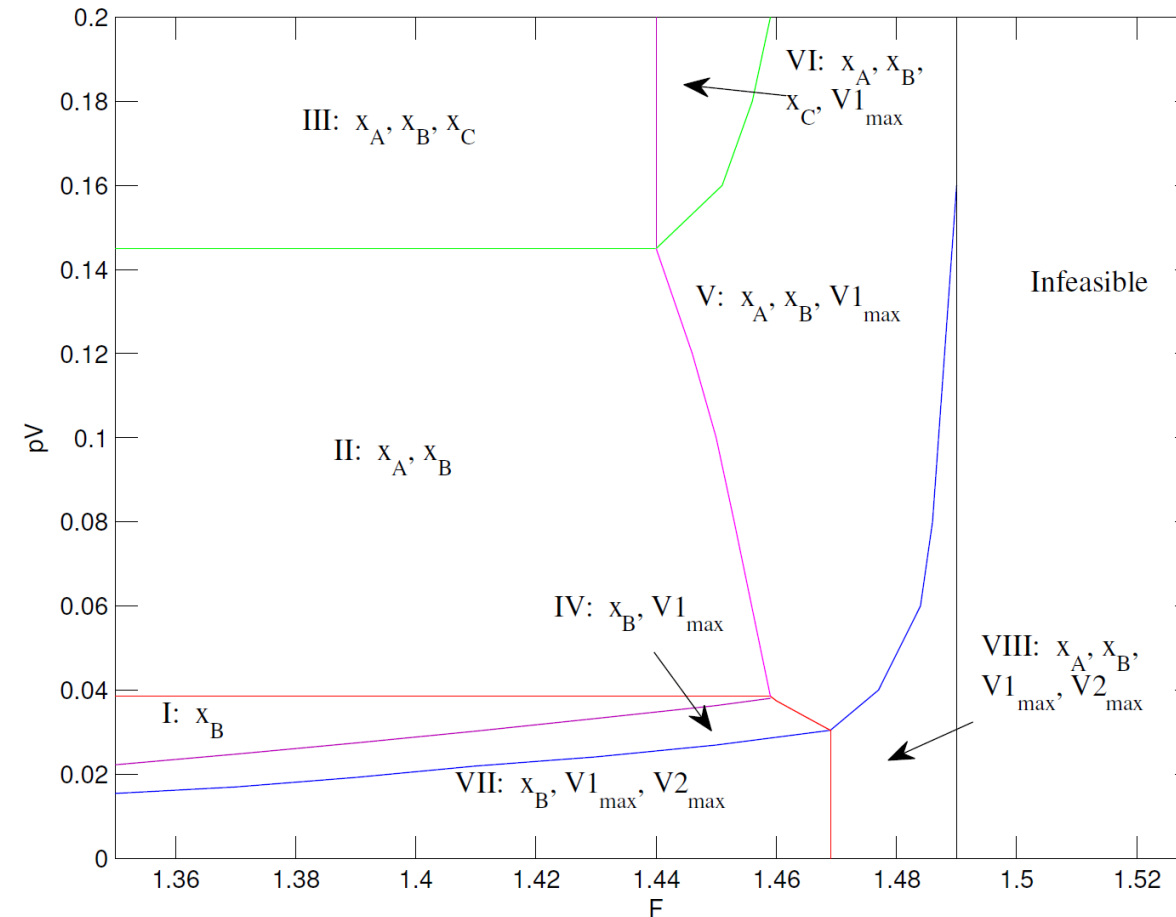
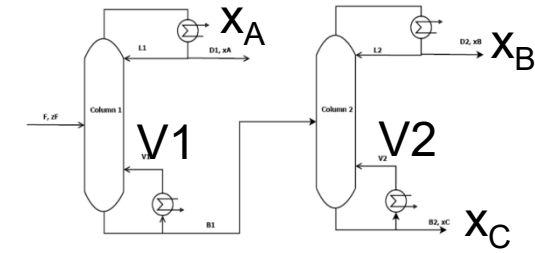
- Not:  $CV = x_A$  in D1! (why?  $x_A$  should vary with F!)
- Maybe: constant L1? ( $CV = L1$ )
- Better:  $CV = x_A$  in B1? Self-optimizing?

## Step S3: Control 3 Active constraints + 1 self-optimizing



Red: Basic regulatory loops

# Vary feedrate (F) and energy price (pV): 8 active constraint regions



- The figure shows the active constraints (between 1 and 4) in each region.  $x_B$  in D2 is always active.
- On the previous slide we only considered region VII («cheap energy» with pV small).
- In the «infeasible» region there are 5 constraints ( $x_A$ ,  $x_B$ ,  $x_C$ ,  $V1_{max}$ ,  $V2_{max}$ ) but only 4 DOFs. Must reduce F

# How many active constraints regions?

- Maximum:  $2^{n_c}$   
where  $n_c$  = number of constraints

**Distillation**

$$n_c = 5$$

$$2^5 = 32$$

BUT there are usually fewer in practice

- Certain constraints are always active (reduces effective  $n_c$ )
- Only  $n_u$  can be active at a given time  
 $n_u$  = number of MVs (inputs)
- Certain constraints combinations are not possible
  - For example, max and min on the same variable (e.g. flow)
- Certain regions are not reached by the assumed disturbance set

**$x_B$  always active**

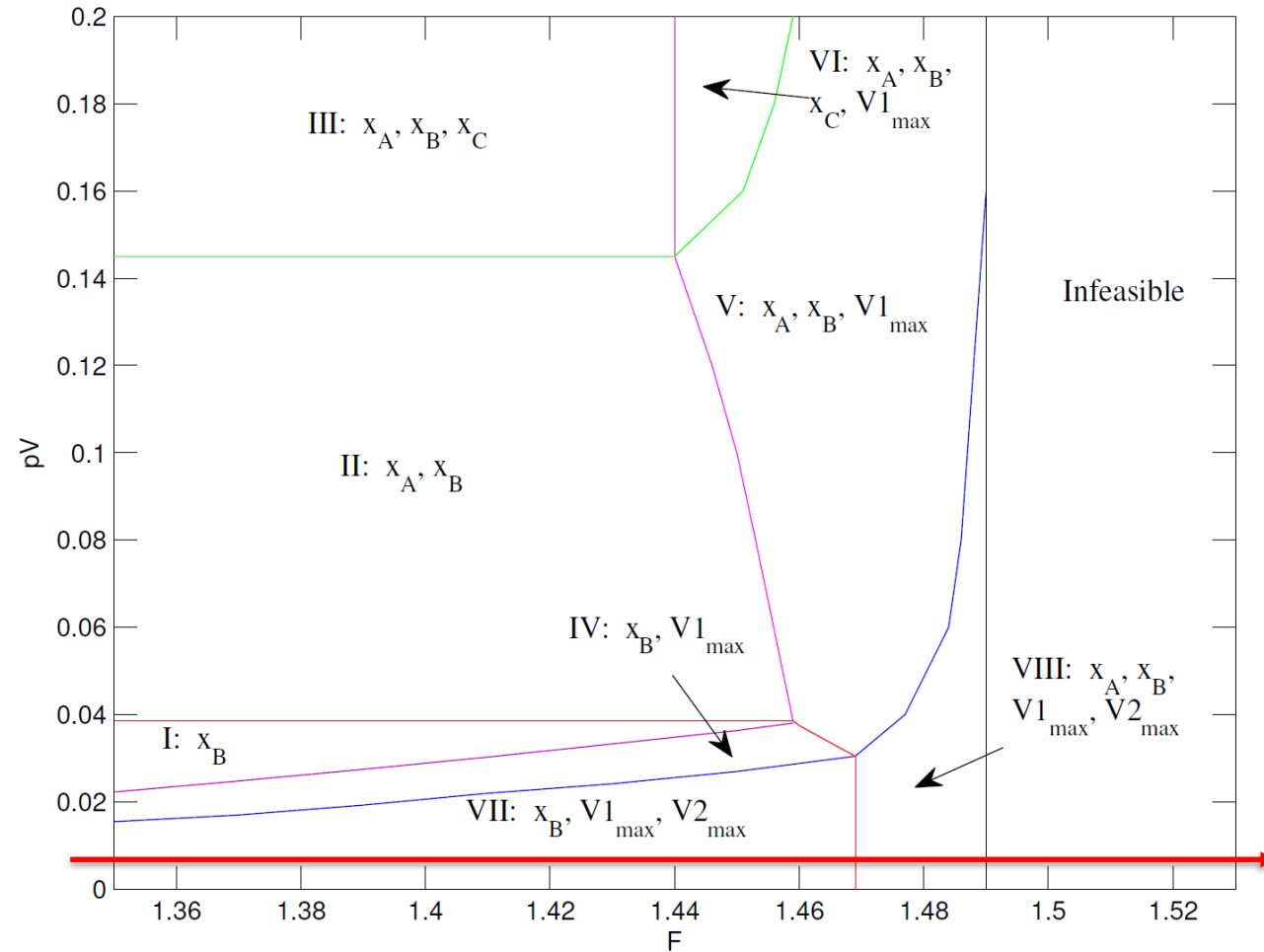
$$2^4 = 16$$

$$-1 = 15$$

**In practice = 8**

This seems complicated..... But knowledge about all regions is rarely (if ever) needed....  
In practice: We use the control system to switch when constraints are encountered.....  
It's much simpler and in many cases optimal.... Try

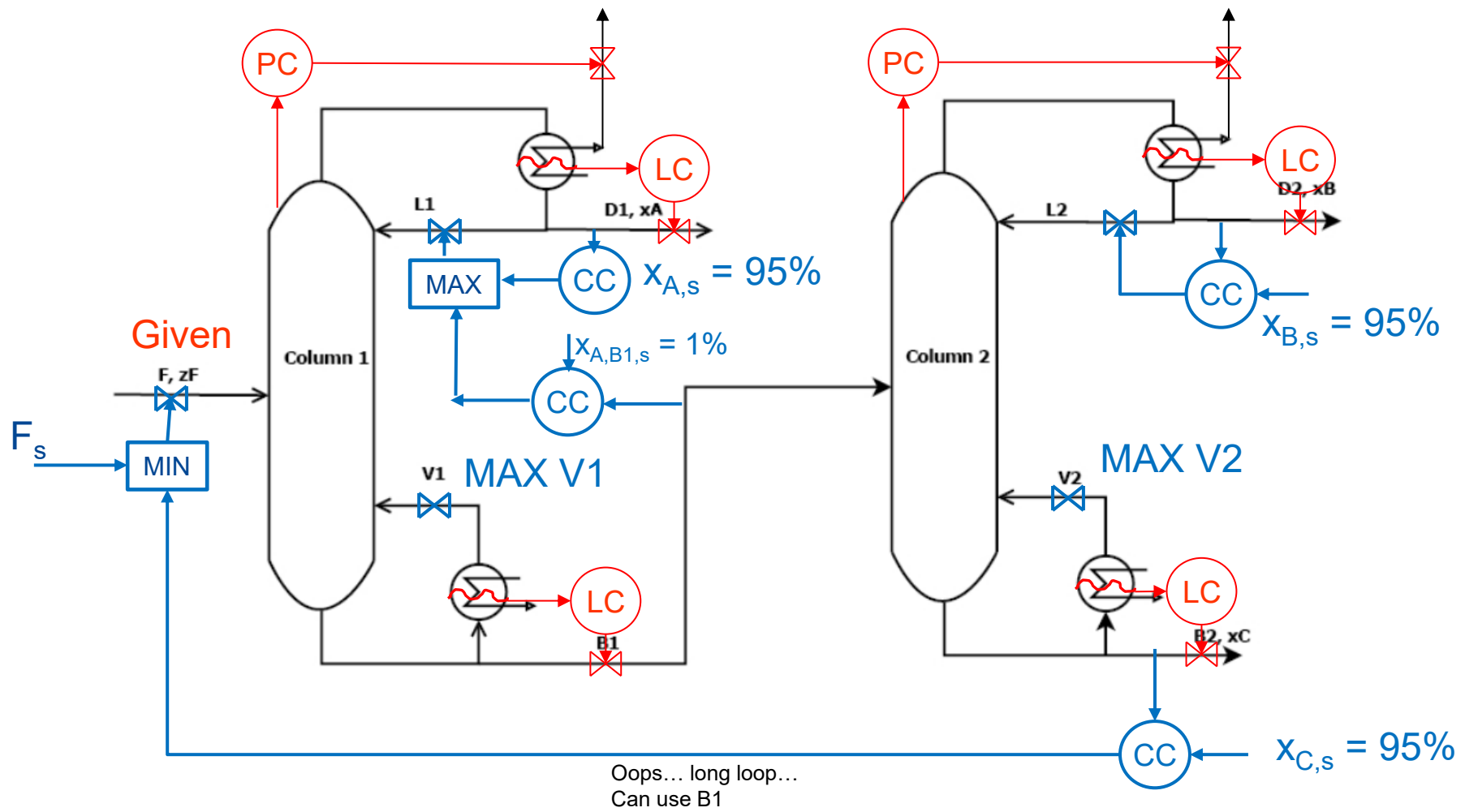
# Preview: How handle increase in $F$ (still with low $pV$ )?



How to control in three regions (VII, VIII and Infeasible)?

# Preview: Control of distillation columns in series in three regions

(but finding a simple control structure with constant setpoints that works in all regions is not possible;  
One solution: 4 composition loops + RTO that optimizes composition setpoints)



Red: Basic regulatory loops