Nonlinear input transformations for disturbance rejection, decoupling and linearization

Sigurd Skogestad

Cristina Zotica

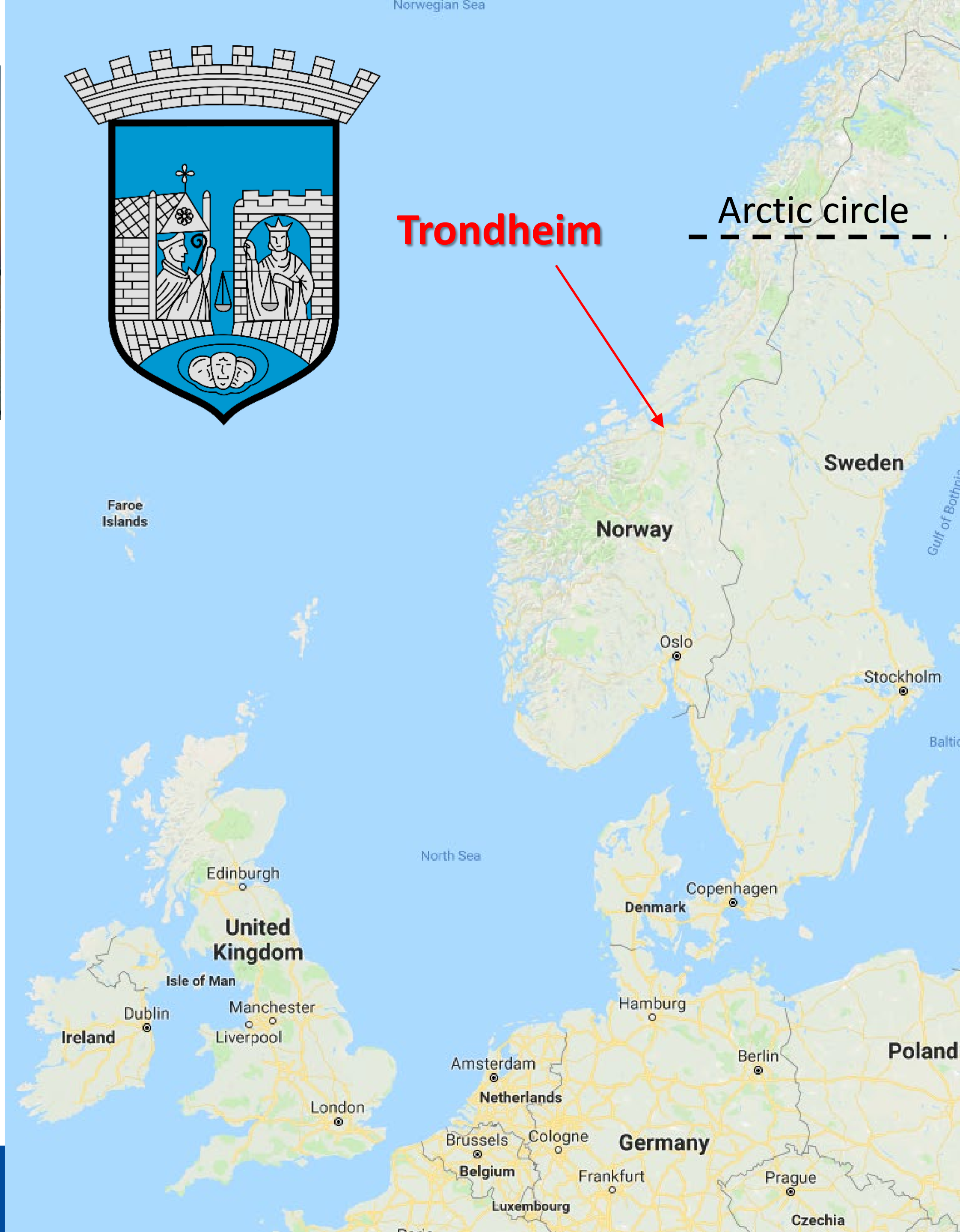
Department of Chemical Engineering

Norwegian University of Science and Technology (NTNU)

Trondheim

Plenary talk (virtual) at Control Conference Africa (CCA2021), 08 December 2021, Magaliesberg, South Africa

Thanks to Nicholas Alsop, Christian Holden and Krister Forsman


Norwegian Sea
Trondheim
Arctic circle
Sweden
Norway
Faroe Islands
Oslo
Stockholm
Baltic
Gulf of Bothnia
North Sea
Edinburgh
Copenhagen
Denmark
United Kingdom
Isle of Man
Manchester
Liverpool
Dublin
Ireland
Hamburg
Amsterdam
Netherlands
Berlin
Poland
London
Brussels
Belgium
Cologne
Germany
Frankfurt
Luxembourg
Prague
Czechia

“The goal of my research is to develop simple yet rigorous methods to solve problems of engineering significance”

Outline

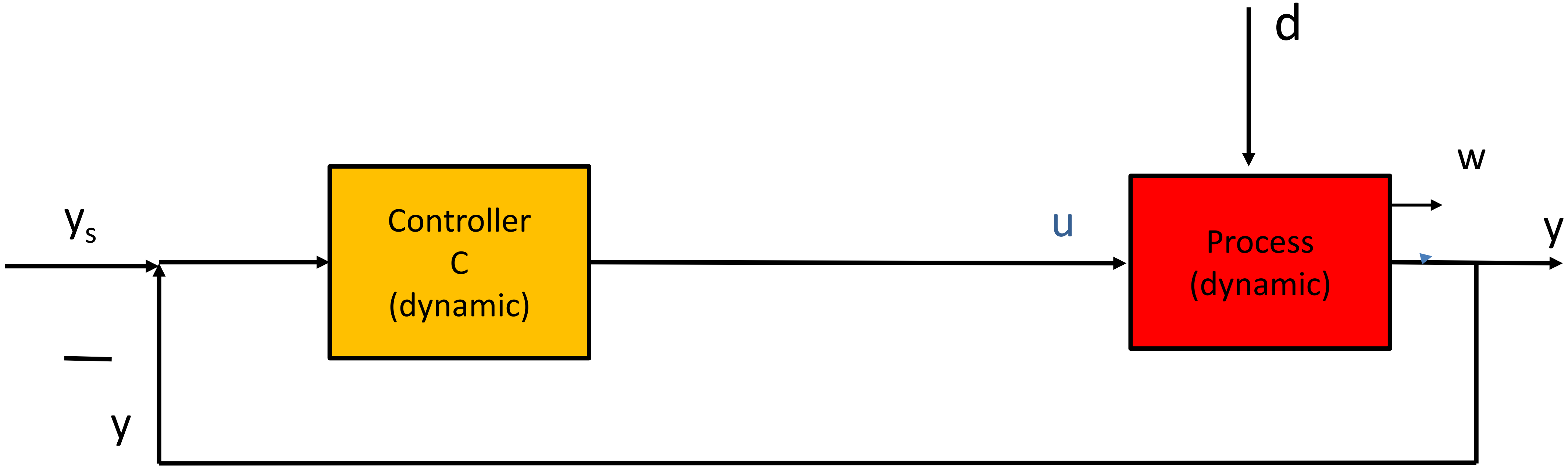
- What are input transformations?
  - Tranformed input (v) = static function of physical input (u)
- Inverting the input transformation
  - Exact inverse (low relative order)
  - Approximate inverse using feedback (cascade control)
- Systematic approaches for deriving input transformation
  - From static model
  - From dynamic model
  - Comparison with «feedback linearization»
  - Chain of transformations (Exact inverse for systems of higher order)
- Potential internal instability with exact inverse
  - Linear analysis,
  - Bode stability condition
- Output transformation
- Discussion/Conclusion

Motivation

- Industry frequently uses static model-based calculations blocks
- … and sometimes combined with cascade control
- Idea: Use physical insight or model equations to derive control strategy

- But no theory for when and how to use

Definition of transformed inputs



u = physical input
y = controlled output
w = other measured output (state)
d = disturbance

Shinskey (1981): "There is no need to be limited to single measurable (y) or manipulable variables (u). If a more meaningful variable happens to be a mathematical combination of two or more measurable or manipulable variables, there is no reason why it cannot be used."

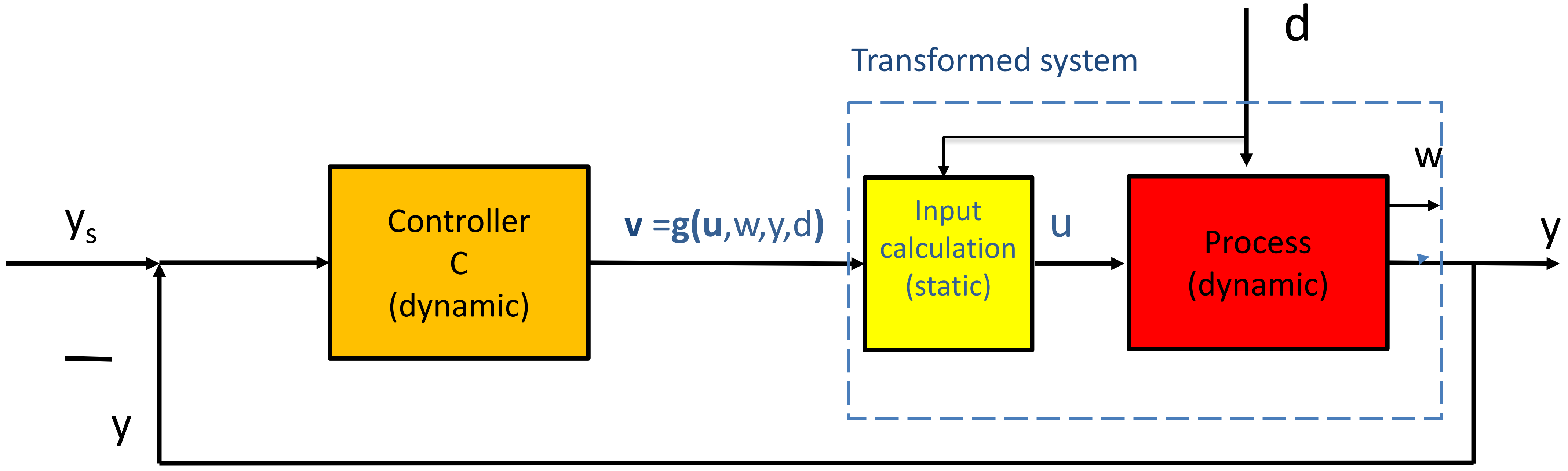
Transformed input: **v=g(u,w,y,d)** (static function)

Here d and w are assumed measured

Transformed output: $z=g_z(y,w,d)$

NTNU

Use of transformed inputs



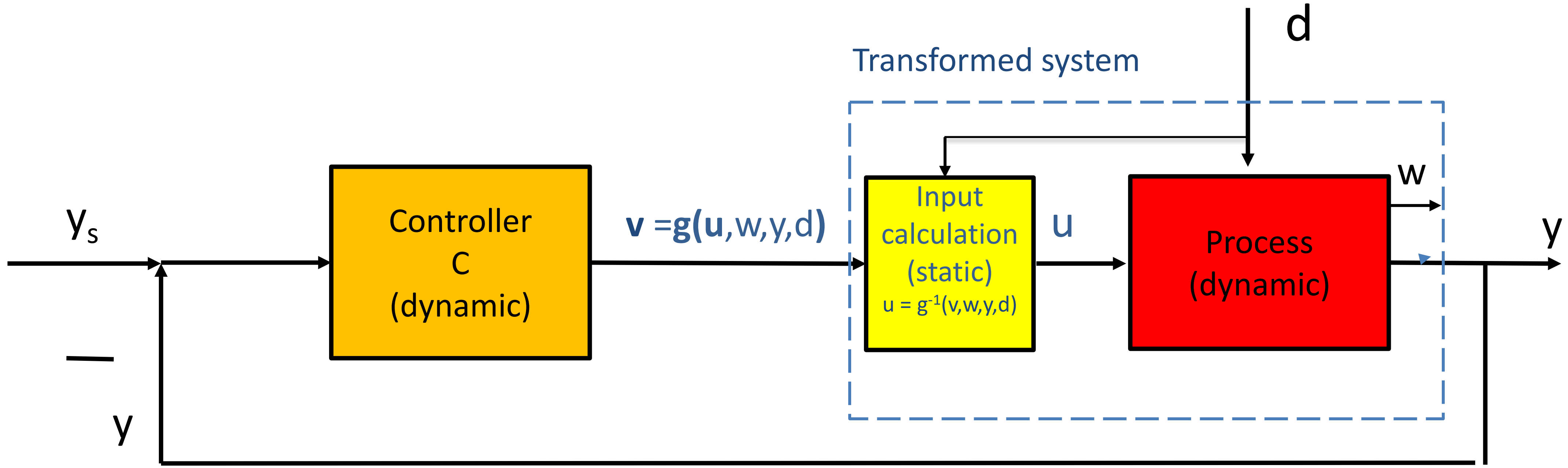
Transformed input v =g(u,w,y,d)

- Replaces the physical input u for control of y.
- Aim: **Transformed system** is easier to control
  - May include:
    - Decoupling
    - Linearization
    - Feedforward

Examples

- $v = u/d$
- $v = u_1/u_2$
- $v = u_1 + u_2$
- $v = w(u)$ -> Cascade control

Use of transformed inputs requires inversion

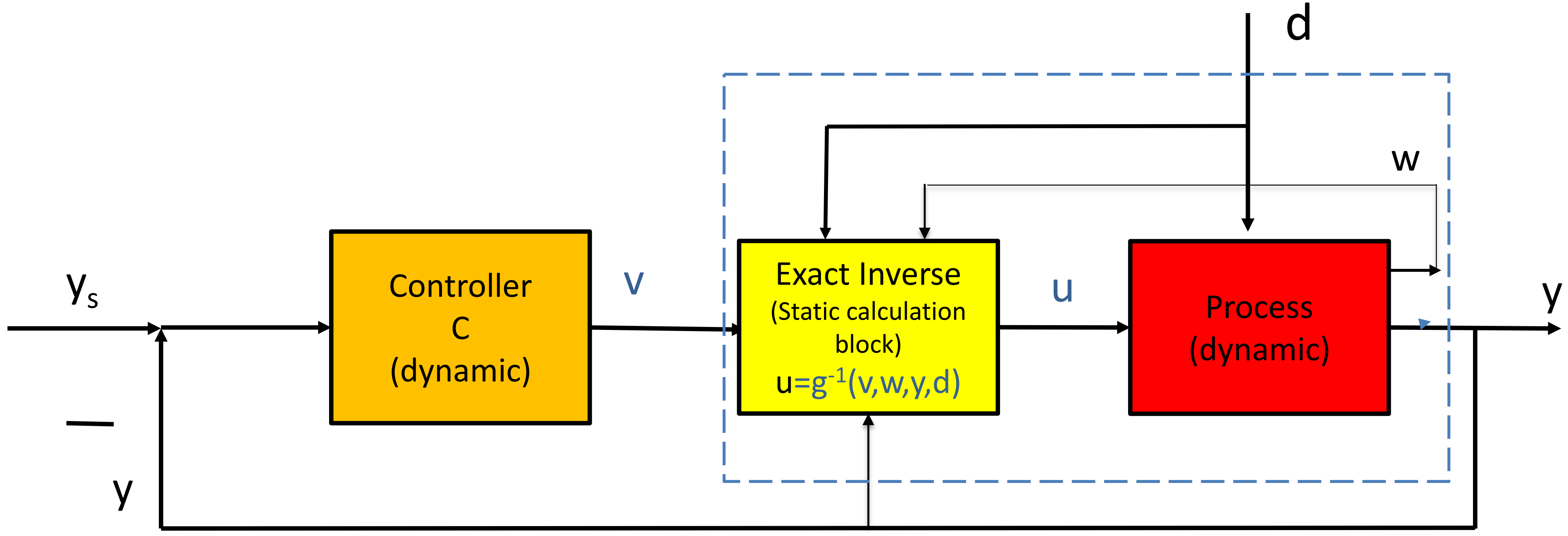


Input calculation block: Need to "invert / reverse" the transformation:

$u = g^{-1}(v,w,y,d)$

- Two main options:
  - Exact inverse
  - Approximate inverse by feedback

1. EXACT INVERSE

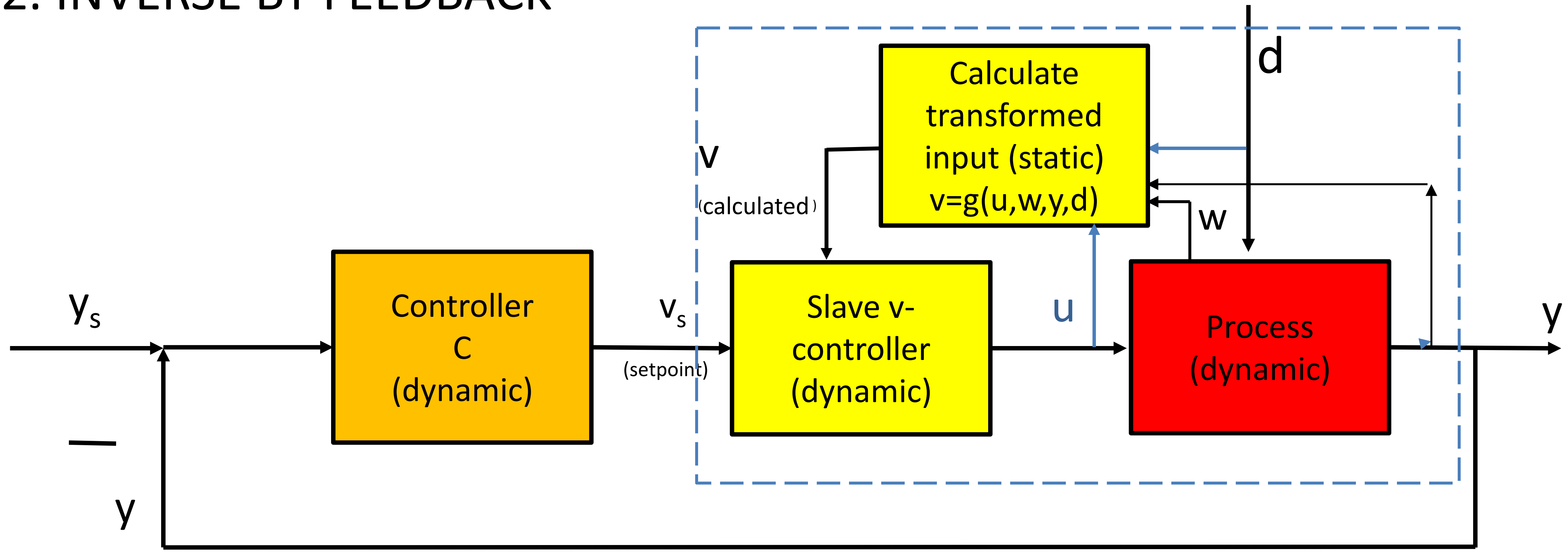


Exact inverse requires that v=g(u,w,y,d) depends explicitly on u

Potential problems:

- Inverse may be complicated, easy to do mistakes
- May get internal instability (because of feedback from w and y)

2. INVERSE BY FEEDBACK



Only option when v=g(w(u),y,d) does not depend explicitly on u. *Example: v = w*

Other advantages:

- Avoid internal instability with exact inverse
- Avoid complicated inverse (and reduce errors!)

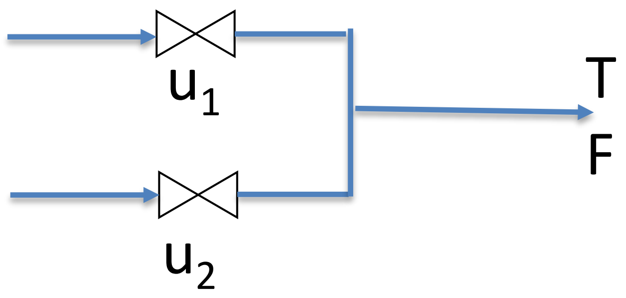
Disadvantages:

- Inverse not perfect dynamically (need fast slave controller)

Examples of transformed inputs

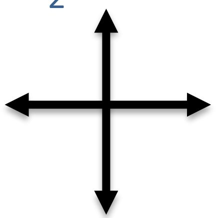
- Example 1: Mixing process (exact inverse)
- Example 2 (Industrial): Control of reactor temperature (inverse by feedback)

Example 1: Mixing of hot (u_1) and cold (u_2) water

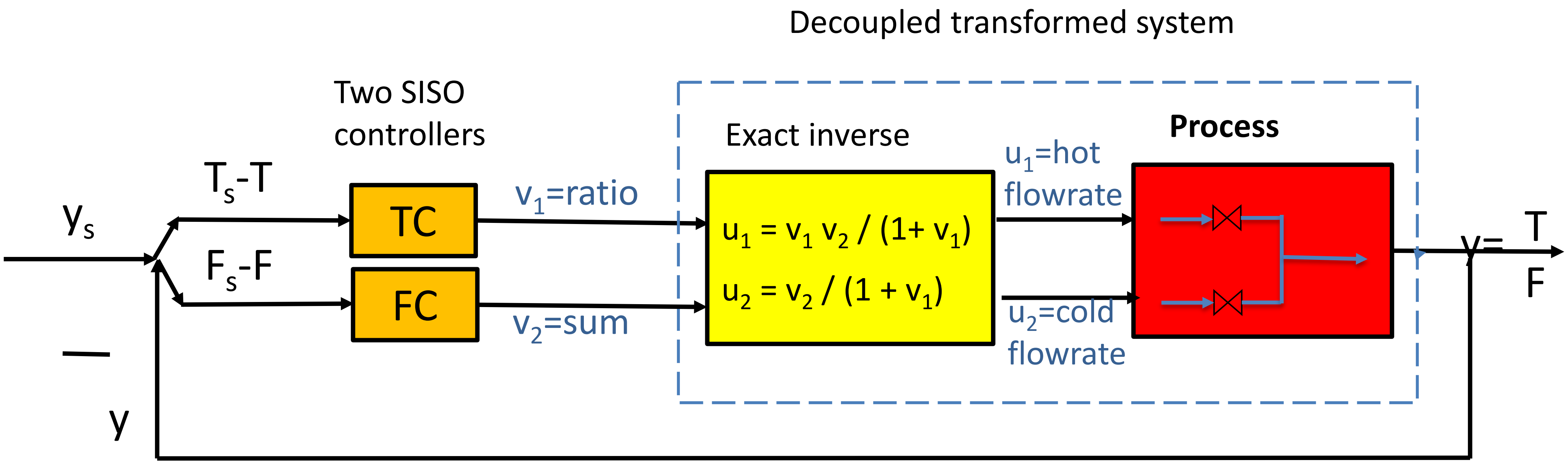


- Want to control
  - y_1 = Temperature T
  - y_2 = total flow F
- Want to use two SISO PI-controllers
  - TC
  - FC
- Get decoupled response with transformed inputs
  - TC sets flow ratio, $v_1 = u_1/u_2$
  - FC sets flow sum, $v_2 = u_1 + u_2$
- Exact inverse («static calculation block»):
  - $u_1 = v_1 v_2 / (1+ v_1)$
  - $u_2 = v_2 / (1 + v_1)$

Mechanical inverse:



1. EXACT INVERSE



Pairings:

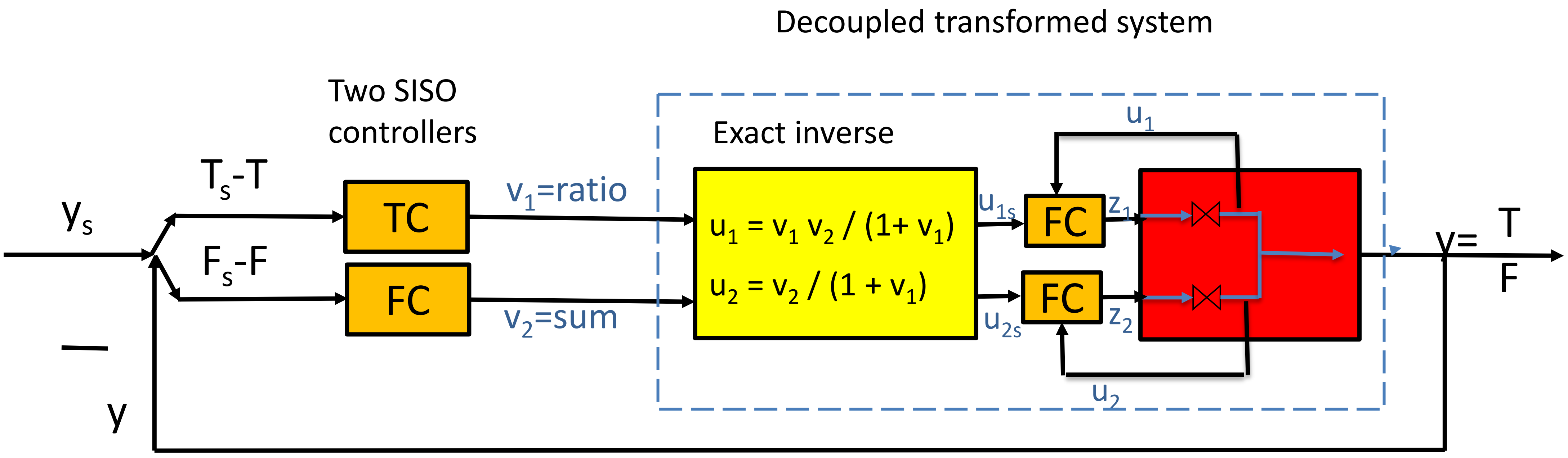
- $T - v_1$
- $F - v_2$

No interactions for setpoint change

Note:

- In practice u=valve position (z)
- So must add two flow controllers
  - These generate inverse by feedback

1. EXACT INVERSE actually requires flow controllers



Pairings:

- $T - v_1$
- $F - v_2$

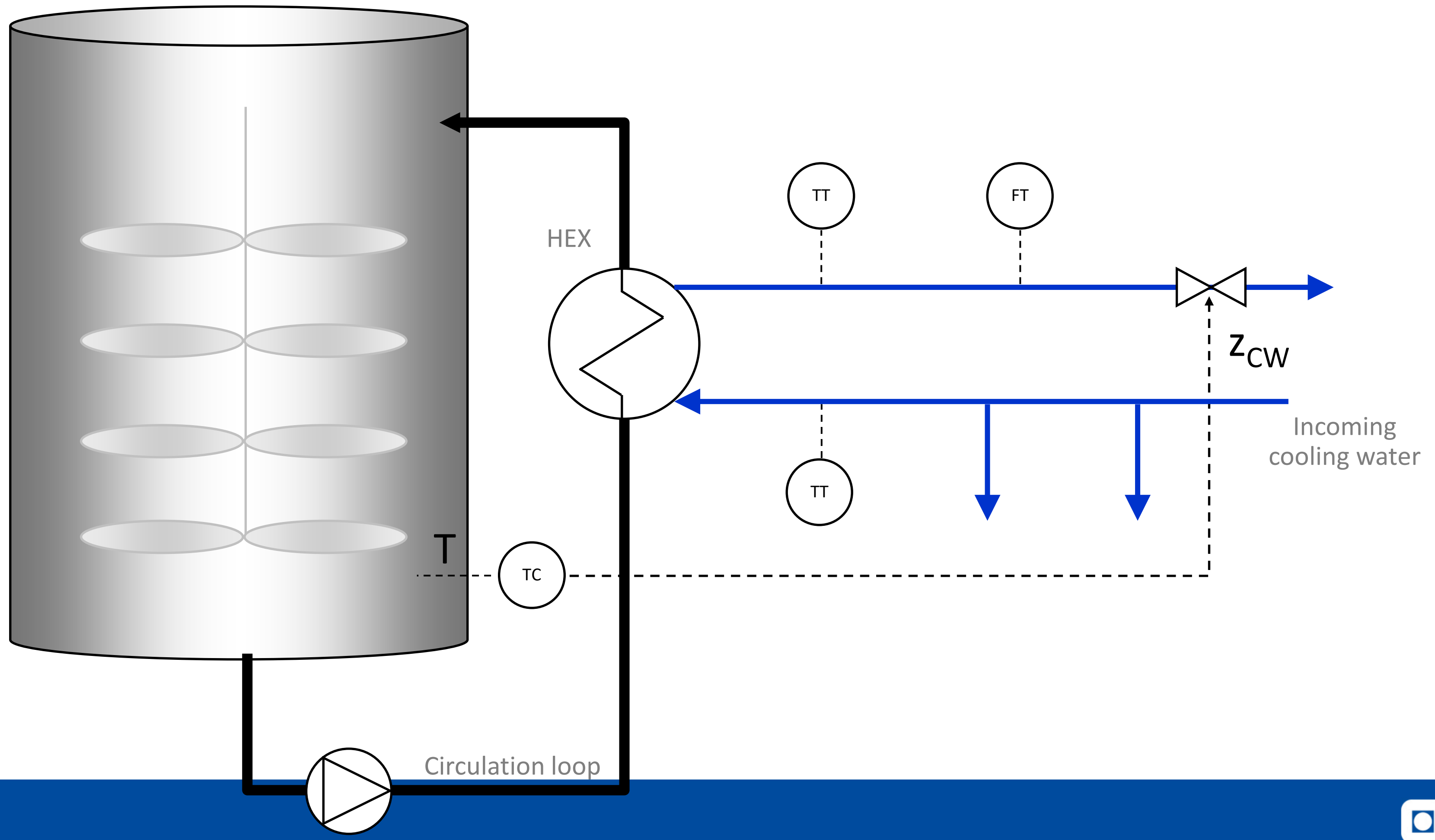
No interactions for setpoint change

Note:

- In practice u=valve position (z)
- So must add two flow controllers
  - These generate inverse by feedback

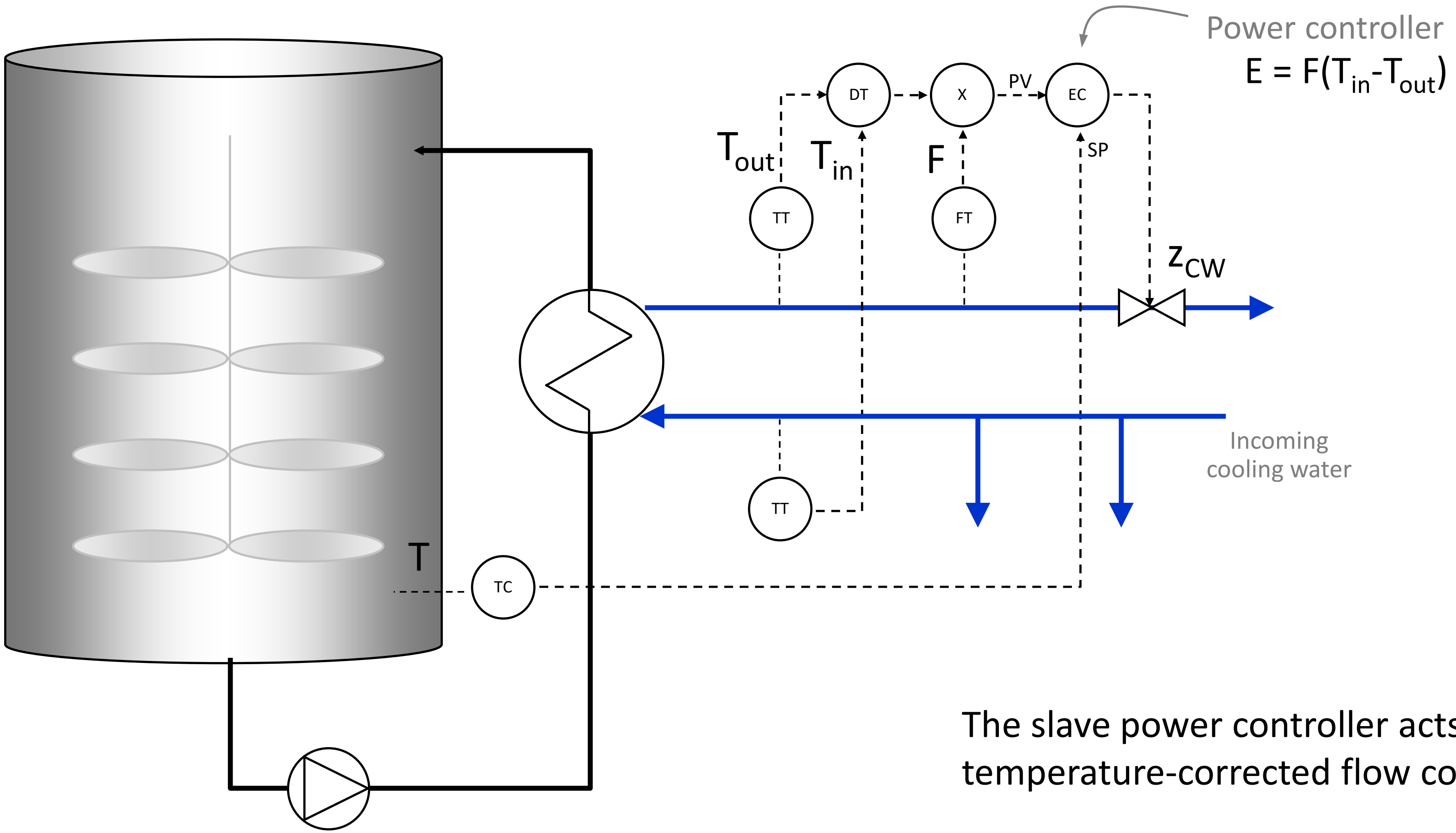
Example 2: Reactor temperature control

The reactor solution is circulated through a heat exchanger (cooler).
The reaction is very exothermic: it is important to control the temperature.
Typical variations/disturbances: Cooling water header pressure, CW temperature



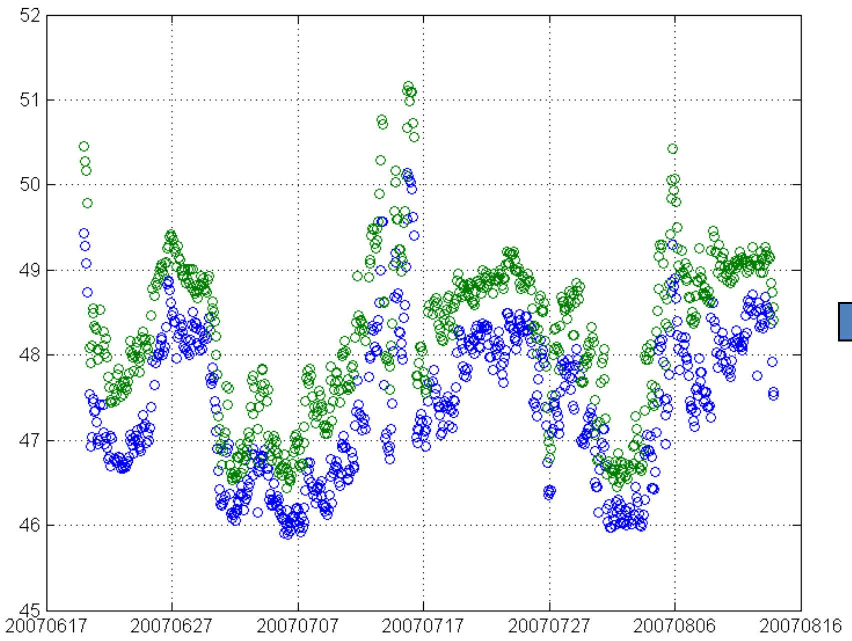
$y = T$
$u = z_{CW}$

New control structure: Power (E) control

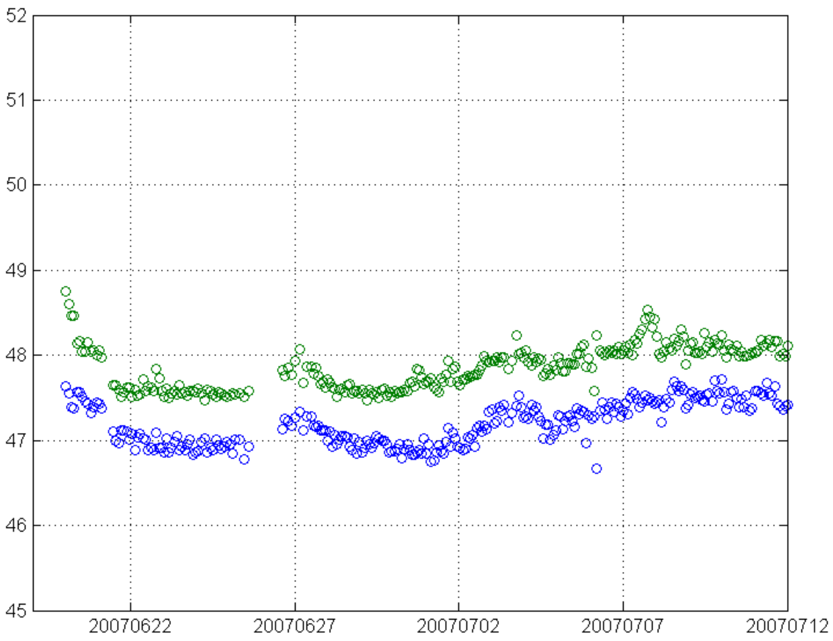


The slave power controller acts as a temperature-corrected flow controller

HEX power control reduces variations between batches



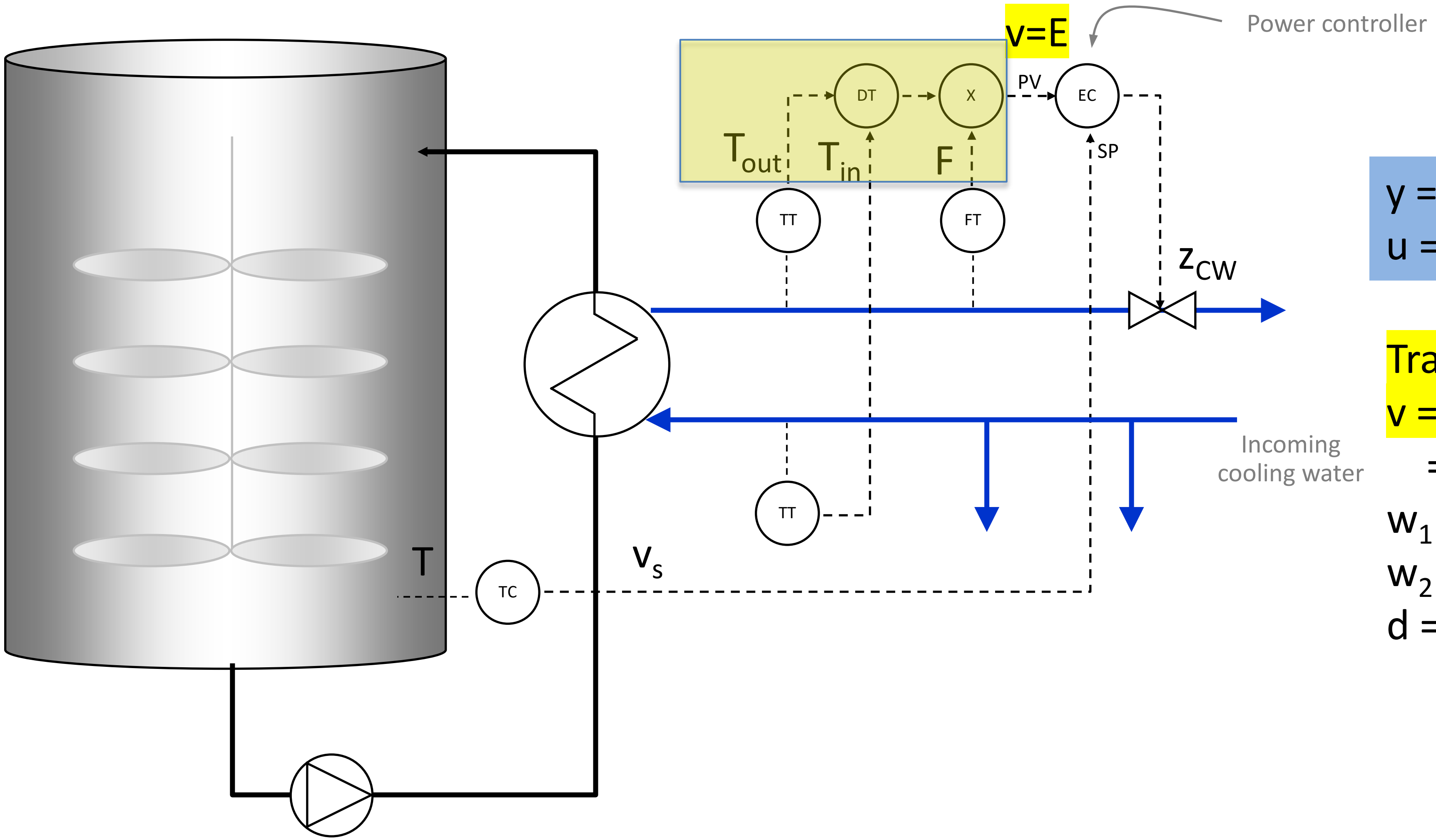
Before



With slave power controller

Thanks to: Krister Forsman

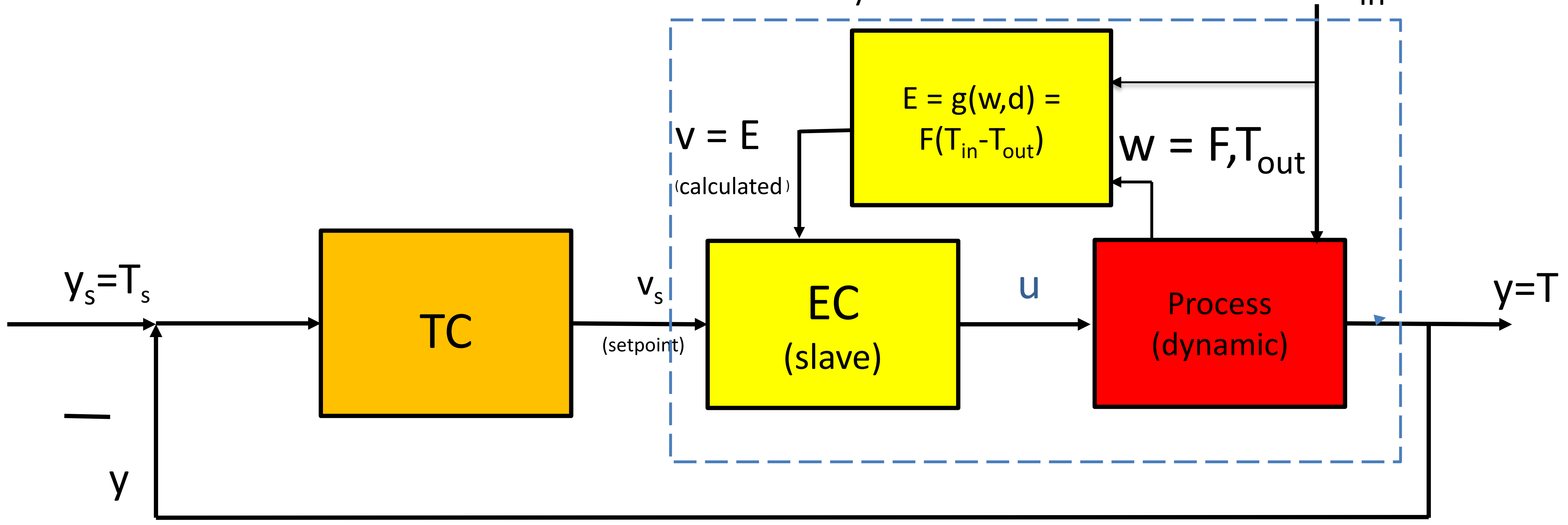
New control structure: Power (E) control



$y = T$
$u = z_{CW}$

Transformed input
$v = g(w,d) = E$
$= F(T_{in} - T_{out})$
$w_1 = F$
$w_2 = T_{out}$
$d = T_{in}$

2. INVERSE BY FEEDBACK



Must generate inverse by feedback (slave v-controller EC) since v=E does not depend explicitly on u=z_{CW}

Looks good..... Works in practice...But is there any theory?

- Not too much
- Question 1: How to derive input transformations in a systematic matter?
- Question 2: Properties of transformed systems. Stability?
  - Potential internal instability if transformed variable v depends on outputs (w)

Q1. Systematic derivation of input transformations

- From static model
- From dynamic model

Input transformation from static model

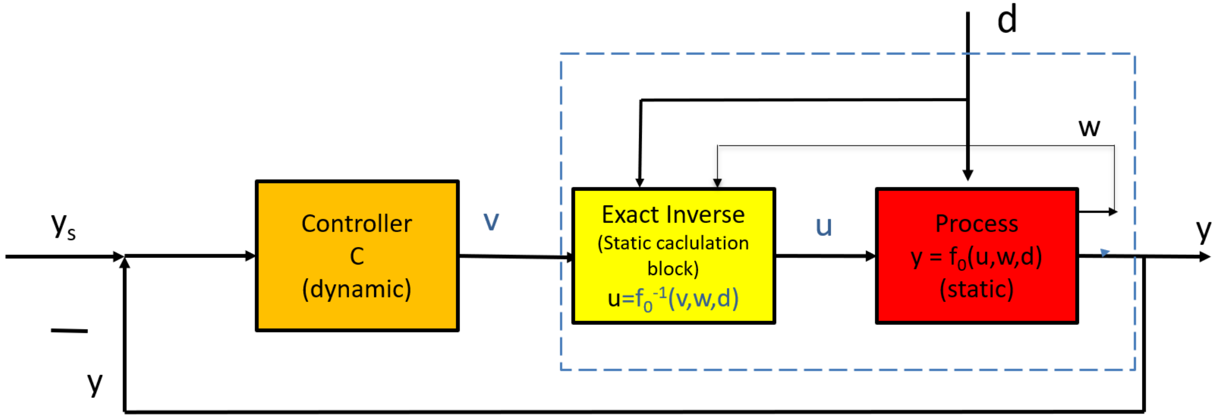
- Write nonlinear process model on form

  $y = f_0(u, w, d)$
- Introduce transformed inputs as RHS

  $v = f_0(u, w, d)$ **(*)**
- Exact inverse: u is solution to (*) for given v:

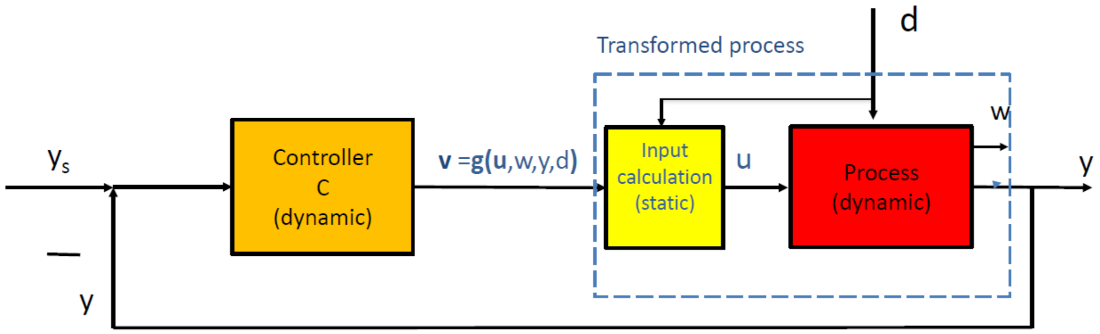
  $u = f_0^{-1}(v, w, d)$
- Resulting transformed system

  $y = v$
- Decoupled, linear and independent of disturbances



- **Assumptions**
  - Know model and measure all disturbances (d)
  - The solution to the static inverse problem exists and satisfies certain properties.
- **Note: If f_0 (and v) does not depend explicitly on u: Use feedback to generate approximate inverse**

Input transformation from dynamic model



- Write nonlinear dynamic process model on form

$$\frac{dy}{dt} = f(u, w, y, d)$$

- Introduce transformed inputs from RHS

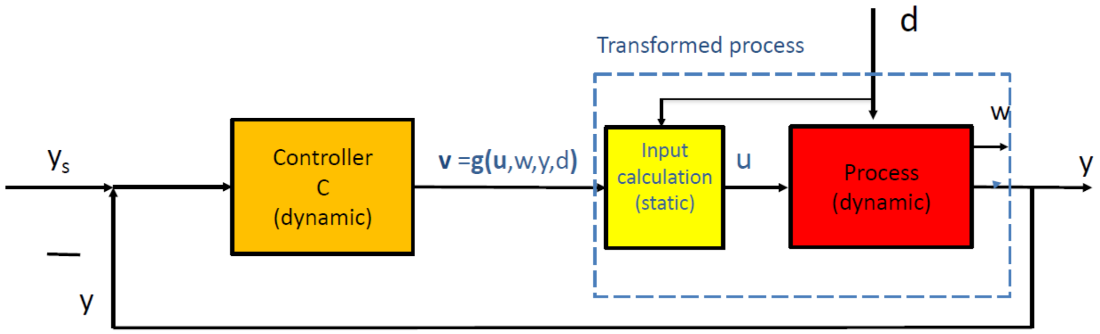
$$v = B^{-1}[f(u, w, y, d) - Ay] \quad (*)$$

- Tuning parameters (usually diagonal matrices): A and B.
- Exact inverse: u is solution to (*) for given v
- Resulting transformed dynamic system

$$\frac{dy}{dt} = Ay + Bv$$

  - linear, decoupled (with A and B diagonal) and independent of disturbances!
- **Assumptions**
  - Know model and measure all disturbances (d)
  - The solution to the static inverse problem exists and satisfies certain properties
- **Note: If f (and v) does not depend explicitly on u: Use feedback to generate approximate inverse**

Input transformation from dynamic model



- Write nonlinear dynamic process model on form

$$\frac{dy}{dt} = f(u, w, y, d)$$

- Introduce transformed inputs

$$v = B^{-1}[f(u, w, y, d) - Ay\,] \quad (*)$$

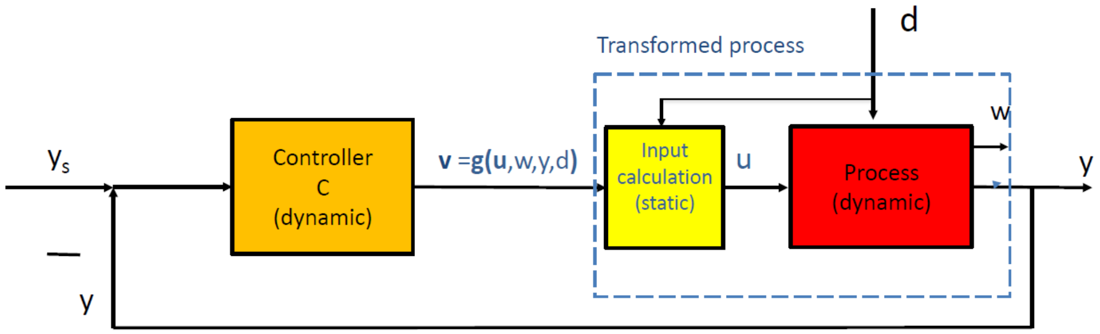
- Tuning parameters (usually diagonal matrices): A and B.
- Exact inverse: u is solution to (*) for given v
- Resulting transformed dynamic system

$$\frac{dy}{dt} = Ay + Bv$$

- Choices for B

1. B=I, → $\frac{dy}{dt} = Ay + v$
2. B=−A, → $\frac{dy}{dt} = A(y - v)$ (y=v at steady-state)

Feedback linearization for system of relative order = 1 (Isidori)



- Nonlinear dynamic system (process)

$$\frac{dy}{dt} = f(u, y, d) = f_1(y, d) + f_2(y, d)u$$

- Introduce transformed inputs

$$v = f(u, y, d)$$

- New transformed system is linear, integrating, decoupled and independent of disturbances:

$$\frac{dy}{dt} = v$$

- Corresponds to B=I and **A=0**

Why is A=0 a poor choice?

- Feedback linearization: Transformed linear system is integrating:

$$\frac{dy}{dt} = v$$

- A=0 (feedback linearization): Transform stable process into integrator (positive feedback from y)
  - Transformed system cannot be operated alone
  - Unknown disturbances will integrate.
  - Industrial experience: Bad!
- Imagine that we want fast control of a process which is already fast.
  - First make slow (integrating) by using A=0 (positive feedback)
  - Then make fast again using controller C (negative feedback)
    - Does not make much sense!
    - Also: Integrating systems are not easy to control using C
- Fortunately, it is not necessary to make choice A=0 in feedback linearization
    - Theory still holds
    - A=0 was chosen as an example for simplicity (Isidori)
- Feedback linearization theory applies to input transformations

Nonlinear Decoupling via Feedback: A Differential Geometric Approach

ALBERTO ISIDORI, MEMBER, IEEE, ARTHUR J. KRENER, MEMBER, IEEE, CLAUDIO GORI-GIORGI, AND SALVATORE MONACO

Abstract—The paper deals with the nonlinear decoupling and noninteracting control problems. A complete solution to those problems is made possible via a suitable nonlinear generalization of several powerful geometric concepts already introduced in studying linear multivariable control systems. The paper also includes algorithms concerned with the actual construction of the appropriate control laws.

Manuscript received June 25, 1979; revised October 19, 1979 and August 25, 1980. Paper recommended by M. Vidyasagar, Past Chairman of the Stability, Nonlinear, and Distributed Systems Committee. This work was supported by the University of Rome under Grant 7.8*P/1978 and by the National Research Council of Italy.

A. Isidori, C. Gori-Giorgi, and S. Monaco are with the Istituto di Automatica, University of Rome, Rome, Italy.
A. J. Krener is with the Department of Mathematics, University of California, Davis, CA 95616.

I. Introduction

CONSIDER a nonlinear system of the form

$$\dot{x} = f(x) + g(x)u \quad (1.1a)$$

$$x(0) = x^0 \quad (1.1b)$$

$$y = h(x) \quad (1.1c)$$

where the input u, the output y, and the state x are l, m, and n dimensional, f and h are vector-valued differentiable functions, and g is a matrix-valued differentiable function, all of the appropriate dimensions. We shall be more precise later on. The input–output behavior of such a system can

0018-9286/81/0400-0331$00.75

From: Alberto Isidori <albisidori@diag.uniroma1.it>
Sent: Sunday, October 4, 2020 5:24 PM
To: Sigurd Skogestad <sigurd.skogestad@ntnu.no>
Subject: Re: Feedback linearization generalization

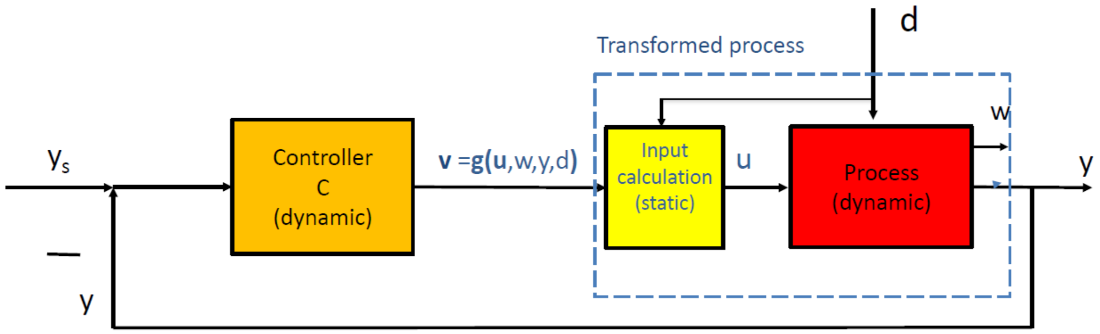
Dear Sigurd
It is nice to hear from you....
Let's move to your questions. I believe that an answer could be as follows.
In feedback linearization, one picks A=0 just as an example. .... The equation
f(y,u) = Ay - v
must be solvable for u. This entails, in the higher-dimensional case, a definition of "relative degree" ....

Best regards
Alberto

Choice of Tuning parameter A

- One idea: Select $A = \frac{df}{dy}$ at nominal operating point
  - Then: No feedback from y into transformation (nominally)
  - Transformed system has the same dynamics as the original system (nominally)
- To get decoupling may choose: $A = diag(\frac{df}{dy})$
  - Will get some feedback from y also nominally.
- May want to «speed up» the response of the transformed systems by selecting a larger A.
  - This involves negative feedback from y, and may as usual give robustness problems if time delay for y
  - «Slowing down» the response (positive feedback from y) does not have robustness problems

B = -A gives steady-state gain I*



- $\frac{dy}{dt} = f(u, w, y, d)$
- Define $T_A = -A^{-1}$
- Select $v = TA\, f(u, w, y, d) + y$

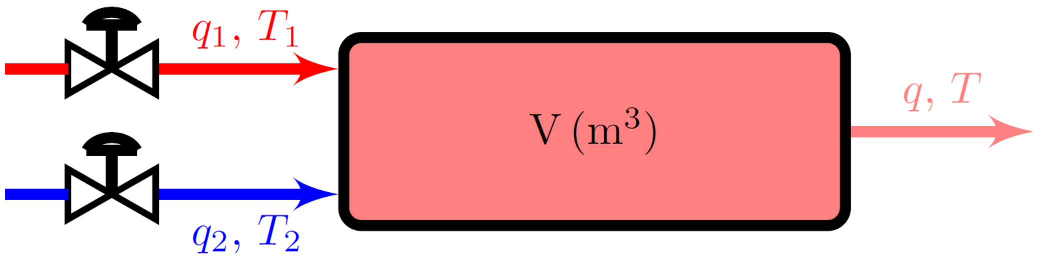
- Get transformed system*: $T_A \frac{dy}{dt} = -y + v$
- Transformed system has linear «setpoint» response (from v to y) with
  - time constant T_A
  - steady-state gain I
- May in theory avoid the outer feedback controller C
  - But note that the transformation works by feedforward action
- **The outer controller C is needed to correct for model errors and unknown disturbances**

*Zotica, Alsop and Skogestad. 2020 IFAC World Congress

Examples («magic»)

- Example 1 (revisit): Mixing with one static and one dynamic equation
- Example 2 (revisit): Reactor temperature control (dynamic)
- Example 3: Heat exchanger (static applied to dynamic system)
- Example 4: CSTR (with exact inverse using w)

Example 1. Mix hot (1) and cold (2) water (shower), y=[q T]



Mass balance: $q = q_1 + q_2$ (static equation for y_1=q)

$$v_0 = q_1 + q_2$$

Energy balance: $\frac{dT}{dt} = \frac{q_1}{V}(T_1 - T) + \frac{q_2}{V}(T_2 - T)$ (dynamic equation for y_2=T)

$$v_A = \frac{q_1}{V}(T_1 - T) + \frac{q_2}{V}(T_2 - T) - AT$$

New transformed inputs: v_0 and v_A

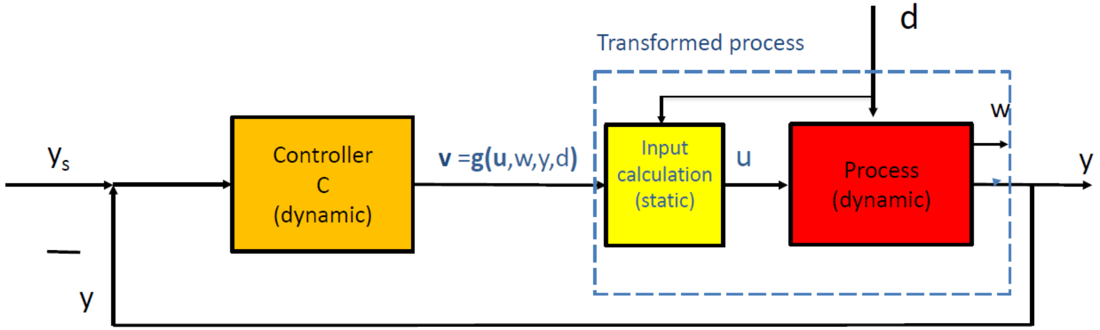
- v_0 = sum of flows
- v_A: not ratio (but would be similar to ratio if we used static energy balance)

Exact inverse transformation (with u_1=q_1 and u_2=q_2):

$$q_1 = \frac{V(v_A + AT) + v_0(T - T_2))}{T_1 - T_2}$$

$$q_2 = v_0 - q_1$$

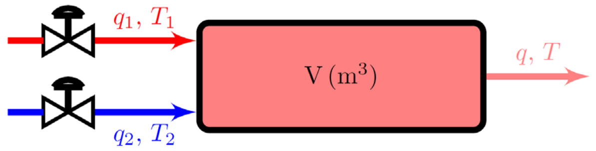
Tuning parameter, A = -(q/V)* (nominal)

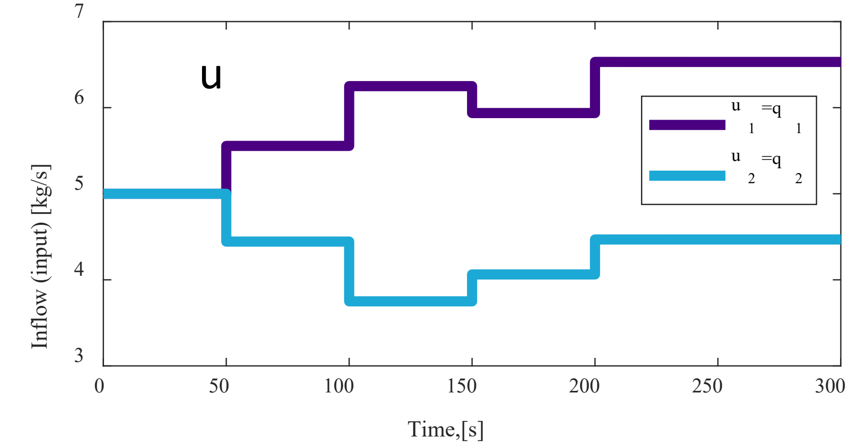


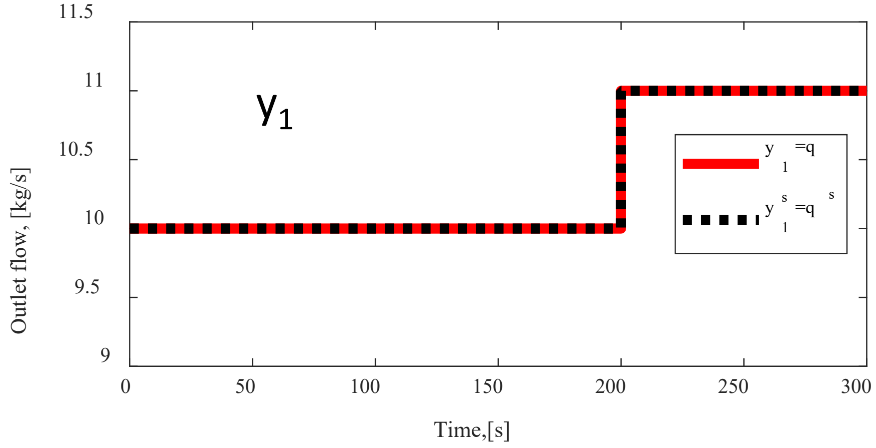
NTNU

Example 1. Simulation responses with transformation only.

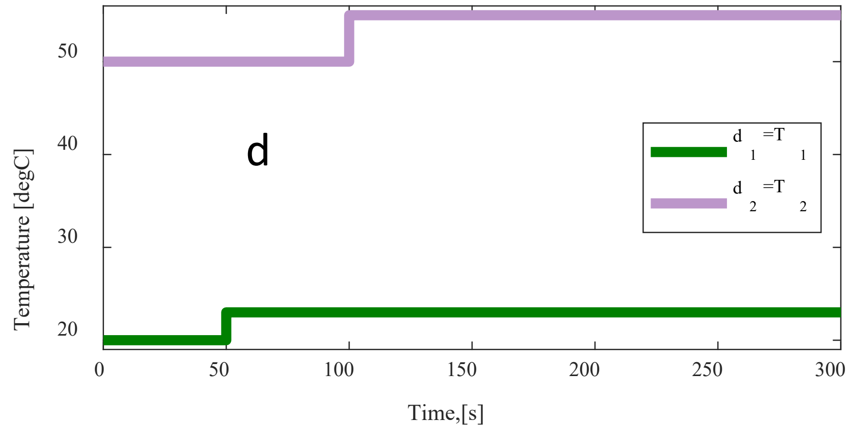
-> Perfect disturbance rejection and decoupling

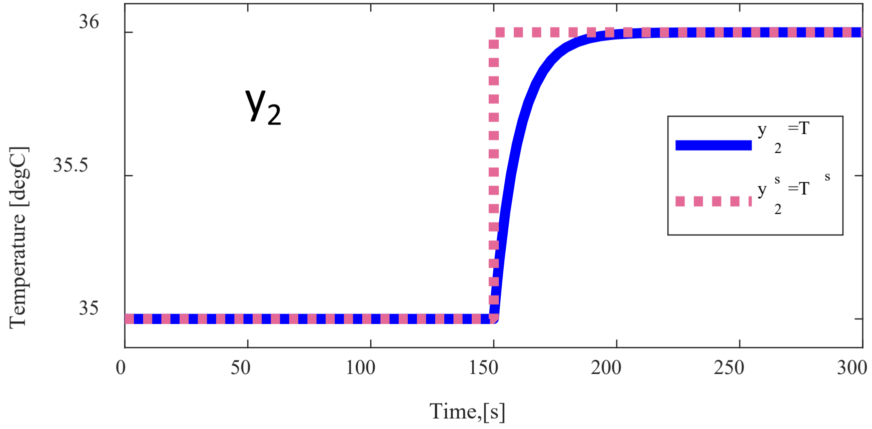






1. $d_1=T_1$: 20 ->22 °C at t = 50 s
2. $d_2=T_2$: 50 ->55 °C at t = 100 s
3. $y_{2s}=T_s$: 35 ->36 °C at t = 150 s
4. $y_{1s}=q_s$: 10 ->11 kg/s at t = 200 s





Example 2. Control of reactor temperature

Energy balance tank:

$$m_1 c_{p1} dT/dt = Q + Q_{rx}$$

Static energy balance for cold side:

$$Q = F c_p (T_{out} - T_{in}) = E c_p$$

Neglecting heat of reaction Q_{rx}, we get

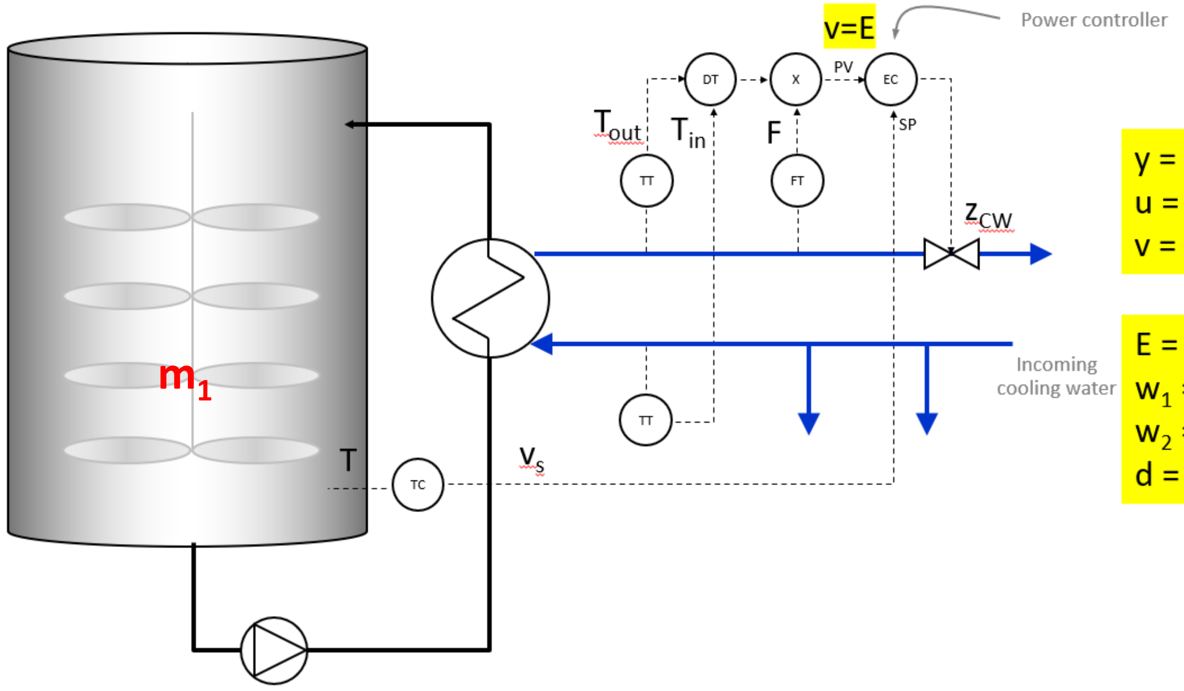
$$dT/dt = k E, \quad k = c_p/(c_{p1} m_1)$$

Transformed input with systematic approach

$$v = kE - AT$$

Note: choice v=E corresponds to A=0

- Some self-regulation removed by EC
- Maybe not so bad for this process which anyway needs to be stabilized (because of Q_{rx})



y = T
u = z_{CW}
v = E

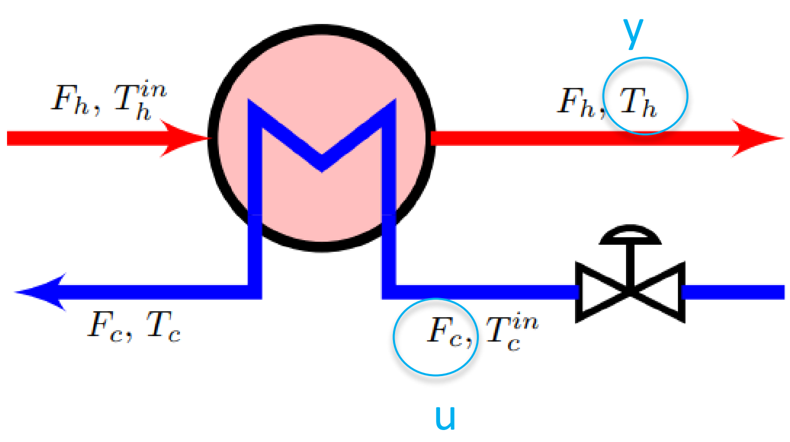
E = g(w,d)=F(T_{in}-T_{out})
w_1 = F
w_2 = T_{out}
d = T_{in}

Q_{rx}(T) is mainly a function of T. Can be handled by design of TC

In practice

- May not measure all disturbances
- Transformation will not longer be «perfect» but still useful

Example 3. Heat exchanger (static)



MVs (original inputs):

$$u = F_c \text{ [kg/s]}$$

CVs (outputs):

$$y = T_h \text{ [°C]}$$

DVs (disturbances):

$$d_1 = T_c^{in} \text{ [°C]}$$
$$d_2 = T_h^{in} \text{ [°C]}$$
$$d_3 = F_h \text{ [kg/s]}$$

Energy balance, countercurrent flow, $Q = UA\Delta T_{LM}$

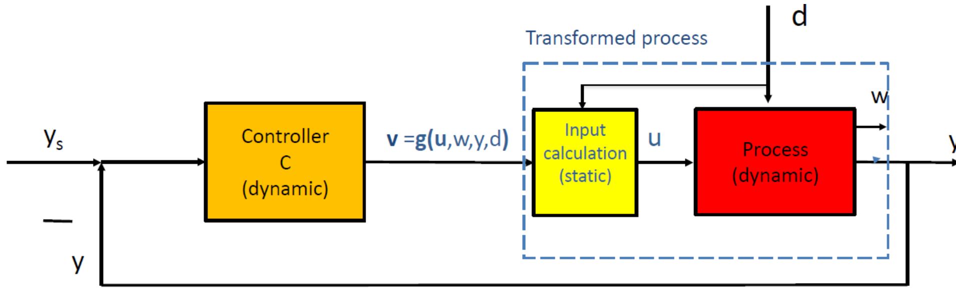
$$T_h = \underbrace{(1-\epsilon_h)T_h^{in} + \epsilon_h T_c^{in}}_{v_0}$$

$$N_{tu} = \frac{UA}{F_c c_{p,c}}$$

$$C = \frac{F_c c_{p,c}}{F_h c_{p,h}}$$

$$\epsilon_c = 1 - \frac{\exp(-N_{tu}(C-1))}{C - \exp(-N_{tu}(C-1))}$$

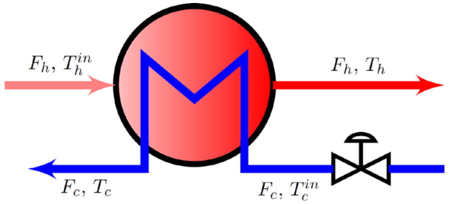
$$\epsilon_h = \epsilon_c C$$



Input calculation:
Use numerical inverse (to find u for given $T_h=v_0$)

NTNU

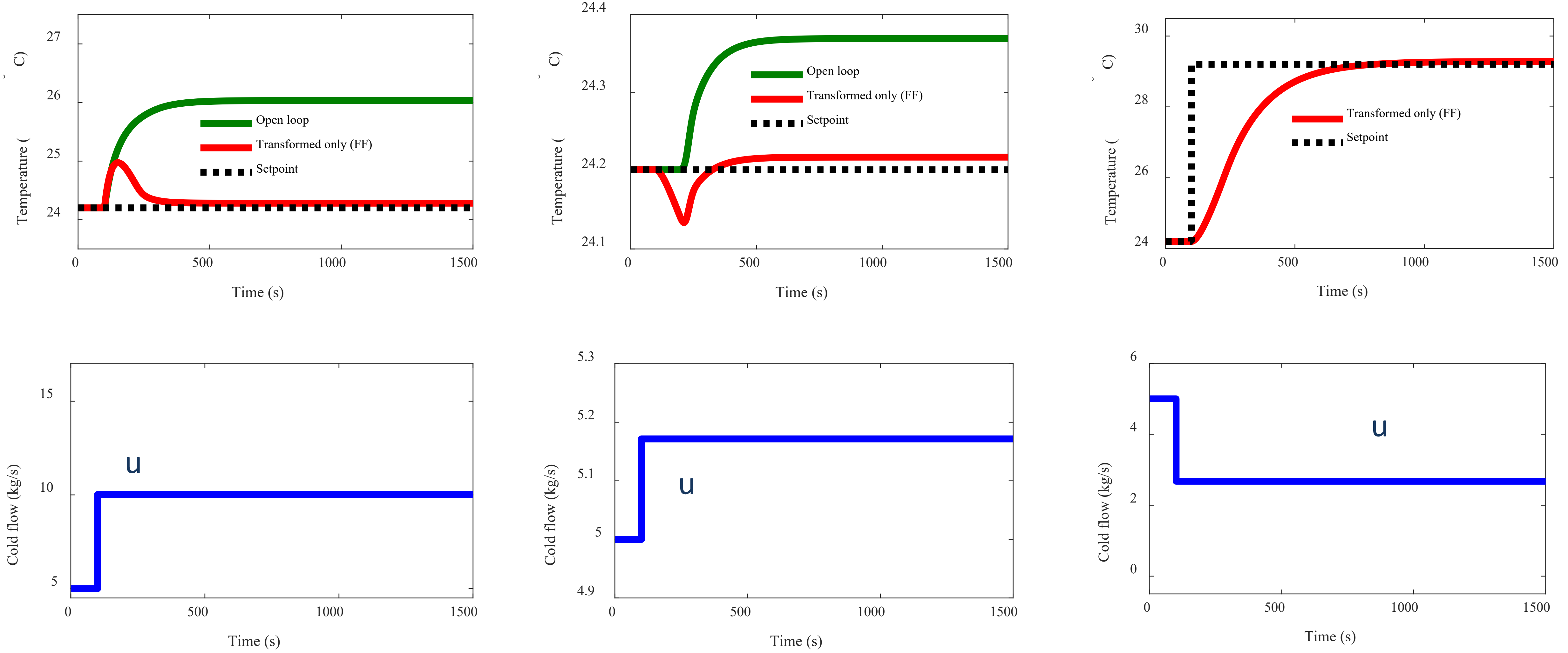
Simulation: Static v_0 with cell dynamic model



$d_1 = \Delta T_c^{in} = +2$ °C

$d_2 = \Delta T_h^{in} = +2$ °C

$v_0 = \Delta T_h^s = +5$°C



Extension: Chain of transformations

- Idea: **Extend exact inverse to systems of higher relative order (when v does not depend explicitly on u)**
- Model for y (static case) or dy/dt (dynamic case)

$$y = f_0(w, d) \quad or \quad \frac{dy}{dt} = f(w, y, d)$$

  – Until now: Cannot use exact inverse

- Alternative 1 (until now): Use feedback control to generate approximate inverse
- Alternative 2 (chain of transformations): Make use of known model for w

$$\frac{dw}{dt} = f_2(u, w, y, d_2)$$

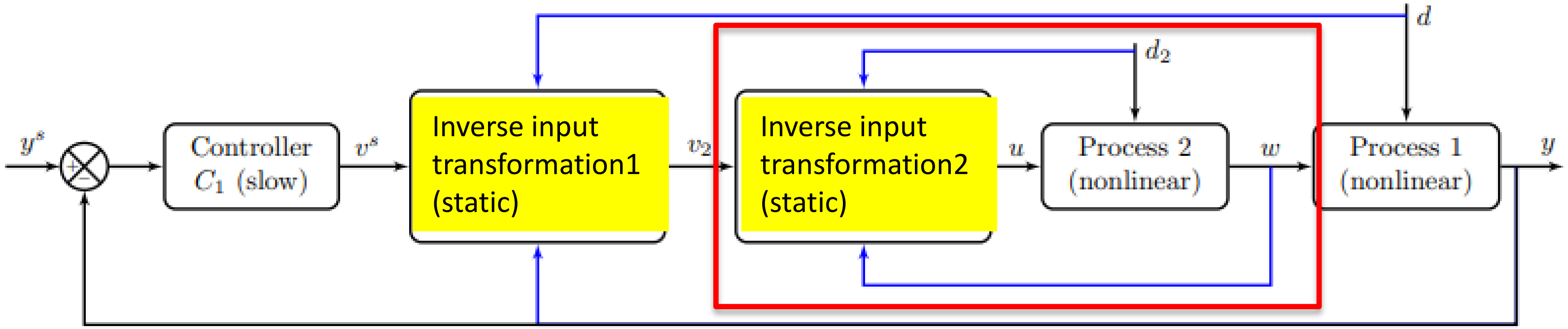
  – Use two exact inverses; find w from f, find u from f_2.
  – May be viewed as alternative to «feedback linearization» for systems with high relative order

NTNU

Chain of transformations

$$dy/dt = f(w, y, d) \quad = Ay + v$$ (Inverse 1: Solve for given v to find w^s)

$$\frac{dw}{dt} = f_2(u, w, y, d_2) \quad = A_2(w - v_2)$$ (Inverse 2: Solve for given given $w^s = v_2$ to find u)



- Input u has relative order 2 (from u to y)
- Get perfect disturbance rejection for d_2 (enters same place as u)
- But not for d since it must go through subsystem 2

$$\tau_2 \frac{dw}{dt} = v_2 - w$$

$\tau_2 = -A_2^{-1}$

$v_2 = w^s$

- Note: Choose B=-A in inner transformations to get steady-state gain of I ($v_2 = w^s$)

NTNU

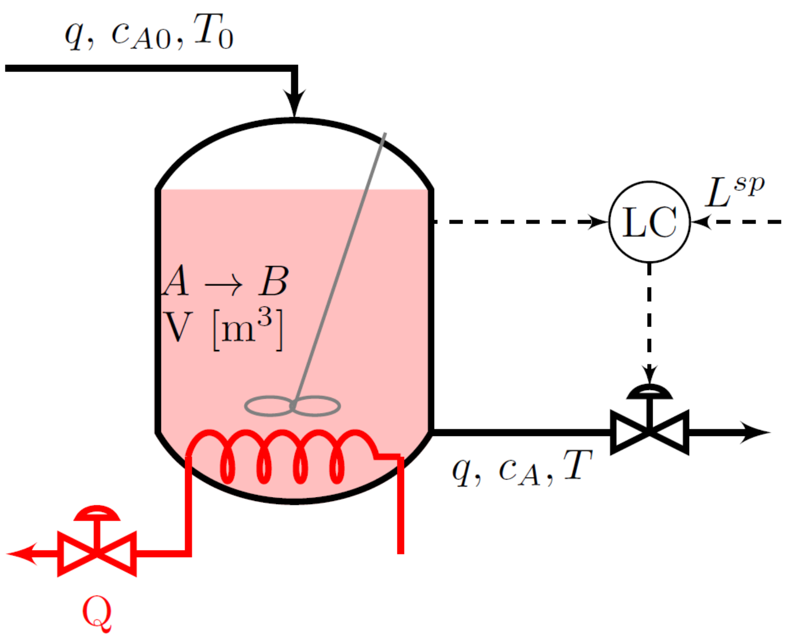
Example 4. CSTR. $y = c_A\,, u = Q, w = T$

Component balance: $\frac{dc_A}{dt} = \frac{q}{V}(c_{A0} - c_A) - k(T)c_A$

$$k = k_0 \exp\left(-\frac{E_A}{R}\left(\frac{1}{T} - \frac{1}{T_{ref}}\right)\right)$$

$$v_A = \frac{q}{V}(c_{A0} - c_A) - k(T)c_A - Ac_A$$

Energy balance: $\frac{dT}{dt} = \frac{q}{V}(T_0 - T) + \frac{Q}{V\rho c_p} - \frac{\Delta H_{rx} k(T) c_A}{\rho c_p}$



Alternative 1: Cascade control of v_A

Alternative 2: Use also model for w=T (energy balance)

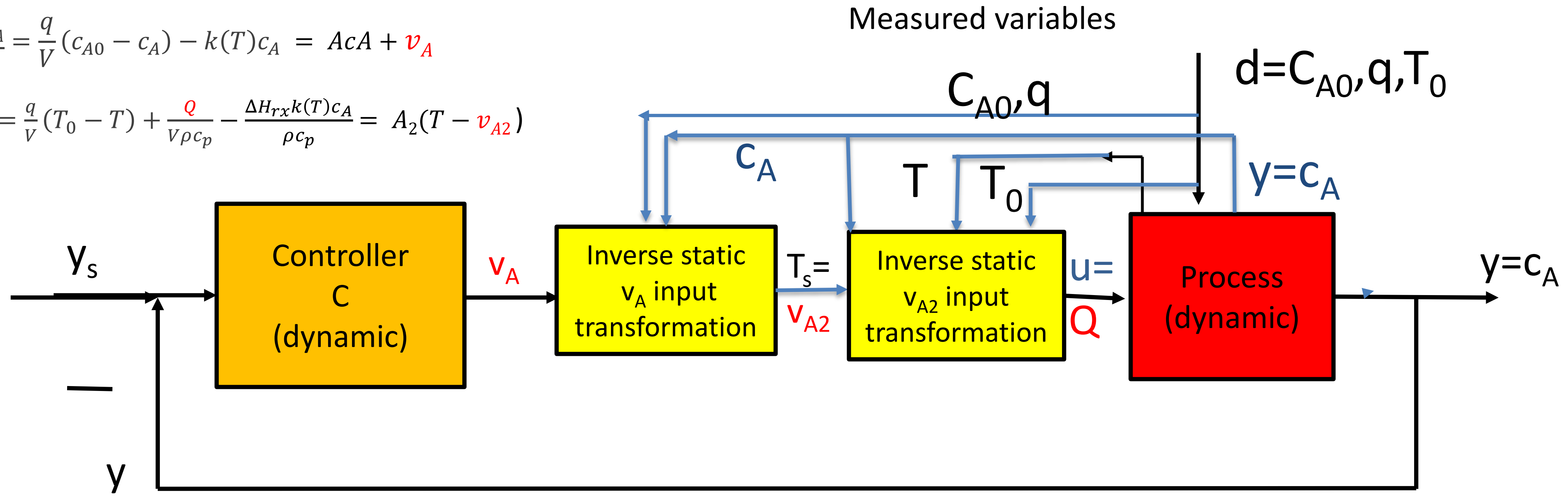
$$\frac{q}{V}(T_0 - T) + \frac{Q}{V\rho c_p} - \frac{\Delta H_{rx} k(T) c_A}{\rho c_p} = A_2\,(T - v_{A2})$$

and use chain of transformations.

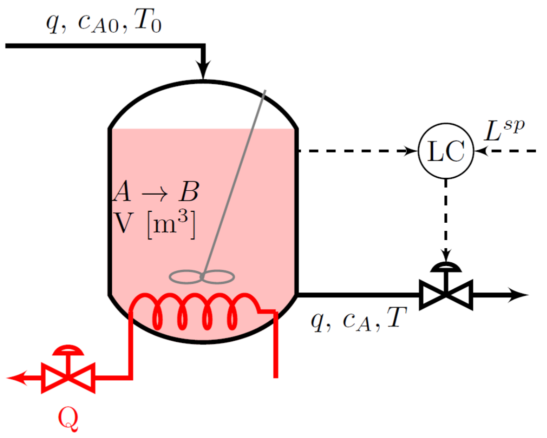
Alt.2 Chain of inverse transformations

$$\frac{dc_A}{dt} = \frac{q}{V}(c_{A0} - c_A) - k(T)c_A \ = \ AcA + v_A$$

$$\frac{dT}{dt} = \frac{q}{V}(T_0 - T) + \frac{Q}{V\rho c_p} - \frac{\Delta H_{rx} k(T) c_A}{\rho c_p} = \ A_2(T - v_{A2})$$

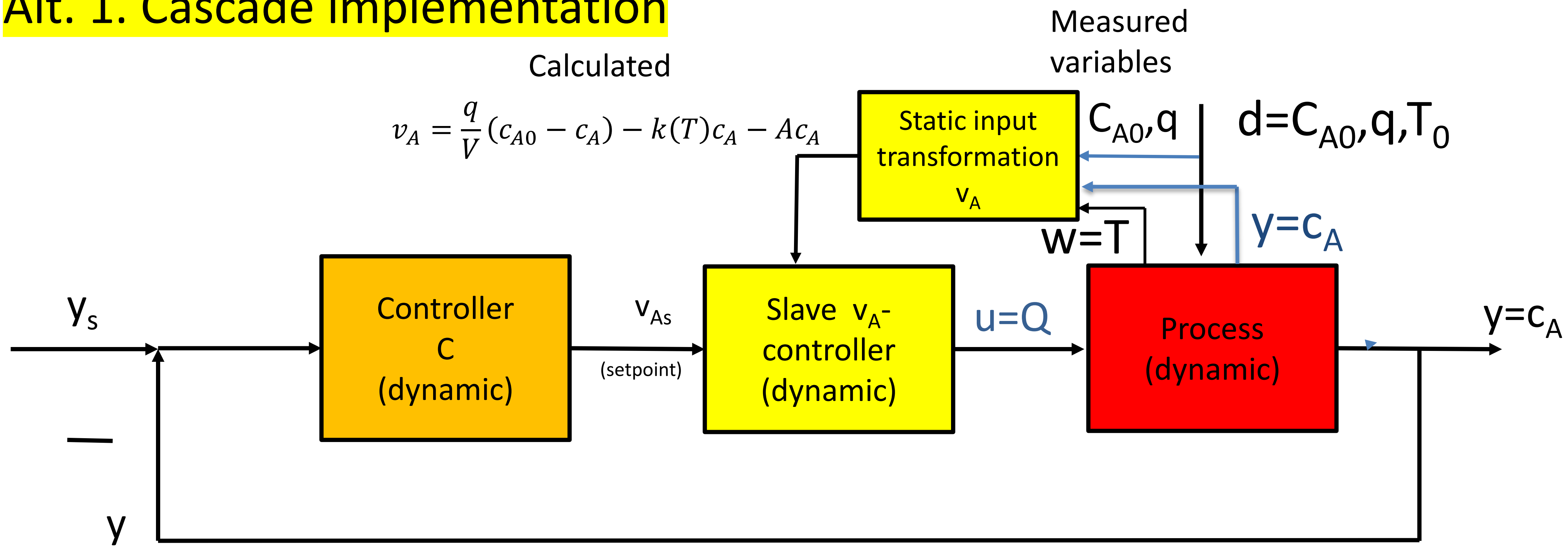


- Make use of model for w=T
- Feedforward control for T_0 will be perfect, but not for C_{A0} and q.
- Need to invert expression for v_A with respect to T -> invert k(T).



NTNU

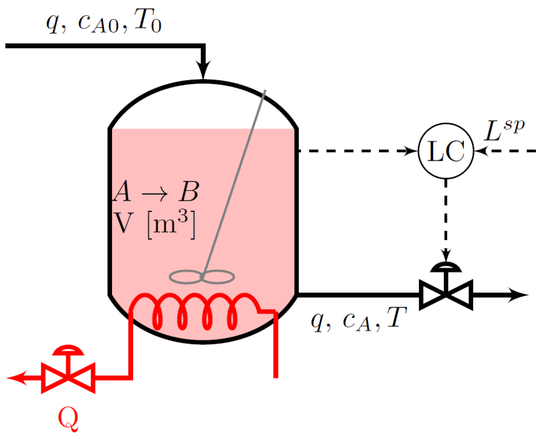
Alt. 1. Cascade implementation



Two reasons to use slave controller

1. u=Q does not appear in v_A
2. Avoid inverting expression for v_A with respect to T

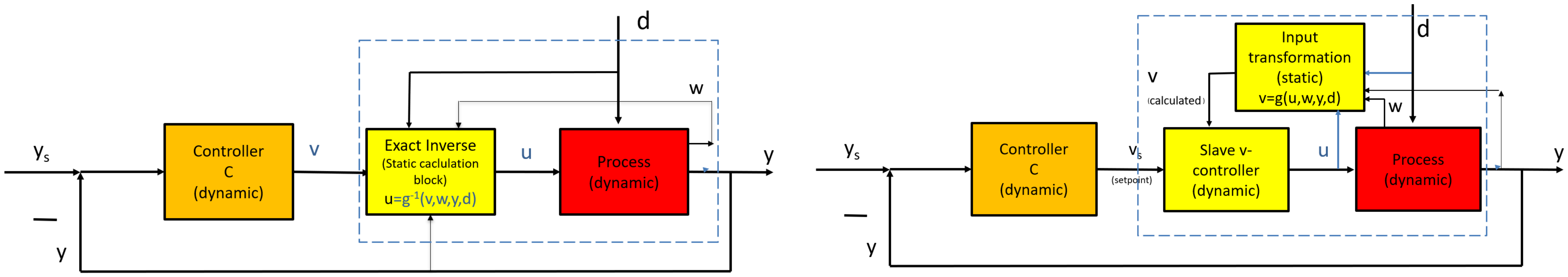
But do not get perfect feedforward control for T_0



NTNU

Q2. Stability problems?

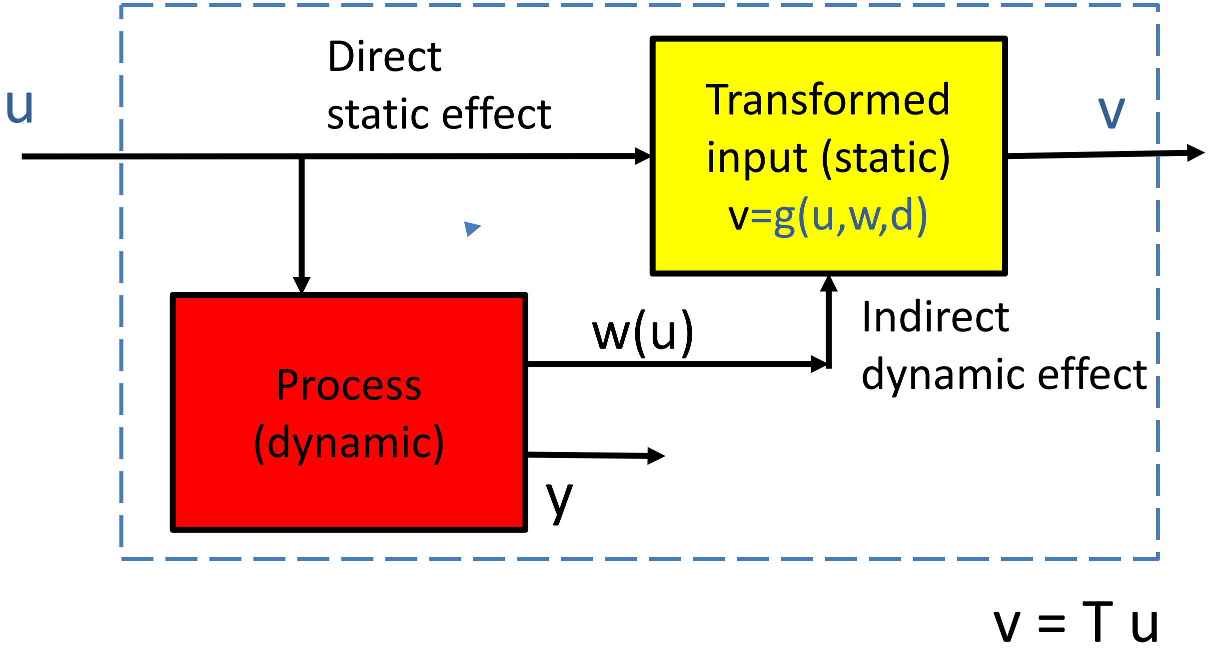
- Consider any transformed input, v = g(u,w,y,d)
- With exact inverse the **transformed system may be internally unstable** because we treat w as disturbance, but actually w depends on u
  - Happens when w causes **unstable zero dynamics** from u to v
- Stability problems can be avoided with feedback (cascade) implementation which gives approximate inverse



Unstable zero dynamics

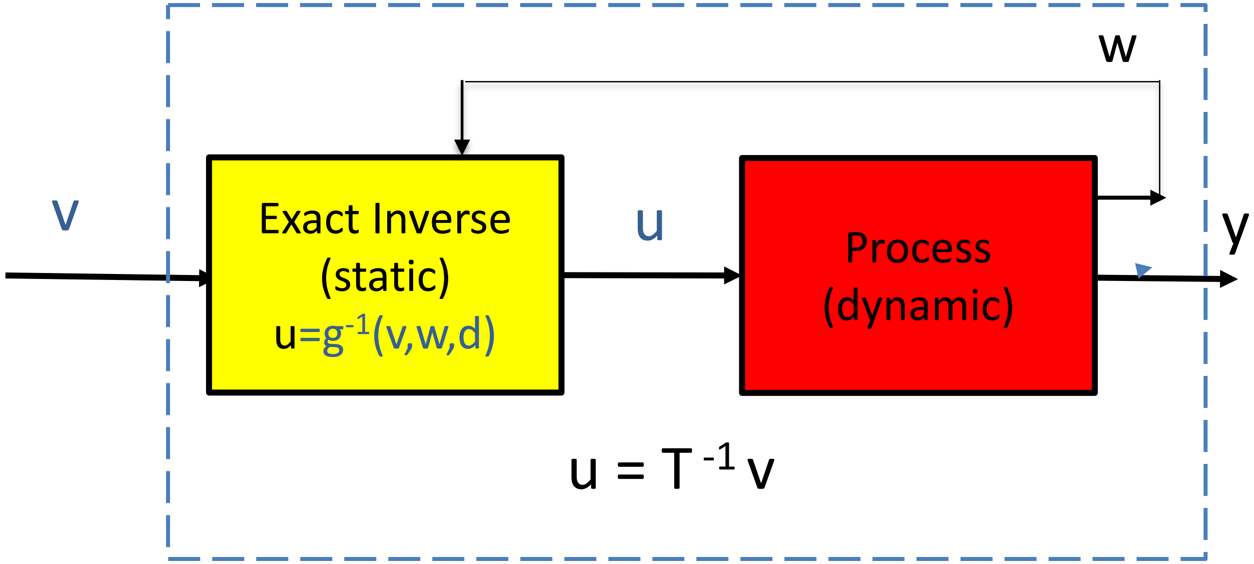
Indirect effect through w may cause

- unstable zero dynamics for T (from u to v)
- = inverse response from u to v (scalar)



Internal instability

Unstable zero dynamics for T give internal unstable tranformed system if we use exact inverse



Internally unstable:

Response from v to y is stable (apparently), but internal signals u and w are unstable

Example 5: Internal instability

Process

$$y = u + w + d$$

$$w = \frac{-2u}{4s+1}$$

Transformed input

$$v_0 = g(u, w, d) = u + w + d$$

$$v_0 = \frac{4s-1}{4s+1} u + d$$

Exact inverse

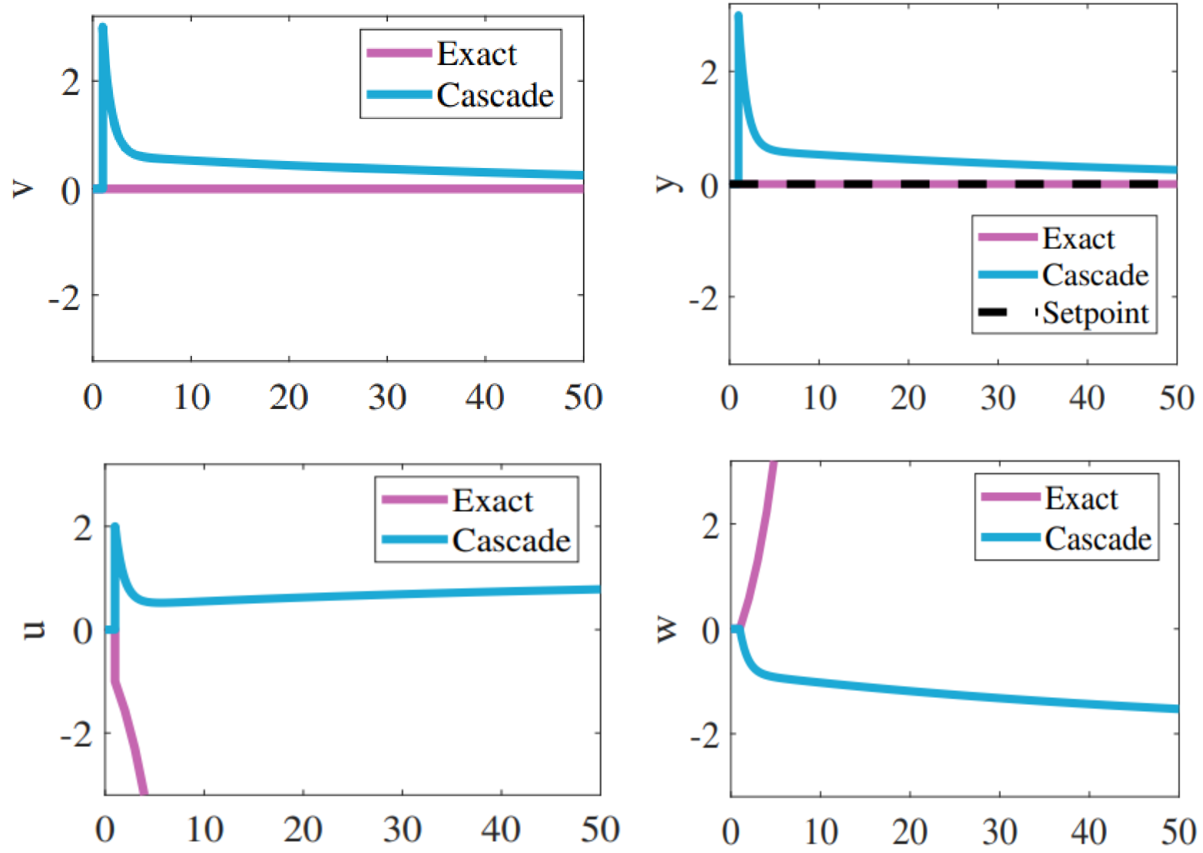
$$u = v_0 - w - d$$

$$u = \frac{4s+1}{4s-1}(v_0 - d)$$

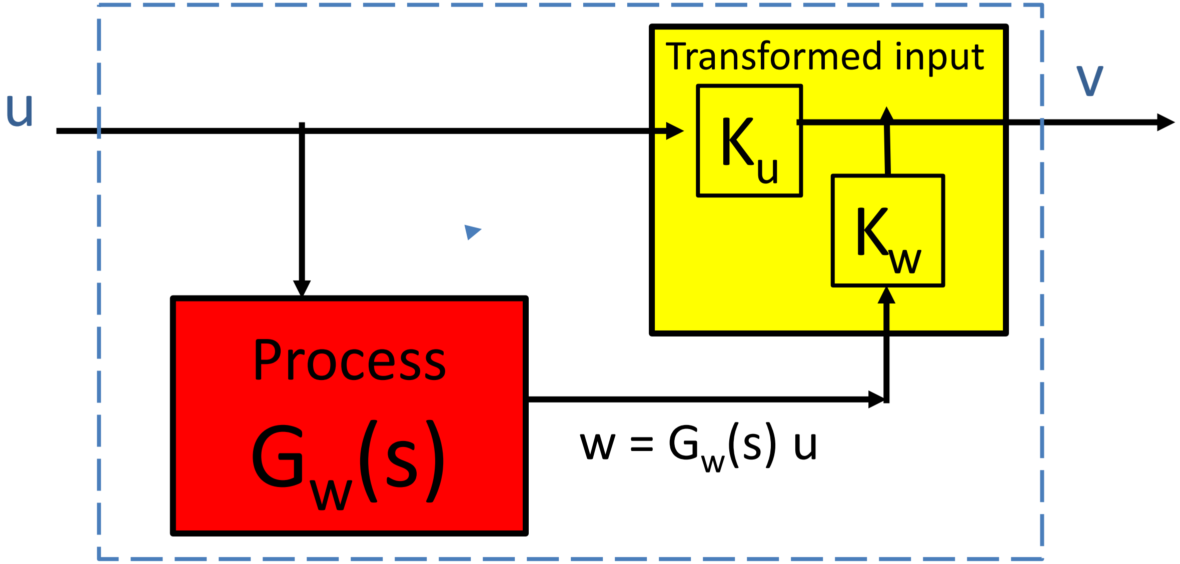
Response of transformed system

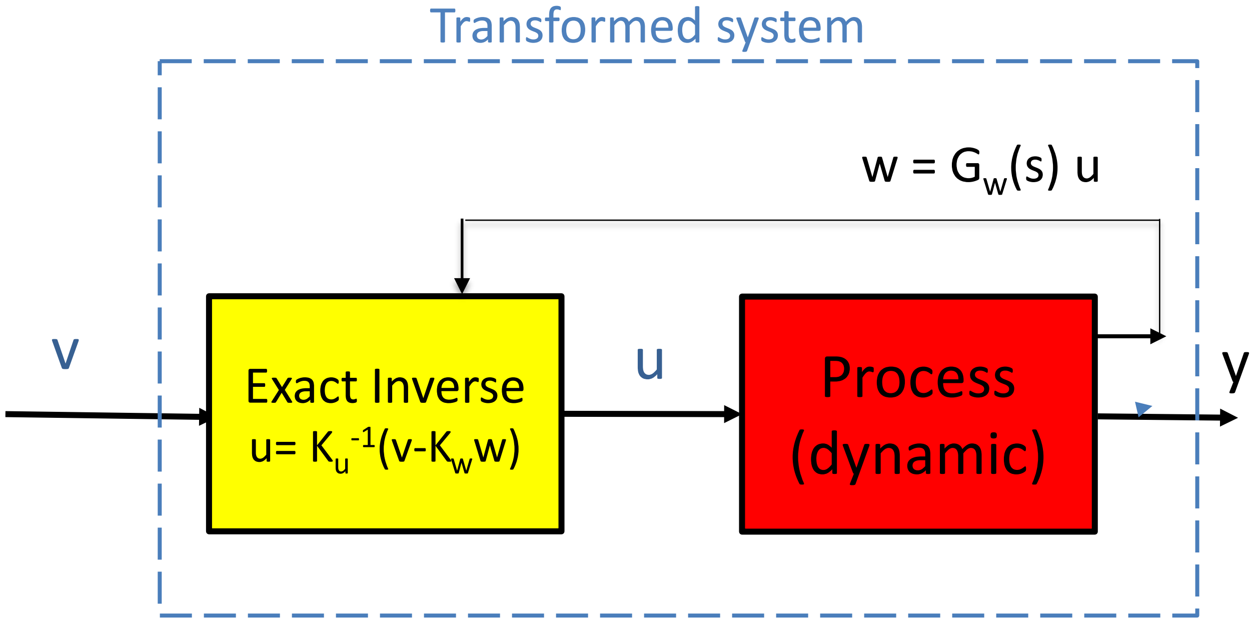
$$y = v_0$$

Response to step disturbance (d=1)



Linear analysis





Transformed input

$$v = K_u u + K_w w$$
$$= (K_u + K_w G_w)u = T u$$

where

$$T = (K_u + K_w G_w) = K_u(I+L_w)$$
$$L_w = K_u^{-1} K_w G_w$$

Transformed system with exact inverse

$$u = T^{-1} v = (I+L_w)^{-1} K_u^{-1} v$$

For internal stability of transformed system:

$$T^{-1} = (I+L_w)^{-1} K_u^{-1} \text{ must be stable}$$

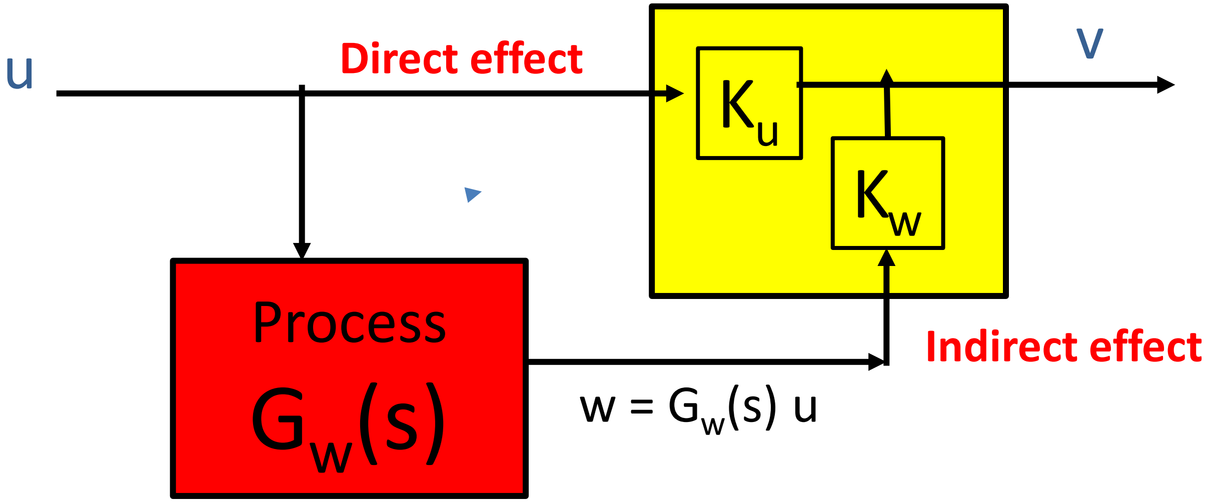
Equivalently: Transfer function

$$T= K_u(I+L_w)$$

from u to v must have stable zero dynamics

Trick: Can use Nyquist/Bode stability condition for L_w

Linear stability theorems



Transformed input

$v = K_u(I+L_w)\ u$

where

$L_w = K_u^{-1} K_w\ G_w$

Stability.
Transformed system is internally stable if and only if

$(I+L_w)^{-1}$ is stable

Bode stability condition: Internally stable

if and only if $|L_w(j\omega_{180})|<1$ (scalar)

Small gain theorem. Stable

if $|L_w(j\omega)|<1$ at all ω

In words: Stable if «indirect effect» $K_w G_w$ (through w)

is smaller than «direct effect» K_u (through u).

Bode stability condition

$L_w = K_u^{-1} K_w G_w$

Bode (scalar): Internally stable if and only if $|L_w(j\omega_{180})|<1$

Two cases

1. $L_w(0) < 0$: Direct and indirect effect are opposite at steady-state. $\omega_{180}=0$

   Get internal instability iff $|L_w(0)|>1$

   - When Indirect effect is larger and opposite at steady state

   Example 5: $K_u=K_w=1$ and $G_w=-2/(4s+1)$ so $L_w(0)=-2$ <-> internally unstable

   Note: Transfer function from u to v is $T = K_u(I+L_w) = (4s-1)/(4s+1)$.

2. $L_w(0) > 0$: Direct and indirect effect in same direction at steady state.

   - Internal instability is less likely.
   - Requires that indirect effect is large and that G_w has unstable zeros (inverse response) or delay

   Example 6: $K_u=1$, $G_w = (-s+1)/(s+1)$. $\omega_{180}=\infty$. $|L_w|=K_w$ at all ω. Get internal instability iff $K_w>1$.

   Note: With $K_w=2$, transfer function from u to v is $T= K_u(I+L_w) = (3-s)/((s+1)$.

What if uncertain about internal instability?

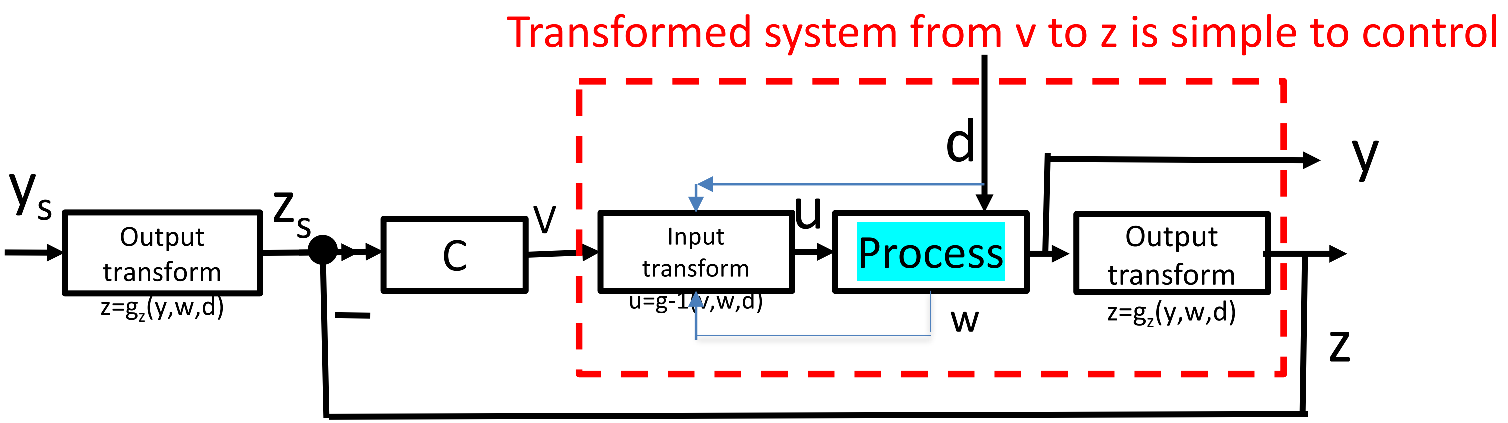
- Use feedback (cascade) implementation
- Slave loop involves controlling v = T(s) u.
  - $T(s) = K_u + K_w G_w(s)$
  - Unstable (RHP) zero or time delay in T(s) implies that slave loop cannot be fast
  - Uncertain model: Can tune slave controller based on experimental T.

NTNU

Transformed output

$z=g_z(y,_{w,d})$

Main idea: Simpler/more linear model for z than for y



Since we use the same transformation on both y and y_s, we will at steady state get $y=y_s$.

Example: y=T (temperature), z=H(T,p,x) (enthalpy).

Easy to write energy balance in terms of z=H

Further discussion…

- We have looked at many other examples
- And in particular we have looked at the effect of uncertainty
  - No big surprises
  - It’s fairly robust!
  - Mater theses by Callum Kingstree and Simen Bjorvand

NTNU

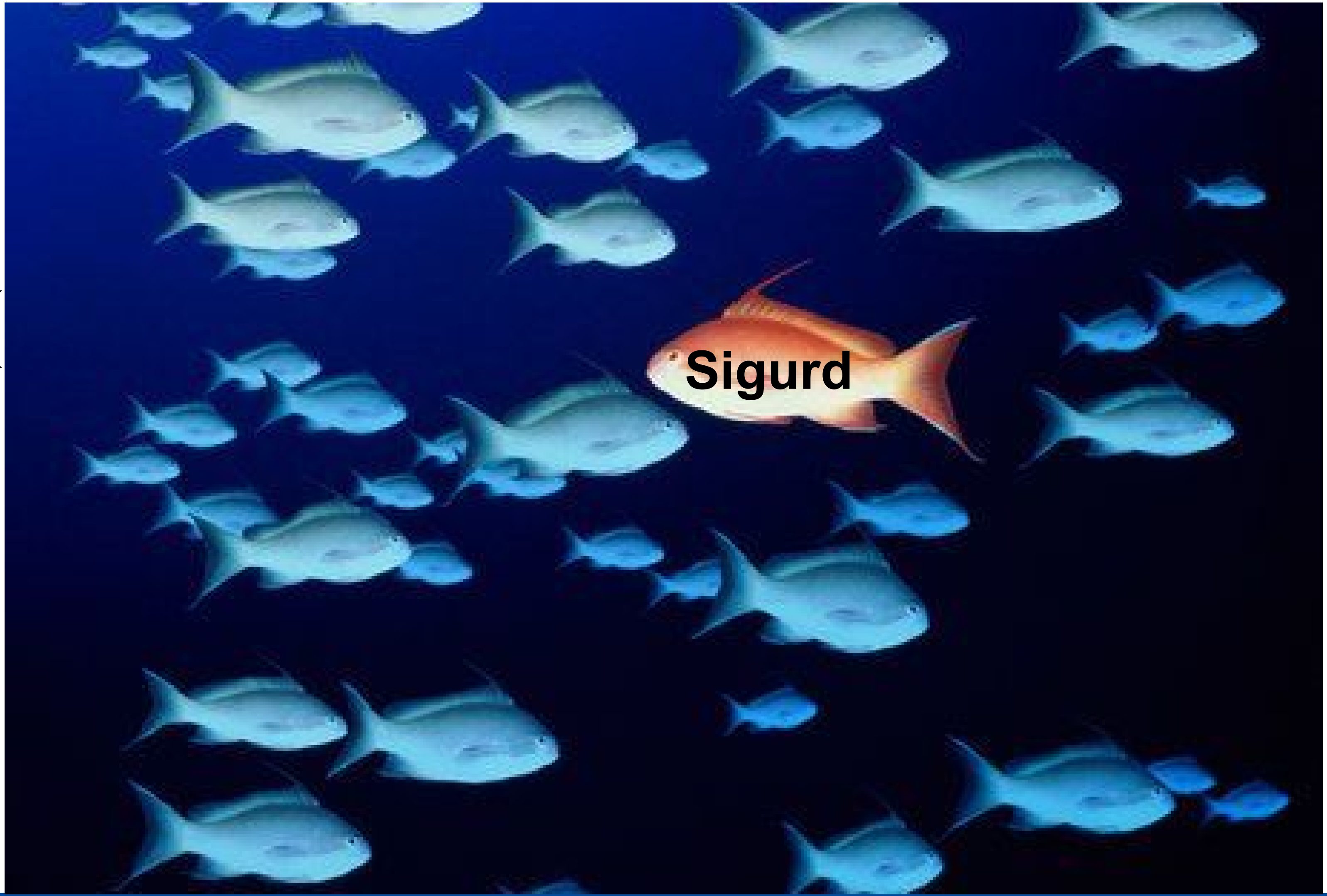
Conclusion

- "Control structures with embedded knowledge through input and output transformations"
- **Based on simple process models, easy to understand and implement**
- Systematic approach for dynamic model
  - $\frac{dy}{dt} = f(u, w, y, d)$
- Transformed input (B=I): $v = f(u, w, y, d) - Ay$
- Can also hande static models: $y=f_0(u,w,d)$. Use $v_0 = f_0(u, w, d)$
- Resulting transformed system from v to y:
  - Linear, independent of disturbances, decoupled
- Potential internal instability with exact inverse
  - No problem if indirect effect on v through w is small
  - Otherwise use cascade implementation

Academic control community fish pond

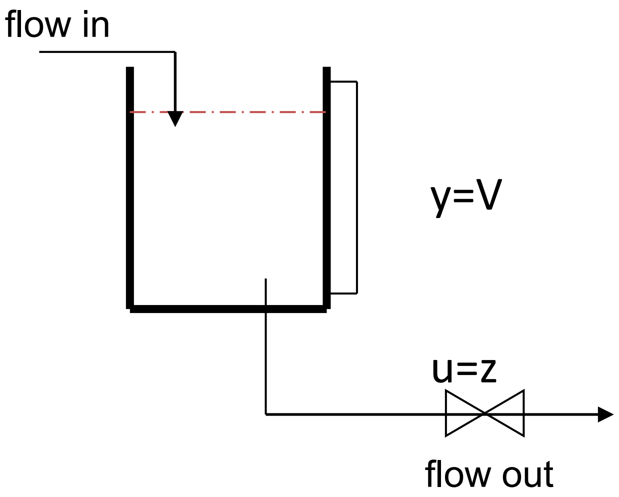
Complex optimal centralized Solution (**EMPC, FL**)

Simple solutions that work (PID++)



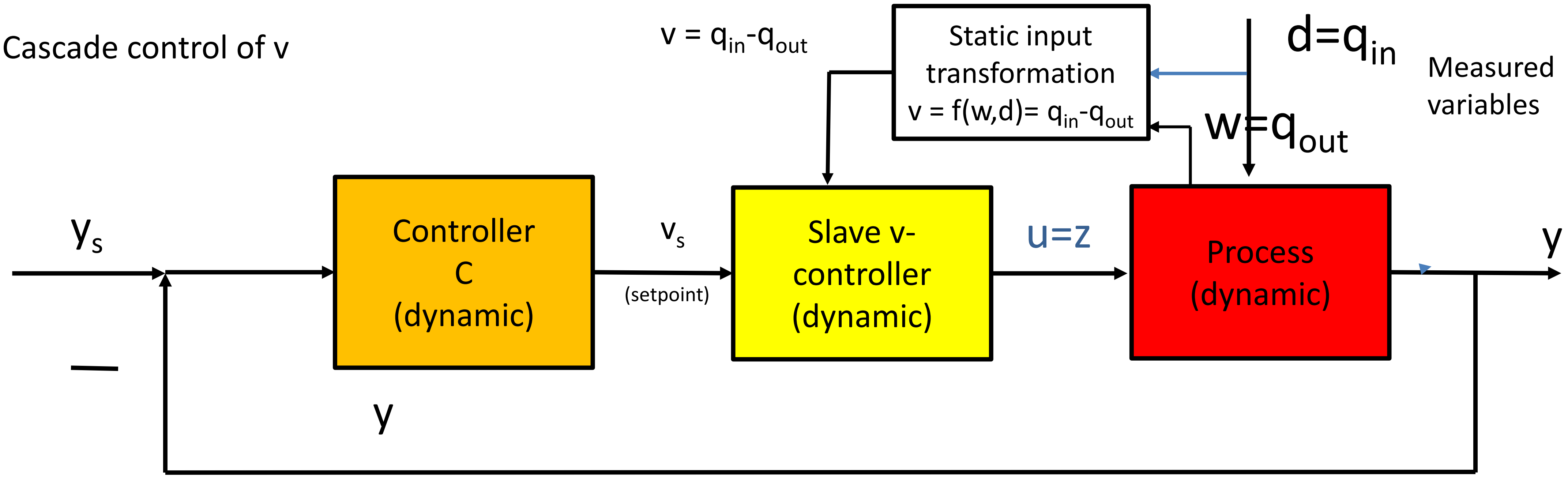
Extra

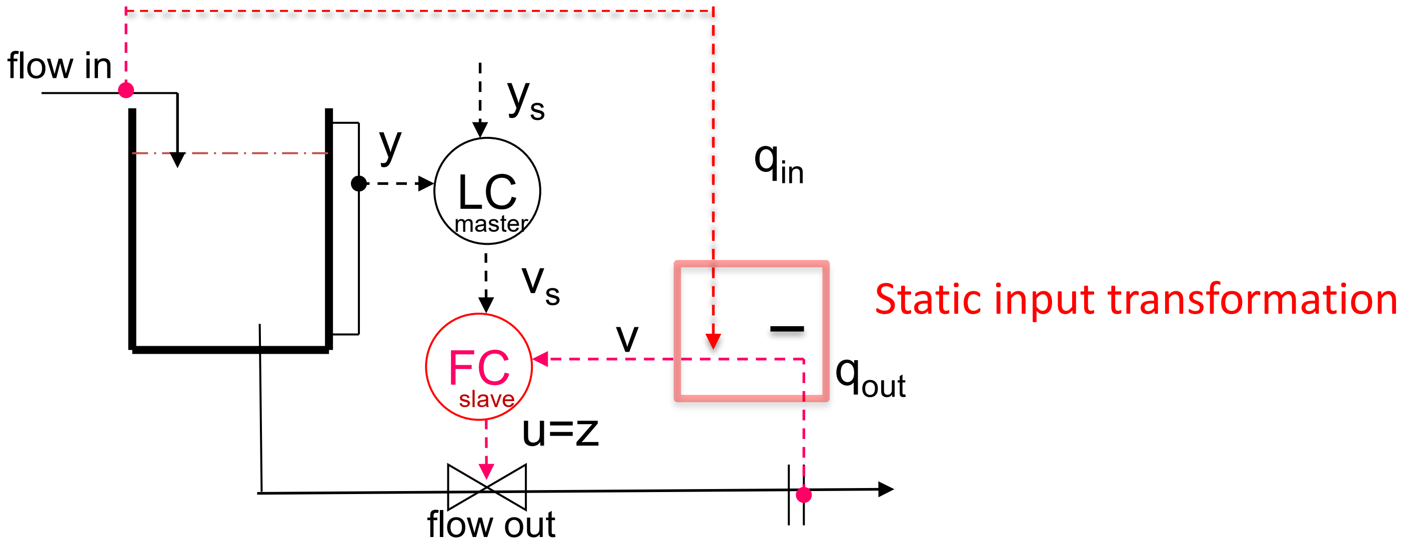
Example 5x. Level control with flow controller



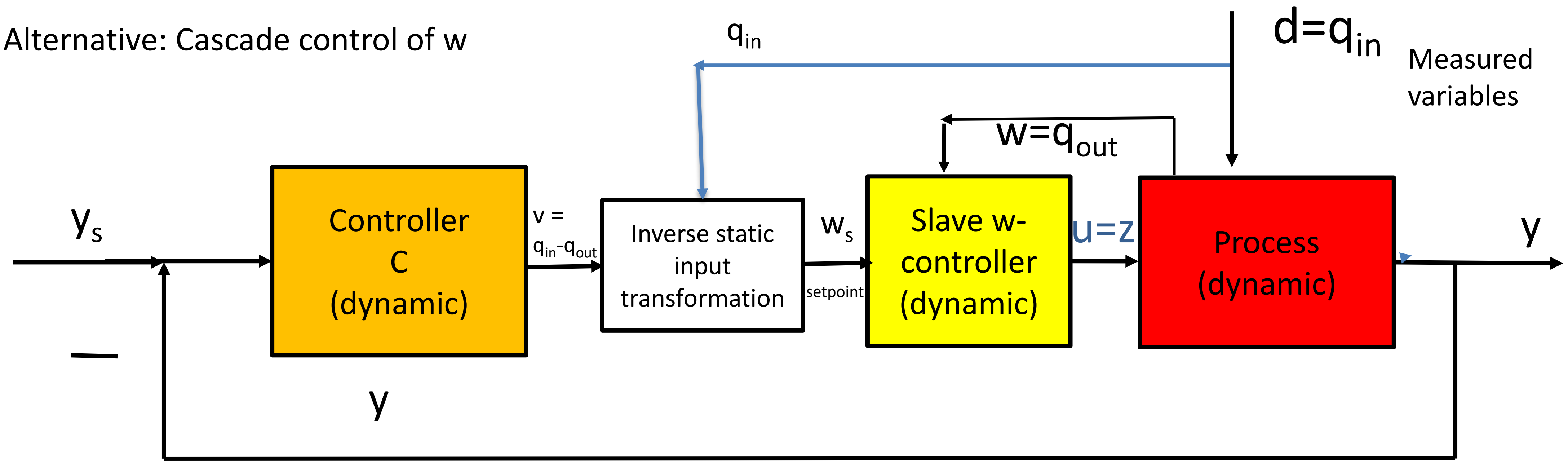
- $y = V$ (level), $u = z$ (valve position), $d = [q_{in}, \Delta P]$
- Model (mass balance): $\frac{dV}{dt} = q_{in} - q_{out}$
  - where (valve equation): $q_{out} = c_V f_{V(Z)} \sqrt{\frac{\Delta P}{\rho}}$
  - $f_v(z)$: nonlinear valve characteristic
- Can use «standard method» with: $f(y, u, d) = q_{in} - c_V f_{V(Z)} \sqrt{\frac{\Delta P}{\rho}}$
  - $v_A = f(y, u, d)$
  - Invert f to find u from given v_A
  - Complicated + Valve characteristic $f_v(z)$ uncertain + need measurement of DP
- Much better if q_{out} is measured: Introduce $w = q_{out}$ and use cascade control
  - Tranformed input: $v = q_{in} - q_{out}$
  - Equivalent to standard solution with cascade control based on flow controller

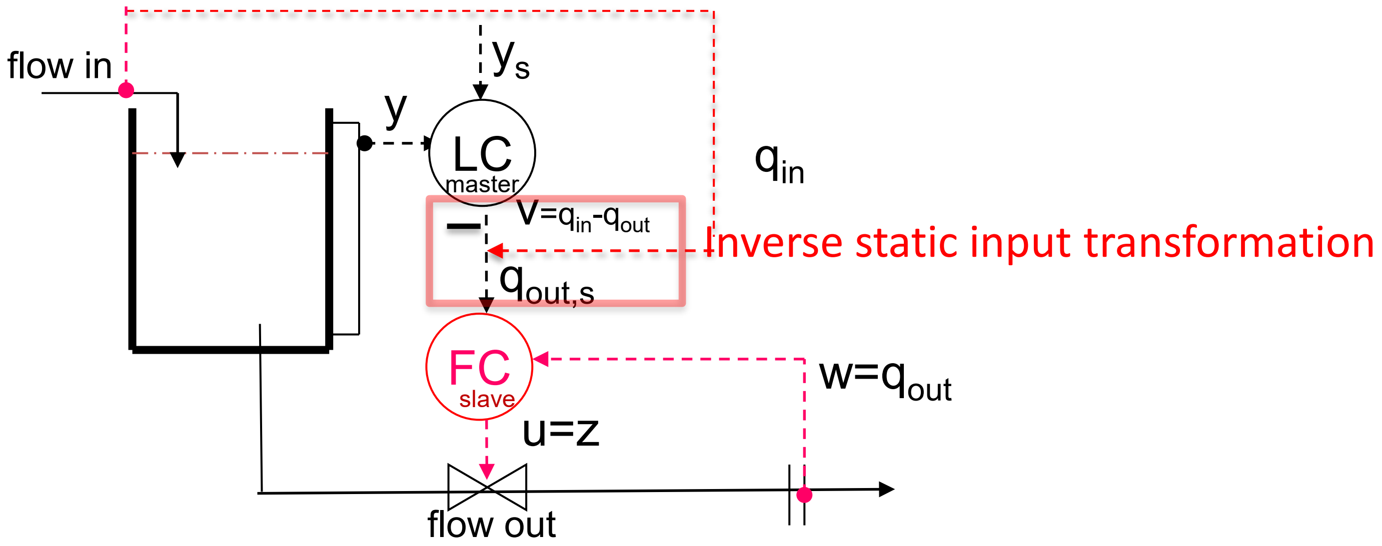
Cascade control of v



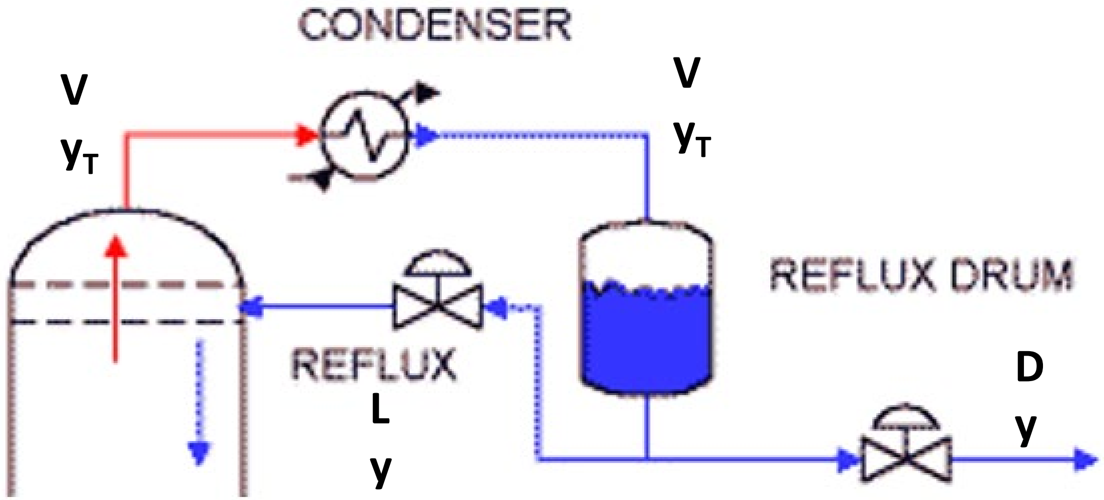


Alternative: Cascade control of w





Example 6: Distillation



y = distillate compostion
u= L (reflux)

Model reflux drum (component balance):

$$M\frac{dy}{dt} = V(y_T - y)$$

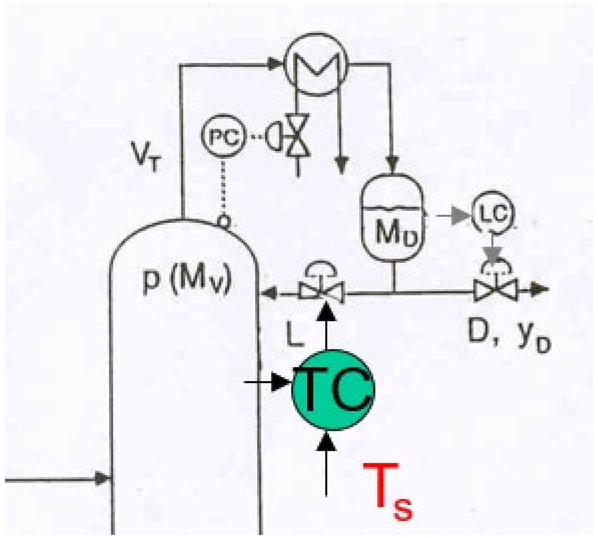
Note: $f = \frac{V}{M}(y_T - y)$ does not depend explicity on u=L.

But y_T depends indirectly on L. Introduce $w=y_T$

$$v_A = f - Ay = \frac{V}{M}(y_T - y) - Ay$$

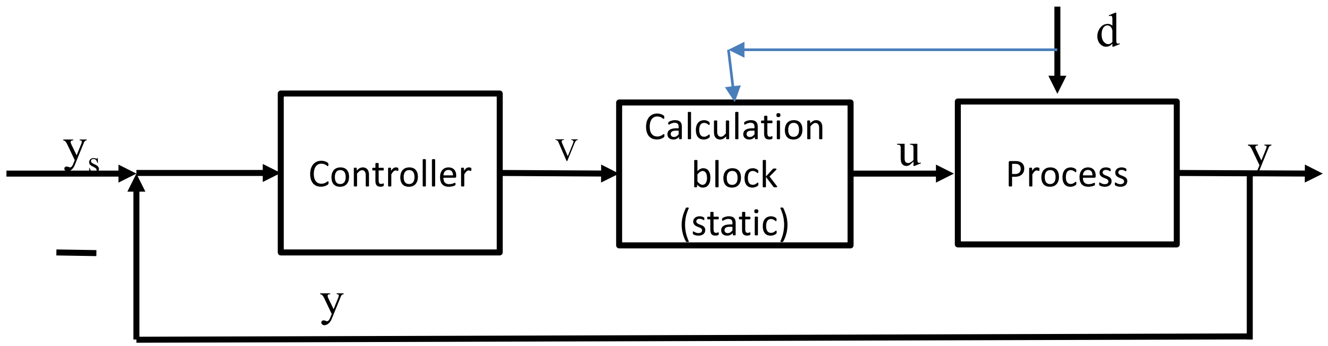
Solution: Cascade control of v_A or $w=y_T$

- y_T is difficult to measure
- But y_T is closely related to temperature
- This leads us towards the conventional solution with temperature cascade!



Nonlinear decoupling and feedforward using calculation blocks*

- Linear decoupling and feedforward often work poorly because of nonlinearity
- Example of nonlinear feedforward: Ratio control
- Generalization: Nonlinear calculation block



Method: Select «transformed inputs» v as right hand side of steady state model equations

*Zotica, Alsop and Skogestad. 2020 IFAC World Congress

Example: Combined nonlinear decoupling and feedforward. Mixing of hot and cold water

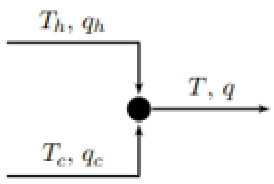
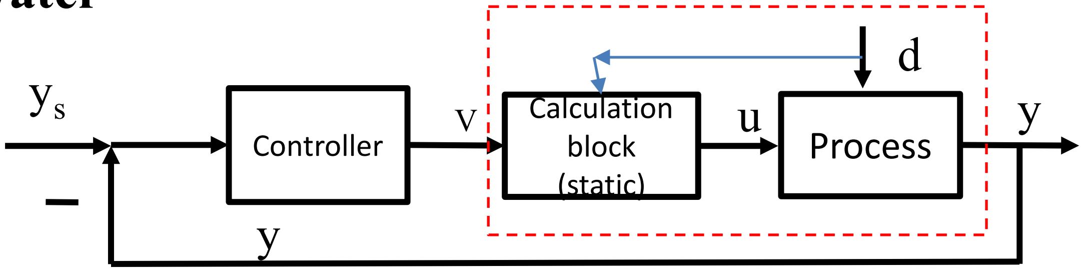


Figure 1: Mixer system



$$u = \begin{pmatrix} q_h \\ q_c \end{pmatrix} \quad d = \begin{pmatrix} T_h \\ T_c \end{pmatrix} \quad y = \begin{pmatrix} T \\ q \end{pmatrix}$$

Steady-state model written as y=f(u,d):

$$T = \frac{q_h T_h + q_c T_c}{q_h + q_c}$$

$$q = q_c + q_h$$

Select transformed inputs as right hand side, v = f

$$v_1 = \frac{q_h T_h + q_c T_c}{q_h + q_c} \quad (1)$$

$$v_2 = q_c + q_h \quad (2)$$

Model from v to y (red box) is then decoupled and with perfect disturbance rejection:

$$T = v_1$$

$$q = v_2$$

- Can then use two single-loop PI controllers for T and q!
  - These controllers are needed to correct for model errors and unmeasured disturbances
- Note that v_1 used to control T is a generalized ratio, but it includes also feedforward from Tc and Th.

Implementation (calculation block) : Solve (1) and (2) with respect to u=(qc qh):

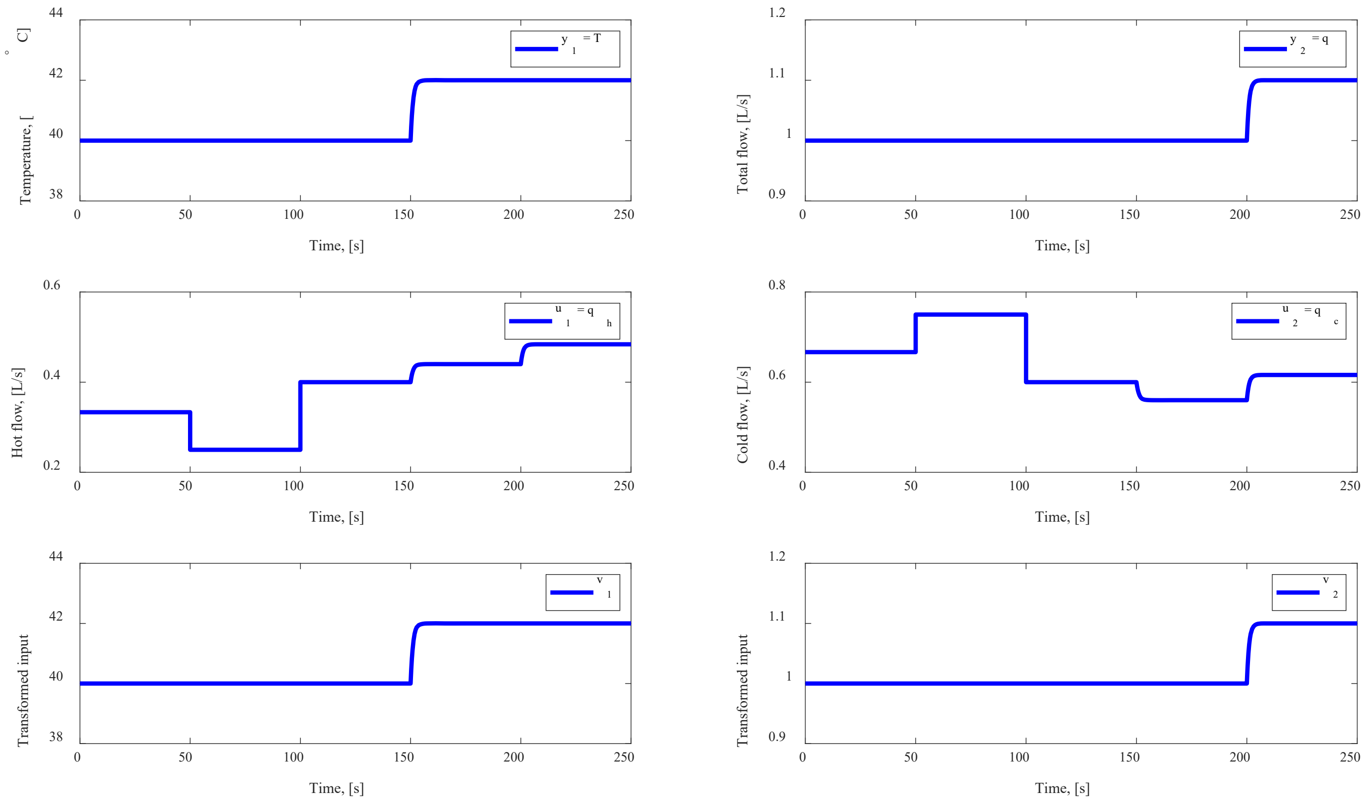
$$q_h = \frac{v_2(v_1 - T_c)}{T_h - T_c}$$

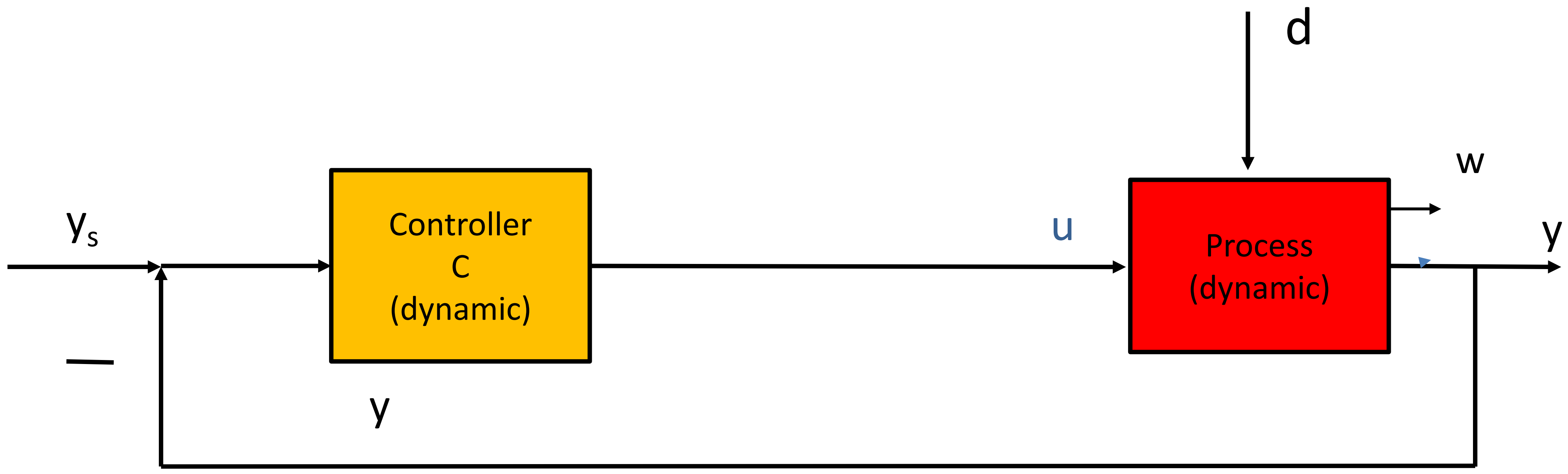
$$q_c = v_2 - q_h$$

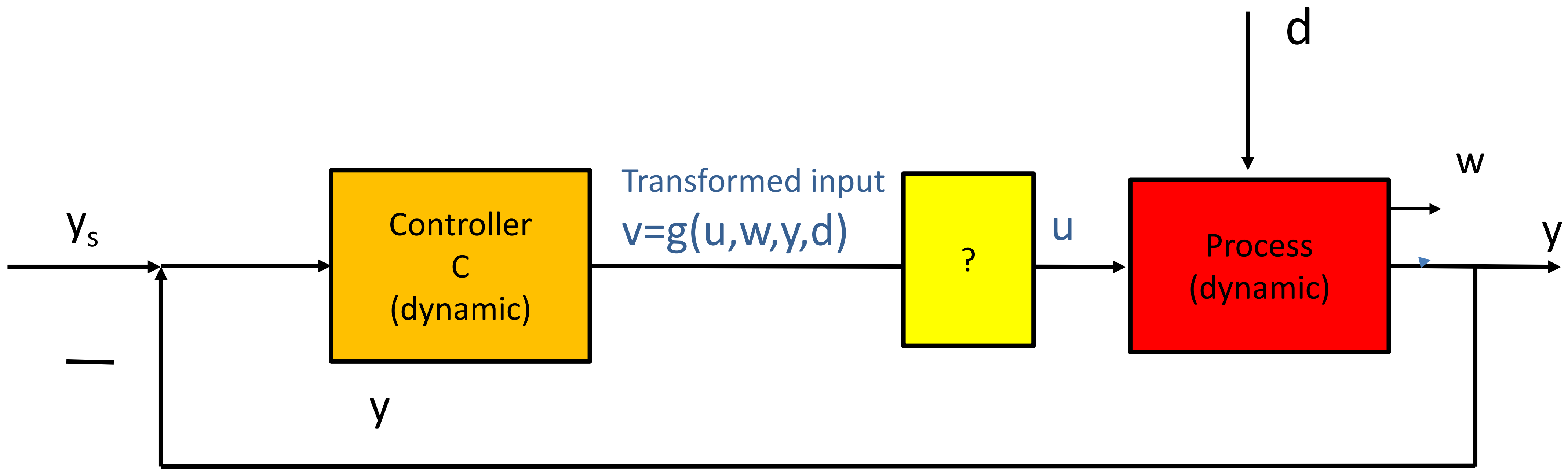
Transformed MVs for decupling, linearization and disturbance rejection
Mixing of hot and cold water (static process)
New system: $T=v_1$ and $q=v_2$

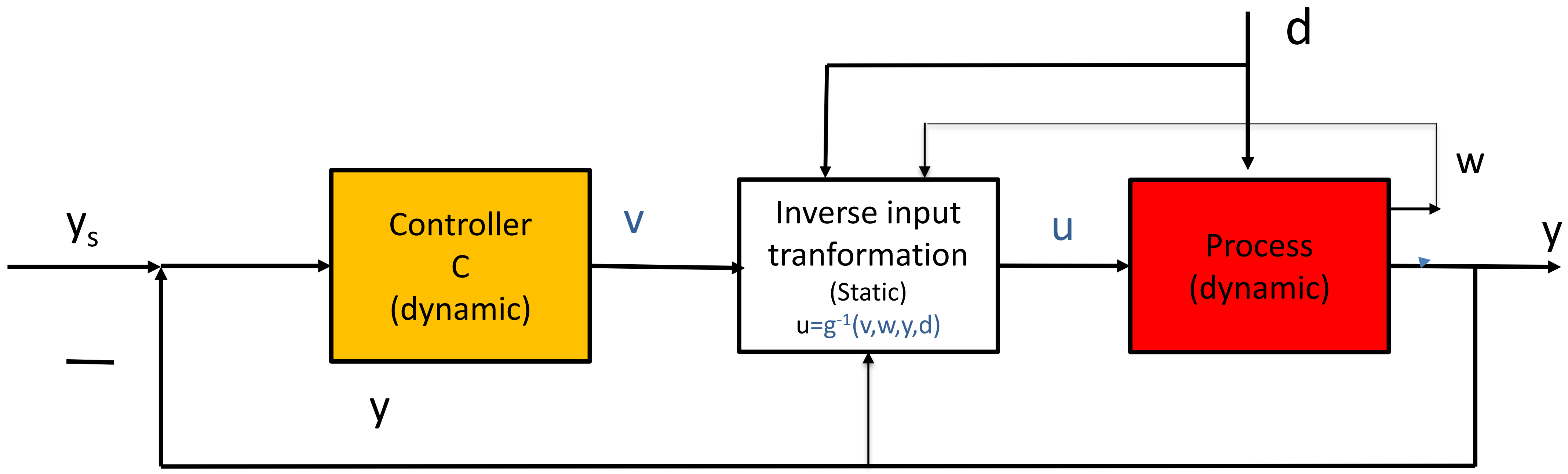
Outer loop: Two I-controllers with $\tau_C = 1$ s

1. T_h: 60->70 °C at t = 50 s
2. T_c: 30->20 °C at t = 100 s
3. T_h^s: 40->42 °C at t = 150 s
4. q^s: 1->1.1 L/s at t = 200 s

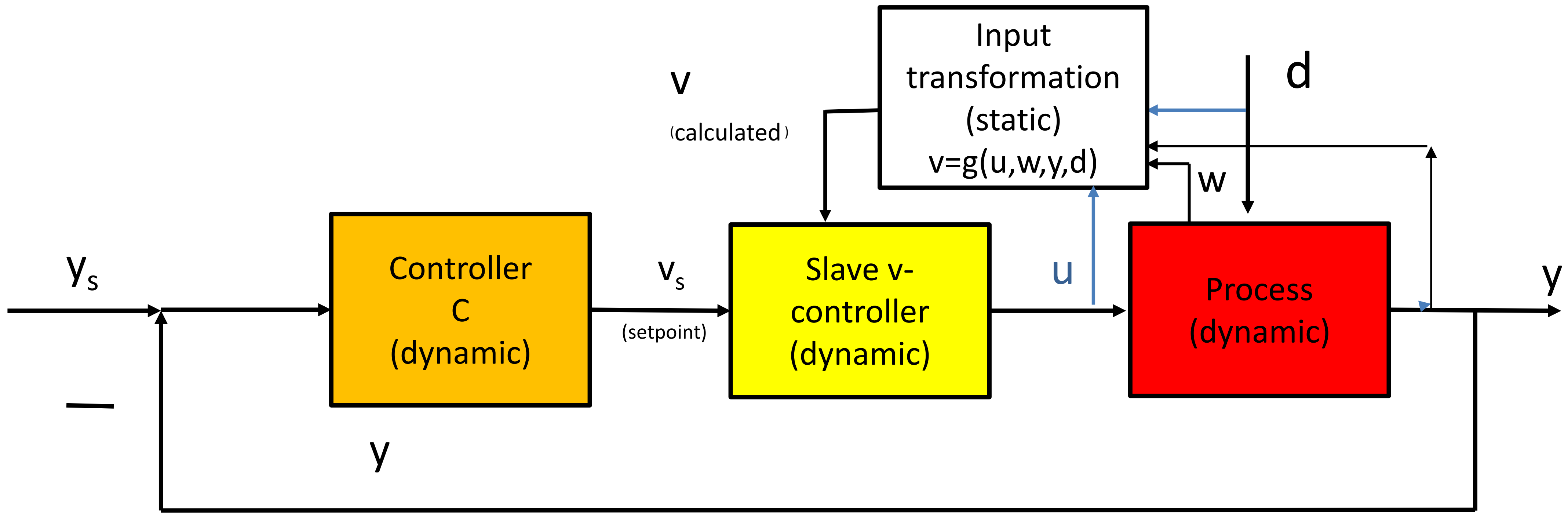



d
w
ys
Controller
C
(dynamic)
u
Process
(dynamic)
y
y


d
w
y_s
Controller
C
(dynamic)
Transformed input
v=g(u,w,y,d)
?
u
Process
(dynamic)
y
–
y

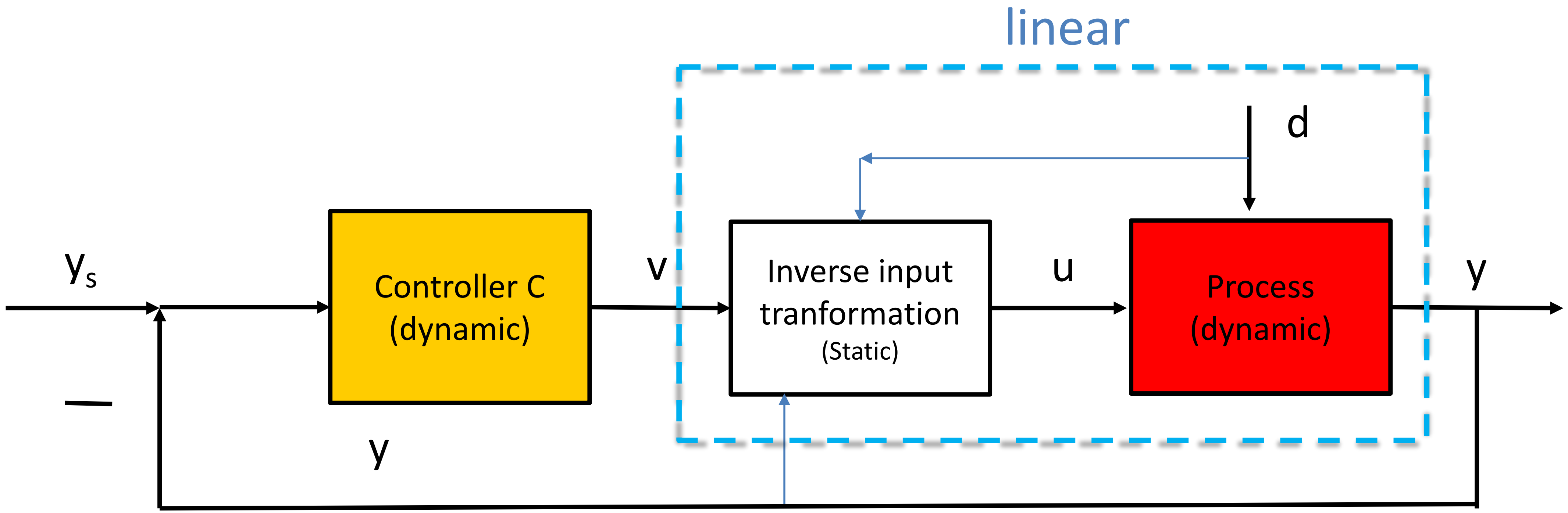


d
w
y_s
Controller
C
(dynamic)
v
Inverse input
tranformation
(Static)
$u=g^{-1}(v,w,y,d)$
u
Process
(dynamic)
y
–
y



Input
transformation
(static)
v=g(u,w,y,d)
v
(calculated)
d
w
ys
Controller
C
(dynamic)
vs
(setpoint)
Slave v-
controller
(dynamic)
u
Process
(dynamic)
y
–
y

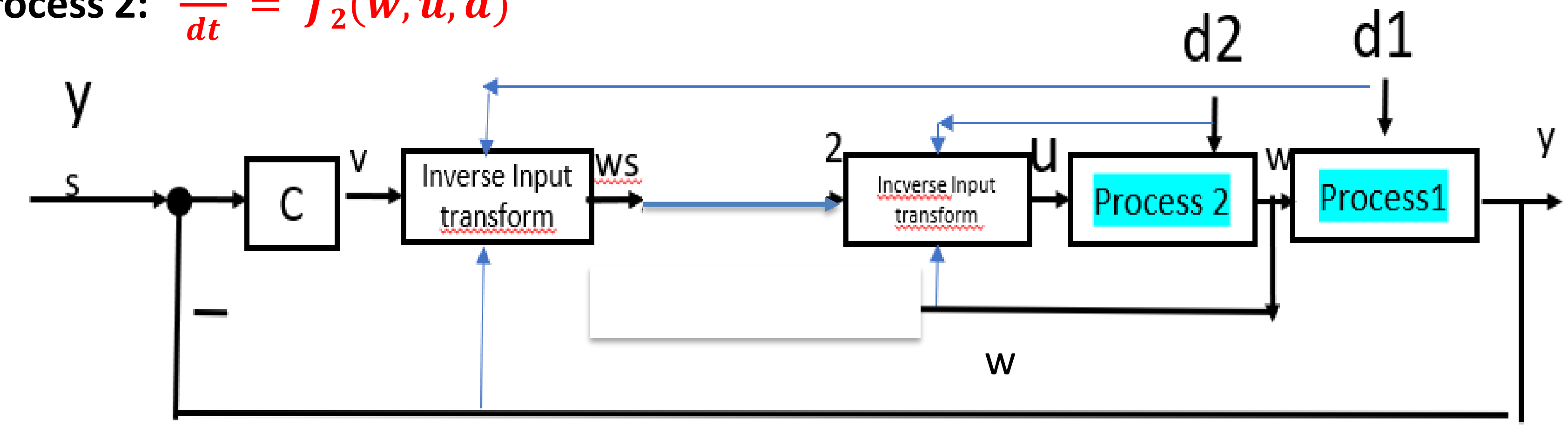
NTNU


linear
d
y_s
Controller C
(dynamic)
v
Inverse input
tranformation
(Static)
u
Process
(dynamic)
y
–
y

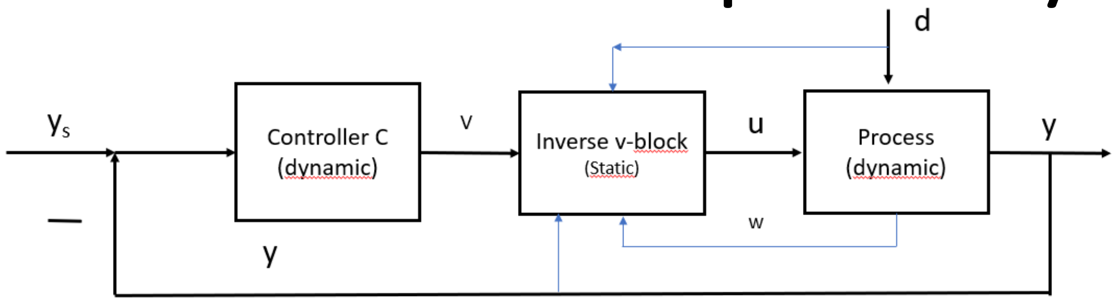
Special case (series system, f is independent of u) : Control of by Chain of transformations

Process 1: $\frac{dy}{dt} = f(y, w, d)$

Process 2: $\frac{dw}{dt} = f_2(w, u, d)$



Some cases: Slave controller can be replaced by static block



- NO: series system (f independent of u. Here a static block for u is impossible so we must use cascade control. Problem: may be difficult to get fast slave loop)
- MAYBE parallell system (dangerous: may get unstable zero dynamics, so recommend cascade)
- YES. recycle system (no big problem, at least if delayed, since recycle gives positive feedback, here a static block may be OK)

- Recycle system
  - $y = G_1 (u + G_2 y)$
  - $y = G_1/(1 - G_2)\, u = T(s)\, u$
  - $G_2 = \frac{k_2 \exp(-\theta_2 s)}{d_2(s)}$
  - $T = \frac{G_1 d_2(s)}{d_2(s) - k_2 \exp(-\theta_2 s)} = G_1$ for initial response (s=infinity)
- But be careful. Cascade is safer because then we can get real dynamics experimentally.