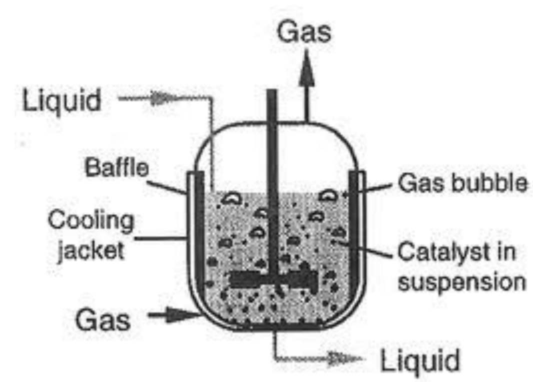
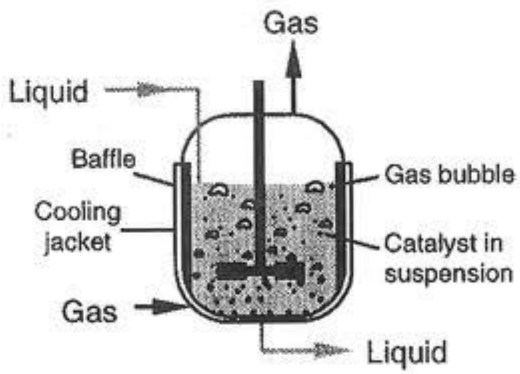
ENGR691X: Fault Diagnosis and Fault Tolerant Control Systems
Fall 2010

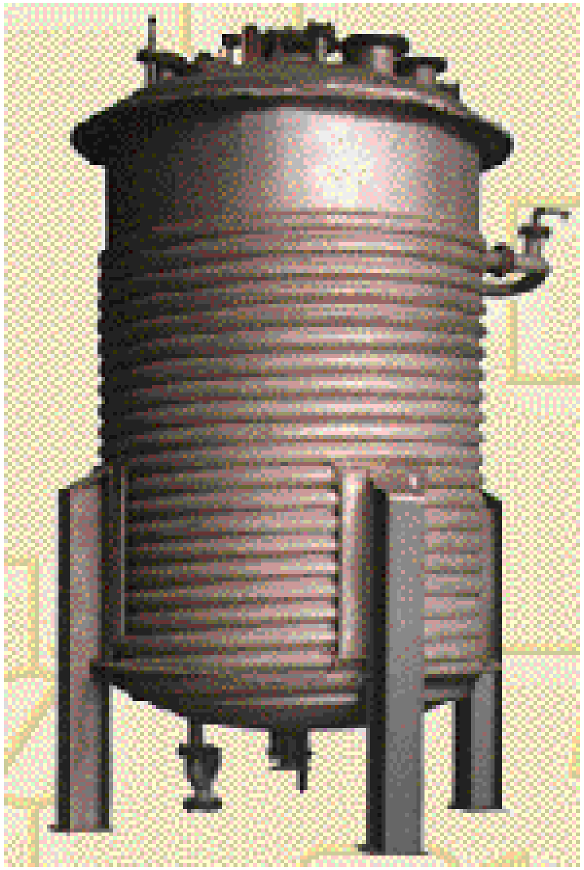
Adaptive Fault Tolerant Control of an unstable Continuous Stirred Tank Reactor (CSTR)

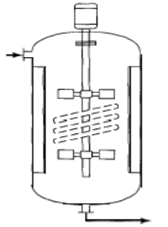
Group Members: Maryam Gholamhossein
Ameneh Vatani



Why CSTR?

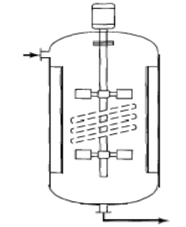

Gas
Liquid
Baffle
Cooling jacket
Gas
Gas bubble
Catalyst in suspension
Liquid





Outlines

- Continuous Stirred Tank Reactor (CSTR) Model
- Controller
  - Conventional controller
  - Fuzzy logic
  - Adaptive Fault tolerant fuzzy PID controller
- Simulation results & analysis under different fault scenarios
- Conclusion and suggestions

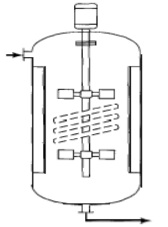


Reactors

- Chemical reactors are one of the most important part of chemical, biochemical and petroleum processes since they transform raw materials into valuable chemical materials.

- Three classical chemical reactors
  - Batch reactor
  - Continuous stirred-tank reactor (CSTR)
  - Plug flow tubular reactor (PFR)

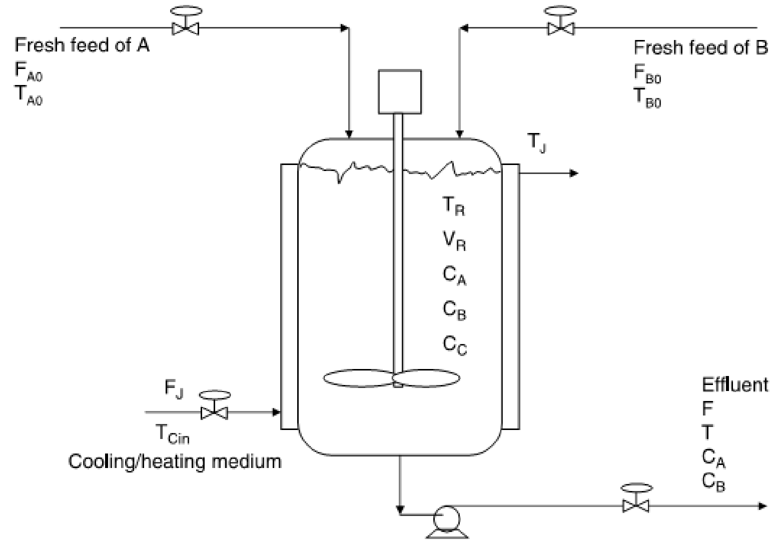
CSTR Model



• The CSTR reactor is usually used for liquid-phase or multiphase reactions that have high reaction rates. Reactant streams are continuously fed into the vessel.

• Perfect mixing of the liquid in the reactor is usually assumed, so the modeling of a CSTR involves ordinary differential equations.

CSTR Model



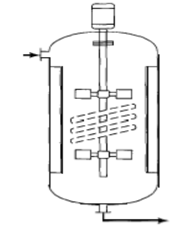
- Main characteristics of a CSTR
  - Constant temperature
  - Constant concentration

- Reaction types:
  - Exothermic (releasing energy)
  - Endothermic (requiring energy)

  - Reversible (balance of reactants and products) $A + B \leftrightarrow C + D$
  - Irreversible (proceeding completely to products) $A + B \rightarrow C$

  - Homogeneous (single-phase)
  - Heterogeneous (multiphase)

CSTR Model

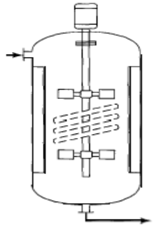


Exothermic and irreversible reactions

Temperature control problems

- Maintaining stable and safe temperature control
- Heat removal methods
  - Jacket cooling
  - Cooling coil

CSTR Model



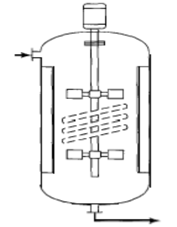
- Three-state CSTR model, exothermic-irreversible first-order reaction (A → B)

$$\frac{dC_a}{dt} = \frac{Q}{V}(C_{af} - C_a) - k_0 exp\left(\frac{-E_a}{RT}\right)C_a$$

$$\frac{dT}{dt} = \frac{Q}{V}(T_f - T) + \left(\frac{-\Delta H}{\rho C_p}\right)k_0 exp\left(\frac{-E_a}{RT}\right)C_a - \frac{UA}{V\rho C_p}(T - T_c)$$

$$\frac{dT_c}{dt} = \frac{Q_c}{V_c}(T_{cf} - T_c) + \frac{UA}{V_c \rho_c C_{pc}}(T - T_c)$$

- Dimensionless …



CSTR Model

- System dimensionless equations*:

x_1: dimensionless concentration x_2: dimensionless reactor temperature

q_c: dimensionless cooling jacket temperature

x_3: dimensionless cooling jacket flow rate x_{2f}: dimensionless feed flow

$$\frac{dx_1}{dt} = q(x_{1f} - x_1) - x_1 k(x_2)$$

$$\frac{dx_2}{dt} = q(x_{2f} - x_2) - \delta(x_2 - x_3) + \beta x_1 k(x_2)$$

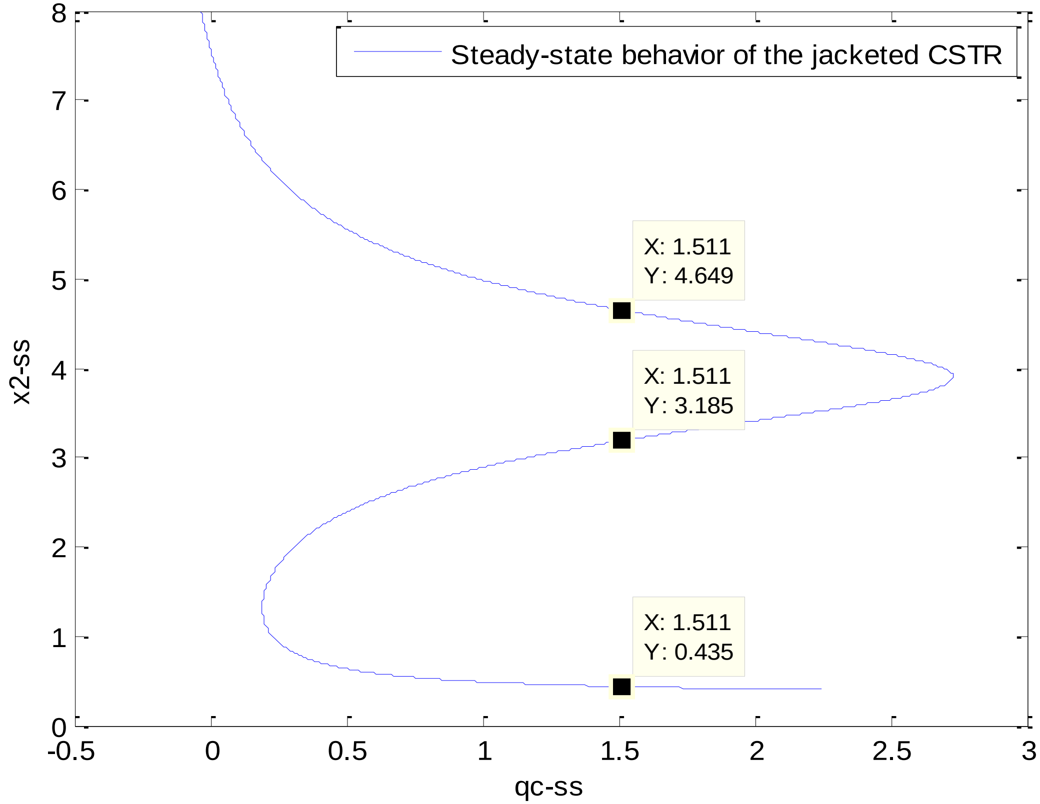
$$\frac{dx_3}{dt} = \delta_1[q_c(x_{3f} - x_3) + \delta\delta_2(x_2 - x_3)]$$

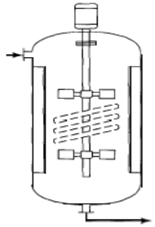
$$k(x_2) = exp\left(\frac{x_2}{1 + x_2/\gamma}\right)$$

* Russo L. P., Bequette B. W., “Impact of process design on the multiplicity behavior of a jacketed exothermic CSTR”, AIchE Journal, 41(1)135

CSTR Model

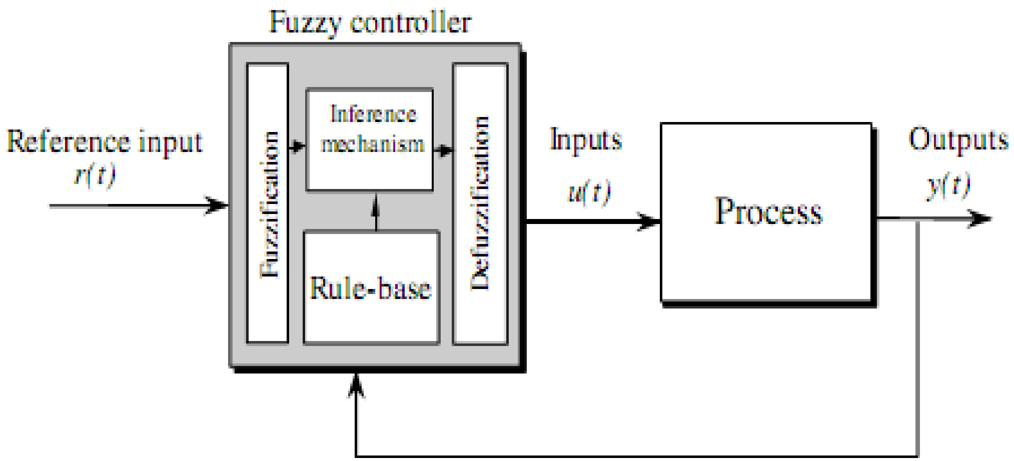
- System non-linearity
- Steady-State design and Multiplicity of CSTR

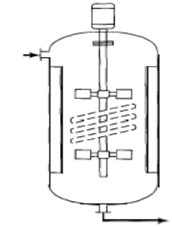




Controller Design

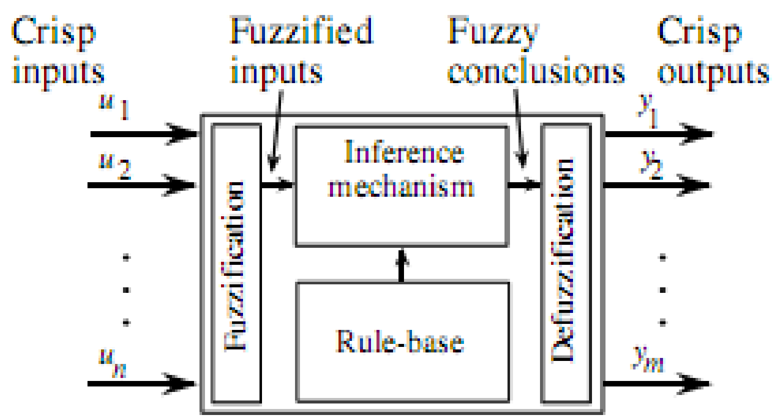
- Conventional controllers
  - (PID control, state-space methods, optimal control, robust control,…)
  - Designing based on the Mathematical models
  - Ignoring heuristic information, as they do not fit into proper mathematical form
- Fuzzy controller
  - An artificial decision maker that can operate in a closed-loop control system

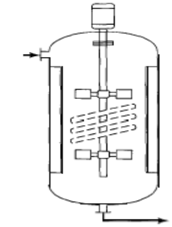




Controller Design

- *Rule-base*, holds the knowledge in the form of a set of rules of how best to control the system (a set of *If-Then* rules)
- *Inference mechanism* (inference engine) evaluates which control rules are relevant at the current time and deciding what the input to the plant should be
- *Fuzzification*, modifying the inputs so that they can be interpreted to the rules in the rule-base
- *Defuzzification*, converting the conclusions of inference mechanism into the plant inputs.

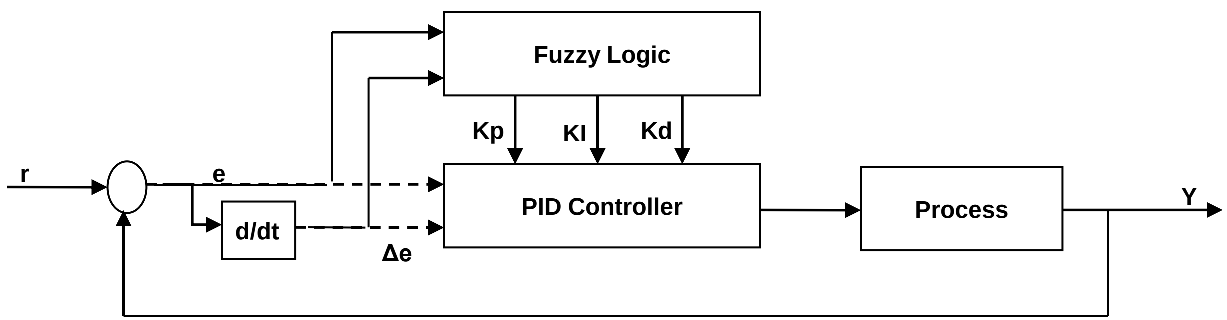




Controller Design

- Adaptive fuzzy controller scheme

  (Fuzzy controller and conventional controller combination)

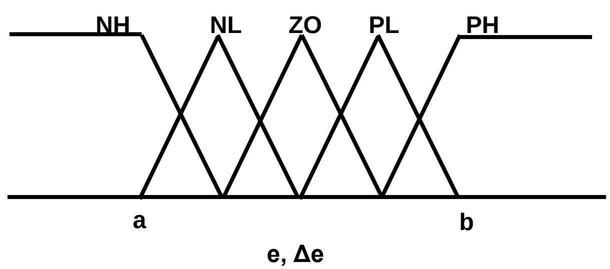


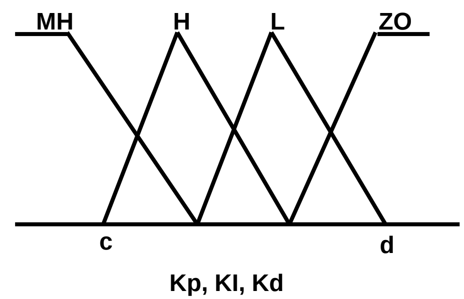
- Tracking and regulatory problem
  - Some continuous process produce different grades of products at different times

Controller Design

- Fuzzy adaptation module steps:

1) Defining the input & output membership functions





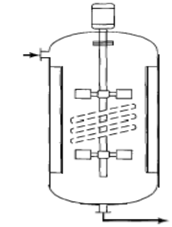
2) Defining the fuzzification and defuzzification methods

3) Defining Inference mechanism

4) Defining the Rules in the form of linguistic structure

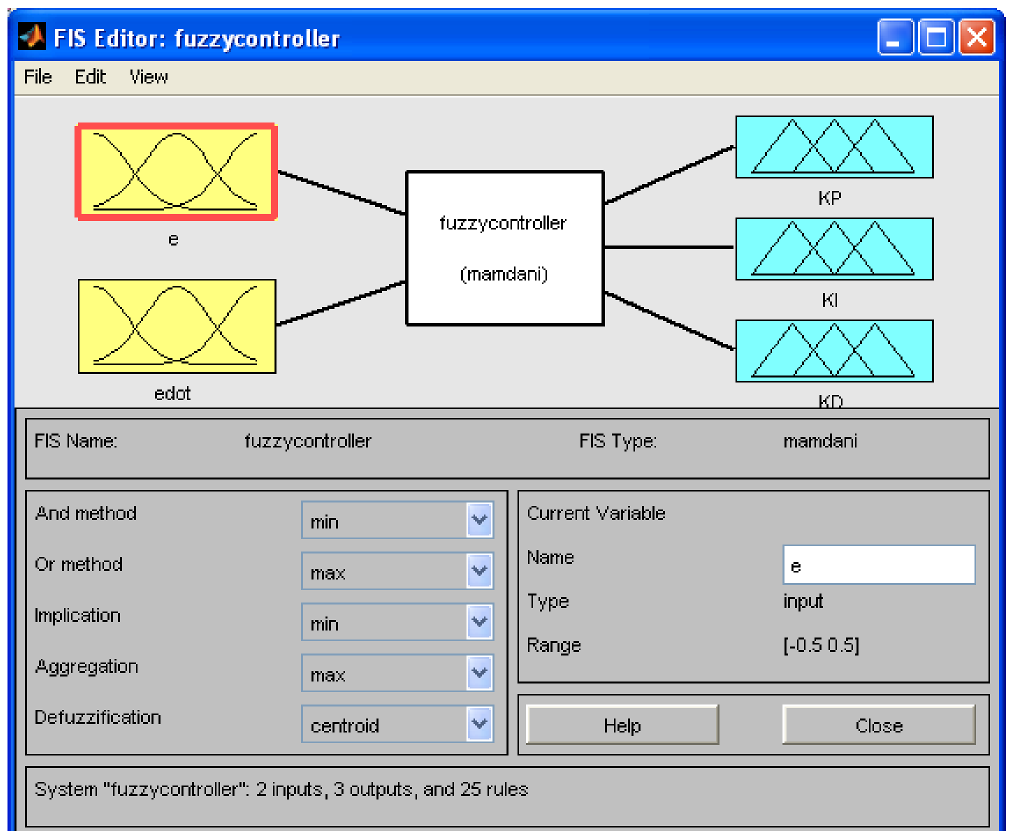
(one of fuzzy implementation challenges!)

If e is X *and* Δe is Y, *then* KI=U, Kp=V, Kd=Z

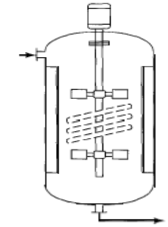


Controller Design

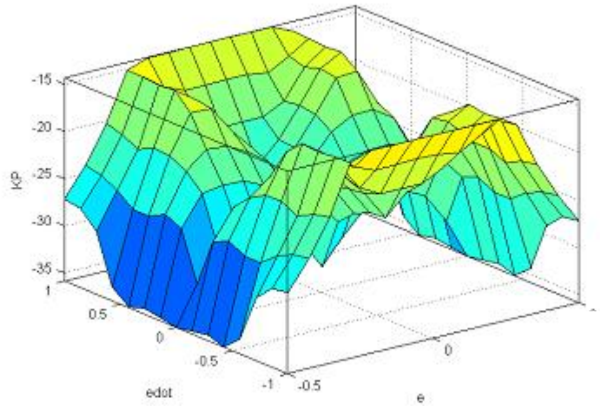
- Fuzzy controller inputs: Error (e) and error changes (Δe)
- Fuzzy controller outputs: PID gains (Kp,Kc,Kd)
- Fuzzy Inference Strategy: Mamdani
- defuzzification method: Centriod

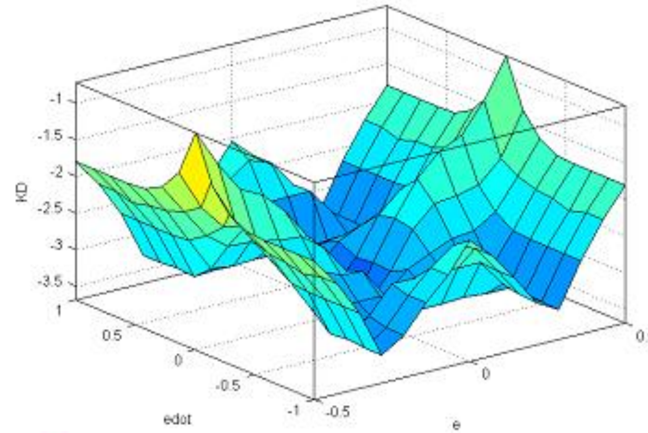


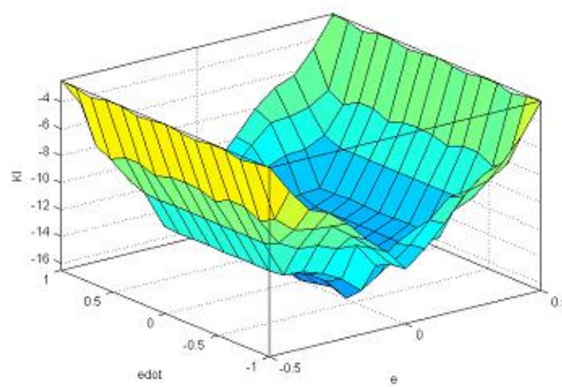
Controller Design



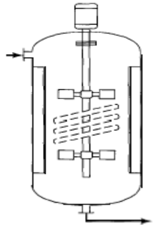
- AFTCS or PFTCS?!





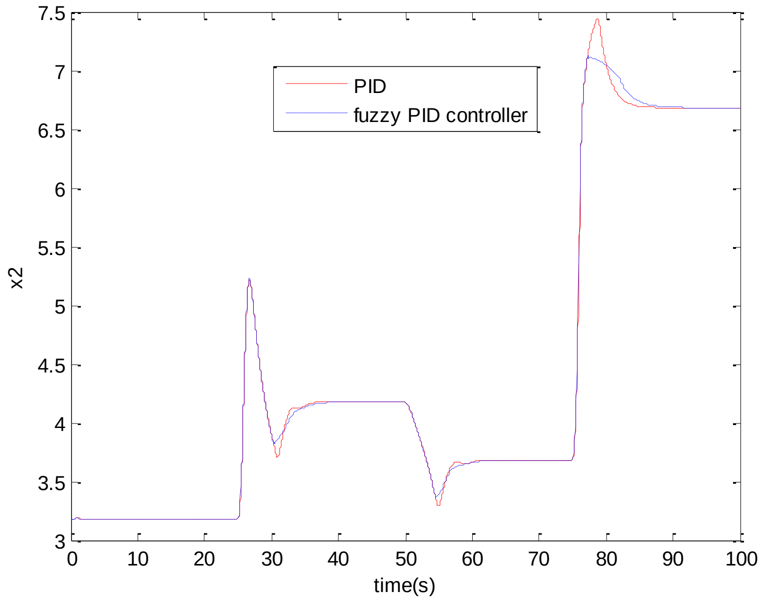


- So where is the FDD part?



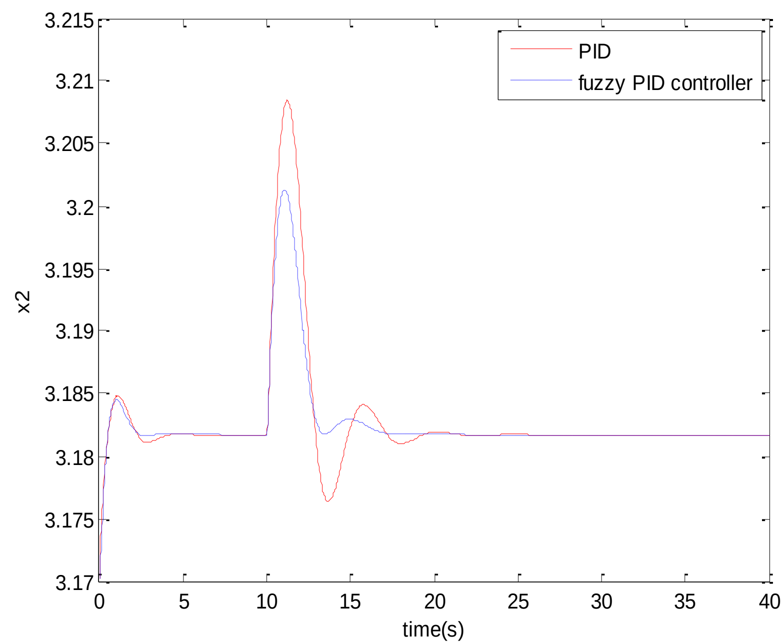
Simulation results

- Fault free tracking response

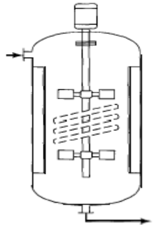


Simulation results

- Actuator faults scenarios
- System parameter fault scenarios
- Sensor Faults scenarios

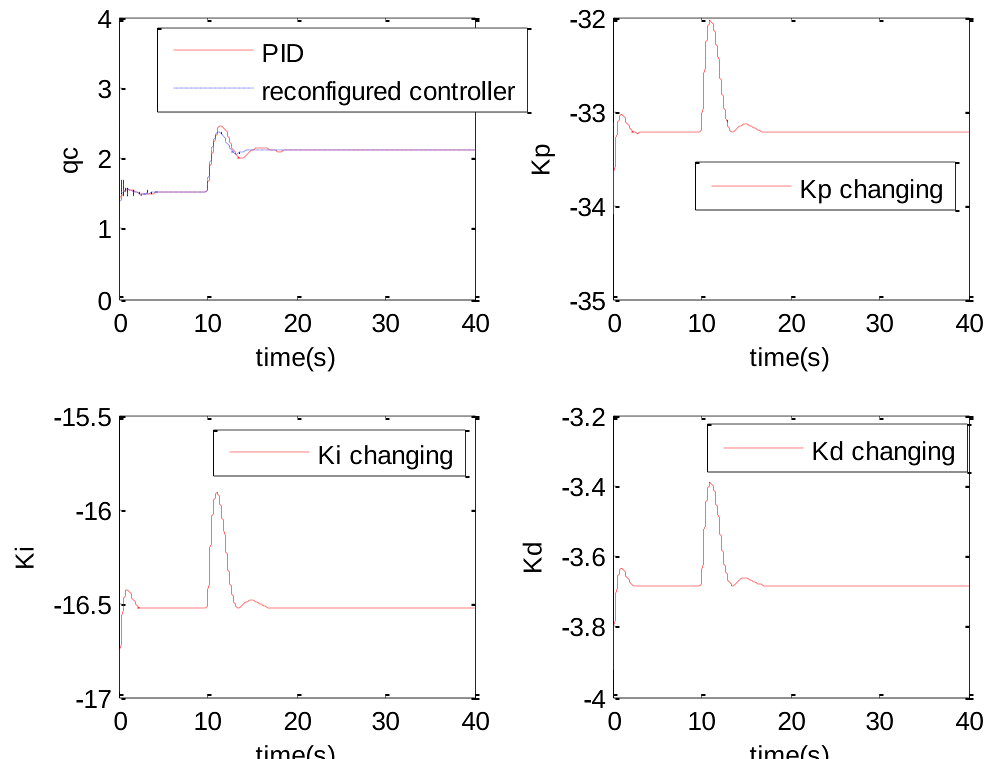


System output response to 15% actuator failure



Simulation results

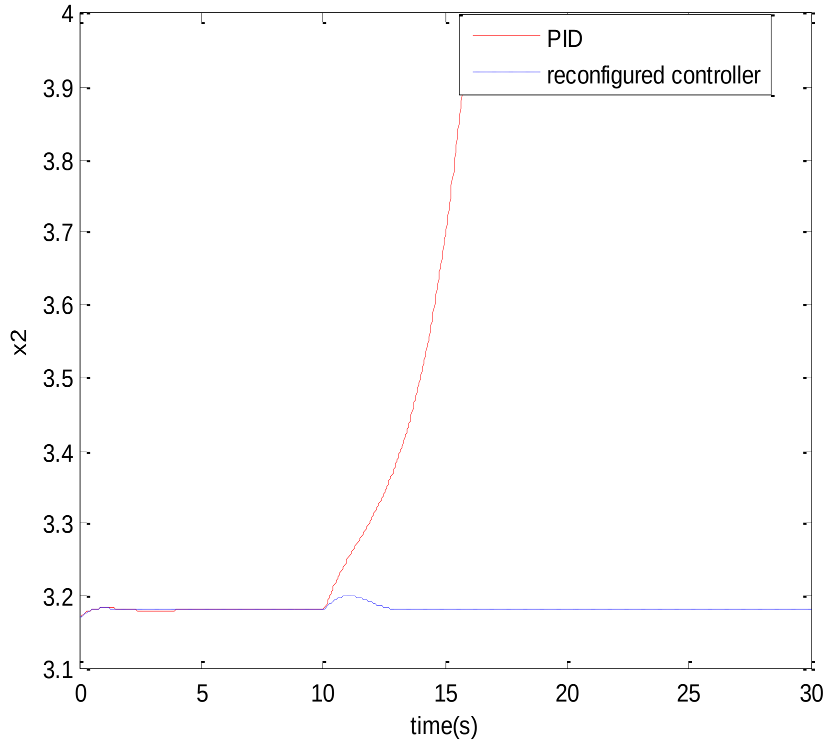
- Actuator faults scenarios
- System parameter fault scenarios
- Sensor Faults scenarios



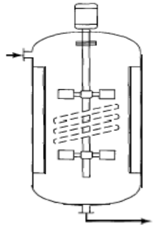
Control input signals and controller gains under 15% actuator failure

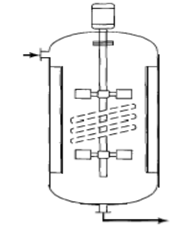
Simulation results

- Actuator faults scenarios
- System parameter fault scenarios
- Sensor Faults scenarios



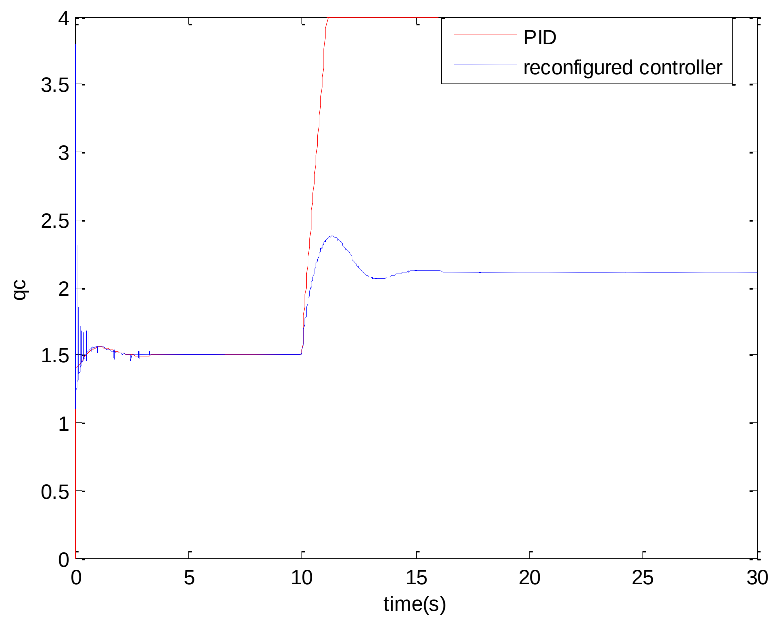
System output response to 25% actuator failure





Simulation results

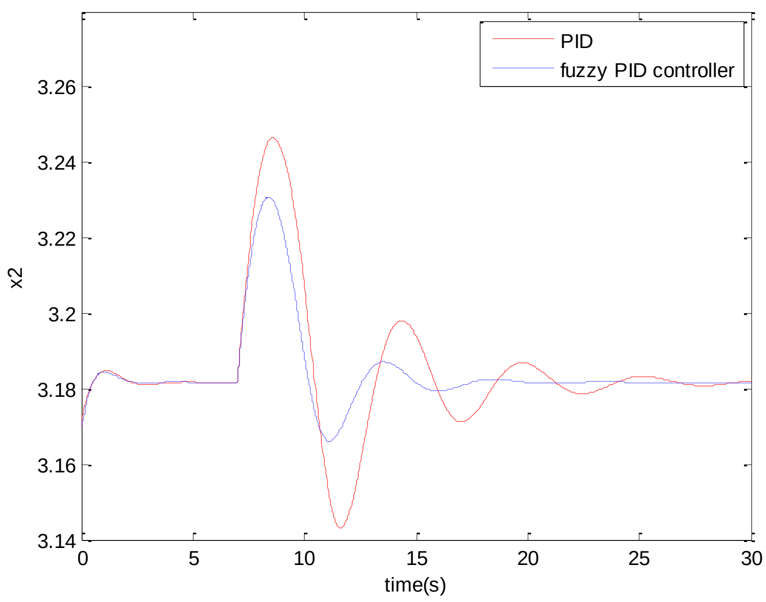
- Actuator faults scenarios
- System parameter fault scenarios
- Sensor Faults scenarios



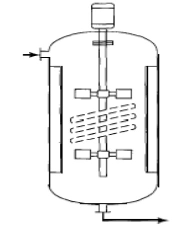
Control input signals under 25% actuator failure

Simulation results

- Actuator faults scenarios
- System parameter fault scenarios
- Sensor Faults scenarios

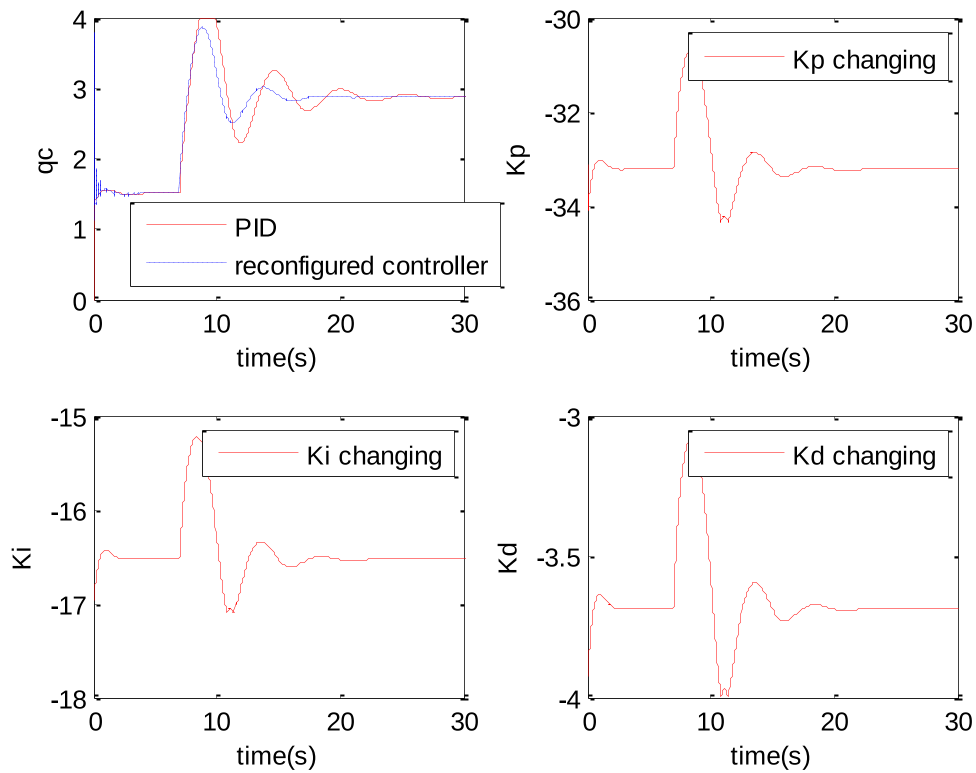


System output response to changing x2f from 0 to 0.08



Simulation results

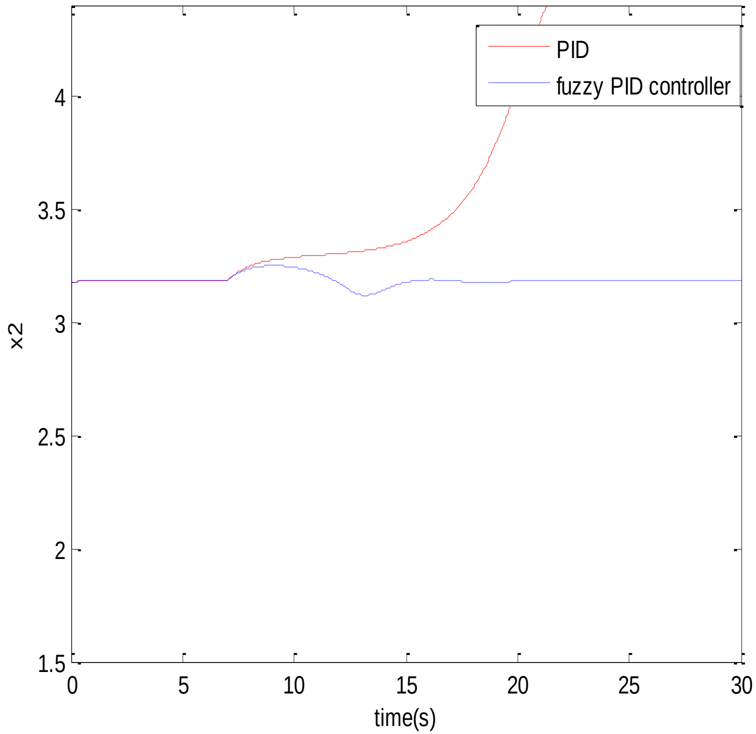
- Actuator faults scenarios
- System parameter fault scenarios
- Sensor Faults scenarios



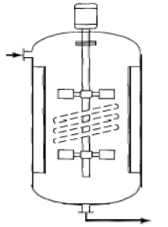
Control input signals and controller gains changing x2f from 0 to 0.08

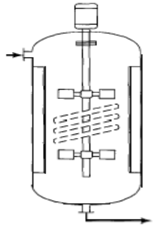
Simulation results

- Actuator faults scenarios
- System parameter fault scenarios
- Sensor Faults scenarios



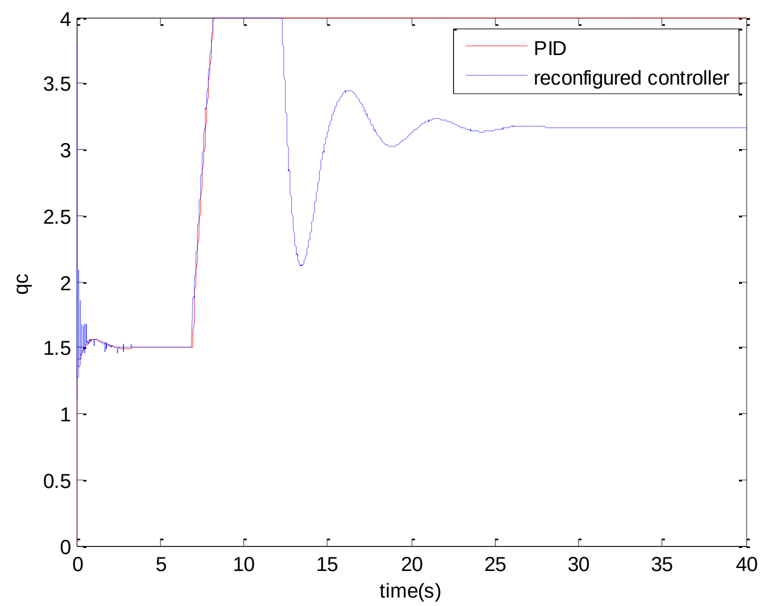
System output response to changing x2f from 0 to 0.1



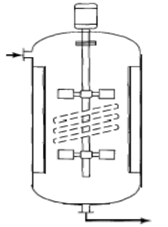


Simulation results

- Actuator faults scenarios
- System parameter fault scenarios
- Sensor Faults scenarios

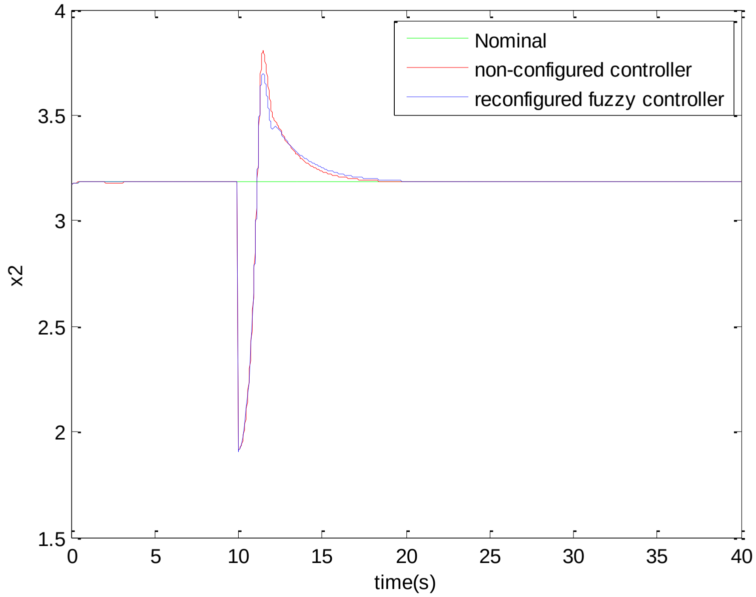


Control input signals to changing x2f from 0 to 0.1



Simulation results

- Actuator faults scenarios
- System parameter fault scenarios
- Sensor Faults scenarios



Output responses in the presence of 40% sensor fault

Conclusions & Suggestions

- In this project the fault tolerant control of a CSTR model under different faults is accomplished.
- Defining the proper fuzzy rules was a very challenging and time-consuming task!
- In spite of the conventional definition for Active FTCS which obligated the system to have a FDD block; here in this project FDD block is inherent in the fuzzy controller.
- When the fault percentage exceeds specific values, the conventional PID fails to control the CSTR while the fuzzy PID can have the pre-fault performance after a short transient time.

Conclusions & Suggestions

- Extending this controller to a MIMO system.
- Taking other parameters as input of fuzzy controller.

Thank you for your attention!

Simulation results