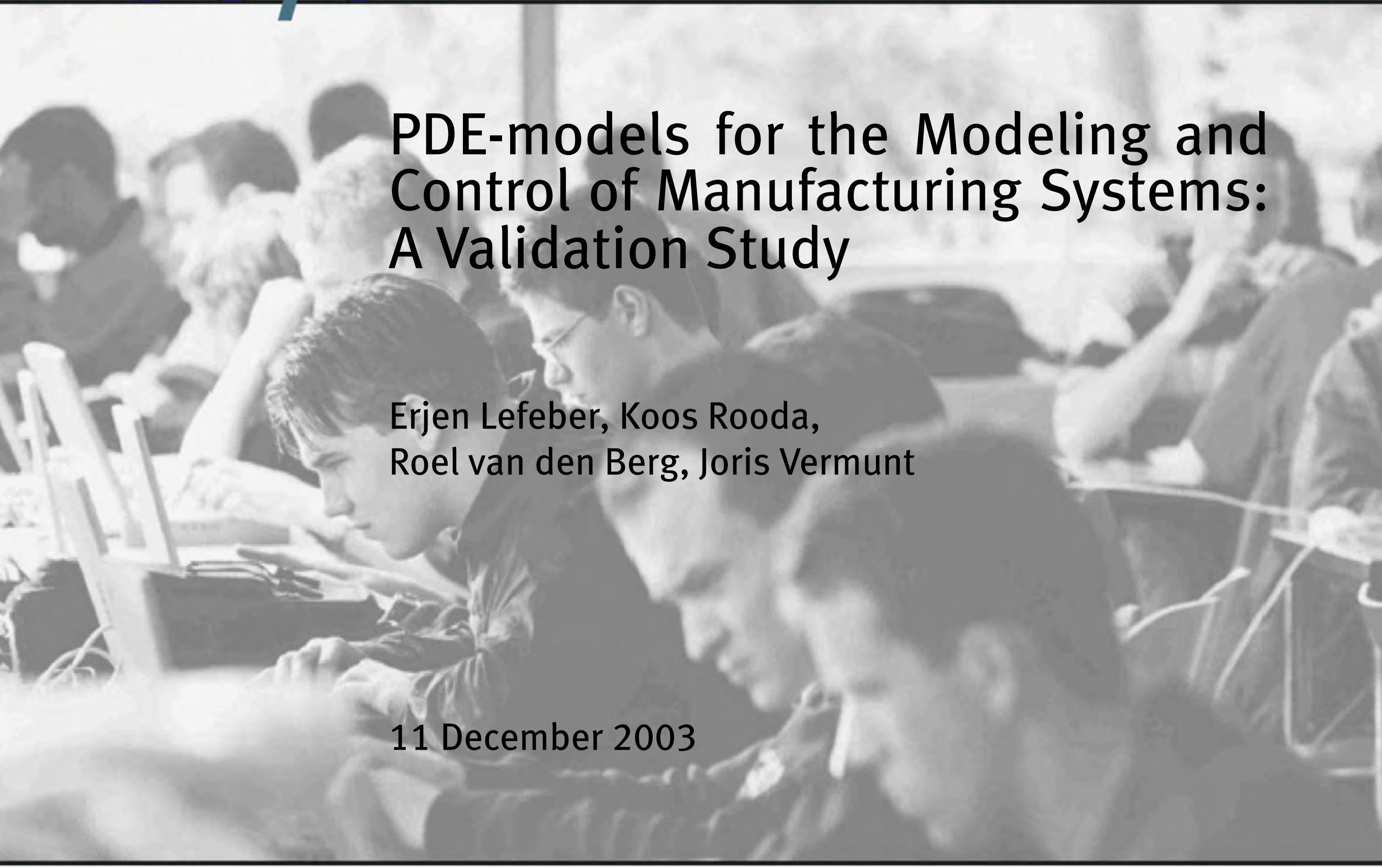
technische universiteit eindhoven



PDE-models for the Modeling and Control of Manufacturing Systems: A Validation Study

Erjen Lefeber, Koos Rooda,
Roel van den Berg, Joris Vermunt

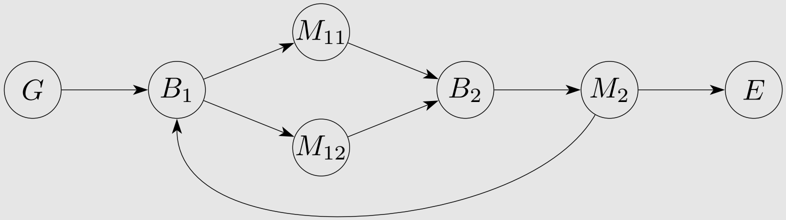
11 December 2003

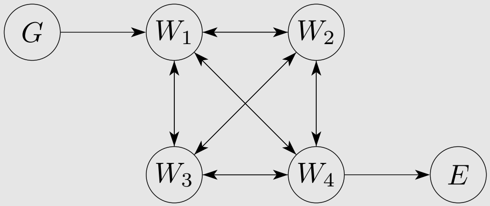
/department of mechanical engineering

Outline

- Control framework
- Modeling problem
- Available models
- PDE models
- Validation studies
  - ramp up/down line
  - oscilating influx to re-entrant line
- Additional properties for PDE-models
- Control
- Conclusions

Manufacturing system





Manufacturing system

Issues

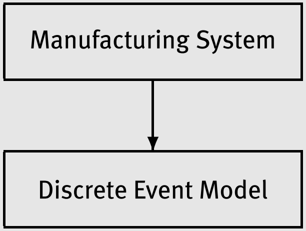
- setup
- finite buffers
- machine failure
- machine maintenance (software upgrade)
- operators (talking, breaks)
- ...

WIP, throughput, cycle time

Control Framework

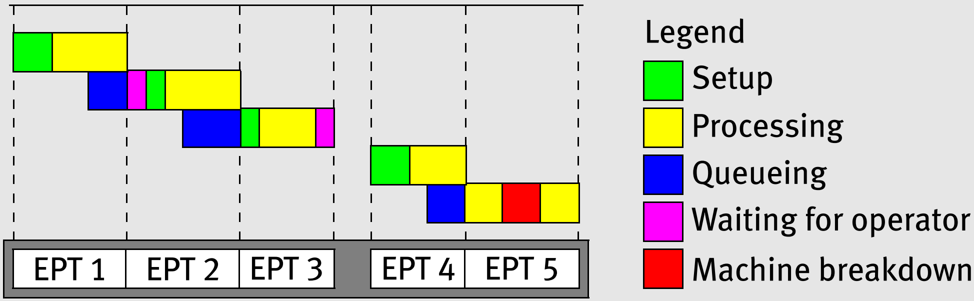
Manufacturing System

Control Framework

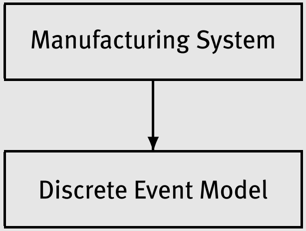


Effective Processing Time

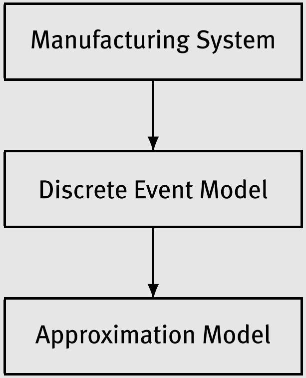
Time a lot experiences (from a logistic point of view)



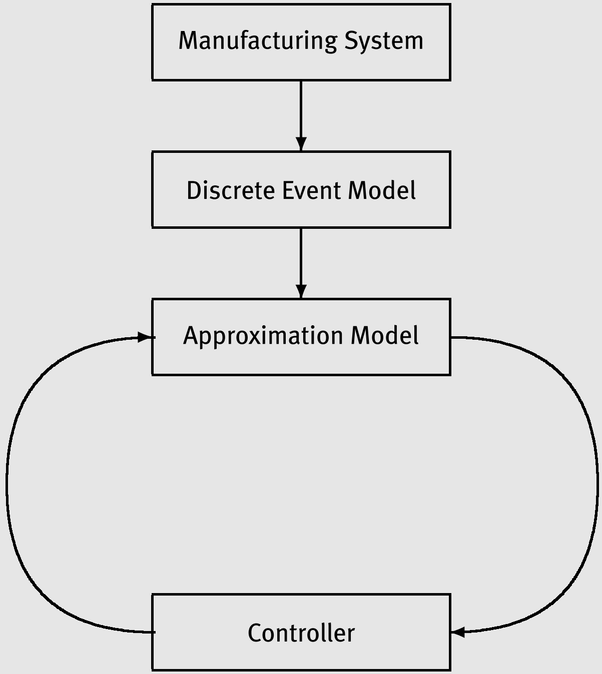
Control Framework



Control Framework



Control Framework



Example: MPC

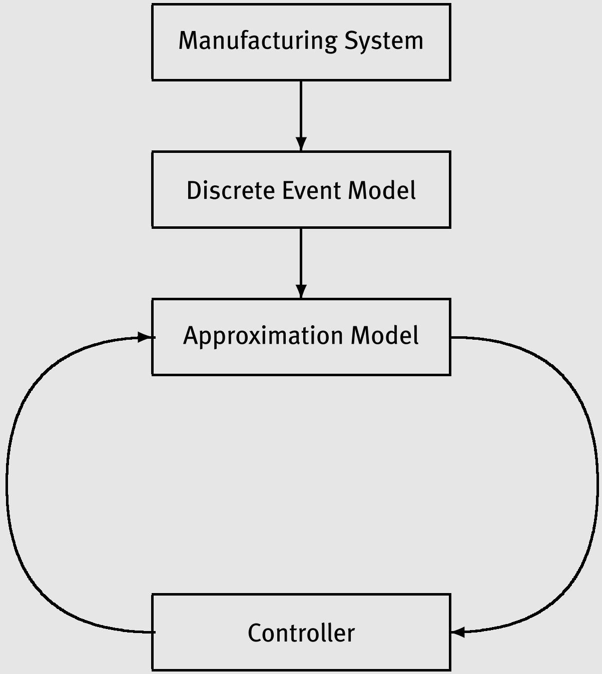
Model-based Predictive Control

- Discrete time model

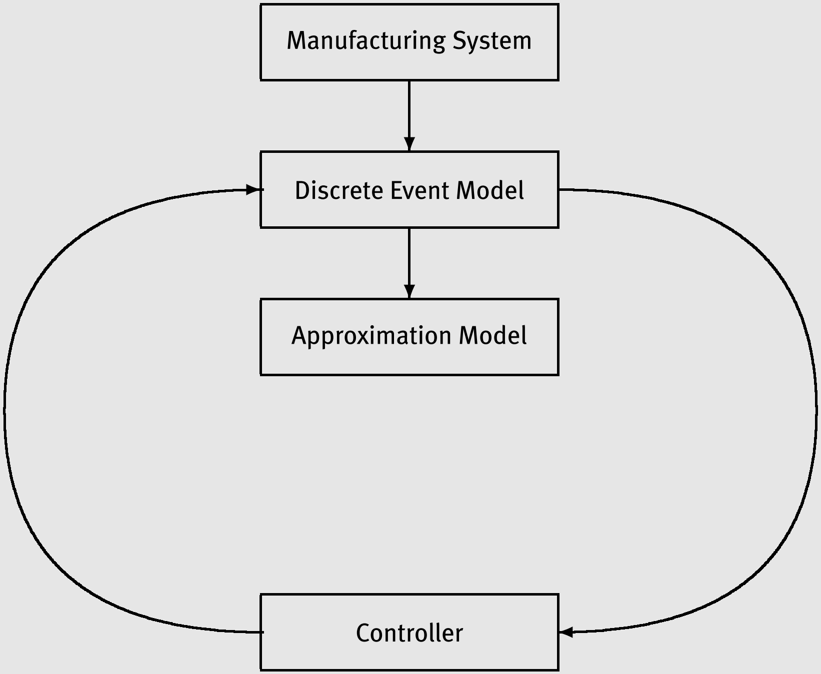
$$\begin{aligned} x(k+1) &= f(x(k), u(k)) \\ y(k) &= h(x(k), u(k)) \end{aligned}$$

- Costs $\min_{u(\cdot)} J(y(k), u(k), k)$
- Prediction horizon (p)
- Control horizon (c, $c \geq p$)
- Yields $u(k), u(k+1), \ldots, u(k+p-1)$. Apply $u(k)$.
- At $k+1$: redo

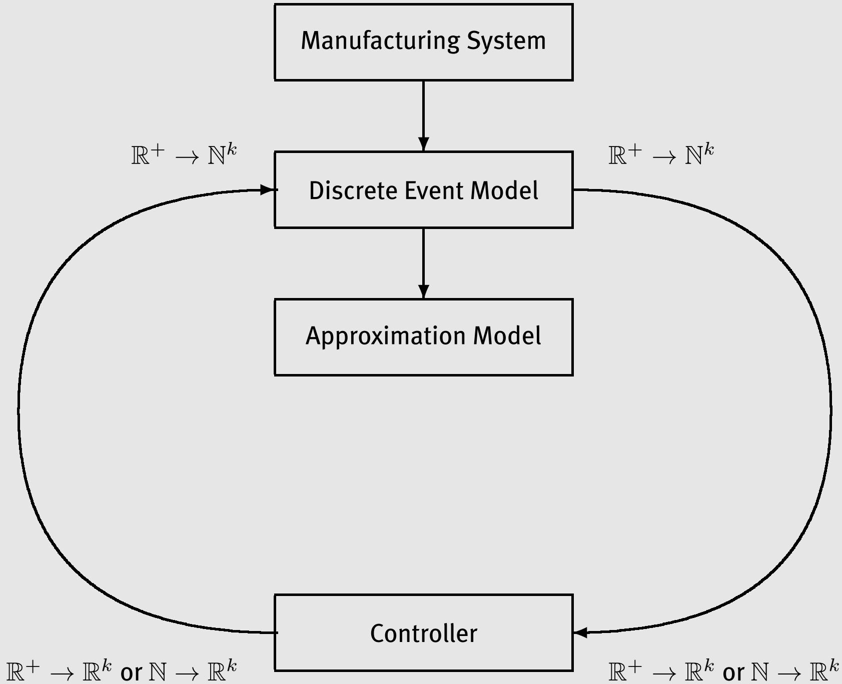
Control Framework



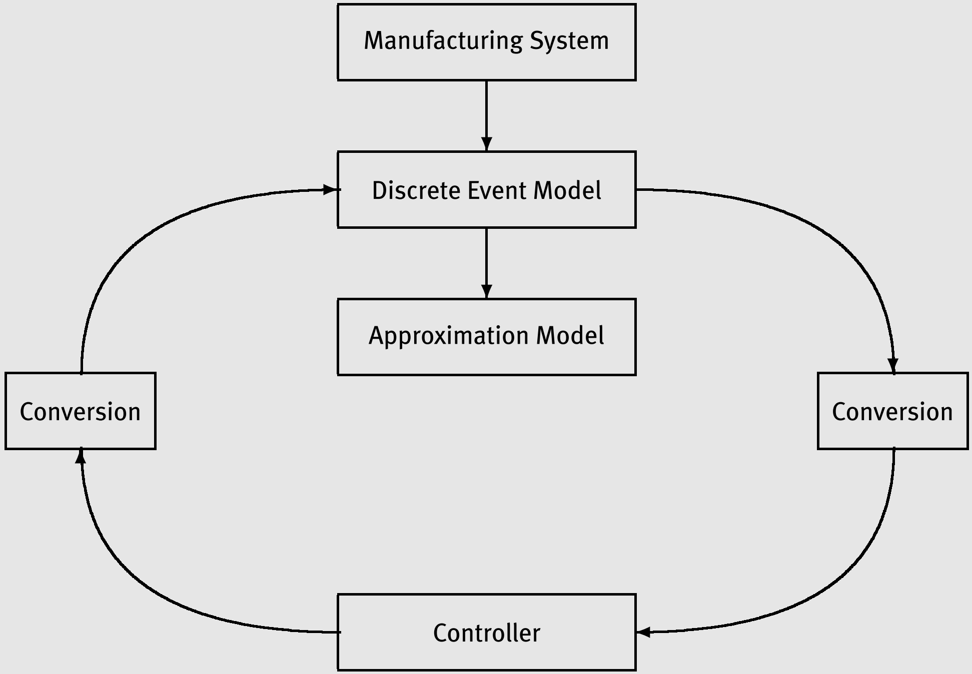
Control Framework



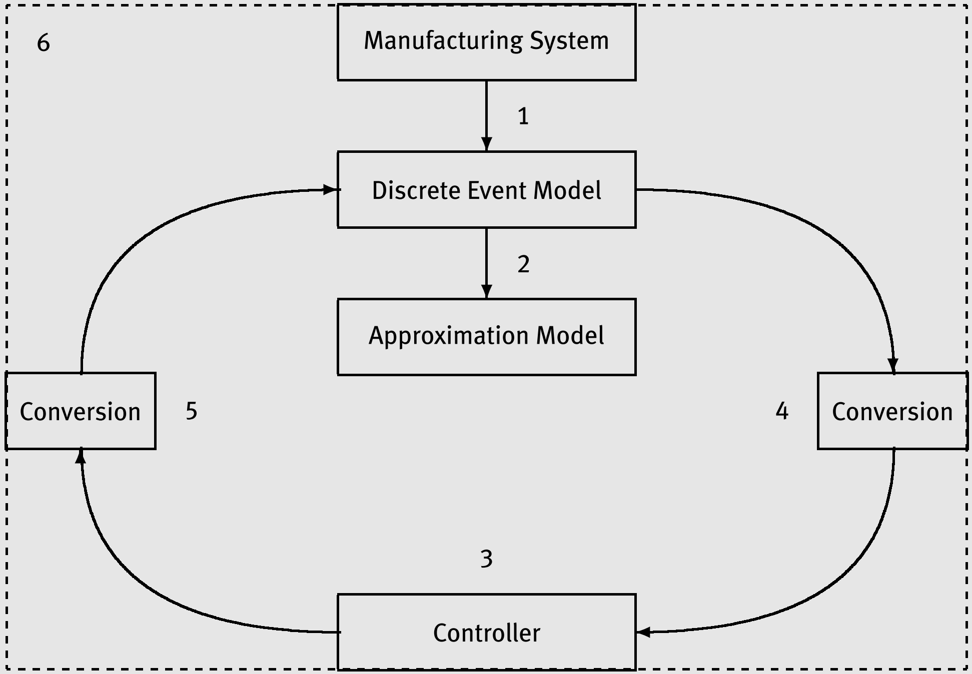
Control Framework



Control Framework



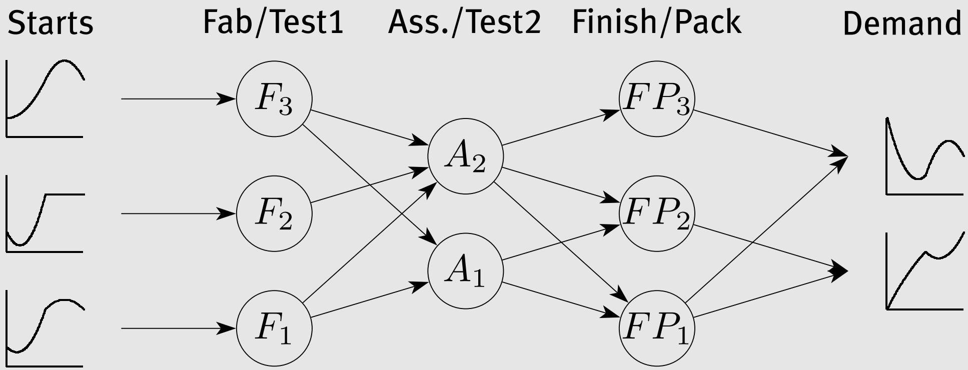
Control Framework



Outline

- Control framework
- Modeling problem
- Available models
- PDE models
- Validation studies
  - ramp up/down line
  - oscilating influx to re-entrant line
- Additional properties for PDE-models
- Control
- Conclusions

Modeling problem



Modeling for control (supply chain/mass production).

- Like to understand dynamics of factories
- Throughput, cycle time, variance of cycle time
- Answer questions like: How to perform ramp up?

Modeling problem

Some observations from practice:

- Quick answers (“What if . . . ”).
- A factory is (almost) never in steady state
- Throughput and cycle time are related

We look for an approximation model that

- is computationally feasible,
- describes dynamics, and
- incorporates both throughput and cycle time

Available models

Discrete Event

- Advantages
  - Include dynamics
  - Throughput and cycle time related
- Disadvantage
  - Clearly infeasible for entire supply chain

Available models

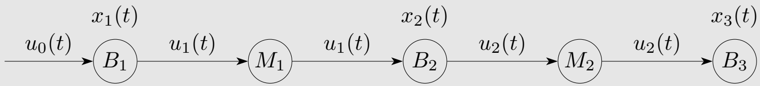
Queueing Theory

- Advantages
  - Throughput and cycle time related
  - Computationally feasible (approximations)
- Disadvantage
  - Only steady state, no dynamics

Available models

Fluid models

- Kimemia and Gershwin: Flow model
- Queuing theorists: Fluid models/Fluid queues



$$\begin{aligned} \dot{x}_1 &= u_0 - u_1 \\ \dot{x}_2 &= u_1 - u_2 \\ \dot{x}_3 &= u_2 \end{aligned} \quad \text{or} \quad \begin{aligned} x_1(k+1) &= x_1(k) + u_0(k) - u_1(k) \\ x_2(k+1) &= x_2(k) + u_1(k) - u_2(k) \\ x_3(k+1) &= x_3(k) + u_2(k) \end{aligned}$$

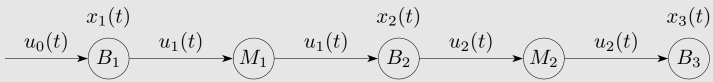
- Cassandras: Stochastic Fluid Model

Available models

Fluid models

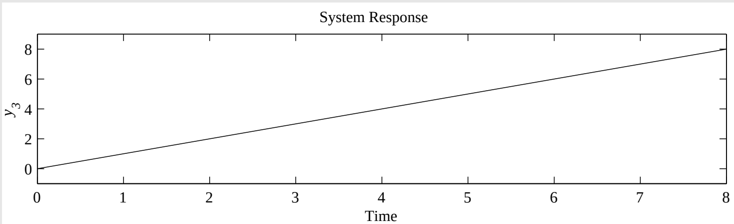
- Advantages
  - Dynamical model
  - Computationally feasible
- Disadvantage
  - Only throughput incorporated in model, no cycle time
  - And more ...

Ramp up of fluid model



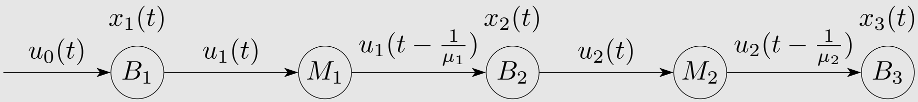
- Initially empty fab, $u_0 = 1$, $\mu_1 = \mu_2 = 1$.
- Machine produces whenever possible:

$$u_i = \begin{cases} 1 & \text{if } y_i > 0 \\ 0 & \text{otherwise} \end{cases} \qquad i \in \{1, 2\}.$$



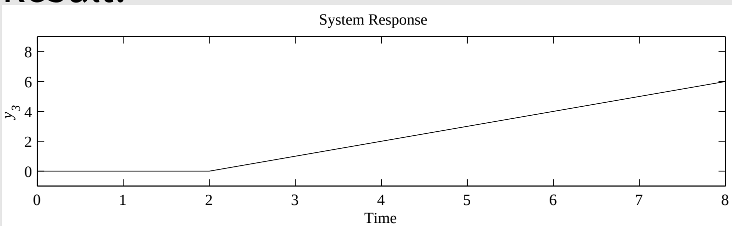
Extension to fluid model (I)

Possible solution: Add delay



$$\dot{x}_1(t) = u_0(t) - u_1(t)$$
$$\dot{x}_2(t) = u_1(t - \frac{1}{\mu_1}) - u_2(t)$$
$$\dot{x}_3(t) = u_2(t - \frac{1}{\mu_2})$$

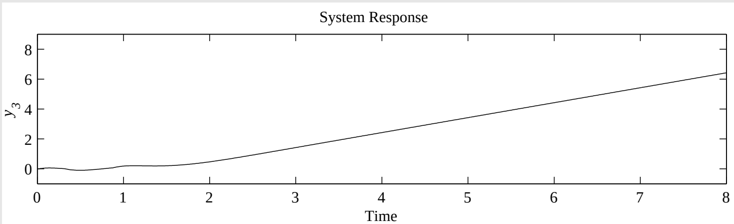
Result:



Extension to fluid model (II)

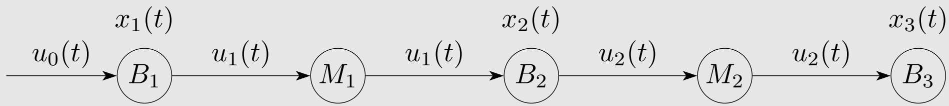
Padé approximations (2nd order):

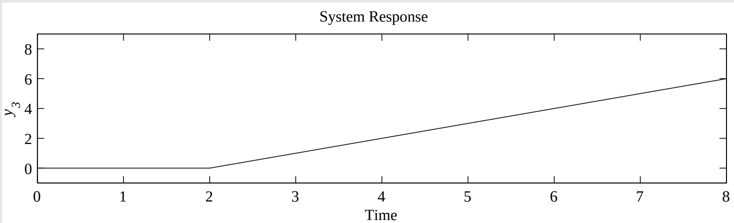
$$\dot{x} = Ax + Bu \qquad x(k+1) = \Phi x(k) + \Gamma u(k)$$



Extension to fluid model (III)

Hybrid model:





Available models (conclusion)

- Discrete Event: Not computationally feasible
  Queuing Theory: No dynamics
  Fluid models: No cycle time
- Need something else!
- Discrete event models (and queuing theory) have proved themselves. Can be used for verification!

Outline

- Control framework
- Modeling problem
- Available models
- PDE models
- Validation studies
  - ramp up/down line
  - oscilating influx to re-entrant line
- Additional properties for PDE-models
- Control
- Conclusions

Traffic flow: LWR model

Lighthill, Whitham ('55), and Richards ('56)

Traffic behavior on one-way road:

- density $\rho(x,t)$,
- speed $v(x,t)$,
- flow $u(x,t) = \rho(x,t)v(x,t)$.

Conservation of mass:

$$\frac{\partial \rho}{\partial t}(x,t) + \frac{\partial u}{\partial x}(x,t) = 0.$$

Static relation between flow and density:

$$u(x,t) = S(\rho(x,t)).$$

Modeling manufacturing flow

- density $\rho(x,t)$,
- speed $v(x,t)$,
- flow $u(x,t) = \rho(x,t)v(x,t)$,
- Conservation of mass: $\frac{\partial\rho}{\partial t}(x,t) + \frac{\partial\rho v}{\partial x}(x,t) = 0.$
- Boundary condition: $u(0,t) = \lambda(t)$

Modeling manufacturing flow

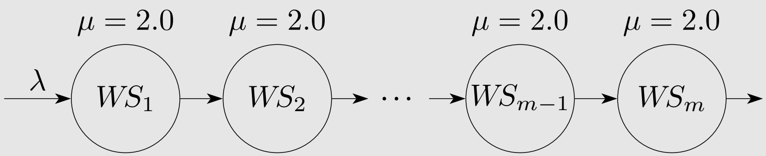
Armbruster, Marthaler, Ringhofer (2002):

- Single queue: $\frac{1}{v(x,t)} = \frac{1}{\mu}(1 + \int_0^1 \rho(s,t)\,\mathrm{d}s)$
- Single queue: $\frac{\partial \rho v}{\partial t}(x,t) + \frac{\partial \rho v^2}{\partial x}(x,t) = 0$
  $\rho v^2(0,t) = \frac{\mu \cdot \rho v(0,t)}{1+\int_0^1 \rho(s,t)\,\mathrm{d}s}$
- Re-entrant: $v(x,t) = v_0 \left(1 - \frac{\int_0^1 \rho(s,t)\,\mathrm{d}s}{W_{\text{max}}}\right)$

Lefeber (2003):

- Line of many identical queues: $v(x,t) = \frac{\mu}{1+\rho(x,t)}$

Validation studies: Study I



- Identical workstations, infinite buffers (FIFO)
- Number of workstations: $m = 10$, $m = 50$
- Processing times: exponential (mean 0.5)
- Inter arrival times: exponential (mean $1/\lambda$)
- From one steady state to the other
  - ramp up: from initially empty to 25%, 50%, 75%, 95% utilization
  - ramp down: from 50%, 75%, 90%, 95% utilization to 25% utilization

Performance measures

- mean WIP (in steady state): w_{ss}
- mean throughput (in steady state): δ_{ss}
- mean cycle time (in steady state): φ_{ss}
- time for reaching 99% of steady state WIP
- time for reaching 99% of steady state throughput
- time for reaching 99% of steady state cycle time
- cycle time for first lot inserted at $t = 0$

- Batches of 100 experiments
- Repeat until in each buffer 95% two sided confidence interval smaller than 2% of mean

Results (ramp up)

	m=10	m=50	m=10	m=50	m=10	m=50
φ_{ss}	$++$	$++$	$++$	$++$	$++$	$++$
time to φ_{ss}	0	0	$+$	$--$	$-$	0
φ 1st lot	$-$	0	$+$	0	$-$	0
δ_{ss}	$++$	$++$	$++$	$++$	$++$	$++$
time to δ_{ss}	0	0	0	$-$	0	0
w_{ss}	$++$	$++$	$++$	$++$	$++$	$++$
time to w_{ss}	0	0	0	$--$	$-$	0

$++$	$<$5%
$+$	5% – 10%
0	10% – 50%
$-$	50% – 100%
$--$	$>$100%

Results (ramp down)

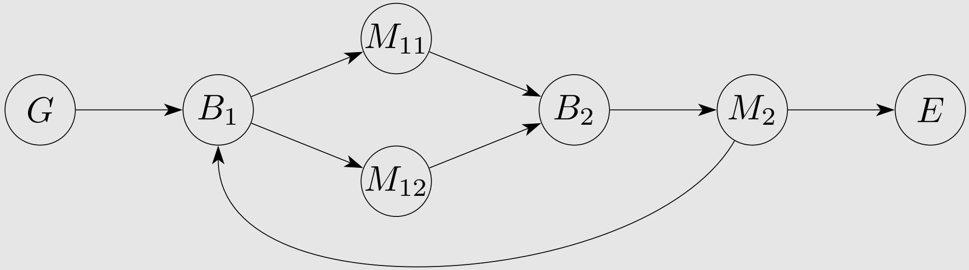
	m=10	m=50	m=10	m=50	m=10	m=50
φ_{ss}	$+$	$+$	$+$	$+$	$+$	$+$
time to φ_{ss}	0	0	0	$-$	0	0
φ 1st lot	0	0	$+$	$++$	$+$	$++$
δ_{ss}	$++$	$++$	$++$	$++$	$++$	$++$
time to δ_{ss}	0	0	$+$	$-$	0	0
w_{ss}	$++$	$++$	$++$	$++$	$++$	$++$
time to w_{ss}	0	0	0	0	0	0

$++$	$<$5%
$+$	5% – 10%
0	10% – 50%
$-$	50% – 100%
$--$	$>$100%

General observations (Study I)

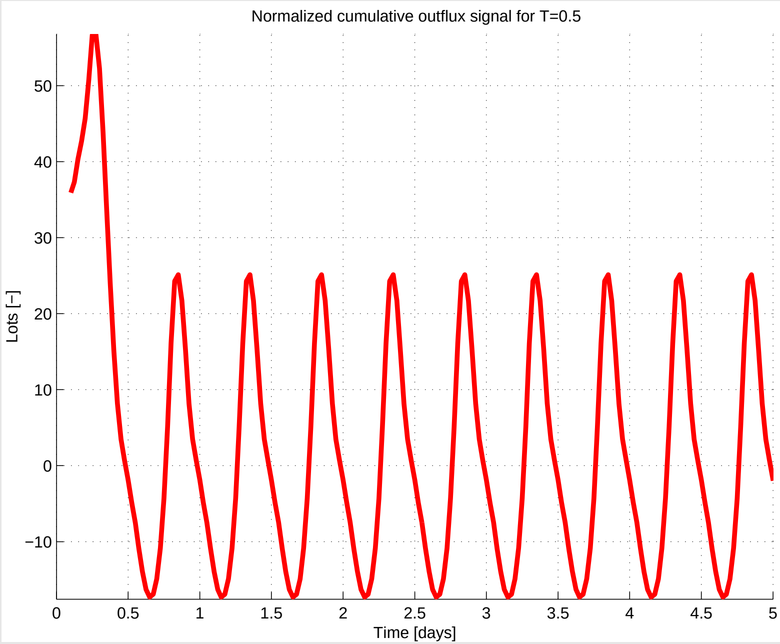
- Steady state performance well described
- Time to reach steady state ill described
- Amount of lots produced before reaching steady state (most cases) relatively small
- Homogeneous velocity results in ill described behavior of throughput
- Simulation run Discrete Event: 4 minutes
  Batch run Discrete Event: 7 hours
  Simulation run PDE: 1 minute

Validation studies: Study II

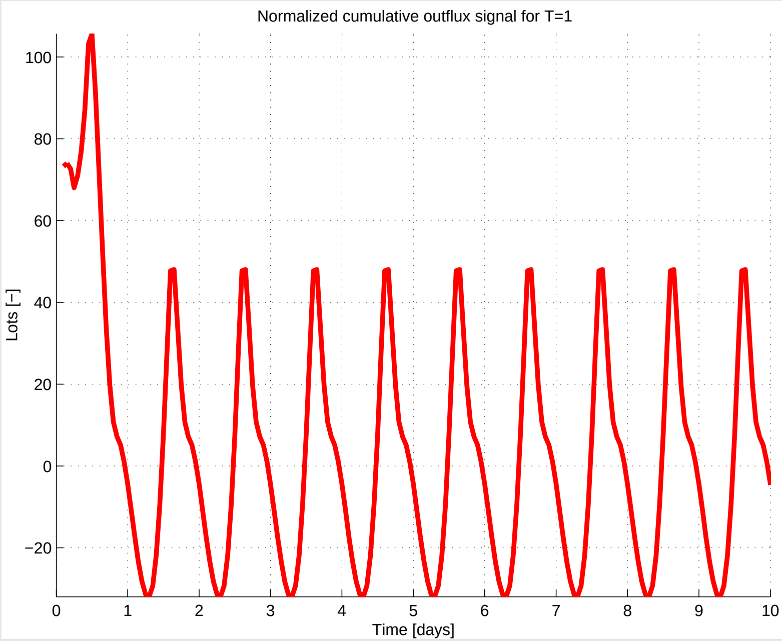


- Rentrant: 4 times
- 5 WS: 22 identical machines (WS 3: 21)
- Deterministic processing times
- Oscillating inflow, different frequencies
- Buffer policies: FIFO, push, pull

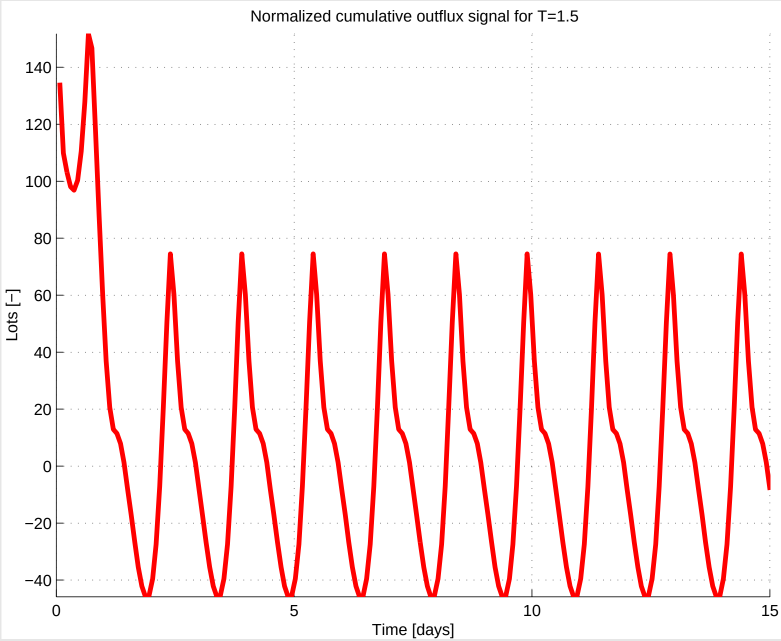
Outs: FIFO, period 0.5 day



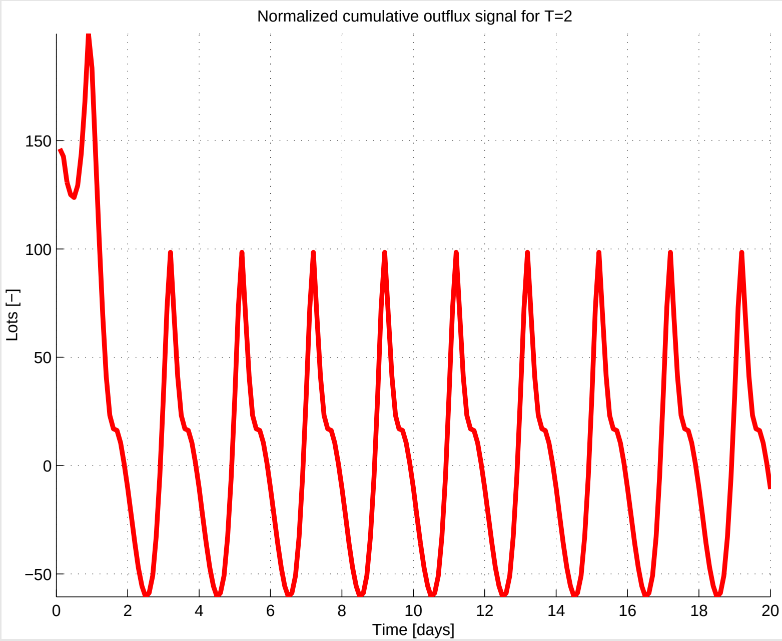
Outs: FIFO, period 1.0 day



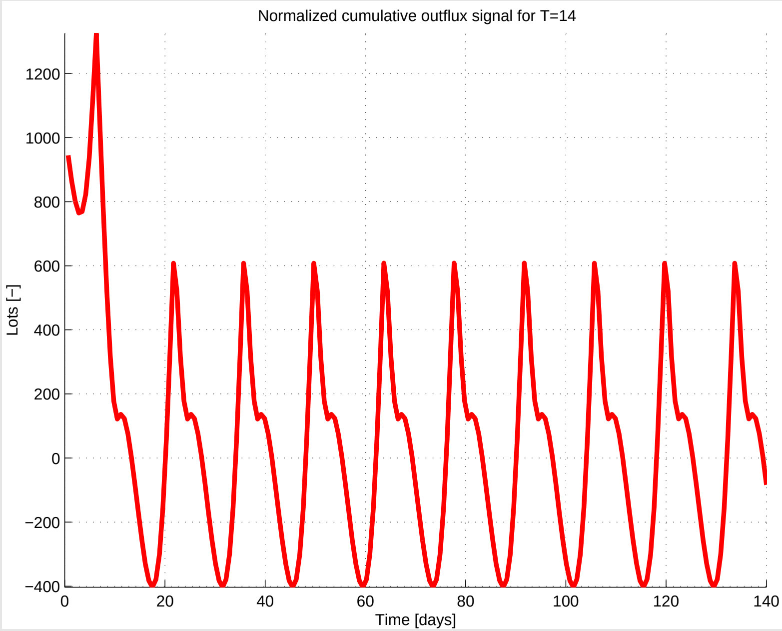
Outs: FIFO, period 1.5 day



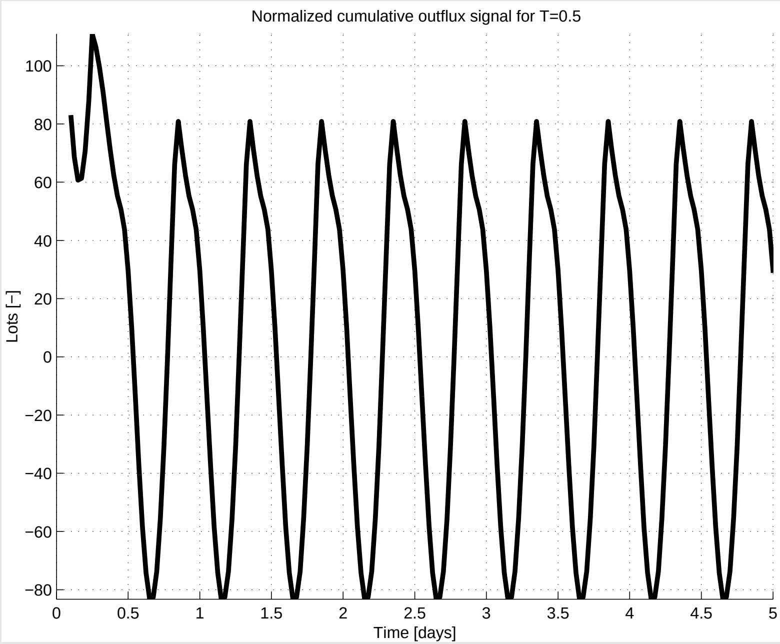
Outs: FIFO, period 2.0 day



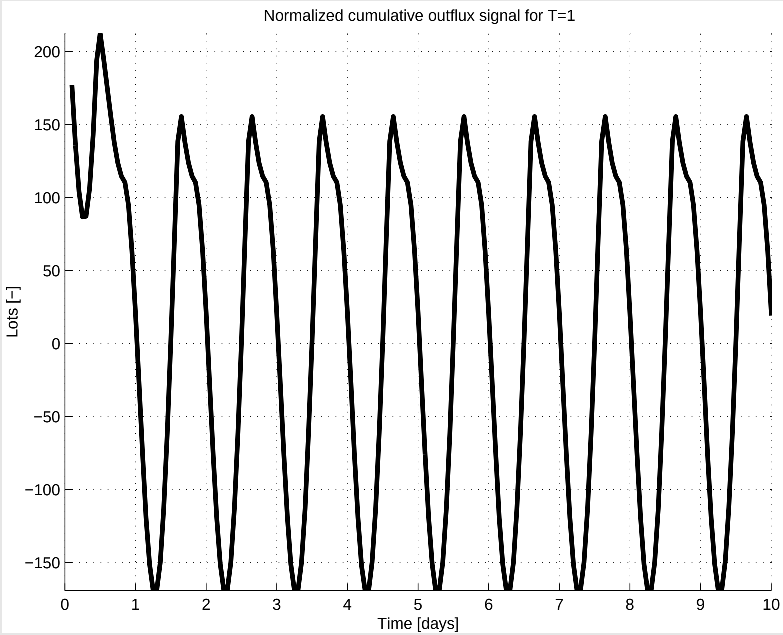
Outs: FIFO, period 14.0 day



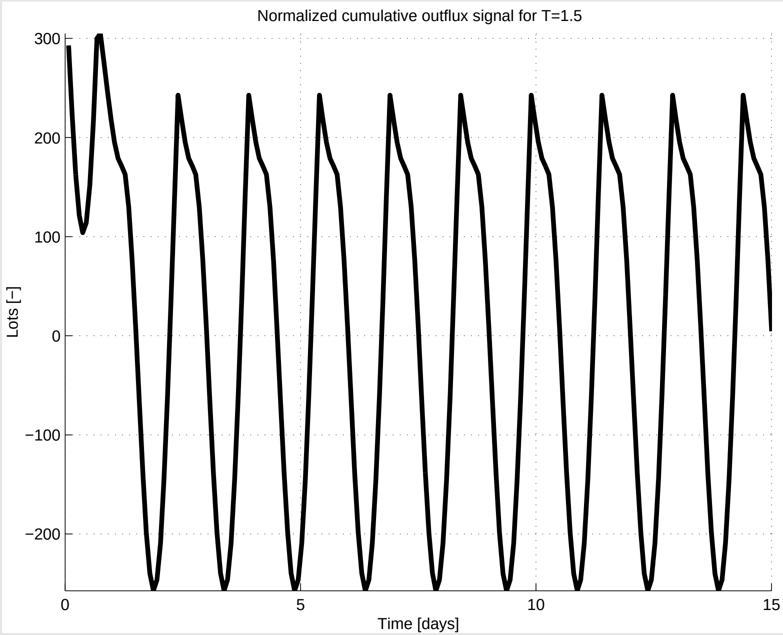
Outs: push, period 0.5 day



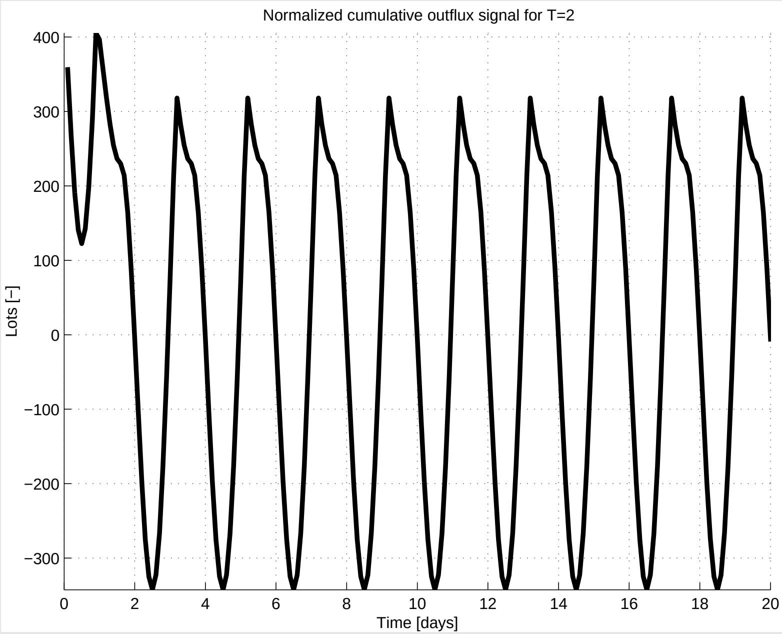
Outs: push, period 1.0 day



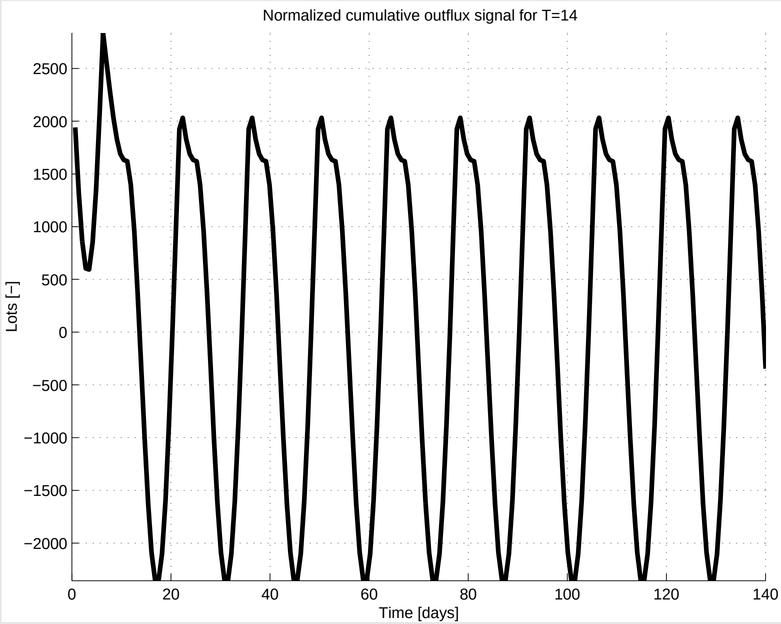
Outs: push, period 1.5 day



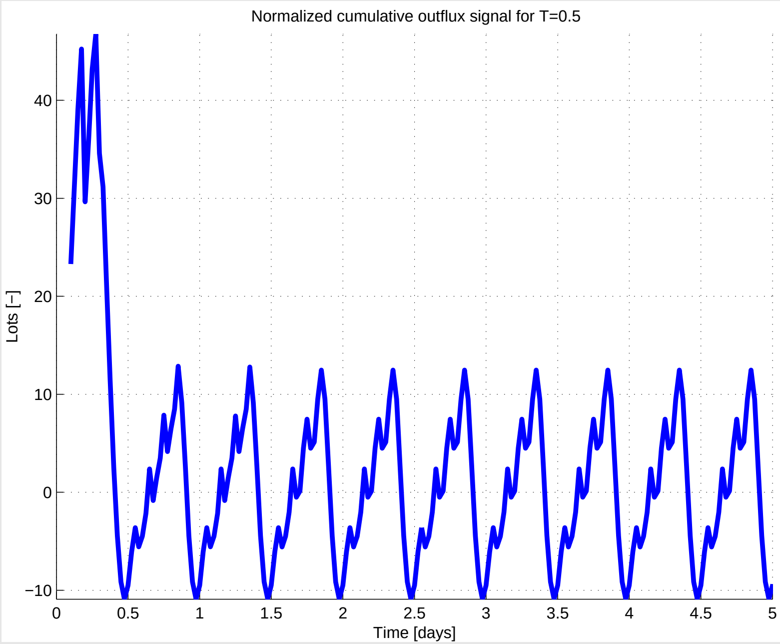
Outs: push, period 2.0 day



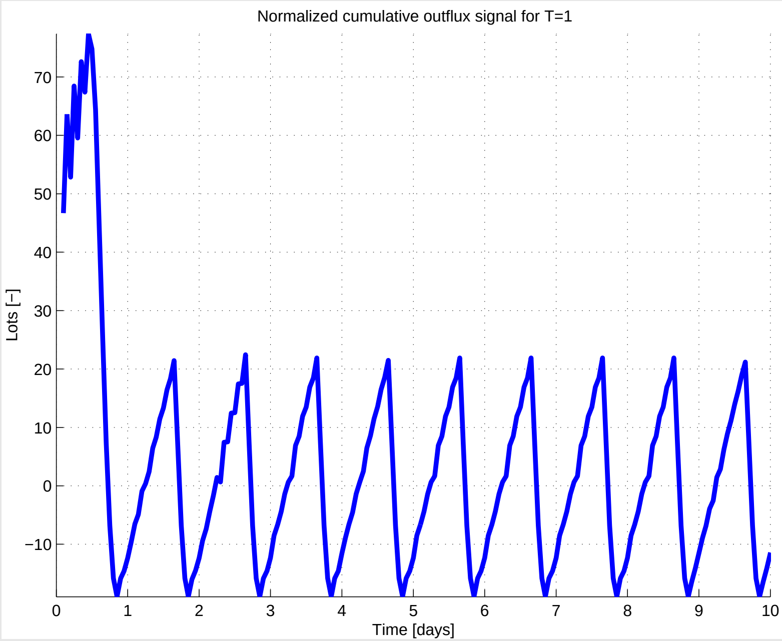
Outs: push, period 14.0 day



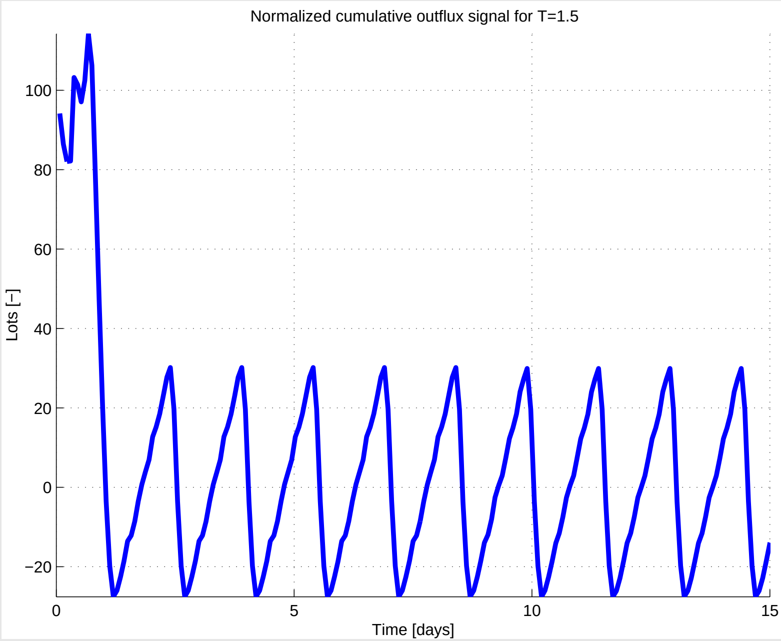
Outs: pull, period 0.5 day



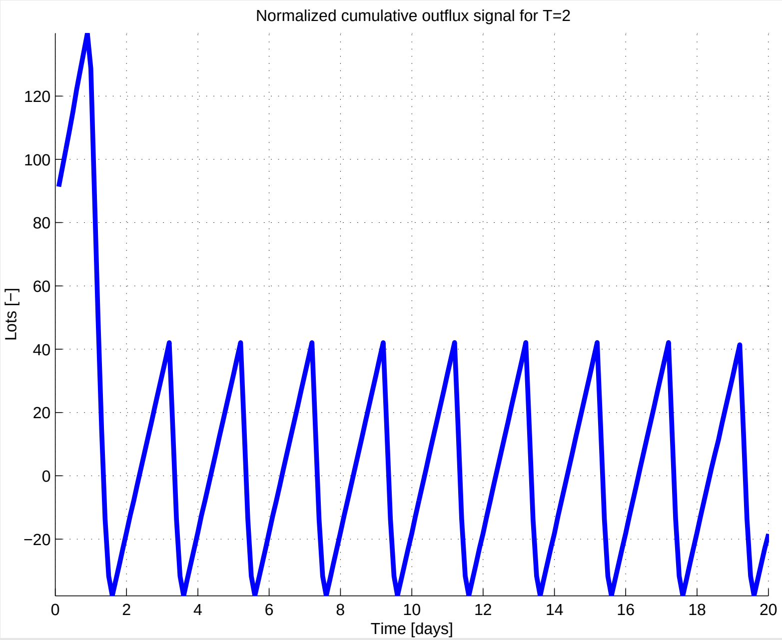
Outs: pull, period 1.0 day



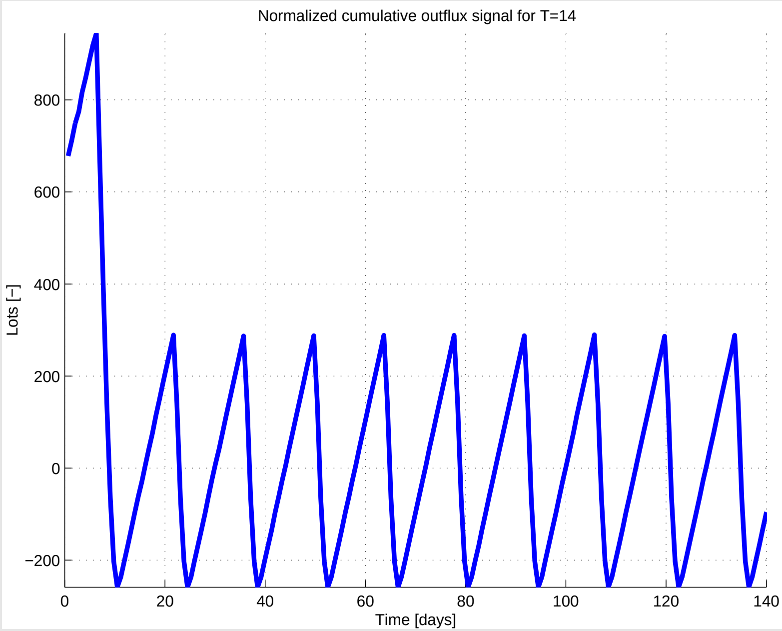
Outs: pull, period 1.5 day



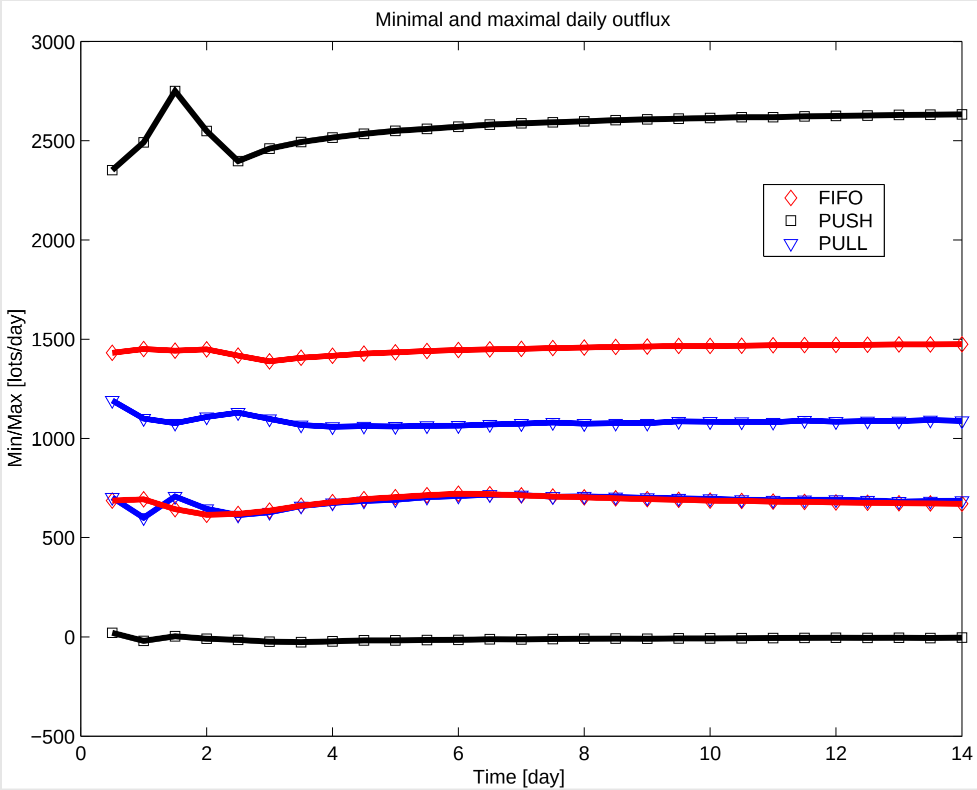
Outs: pull, period 2.0 day



Outs: pull, period 14.0 day



Results (outflux)



Validation study II: conclusions

- Outflux is oscillating (with frequency of influx)
- Almost no resonance effects
- Buffer policy *does* matter

Conclusion of validation studies

Search for valid PDE models continues...

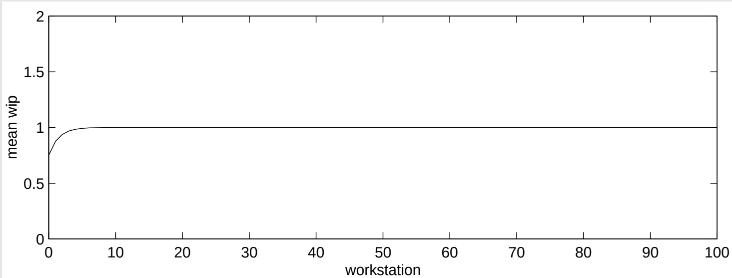
Properties

- No backward-flow allowed
- No negative density
- Stable steady states
  - constant feed rate $\longrightarrow$ equilibrium
  - equilibrium meets relations queuing theory

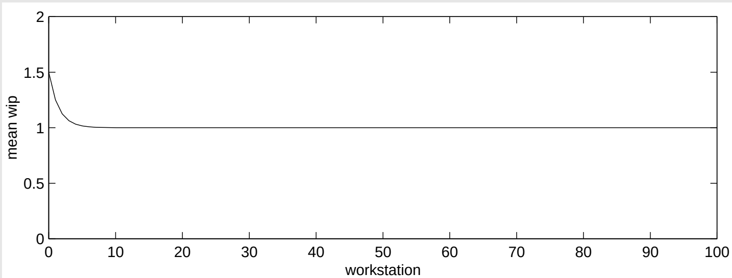
Properties

100 machines, $\mu = 1$, exponential. Utilization: 50%.

- Regular arrivals: $c_a^2 = 0$



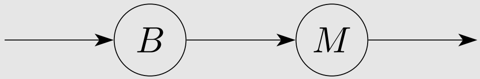
- Irregular arrivals: $c_a^2 = 3$



Properties

Variability needs to be included. However, ...

1 machine, $\mu = 1$, exponential



- Push control: exponential arrivals. Utilization 50%
  - Throughput: 0.5 lots per unit time
  - Cycle time: 2 hours
  - Mean wip: 1 lot
- CONWIP control: WIP=1
  - Throughput: 1 lots per unit time
  - Cycle time: 1 hours
  - Mean wip: 1 lot

Outline

- Control framework
- Modeling problem
- Available models
- PDE models
- Validation studies
  - ramp up/down line
  - oscilating influx to re-entrant line
- Additional properties for PDE-models
- Control
- Conclusions

Control: example

- Conservation of mass: $\frac{\partial \rho}{\partial t}(x,t) + \frac{\partial \rho v}{\partial x}(x,t) = 0.$
- Line of m identical queues: $v(x,t) = \frac{\mu/m}{1+\rho(x,t)/m}$
- Initial condition: $\rho(x,0) = \rho_0(x)$

- Input: $u(0,t) = \lambda_{\mathsf{in}}(t)$
- Outputs: $\lambda_{\mathsf{out}}(t) = u(1,t)$, $w(t) = \int_0^1 \rho(x,t)\,\mathrm{d}t$

How to reach desired steady state?

Lyapunov based controller design

Control: $\lambda_{\text{in}}(t) = f(\lambda_{\text{out}}(t), w(t))$

As quickly as possible:

Control: $\lambda_{\text{in}}(t) = \lambda_{\text{des}}$

MPC based controller design

Approximation model (nonlinear)

$$x_1(k+1) = x_1(k) - \frac{\mu x_1(k)}{m + x_1(k)} + \lambda_{\text{in}}(k)$$

$$x_2(k+1) = x_2(k) - \frac{\mu x_2(k)}{m + x_2(k)} + \frac{\mu x_1(k)}{m + x_1(k)}$$

$$\vdots$$

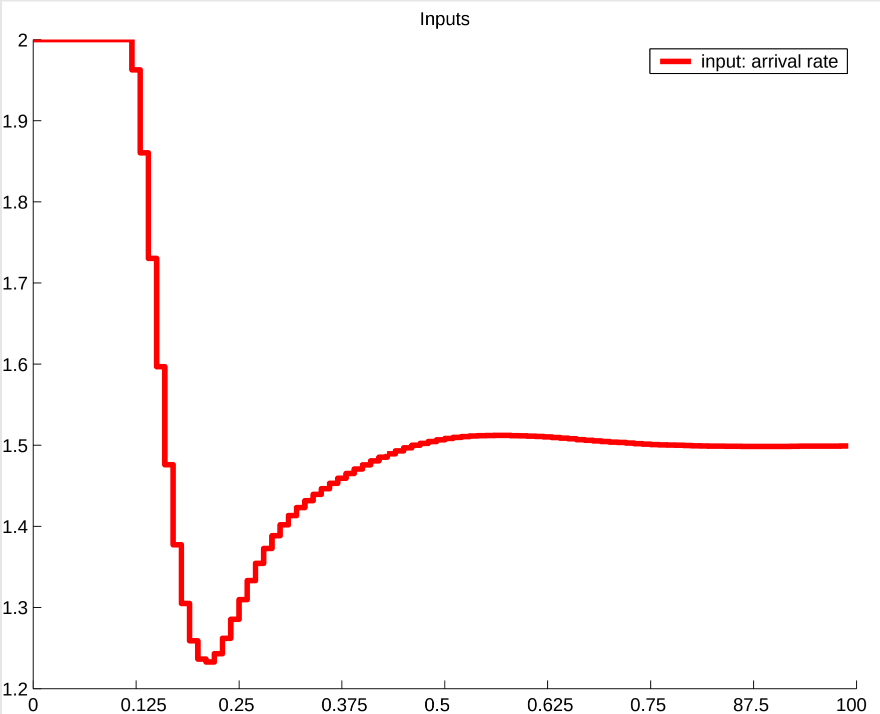
$$x_m(k+1) = x_m(k) - \frac{\mu x_m(k)}{m + x_m(k)} + \frac{\mu x_{m-1}(k)}{m + x_{m-1}(k)}$$

$$y(k) = \frac{\mu x_m(k)}{m + x_m(k)}$$

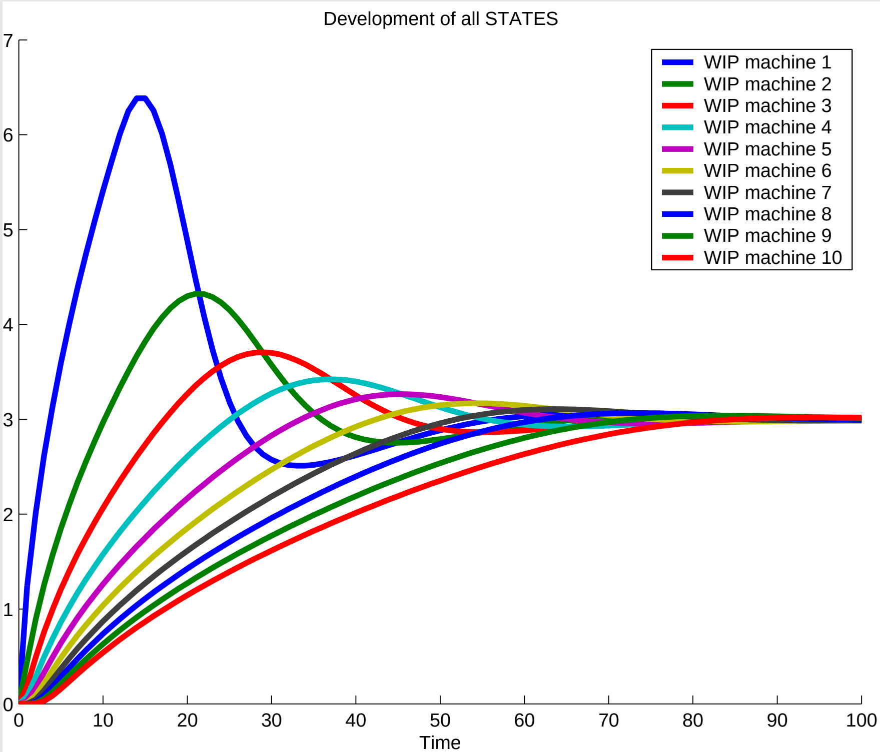
MPC based controller design

- Number of machines $m = 10$
- Mean processing time: 0.5h
- Desired $u = 0.75$ (1.5 lot per h)
- Initial WIP $x_i(0) = 0$
- Prediction horizon $p = 100$h
- Control horizon $p = 5$h
- Control constant over periods of 1h
- Time sampling: 40 steps per 1h

MPC based controller design



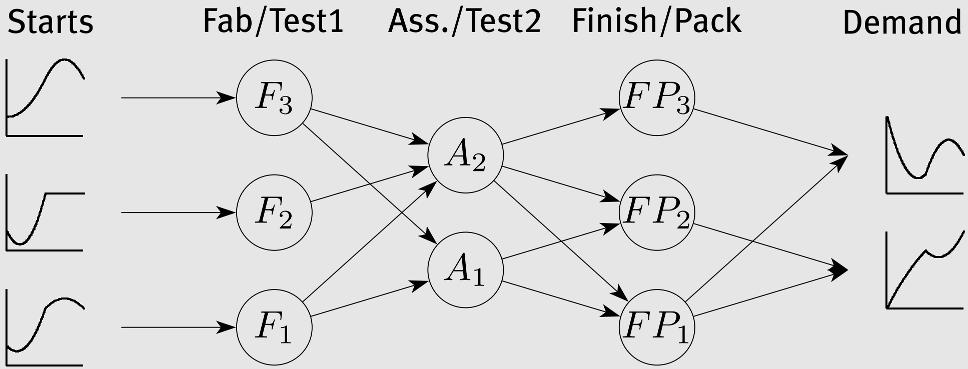
MPC based controller design



Promising developments

A.J. van der Schaft, B. Maschke (2003):

Hamiltonian framework (boundary control of PDE's)



Conclusions

- Control framework (EPT)
- Modeling
  - NOT: Discrete event, Queuing theory, Fluid models
  - Possible: PDE-models
    - Correct steady state behavior
    - Better description transient needed
    - Resonance needs better study
    - Second moment and correlation needs to be included
    - Queueing theory, discrete event models can be used for validation of PDE models
- Next step: PDE-based controller design