Multi-Agent
Model Predictive Control
of Irrigation Canals

Rudy Negenborn

Peter-Jules van Overloop

Tamás Keviczky, Bart De Schutter

Irrigation Channels and Related Problems – Oct. 2–4, 2008

Delft Center for Systems and Control

TU Delft

Delft University of Technology

Overview

1. Open water systems
2. Multi-agent model predictive control
3. Application to an irrigation canal
4. Concluding remarks

1. Open water systems

- Water has many uses (drinking, agriculture, transportation, recreation, energy production, . . .)
- Wide variety in roles of water
  $\rightarrow$ many organizations manage human interaction with water

- Large rivers cover various countries
- Regional canals are operated by various water boards
- For sewer systems there is interaction among municipality and neighboring canals
- Large multi-purpose reservoirs connected do usually not take one another into account
- For irrigation canals there is interaction among irrigation districts, users, and water sources

1. Open water systems

Automatic control

- Existing agreements updated once every 2–3 years
- Dynamic behavior of water varies at much faster timescales
  → Coordination at much higher frequencies (daily, hourly, . . .) can be beneficial

1. Open water systems

Automatic control

- Existing agreements updated once every 2–3 years
- Dynamic behavior of water varies at much faster timescales
  → Coordination at much higher frequencies (daily, hourly, . . .) can be beneficial
- Centralized control not feasible
  - due to complexity and size of water systems
  - due to the existing organization structures

1. Open water systems

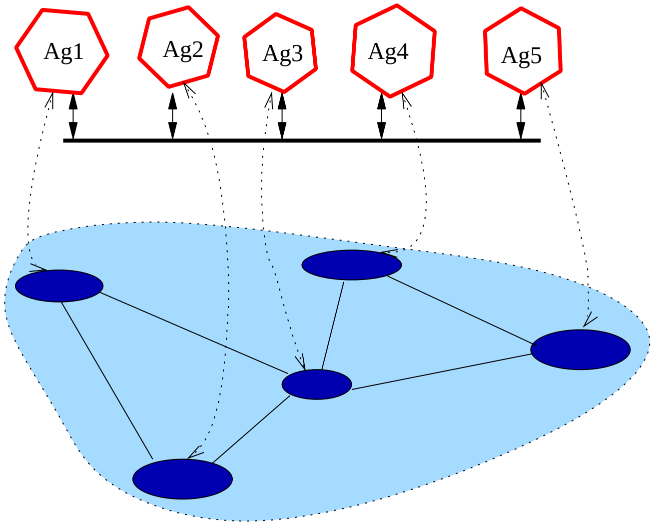
Automatic control

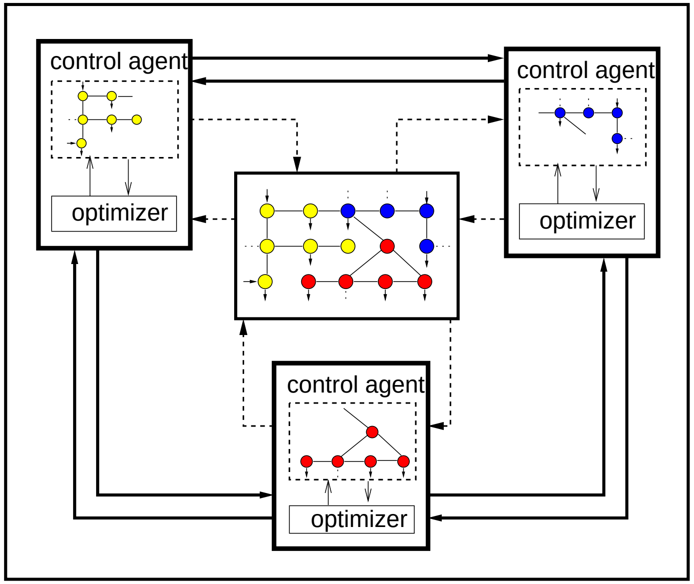
- Existing agreements updated once every 2–3 years
- Dynamic behavior of water varies at much faster timescales
  $\rightarrow$ Coordination at much higher frequencies (daily, hourly, . . .) can be beneficial
- Centralized control not feasible
  due to complexity and size of water systems
  due to the existing organization structures
- Potential benefits of distributed control for
  - Rivers: Better use of inundation areas by linking flood protection schemes
  - Canals: Flexible maximum discharges, such that collapsing of embankments in one area can be prevented if storage capacity in another is available
  - Sewer systems: Flexible volume of water allowed to be spilled
  - Reservoirs: Improved use of available storage in case of extreme precipitation
  - Irrigation canals: Better spreading of available water towards areas under stress

2. Multi-agent MPC

Multi-agent control: more than 1 decision maker

- subnetworks instead of overall network
- single agent for each subnetwork
  - limited action capabilities
  - limited information gathering





Challenge
agents should choose local inputs that are globally optimal using communication, coordination, cooperation, and negotiation

2. Multi-agent MPC

Local control problem of agent i at decision step k

$$\min_{\tilde{\mathbf{u}}_i(k),\tilde{\mathbf{x}}_i(k+1)} J_{\text{local},i}(\tilde{\mathbf{u}}_i(k), \tilde{\mathbf{x}}_i(k+1))$$

subject to

- subnetwork dynamics

$$\mathbf{x}_i(k+1) = \mathbf{f}_i(\mathbf{x}_i(k), \mathbf{u}_i(k), \mathbf{d}_i(k), \ldots)$$

$$\vdots$$

$$\mathbf{x}_i(k+N) = \mathbf{f}_i(\mathbf{x}_i(k+N-1), \mathbf{u}_i(k+N-1), \mathbf{d}_i(k+N-1), \ldots)$$

- initial local state, disturbances, and additional constraints

2. Multi-agent MPC

Local control problem of agent i at decision step k

$$\min_{\tilde{\mathbf{u}}_i(k),\tilde{\mathbf{x}}_i(k+1)} J_{\text{local},i}(\tilde{\mathbf{u}}_i(k), \tilde{\mathbf{x}}_i(k+1))$$

subject to

- subnetwork dynamics

$$\mathbf{x}_i(k+1) = \mathbf{f}_i(\mathbf{x}_i(k), \mathbf{u}_i(k), \mathbf{d}_i(k), \mathbf{w}_{\text{in},j_1 i}(k), \ldots, \mathbf{w}_{\text{in},j_{m_i} i}(k))$$

$$\mathbf{w}_{\text{out},ji}(k+1) = \mathbf{h}_{\text{out},ji}(\mathbf{u}_i(k), \mathbf{d}_i(k), \mathbf{x}_i(k+1)) \quad \text{for all neighbors } j$$

$$\vdots$$

$$\mathbf{x}_i(k+N) = \mathbf{f}_i(\mathbf{x}_i(k+N-1), \mathbf{u}_i(k+N-1), \mathbf{d}_i(k+N-1), \mathbf{w}_{\text{in},j_1 i}(k+N-1), \ldots, \mathbf{w}_{\text{in},j_{m_i} i}(k+N-1))$$

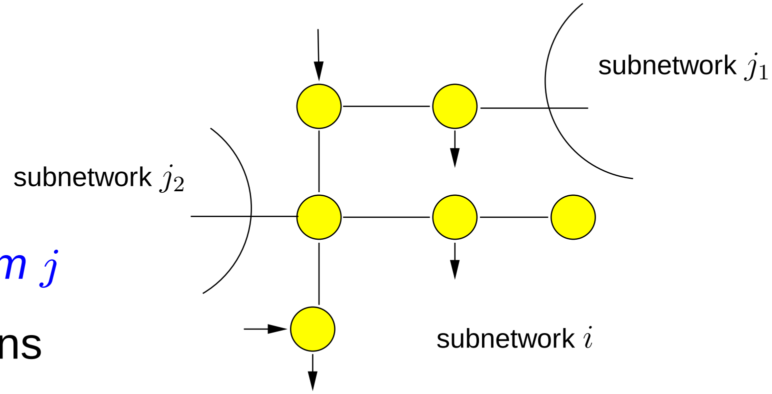
$$\mathbf{w}_{\text{out},ji}(k+N) = \mathbf{h}_{\text{out},ji}(\mathbf{u}_i(k+N-1), \mathbf{d}_i(k+N-1), \mathbf{x}_i(k+N)) \quad \text{for all neighbors } j$$

- initial local state, disturbances and additional constraints

2. Multi-agent MPC

Interconnecting constraints

- constraints on interconnecting variables
- imposed by dynamics of overall network
- *what goes in into i equals what goes out from j*
- satisfaction necessary for accurate predictions

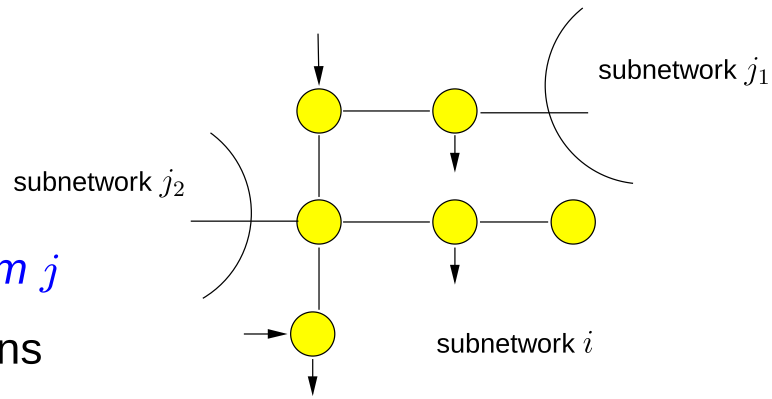


$$
\begin{aligned}
\mathbf{w}_{\text{in},ji}(k) &= \mathbf{w}_{\text{out},ij}(k) \\
\mathbf{w}_{\text{out},ji}(k) &= \mathbf{w}_{\text{in},ij}(k) \\
\vdots \quad & \quad \vdots \\
\mathbf{w}_{\text{in},ji}(k+N-1) &= \mathbf{w}_{\text{out},ij}(k+N-1) \\
\mathbf{w}_{\text{out},ji}(k+N-1) &= \mathbf{w}_{\text{in},ij}(k+N-1)
\end{aligned}
$$

2. Multi-agent MPC

Interconnecting constraints

- constraints on interconnecting variables
- imposed by dynamics of overall network
- *what goes in into i equals what goes out from j*
- satisfaction necessary for accurate predictions



$$\mathbf{w}_{\text{in},ji}(k) = \mathbf{w}_{\text{out},ij}(k)$$
$$\mathbf{w}_{\text{out},ji}(k) = \mathbf{w}_{\text{in},ij}(k)$$
$$\vdots \qquad \vdots$$
$$\mathbf{w}_{\text{in},ji}(k+N-1) = \mathbf{w}_{\text{out},ij}(k+N-1)$$
$$\mathbf{w}_{\text{out},ji}(k+N-1) = \mathbf{w}_{\text{in},ij}(k+N-1)$$

For agent controlling subnetwork i

- $\mathbf{w}_{\text{in},ij}$ and $\mathbf{w}_{\text{out},ij}$ of neighbor j unknown
- how make accurate predictions?
  $\rightarrow$ ignore, assume constant, predict, negotiate, . . .

2. Multi-agent MPC

- Let the dynamics of each subnetwork i be described by

$$\mathbf{x}_i(k+1) = \mathbf{A}_i \mathbf{x}_i(k) + \mathbf{B}_{1,i}\mathbf{u}_i(k) + \mathbf{B}_{2,i}\mathbf{d}_i(k) + \mathbf{B}_{3,i}\mathbf{v}_i(k)$$

- Let the interconnecting inputs and outputs be defined as

$$\mathbf{w}_{\text{in},i}(k) = \mathbf{v}_i(k)$$
$$\mathbf{w}_{\text{out},i}(k) = \mathbf{K}_i[\mathbf{u}_i^{\mathsf{T}}(k)\ \mathbf{d}_i^{\mathsf{T}}(k)\ \mathbf{x}_i^{\mathsf{T}}(k+1)]^{\mathsf{T}},$$

  where $\mathbf{K}_i$ is an interconnecting output selection matrix

- Let the interconnecting constraints between i and j be given by

$$\mathbf{w}_{\text{in},ji}(k) = \mathbf{w}_{\text{out},ij}(k)$$
$$\mathbf{w}_{\text{out},ji}(k) = \mathbf{w}_{\text{in},ij}(k)$$

- Let the local objective function $J_{\text{local},i}$ be quadratic and positive (semi-)definite.

2. Multi-agent MPC

A multiple-iterations serial scheme

- accurate predictions by agreeing on values of interconnecting variables
- each agent
  - computes optimal local *and* interconnecting variables
  - communicates interconnecting variables to neighbors
  - updates parameters $\tilde{\lambda}_{\text{in}}^{ji}$, $\tilde{\lambda}_{\text{out}}^{ji}$ of additional cost term J_{inter}^{i}
- iterations until stopping criterion satisfied
- at termination, scheme has converged to overall optimal solution

$$\min_{\tilde{\mathbf{u}}_i(k),\tilde{\mathbf{x}}_i(k+1),\tilde{\mathbf{w}}_{\text{in},li}(k),\tilde{\mathbf{w}}_{\text{out},li}(k)} J_{\text{local},i}(\tilde{\mathbf{u}}_i(k),\tilde{\mathbf{x}}_i(k+1)) + \sum_{j\in\text{neighbors}_i} J_{\text{inter},i}(\tilde{\mathbf{w}}_{\text{in},ji}(k),\tilde{\mathbf{w}}_{\text{out},ji}(k))$$

subject to

- dynamics of subnetwork i over the horizon
- initial local state, disturbances, additional constraints

2. Multi-agent MPC

- Additional objective function $J^{(s)}_{\text{inter},i}(\tilde{\mathbf{w}}_{\text{in},ji}(k), \tilde{\mathbf{w}}_{\text{out},ji}(k)) =$

$$\begin{bmatrix} \tilde{\boldsymbol{\lambda}}^{(s)}_{\text{in},ji}(k) \\ -\tilde{\boldsymbol{\lambda}}^{(s)}_{\text{out},ij}(k) \end{bmatrix}^{\text{T}} \begin{bmatrix} \tilde{\mathbf{w}}_{\text{in},ji}(k) \\ \tilde{\mathbf{w}}_{\text{out},ji}(k) \end{bmatrix} + \frac{\gamma}{2} \left\| \begin{bmatrix} \tilde{\mathbf{w}}_{\text{in,prev},ij}(k) - \tilde{\mathbf{w}}_{\text{out},ji}(k) \\ \tilde{\mathbf{w}}_{\text{out,prev},ij}(k) - \tilde{\mathbf{w}}_{\text{in},ji}(k) \end{bmatrix} \right\|^2_2,$$

where for each j that is a neighbor that solved its problem before i in iteration s:

$$\tilde{\mathbf{w}}_{\text{in,prev},ij}(k) = \tilde{\mathbf{w}}^{(s)}_{\text{in},ij} \quad \text{and} \quad \tilde{\mathbf{w}}_{\text{out,prev},ij}(k) = \tilde{\mathbf{w}}^{(s)}_{\text{out},ij},$$

and where for each j that has not solved its problem in iteration s yet:

$$\tilde{\mathbf{w}}_{\text{in,prev},ij}(k) = \tilde{\mathbf{w}}^{(s-1)}_{\text{in},ij} \quad \text{and} \quad \tilde{\mathbf{w}}_{\text{out,prev},ij}(k) = \tilde{\mathbf{w}}^{(s-1)}_{\text{out},ij}$$

- Update of $\tilde{\boldsymbol{\lambda}}_{\text{in},ji}$:

$$\tilde{\boldsymbol{\lambda}}^{(s+1)}_{\text{in},ji}(k) = \tilde{\boldsymbol{\lambda}}^{(s)}_{\text{in},ji} + \gamma \left(\tilde{\mathbf{w}}^{(s)}_{\text{in},ji}(k) - \tilde{\mathbf{w}}^{(s)}_{\text{out},ij}(k) \right)$$

- Stopping condition:

$$\left\| \tilde{\boldsymbol{\lambda}}^{(s+1)}_{\text{in},ji} - \tilde{\boldsymbol{\lambda}}^{(s)}_{\text{in},ji} \right\|_\infty \leq \epsilon$$

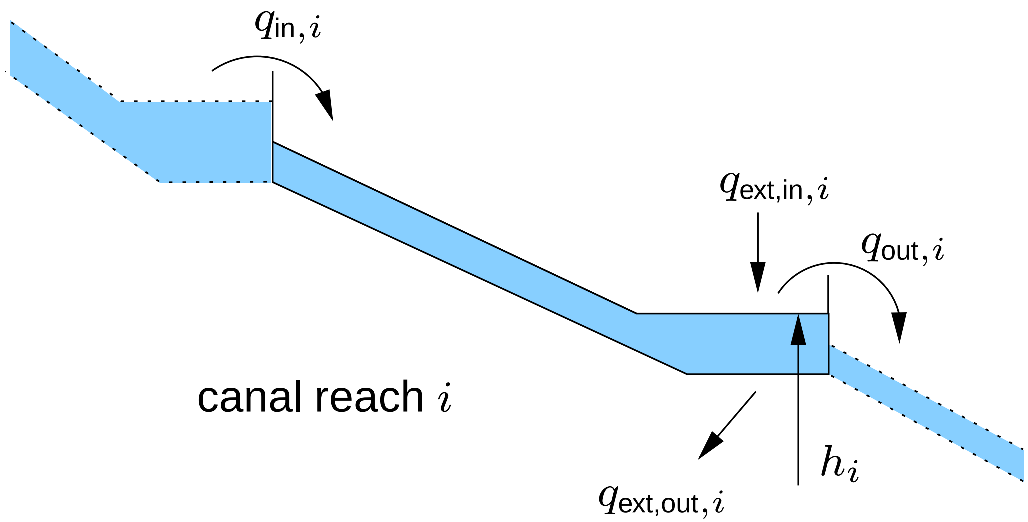
3. Control of an irrigation canal

West-M irrigation canal in the south of Phoenix, Arizona; 10 km long.

Keep water levels close to pre-specified set-points by manipulating undershot gates.

3. Control of an irrigation canal

Dynamics of a canal reach



$$h_i(k+1) = h_i(k) + \frac{T_\text{c}}{c_i} q_{\text{in},i}(k - k_{\text{d},i}) - \frac{T_\text{c}}{c_i} q_{\text{out},i}(k) + \frac{T_\text{c}}{c_i} q_{\text{ext,in},i}(k) - \frac{T_\text{c}}{c_i} q_{\text{ext,out},i}(k)$$

$$q_{\text{out},i}(k) = q_{\text{in},j}(k)$$

So:

$$\mathbf{w}_{\text{in},j_{i,\text{down}}i}(k) = q_{\text{out},i}(k) \qquad \mathbf{w}_{\text{out},j_{i,\text{up}}i}(k) = q_{\text{in},i}(k)$$

3. Control of an irrigation canal

Control objectives of a canal reach controller

- Minimize deviations of water levels from water level set-points
- Minimize changes in determined flow set-points
- So:

$$J_{\text{local},i}(\cdot) = \sum_{l=0}^{N-1} p_{h,i}\left(h_i(k+1+l) - h_{\text{ref},i}\right)^2 + \sum_{l=0}^{N-1} p_{u,i}\left(u_i(k+l) - u_i(k-1+l)\right)^2$$

  where
  - $p_{h,i}$ quadratic cost on water level deviation,
  - $p_{u,i}$ quadratic cost on change in the flow set-points
- Hard constraints have to be satisfied (min and max water level)

3. Control of an irrigation canal

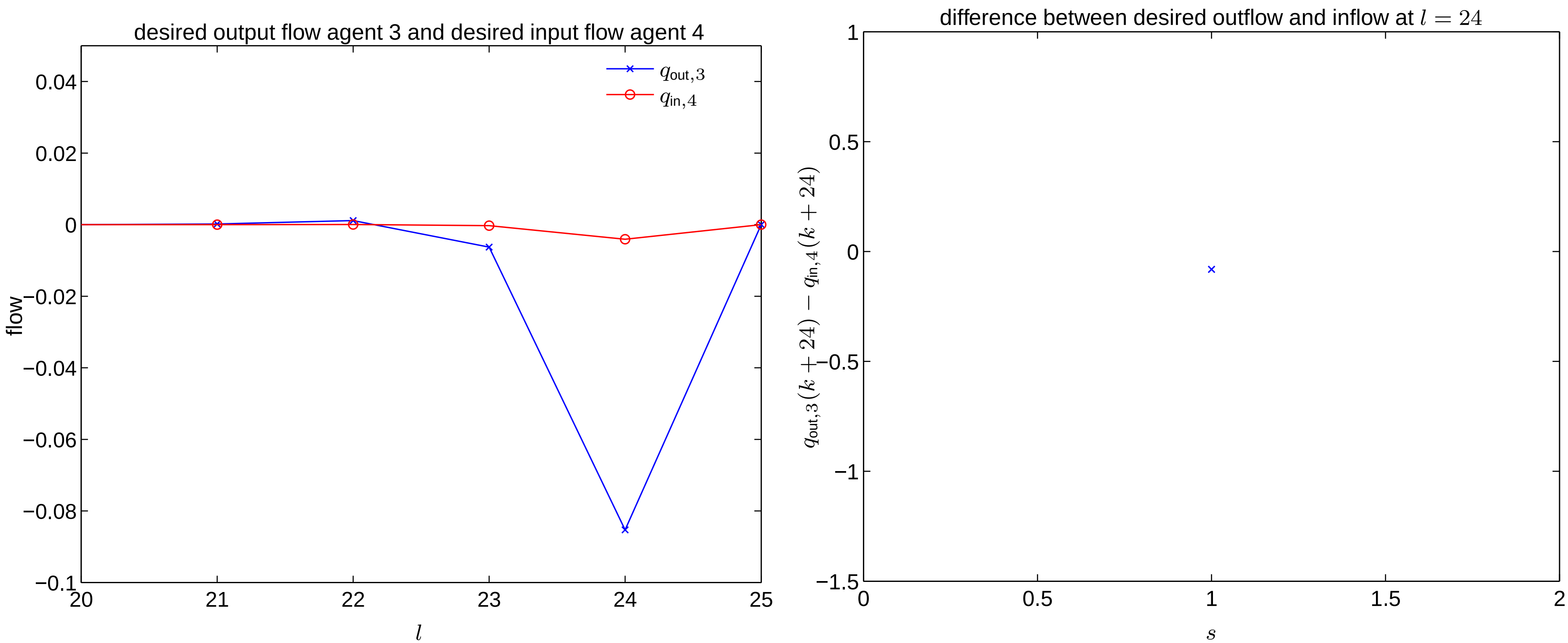
Simulation

- Implementation
  - Benchmark system implemented in Matlab 7.3.
  - Optimization using CPLEX v10.0 through Tomlab 5.7 interface
- Parameters
  - $T_{\mathrm{c}} = 240\,$s, $N = 31$ steps
  - Multi-agent MPC scheme parameters: $c = 1$, $\epsilon = 1.10^{-4}$
  - Cost coefficients: $p_{h,i} = 100$, $p_{u,i} = 10$
- Scenario
  - at $k = 30$: increase of 0.1 m$^3/s$ in offtake of reach 3
  - at $k = 70$: decrease of 0.1 m$^3/s$ in offtake of reach 3

3. Control of an irrigation canal

Results for a particular iteration

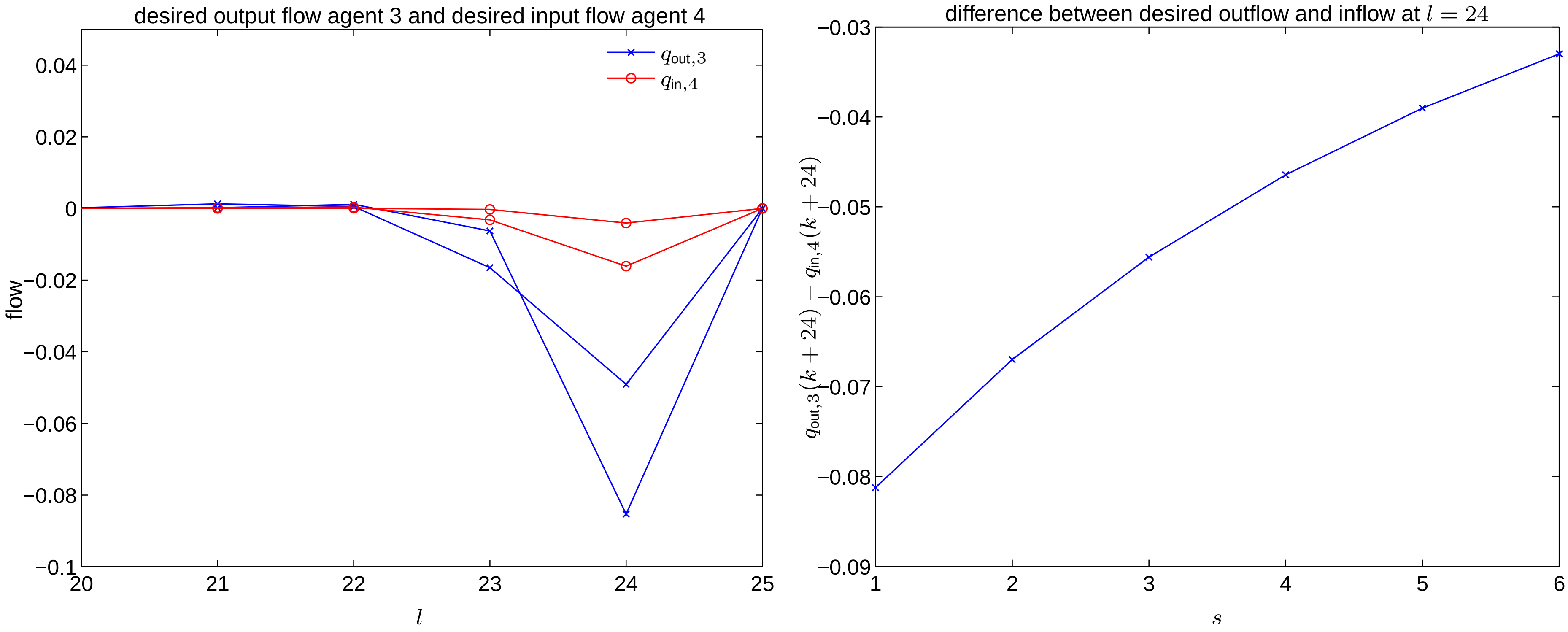
- Right after first disturbance has appeared in the horizon
- Iterations between agent 3 and 4



3. Control of an irrigation canal

Results for a particular iteration

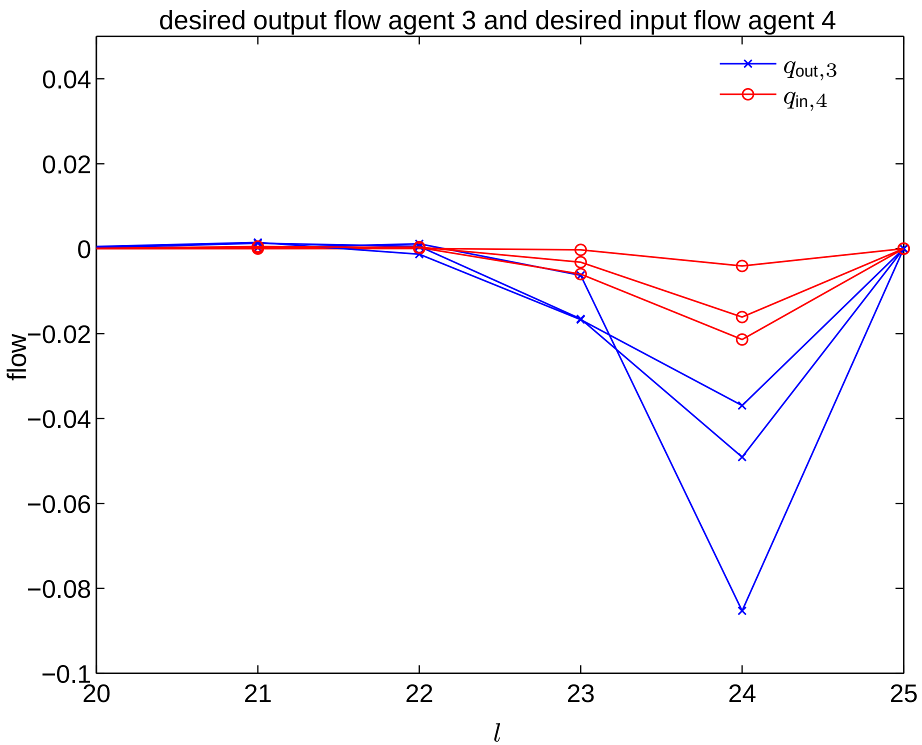
- Right after first disturbance has appeared in the horizon
- Iterations between agent 3 and 4

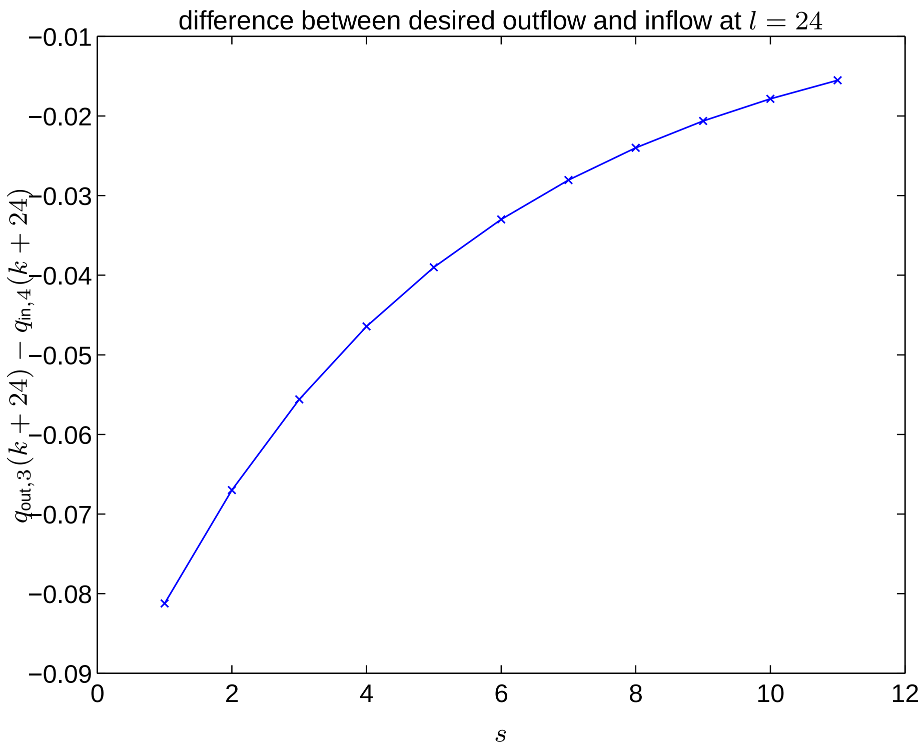


3. Control of an irrigation canal

Results for a particular iteration

- Right after first disturbance has appeared in the horizon
- Iterations between agent 3 and 4

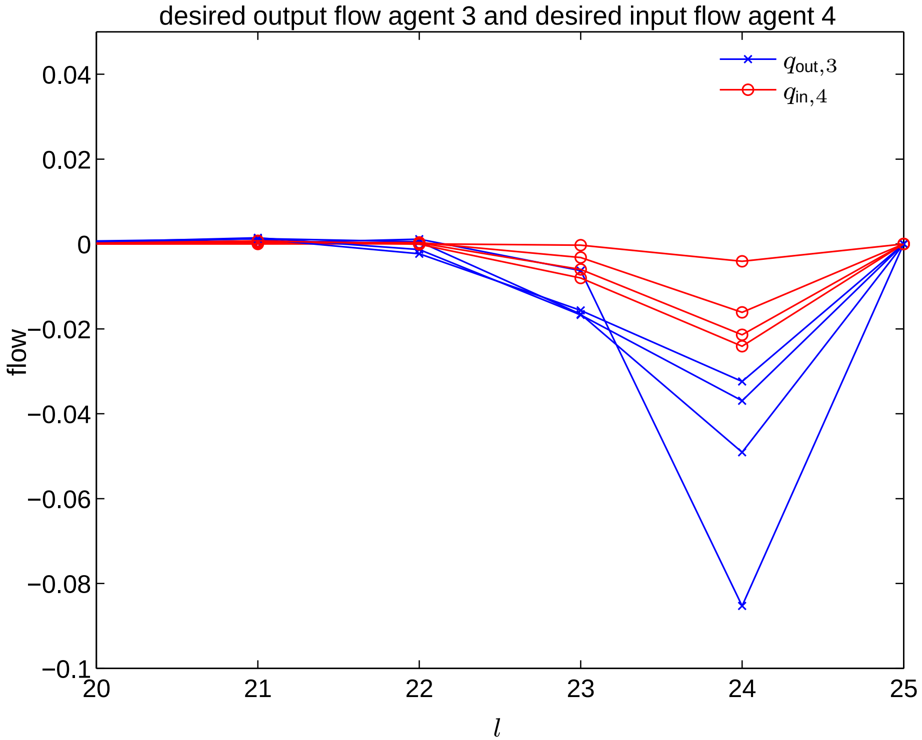


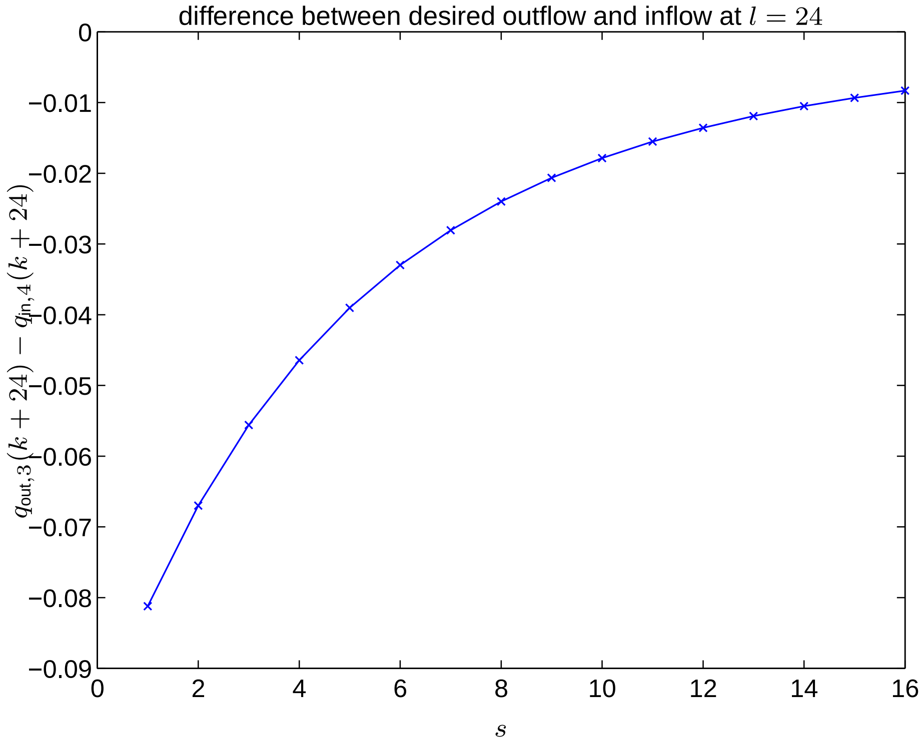


3. Control of an irrigation canal

Results for a particular iteration

- Right after first disturbance has appeared in the horizon
- Iterations between agent 3 and 4

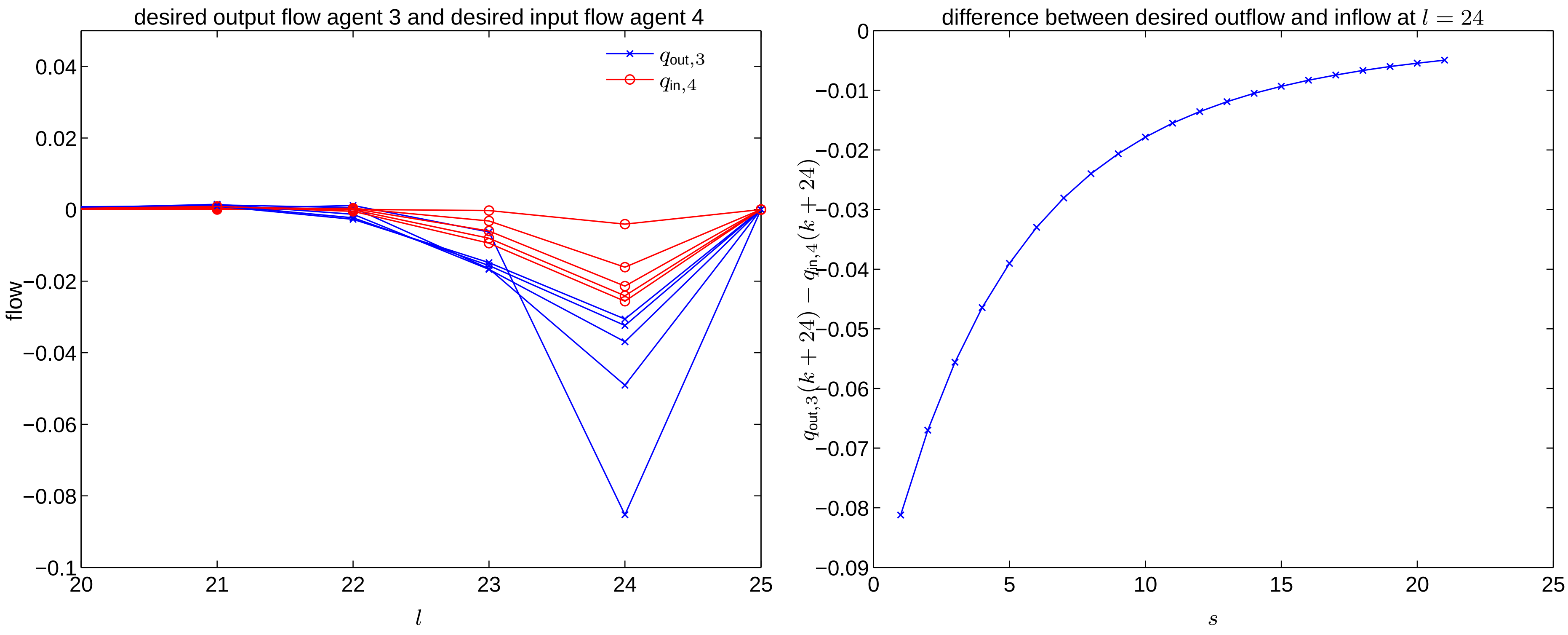




3. Control of an irrigation canal

Results for a particular iteration

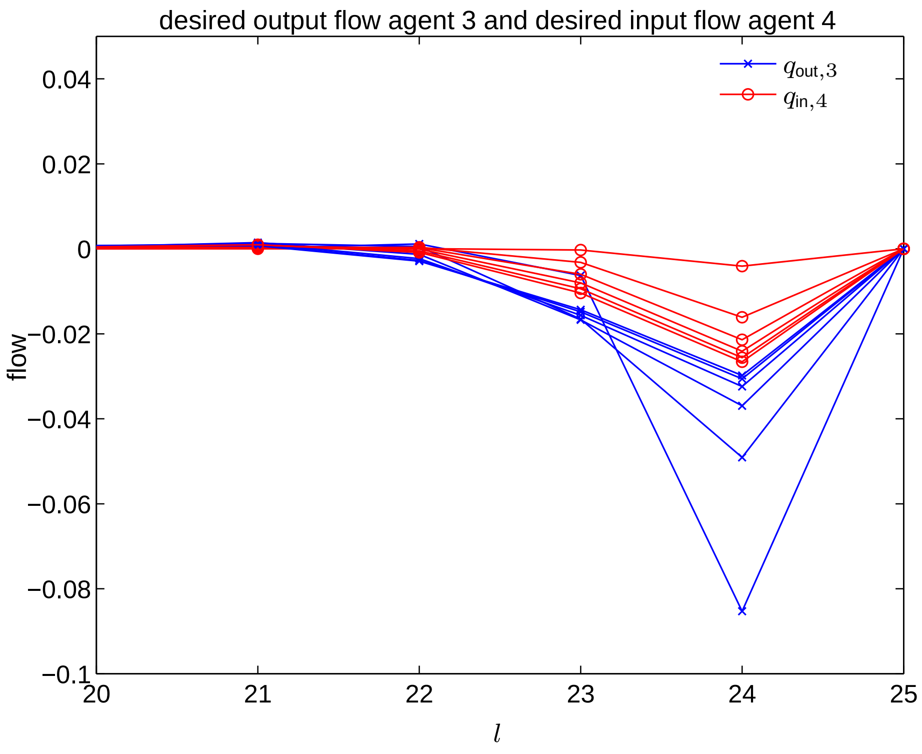
- Right after first disturbance has appeared in the horizon
- Iterations between agent 3 and 4

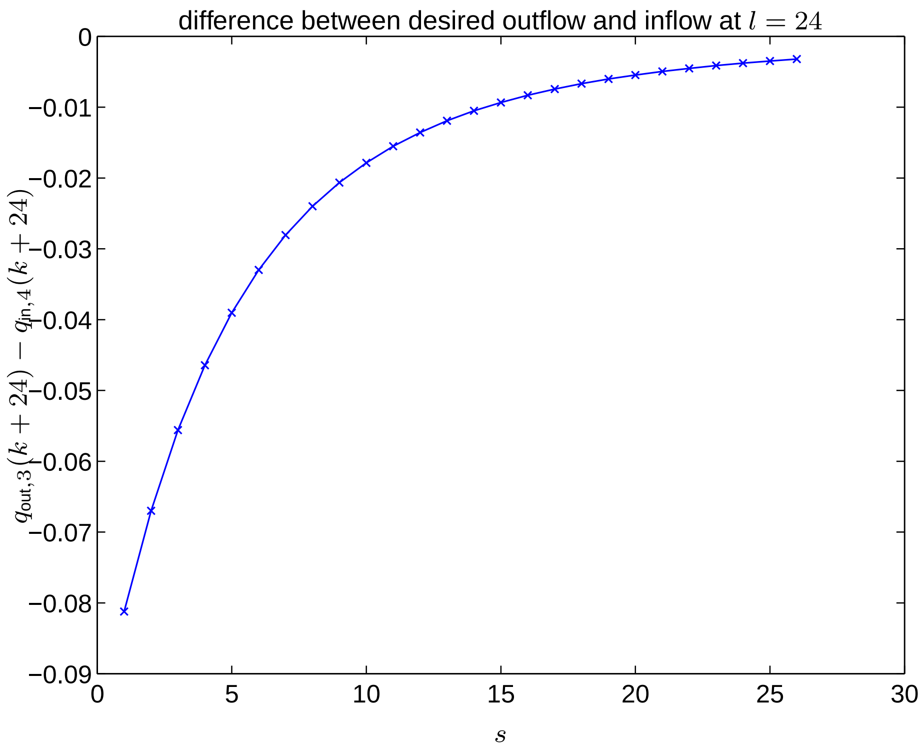


3. Control of an irrigation canal

Results for a particular iteration

- Right after first disturbance has appeared in the horizon
- Iterations between agent 3 and 4

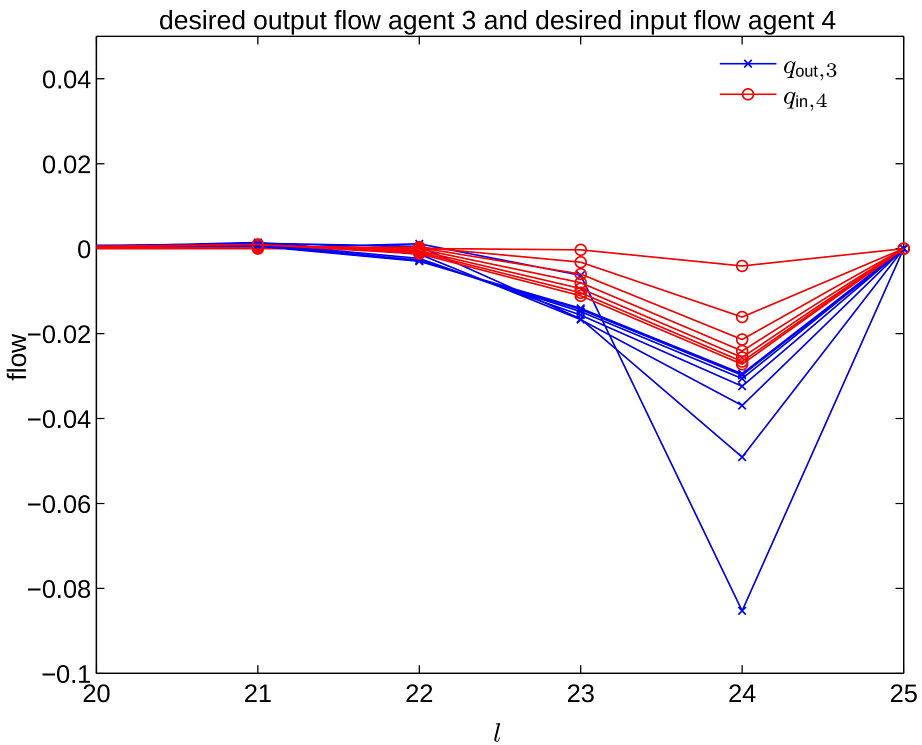


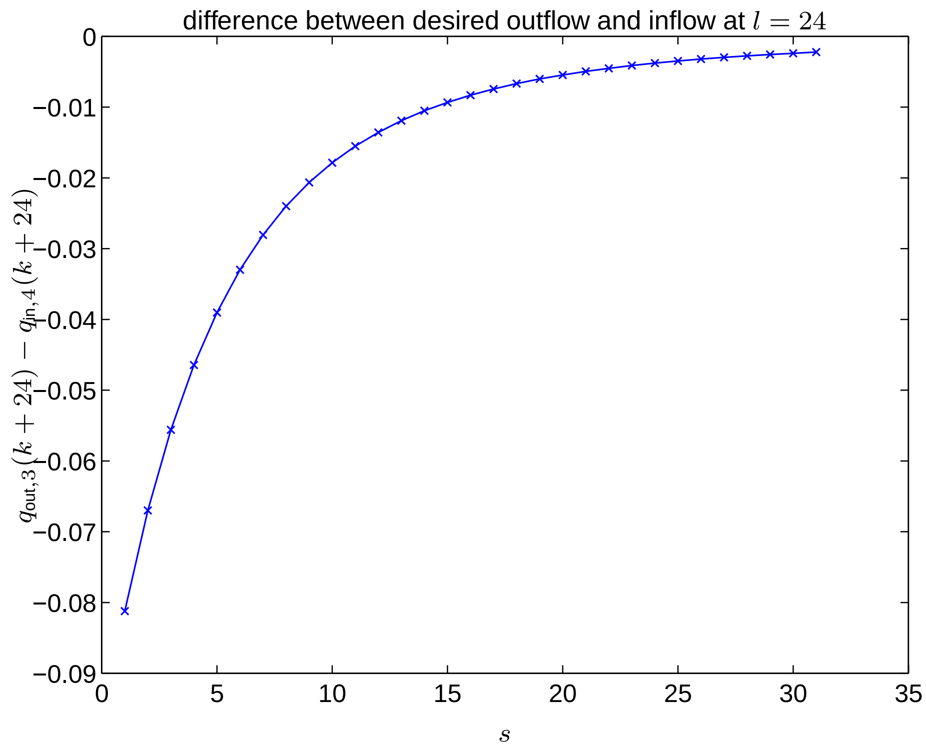


3. Control of an irrigation canal

Results for a particular iteration

- Right after first disturbance has appeared in the horizon
- Iterations between agent 3 and 4

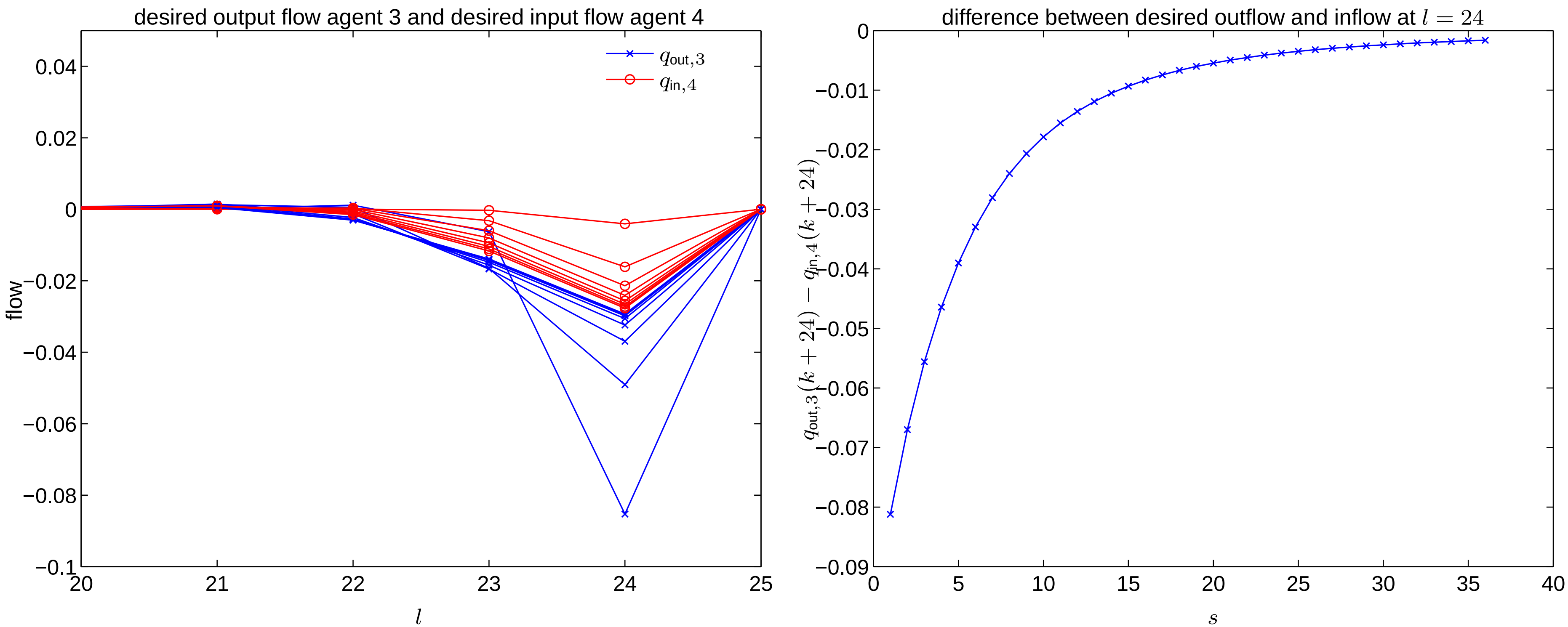




3. Control of an irrigation canal

Results for a particular iteration

- Right after first disturbance has appeared in the horizon
- Iterations between agent 3 and 4

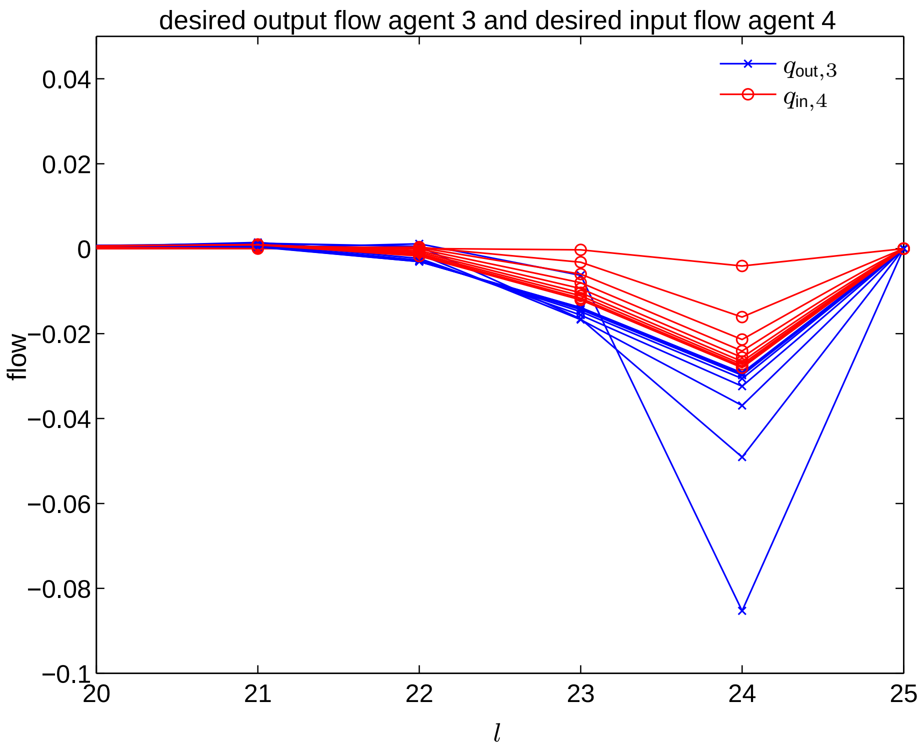


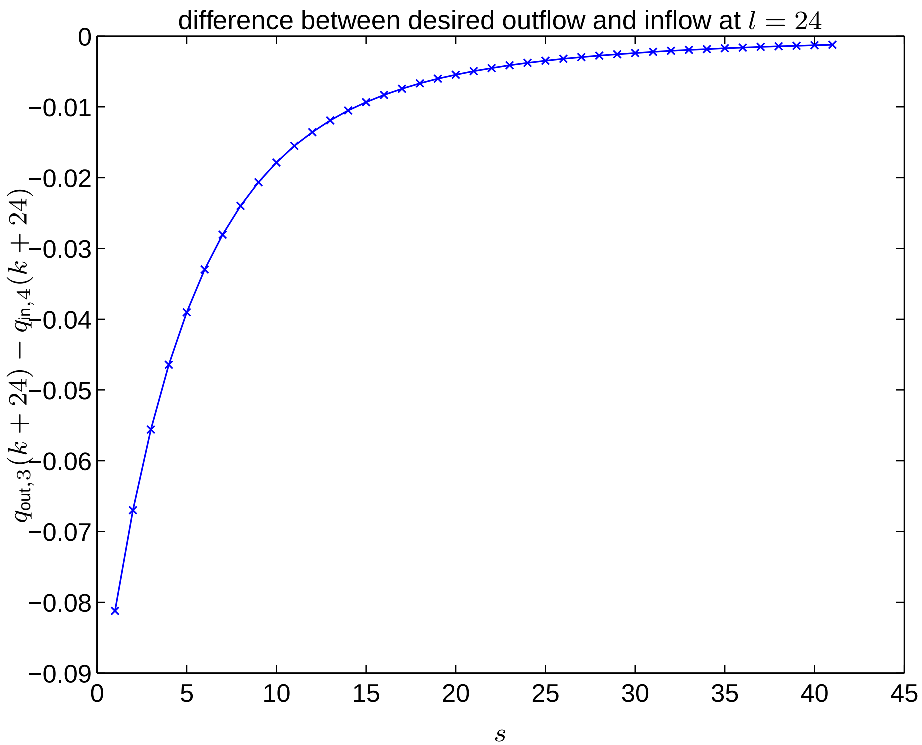
TU Delft

3. Control of an irrigation canal

Results for a particular iteration

- Right after first disturbance has appeared in the horizon
- Iterations between agent 3 and 4

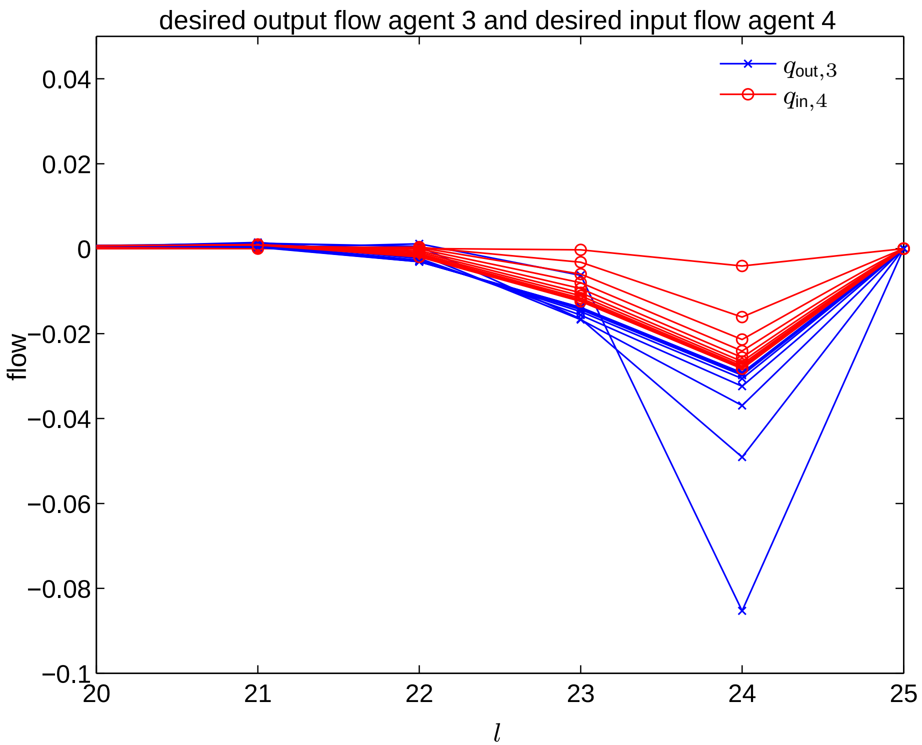


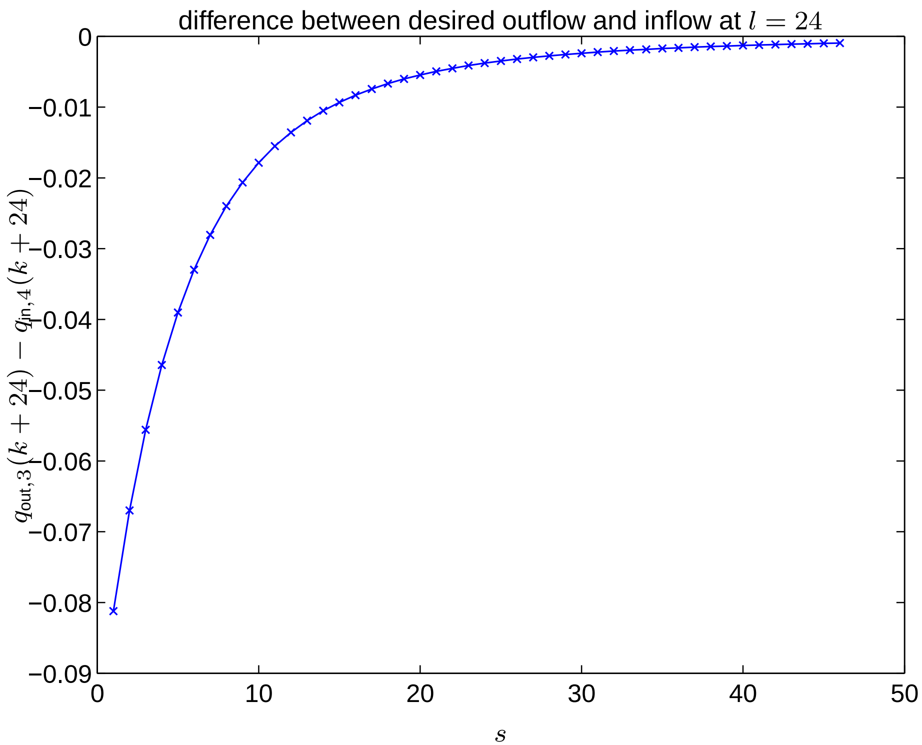


3. Control of an irrigation canal

Results for a particular iteration

- Right after first disturbance has appeared in the horizon
- Iterations between agent 3 and 4

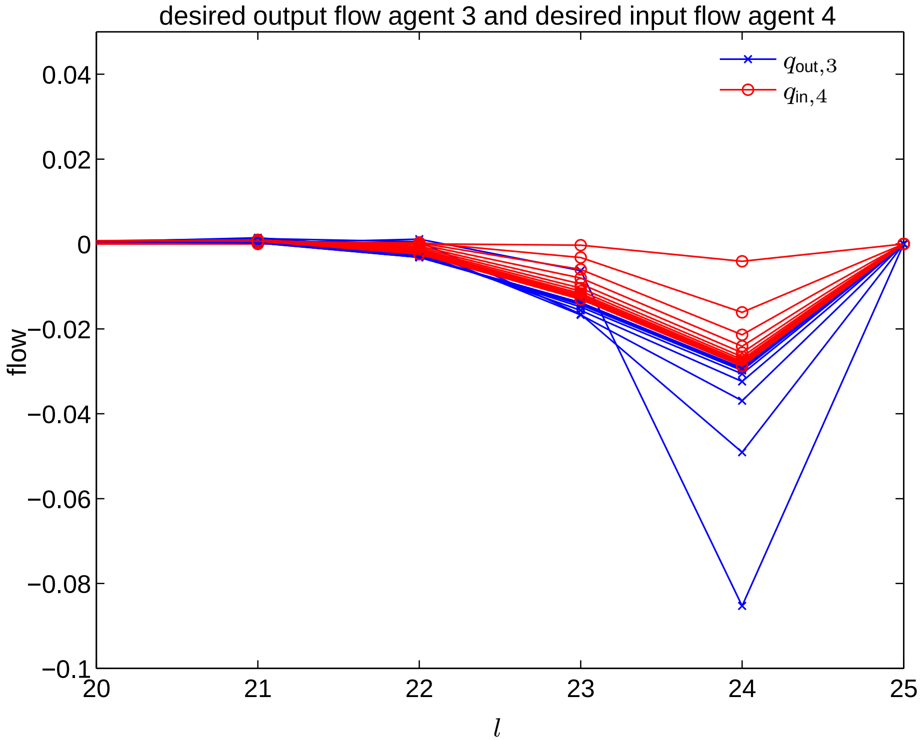


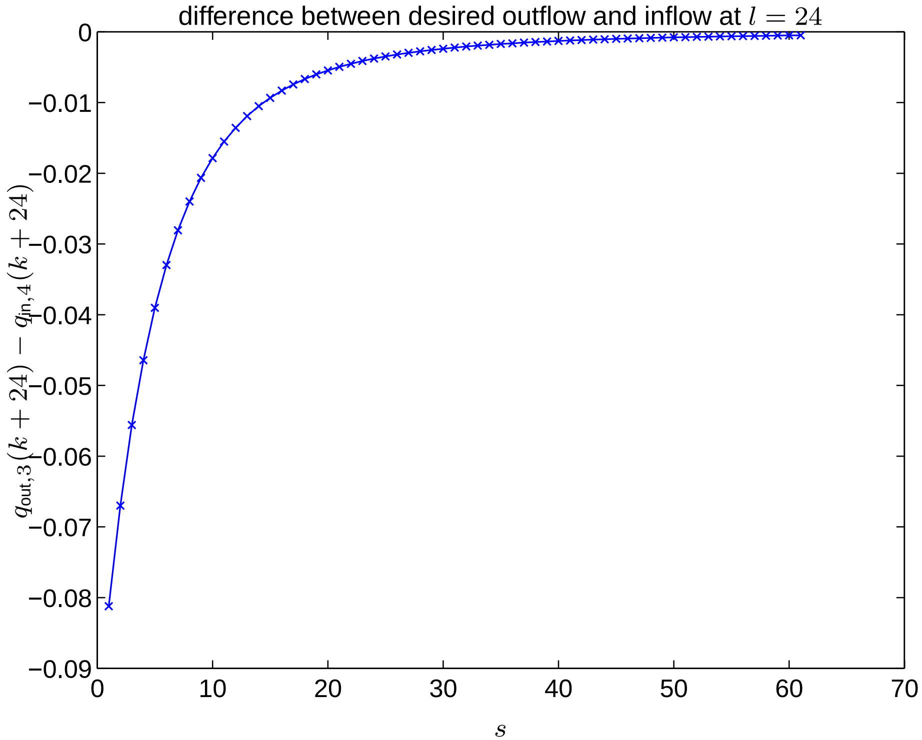


3. Control of an irrigation canal

Results for a particular iteration

- Right after first disturbance has appeared in the horizon
- Iterations between agent 3 and 4

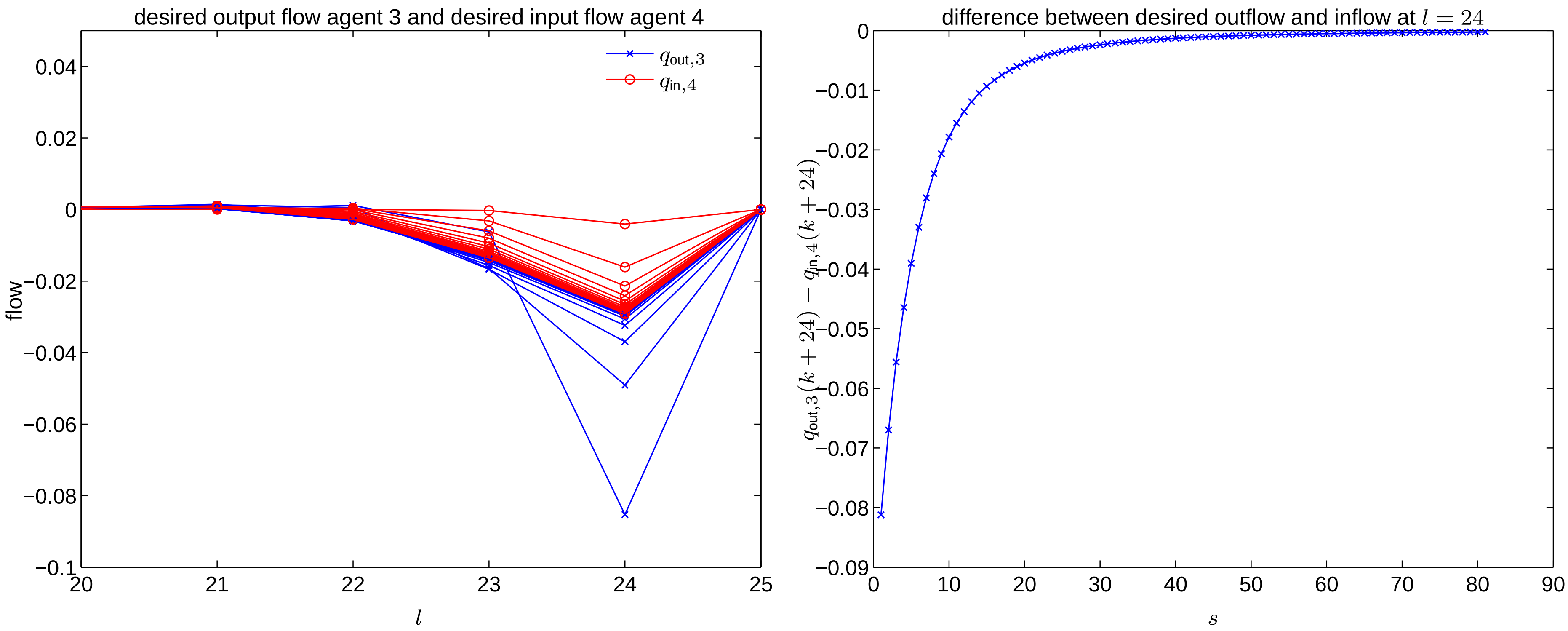




3. Control of an irrigation canal

Results for a particular iteration

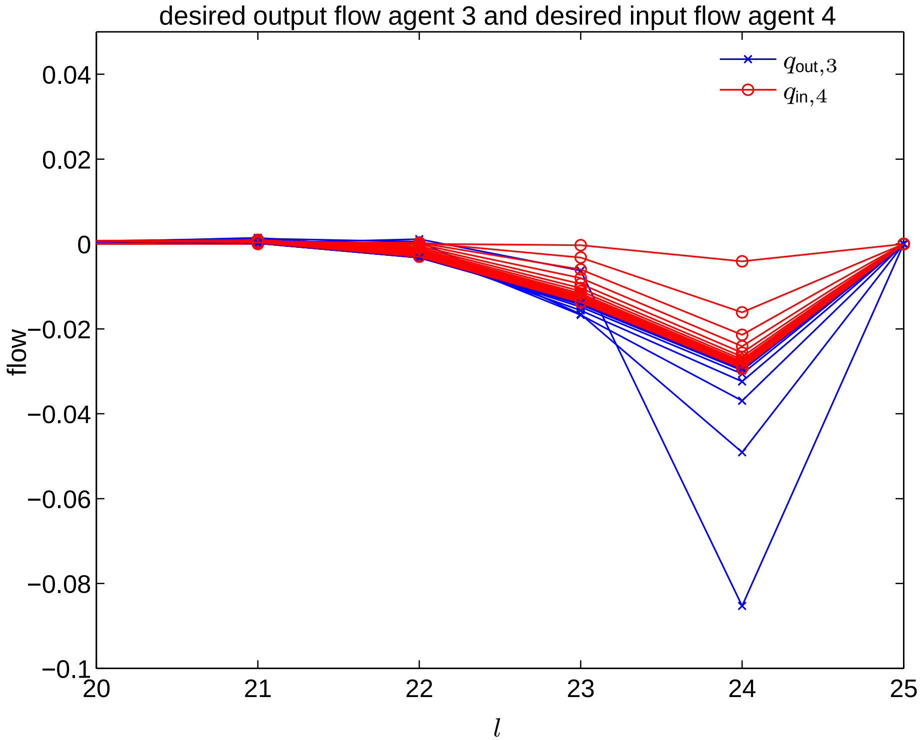
- Right after first disturbance has appeared in the horizon
- Iterations between agent 3 and 4

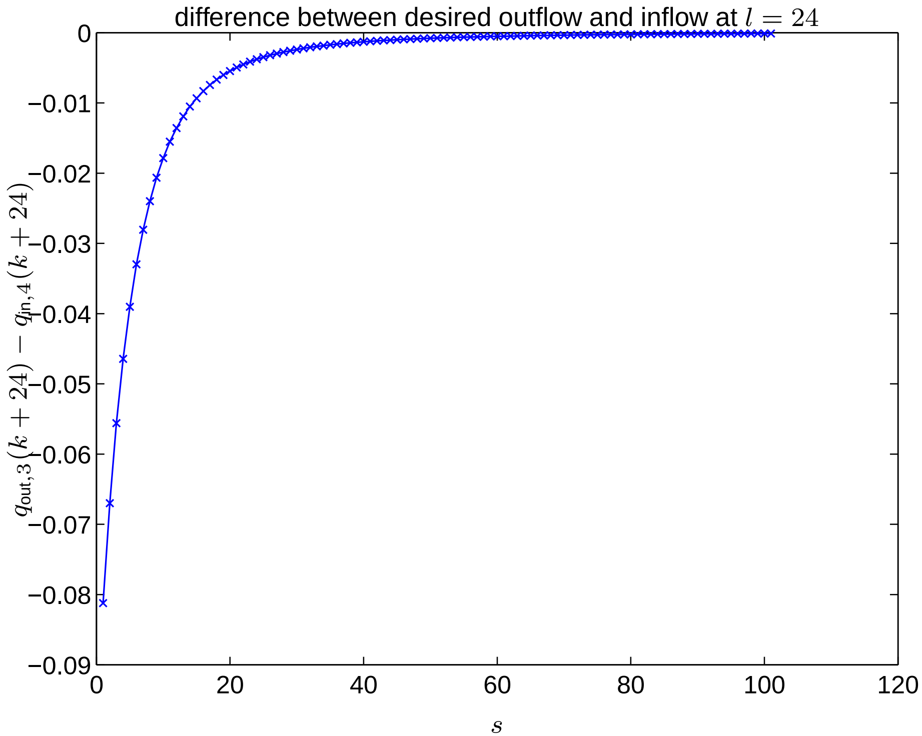


3. Control of an irrigation canal

Results for a particular iteration

- Right after first disturbance has appeared in the horizon
- Iterations between agent 3 and 4

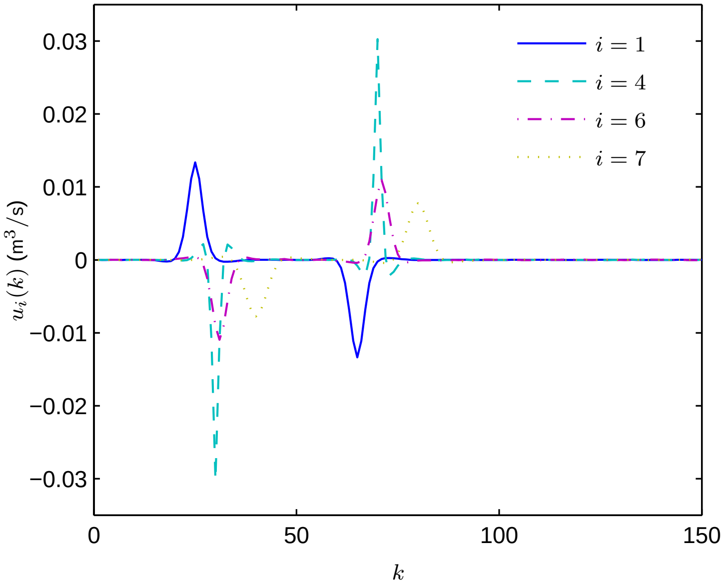




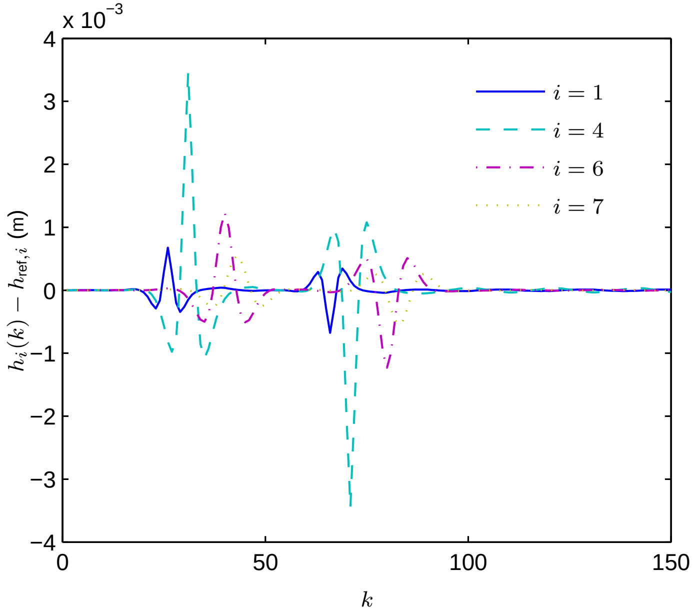
3. Control of an irrigation canal

Results over full simulation of 150 steps

set-points



deviation from reference values



Total cost over simulation within 1% of ideal overall cost

4. Concluding remarks

- Operation of many water systems involves multiple operators/decision makers
- Coordination between operators necessary to improve performance
- Multi-agent model predictive control suitable for optimization-based coordination

- Future work
  - Search for relevant case studies in large-scale systems
  - Cooperative versus non-cooperative schemes
  - Taking into account uncertainty
  - Irrigation canal benchmark:
    - further assessment of the presented scheme
    - simulation experiments on nonlinear SOBEK model
    - prediction model improvement (using PWA models)
    - implementation on a physical system