

# Adjoint-based Control of Traffic Systems

## Application to Ramp Metering

Samitha Samaranayake   Walid Krichene  
Jack Reilly   Maria-Laura Delle Monache

November 23, 2012

# Outline

- 1 Introduction to the adjoint method
  - Optimization of a PDE-constrained system
  - Example: linear system
  - Solving the original problem
- 2 Discretized system dynamics
  - Forward System
  - Lower triangular forward system
  - Exploiting system structure
- 3 Solving the optimization problem via the adjoint method
  - Overview of procedure
  - A few details
- 4 Numerical results

# Outline

- 1 Introduction to the adjoint method
  - Optimization of a PDE-constrained system
  - Example: linear system
  - Solving the original problem
- 2 Discretized system dynamics
  - Forward System
  - Lower triangular forward system
  - Exploiting system structure
- 3 Solving the optimization problem via the adjoint method
  - Overview of procedure
  - A few details
- 4 Numerical results

# Optimization of a PDE-constrained system

## Optimization problem

$$\begin{aligned} & \text{minimize}_{u \in \mathcal{U}} && J(x, u) \\ & \text{subject to} && H(x, u) = 0 \end{aligned}$$

- $x \in \mathcal{X} \subseteq \mathbb{R}^n$ : state variables
- $u \in \mathcal{U} \subseteq \mathbb{R}^m$ : control variables

$$\begin{aligned} J : \mathcal{X} \times \mathcal{U} &\rightarrow \mathbb{R} \\ (x, u) &\mapsto J(x, u) \end{aligned}$$

$$\begin{aligned} H : \mathcal{X} \times \mathcal{U} &\rightarrow \mathbb{R}^{n_H} \\ (x, u) &\mapsto H(x, u) \end{aligned}$$

Want to do gradient descent. How to compute the gradient?

# Outline

- 1 Introduction to the adjoint method
  - Optimization of a PDE-constrained system
  - **Example: linear system**
  - Solving the original problem
- 2 Discretized system dynamics
  - Forward System
  - Lower triangular forward system
  - Exploiting system structure
- 3 Solving the optimization problem via the adjoint method
  - Overview of procedure
  - A few details
- 4 Numerical results

## Example: linear system

### Discrete linear dynamics

$$x_{t+1} = Ax_t + Bu_t, \quad t \in \{0, \dots, T-1\}$$

with initial condition  $x_0$ .

Let

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_T \end{bmatrix}$$

$$u = \begin{bmatrix} u_0 \\ \vdots \\ u_{T-1} \end{bmatrix}$$

## Example: linear system

$$x = \begin{bmatrix} Ax_0 + Bu_0 \\ Ax_1 + Bu_1 \\ \vdots \\ Ax_{T-1} + Bu_{T-1} \end{bmatrix}$$
$$= \begin{bmatrix} 0 & & & & & \\ A & \ddots & & & & \\ & \ddots & \ddots & & & \\ & & & A & & \\ & & & & & 0 \end{bmatrix} x + \begin{bmatrix} B & & & & \\ & \ddots & & & \\ & & \ddots & & \\ & & & \ddots & \\ & & & & B \end{bmatrix} u + \begin{bmatrix} Ax_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Can be written as

$$(\tilde{A} - I)x + \tilde{B}u + c = 0$$

Note:  $(\tilde{A} - I)$  is invertible (lower triangular, with  $-1$  on diagonal). Good: system is deterministic!

# Example: linear system

## Linear system

$$H_x x + H_u u + c = 0$$

- $x \in \mathbb{R}^n$  state
- $u \in \mathbb{R}^m$  control, with  $m \leq n$
- $H_x \in \mathbb{R}^{n \times n}$ , assume invertible
- $H_u \in \mathbb{R}^{n \times m}$
- $c \in \mathbb{R}^n$

want to minimize linear cost function

$$\begin{aligned} & \text{minimize}_{u \in \mathcal{U}} && J_x x + J_u u \\ & \text{subject to} && H_x x + H_u u + c = 0 \end{aligned}$$

$J_x \in \mathbb{R}^{1 \times n}$  and  $J_u \in \mathbb{R}^{1 \times m}$  are given row vectors.



## Example: linear system

### Optimization problem

$$\begin{aligned} & \text{minimize}_{u \in \mathcal{U}} && J_x x + J_u u \\ & \text{subject to} && H_x x + H_u u + c = 0 \end{aligned}$$

An equivalent problem is

$$\text{minimize}_{u \in \mathcal{U}} - J_x H_x^{-1} (H_u u + c) + J_u u$$

and the gradient is

### Gradient

$$\nabla_u J = -J_x H_x^{-1} H_u + J_u$$

# Example: linear system

## Gradient

$$\nabla_u J = -J_x H_x^{-1} H_u + J_u$$

Two ways to compute the first term

## Forward

$$\begin{aligned} J_x M \\ H_x M = -H_u \end{aligned}$$

Solve for  $M \in \mathbb{R}^{n \times m}$ :  $m$  inversions

$$H_x [ M_1 \mid \dots \mid M_m ] = [ H_{u_1} \mid \dots \mid H_{u_m} ]$$

Cost  $O(mn^2)$ .

Then product  $1 \times n$  times  $n \times m$ :  $O(nm)$

## Adjoint

$$\begin{aligned} \lambda^T H_u \\ \lambda^T H_x = -J_x \end{aligned}$$

Solve for  $\lambda \in \mathbb{R}^n$ : 1 inversion

$$H_x^T \lambda = J_x^T$$

Cost  $O(n^2)$ .

Then product  $1 \times n$  times  $n \times m$ :  $O(nm)$

# Outline

- 1 Introduction to the adjoint method
  - Optimization of a PDE-constrained system
  - Example: linear system
  - Solving the original problem
- 2 Discretized system dynamics
  - Forward System
  - Lower triangular forward system
  - Exploiting system structure
- 3 Solving the optimization problem via the adjoint method
  - Overview of procedure
  - A few details
- 4 Numerical results

# Optimization of a PDE-constrained system

## General problem

$$\begin{aligned} &\text{minimize}_{u \in \mathcal{U}} && J(x, u) \\ &\text{subject to} && H(x, u) = 0 \end{aligned}$$

$$\nabla_u J = \frac{\partial J}{\partial x} \nabla_u x + \frac{\partial J}{\partial u}$$

On trajectories,  $H(x, u) = 0$  constant,  
thus  $\nabla_u H = 0$

$$\frac{\partial H}{\partial x} \nabla_u x + \frac{\partial H}{\partial u} = 0$$

## Linear system

$$\begin{aligned} &\text{minimize}_{u \in \mathcal{U}} && J_x x + J_u u \\ &\text{subject to} && H_x x + H_u u + c = 0 \end{aligned}$$

$$\nabla_u J = J_x M + J_u$$

$$H_x M = -H_u$$

# Optimization of a PDE-constrained system

## General problem

$$\begin{aligned} &\text{minimize}_{u \in \mathcal{U}} && J(x, u) \\ &\text{subject to} && H(x, u) = 0 \end{aligned}$$

$$\nabla_u J = \frac{\partial J}{\partial x} \nabla_u x + \frac{\partial J}{\partial u}$$

On trajectories,  $H(x, u) = 0$  constant,  
thus  $\nabla_u H = 0$

$$\frac{\partial H}{\partial x} \nabla_u x + \frac{\partial H}{\partial u} = 0$$

## Linear system

$$\begin{aligned} &\text{minimize}_{u \in \mathcal{U}} && J_x x + J_u u \\ &\text{subject to} && H_x x + H_u u + c = 0 \end{aligned}$$

$$\nabla_u J = J_x M + J_u$$

$$H_x M = -H_u$$

Instead, solve for  $\lambda \in \mathbb{R}^n$

## Adjoint

$$H_x^T \lambda = J_x^T$$

then

$$\nabla_u J = \lambda^T H_u + J_u$$

# Optimization of a PDE-constrained system

## General problem

$$\begin{aligned} &\text{minimize}_{u \in \mathcal{U}} && J(x, u) \\ &\text{subject to} && H(x, u) = 0 \end{aligned}$$

$$\nabla_u J = \frac{\partial J}{\partial x} \nabla_u x + \frac{\partial J}{\partial u}$$

On trajectories,  $H(x, u) = 0$  constant,  
thus  $\nabla_u H = 0$

$$\frac{\partial H}{\partial x} \nabla_u x + \frac{\partial H}{\partial u} = 0$$

## Adjoint

$$\frac{\partial H^T}{\partial x} \lambda = \frac{\partial J}{\partial x}$$

then

$$\nabla_u J = \lambda^T \frac{\partial H}{\partial u} + \frac{\partial J}{\partial u}$$

## Linear system

$$\begin{aligned} &\text{minimize}_{u \in \mathcal{U}} && J_x x + J_u u \\ &\text{subject to} && H_x x + H_u u + c = 0 \end{aligned}$$

$$\nabla_u J = J_x M + J_u$$

$$H_x M = -H_u$$

Instead, solve for  $\lambda \in \mathbb{R}^n$

## Adjoint

$$H_x^T \lambda = J_x^T$$

then

$$\nabla_u J = \lambda^T H_u + J_u$$

# Computing $\nabla_u J(x, u)$

Want to evaluate

$$\frac{\partial J}{\partial x} \nabla_u x$$

where  $\frac{\partial H}{\partial x} \nabla_u x + \frac{\partial H}{\partial u} = 0$

# Computing $\nabla_u J(x, u)$

Want to evaluate

$$\frac{\partial J}{\partial x} \nabla_u x$$

where  $\frac{\partial H}{\partial x} \nabla_u x + \frac{\partial H}{\partial u} = 0$

If  $\lambda$  is solution to the adjoint equation

$$\frac{\partial J}{\partial x} + \lambda^T \frac{\partial H}{\partial x} = 0$$



# Computing $\nabla_u J(x, u)$

Want to evaluate

$$\frac{\partial J}{\partial x} \nabla_u x$$

where  $\frac{\partial H}{\partial x} \nabla_u x + \frac{\partial H}{\partial u} = 0$

If  $\lambda$  is solution to the adjoint equation

$$\frac{\partial J}{\partial x} + \lambda^T \frac{\partial H}{\partial x} = 0$$

Then

$$\frac{\partial J}{\partial x} \nabla_u x = -\lambda^T \frac{\partial H}{\partial x} \nabla_u x = \lambda^T \frac{\partial H}{\partial u}$$

## Adjoint solution $\lambda$

$$\nabla_u J = \lambda^T \frac{\partial H}{\partial u} + \frac{\partial J}{\partial u}$$

Also useful for sensitivity analysis.

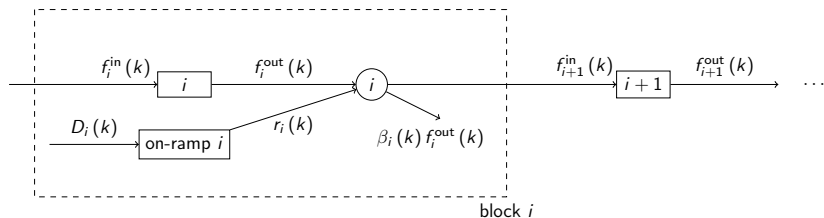
### Sensitivity analysis

$\lambda_k$  is the price of changing  $H_k$

# Outline

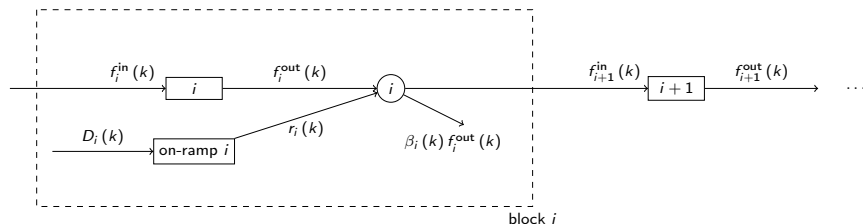
- 1 Introduction to the adjoint method
  - Optimization of a PDE-constrained system
  - Example: linear system
  - Solving the original problem
- 2 Discretized system dynamics
  - Forward System
  - Lower triangular forward system
  - Exploiting system structure
- 3 Solving the optimization problem via the adjoint method
  - Overview of procedure
  - A few details
- 4 Numerical results

# Forward system



- Dynamics based on the discretized LWR PDE
- Piecewise affine system
- Junction solver based on the modified Piccoli model presented a few weeks ago

# Mass conservation



## Density evolution

$$\rho_i(k) = \rho_i(k-1) + \frac{\Delta t}{\Delta x} \left( f_i^{\text{in}}(k-1) - f_i^{\text{out}}(k-1) \right) \quad \forall i \in \{1, \dots, N-1\}, k \in \{1, \dots, T\} \quad (\text{H1a})$$

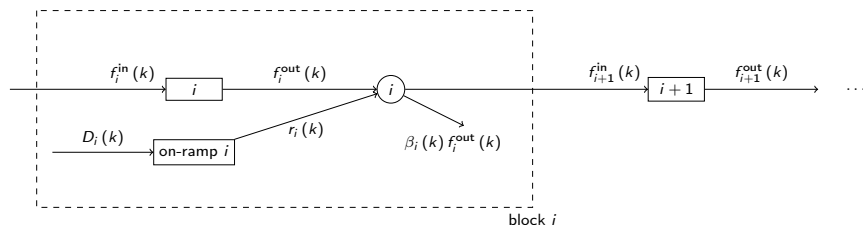
$$\rho_0(k) = \rho_0(k-1) + \frac{\Delta t}{\Delta x} \left( D_0(k-1) - f_0^{\text{out}}(k-1) \right) \quad \forall k \in \{1, \dots, T\} \quad (\text{H1b})$$

$$\rho_N(k) = \rho_N(k-1) + \frac{\Delta t}{\Delta x} \left( f_N^{\text{in}}(k-1) - \delta_N(k-1) \right) \quad \forall k \in \{1, \dots, T\} \quad (\text{H1c})$$

## and initial condition

$$\rho_i(0) = \rho_i^0 \quad \forall i \in \{0, \dots, N\} \quad (11)$$

# Ramp buffer



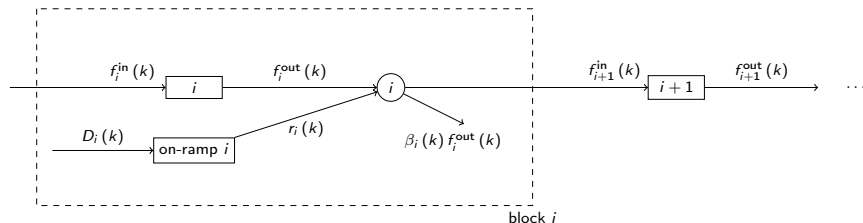
Queue evolution

$$l_i(k) = l_i(k-1) + \Delta t (D_i(k-1) - r_i(k-1)) \quad \forall i \in \{1, \dots, N-1\}, k \in \{1, \dots, T\} \quad (\text{H2})$$

and initial condition

$$l_i(0) = l_i^0 \quad \forall i \in \{1, \dots, N-1\} \quad (12)$$

# Junction supply and demand

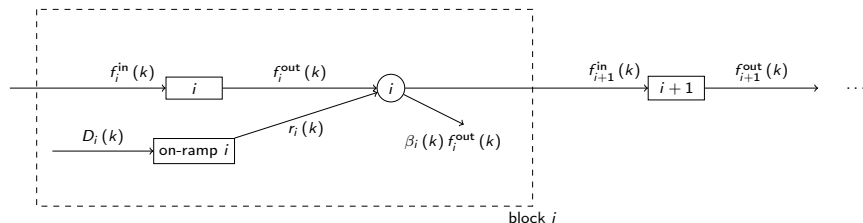


$$\delta_i(k) = \min(F_i, v_i \rho_i(k)) \quad \forall i \in \{0, \dots, N\}, k \in \{0, \dots, T-1\} \quad (\text{H3})$$

$$\sigma_i(k) = \min\left(F_i, w_i \left(\rho_i^{\text{jam}} - \rho_i(k)\right)\right) \quad \forall i \in \{1, \dots, N\}, k \in \{0, \dots, T-1\} \quad (\text{H4})$$

$$d_i(k) = \min(l_i(k), u_i(k)) \quad \forall i \in \{1, \dots, N-1\}, k \in \{0, \dots, T-1\} \quad (\text{H5})$$

# Junction outflow

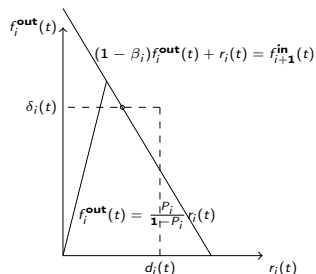
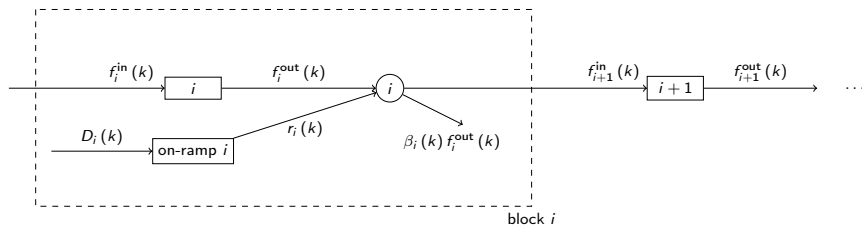


$$f_i^{\text{in}}(k) = \min(\delta_{i-1}(k)(1 - \beta_{i-1}(k)) + d_{i-1}(k), \sigma_i(k)) \quad \forall i \in \{2, \dots, N\}, k \in \{0, \dots, T-1\} \quad (\text{H6a})$$

$$f_1^{\text{in}}(k) = \min(\delta_0(k), \sigma_1(k)) \quad \forall k \in \{0, \dots, T-1\} \quad (\text{H6b})$$

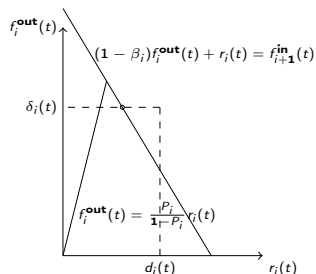
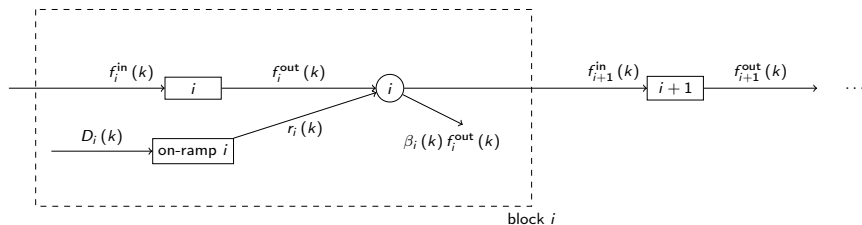


# Junction inflow



$$f_{i+1}^{\text{in}}(k) = \underbrace{r_i(k)}_{=(1-P)f_{i+1}^{\text{in}}(k)} + \underbrace{f_i^{\text{out}}(k)(1 - \beta_i(k))}_{=P f_{i+1}^{\text{in}}(k)}$$

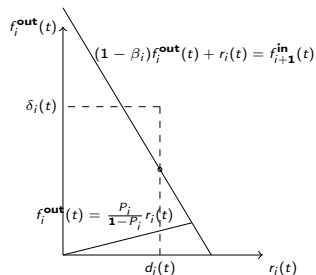
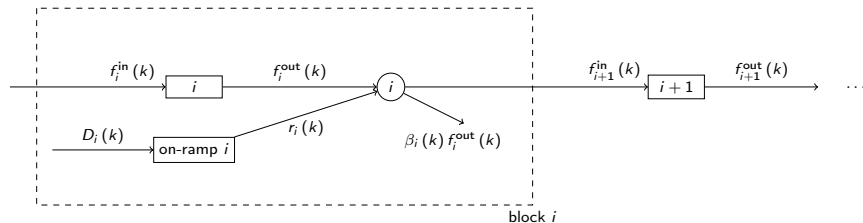
# Junction inflow



$$f_{i+1}^{\text{in}}(k) = \underbrace{r_i(k)}_{=(1-P)f_{i+1}^{\text{in}}(k)} + \underbrace{f_i^{\text{out}}(k)(1-\beta_i(k))}_{=P f_{i+1}^{\text{in}}(k)}$$

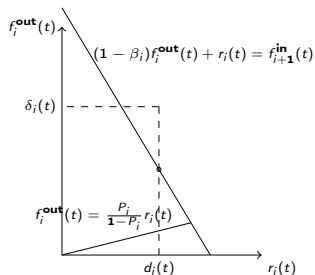
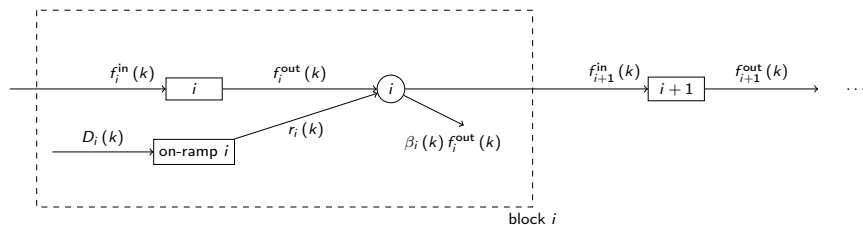
$$f_i^{\text{out}}(k) = \delta_i(k) \quad \text{if } P_i f_{i+1}^{\text{in}}(k) > (1 - \beta_i(k)) \delta_i(k)$$

# Junction inflow



$$f_{i+1}^{\text{in}}(k) = \underbrace{r_i(k)}_{=(1-P)f_{i+1}^{\text{in}}(k)} + \underbrace{f_i^{\text{out}}(k)(1 - \beta_i(k))}_{=P f_{i+1}^{\text{in}}(k)}$$

# Junction inflow

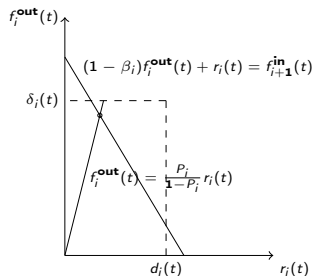
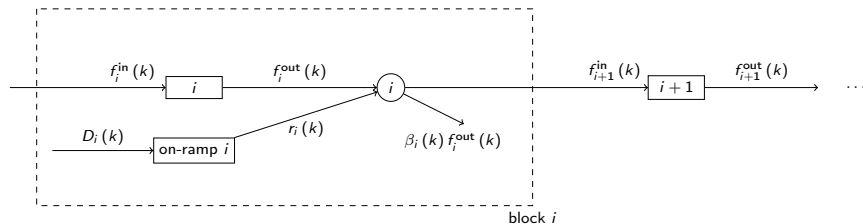


$$f_{i+1}^{\text{in}}(k) = \underbrace{r_i(k)}_{=(1-P)f_{i+1}^{\text{in}}(k)} + \underbrace{f_i^{\text{out}}(k)(1-\beta_i(k))}_{=P f_{i+1}^{\text{in}}(k)}$$

$$f_i^{\text{out}}(k) = \frac{f_{i+1}^{\text{in}}(k) - d_i(k)}{1 - \beta_i(k)}$$

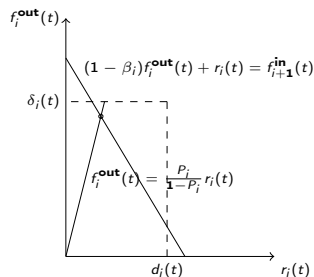
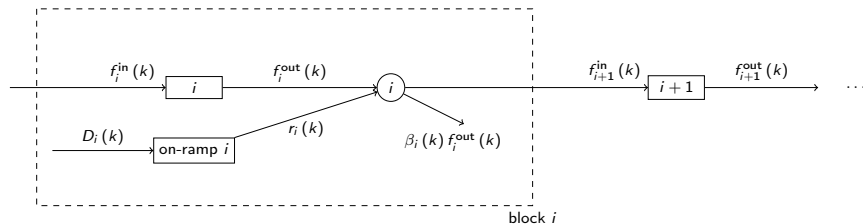
if  $(1 - P_i) f_{i+1}^{\text{in}}(k) > d_i(k)$

# Junction inflow



$$f_{i+1}^{\text{in}}(k) = \underbrace{r_i(k)}_{=(1-P)f_{i+1}^{\text{in}}(k)} + \underbrace{f_i^{\text{out}}(k)(1 - \beta_i(k))}_{=P f_{i+1}^{\text{in}}(k)}$$

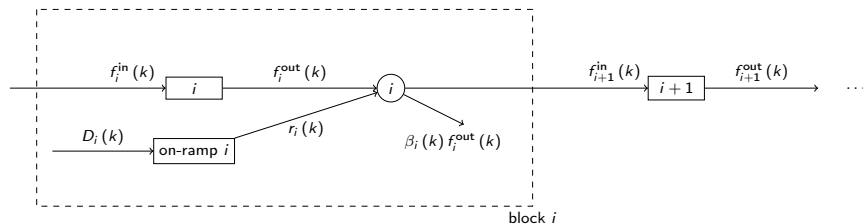
# Junction inflow



$$f_{i+1}^{\text{in}}(k) = \underbrace{r_i(k)}_{=(1-P)f_{i+1}^{\text{in}}(k)} + \underbrace{f_i^{\text{out}}(k)(1 - \beta_i(k))}_{=P f_{i+1}^{\text{in}}(k)}$$

$$f_i^{\text{out}}(k) = \frac{P_i f_{i+1}^{\text{in}}(k)}{1 - \beta_i(k)} \quad \text{otherwise}$$

# Junction inflow

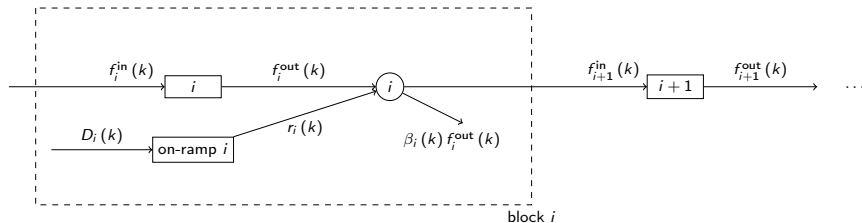


$$f_i^{\text{out}}(k) = \begin{cases} \delta_i(k) & \text{if } (R1_{k,i}) : P_i f_{i+1}^{\text{in}}(k) > (1 - \beta_i(k)) \delta_i(k) \\ \frac{f_{i+1}^{\text{in}}(k) - d_i(k)}{1 - \beta_i(k)} & \text{if } (R2_{k,i}) : (1 - P_i) f_{i+1}^{\text{in}}(k) > d_i(k) \\ \frac{P_i f_{i+1}^{\text{in}}(k)}{1 - \beta_i(k)} & \text{otherwise } (R3_{k,i}) \end{cases}$$

$$\forall i \in \{1, \dots, N-1\}, k \in \{0, \dots, T-1\} \quad (\text{H7a})$$

$$f_0^{\text{out}}(k) = f_1^{\text{in}}(k) \quad \forall k \in \{0, \dots, T-1\} \quad (\text{H7b})$$

# Ramp flow



$$r_i(k) = f_{i+1}^{\text{in}}(k) - f_i^{\text{out}}(k)(1 - \beta_i(k)) \quad \forall i \in \{1, \dots, N-1\}, k \in \{0, \dots, T-1\} \quad (\text{H8})$$



# Outline

- 1 Introduction to the adjoint method
  - Optimization of a PDE-constrained system
  - Example: linear system
  - Solving the original problem
- 2 Discretized system dynamics
  - Forward System
  - Lower triangular forward system
  - Exploiting system structure
- 3 Solving the optimization problem via the adjoint method
  - Overview of procedure
  - A few details
- 4 Numerical results

## Lower triangular forward system

The forward system  $H$  has  $C$  constraints that need to be satisfied over  $T$  time steps for  $N$  cells.

- 1 The causality of the system implies that the state at time  $t$  only depends on the state at times  $t' \leq t$ , so we first iterate over time.
- 2 At each time step, the nature of our system also allows a topological ordering of the variables (no loops in the dependency graph!)
- 3 Finally we iterate over the cells, as there is no equation that couples the same variable at a given time step for two separate cells.

# Lower triangular forward system

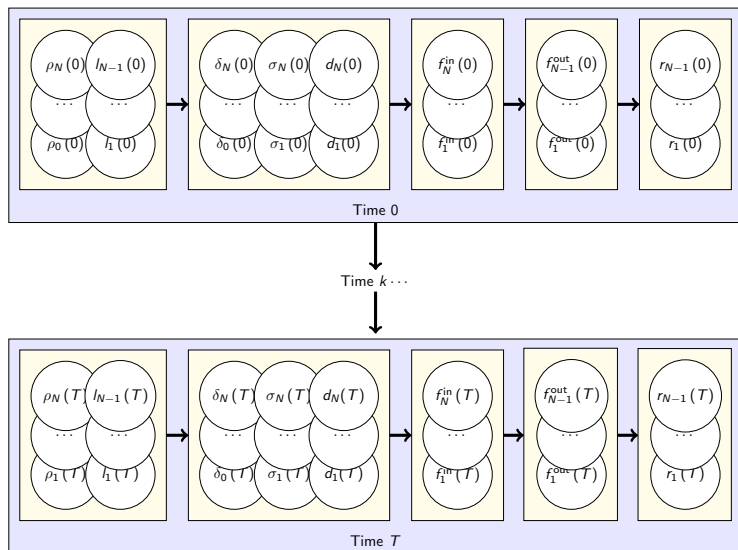


Figure: Dependency diagram of the variables in the system.

## Lower triangular forward system

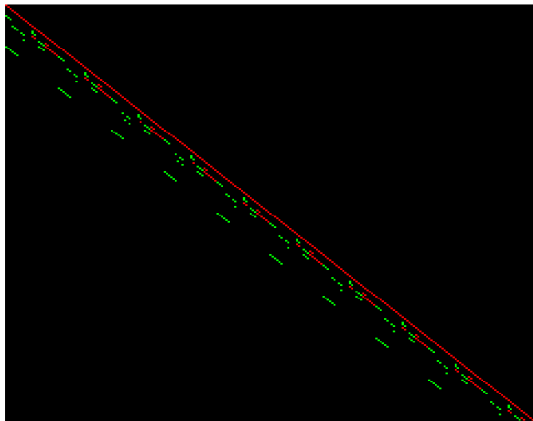
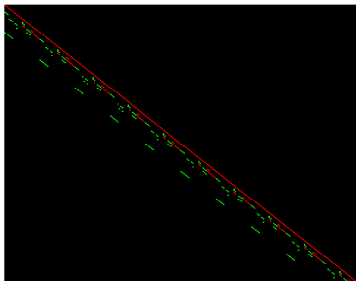


Figure:  $\frac{dH}{dX}$  matrix

# Lower triangular forward system



$$\rho_i(k) = \rho_i(k-1) + \frac{\Delta t}{\Delta x} \left( f_i^{\text{in}}(k-1) - f_i^{\text{out}}(k-1) \right)$$

$$l_i(k) = l_i(k-1) + \Delta t (D_i(k-1) - r_i(k-1))$$

$$\delta_i(k) = \min(F_i, v_i \rho_i(k))$$

$$\sigma_i(k) = \min \left( F_i, w_i \left( \rho_i^{\text{jam}} - \rho_i(k) \right) \right)$$

$$d_i(k) = \min(l_i(k), u_i(k))$$

$$f_i^{\text{in}}(k) = \min \left( \delta_{i-1}(k) (1 - \beta_{i-1}(k)) + d_{i-1}(k), \sigma_i(k) \right)$$

$$f_i^{\text{out}}(k) = \begin{cases} \delta_i(k) & \text{if } P_i f_{i+1}^{\text{in}}(k) > (1 - \beta_i(k)) \delta_i(k) \\ \frac{f_{i+1}^{\text{in}}(k) - d_i(k)}{1 - \beta_i(k)} & \text{if } : (1 - P_i) f_{i+1}^{\text{in}}(k) > d_i(k) \\ \frac{P_i f_{i+1}^{\text{in}}(k)}{1 - \beta_i(k)} & \text{otherwise} \end{cases}$$

$$r_i(k) = f_{i+1}^{\text{in}}(k) - f_i^{\text{out}}(k) (1 - \beta_i(k))$$

- H is piecewise affine
- Solving the forward system gives H(x) as a linear system
- $\left. \frac{dH}{dX} \right|_{X=x}$  is computed simultaneously

# Outline

- 1 Introduction to the adjoint method
  - Optimization of a PDE-constrained system
  - Example: linear system
  - Solving the original problem
- 2 Discretized system dynamics
  - Forward System
  - Lower triangular forward system
  - Exploiting system structure
- 3 Solving the optimization problem via the adjoint method
  - Overview of procedure
  - A few details
- 4 Numerical results

# Exploiting system structure

The structure of the forward system influences the efficiency of computing the adjoint system

Solving for  $\lambda$

$$\frac{\partial H^T}{\partial x} \lambda = -\frac{\partial J}{\partial x}$$

# Exploiting system structure

The structure of the forward system influences the efficiency of computing the adjoint system

## Solving for $\lambda$

$$\frac{\partial H^T}{\partial x} \lambda = -\frac{\partial J}{\partial x}$$

Since  $\frac{\partial H}{\partial x}$  is lower triangular,  $\frac{\partial H^T}{\partial x}$  is an upper triangular matrix

$\implies$  The adjoint system can be solved efficiently using backwards substitution



# Outline

- 1 Introduction to the adjoint method
  - Optimization of a PDE-constrained system
  - Example: linear system
  - Solving the original problem
- 2 Discretized system dynamics
  - Forward System
  - Lower triangular forward system
  - Exploiting system structure
- 3 Solving the optimization problem via the adjoint method
  - Overview of procedure
  - A few details
- 4 Numerical results

# Optimization algorithm

---

## Algorithm 1 Gradient descent loop

---

Pick initial control  $u^{init}$

**while** not converged **do**

$x = \text{forwardSim}(u, IC, BC)$

solve for state trajectory (forward system)

$\lambda = \text{adjointSln}(x, u)$

solve for adjoint parameters (adjoint system)

$\Delta u = \nabla_u J = \lambda^T \frac{\partial H}{\partial u} + \frac{\partial J}{\partial u}$

Compute the gradient (search direction)

$u \leftarrow u + t\Delta u$

update  $u$  using line search along  $\Delta u$

**end while**

---

# Line search

Example 1: decreasing step size

$$t^{(k)} = t^{(1)} / k$$

Example 2: backtracking line-search

- fix parameters  $0 < \alpha < 0.1$  and  $0 < \beta < 1$
- given search direction  $\Delta u$

---

**Algorithm 2** Backtracking line search

---

```
while  $J(u + t\Delta u) - J(u) > \alpha(\nabla_u J)^T(t\Delta u)$  do  
     $t \leftarrow \beta t$   
end while
```

---

# Objective function

The objective function is the total travel time, given by

$$J(x, u) = \sum_{k=0}^T \left( \sum_{i=0}^N \Delta x \rho_i(k) + \sum_{i=1}^{N-1} l_i(k) \right)$$

# Outline

- 1 Introduction to the adjoint method
  - Optimization of a PDE-constrained system
  - Example: linear system
  - Solving the original problem
- 2 Discretized system dynamics
  - Forward System
  - Lower triangular forward system
  - Exploiting system structure
- 3 Solving the optimization problem via the adjoint method
  - Overview of procedure
  - A few details
- 4 Numerical results

# Solving the optimization problem: Constrained optimization?

## Original optimization problem

$$\begin{aligned} & \text{minimize}_{u \in \mathcal{U}} && J(x, u) \\ & \text{subject to} && H(x, u) = 0 \end{aligned}$$

Example: may want to impose minimal metering rate  $u \geq u^{\min}$

## Modified optimization problem: log barrier function

Fix  $t > 0$ .

$$\begin{aligned} & \text{minimize}_{u \in \mathbb{R}^m} && tJ(x, u) - \sum_{i=1}^m \log(u_i - u_i^{\min}) \\ & \text{subject to} && H(x, u) = 0 \end{aligned}$$

As  $t \rightarrow \infty$ , the solution converges to the solution of the original problem.

## Solving the optimization problem: Zero gradient?

$\nabla_u J = \lambda^T \frac{\partial H}{\partial u} + \frac{\partial J}{\partial u}$ . Only non-zero terms in  $\frac{\partial H}{\partial u}$  are

$$H_{k,i}^5 : \quad d_i(k) = \min(l_i(k), u_i(k))$$

$\frac{\partial H}{\partial u}$

$$\frac{\partial H_{i,k}^5}{\partial u_i(k)} = \begin{cases} 0 & \text{if } u_i(k) > l_i(k) \\ 1 & \text{if } u_i(k) < l_i(k) \end{cases}$$

What if we start at  $u > l$ ? Then gradient is zero. Need to add a penalty term

### Penalized problem

$$J(x, u) \leftarrow J(x, u) + h(l - u)$$

where  $h$  is a penalty function, e.g.

- $h(l - u) = c((u - l)^+)^r$  will push the solution to  $l - u \geq 0$
- $h(l - u) = -\frac{1}{t} \log(l - u)$  will keep the solution inside the region  $l - u > 0$

# Notes on oscillation, slow convergence, and second order methods

- System is piecewise affine.  
Gradient descent sometimes oscillates around boundary of two (affine) regions.
- Seems to be improved by using a second order method:  
computes an approximate Hessian (note the real Hessian is zero!)



# Numerical Results

# Numerical Results

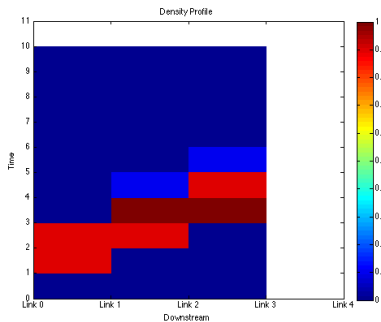


Figure: No ramp metering

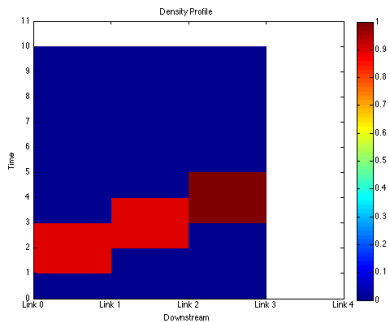
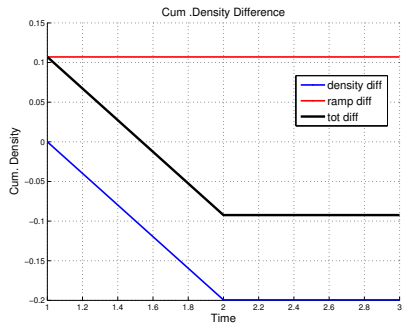
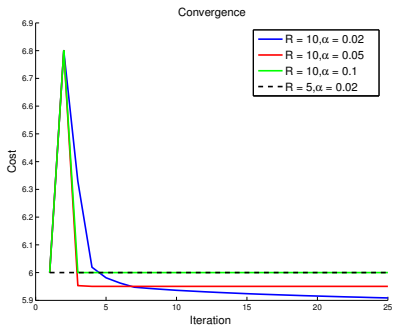


Figure: Optimal ramp metering

# Numerical Results



Thank you.