Adjoint-based optimization of discretized networks of hyperbolic PDEs: application to

~~Efficient method for~~ coordinated ramp metering ~~using the discrete adjoint method~~

Jack Reilly Walid Krichene Samitha Samaranayake
Maria Laura Delle Monache Paola Goatin Alexandre M. Bayen

March 15, 2013

Abstract

abstract

Terminology

Symbol	Description
ρ	Density
ℓ	Ramp Queue
f	Flux
L	Link length
v	Free-flow speed
w	Congestion-wave speed
ρ^c	Critical density
$\rho^{\max}$	Max density
$f^{\max}$	Max flux
$r^{\max}$	Max ramp flux
β	Junction split ratio
p	Junction priority
$\rho^{\max}$	Max density
δ	Link demand
σ	Link supply
d	Ramp demand
f^{in}	Flux in of link
f^{out}	Flux out of link
r	Flux ouf of ramp
D	Demand boundary condition
H	System matrix
C	Cost matrix
x	State vector
u	Control vector

(1) As I explained to Jack, We need to write this article as a general method for networks of PDEs, → discretized → Adjoint and THEN instantiate for freeways.

Alex - 5/8.

1 Introduction

- why is ramp metering useful?
- Coordinated Ramp metering overview
  - Feedback - HERO [14]
  - LP formulations
    - ACTM [6]
    - LN-CTM [13]
  - NL Control
    - [9] papa
    - cooperative, decentralized [15]
- talk about where we fit in here Our approach allows for nonlinear solvers to take advantage of gradient information with no relaxation of model. Additionally, it can be incorporated in an MPC framework to act as a predictive controller with feedback.
- introduce adjoint methods
  - aircraft optimization [4]
  - air traffic control [1]
  - jacquet stuff w/ continuous ramp metering [8]
- overview of rest of paper
  - model
  - objectives
  - adjoint method
  - numerical methods
  - evaluations
  - conclusion

2 Model

This sentence should be written not for a highway but for a general network (as we said, the study will be instantiated for freeways later.)

2.1 Network model

We model a freeway stretch as a sequence of junctions $\mathcal{J} = [J_j]_j, j = \{0, \dots, N-1\}$. Furthermore, each junction J_j has an onramp with a maximum out flux $r_j^{\max}$ and a merging priority p_j, and an offramp with a throughput split ratio β_j[1]. Connecting junctions J_j and J_{j+1} is road link $I_j \in \mathcal{I}$, which has length L_j, and the triangular fundamental diagram parameters: free-flow speed v_j, congestion wave speed w_j, critical density ρ_j^c, max density $\rho_j^{\max}$, and max flux $f_j^{\max}$. ~~The choice of a triangular fundamental diagram is supported by empirical evidence on freeway networks of a distinct "free-flow" phase with constant velocity as a function of density, and a congestion phase with decreasing capacity as a function of density.~~ Figure 1 summarizes the network topology and parameters.

[1] The network parameters can all be time-dependent in the general case, but are presented as constant for simplicity.

do you mean incoming flux

flux function.

explain this section in terms of horizontal queues.

too specific (and do we really care!) could we do it with one flux!

Figure 1: Freeway ~~stretch~~ network and parameters

2.2 Dynamics

~~Since we wish to have numerical stability in our model dynamics, we use a macroscopic traffic flow model, derived from partial differential equations.~~ We use a first-order LWR [10, 16] network model developed in [11], which is ~~particularly~~ suitable for freeway control and can be seen as an extension of the model in [3]. The model in [11] is employed for its ~~strong~~ boundary conditions on the onramps to ensure all flux demand passes through the network [17] and ability to accurately model 2×2 junctions without blocking onramp flux.

~~In order to simulate the model or use it within an optimization framework,~~ we first discretize the model using the Godunov scheme [5], as detailed in [11]. The initial conditions of the model are set by specifying an initial density ρ^0 (vehicles per unit length) for all links and an initial queue length ℓ^0 (vehicles) for all onramps. The boundary conditions are specified as ramp flux demands D for all ramps and all time steps, noting that the upstream mainline source is modeled as an onramp.

The system of equations for the discretized system is given in Equation (1), for a given link $i \in [0, \dots, T]$ and time step $k \in [0, \dots, T]$.

$$
\begin{aligned}
\rho_{i,k} &= \rho_{i,k-1} + \frac{\Delta t}{L_i}\left(f^{\text{in}}_{i,k-1} - f^{\text{out}}_{i,k-1}\right) \\
\ell_{i,k} &= \ell_{i,k-1} + \Delta t\left(D_{i,k-1} - r_{i,k-1}\right) \\
\delta_{i,k} &= \min\left(f_i^{\max}, v_i \rho_{i,k}\right) \\
\sigma_{i,k} &= \min\left(f_i^{\max}, w\left(\rho_i^{\max} - \rho_{i,k}\right)\right) \\
d_{i,k} &= \min\left(r_i^{\max}, \ell_{i,k}/\Delta t\right) \\
f^{\text{in}}_{i,k} &= \min\left(\beta_{i-1}\delta_{i-1,k} + d_{i,k}, \sigma_{i,k}\right) \\
f^{\text{out}}_{i,k} &= \begin{cases} \delta_{i,k} & \frac{\sigma_{i+1,k}p_{i+1}}{1+p_{i+1}} \geq \delta_{i,k}\beta_{i+1}k \\ \left(f^{\text{in}}_{i+1,k} - d_{i_1,k}\right)/\beta_{i+1} & \frac{\sigma_{i+1,k}}{1+p_{i+1}} \geq d_{i,k} \\ \frac{\sigma_{i+1,k}p_{i+1}}{(1+p_{i+1})\beta_{i+1}k} & \text{otherwise} \end{cases} \\
r_{i,k} &= f^{\text{in}}_{i,k} - \beta_i f^{\text{out}}_{i-1,k}
\end{aligned}
\tag{1}
$$

The intermediate variables (those used to calculate the states variables, ρ and ℓ) represent the following values; δ, the demand, is the amount of flux that a link can send downstream in a time step; σ, the supply, is the amount of flux that a link can receive; d, the ramp demand, is the amount of flux that an onramp can send; $f^{\text{in}}/f^{\text{out}}$ is the flux that enters/exits upstream/downstream of a link; r is the flux that exits an onramp.

[2] This is a simplification of [11] for the case when the ramp empties on a time step, with minimal effect on numerical results.

We consider a set of networked partial differential equations to describe the coupled dynamics of the system on each link.

For the case of freeway traffic,

this needs to be explained better so it's understandable. Weak & strong BCs need to be introduced with proper explanation

topology and

this paragraph is supposed to be the ground for the network & PDEs

explain acronym

this paragraph needs to be specific for our model.

too hard to understand: people won't have no idea about the discrete adjoint,

Because the framework proposed in the article relies on

HERE + explain the indexing i,k

say if this is time dependent.

etc.

even if these are defined in the nomenclature, these need to be defined properly in the text, especially since they have double subscripts.

(1)

at ? during ?

put in line 3,

A/ex - 5.8.

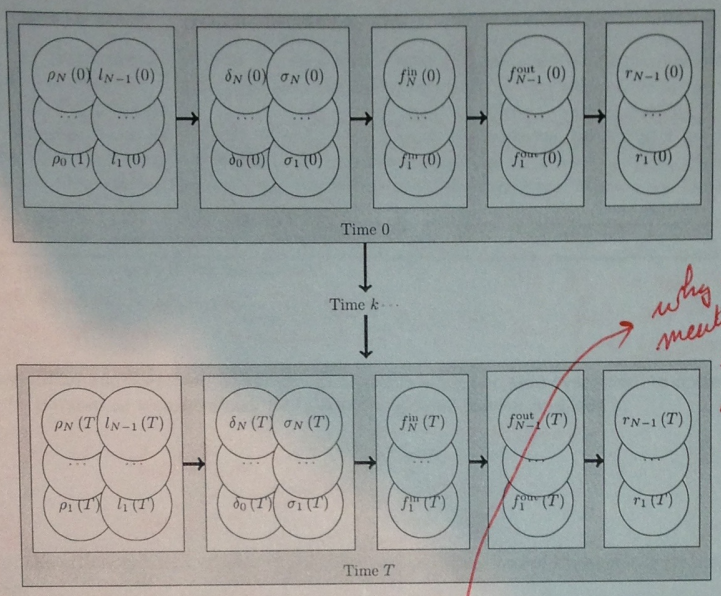

Figure 2: Forward system variable dependencies. Ordering in this fashion leads to a lower-triangular system, which is efficient to solve.

Control parameters. In order to affect the system externally, we must introduce some control parameters. We denote control parameters with u. One method of control, known as variable speed limits [12], would set u as the free-flow velocity of the fundamental diagrams. State estimation problems can be posed in this fashion by letting u be the initial conditions, and searching for an initial condition that evolves to a known final state [7]. Ramp metering sets u as the maximum outflow from onramps.

JDR Comment: below maybe moved to ramp metering specific section

More precisely, we say $u_{i,k}$ sets an upper limit on the flux out of onramp i at time k. For the case of ramp metering, we modify the d equation in (1) to account for this new parameter:

$$d_{i,k} = \min(u_{i,k}, r_i^{\max}, \ell_{i,k}/\Delta t) \tag{2}$$

System matrix. By taking the dynamics conditions in Equation (1), the initial and boundary conditions, and subtracting the right hand side of each equation in the system, we can construct the system matrix $H(x,u) = 0$, where $x \in \mathbb{R}^n$ is the entire state vector, ordered first by time step, then by variable type (e.g. ρ, ℓ, δ, etc), and finally by link. The control vector $u \in \mathbb{R}^m$ is ordered similarly. More details about the ordering and its effect on the efficiency of gradient computation is given in Section 3.4. The system of equations can be solved in linear time via forward-substitution (a result of the system being triangular), similar to other first-order network models such as the cell transmission model [2]. The variable dependencies are graphically depicted in Figure 2.

2.3 Cost function

Given a network of junctions $\mathcal{J}$ and links $\mathcal{I}$, a dynamical system H, and a set of control parameters u, we wish to find an control parameter u^* that minimizes some cost function $C(x, u)$ subject to the physical system's constraints in H. For instance, in the state estimation problem discussed in Section 2.2, an appropriate cost function would penalize for the difference between a known final density state $\bar{\rho}$ and the final estimated density ρ'. For ramp metering, we would seek to reduce congestion on a freeway/ramp system caused by excess ramp flow. One metric that captures congestion is *total travel time*, or the total time spent on the network by every vehicle that enters the network.

JDR Comment: again, this can be moved to later section

We define the cost function for total travel time:

$$C_{\text{ramp}}(x, u) = \sum_{i,k} \rho_{i,k} L_i + \ell_{i,k} \tag{3}$$

Given some initial and boundary conditions that specify a time horizon over which to optimize, then this problem can be phrased as a constrained, finite-horizon optimal control problem:

$$\begin{aligned} &\underset{u}{\text{minimize}} && C(x, u) \\ &\text{subject to:} && H(\dot{x}, u) = 0 \end{aligned} \tag{4}$$

The rest of the article details our solution methods and modifications of Problem (4) for discretized LWR PDE networks.

3 Efficient gradient solutions using adjoint method

3.1 Solving for gradient

Since the LWR PDE is in general nonlinear, gradient-based methods will not give any guarantees of convergence to globally optimal values, nonlinear optimization methods still benefit from gradient information because we seek at least a local optimum. Thus, we seek an efficient method for computing the gradient ∇C_u for an arbitrary cost function C with respect to the control parameter, u. The adjoint method is a way of eliminating intermediate calculations, which are explicitly computed by considering the standard, "forward" system, and reduces the complexity by a factor proportional to the number of control variables.

JDR Comment: need some reference here, and some more information supporting the above statement.

3.2 Adjoint derivation

To calculate the gradient at a particular control value u, we must first perform a forward simulation step to obtain the corresponding state vector x by solving the system $H(x, u) = 0$. Recalling the problem formulation in (4), we can first apply the chain rule to obtain:

$$\nabla C_u = \frac{\partial C}{\partial x}\frac{dx}{du} + \frac{\partial C}{\partial u} \tag{5}$$

where the $\frac{dx}{du}$ term is difficult to compute due to the coupling equation $H(x, u)$.
We then note that since $H(x, u) = 0$, then its total derivative with respect to u must equal:

$$\frac{dH(x,u)}{du} = \frac{\partial H}{\partial x}\frac{dx}{du} + \frac{\partial H}{\partial u} = 0 \tag{6}$$

The resulting expressions in the $\frac{\partial H}{\partial x}$ and $\frac{\partial H}{\partial u}$ matrices are evaluated at (x, u) to obtain matrix values for both terms. Thus, solving for $\frac{dx}{du}$ Directly solving this system for $\frac{dx}{du}$ can be expensive, since $\frac{dx}{du}$ is an $n \times m$ matrix that must be solved for from the above system. This, in general, costs $O(n^3 m)$. Once $\frac{dx}{du}$ is solved, then it can be directly substituted into Equation (5), and thus obtaining the gradient.

We seek a more efficient solution by not solving for $\frac{dx}{du}$ explicitly, but rather making the substitution:

$$\frac{dx}{du} = \left(\frac{\partial H}{\partial x}\right)^{-1} \frac{\partial H}{\partial u} \tag{7}$$

Since our system in lower triangular, the partial derivative with respect to x will also be lower triangular, and will have ones on the diagonal. Thus, the inverse exists.

Then we substitute this expression into Equation (5):

$$\nabla C_u = \frac{\partial C}{\partial x}\left(\frac{\partial H}{\partial x}\right)^{-1} \frac{\partial H}{\partial u} + \frac{\partial C}{\partial u} \tag{8}$$

or if we take the transpose:

$$\nabla C_u^T = \frac{\partial H}{\partial u}^T \left(\frac{\partial H}{\partial x}^{-1}\right)^T \left(\frac{\partial C}{\partial x}\right)^T + \left(\frac{\partial C}{\partial u}\right)^T \tag{9}$$

We make the substitution $\lambda = \left(\frac{\partial H}{\partial x}\right)^{-1^T} \left(\frac{\partial C}{\partial x}\right)^T$, where $\lambda \in \mathbb{R}^n$ is the solution of the following system of equations:

$$\left(\frac{\partial H}{\partial x}\right)^T \lambda = -\left(\frac{\partial C}{\partial x}\right)^T \tag{10}$$

Solving this system is now in general $O(n^3)$, since we only need to solve for one vector $\in \mathbb{R}^n$ instead of a matrix $\in \mathbb{R}^{nm}$ in Equation (6). In Section 3, we show how the cost can be reduced to $O(X)$.

3.3 Adjoint system for ~~LWR~~ networks

The adjoint system described in Equation (10) can be written explicitly by considering the expressions generated from the partial derivatives in $\frac{\partial H}{\partial x}$ and $\frac{\partial C}{\partial x}$. One reason this system is interesting is its independence on the control method being applied. Thus, the derivations in this section generalize to control problems that use this discretized LWR network as the underlying physical model. Similarly, we keep the cost function C general so that we may extend our results to arbitrary cost functions without effecting computational results in this section.

To keep the presentation generic enough to extend to different junction solvers

JDR Comment: mention the LN-CTM, straight Piccoli

, we introduce a function F, which takes a particular cell's previous density, the neighboring cells' previous densities, and the neighboring ramps previous queue sizes, and outputs the density at the next time step:

$$\begin{array}{rccc} F: & \mathbb{R}^5 & \to & \mathbb{R} \\ & (\rho_{i,k}, \rho_{i-1,k}, \rho_{i+1,k}, \ell_{i,k}, \ell_{i+1,k}) & \mapsto & \rho_{i,k+1} \end{array} \tag{11}$$

Similarly, we introduce a function G which updates a queue's length:

$$\begin{array}{rccc} G: & \mathbb{R}^3 & \to & \mathbb{R} \\ & (\ell_{i,k}, \rho_{i-1,k}, \rho_{i,k}) & \mapsto & \ell_{i,k+1} \end{array} \tag{12}$$

Remark. The F and G equations can be made equivalent to the H system by introducing the intermediate variables given in Equation (1), but we gain clarity by limiting the variables to the state variables (density and queue lengths), the sufficient information necessary for solving the forward simulation.

Remark. The upstream and downstream dependencies[3] of the density update equation is a result of the Godunov discretization and the fact that traffic flow propagates information both forward and backward in space. The onramps only connect to the network on their downstream end, and thus only depend on the densities of the mainline links adjacent to the downstream end.

Remark. To handle possibly undefined variables at the boundaries of the network, we develop the convention that at the first link, the upstream density is taken to be zero, and similarly for the downstream density and queue length at the last link. This condition can be considered the mainline boundary conditions. Furthermore, densities and queue lengths at time $k = 0$ are input into the system and considered initial conditions.

Thus, we can recast our system to the following equivalent formulation:

$$\begin{aligned} \rho_{i,k+1} - \quad & F(\rho_{i,k}, \rho_{i-1,k}, \rho_{i+1,k}, \ell_{i,k}, \ell_{i+1,k}) && = 0 \quad \forall i \in [0, \dots, N-1], \forall k \in [0, \dots, T-1] \\ \ell_{i,k+1} - \quad & G(\ell_{i,k}, \rho_{i-1,k}, \rho_{i,k}) && = 0 \quad \forall i \in [0, \dots, N-1], \forall k \in [0, \dots, T-1] \end{aligned} \tag{13}$$

Considering the adjoint system in Equation (10), we introduce the notation for adjoint variables where $\lambda_{a,i,k}$ is the adjoint variable corresponding to the constraint in Equation (13) with the variable $a_{i,k}$ on the far left (a can beither either ρ or ℓ in our formulation). Then we can express the adjoint system in terms of partial derivatives.

$$\lambda_{\rho,i,k} - \frac{\partial F_{i-1,k+1}}{\partial \rho_{i,k}} \lambda_{\rho,i-1,k+1} - \frac{\partial F_{i+1,k+1}}{\partial \rho_{i,k}} \lambda_{\rho,i+1,k+1} - \frac{\partial F_{i,k+1}}{\partial \rho_{i,k}} \lambda_{\rho,i,k+1} - \frac{\partial G_{i,k+1}}{\partial \rho_{i,k}} \lambda_{\rho,i,k+1} - \frac{\partial G_{i-1,k+1}}{\partial \rho_{i,k}} \lambda_{\rho,i-1,k+1} = \frac{\partial C}{\partial \rho_{i,k}} \tag{14}$$

$$\lambda_{\ell,i,k} - \frac{\partial F_{i-1,k+1}}{\partial \ell_{i,k}} \lambda_{\ell,i-1,k+1} - \frac{\partial F_{i,k+1}}{\partial \ell_{i,k}} \lambda_{\ell,i,k+1} - \frac{\partial G_{i,k+1}}{\partial \ell_{i,k}} \lambda_{\ell,i,k+1} = \frac{\partial C}{\partial \ell_{i,k}} \tag{15}$$

for $i \in [0, \dots, N-2]$, $k \in [0, \dots, T-1]$, where we make the notational simplifications:

$$F_{i,k+1} = F(\rho_{i,k} \rho_{i-1,k}, \rho_{i+1,k}, \ell_{i,k}, \ell_{i+1,k})$$

$$G_{i,k+1} = G(\ell_{i,k}, \rho_{i-1,k}, \rho_{i,k})$$

The initial conditions for the adjoint system is:

$$\lambda_{\rho,i,T} - \frac{\partial C}{\partial \rho_{i,T}} \quad \forall i \in [0, \dots, N-1]$$

$$\lambda_{\ell,i,T} - \frac{\partial C}{\partial \ell_{i,T}} \quad \forall i \in [0, \dots, N-1]$$

This is consistent with the notion that adjoint systems are solved "backwards" in time (a result of the transpose operation in Equation (10)), and thus have "initial" conditions at $k = T$. We note that traditional boundary conditions are embedded in the ramp boundary conditions, and are considered "strong" boundary conditions due to its vertical queue model.

Similar to the forward system, the adjoint system can be solved efficiently through back-substitution techniques if the proper ordering of the adjoint variables is taken. We discuss this ordering in the next subsection.

[3] In numerical methods, this is equivalent to having an upwind-downwind discretization scheme.

3.4 Efficient solution of system via depth-first-search system solution

As noted in Section 3.2, the adjoint approach reduces the complexity of solving for the adjoint from $O(mn^3)$ to $O(n^3)$. But this complexity is for general matrices. Our specific The running time of the system solution is then $O(k)$, where k is the number of non-zero elements.

4 Numerical methods

An array of issues that must be considered for this particular problem

- Given affine nature, control parameter must always be 'pushed back' into the active region
- Projection must be done to account for geometric constraints in adjoint. For our case, we must take care to assure non-zero control parameters
- Barrier functions effective, but introduce tuning parameters, but more robust than projections.
- Non-linear solvers can also be used which can take advantage of gradient information.
- Maybe look into what BFGS can handle. If we use bfgs, compare time that it takes to time to compute gradient of u.

5 Evaluation

Evaluation of running time:
show that it can scale linearly with the size of the network. Also, give indication what size networks and time horizons are possible for real-time use in MPC applications.

comparison with similar methods:
This part will have to show under what circumstances our method is better. Which in my mind is the accurate representation of the dynamics and when the predictive nature is beneficial. Need to understand when shortsightedness is harmful, for the HERO case.

6 Conclusions

future work:
rerouting, decentralized, cooperative control

References

[1] A.M. Bayen, R.L. Raffard, and C.J. Tomlin. Adjoint-based control of a new eulerian network model of air traffic flow. *IEEE Transactions on Control Systems Technology*, 14(5):804–818, September 2006.

[2] C. F. Daganzo. The cell transmission model, part II: network traffic. *Transportation Research Part B: Methodological*, 29(2):79–93, 1995.

[3] M. Garavello and B. Piccoli. *Traffic flow on networks*, volume 1. American institute of mathematical sciences Springfield, MA, USA, 2006.

[4] MB Giles and NA Pierce. An introduction to the adjoint approach to design. *Flow, Turbulence and Combustion*, 2000.

[5] S.K Godunov. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. *Matematicheskii Sbornik*, 89(3):271–306, 1959.

[6] G. Gomes and R. Horowitz. Optimal freeway ramp metering using the asymmetric cell transmission model. *Transportation Research Part C: Emerging Technologies*, 14(4):244–262, 2006.

[7] D. Jacquet, C. Canudas de Wit, and D. Koenig. Traffic Control and Monitoring with a Macroscopic Model in the Presence of Strong Congestion Waves. *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 2164–2169.

[8] Denis Jacquet, Carlos Canudas de Wit, and Damien Koenig. Optimal Ramp Metering Strategy with Extended LWR Model, Analysis and Computational Methods. In *Proceedings of the 16th IFAC World Congress*, 2005.

[9] a. Kotsialos and M. Papageorgiou. Nonlinear Optimal Control Applied to Coordinated Ramp Metering. *IEEE Transactions on Control Systems Technology*, 12(6):920–933, November 2004.

[10] M.J. Lighthill and G.B. Whitham. On kinematic waves. II. A theory of traffic flow on long crowded roads. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 229(1178):317, 1955.

[11] M.L. Delle Monache, J. Reilly, S. Samaranayake, W Krichene, and A M Bayen. A pde-ode model for a junction with ramp buffer. 2013.

[12] Ajith Muralidharan, Gunes Dervisoglu, and Roberto Horowitz. Freeway traffic flow simulation using the Link Node Cell transmission model. *American Control Conference*, pages 2916–2921, 2009.

[13] Ajith Muralidharan and Roberto Horowitz. Optimal control of freeway networks based on the Link Node Cell Transmission model. (c).

[14] I. Papamichail and M. Papageorgiou. Traffic-Responsive Linked Ramp-Metering Control. *IEEE Transactions on Intelligent Transportation Systems*, 9(1):111–121, March 2008.

[15] José Ramón, Domínguez Frejo, and Eduardo Fernández Camacho. Global Versus Local MPC Algorithms in Freeway Traffic Control With Ramp Metering and Variable Speed Limits. 13(4):1556–1565, 2013.

[16] P.I. Richards. Shock waves on the highway. *Operations research*, 4(1):42–51, 1956.

[17] I.S. Strub and A.M. Bayen. Weak formulation of boundary conditions for scalar conservation laws: An application to highway traffic modelling. *International Journal of Robust and Nonlinear Control*, 16(16):733–748, 2006.