Coordinated Look-Ahead Scheduling for Real-Time Traffic Signal Control

Xiao-Feng Xie, Stephen F. Smith, Gregory J. Barlow
{xfxie,sfs,gjb}@cs.cmu.edu The Robotics Institute, Carnegie Mellon University

Problem

- Traffic congestion is a *practical* problem resulting in substantial delays and extra fuel costs for drivers, and has negative impacts on environmental conditions

- For urban road networks, better traffic flow requires better traffic signal control, and *real-time, adaptive strategies* offer the biggest payoff

- Real-time decisions: traffic light cycles through a sequence of phases I, each phase i has a variable duration that can range between a minimum and a maximum
- Local observation: inflows of vehicles in the prediction horizon (H), the current phase index and duration of traffic light, and the current decision time

Challenge

Goal: Scalable network-wide optimization

- Intersection level: the number of joint signal control sequences and local observations is huge in the prediction horizon
- Network level: effective coordination for handling non-local impacts between tightly-coupled intersections in a complex network

Contributions

- Real-time traffic signal control based on coordinated look-ahead scheduling

- Each intersection is locally controlled by an agent using a schedule-driven intersection control strategy (SchIC) [3]. At each decision point, each agent constructs a schedule that optimizes movement of the observable traffic through its intersection, and uses this schedule to determine the best control action to take
- For strengthening its local view, each agent queries the scheduled outflows from its upstream neighbors to obtain an optimistic observation, which is capable of incorporating non-local impacts from indirect neighbors

⊛ Summary: *Multi-agent coordination = look-ahead scheduling + coordination mechanism(s)*

References

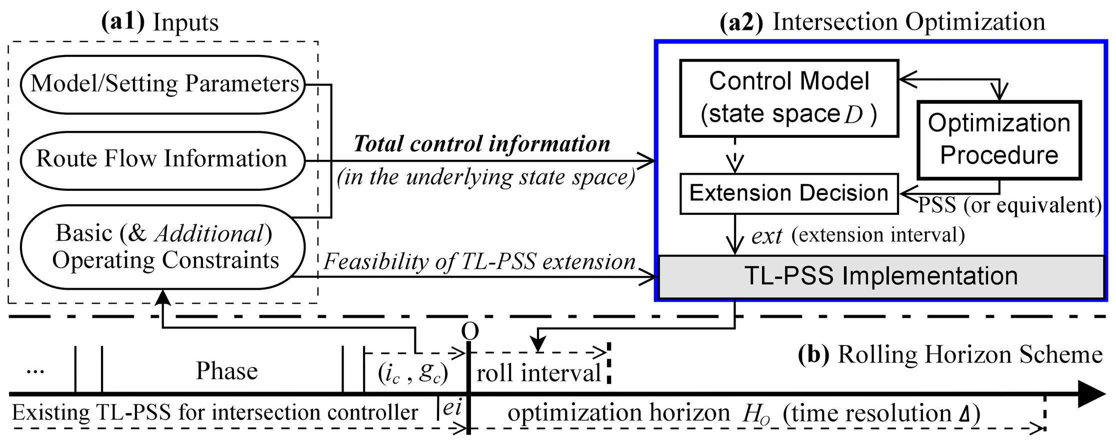
[1] G. J. Barlow. *Improving Memory for Optimization and Learning in Dynamic Environments*. PhD Thesis, 2011.
[2] S. Sen and K. L. Head. Controlled optimization of phases at an intersection. *Transport. Sci.*, 31, 1997.
[3] X.-F. Xie, S. F. Smith, L. Lu, G. J. Barlow. Schedule-driven intersection control. *Transport. Res. C*, 2012.

Acknowledgements

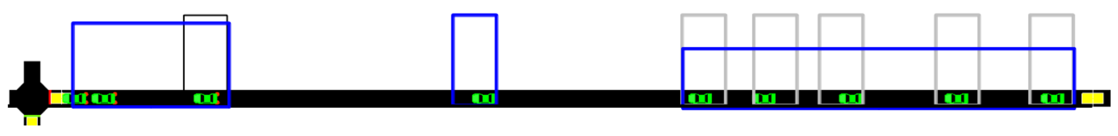
This research was supported in part by the Traffic21 Initiative at Carnegie Mellon University, with support from the Hillman Foundation and the Heinz Endowments, and the CMU Robotics Institute.

Intersection Control

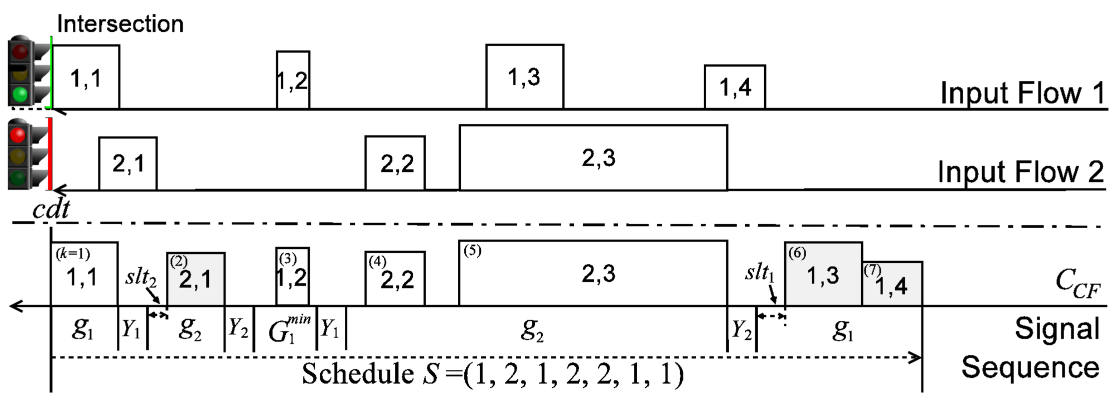
- *Look-ahead scheduling* in a rolling horizon



- Aggregate non-uniform flow into *jobs*



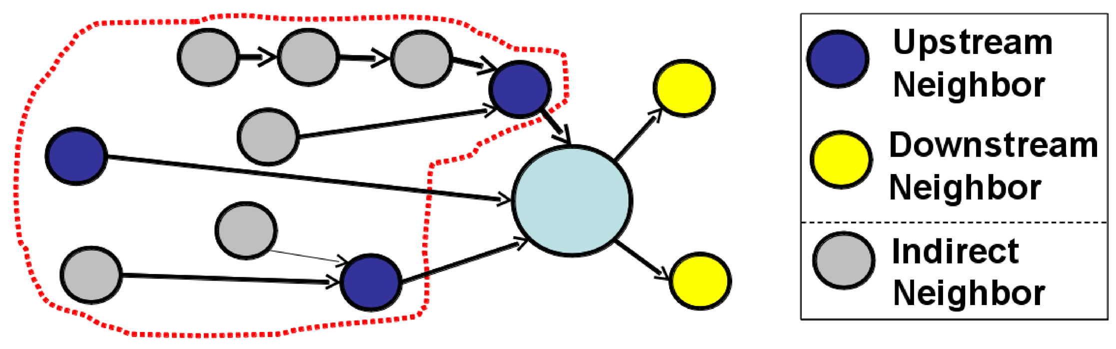
- Optimize in a *scheduling search space*



- Construct a schedule that optimizes movement of the currently approaching traffic in the local observation
- Worst-case complexity: $|I|^2 \cdot \prod_{i=1}^{|I|}(|C_{IF,i}|+1)$ state updates, where $|I|$ is the number of phases/inflows, and $|C_{IF,i}|$ is the number of jobs on inflow i
- Performs 2-4 orders of magnitude faster than COP [2]

Coodination in Network

- Consider non-local impacts in the network



Intuition: Predicted inflows (IFs) $\xrightarrow{schedule}$ Control flow (CF) $\Rightarrow$ Planned outflows (OFs) = Predicted non-local inflows for downstream neighbors

- Use an optimistic coordination protocol

- *Decentralized / Scalable*: each agent only communicates with its direct neighbors, but can incorporate non-local impacts from direct and indirect upstream neighbors
- *Optimistic*: each agent tries to follow its own schedule, minor schedule changes in neighbors can be absorbed

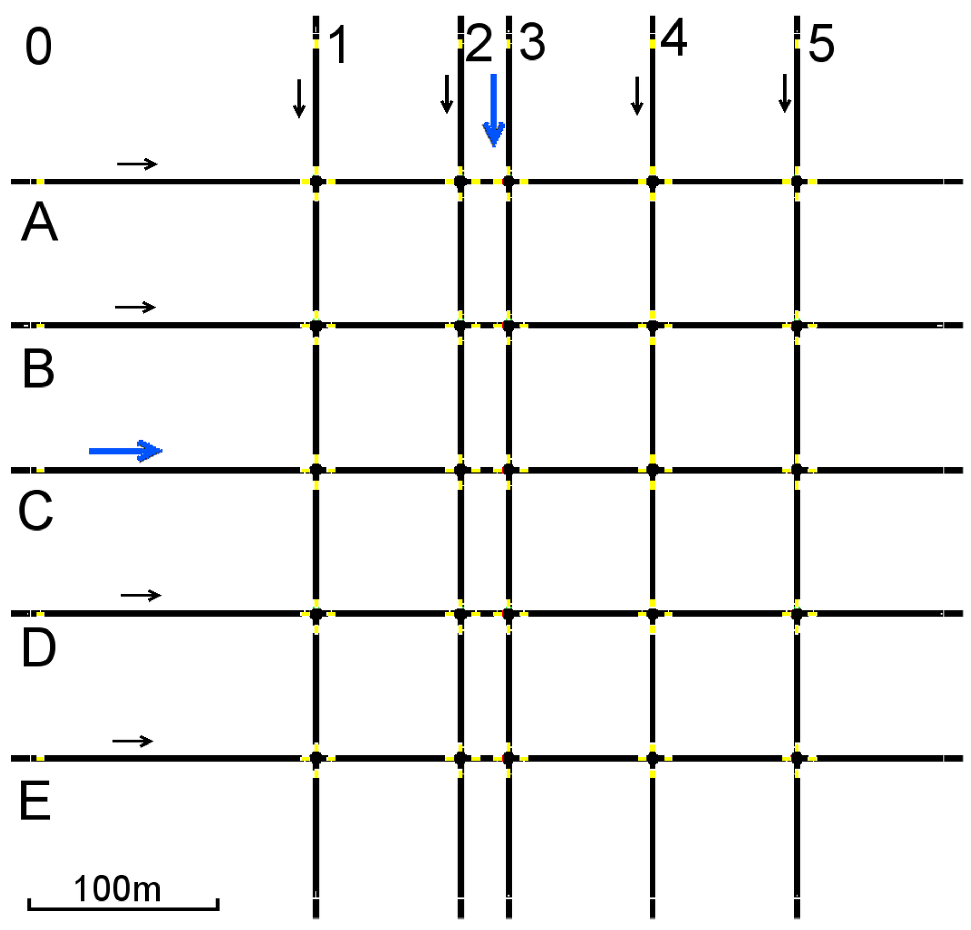
Algorithm 1 Obtain an optimistic non-local observation

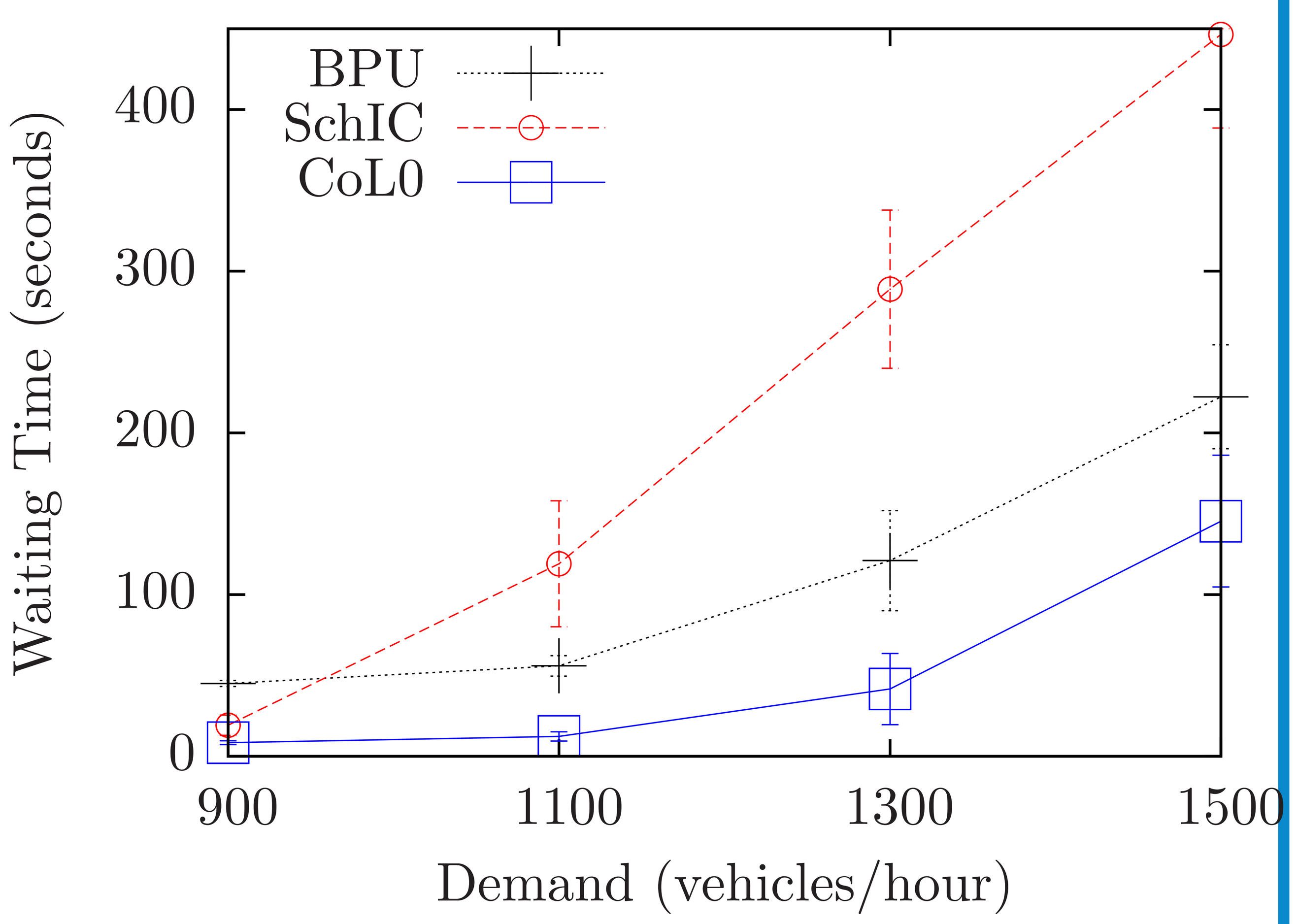
1: $m = GetEntryRoadByPhase(i)$ {For each phase i}
2: $UpAgent = GetUpstreamAgent(m)$
3: Request C_{OF} from $UpAgent$ using (cdt, m, H_{ext})
4: $Shift(C_{OF}$, the travel time on $m)$
5: Append C_{OF} into $C_{IF,i}$

Algorithm 2 Return C_{OF} for a message (cdt, n, H_{ext})

1: $(C_{OF}, S_{OF}) = (C^*_{CF}, S^*) \cap [cdt, cdt + H_{ext}]$
2: **for** $k = |C_{OF}|$ to 1 **do**
3: $|c_{OF,k}| = |c_{OF,k}| \cdot tp(s_{OF,k}, n)$ {turning proportion}
4: **end for**

Performance Evaluation





Simulation Settings

- Scenario: grid network with tightly-coupled intersections (with 2.5 or 7.5-second travel time on one edge)
- Dynamic demands on the bottleneck intersection C3

Control Strategies

- BPU: Balanced phase utilization [1] (offset calculation)
- SchIC: Schedule-driven intersection control [3]
- CoL0: SchIC + Optimistic non-local observation

- CoL0 produced lower waiting times than both other strategies. Comparison to SchIC demonstrates the added benefit of optimistic non-local observation. Furthermore, CoL0 outperforms BPU without requiring explicit offset calculation; coordination between neighbors is instead accomplished implicitly by looking ahead to upstream output flows.

Ongoing Work and Future Directions

- Pilot test: *Scalable urban traffic coordinator*

- Currently testing approach on a 9-intersection 2-way road network in the East Liberty area of Pittsburgh, PA, USA
- Real-world challenges: uncertainty, robustness to failures
- URL: `https://pilot.surtrac.net` (available soon)

- Advanced coordination mechanisms

- Pricing mechanisms to dampen any disruptive changes on schedules made by upstream agents (intersections)
- Negotiation mechanisms to reach for an equilibrium in an over-saturated traffic sub-network
- Dynamic learning of edge weights for critical flows