



A USDOT NATIONAL  
UNIVERSITY TRANSPORTATION CENTER

Carnegie Mellon University



---

## Optimizing Snow Plowing Operations in Urban Road Networks, Phase II

Stephen F. Smith (PI), <https://orcid.org/0000-0002-7053-3166>

Joris Kinable, Willem van Hove

FINAL RESEARCH REPORT

Contract # 69A3551747111

### DISCLAIMER

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the U.S. Department of Transportation's University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.

# Optimizing Snow Plowing Operations in Urban Road Networks

## Phase II Final Report

Stephen F. Smith (PI), Joris Kinable, Willem van Hove

June, 2019

### 1. Overview

In this report, we summarize work performed under Carnegie Mellon (CMU) University Transportation Center (UTC) Contract No. DTRT-13-GUTC-26 toward the development of a system for real-time dynamic optimization of municipal snow plowing operations. Building on our earlier work in this area, [Kinable 2016a, 2016b], this work has taken further steps in both in the design, analysis, and testing of scalable algorithms generating snow plow routing plans, and in the engineering and hardening of a in-vehicle device for generating turn-by-turn snow plowing instructions for a given snowing plan. With regard to snow plow routing, there were multiple accomplishments. First, the static snow plow algorithms developed in our earlier work were extended to incorporate additional types of real-world side constraints and further analyzed experimentally [Kinable 2019]. Second, a new, extended "late-acceptance" (LA) heuristic search procedure (referred to below as the CMU Planner) was developed to provide a fully scalable solution that also capable of addressing dynamic re-routing if problems arise during execution of the plan. In a comparative analysis of snow plow routing plans generated by the CMU Planner and those generated by the City of Pittsburgh's current system ( a product called RouteSmart) on City street data for a section of the Pittsburgh East End, the CMU planner was shown to generate plans that were 12% more efficient in terms of travel time while respecting all specified constraints. When configured to give priority to primary road segments over secondary road segments, the CMU planner continues to outperform RouteSmart with respect to total travel time, but completes clearing primary segments in less than one third of the time required by RouteSmart.

An enhanced in-vehicle navigation device for issuing turn-by-turn instructions to the snow plow driver was also developed, including an API for loading current City of Pittsburgh routes as well as those generated by the CMU Planner, and providing "skip" button and local rerouting capability to get vehicles back onto the planned route if the driver is forced to deviate. The in-vehicle navigation app was successfully pilot tested by a driver from the 3rd division of the City of Pittsburgh Department of Public Works (DPW) and feedback was positive.

Based on these collective results, a proposal was subsequently submitted to a City of Pittsburgh Request for Proposals (RFP) on "Telematics and Snow Plow Optimization" for

purposes of transitioning the developed technology into City operations and turning it into a commercial product.

Before describing the accomplishments in more detail, we first review the snow plow routing problem considered.

## **2. The Snow Plow Routing Problem**

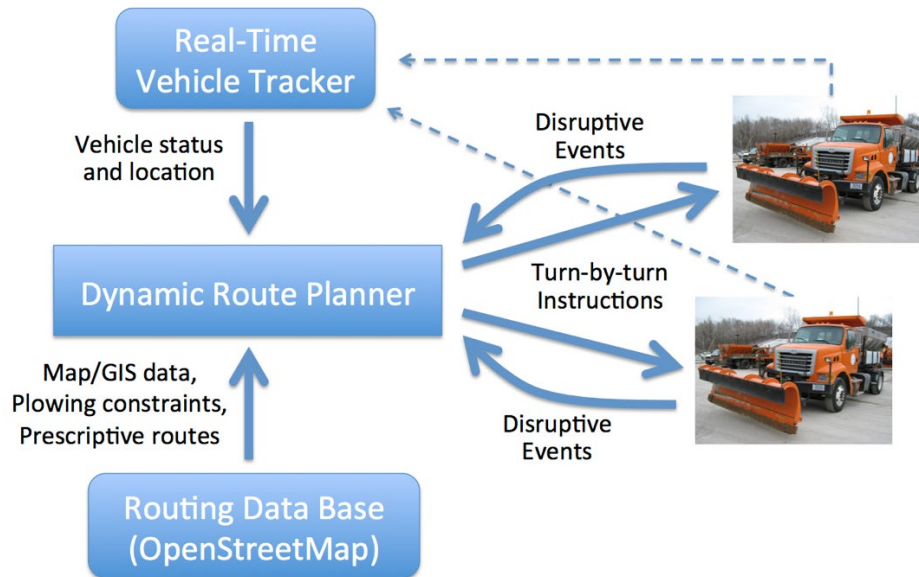
In cold weather cities like Pittsburgh, snowstorms can have a significant disruptive effect of on both mobility and safety, and consequently the faster that streets can be cleared the better. Yet in most cities (including Pittsburgh), static plans for snowplowing are developed using simple allocation schemes, e.g., streets are divided among vehicles based on geographic area, and each driver determines his/her individual route. These schemes are used because they are easy to implement and they work but the resulting snow plowing operation can be quite inefficient. In extreme cases (e.g., the Pittsburgh storm of February 5-6, 2010 where snow removal took several days to accomplish), this inefficiency can lead to protracted periods of hardship for urban residents and travelers.

The generation of efficient vehicle routes for snow removal is a challenging optimization problem, requiring consideration of constraints relating to resources (vehicle and crew availability; vehicle speed, range, home location), coverage topology (number of passes per road, one way roads, dead ends, refueling locations), snow-clearing priorities (main arteries before side streets) and refresh rates (depending on storm intensity). Variations of the problem have been formalized as the Chinese Postman problem and the Capacitated Arc Routing Problem, and both problems have been shown to be inherently difficult to solve optimally. Nevertheless, it is possible to produce near-optimal solutions with a variety of approximate search procedures (e.g., [Perrier 2008, Salazar 2012]), and these solutions can lead to substantial improvements in snowplowing performance. Centennial, Colorado, for example, recently reduced the time required to clear streets by 28-40% by focusing on optimization of routes [Sedlak 2013].

Any ability to produce efficient snowplowing routes, however, is susceptible to execution-time dynamics. Impassible road segments due to blocking vehicles that are stuck or to unaccounted for construction projects will mandate deviations from pre-planned routes, which can result in significant efficiency loss if left to driver response. Events such as snow plow breakdowns or shifts in storm intensity can similarly render pre-planned routes obsolete and require re-optimization. As has been recently suggested in [ITS 2014], even greater improvements to snow removal operations can be expected if snow plow route optimization is coupled with a real-time ability to re-route vehicles when circumstances warrant.

Thus, our broad technology objective has been to develop a dynamic snow plow routing system, capable of generating near-optimal plowing plans that fit current operational constraints and reactively maintaining them through execution as unexpected events force changes. The overall concept of operations for the envisioned system is summarized in

Figure 1. Prior to execution, relevant map data and plowing constraints are imported, along with current information of the availability and locations of snow plowing assets, and a storm specific plowing plan is generated. If available, a default plowing plan may also be imported and customized to fit the current operational constraints. Once the plowing plan is generated, routes are communicated to each vehicle and turn-by-turn instructions are presented to each driver through an onboard navigation app. In the event that unexpected problems are encountered during execution of a route, the disruptive event is communicated by to the dynamic planner, and the route is revised (in real-time) to circumvent or otherwise overcome the disruptive event.



**Figure 1:** Overall Concept of Operations

In the sections below, we summarize the progress made toward this goal over the course of this Phase II project.

### 3. Scalable Snow Plow Routing Algorithms

One principal thrust of the project was the further development of algorithms for generation of snow plow routes. To build understanding of the structure and complexity of the snow route planning problem, our prior work developed and compared a range of different state-of-the-art solution generation approaches, including a mixed-integer linear programming (MILP) model, a constraint programming (CP) model, and a constraint-based tree search procedure utilizing a late acceptance heuristic [Kinable et.al 2016a, 2016b]. The results obtained here confirmed basic intuitions about the scalability issues associated with some of these approaches, namely the MILP model and, to a lesser extent, the black-box CP model. But these initial models also ignored certain real-world resource constraints, such as fuel and salt consumption and replenishment, and to further understand how these constraints influenced scalability, extended formulations were developed and evaluated as an initial step toward developing a more viable solution to the dynamic snow plow planning problems. The results

obtained, which do not significantly change the conclusions reached in [Kinable et.al 2016b], are reported in [Kinable et.al 2019] (attached as an appendix to this report).

Both of these analyses pointed to the need for a heuristic solution to achieve a scalable approach, and the heuristic tree search method was found to provide a basis for finding a reasonably good solution fast. Accordingly, a variant of this procedure was integrated within a local improvement search framework to allow for scalable generation of efficient snow plow routing problems over time. A second benefit of utilizing a local improvement search framework is that the fact that it operates by modifying the existing solution to generate new better ones makes it equally appropriate for repairing a route that has been unexpectedly found to be impassable during execution (e.g., by an abandoned vehicle). In other words, the algorithm is equally suitable for *static* generation of snow plowing plans and *dynamic* revision of plans when they are discovered to be no longer feasible. We refer to this scalable dynamic route planning algorithm as the CMU route-planning algorithm.

### **3.1 The CMU Route-Planning Algorithm**

The CMU route-planning algorithm generates routes in an iterative manner. It accepts as input a set of street segments to be plowed (referred to as “plow jobs”), a larger set of street segments that can be used for moving from one plow job to the next if necessary, the set of vehicles that are available to handle plow jobs, along with their plowing constraints (e.g., capacity, initial location), and the location of other relevant resources (e.g., salt depots). Underlying map data associates turning information, speed limits, and other plowing constraints with input street segments. The plow jobs, street segment data and depots are used to build a plowing graph and a routing graph respectively (see Figures 4 and 5 below for an example of each), which, in turn, are used by the algorithm to calculate efficient, feasible routes. The algorithm begins by generating an initial feasible route, using a specific search heuristic that we have developed. Once an initial route has been generated, it and subsequently generated routes are repeatedly revised to produce new alternative routes. At each step, the quality of each new route that is generated is evaluated (e.g., by considering its projected travel time), and the search continues from the best routes found so far.

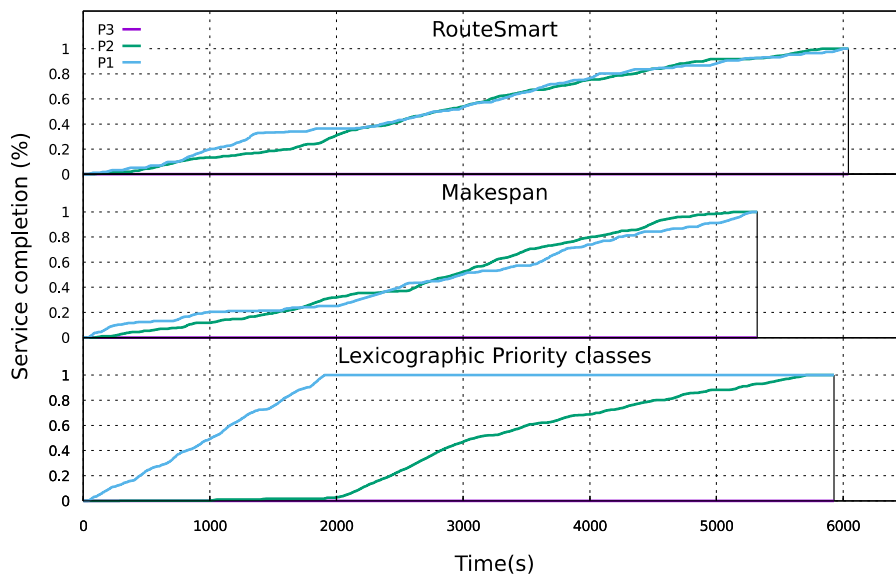
This iterative search puts the core route generation algorithm into a class of algorithms referred to as “anytime” algorithms – it generates a good feasible solution fast, and then continues to improve on that solution as you give it more time. If generating a route in advance of execution, then one can run the algorithm for an extended amount of time to produce a more optimized route. However, if less time is available for generating a route, then a less optimal, but still feasible route can be obtained. This makes the core route generation algorithm ideally suited to provide the dynamic planning capabilities that will be inevitably needed during execution of the plowing plan.

### **3.2 Experimental Comparison**

To calibrate the performance of the CMU Route Planner, we conducted an experimental comparison with the Routesmart system, which is currently used by the City of Pittsburgh to generate its routes. The comparative analysis was carried out using City of Pittsburgh

snowplow routing data for the neighborhood of Greenfield. The area in question has 629 street segment lanes (i.e., plow jobs) that have to be sequenced. These street segments were collected from one of the City's current snowplow routes. This set of street segments was then provided as input to the CMU route planner and the resulting snowplow route was compared to the current City route.

Figure 2 shows three plowing schedules, the top one produced by Route Smart and the bottom two variants generated by the CMU route planner (one that minimizes the total route duration while covering all segments (formally defined as 'makespan'), and one that prioritizes decisions to plow primary road segments before secondary roads, etc.). The graph depicts service over time, ranging from time 0 (start of plowing time) where 0% of the area has been plowed, to the end of the schedule where 100% of the area has been plowed. The different colored lines represent priority classes: P<sub>1</sub>=primary streets, P<sub>2</sub>=secondary streets, P<sub>3</sub>=tertiary streets. So for example, when instance line P<sub>1</sub> hits 100%, the inference is that at that time, 100% of the primary road segments have been plowed. The Greenfield pilot contains only primary and secondary roads, and no tertiary roads or lower categories. The black vertical lines in the graph mark the end of a schedule (makespan).



**Figure 2:** Percentage of service provide/completed over time

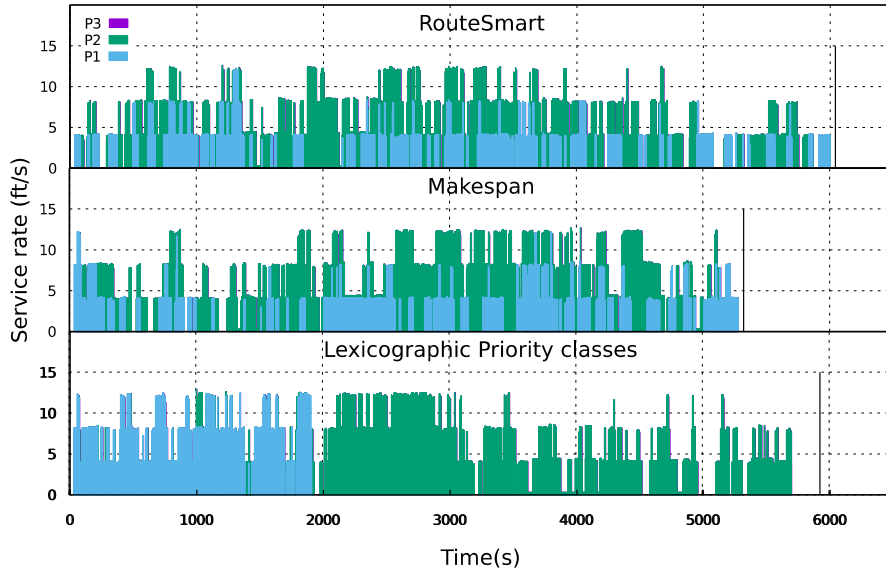
	Route Duration	Completion Time: Primary	Completion Time: Secondary	Total U-turns
RouteSmart	01:40:42h	6002s	5836s	18
CMU Planner: Make span Minimization	01:28:42h	5282s	5137s	17
CMU Planner: Lex (prioritized) Search	01:38:48h	1906s	5705s	17

**Table 1:** Comparative expected plowing completion times

Table 1 provides the numerical results for each of the three alternatives. A couple of observations can be made:

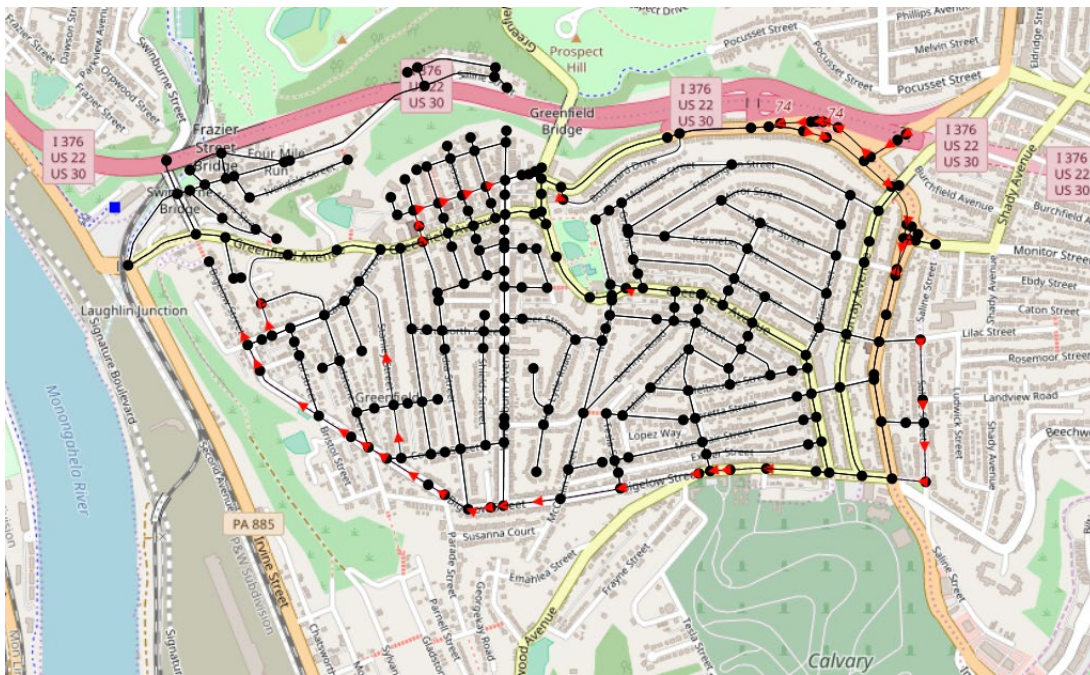
- Routesmart has the longest makespan, the most u-turns, and is overall the slowest in comparison to the routing plan that was provided to us by the City of Pittsburgh.
- Optimizing towards makespan clearly results in the shortest schedule. Interestingly however, if we optimize using Lexicographic (i.e., prioritized) search, we still end up with a shorter schedule than the RouteSmart schedule. The Lex search schedule takes about 10 minutes longer to complete than the makespan schedule, but the primary roads are cleared significantly faster (56 minutes).

The graph in Figure 3 gives a different view of the same data as the previous graph. Here the service rate is displayed, expressed in ft/s, aggregated over all routes. A service rate of 0 means that no vehicles in the schedule are plowing at that moment. For instance, at the end of the schedule, just before the makespan marker, the vehicles are done plowing and are returning to the depot, hence the service time at that time is 0. Notice that the bars are stacked (they are NOT behind each other). The height of the bar depends on whether all vehicles are plowing, and their speed. The colors again indicate the priority class: if the bar is blue, only primary roads are being plowed; if a bar is green, only secondary roads are being plowed. If a bar is both blue and green, at least one primary and one secondary road are simultaneously being plowed at that time. Note that the Lexicographic (i.e., prioritized) search, does not simplistically enforce that all primaries are plowed strictly before secondaries; instead, a more sophisticated approach is utilized where priority is given to the primary roads, but plowing secondary roads is allowed whenever this is advantageous to the overall schedule.



**Figure 3:** Plowing service rates

Finally, Figures 4 and 5 show the plowing graph and the routing graph respectively for the Greenfield neighborhood experiment. The plowing graph (Figure 4) contains all streets that require plowing. The blue square on the left is the 3rd division depot, where the salt is stored and the vehicles are located. The plowing graph is a subgraph of the routing graph (Figure 5) that represents the larger set of “deadhead” road segments that the vehicle can travel when moving from one plowing segment to the next.



**Figure 4:** The Plowing Graph for Greenfield



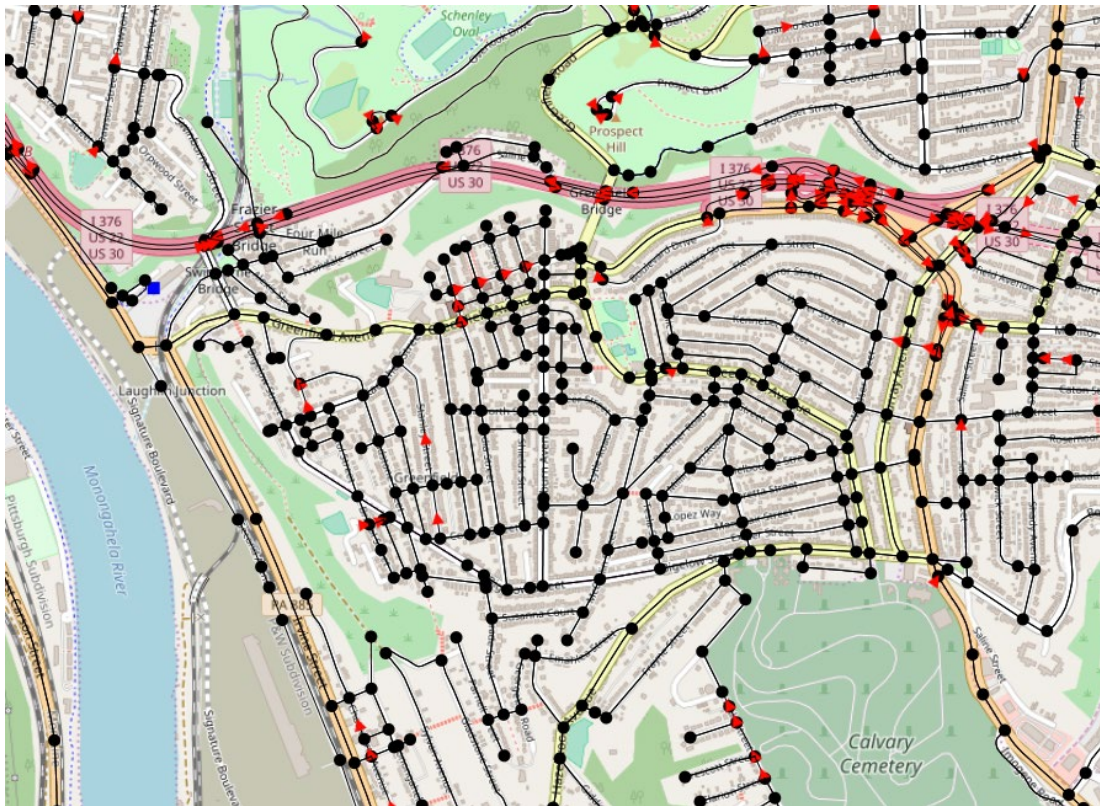


Figure 5: The Routing Graph for Greenfield

#### 4. In-Vehicle Navigation

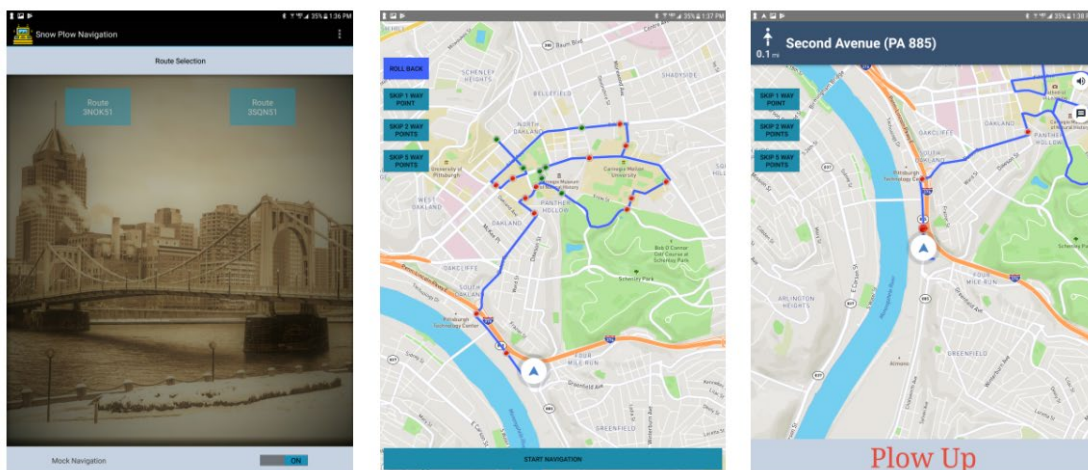
A second major thrust of the project was the further development of a mobile app that provides turn-by-turn instructions to the driver during execution of a previously planned snow plow route. Building on our previously work, the prototype described in [Kinable 2016a] was extended and re-engineered in several ways:

- *Switch to mapbox as a route navigation interface* – One significant limiting factor of the initial prototype was its use of *Scobler* as a route navigation interface. The fact that *Scobler* only allows a client to provide the display with one advance waypoint at a time led to weird user interaction behavior, where upon reaching each waypoint (i.e., each intersection) caused the app to announce that the route was completed before it was possible to reinitialize and feed in the next waypoint for display. The mobile app was re-engineered to instead use the *mapbox* navigation interface, which allows input of a sequence of up to 25 waypoints and minimizes this problem. (Google Maps provides a similar capability and in a production version this might be the way to go. There will be a charge.)
- *Skip button* – A skip button was introduced to allow the driver to indicate that the next road segment being announced in the plan cannot be executed (e.g. due to obstacles or

unacceptable conditions). In this case, the app will determine which alternative route the driver has taken and then issue a query to *mapbox* to reroute to the next road segment on the originally planned route. In a future version of the app, a call will be made back to a server running the dynamic planner described in Section 3.1 to compute a potentially better recovery plan (possibly even involving changes to other vehicle's routes).

- *Multiple modality communication* – The app was extended to provide both visual and auditory guidance to the driver. In point of fact, it is quite loud in the truck during plowing operations and visual communication will likely be the most effective medium. The basic visual display shows where the vehicle is on the map, when the next turn is, and whether the plow should be up or down for this road segment.

Figure 6 shows a few visualizations from the current in-vehicle app. It is designed to run on an IOS device, specifically an IPAD. This decision was made based on preferences indicated to us by the City of Pittsburgh. The app could straightforwardly be ported to an Android device.



**Figure 6:** The In-Vehicle Navigation App

## Conclusions

This research has focused on developing snow plow route planning and execution technology that can enable cold weather cities to better cope with the disruptions of winter storms. Key to our focus and approach is (1) the notion that generated plowing plans should match current storm conditions and resource availability (instead of assuming a single pre-generated plan based on expected conditions), (2) the expectation that plans can be dynamically revised in real time in response to unexpected circumstances that are encountered during execution. Two complementary technologies have been developed that promote these key ideas of offer the potential for better logistical solutions to snow removal in urban environments. The first is a scalable snow plow route planning algorithm capable of quickly finding a good quality snow

plowing plan for a given storm situation and set of available plowing resources, and then improving that solution if more computing time is available before it is time to execute. Given the nature of this algorithm, it is also equally suited for revising a plowing plan in real-time during execution if unexpected circumstances arise. The second technology developed over the course of this research is an in-vehicle mobile app that is capable of accepting a previously generated plan and using it to provide turn-by-turn instructions to a snow plow vehicle driver during execution. Based on collective results obtained with both of these technologies, a proposal has been submitted in response to a City of Pittsburgh RFP for transition of these technologies into City operations and commercialization for subsequent transfer to other cities.

## References

- [ITS 2014] "Solving the Snow Plough Conundrum", *ITS International*, Jul/Aug 2014.
- [Kinable 2016a] Kinable, J., S.F. Smith and W.J. van Hoeve, "Optimizing Snow Plowing Operations in Urban Road Networks: 2015 Final Report", CMU UTC Technical Report, January 2016.
- [Kinable 2016b] Kinable, J., W.J. van Hoeve and S.F. Smith, "Optimization Models for a Real-World Snow Plow Routing Problem", *Proceedings 13th International Conference on Integration of Artificial Intelligence and Operations Research Techniques in Constraint Programming (CPAIOR 2016)*, Banff Canada, May 2016.
- [Kinable 2019] Kinable, J., W.J. van Hoeve and S.F. Smith, "Snow Plow Route Optimization: A Constraint Programming Approach", Unpublished document, June 2019. (submitted for publication)
- [Salazar 2012] Salazar-Aguilar, M.A., A. Langevin and G. Laporte, "Synchronized Arc Routing for Snow Plowing Operations", *Computers and Operations Research*, 39(7), July 2012.
- [Perrier 2008] Perrier, N., A. Langevin, and C-A Amaya, "Vehicle Routing for Urban Snow Plowing Operations", *Transportation Science*, 42(1):44-56, Feb 2008.
- [Sedlak, 2013] Sedlak, M. "Plot a Better Plowing Plan", *Public Works Magazine*, July 2013, <http://www.pwmag.com/mobile-technology/plot-a-better-plowing-plan.aspx>