Anticipatory Dynamic Traffic Sensor Location Problems with Connected Vehicle Technologies

Hyoshin (John) Park, Ali Haghani, Song Gao, Michael A. Knodler, Siby Samuel

Abstract. Technological advancements in wireless systems enable the advancement of traffic operations. When onboard equipment in connected vehicles transmits data to roadside sensors integrated with a traffic signal controller, green phases are redistributed to directly and indirectly reduce network delays. Despite the potential benefits of sensor technologies, the challenges associated with identifying optimal sensor locations for multiple time stages throughout a day with uncertain demand patterns has received little attention. In this paper, we focus on proactively reducing the network delay by controlling traffic signals through an optimized sensor deployment. The framework is based on portable sensors that may be repositioned within day and day to day to new locations such that delay savings over multiple time stages will be maximized. To tackle this multiperiod stochastic problem, dynamic models are proposed, considering the future sensor locations given budget constraints on the sensor costs and relocation costs, and the effect of control is tested on various demand profiles and penetration rates of an urban transportation network. A subproblem decomposed by Lagrangian relaxation enhanced with valid cuts has a better bound, and a variable neighborhood search algorithm quickly finds solutions. Two dynamic models that constrain flexible and restricted relocation, respectively, present higher savings compared to the stationary model without sensor relocation. The flexible relocation model guarantees higher savings than the restricted model by achieving the same maximum savings with fewer sensors. The gap between two dynamic models decreases when more sensors are available.

Keywords: sensor location problem • connected vehicle • signal control • multistage stochastic optimization • dynamic programming • Lagrangian relaxation

1. Introduction

Wireless connectivity among vehicles, infrastructure, and mobile devices has brought about innovative solutions to improve the safety and mobility of transportation systems. One of the more promising solutions has been the deployment of Bluetooth sensors anonymously using the machine access control address of a cell phone without privacy concerns (Haghani et al. 2010). On arterials, the full potential of Bluetooth technology is hindered by latency issues, and market penetration has been relatively flat because individuals must enable discovery mode on their mobile phones for the technology to work. An alternative solution is to use connected vehicle (CV) technology, which is expected to have a steep increase in market penetration given current levels of governmental support. Recent developments in dedicated short-range communications (DSRC) have enabled the onboard equipment (OBE) of a vehicle to interact with the OBE in other vehicles and transmit the information to roadside sensors. Overcoming the limitation of loop detectors (Ramezani and Geroliminis 2015), which inconsistently estimate incoming volumes, we employ CV technology in early queue detection and redistribute green phases to reduce network delay.

Despite the significant potential benefits of optimizing sensor locations, the challenges associated with identifying optimal sensor locations for multiple time stages throughout a day with uncertain demand patterns have received little attention (Park and Haghani 2015). The CV test bed has expanded beyond Michigan into California (San Mateo and Pasadena), Florida,
New York, Texas (Dallas), and Arizona (Phoenix), which will serve as models for future permanent deployments. The next generation of CV test bed is progressing toward national deployment, focusing on active transportation and demand management and dynamic mobility applications (Yelchuru et al. 2016). There is a specific need to optimize the location of roadside sensors under budget. Specifically, it is essential to decide where best to locate sensors to maximize the benefit of CV deployment.

To date, traditional studies related to sensor positioning have focused on locating sensors to enhance the quality of traffic origin–destination (OD) demand (Zhou and List 2010) or travel time (Mirchandani, Gentili, and He 2009) estimations. These permanently installed sensors may produce meaningful information for traffic management, but are constrained by their lack of portability. Consider, for example, a sensor that may provide useful data during the morning rush hour, but may produce meaningless information in the afternoon with different traffic patterns. Given practical limitations associated with resources, the traffic operator cannot install costly sensors (Park and Malakorn 2013) at every congested location. Uncertainty in traffic congestion has made the problem more difficult, and in practice, sensors are often located at locations with high likelihood of recurrent congestion during peak or off-peak periods (Haghani et al. 2013). More recently, Park and Haghani (2015) proposed a stochastic portable sensor location model with scenario-based sensor arrangement solutions to account for highly varying spatial congestion patterns throughout a typical day. As portable sensors become more viable due to advanced DSRC, smaller form factors, and streamlined initializations, exciting opportunities for sensor relocation are increasingly possible.

Given the emergence of this potential application, we consider the sensor location and relocation criteria. Specifically, we locate sensors to maximize expected delay savings as a difference between before and after deployment, given a budget resource constraint. The benefit of CV technology is identified as expected delay savings under model anticipatory control, as demonstrated previously in the literature (Geroliminis, Haddad, and Ramezani 2013; Venkatanarayana et al. 2011). As traffic condition is closer to saturated condition, coordination or queue spillback prevention has higher rewards and requires lower market penetration rates. With CV technology, a sensor on an upstream intersection manages a signal based on the early detection of directional downstream queues. A set of candidate locations are identified to maximize the delay savings on the network.

To decide where to locate sensors, we have also considered how to connect these different locations in several periods throughout the day by relocating sensors. For each period, the traffic operator must decide whether a sensor should remain in the same location to avoid any relocation costs, despite the notion that it may be more efficient if it were to be relocated to a new location. This decision strongly depends on the trade-off between the traffic delay savings obtained by relocating sensors and the relocation costs. To justify a change in location, the corresponding efficiency gains need to compensate for the resultant relocation costs. Compared with sensor deployment, sensor relocation has additional constraints. First, sensor relocation may have a strict response time requirement that is time dependent. For example, if a queue spillback is predicted at a certain location in a current time stage, another sensor should move to detect the queue as early as possible. Furthermore, queues develop and dissipate in different manners during different time stages. Second, sensors cannot collect data while in transition from one location to another, and this loss should be considered in relocation decisions. Since the time to relocate a sensor depends on the time of the day, it may be more beneficial to continue to leave a sensor at its current location and wait for next stage decision. In this paper, we demonstrate that some scenarios provide a better solution by relocating sensors to new locations when relocation cost is lower with the same level of benefit. As traffic authorities deploy CV technologies to more congested urban cities, there will be even greater benefits of sensor relocation.

The issues noted previously motivated this study to extend the previous stochastic model (Park and Haghani 2015) to a multiperiod stochastic dynamic model. This study proactively reduces the network delay by controlling traffic signals through an optimized sensor deployment, given budget constraints such as sensor costs and relocation costs. The framework is based on portable sensors that may be repositioned within the day to new locations such that delay savings over multiple time stages will be maximized. The primary decision for which we have demonstrated a solution to better realize network optimization is the sensor deployment in the current stage, given the potential for relocation in future stages. The optimal solution considers the relocation frequency and benefits in the future stages, based on expected future traffic conditions, market penetration rates, and travel demand patterns. We assume that historical and current data are available for predicting the expected value of future information.

The main contributions of this study compared to tradition sensor location studies are highlighted in Section 2. After description of the anticipatory microsimulation signal control with CV queue detection, multiperiod stochastic models are introduced in Section 3, with solution algorithms given in Section 4. The models are demonstrated on a portion of the Burlington, Vermont, arterial network in Section 5, and applied considering different penetration rates, demand profiles, and feasibility in Section 6. Conclusions and
future research directions are presented in Section 7. The formulations of signal control, a metamodel, and submodularity are presented in Appendices A to D.

2. Sensor Location and Relocation Problem

The sensor location problem (SLP) involves selection of certain arcs or nodes for the sensors. The existing problems differ according to different types of sensors and measurement of interests. Table 1 describes the sensor type problem depending on the traditional detection technologies (e.g., loop, image, fixed vehicle identification (ID) and more recent technologies (e.g., portable vehicle and path ID) (Gentili and Mirchandani 2012). Based on the capability of a sensor, traffic measurements that have been used in SLP studies are (1) OD flow observability, (2) OD flow estimation, (3) travel time, and (4) signal control.

The first problem, OD flow observability inspired by covering location models (Daskin 1983), provides full or partial flow observability for the sensor coefficient matrix (Ashok and Ben-Akiva 2000; Bianco, Confessore, and Reverberi 2001). There are two types of OD observation problems: full flow-observability problems, with counting sensors located on links to observe either OD trips or route/link flows, or located on nodes and known split ratios, and partial flow-observability problems with path ID sensors located on links to observe route flows or vehicle ID sensors located on the links of the network (Bianco, Confessore, and Gentili 2006; Gentili and Mirchandani 2005; Ng 2012). Recently, new concepts were successfully implemented using a simple and intuitive metric to assess partial observability solutions (Viti et al. 2014) and in a nonplanar hole-generated network (Castillo et al. 2014).

The second problem, the OD estimation problem, estimates the flow without full rank to overcome the underestimation (Yang, Iida, and Sasaki 1991; Yang and Zhou 1998). Decades of researches have focused on OD estimation problems including total demand scale (Bierlaire 2002), or approaches using counting sensors and path flows with point-to-point sensors (Castillo, Menéndez, and Jiménez 2008; Hadavi and Shafahi 2016; Xing, Zhou, and Taylor 2013; Zhou and List 2010).

The third problem tries to find different sensor location layouts to minimize the traffic measurement errors (Larson 1978) such as density and flow (Liu and Danczyk 2009). Optimization models were proposed to determine the optimal placement of automatic vehicle identification (AVI) readers for travel time estimation with maximum benefit (Sherali, Desai, and Rakha 2006). Greedy heuristics were proposed to maximize the total vehicle miles monitored, and to minimize the variance of predicted travel times.
A multiobjective optimization problem was formulated to locate sensors for reliable traffic data (Asudegi and Haghani 2013). A path travel time uncertainty criterion was selected to construct a joint sensor location and travel time estimation/prediction framework with unified modeling of both recurring and nonrecurring traffic conditions (Xing, Zhou, and Taylor 2013).

The fourth problem deals with adaptive traffic control to estimate incoming volumes and queue blocking probability using traditional sensors. However, these problems have concentrated on traffic volume coverage with maximum information gain at permanent locations. An extensive modeling effort is required to quantify the uncertainty (Deng, Lei, and Zhou 2013). Traditional detectors have several disadvantages. For example, often, up to half of detectors are malfunctioning (Sunkari et al. 2010), and advanced algorithms have been developed to overcome measurement errors of inductance loop detectors (single and dual; Rakha and Arafeh 2010). Making adjustments to loop detectors and video detection to provide the level of detection needed to be fully adaptive to real-time traffic is oftentimes inaccurate, expensive, and unreliable, and they can be limited in physical range (Goodall, Smith, and Park 2013). The AVI method has a privacy concern. If the traffic demand pattern is regular, the benefit of local signal actuation is minimal.

As an alternative, Bluetooth sensors were used in traffic signal coordination (Park and Haghani 2014). By placing sensors along roads, tracking Bluetooth devices in passing vehicles, the solution was able to accurately detect and record how long it took a car to drive along a corridor, segment by segment. Measured travel time data were used as input for an optimal bandwidth progression algorithm. Compared to the traditional method, depending on the point speed at their fixed locations, Bluetooth technology provided point-to-point travel time over the segments. According to bandwidth efficiency and attainability, the signal timing yielded lower delays than the current signal planning. The optimal solutions for freeway sensor number and locations were provided by a two-stage stochastic programming model (Park and Haghani 2015). Park and Haghani (2015) considered uncertainty of spatial and temporal correlations based on scenarios generated by principal component and clustering analysis. However, for arterial signal control, this point-to-point detection-based Bluetooth technology still has latency issues.

In this study, we fill these gaps in the following ways (Figure 1). First, we proactively reduce the network delay through optimized deployments of CV roadside equipment (RSE) integrated with a traffic signal controller to reallocate the green time of downstream links with evolving queues toward other phases with shorter queues. A metamodel is developed based on

![Figure 1. Sensor Location Problem](image-url)
simulation driven parameters on a real city. Different penetration rates associated with delay savings are tested in deciding where to locate sensors.

Second, we propose a multiperiod stochastic dynamic model that builds on the stationary model of Park and Haghani (2015). With limited relocations, our dynamic model provides benefits equivalent to a greater number of sensors than in a stationary model. By relaxing constraints of the dynamic model, further savings are secured with flexible relocations to nearby coordinated intersections. The benefit of relocation is highlighted with a higher rate of diminishing savings with an additional sensor by achieving the same maximum savings with a fewer number of sensors.

Third, we employ heuristic algorithms to solve the proposed combinatorial optimization problem. Lagrangian relaxation decomposes the problem into two subproblems. Within feasible solutions provided by the first subproblem (relocation problem), the second subproblem (location problem) is solved until the best bound is found. A cutting plane method adds a valid cut to the subgradient algorithm with the better bound. To remedy the convergence problem in the subgradient algorithm, the location problem is solved with a variable neighborhood search (VNS) method.

3. Methodology

We start with the problem setting, simulation-driven signal control strategy, and then proceed with the SLP formulation.

3.1. Problem Setting

Previous studies have been reluctant to relocate sensors because sensors are fixed below the road surface. With the proliferation of advanced technologies, portable sensors are now available that offer much more flexibility. We integrate signal control into the sensor location problem with two assumptions.

First, we assume that the CV market penetration rate in the test-bed city is high enough to correctly estimate the occurrence of queues, and we focus on the problem of where to control signal by locating sensors in Section 5. We assume that queue detection is based on the penetration rate that influences the likelihood of the CV being the last vehicle in the queue (Christofa, Argote, and Skabardonis 2013). The effect of different penetration rates on delay savings is also explored in Section 6.

Second, we assume that the proposed model is applied in discrete time stages of operations. Based on data from the Chittenden County Regional Planning Commission, a 24-hour day is divided into multiple time stages, with distinctively different traffic patterns and different demand across 12:00 to 6:00 A.M., 6:00 to 9:00 A.M., 9:00 A.M. to 4:00 P.M., 4:00 to 7:00 P.M., and 7:00 P.M. to 12:00 A.M. Updated real-time traffic data from the roadway detectors is sent to the simulation model. Once the transportation authority believes that the signal control is not appropriate, we can make another round of decisions based on the proposed model with updated demand distributions.

Table 2 contains the notation, including sets, superscripts, subscripts, parameters, variables, and functions.

3.2. Simulation-Driven Signal Control Strategy

Among several signal control approaches (e.g., reverse offsets), this study uses a green time allocation approach to maximize the benefit of CV technology. This
The allocation model is a function of vehicle arrival on all approaches to links of an intersection to minimize the negative effect of the new signal strategy. To estimate the impact of signal changes in a set of intersections, a metamodel is developed based on simulation-driven parameters on a real city. Different penetration rates associated with delay savings are tested in deciding where to locate sensors.

The simulation-driven model can minimize the resources spent while maximizing the information obtained in a simulation experiment. Most probabilistic network models are simulation based (Osorio and Bierlaire 2013). The most common choice for a functional metamodel is the use of low-order polynomials (e.g., linear or quadratic). The metamodel is a deterministic function that is much less expensive to evaluate.

We analytically approximate the impact of the proposed signal strategy on the network delay by using a metamodel (or surrogate model). We assume that different combinations of sensor locations will lead to different delay savings. The proposed signal strategy is triggered whenever the downstream queue is more than a predefined threshold. The computed delay savings are used in updating the parameters of the metamodel, which has interaction terms among explanatory variables to capture signal coordination across consecutive intersections. The metamodel is fitted based on a set of simulated observations from a microscopic traffic simulator, TransModeler, that takes the proposed control strategy as an input and the network delay savings as an output. We assume that traffic equilibrium is achieved by TransModeler, which models the dynamic route choices of drivers based on simulated time-dependent travel times (Caliper 2015). The fitted metamodel is used in sensor location problem. The detail formulation of the metamodel is presented in Appendix B.

Literature has used fixed-pavement loop detectors to detect the last equipped vehicle in a queue (Christofa, Argote, and Skabardonis 2013). Let $\phi'$ be a control policy with loop detectors, and the corresponding network travel time at time period $t \in T$ is denoted by $\psi(\phi')$. With a new control policy with CV technology installed on location vector $x'$ at time period $t$, we can obtain a reduced network travel time $\psi(x')$. We install CV technology only at locations where significant reduction in network travel time is expected. The objective is to maximize the difference between $\phi'$ (old) and $x'$ (new) control strategies. Since network travel times with the old policy $\psi(\phi)$ are different across time periods $t \in T$ (i.e., $\psi(\phi') \neq \psi(\phi^2) \neq \ldots \neq \psi(\phi^i) \neq \ldots \neq \psi(\phi^o)$), we cannot simply make $\psi(x')$ as a minimization problem. A delay savings in network travel time, $\psi(x')$, is calculated as $\psi(\phi') - \psi(x')$.

A set of candidate locations at $x$ will have queue detection–enabled sensors. This will lead to an optimal
3.3. Anticipatory Dynamic SLP with Flexible Relocations

Here, we introduce a look-ahead model that can capture better solutions with anticipatory representation of decisions in the future. The multiperiod stochastic problem is solved in the framework of the dynamic program, considering the future sensor locations given budget constraints on the sensor costs and relocation costs. We start with sensor relocations in this section, and then relax the anticipatory assumption to obtain an approximate solutions in Section 3.4. A stationary model without relocation is presented in the Section 3.5.

In previous studies, the deterministic sensor location strategy could work for a specific pattern during peak hours (e.g., 6:00 a.m.–9:00 a.m.). However, a single value in the deterministic model would not accommodate the uncertainties in demand, and would overestimate or underestimate the value in a real scenario (Park and Haghani 2015). If a location is expected to have a below-average queue, then no sensor will be installed as a result of the deterministic strategy. However, because of the variable nature of the traffic flow, there could still be frequent long queues at this location and the lack of sensor at those times could lead to an inefficiency. Inability of the model to handle uncertainty in the future introduces significant weaknesses.

To remedy this issue, the stochastic location model developed herein builds on an existing scenario-based stochastic model (Park and Haghani 2015). This two-stage stochastic SLP with recourse is extended to multiple time periods to explicitly use predicted information to make a decision that will be implemented in the future. The multiperiod stochastic SLP incorporates uncertainty in delay savings throughout a day, estimated by our signal control strategy.

Let \( G(N, A) \) be a stochastic time-dependent network, where \( N \) is a set of nodes \( i \) and \( A \) is a set of links \( a \). We define \( x_i^t \) (\( i \in N \)) as a binary variable equal to 1 if a sensor is located on node \( i \) in time period \( t \in T \) and 0 otherwise. Let \( x' = [x_{i}^{t+1}, \ldots, x_{i}^{T}] \) be a particular location vector at node \( i \in N \) at stage 1. After actual realization of demand \( d \in D \) in current period \( t \), demand and traffic condition in the future periods \( t + 1, \ldots, T \) are predicted to make a more accurate decision, driven by the random process \( \xi_{(d)} \). The expectation of delay savings for a certain period \( t \), \( \mathbb{E}[\psi(x', \xi)] \), is taken with respect to the random vector whose probability distribution is assumed to be known, and a particular realization of demand is denoted by \( \xi^t \).

Even though we considered demand variation (DV) for a certain time period, \( \xi^t \) also changes with demand scenarios at different times of a day \( t \in T \) (\( \xi^t = \xi^1, \xi^2, \ldots, \xi^T \)). Additional sensors are required to meet the demand realization occurring sequentially during the time of day. Without increasing the total budget, one way to solve this multistage stochastic SLP is to sequentially change the configuration of the sensor network on the nodes in each time period. In this multistage stochastic problem, we try to find a sequence of decisions of relocating sensors. Although researchers have generally assumed that all sensors are placed at the same time, it is critical to respond to future traffic conditions that evolve over time. As shown in Park, Shafahi, and Haghani (2016), consideration of future relocation decisions in the current location decisions produces significant benefits in the solutions. For example, an occurrence of nonrecurring congestion may change the severity of traffic conditions afterward until the end of the day.

Let \( S^i(x_{(i)}^{t}, \xi^t) \) be the state variable at time period \( t \in T \) that depends on the given sensor locations at \( t - 1 \) and demand realization at \( t \). Given start of any period \( t \), the state summarizes all past information that is needed for the look-ahead optimization problem. The decision vector \( x_{(i)}^{t-1} \) is the action that chooses sensor location vectors at previous period \( t - 1 \). The dynamic programming problem yields the optimal policy mapping states to actions. \( \mu: (x_{(i)}^{t-1}, \xi^t) \rightarrow x_i^t \), for all possible \( t \) and \( S^i \).

Let \( y_{(j)}(l, l \in N) \) be a binary variable equal to 1 if there was a relocation from location \( j \) at time \( t1 \) to location \( l \) at time \( t + 1 \) and 0 otherwise. We introduce a relocation matrix \( y' = [y_{(j)}^{1}(l), y_{(j)}^{2}(l), \ldots, y_{(j)}^{T}(l)] \) from the current location \( x_{(j)} \) to the next location \( x_{(l)} \). We can replace the row vector \( y_{(i)}(l) \) by the row vector \( y' \) and the matrix \( y' \) as \( x' \times y_{(j)}^{T+1} = x_{(i)}^{T+1} \). For example, the problem of relocating a sensor from location \( (j = 1) \) at time \( t \) to location \( (l = 2) \) at \( t + 1 \) can be expressed as follows:

\[
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1
\end{bmatrix} \times
\begin{bmatrix}
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1
\end{bmatrix} =
\begin{bmatrix}
0 & 1 & 1 & 0 & 1
\end{bmatrix}.
\] (1)

The data loss from repositioning sensors depends on the time-dependent travel time matrix: \( \pi_{(i)(j)}(x', \xi^t) \). The shortest path algorithm is used to find the value \( \pi_{(i)(j)}(x', \xi^t) \). We introduce two types of constraints on the sensor location and relocation. Let \( c \) be a maximum number of available sensors. We impose a general budget constraint (Zhou and List 2010) for each time period as well as the total time period. Let \( b_{(l)} \) be the budget allocated for period \( t \in T \) in relocation, which is either spent or carried over to the next period. Denote
by $b^{(2)}_i$, the total relocation budget available at period $t \in \mathcal{T}$. Since each sensor location decision vector has a different relocation cost, relaxing the maximum budget may produce more feasible solutions (e.g., similar to the maximum distance constraints; Berger, Coullard, and Daskin 2007).

In addition to the advantage of relocation over the stationary model, a more flexible relocation has two additional advantages. First, with other sensors fixed in their locations, relocating one sensor to nearby intersections several times may save relocation cost. Second, with similar benefit, a sensor may move to a location where a significant queue is expected in an earlier stage in a nonpeak hour so that relocation cost is reduced. Let $b^{(3)}_i$ be the relocation frequency for one day operation $T$. In general, to monitor time-dependent traffic congestions during a day or over the week, more sensors are required than the optimal number. Therefore, more relocations will secure more delay savings. By fixing the maximum frequency and testing feasible numbers, we can reach the maximum solution faster. As shown in Zhou and List (2010), general stochastic sensor location problem has a bounded number of relocations per time period. In this paper, for the multiperiod relocation, we added one more budget constraint so that we can save limited relocation for a stage in which more delay savings are expected.

For any realizations of the random elements of $\xi^t$ that become known at stage $t$, the formulation takes the form of the multiperiod stochastic dynamic programming (MSDP) problem. The multiperiod stochastic SLP with uncertain demand is defined as a dynamic SLP, which we denote by MSDP1:

$$\text{MSDP1: } \max_{\mu} \mathbb{E}_{\xi^1, \xi^2, \ldots, \xi^T} \left[ \sum_{t=1}^{T} \psi^t(x^t, \xi^t) \right]$$

subject to:

$$(2) \quad \begin{align*} x^t &= \phi(\xi^t); \\
&= \mu(x^{t-1}, \xi^t) \quad \forall 2 \leq t, t \in \mathcal{T}; \\
\sum_{i \in N} x^t_i &= c \quad \forall t \in \mathcal{T}; \end{align*}$$

$$(3) \quad \begin{align*} \sum_{t \in \mathcal{T}} |y^t_i| &= b^{(1)}_i \quad \forall t \in \mathcal{T}, j = l; \\
b^{(2)}_i &= b^{(1)}_i - \sum_{j=1}^{N} y^t_j \quad \forall t \in \mathcal{T}; \end{align*}$$

$$(5) \quad \begin{align*} x^t_i &= \sum_{j=1}^{N} y^t_j \quad \forall i; \\
\sum_{i \in \mathcal{N}} x^t_i &= b^{(3)}; \end{align*}$$

$$(8) \quad x^t \in \{0, 1\} \quad y^t \in \{0, 1\} \quad \forall t.$$

The objective function represents maximizing the delay savings by installing sensors on optimal locations with feasible relocations. Constraints (2) enforce that the decision vector in the first period depends only on the demand realization, but in later periods depends on both past decisions and the demand realization. Constraints (3) ensure that the number of sensors is under budget limit regardless of time. Constraints (4) ensure that no more than accumulated budget shall be used in sensor deployments, while constraints (5) pass the unused savings to the subsequent stage. Constraints (6) ensure a sensor can be located only when a sensor was relocated in that location in the previous stage. Constraints (7) enforce the total number of relocations in all time periods. Constraints (8) define the decision variables as binary.

Note that $\xi^t \in \xi$ are data vector elements that can be random. The recourse takes the form of

$$\psi^t(x^t, \xi^t) = \max_{\phi \in \phi} \sum_{t=1}^{N} a^t_{(i)} \phi(x^t_i) + \beta^t_{(i)} \phi(x^t_i) \phi(x^t_i);$$

the detailed network delay formulation is presented in Appendix B.

We deal with the complexity of the problem by using Lagrangian relaxation and a variable neighborhood search algorithm in Section 4. However, in practice, high labor fee and data loss may restrict the frequency of relocation. Since CV data are expensive, losing some data elements over a few relocations may significantly lower the quality of data collection more than we expect. The transportation authority may want to obtain the solution faster while losing some delay savings, especially when delay savings are relatively lower because of low penetration rate. In the next section, we restrict the relocation frequency, while future relocation decisions are still dependent on the current decision.

### 3.4. Anticipatory Dynamic SLP with Restricted Relocations

Until the increase in the penetration rate reaches a certain point, the transportation authority may be reluctant to relocate sensors because the rewards are relatively lower. By restricting the relocation frequency to once per sensor, we can have a partially anticipatory assumption. In this restricted look-ahead problem setting, once one sensor is relocated, no more relocation can occur to that sensor. The formulation is simplified by assuming that there is no linkage between demand realizations and location decisions between some time periods. The independence assumption enables us to rewrite the multistage stochastic program as a large two-stage stochastic program. This assumption greatly reduces the complexity of the problem, which has the benefit of allowing the transportation authority to solve much larger and more realistic instances.

We introduce a new auxiliary variable, $z^t_{(i)} \in \{0, 1\}$, equal to 1 if node $i$ has a new sensor installed, $-1$ if a sensor at node $i$ is relocated to another location, and 0 if
there is no relocation. The vector difference of location is expressed as the sum of relocation variables $y^{t}_{(j,l)}(j, l \in N)$ that is equal to 1 if there is a relocation from location $j$ at time $t$ to location $l$ at time $t + 1$:

$$\sum_{j} y^{t}_{(j,l)} - \sum_{l} y^{t}_{(j,l)} = z^{t}_{(i)} \quad \forall i \in N, \forall t \in T. \quad (9)$$

We enforce that no more sensor removal can occur when $z^{t}_{(i)} = -1$, and a sensor cannot be installed at a location with an existing sensor when $z^{t}_{(i)} = 1$. Let $z$ be a decision vector; then a sequence of $z^{t}_{(i)}$ for all time periods $t \in T$ can be defined as $[z^{1}_{(i)}, \ldots, z^{\vert T \vert}_{(i)}]$. The frequency of $z = \{-1, 1\}$ is restricted to less than once for given operation period $T$ as follows:

$$|z = -1, 1| \leq 1 \quad \forall i \in N. \quad (10)$$

We replace these relocation associated constraints and present the multiperiod dynamic SLP with restricted relocation, which we denote by MSDP2:

$$\text{MSDP2} \quad \max_{\mu} \mathbb{E}_{\xi_{1}, \xi_{2}, \ldots, \xi_{T}} \left[ \sum_{t=1}^{T} \psi^{t}(x^{t}, \xi^{t}) \right]$$

s.t. \begin{align}
    & x^{1} = \mu(\xi_{1}); \\
    & x^{t} = \mu(x^{t-1}, \xi^{t}) \quad \forall 2 \leq t, t \in T; \\
    & \sum_{t \in T} x^{t} \leq c \quad \forall t \in T; \\
    & [y^{t}_{(j,l)}]^{t_{1}, t_{2}} \leq b^{t}_{(j,l)} \quad \forall t \in T, j = l; \\
    & b^{t}_{(j,l)} = b^{t-1}_{(j,l)} - [y^{t}_{(j,l)}]^{t_{1}, t_{2}} + b^{t}_{(l)} \quad \forall t \in T; \\
    & x^{t}_{(i)} \leq \sum_{j \in N} y^{t-1}_{(j,l)} \quad \forall i; \\
    & |z = -1, 1| \leq 1 \quad \forall i \in N; \\
    & x^{t} \in \{0, 1\} \quad y^{t} \in \{0, 1\} \quad \forall t. \quad (17)
\end{align}$$

The objective function represents maximizing the delay savings by installing sensors at optimal locations with restricted relocations. Constraints (11)–(15) and (17) are equivalent to MSDP1, while constraint (16) ensures that once a sensor has been used in relocation in the past and current periods, that same sensor cannot be relocated in the future.

We further simplify MSDP2 with one-stage lookahead with the restriction property on sensor relocation. By doing so, we can significantly reduce the computational burden in dynamic programming. The first stage ($t = 1$) and later stages ($t = 2, \ldots, T$) can be dependent, but we assume that later stages (the second to the last stage problems) can be solved independently of each other. That at most one sensor can be relocated makes the decisions in the periods $t + 1, \ldots, T$ nonanticipatory. Here the nonanticipativity conditions $x^{t}_{(j,l)}$ state that the second-stage decision depends only on the scenario that will prevail in the first stage. The first ($t = 1$) and the second to the last stage ($t = 2, \ldots, T$) problems can be solved independently of each other. The objective function in MSDP1 can be replaced as follows.

**Proposition 1.** Suppose that demand realizations $\xi^{1}$ are independent from $\xi^{t+1}$. Here the nonanticipativity conditions $x^{2}_{(j,l)}, \ldots, x^{T}_{(j,l)}$ state that the second-stage decision should not depend on the scenario that will prevail in the later stage. The multiperiod dynamic SLP MSDP2 is reformulated as a two-stage stochastic program, which we denote by MSDP2’:

$$\text{MSDP2’} \quad \max_{\mu} \mathbb{E}_{\xi_{1}, \xi_{2}, \ldots, \xi_{T}} \left[ \sum_{t=1}^{T} \psi^{t}(x^{1}, \xi^{1}) \right]$$

s.t. \begin{align}
    & x^{1} = \mu(\xi^{1}); \\
    & x^{t} = \mu(x^{1}, \xi^{t}) \quad \forall 2 \leq t, t \in T; \\
    & s.t. (12), (13), (14), (15), (16), (17).
\end{align}$$

**Proof.** The decision $x^{1}$ with $\xi^{1}$ is dependent on the future realization of uncertain demand $\xi^{2}, \xi^{3}, \ldots, \xi^{T}$. With restricted relocation, $\xi^{2}, \xi^{3}, \ldots, \xi^{T}$ are independent of each other. It implicitly accounts for the decision that $x^{2}$ is independent of the future realization of uncertain demand $\xi^{3}, \xi^{4}, \ldots, \xi^{T}$. Therefore, the decision $x^{2}$ is contingent on the outcome of random vector $\xi^{2}$, but is unique for all random parameters that are realized in the future, $\xi^{3}, \xi^{4}, \ldots, \xi^{T}$. Because of this independence, we can omit the conditional expectations from MSDP2. We note that the constraint (18) is deterministic that depends on $\xi$ only through the decision of $x$. There is no constraints linking realization of random demands $\xi$ for different time periods $t \in T$. Since the value of $\psi$ depends only on $x^{2}, b^{2}_{2}$ at $t = 2$, $x^{2}(x^{1}, b^{2}_{2}, \xi^{1})$ will be equal to $\psi(x^{2}, b^{2}_{2}, \xi^{1})$. □

In the previous formulation, we make decisions of the locations of all sensors in all time periods at the beginning of the planning horizon. Some studies found correlations between morning and evening commute distance and time (Kung et al. 2014), and evening commute as the mirror image of the morning commute (DePalma and Lindsey 2002). However, the morning and evening commutes can be independent because of different schedule preferences (Gonzales and Daganzo 2013). User equilibrium, for the evening commuters seeking to minimize the cost of their trip, must be a pattern of bottleneck arrivals and departures that allows no commuter to reduce his or her own cost by choosing another arrival position at the bottleneck. In this study, correlations between morning rush hour demand and the rest of the day are considered through an optimal relocation policy for each scenario based on conditional probability and expected delay savings.

In the next section, we present a stationary model without relocation.
3.5. Multiperiod Stochastic SLP
We present a baseline model proposed by Park and Haghani (2015) and compare it to relocation models. In this myopic problem setting, sensors are fixed in their optimal locations throughout the day without moving to other better locations in different time periods. The fact that the sensor location decision at time $t = 1$ is identical to that at $t = 2, \ldots, T$ is equivalent to the same decision vectors $x^1 = x^2 = \cdots = x^T$. This property makes the model nonanticipatory. Having an identical set of $x$ for all periods makes multistage stochastic programming, MSP, a stationary model. The solution needs to be compromised to incorporate the scenario from $t = 1$ to $t = T$ into one decision vector $x$. While constraint (18) is further simplified to $x = \mu(\xi)$, relocation constraints (12)–(17) are not used in MSP. Assuming demand realizations in different periods $\xi = \xi^1, \xi^2, \ldots, \xi^T$, we can bring all the maximization terms outside the expectations and present a formulation equivalent to MSP:

$$\text{MSP} \quad \max_{\mu} \mathbb{E}_{\xi^1, \xi^2, \ldots, \xi^T} \left( \sum_{t=1}^{T} \psi(x, \xi^t) \right)$$

s.t. $x = \mu(\xi)$; \hspace{1cm} (19)
$$\sum_{i \in N} x_i \leq c \quad \forall t \in \mathcal{T};$$ \hspace{1cm} (20)
$$x \in \{0,1\} \quad \forall t.$$ \hspace{1cm} (21)

The objective function represents maximizing the delay savings by installing sensors on optimal locations identical across different time periods without relocations. Constraints (19) ensure the stationary sensor location vector depends on demand realization in each period. Constraints (20) enforce the maximum number of available sensors. Constraints (21) ensure a binary decision variable.

3.6. Rolling Horizon Procedure
We propose $t'$-time-step anticipatory dynamic models with relocations. To make a sensor location decision in the current period, we roll the horizon forward one time period. On this rolling horizon procedure, we make decisions over the planning horizon $t' = t, \ldots, T$, and the decisions we make at time periods $t+1, \ldots, T$ are purely for the purpose of making a better decision at time $t$. When we are at time $t$ (in state $S'$), we can solve the problem optimally over the horizon from $t$ to $t + T$. Let $\psi(x', \xi^t)$ be the minimum delay we earned from implementing decision $x'$. After an implementation of our best decision on MSDP1, we then repeat the process by optimizing over the interval $t + 1$ to $t + T + 1$. We replace the solution of the old policy $\mu(x', \xi^{t+1})$ with the new policy $x'^{t+1}$. The new state $S^{t+1}'$ will have updated relocation time matrices following shortest path $\pi^{t+1}'$ and resulting delay savings $\psi(x'^{t+1}, \xi^{t+1}')$. This process would have more benefit with automation in relocation. This real-time process can be conducted by repeatedly using MSDP1, but this paper focuses on testing the model itself.

In the longer period, we may need one backup sensor for sensor failure, an unpredicted traffic crash, or a weather event. These components will be considered in a future study.

3.7. Source of Errors
The method proposed in this study will have more advantages when the market penetration rate is high enough. The market penetration rate of CVs plays a significant role for detecting queues; less accurate sampling leads to lower delay savings. The currently low market penetration rate may cause a relatively high error rate for queue detection and limited benefit of dynamic sensor relocation. However, compared to traditional queue detection (e.g., loop detectors), which takes longer, the model in this paper has a big advantage in using connected vehicle RSE for quicker queue detection for signal control. As market penetration rate increases, queue detection at signalized intersections will improve. The increasing trend in market penetration rate will justify the use of the proposed method in the near future.

In this study, contrary to Park and Haghani (2015), sampling errors from travel time that may appear during data processing are not used. Instead, the main focus of this study is on detecting the last vehicle in the queue by installing Cv sensors. Most of the connected vehicle test beds have focused on monitoring the road network travel time with communication latency, accuracy of the Global Positioning System, and light detection and ranging accuracy. Network latency can be an issue with long queues and OBE; however, this paper does not have the issue of latency or reliability of messages passing through roadside DSRC.

4. Solution Method
To solve large instances of the dynamic sensor location problem, we enhance the solution efficiency through decomposition. We introduce a tight Lagrangian bound and an efficient dual heuristic with an embedded search heuristic.

4.1. Nonsubmodularity in Dynamic SLP
In location problems, numerous studies have used monotone submodular functions. The greedy algorithm provides a good approximation to the optimal solution of the NP-hard optimization problem (Nemhauser, Wolsey, and Fisher 1978). However, the submodularity, the property that exhibits a natural diminishing returns property (Krause, Singh, and Guestrin 2008) cannot be applied in our sensor location problem. The submodular property is assumed to not exist because of the interaction effect of nearby signals and dynamic relocation of sensors at different times of a day. The sensor location
problem is solved with a different number of sensors as the constraint, assuming diminishing marginal delay savings. We will show in Section 5 that even with a reduced number of sensors, fair delay savings are guaranteed under feasible relocations. After reaching the maximum efficiency of the relocation, the level of diminishing marginal delay savings will become identical to that in a model without relocation.

The proposed signal control strategy may have less negative impact by locating another sensor in the nearby intersection. This coordination effect of sensors does not preserve the submodular property in the SLP. The marginal gain from two to three sensors may be higher than the gain from one to two sensors because of high interactions between the third sensor and other sensors. We prove that the submodular property does not hold on SLP in Appendix C.

The MSDP1 and MSDP2 with relocation make the problem more complicated, and the myopic greedy algorithm cannot efficiently solve complex combinatorial optimization problems. However, even though there is no submodularity, the effect of relocation on the diminishing return is presented in the case study (Section 5). With a reasonable relocation cost, a few sensors can have good performance in maximizing delay savings for the whole network. The increase in delay savings diminishes as we add an additional sensor. There is an exception that when relocation expense is more than highway administrations can afford to pay, instead of relocating sensors, additional sensor deployment is more economical. However, with the help of emerging sensor technology and automation, relocation cost will go down as time passes.

Since we cannot solve our SLP with the submodular function, we introduce Lagrangian relaxation in the next section.

### 4.2. Lagrangian Relaxation

We decompose the search space to two subproblems: a location problem and a relocation problem. Feasible solutions for each time period $t$ are connected by relocation.

There will be an explosion of problem size and curse of dimensionality when we try to solve this problem over a horizon. The explosive exponential complexity of the search space precludes the use of commercial solvers. For fair comparison between different sensor deployment concepts, we test our heuristics and focus on benefits of flexible relocations. This is a combinatorial optimization problem because we have to choose optimal location of sensors among candidate locations for each uncertain demand realization in multiple stages. The size of the state space typically grows exponentially in the number of policies considered, which depends on previous decisions.

As shown in Figure 2, as the number of periods increases toward the end of the day, the scenario tree grows exponentially, making it very challenging to optimize. The travel time between stages, $\pi_{ij}^{t+1}$, represent relocation costs at period $t + 1$. A node with black circle presents infeasible solutions that do not have to be considered in the search space. Therefore, by solving the relocation problem with constraints associated with relocation cost, future location changes at $t + 1, \ldots, T$ are fixed in a reduced search space. We introduce a discretization of the decision space by an iterative process. First, we solve the relocation problem to provide initial solutions with feasible links between optimal locations in each time period. Second, by fixing feasible links on the tree, the problem is simplified to find a reduced set of locations with some fixed locations defined by future relocations.

Applying a relaxation guided variable neighborhood search to the reduced problem instances yields significantly better solutions in shorter times than applying these metaheuristics to the original instances. We introduce Lagrangian relaxation to separate the problem into two. Then, the cutting plane algorithm is introduced to solve the Lagrangian dual problem, and we move into the search heuristic.

The dynamic sensor location problem in MSDP1 and MSDP2 exhibits a special structure that is suitable for Lagrangian relaxation. Let us associate nonnegative Lagrangian variables with $\lambda_{ij}^t$ constraints and apply Lagrangian relaxation.

The resulting Lagrangian problem is as follows:

$$L(\lambda) = \max \sum_{e, \gamma} \psi_e(x', \xi') - \sum_{t=1}^T \lambda_{ij}^t [y_t^i - x']$$

subject to

$$\lambda_{ij}^t \geq 0, \quad \text{for all } i, j, t.$$  \hspace{1cm} (22)

Note that $x'$ is contained only in constraint (3). This allows us to separate the aforementioned problem into two subproblems. The first subproblem is given as

$$L_1(\lambda) = \max \sum_{i, l} \lambda_{ij}^t y_{ij}^t$$

subject to

$$\text{(4), (5), (6), (7).}$$ \hspace{1cm} (23)

The second subproblem is defined as

$$L_2(\lambda) = \max \sum_{e, \gamma} \psi_e(x', \xi') - \sum_{t=1}^T \lambda_{ij}^t x'^t$$

subject to

$$\text{(2), (3), (8).}$$ \hspace{1cm} (24)

From the relations discussed previously, we observe that

$$L(\lambda) = L_1(\lambda) + L_2(\lambda)$$ \hspace{1cm} (25)

and note that $L(\lambda)$ yields an upper bound to the original problem, assuming that $\lambda \geq 0$. 


The decomposition of $L(\lambda)$ offers significant computational advantages over the original formulation. We calculate $L_1(\lambda)$ by solving solely the relocation problem. We calculate $L_2(\lambda)$ by scenarios only with the first stage influencing the rest of the stages. Denote by $x^*$ a reduced set of location decision vectors that has future relocations defined. The left term of $L_2(\lambda)$ can be replaced by $\sum_{i=1}^{T} E_{\xi^1,\xi^2,...,\xi^T} \left[ \psi(\{x^t, \xi^t\}) \right]$ with fixed relocations.

Since we find an upper bound for each value of $\lambda$, as long it is nonnegative, we want to find the value of $\lambda$ that leads to the tightest upper bound. We define
this as Lagrangian dual problem as \( \min_{\lambda \geq 0} L(\lambda) \). By calculating the optimal solutions of two subproblems \( \bar{y}^i \) and \( \bar{y}^j \), we can solve \( L_1(\lambda) \) and \( L_2(\lambda) \). A subgradient of \( L(\lambda) \) is expressed as follows:

\[
\delta^i_\lambda = \bar{y}^i - \bar{x}^i \quad \text{for } i = j.
\]

### 4.3. Cutting Plane Algorithm

In this paper, we use a subgradient algorithm enhanced with valid cuts and a dual heuristic. The subgradient algorithm starts by fixing the value of the Lagrangian variables \( k \) and solving for the primal variable vectors \( x \) and \( y \). Then the Lagrangian variables are updated based on the violation of the relaxed constraints. The algorithm stops when a maximum number of iterations is reached (Algorithm 1).

Let \( \bar{y} \) be a solution of the first subproblem \( (L_1(\lambda)) \), and let \( \bar{R} \) be the set of vertices such that \( \bar{y}^i = 1 \). The set \( \bar{R} \) gives a feasible location of repositioned sensors; in an optimal solution, either \( \sum^\bar{R} \|y^i\| = b_{(3)} \) or \( \sum^\bar{R} \|y^i\| \leq b_{(3)} - 1 \). As presented, either \( \bar{y} \) is optimal or \( \sum^\bar{R} \|y^i\| \leq b_{(3)} - 1 \) provides a valid cut to the problem. With the second subproblem \( (L_2(\lambda)) \) unchanged, the cut affects \( L_1(\lambda) \). With the \( L_1(\lambda) \) fixed associated with \( b_{(3)} \) locations, \( L_2(\lambda) \) is equal to \( \text{MSP} \).

In the algorithm, the cut \( \sum^\bar{R} \|y^i\| \leq b_{(3)} \) divides the solution space into two subregions at any iteration. In the first subregion, the constraint \( \sum^\bar{R} \|y^i\| = b_{(3)} \) is enforced. The associated objective value \( \sum^\bar{R} \sum_{t=1}^\tilde{R} \mathcal{L} \left( \psi^i(x^\lambda, \xi^j) \right) \) is equal to the lower bound \( Lbd \) calculated by the dual heuristic. In the second subregion, the cut states that at most \( b_{(3)} \) of the relocations in solution \( y’ \) can be used at a time. The two subregions are disjoint, and their union is the feasible region of \( \text{MSDP}_1 \). In subsequent iterations, either the lower bound is updated or the algorithm stops. If the lower bound is updated, a better feasible solution is found. In this case, the cut is updated accordingly. Otherwise, the algorithm stops with an upper bound \( Ubd \). This upper bound is not valid for the original problem \( \text{MSDP}_1 \), but rather on its restriction defined on the subregion defined by the best feasible solution found. Hence, the upper bound \( Ubd \) can be more than the lower bound \( Lbd \). In fact, if \( Ubd \leq Lbd \), the best feasible solution \( y’ \) is optimal to \( \text{MSDP}_1 \). This is true since the optimal objective value on the second subregion is less than or equal to \( Lbd \). Hence, \( Lbd \) cannot be improved. If \( Ubd \leq Lbd \), then \( Ubd \) is at most 500 \times \((Ubd - Lbd)/Ubd\) from the optimal objective value of \( \text{MSDP}_1 \).

**Algorithm 1** (Lagrangian heuristic with cutting plane)

/Step 1: Initialization/
Initialize the parameter of the algorithm;
set the step size \( \zeta := 1 \);
set the Lagrangian variables \( \lambda \) := 0, \( j \in J \);
set initial bounds \( Ubd = \infty \) and \( Lbd = 0 \);
Calculate \( \lambda = 0 \);
while iterations \( \zeta = \frac{\zeta}{2} \) do
/Step 2:/
set the \( \bar{x} \) and \( \bar{y} \) as the optimal solutions of the two subproblems;
Set the initial upper bounds \( Ubd = \min(Ubd, L_1(\lambda) + L_2(\lambda)) \);
/Step 3:/
Calculate a feasible solution of \( y’ \), and find associated max \( \text{DelaySavings} \)
\[ \sum_{t=1}^\tilde{R} \sum_{J=1}^\tilde{R} \mathcal{L} \left( \psi^i(x^\lambda, \xi^j) \right) \] using VNS;
Update lower bound \( Lbd = \max(Lbd, \sum_{t=1}^\tilde{R} \sum_{J=1}^\tilde{R} \mathcal{L} \left( \psi^j(x^\lambda, \xi^j) \right)) \);
/Step 4:/
Calculate subgradient \( \delta^i_\lambda = \bar{y}^i - \bar{x}^i \) for \( i = j \);
Calculate step size \( \eta := \zeta(Ubd - Lbd) \);
Set \( \lambda := \max(0, \lambda - \eta/(\lambda)) \);
/Step 5:/
if \( Ubd \geq \sum_{t=1}^\tilde{R} \mathcal{L} \left( \psi^i(x^\lambda, \xi^j) \right) \) in 500 iterations then
\( \text{stop and } Ubd := \sum_{t=1}^\tilde{R} \mathcal{L} \left( \psi^i(x^\lambda, \xi^j) \right) \)
else
\( \text{go to step 2} \)
end if
if \( 100 \times (Ubd - Lbd) \) \( \leq \sigma \) then
\( \text{stop} \)
else
\( \text{go to step 2} \)
end if
end while

When the lower bound is improved, it is possible to add the new cut instead of replacing the existing one. However, adding multiple cuts makes the second subproblem difficult to solve. The way we integrate valid cuts within a subgradient algorithm leads to a new cutting plane algorithm. The cutting plane algorithm calculates both an upper bound and a lower bound and generates a feasible solution. The gap between the bounds can be used as a stopping criterion. This overcomes one of the weaknesses of subgradient optimization, which is the stopping criterion. The algorithm stops when the bound does not improve for 500 iterations.

Given a solution \( \bar{y} \) of \( L_1(\lambda) \) with feasible relocations at \( t + 1, \ldots, \tilde{T} \), a stable and reduced location decision of \( \text{MSP} \) is solved with variable neighbor search in the next section.

### 4.4. Variable Neighborhood Search

To remedy the slow convergence issue, we employ variable neighborhood search, a metaheuristic based on transformations of solutions that determine one neighborhood structure on the solution space (Mladenović
and Hansen 1997). In VNS, a perturbation to the current neighborhood operator at a local minimum enables the search to reach a solution that could not have been reached by the current local search mechanism. It yields a broader exploration of the search space by visiting several high-quality local optima in the same CPU time.

A VNS structure uses a finite set of preselected neighborhood structures denoted by \( \Xi_\omega \). We start the algorithm with (1) initialization that selects the set of neighborhood structures \( \Xi_\omega \), for \( \omega = 1, \ldots, \omega_{\text{max}} \) used in the shaking phase, the set of neighborhood structures \( \Xi_v \) for \( v = 1, \ldots, v_{\text{max}} \) used in the local search, and a stopping condition. In the (2) shaking step, the incumbent solution is perturbed. The algorithm generates a solution \( x' \) at random from \( \omega \)th neighborhood \( \Xi_\omega \) of \( x = (x_1, \ldots, x_i) \). It takes sensor location to be inserted at random, if it satisfies \( \text{DelaySavings}(x') > \text{DelaySavings}(x) \), and finds location to be deleted at random. In the (3) local search step, the algorithm explores neighborhood to find the best neighbor \( x'' \) of \( x' \) in \( \Xi_v(x') \). In the (4) move or not step, if the local optimum \( x'' \) is better than the incumbent, move there (\( x \leftarrow x'' \)) and continue the search with \( \Xi_{\omega+1}(\omega \leftarrow 1) \); otherwise, set \( \omega \leftarrow \omega + 1 \). The steps of a basic VNS structure are defined as in the Algorithm 2.

**Algorithm 2** (Variable neighborhood search)

**Initialization**: Select the set of neighborhood structures \( \Xi_\omega \) and \( \Xi_v \);

**Input**: Find initial solution \( \hat{x} \leftarrow \text{StartingSolution} \);

**Output**: \( \text{DelaySavings}(\hat{x}) = \sum_{i=1}^N \sum_{j=1}^N \sum_{l=1}^{\text{Nls}} E_{ijl} [\psi(l(x_i, x_j))] \)

\( x_1 = x_2 = \cdots = \hat{x} \);

for \( \omega = 1 \) to \( \omega_{\text{max}} \): do

repeat the following steps:

/* Shaking */

for \( i = 1 \) to \( N \) do

if \( \text{DelaySavings}(x') > \text{DelaySavings}(\hat{x}) \) then

\( \hat{x} \leftarrow x' \)

\( \text{BestSavings} \leftarrow \text{DelaySavings}(x') \);

iteration \( \leftarrow 0 \);

end if

end if

end for

/* Local search */

for \( v = 1 \) to \( v_{\text{max}} \) do

if \( \text{DelaySavings}(x'') > \text{DelaySavings}(x') \) then

\( x' \leftarrow x'' \);

\( v \leftarrow 1 \);

else

\( v \leftarrow v + 1 \);

end if

end for

/* Move or not */

if \( \text{DelaySavings}(x'') > \text{BestSavings} \) then

\( \hat{x} \leftarrow x'' \)

\( \text{BestSavings} \leftarrow \text{DelaySavings}(x'') \);

end if

end for

**5. Case Study**

This section addresses a sensor location problem applied in downtown Burlington, Vermont, where spatial congestion patterns at different times of day vary. The network setup is explained in the Section 5.1, and the proposed models are applied in Section 5.2.

**5.1. Network Description**

The case study uses a calibrated microscopic traffic simulation model of Burlington, Vermont. The TransModeler simulator replicates sensor installation based on an anticipatory signal control model.

Figure 3 is a map of the Burlington road network. We consider a subnetwork located in the city center delimited by a circle, where the congestion is highest during peak hours. The network contains 71 links and 37 intersections, 19 of which are signalized candidate locations to install sensors and control the traffic flow. The existing signal plans were obtained from the City of Burlington. There are a total of 70 phases with a cycle time of 90 or 120 seconds.

The model was calibrated to match time-dependent traffic count and speed data observed in the field. We control a set of signalized intersections in the City of Burlington’s downtown network considering temporal and spatial variations in demand scenarios. Demand arises at the nine centroids nearest to these intersections. Under the rolling horizon scheme, indirect impact of the proposed control (user’s routing change) is random and realized after the simulation run. A more accurate microscopic simulator involves running multiple simulation replications. The simulation setup consists of 10 replications of each period, preceded by a 15 minute warmup time. To obtain lane-specific distributions from road-specific distribution, we disaggregate the flow data (i.e., external outflow, turning flow, external inflow) according to the method used by previous studies (Osorio 2010, Osorio and Chong 2015, Chong and Osorio 2018). Within this time period, congestion gradually increases. The average flow of the roads in the subnetwork steadily decreases and the average density increases. The dynamic OD matrix is used to generate trips, along with a headway model. Time-varying OD trip matrices and congested dynamic network travel times are estimated.

We use real-world data from field traffic counts, intersection turning movements (i.e., through, left turn, and right turn), origin-destination routing, and current traffic signal phase information obtained from...
the transportation data management system of the Chittenden County Regional Planning Commission (CCRPC; Vermont Agency of Transportation 2016). The calibrated version of the Chittenden County Metropolitan Planning Organization TransCAD model is integrated into TransModeler based on calibration of trip generation, trip distribution, mode choice, and vehicle assignment steps verified against the National Household Travel Survey and 2040 Metropolitan Transportation Plan.

Morning commutes were generally more consistent than evening commutes, with most people arriving at work between 7:00 a.m. and 9:00 a.m. Nearly a third of the participants felt it was possible to work from home occasionally, but not on a consistent basis (CCRPC; Vermont Agency of Transportation 2016). Traffic flow was substantially higher in the southbound direction during the morning peak hours, and higher in the northbound direction during the afternoon. For example, the southbound morning peak movement (1,217 vehicles/hour) was 3.3 times higher than northbound movement (365 vehicles/hour), and northbound afternoon peak movement (951 vehicles/hour) was 1.7 times higher than southbound movement (549 vehicles/hour) in the west part of the map in Figure 3, where Battery St. and Pearl St. intersect. The traffic volume patterns during the morning and afternoon commute peak hours present significant locational differences that justify a sensor relocation policy in Burlington.

We compare the performance of the best plan derived by anticipatory signal control. For each sensor deployment strategy, we derive the optimal signal plan on a set of intersections, and then use the simulation model to estimate the parameters with the signal control effect on the subnetwork. The objective is to maximize the expected delay savings associated with signal control over the three periods ($|T| = 3$) for different numbers of sensors. This is a suitable short horizon that was chosen to capture important behaviors. Two demand scenarios $\xi$ are considered: high and low demand. Entry demands were 1.5 times ($\xi_{\text{high}}$ demand scenario) and 0.75 times ($\xi_{\text{low}}$ demand scenario) of the original demand for the first, second, and third 15-minute intervals, and equal to the original demand for the last 15-minute interval.

A threshold for the end of a queue was determined through simulation for the different signal control strategies that were tested. For different interactions, the queue length threshold $\theta$ ranges from 0.6 to 0.9. The next section presents how the dynamic location approach derives signal plans that perform significantly better than the Burlington plan for some problematic intersections.

5.2. Results

Peak hours are spaced one period apart, corresponding to the congestion pattern of the time of day. The performances of difference sensor location strategies, MSP, MSDP1, and MSDP2, are tested and compared. Then the signal strategy changes according to the delay improvement, with different levels of penetration rate presented. Lagrangian relaxation and the VNS...
algorithm enabled us to solve the dynamic sensor location problem in a reasonable time.

5.2.1. Anticipatory Dynamic SLP with Relocations. We test the proposed multiperiod stochastic sensor location models. First, MSDP1 has a budget $c$ to implement two to nine sensors with different maximum relocation frequencies.

The most similar demand profiles are clustered into stages. In this study, three main stages (morning peak hours, nonpeak hours, and afternoon peak hours) are used to find the effectiveness of sensor relocation. Even though it would be interesting to investigate the feasibility of relocation frequency under shorter intervals (e.g., every one hour) and for the whole day starting at 12:00 a.m., this paper focuses on the feasibility of the best deployment and signal control strategy under a given interval. Optimal three-period deployment plans for different numbers of installed sensors $c$, the corresponding delay savings, and computation times are presented in Table 3. The magnitude of relaxation can be adjusted for the effectiveness of sensor relocation based on the dependency between intervals, considering relocation expense and computational efficiency. The deployment plans for $c = 2, 3, 4, 5, 6, 7$ are illustrated in Figure 4, while a similar illustration for $c = 8, 9$ is provided in Appendix D (Figure D.2).

The metamodel coefficients consist of the direct control strategy on signalized intersections derived by the proposed formulation of green time allocation (Appendix A) and the indirect influence on nearby links and intersections estimated by simulation (Appendix B). The proportion of the indirect effect of signal control on the whole network ranges from 2.1% to 16.5% of total impact on the delay savings. To be more specific, total delay savings (872 s) is subtracted from the direct effect of control (890 s) to calculate the indirect effect of control (18 s), 2.1% of total delay savings with two available sensors. This is reasonable because it takes time for users to learn a new system. More advantageous users may choose their routines by switching to different routes and explore new systems. On the contrary, more conservative users may stick to their routines because they are reluctant to explore new systems. As time goes by after the onset of the new signal system, the proportion of users switching to different routes will increase. In this paper, we assume underlying user equilibrium for every implementation of a new system. It is a metamodel that has a unique contribution, and it is appropriate for the real-time framework in this study. In a future study, the optimal timing of changing the system optimum considering the learning rate of users will be considered. That model can be based on iteration of a bilevel combinatorial optimization that may significantly increase the computational time.

The results of Table 3 reveal interesting patterns. First, Table 3 presents the nonexisting submodular property with the diminishing return effects, only after the number of sensors reaches $c = 3$. The increase in the delay savings by having more flexible relocation diminishes when we have more sensors. Compared to limited relocation in Table 5, more delay savings are observed for all numbers of sensors in Table 3. Unsurprisingly, when the budget for the number of sensors is relatively large (e.g., $c = 5$), there is less need to relocate sensors. Maximized delay savings using a small number of sensors $c = 2, 3, 4$ is equivalent to having more sensors. Since intersections with higher delay savings will have a priority, adding one more sensor will have less gain in delay savings:

$$\Delta(x' | [x_{14}, x_{15}, x_{16}]) \geq \Delta(x' | [x_{13}, x_{14}, x_{15}, x_{16}]).$$

In this example, the marginal benefit provided by placing a sensor, given deployed sensors at locations $[x_{14}, x_{15}, x_{16}]$ does not increase as we deploy one more sensor $[x_{13}]$.

Second, more relocation is given to improve the solution of MSDP1 with $c$ sensors in each scenario. Note that decisions we make at time period $t = 3$ are purely for the purpose of making a better decision at time $t$. We project location decisions over this horizon because we need to know what we would do in the future to know what we should do right now. Without forecasting stochastic variations from the mean, we may lose the chance to relocate during nonpeak hours, which has a lower relocation cost, and relocate during peak hours, which has a higher relocation cost.

<table>
<thead>
<tr>
<th>Budget</th>
<th>$t = 1$ Location</th>
<th>$t = 2$ Location</th>
<th>$t = 3$ Location</th>
<th>Total delay savings (Indirect %)</th>
<th>CPU (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c = 2$ sensors</td>
<td>8, 13</td>
<td>7, 8</td>
<td>1, 5</td>
<td>872 s (2.1 %)</td>
<td>537</td>
</tr>
<tr>
<td>$c = 3$ sensors</td>
<td>14, 15, 16</td>
<td>8, 9, 11</td>
<td>8, 9, 11</td>
<td>1,058 s (6.6 %)</td>
<td>598</td>
</tr>
<tr>
<td>$c = 4$ sensors</td>
<td>13, 14, 15, 16</td>
<td>7, 8, 9, 11</td>
<td>1,7, 8, 9</td>
<td>1,202 s (7.9 %)</td>
<td>667</td>
</tr>
<tr>
<td>$c = 5$ sensors</td>
<td>10, 13, 14, 15, 16</td>
<td>7, 8, 9, 11, 13</td>
<td>1,7, 8, 9, 13</td>
<td>1,270 s (9.3 %)</td>
<td>735</td>
</tr>
<tr>
<td>$c = 6$ sensors</td>
<td>5, 10, 13, 14, 15, 16</td>
<td>5, 7, 8, 9, 11, 13</td>
<td>1,5, 7, 8, 9, 13</td>
<td>1,338 s (10.8 %)</td>
<td>719</td>
</tr>
<tr>
<td>$c = 7$ sensors</td>
<td>5, 10, 13, 14, 15, 16, 18</td>
<td>5, 7, 8, 9, 11, 13, 18</td>
<td>1,5, 7, 8, 9, 13, 18</td>
<td>1,397 s (12.5 %)</td>
<td>638</td>
</tr>
<tr>
<td>$c = 8$ sensors</td>
<td>4, 5, 10, 13, 14, 15, 16, 18</td>
<td>4, 5, 7, 8, 9, 11, 13, 18</td>
<td>1,4, 5, 7, 8, 9, 13, 18</td>
<td>1,450 s (14.4 %)</td>
<td>995</td>
</tr>
<tr>
<td>$c = 9$ sensors</td>
<td>4, 5, 10, 12, 13, 14, 15, 16, 18</td>
<td>4, 5, 7, 8, 9, 11, 12, 13, 18</td>
<td>1,4, 5, 7, 8, 9, 12, 13, 18</td>
<td>1,494 s (16.5 %)</td>
<td>543</td>
</tr>
</tbody>
</table>
Figure 4. (Color online) An Illustration of Flexible Relocations in the City of Burlington: By Adding an Additional Sensor, the Strategy for Sensor Configuration on the Network Changes

(a) Sensor deployment plan $c = 2$
(b) Sensor deployment plan $c = 3$
(c) Sensor deployment plan $c = 4$
(d) Sensor deployment plan $c = 5$
(e) Sensor deployment plan $c = 6$
(f) Sensor deployment plan $c = 7$

Notes. The main effects and interactions are presented as coordination effects between intersections. We maximize the delay savings using only a small number of sensors, $c = 2, 3, 4$. Relocation occurs during nonpeak hours to travel further. The transportation authority can save budget. Having more than $c = 5$ sensors presents diminishing return in delay savings.
Third, different spatial congestion patterns at different times of day are presented on the network. Demand and queue tend to follow a daily rush-hour pattern; queue grows during the morning peak hours and then disappears. The significant morning peak and afternoon peak traffic influence the delay savings. Therefore, it is beneficial to install sensors at places where delay savings can be maximized: at \([x_{14}, x_{15}, x_{16}]\) during the morning peak and with relocation to \([x_8, x_9, x_{11}]\) during the afternoon nonpeak. The installed sensors at two consecutive intersection help to maximize the delay savings. The spatial congestion patterns are in line with the consideration of spatial correlation of different road segments in the sensor location problem (Park and Haghani 2015).

Fourth, information loss (relocation cost) around noon tends to be lowest. This is due to the temporal congestion patterns at different times of day. It is better to flexibly move sensors without extra relocation cost during nonpeak hours. Instead of saving the relocation budget for future periods, with a similar benefit, a sensor may move to a location where a significant queue is expected at an earlier stage during nonpeak hours. Instead of saving the relocation changes with sensor relocation (Table 4) and without sensor relocation (Table 5). Table 4 presents MSDP2 with the relocation frequency limited to once. By reducing the complex dynamic problem MSDP1 to a two-stage stochastic problem, the approximate solution of MSDP2 reduces the computational time by least 20.99% to at most 38.87%. The best solution gap to the best known values of MSDP2 represents some loss of reward in delay savings. The gap between the solution of MSDP1 and approximate solution of MSDP2 ranges from 2.68% to 25.69%. Having more than six sensors has a less than 10% gap.

The sensor relocation between time periods improves the solution with complexity. As a result, the computational time of MSDP1 is in general at least three times longer than that of MSP (Table 5). Interestingly, this stationary model has an average gap of 22.32%, close to highest gap of MSDP2. Therefore, the approximate relocation model can achieve both computational efficiency and less quality loss in solutions by capturing complex interactions taking place within our optimization of the future. In practice, with an enough budget for the sensor deployment, transportation authority personnel may prefer a policy from MSDP2, which has an average 32.18% in computational savings compared to MSDP1.

In MSP, because more sensors are installed at the beginning, more delay savings are expected with the same sensor installation cost. Problem MSP presents diminishing return in delay savings with additional sensors from \(c = 2\) to \(c = 4\) and from \(c = 5\) to \(c = 9\). Problem MSDP2 presents a one-time-period look-ahead policy by explicitly optimizing with an approximation of future information. As shown in Table 5, in MSDP2, making a decision at time \(t = 1\) considering

---

**Table 4.** Optimal Deployment Plans Without Relocation for Three Periods of a Day

<table>
<thead>
<tr>
<th>Budget</th>
<th>(t = 1) Location</th>
<th>(t = 2) Location</th>
<th>(t = 3) Location</th>
<th>Total savings (s)</th>
<th>Gap (%)</th>
<th>CPU (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(c = 2) sensors</td>
<td>8, 15</td>
<td>8, 15</td>
<td>8, 15</td>
<td>520</td>
<td>40.37</td>
<td>129</td>
</tr>
<tr>
<td>(c = 3) sensors</td>
<td>7, 8, 15</td>
<td>7, 8, 15</td>
<td>7, 8, 15</td>
<td>675</td>
<td>36.20</td>
<td>160</td>
</tr>
<tr>
<td>(c = 4) sensors</td>
<td>7, 8, 13, 15</td>
<td>7, 8, 13, 15</td>
<td>7, 8, 13, 15</td>
<td>836</td>
<td>30.45</td>
<td>182</td>
</tr>
<tr>
<td>(c = 5) sensors</td>
<td>8, 9, 11, 14, 15</td>
<td>8, 9, 11, 14, 15</td>
<td>8, 9, 11, 14, 15</td>
<td>976</td>
<td>23.15</td>
<td>210</td>
</tr>
<tr>
<td>(c = 6) sensors</td>
<td>1, 8, 9, 11, 14, 15</td>
<td>1, 8, 9, 11, 14, 15</td>
<td>1, 8, 9, 11, 14, 15</td>
<td>1,118</td>
<td>16.44</td>
<td>173</td>
</tr>
<tr>
<td>(c = 7) sensors</td>
<td>1, 8, 9, 11, 13, 14, 15</td>
<td>1, 8, 9, 11, 13, 14, 15</td>
<td>1, 8, 9, 11, 13, 14, 15</td>
<td>1,224</td>
<td>12.38</td>
<td>190</td>
</tr>
<tr>
<td>(c = 8) sensors</td>
<td>1, 5, 8, 9, 11, 13, 14, 15</td>
<td>1, 5, 8, 9, 11, 13, 14, 15</td>
<td>1, 5, 8, 9, 11, 13, 14, 15</td>
<td>1,298</td>
<td>10.48</td>
<td>134</td>
</tr>
<tr>
<td>(c = 9) sensors</td>
<td>1, 5, 8, 9, 11, 13, 14, 15, 18</td>
<td>1, 5, 8, 9, 11, 13, 14, 15, 18</td>
<td>1, 5, 8, 9, 11, 13, 14, 15, 18</td>
<td>1,358</td>
<td>9.10</td>
<td>122</td>
</tr>
</tbody>
</table>
future relocations at time $t = 2$ and time $t = 3$ produces more benefit than MSP. Figure 5 illustrates the effect of relocation in SLP application with number of sensor $c = 2, 3, 4, 5, 6, 7$. A similar description for $c = 8, 9$ is provided in Appendix D (Figure D.1).

As expected, instead of the fixed solution at $[x_7, x_8, x_{15}]$, the flexible solution with relocations at $[x_7, x_8, x_{15}]$ and $[x_7, x_9, x_{15}]$ present higher delay savings (Table 5). Interestingly, with sensor number $c = 4$, relocation enables sensors to be located in optimal locations during the peak hours. There is a steep increase in delay savings with sensor number $c = 5, 6$ that has an equivalent effect of having more sensors.

Relocation increases the level of diminishing returns in the delay savings by having higher delay savings with sensor number $c = 4, 5, 6$. After $c = 7$, a steep level of diminishing returns is presented. As described in Park and Haghani (2015), with a limited number of sensors, a good relocation policy can have reasonable delay savings. In fact, installing 8 sensors with relocation is equivalent to installing 10 sensors without relocation.

By adding an additional sensor, the relocation increases the gain on delay savings from having coordinated signal control. The interaction terms capture the coordination across consecutive intersections. If this strategy is applied only at a single link and there is not sufficient queue storage capacity at upstream links, the queue problem could transfer to another part of the network, causing long queues at intersections located further upstream (Christofa, Argote, and Skabardonis 2013). By deploying sensors at two consecutive intersections, queues are stored on longer links and do not cause intersection blockages at other parts of the network. In fact, most of the time, the green phases in Equation (A.2) are allocated to another critical phase instead of being evenly distributed to all directions. This property shows that the two-way interaction in the metamodel in this study properly reflects network delay savings.

The ratio of the difference between the best known optimal solution and approximate solution divided by best known optimal solution was consistent (average, 1.03%; standard deviation, 1.11%) and varied between 0% to 4.72% over a variety of instances from two to nine sensors. Given the combinatorial optimization problem, the size of the instance being solved increases exponentially. The proposed variable neighborhood search method systematically changes the neighborhood structures within the search to avoid local optima.

To summarize the result, there is a clear submodular property in MSP when we increase the number of sensors used from $c = 2$ to $c = 9$. Dynamic models break the assumption of the submodular property by accelerating delay savings with sensor relocations. Figure 6 presents different levels of diminishing return in delay savings. With flexible relocations, MSDP1 presents the highest rate of delay savings with smaller sensor numbers $c = 2, 3, 4$, to save the budget of the transportation authority. The largest delay savings (1,494 s, $c = 9$), 17.3% of total network delay, is achieved with MSDP1. With four available sensors ($c = 4$), 14.5% of total delay is saved with MSDP2, compared to 15.5% with MSDP1 and 12.9% with MSP. With limited relocations, MSDP2 presents the highest rate of delay savings with sensor numbers $c = 4, 5, 6$ and diminishing return in delay savings.

### Table 5. Optimal Deployment Plans with Limited Relocation for Three Periods of a Day

<table>
<thead>
<tr>
<th>Budget</th>
<th>$t = 1$ Location</th>
<th>$t = 2$ Location</th>
<th>$t = 3$ Location</th>
<th>Total savings (s)</th>
<th>Gap (%)</th>
<th>CPU (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c = 2$ sensors</td>
<td>8, 15</td>
<td>8, 15</td>
<td>5, 15</td>
<td>648</td>
<td>25.69</td>
<td>411</td>
</tr>
<tr>
<td>$c = 3$ sensors</td>
<td>7, 8, 15</td>
<td>1, 15</td>
<td>1, 15</td>
<td>794</td>
<td>24.95</td>
<td>439</td>
</tr>
<tr>
<td>$c = 4$ sensors</td>
<td>7, 8, 13, 15</td>
<td>1, 8, 13, 15 12</td>
<td>1, 5, 13, 15</td>
<td>980</td>
<td>18.47</td>
<td>927</td>
</tr>
<tr>
<td>$c = 5$ sensors</td>
<td>8, 9, 14, 15, 16</td>
<td>8, 9, 14, 15, 16</td>
<td>8, 9, 11, 14, 15</td>
<td>1,122</td>
<td>11.65</td>
<td>474</td>
</tr>
<tr>
<td>$c = 6$ sensors</td>
<td>7, 8, 9, 14, 15, 16</td>
<td>1, 8, 9, 14, 15, 16</td>
<td>1, 8, 9, 11, 14, 15</td>
<td>1,254</td>
<td>6.28</td>
<td>446</td>
</tr>
<tr>
<td>$c = 7$ sensors</td>
<td>7, 8, 9, 13, 14, 15, 16</td>
<td>1, 8, 9, 13, 14, 15, 16</td>
<td>1, 8, 9, 11, 13, 14, 15</td>
<td>1,324</td>
<td>5.23</td>
<td>390</td>
</tr>
<tr>
<td>$c = 8$ sensors</td>
<td>5, 7, 8, 9, 13, 14, 15, 16</td>
<td>1, 5, 8, 9, 13, 14, 15, 16</td>
<td>1, 5, 8, 9, 11, 13, 14, 15</td>
<td>1,396</td>
<td>3.72</td>
<td>376</td>
</tr>
<tr>
<td>$c = 9$ sensors</td>
<td>5, 7, 8, 9, 13, 14, 15, 16, 18</td>
<td>1, 5, 8, 9, 13, 14, 15, 18</td>
<td>1, 5, 8, 9, 11, 13, 14, 15, 18</td>
<td>1,454</td>
<td>2.68</td>
<td>337</td>
</tr>
</tbody>
</table>

### 6. Application

In this study, three stages of a look-ahead model with relocation were explored. To expand the sensor location model over a five-day period, there are two potential strategies. First, a look-ahead policy that extends one day is enough to produce high-quality decisions. We can then simulate our policy to produce forecasts of congestion over all five days. Second, we can solve the multistage problem for daily operation today, and after observing all the outcomes of random variables, reoptimize the decision and run it again tomorrow.

#### 6.1. Penetration Rates

The proposed models in this study have the benefits of installing sensors in congestion. Argote, Christofa, and Skabardonis (2015) showed that oversaturated traffic conditions require relatively lower penetration rates...
Figure 5. (Color online) An Illustration of Sensor Installation with Limited Relocation: By Relocating Sensors, the Strategy for Sensor Configuration on the Network Changes

(a) Sensor deployment plan $c = 3$  
(b) Sensor deployment plan $c = 4$

(c) Sensor deployment plan $c = 5$  
(d) Sensor deployment plan $c = 6$

(e) Sensor deployment plan $c = 7$  
(f) Sensor deployment plan $c = 8$

Note. The main effects and interactions between sensor locations are improved by relocating sensors.

for accurate estimation of speed ($p = 0.05$) or delay calculation ($p = 0.15$). According to current low penetration rates of CV technology on most transportation infrastructures, we explore different levels of penetration rates. Figure 7 presents the model’s performance at three levels of penetration rate ($p = 0.05, 0.15, 0.25$).
for the same demand scenarios. For accurate queue detection, an appropriate portion of CV technology-equipped vehicles among other traffic is required. Based on the results, a 15% penetration rate of CV technology-equipped vehicles presents the steepest gradient when we add one more sensor into the network. A 5% penetration rate presents the lowest increase in total delay savings with an additional sensor. The delay savings from a penetration increase from 5% to 15% is, on average, 1.5 times higher than that from a penetration increase from 15% to 25%.

The numbers in parentheses in Figure 7 show the sum of all movements (relocation frequency) in each stage. As the penetration rate increases, there is more chance to reduce delay and more chance to relocate sensors. This is due to the trade-off between relocation cost and delay saving benefits. As the penetration rate decreases, relocation frequency limited without guaranteed benefit with labor cost.

The average total delay for different numbers of sensors (i.e., \( n = 2, \ldots, 9 \)) with lower penetration rates \( (p = 0.05) \) with less delay savings was 1.4 times the average delay with penetration rate \( p = 0.15 \), and that with higher penetration rates \( (p = 0.25) \) with more delay savings was 0.6 times the average delay with penetration rate \( p = 0.15 \). With low rates \( (p = 0.05, 0.1) \), the transportation authority can trigger the signal less frequently, and after some time period, frequent signal trigger. Based on their empirical observations, a transportation authority can use the proposed method at different levels of market penetration rate.

Compared to previous studies, our signal control model takes account of all approaches to intersection vehicle departures. Therefore, this approach prevents the case where network delay increases because of negative influence on a critical direction when two directions are served by the same phase.

For connected vehicle applications at low market penetration rates, many researchers have conducted estimates of the positions of noncommunicating (unequipped) vehicles. Therefore, we can have more delay savings in the near future.

### 6.2. Demand Profile

As a follow-up on this point, considering your earlier response to comment 2 where you state that average relocation frequencies could be in the order of a few times per week, how does this relate to distinguishing/incorporating variability at the level of within day or day to day or perhaps seasonal? Do low relocation frequencies in principle limit analysing (high frequent) within day variability? This is a question arising when reading this part of your paper, that I assume other readers may also have, so it would be helpful to clarify this matter. The current application should be running the model every day considering within-day variability, but in the future, we can look ahead further than one day considering day-to-day variability. The everyday model is more feasible to update traffic information from the previous day to have more accurate simulation with precise solution for a specific scenario, and to take advantage of having a nonanticipatory constraint. However, if we have more accurate future prediction, day-to-day variability is considered with more approximation to the future stages and computational
time. It will require an anticipation with longer look-ahead stages beyond the rest of the day, without a nonanticipatory constraint. In addition, the last location of one day needs to be conserved for the next day in calculating the optimal relocation frequency. For example, when within-day variability for a specific day is not beneficial enough, there is no relocation for that day. However, if day-to-day variability for a week is beneficial, there will be relocation, and vice versa.

We compare two different DV scenarios composed of temporal and spatial variation using OD matrix in a specific time stage with total four available sensors. In the three-stage setting, let $DV'_{12}$ be the changes of each cell ($i \in$ total number of ODs) in the OD matrix from the morning peak (stage 1) to the nonpeak (stage 2), and let $DV'_{23}$ be the changes from nonpeak (stage 2) to the afternoon peak (stage 3). Each cell in the OD matrix represents within-day variation measurement $DV'_{ij} = DV'_{i} + DV'_{j}$. When there is a temporal variation only (without spatial variation) between stages, $DV'_{i}$ is uniformly distributed. When there are both temporal and spatial variations, the distribution of $DV'_{i3}$ is more deviated from the center.

In time-dependent OD matrices, compared to the existing demand profile, the daily demands are explored to be less stochastic (similar demand) and more stochastic (very different demand) in different time stages. In the time-dependent OD table, two parameters (time variation and spatial variation) are used as main factors. With different demand for flexible relocations during three periods of a day (morning peak and afternoon nonpeak and peak), the performance of sensor relocation is evaluated.

The difference in each cell in the OD matrix from the first to the last stage, $DV'_{i}$ generates a different level of variation: standard deviations of $DV'_{i}$ for different OD pairs $i$. With the variation less than 20%, there was a minimal increase in total delay savings. The optimal deployment plan in Table 3 is with a medium demand distribution with standard deviation $\sigma_{DV'_{i}} = 2.3$. For more a deterministic demand distribution with average $\sigma_{DV'_{i}} = 0.1$ between different numbers of sensors, the overall relocation frequency decreased. Total delay savings were 854 s for two sensors, which is a 2.1% decrease due to deterministic demand distribution; 1,027 s (2.9%) for three sensors; 1,146 s (4.7%) for four sensors; 1,209 s (4.8%) for five sensors; 1,284 s (4.0%) for six sensors; 1,352 s (3.2%) for seven sensors; 1,412 s (2.6%) for eight sensors; and 1,473 s (1.4%) for nine sensors. When there is a temporal variation only (without spatial variation) between stages or there is a uniform growth factor to increase or decrease overall traffic volume in each cell of the OD matrix, the traffic control may operate well on both real-time measurements and historical distributions.

For $\sigma_{DV'_{i}} = 4.6$ between different numbers of sensors, total delay savings were 887 s for two sensors, which is a 1.7% increase due to stochastic demand distribution; 1,084 s (2.5%) for three sensors; 1,245 s (3.6%) for four sensors; 1,308 s (3.0%) for five sensors; 1,375 s (2.8%) for six sensors; 1,424 s (1.9%) for seven sensors; 1,466 s (1.1%) for eight sensors; and 1,507 s (0.9%) for nine sensors. This is because highway capacity is not easily altered, but sensor relocations can be adjusted in response to the dramatic changes in traffic demand. In case of more stochastic demand profile, the real-time measurements are necessary. The trade-off between fixed locations and portable sensors with relocation is justified.

By knowing the demand profile, transportation agencies can adjust budgets for deployment based on a feasible relocation frequency to maximize the travel time delay savings. The true benefit of the proposed relocation comes from real-time collected CV data that can explain traffic dynamics compared to historically collected empirical demand distributions.

6.3. Deployment Feasibility and Future

6.3.1. Feasibility of Relocation. This paper proposes initial sensor locations considering potential sensor relocations. If relocation is not feasible, there is no relocation. The sensor relocation problem is closely related to variations of within-day demand, day-to-day demand, and marker penetration rate.

It is important to consider potential relocation scenarios when locating sensors. According to Intelligent Work Zone Deployments conducted by the Iowa Department of Transportation, sensor relocation cost is $480 (Jackels et al. 2015), purely from relocation labor. The installation cost includes initialization and setting up new equipment, and it costs up to $2,500 (Wright et al. 2014).

Once new equipment is set up, we only need to relocate sensors with minor tunings but not major reconstructions. Considering the sensor cost by the equipment itself ($10,000) and maintenance cost ($3,000) every year, the total cost of six fixed sensors with a life cycle of five years is $165,000. On the contrary, two portable sensors with an average 9 relocations every month for five years is $158,686. If we increase the average monthly relocation frequency to 10 times, the portable sensor strategy is more expensive than the fixed sensor strategy.

As the specialization of labor increases, the productivity increases. As a return in economy scale, the average cost decreases. If the benefit is not enough, there is no relocation. Since labor fee is very high, we need a very accurate model to justify relocation. In addition to network delay savings used in this paper, key potential benefits from value of time, environmental sustainability impact (e.g., CO₂ reduction), energy savings, and
safety for each vehicle on the network increase the feasibility of relocation.

The proposed approach is more feasible when there are already existing sensors such as point sensors (loop detectors) or point-to-point sensors (automatic vehicle identification detectors; Zhou and List 2010). These sensors can support close/open loop calibrations to validate the proposed simulation against real-world speed and travel time on the network.

A portable CV–based sensor idea has been widely used in the live transmission of safety measures in the vicinity of incidents, weather events, and work zones (Maitipe, Hayee, and Kwon 2011). These sensors are fully portable and can be easily installed at any work zone site. The initial setup requires a quick and simple configuration of input parameters of the road to be monitored.

A portable traffic signal takes 10 minutes to transform from trailer to operating position. Such application can be quickly and easily transported and deployed (Maitipe, Hayee, and Kwon 2011). The temporary traffic signal marketplace now offers quickly deployable tools to help manage work zone traffic. These tools offer the ability to enable the motorizing public to experience a minimum of delays, and wireless operation between nonconnected portable signal trailers. Many patents for state-of-the-art technology that will potentially make relocation more feasible and easier are under review for portable remote-controlling traffic signal (Christiansen et al. 2016; Cardote 2016; Cherewka 2016).

6.3.2. Future Applications. Periodic retiming of traffic signals has been shown to yield road-user benefits that typically exceed the cost of the retiming by as much as a 40 to 1 ratio (Bonneseon and Sunkari 2009). Because of changes in travel demand pattern over time, the signal timing plan should be periodically updated to maintain intersection safety and efficiency. Calibration intervals can be adjusted more or fewer times based on actual deployment compared to the simulation optimization model. More frequent calibration can be required with higher uncertainties in demand and lower penetration rates of CV technology-equipped vehicles to compensate for accurate queue detection and guaranteed delay savings.

Vehicle-to-infrastructure applications have been improved, and various applications of temporary deployment of portable sensors have already been implemented in many urban cities for purposes such as reduced speed work zone warnings, speed zone warnings, emergency communications and evacuation, approach lane use management, CV technology-enabled turning movement, emissions analysis, hazmat monitoring and response, probe-based traffic monitoring, and pavement maintenance shipment Wright et al. (2014).

The proposed optimization approach offers the ability to help transportation agencies to achieve minimum delays in a variety of work zones. The concept of sensor relocation can be applied to portable traffic signal equipment to minimize delays and increase work zone safety with controlled costs and improved profitability. These sophisticated portable traffic signals are now the smart tools for today’s work zone traffic control and the emerging direct current (DC)-powered traffic control technology, DC-powered LED technology, solar power technology, nonintrusive vehicle detection, and frequency-hopping spread spectrum radios (Stout and Yost 2011).

Sensor relocation must be scheduled and synchronized. Employing humans for this repetitious operation is expensive. For this task, synchronously commanding robots, drones, and autonomous vehicles would be ideal, rather than employing humans, which cost more for reliable performance. Scheduling would be much faster. For example, unmanned aerial vehicles (UAVs) have been used for monitoring recurring and nonrecurring traffic conditions and special events on transportation networks (Zhang et al. 2015). Once issues such as battery limitation and space regulations are resolved, a UAV can be a CV sensor by delivering from the previous location to the next location. Though the current moving cost is expensive, UAVs and robots are becoming cheaper.

The performance of the proposed solution as the network size grows depends on the following aspects. The first is exceeding distance from the previous location to the next location: the relocation expense and data loss during relocation can be minimized by current network size. The second is the minimum penetration rate of CV technology-equipped vehicles: the sparse distribution of connected vehicles on the network and existing signalized intersections with signal controllers will be a major consideration. The third is the rush-hour congested demand pattern: as temporal and spatial variation in demand increases, the benefit of relocation is maximized. The fourth is that careful estimation of direct and indirect influences from signal control through CV sensors is required. The fifth is that the scenario size grows exponentially in the worst case as the network size increases. The nonanticipatory assumption greatly reduces the complexity of the problem, which has the benefit of allowing the transportation authority to solve much larger and more realistic instances. Finally, we prove in Appendix C that the SLP proposed in this study does not preserve a submodular property.

7. Conclusion and Future Study
Offering an efficient solution to urban traffic congestion, this paper explores the dynamic relocation of sensors
to improve network delay by controlling traffic signals under demand uncertainty. The proposed methodology combining CV technology with microsimulation can be applied to any sensor location problem handled with portable devices. Lagrangian relaxation and the cutting plane method add a valid cut with a better bound, and the second subproblem is solved faster with a variable neighborhood search method. Among three multiperiod stochastic models, the look-ahead policy provides the maximum benefit, and accelerated diminishing delays with additional sensors. With limited budget, the traffic operation may achieve maximum benefit by having more relocations. The proposed model can be applied repeatedly in each stage in a rolling horizon.

Along with nationally increasing budgets in the CV environment, a significant benefit of higher penetration rates of CV technology-equipped vehicles exists. Higher delay savings would make transportation authorities willing to relocate sensors, and lower delay savings would make them reluctant to relocate. Projects for each year can be strategically budgeted when estimated penetration rates are available. More relocation cost can be assigned to a target year with a minimum penetration rate.

A number of possible future research directions exist to extend this topic, including submodularity, moving onboard DSRC units, other metaheuristic approaches, and bilevel optimization:

- **Submodular property.** This study tested dynamic relocation models against stationary models. However, future studies may simplify the relocation to a dynamic approximate function or truncate the nonsubmodular term and express as the submodular function to solve the problem with greedy heuristics.

- **Moving onboard DSRC units.** A possible solution would be to take advantage of onboard DSRC units of the vehicles utilizing vehicle-to-vehicle communication for increased congestion coverage length and message broadcast range. By doing so, the system would remain portable with a requirement of only one roadside DSRC unit at the sight of the congestion area, which would engage all vehicles carrying onboard DSRC units to participate in traffic data acquisition and transportation of useful traffic safety information back to travelers. Then the problem would be essentially solving the distribution of these CV vehicles with DSRC units instead of roadside units.

- **Alternative metaheuristics.** Metaheuristics are alternative methods for signal control and sensor location. However, to explicitly model the dynamic sensor relocation in a close form, we used a metamodel and relaxation method to solve the problem. We can formulate more accurate metamodels, where the delay savings estimations are improved.

- **Bilevel optimization.** In the future study, two possible bilevel optimizations can be used in the dynamic sensor location problem with relocation. First, signal control can be solved in the lower level against sensor location in the upper level. Second, flows can be adjusted to a new user equilibrium and the process repeated until both flows are at equilibrium and signal timings are optimal given the flows. The bilevel optimization needs to estimate the minimum increase in departure flow with respect to the marginal increase in green time. We can test the impact of some predefined increase rate in departure flow in the relocation decision on sensor installation.

### Appendix A. Signal Control Strategy Under CV Technology

Under current signal control (with the help of the TransModeler simulator), we can observe that queues reach some portion of the length of a link. Each intersection $i$ has four directional links $a_i(h = 1, 2, 3, 4)$ with the entry flow $u_{ia}(h)$ at time $k$. By installing one roadside sensor on an intersection, we can manage the queue in four directions, especially for a moment that has a queue more than threshold $\gamma_i$. Each arc has a flow $u_{ia}(h)$ with three directions (left, straight, right) that are assigned to phase $m$. Once the link $a_{i,1}$ has a queue more than threshold $\gamma_i$, during time period $k (\geq \sigma_{a,1})$, the roadside sensor detects the queue and an alternative signal control strategy is activated to allocate the green time toward to queued direction. We modify the upstream signal without any changes in the critical intersection. The queues in four directions are considered in the allocation of green time with the green time allocation weight $\gamma_{a,b}$ for link $a$ of intersection $i$ to direction $h$.

Let $G_{i,m}^{\min}$ be the minimum green time, and let $G_{i,m}(h)$ be the green time for phase $m$ of intersection $i$ at time $k$. Our strategy is to distribute available green time (lost green time $G_i^2(k)$—minimum $G_i^{\min}$) from the current phase ($m = 2$) to green time $G_{i,m}(h)(m \neq 2)$ in other phases ($m = 1, 3, 4$). However, by blocking phase 2, the flow $u_{i,1}^1(h)$, $u_{i,2}^1(h)$, $u_{i,3}^1(h)$, $u_{i,4}^1(h)$ will be in delay. Note that the arc toward $a_1$ was blocked, so flows $u_{i,1}^2(h), u_{i,2}^3(h), u_{i,3}^1(h), u_{i,4}^4(h)$ cannot move anyway. The expected delay savings for one direction $E[\phi_{i,a,1}^m]$ are estimated as

$$\text{Max}E[\phi_{i,a,1}^m] = -G_i^2(u_{i,2}^1 + u_{i,3}^1 + u_{i,4}^1 + u_{i,4}^3) + G_i^1(u_{i,2}^1) + G_i^1(u_{i,3}^1) + G_i^1(u_{i,4}^1) + 0 \times (u_{i,1}^2 + u_{i,2}^3 + u_{i,3}^1 + u_{i,4}^4), \quad (A.1)$$

where $G_i^2$ is replaced by $G_i^{\min} + G_i^1 + G_i^1$. Assuming that $G_i^1$ and $G_i^1$ are the critical movements, $G_i^2$ and $G_i^1$ are equal to 0, and Equation (A.1) is simplified. With fully directional properties, we can get the expected total delay savings for intersection $i$:

$$E[\phi_{i,a}^m] = \left\{ \gamma_{i,a}(G_i^2 - G_i^{\min}) \sum_{h \in \{1\}}^4 (u_{i,a,h}^4 - u_{i,a,h}^3) + \gamma_{i,a}(u_{i,a}^2) \sum_{h \in \{1\}}^4 (u_{i,a,h}^1 - u_{i,a,h}^2) \right\}$$
All parameters \( \phi_i = \{ \gamma_i, \alpha_{i,j}, G^m_i, G^{\min}_i, u_i \} \) for delay estimation \( \mathbb{E}[\phi_i] \) will be known in advance through simulation. The green time allocation with amount of \( G^m_i - G^{\min}_i, G^{\min}_i \) may cause drivers to change their original route and result in an increase in \( u_i \). We assume that the phasing and cycle time for each intersection are given. The procedure iteratively sets signal timings at each intersection to reduce network delay.

Delay savings estimated for each intersection \( i \) are used as input for decision making on a set of optimal sensor locations.

Appendix B. Simulation-Driven Metamodel

The parameters of the metamodel in this section are based on the signal control for each intersection \( i \) in Appendix A. In the sensor location problem, the decision variable \( x_i \) for locating sensors at intersection \( i \) is used to represent associated expected delay savings \( \phi(x_i) \).

The signal influence on a transportation network can be divided by two partitions: the domain \( \Gamma \) into two subdomains \( \Omega^x \) (with sensors) and \( \Omega^{\rm{no}} \) (without sensors), and with interface \( \Gamma \) such that \( \Omega^x \cup \Omega^{\rm{no}} = \Gamma \). \( \Omega^x \) presents direct effects on controlled intersections with redistributed green times that embed microscopic models, and \( \Omega^{\rm{no}} \) presents indirect effects on other intersections due to user equilibrium and reduction of green time.

The effects of signal control at \( x_{i(1)} \) and \( x_{i(2)} \) on the transportation network delay \( \psi \) are not additive. Let upstream intersection 1 and downstream intersection 2 each have four directional links. Then dependency structure between the link’s upstream and downstream boundary conditions will have interactions. Therefore, the unit contribution of \( x_{i(1)} \) on \( \psi \) is a function of \( x_{i(2)} \).

The impact of a few intersection signal changes on the whole network may not be high when we have a very high penetration rate. To incorporate the indirect influence of \( \Omega^x \) on \( \Omega^{\rm{no}} \), we use a metamodel. For simplicity of the stochastic dynamic relocation model introduced in the next chapter, we keep the complexity of the model at a two-way intersection. In fact, green phases are more likely to be distributed to a critical direction. Let \( x = [x_1, \ldots, x_i, x_j] \in N \) be a selection of sensors for signal controlling purposes. We present a metamodel combining individual intersection delay savings and network effects as a generalized linear function:

\[
\psi(x) = \alpha_1 \phi(x_1) + \alpha_2 \phi(x_2) + \cdots + \alpha_i \phi(x_i) + \epsilon_{(1)(2)} + \cdots + \epsilon_{(i-1)(i)}.
\]

The partial least squares method is used to find the best coefficients and minimize the sum of squared errors \( \sum_i \epsilon_{(i-1)(i)} \).

Let \( x = [x_1, x_2] \) be a set of controls on intersections and other links and intersections \( i \in \mathcal{I} \) on the network. By running the proposed simulation optimization model, we can output total delay as an effect of control \( \psi(x) \). We present \( \psi(x) = x_1, x_2 \) as the sum of direct effect \( \alpha_1^r \phi(x_1), \alpha_2^r \phi(x_2) \) and indirect effect \( \beta_{i(1)(2)}(\phi(x_1) \phi(x_2)) \); then the modified delay function is as follows:

\[
\psi(x) = x_1 \phi(x_1) + x_2 \phi(x_2) + \sum_{i=3}^I \alpha_i^r \phi(x_i) + \sum_{i=3}^I \beta_{i(1)(2)}(\phi(x_1) \phi(x_2)) + \epsilon_{(1)(2)} + \cdots + \epsilon_{(i-1)(i)}.
\]

where \( \alpha_i^r \phi(x_i) \) is delay on intersection 1 equivalent to \( \mathbb{E}[\phi(x_i)] \) as a direct effect of signal control on intersection 1. The indirect effect \( \beta_{i(1)(2)}(\phi(x_1) \phi(x_2)) \) can be expressed as the impact by main control on other intersections \( i \), where \( \beta_{i(1)(2)} \) is the parameter from intersection 1 and intersection 2 to vicinity of the intersection controller that has sensors installed. With calculated \( \alpha_i^r \) (from Appendix A) and \( \phi(x_i) \) (from simulation run), we estimate \( \beta_i^r \) by subtracting \( \alpha_i \times \phi(x_i) \) from \( \phi(x_i) \). The magnitude of \( \alpha_i \) and \( \beta_i \) present the direct and indirect effects.

For example, let intersection 1 be equipped with a sensor to detect any queue from four direction links. The optimal green time allocation results in 91 seconds of delay savings \( \psi(x = x_i) \), caused by a direct influence of the signal change. From the simulation result, we observe a total of 98 seconds of delay savings; then we estimate \( \alpha_i \phi(x_i) \) and \( \epsilon_{(i-1)(i)} \) and conclude that 7 seconds of delay savings were caused by indirect influence. Parameters on delay savings of all possible combination of intersections are considered and fed as an input to the proposed sensor location problem. More examples that present main optimal locations based on different numbers of sensors and optimization technique are illustrated in Section 5.2.

In this paper, \( \beta_{i(1)(2)} \) implicitly considers users’ route change behaviors. This study can be extended to consider the stochastic user equilibrium, which assumes travelers do not have perfect information concerning network attributes and they perceive travel costs in different ways. However, this bilevel extension approach will require significant computational cost in the iteration process for convergence (Sheffi 1985). The current study focuses on optimal sensor locations considering users’ routing choices through simulation optimization.

Appendix C. Nonsubmodular Property in Dynamic SLP

We prove that the SLP proposed in this study does not preserve a submodular property. We define the submodular property first, and then present a counterexample.

**Definition C.1.** A set function \( f : 2^X \subseteq \mathbb{R} \) is a function assigning a real value to every sensor location subset \( x \subseteq X \) of a given ground set \( X \). We can consider a finite ground set \( x = [x_1, x_2, \ldots, x_i] \).

**Definition C.2.** A set function \( f \) is nonnegative if for every \( x \subseteq X \) we have \( f(x) \geq 0 \).

**Definition C.3.** A set function \( f \) is normalized if \( f(\varnothing) = 0 \).
Using these definitions, we present a counterexample of the submodular property in the sensor location problem. To investigate the consequence of changes in the location $i = 1$ on the network delay savings, we can take the first derivative to obtain the marginal effect as a composite coefficient estimate:

$$\beta_1 x_2 + \alpha_1.$$ We formalize the first-order interaction model with a submodular property. Let $\phi$ be the best solution with $c$ sensors as a subset of $\Pi$, and let $\upsilon$ be the best solution with $c + 1$ sensors as a subset of $\Upsilon$.

**Theorem C.1.** Given a set function $f$, a sensor location set $x \subseteq X$, and an element $x \in X$, the marginal contribution of $x$ to $x$ with respect to $f$ is defined as

$$f_x(x) = f(x \cup \{x\}) - f(x).$$

Let $V$ be a finite set for $\forall \phi \subset \upsilon \subset V$. The following function cannot be satisfied, and the function is nonsubmodular:

$$f(\phi) + f(\upsilon) \geq f(\phi \cap \upsilon) + f(\phi \cup \upsilon) \quad \text{(C.1)}$$

**Proof.** Counterexample: Order the sensor locations in decreasing order of their solutions: $[x_1, x_2, x_3, \ldots, x_i]$ and $[\alpha_1 > \alpha_2 > \alpha_3 > \cdots > \alpha_i]$. We start with one sensor at $x_1$. By adding one sensor $x_2$ to the network, the marginal delay savings are $\beta_1 x_2 + \alpha_1$ as a function of $f(\phi)$. In the next step, adding one more sensor $x_3$ produces marginal delay savings of $\beta_1 x_3 + \beta_2 x_3 + \alpha_3$ as a function of $f(\upsilon)$. In the peak hours, when several consecutive intersections are congested, $\beta_1 x_3 + \beta_2 x_3$ are expected to be higher than $\beta_1 x_2$. Therefore, when this counterexample satisfies, the marginal effect of one additional sensor does not always present diminishing return, and the submodular modularity does not exist:

$$\beta_1 x_3 + \beta_2 x_3 + \alpha_3 \geq \beta_1 x_2 + \alpha_1. \quad \square \quad \text{(C.2)}$$

As described in Park and Haghani (2015), a higher sensor cost results in the deployment of fewer sensors, each sensor being relocated more frequently. In this paper, we assume that the labor fee, data loss, and transportation cost of relocation are not higher than the sensor cost. Then, benefit of relocation is more when we have fewer sensors.
References


Vermont Agency of Transportation (2016) Transportation data management system: Turning Movement Database. Chittenden County Regional Planning Commission, Winooski, VT.


