**Submitted to *Operations Research***
**manuscript OPRE-2014-02-079**

**Authors are encouraged to submit new papers to INFORMS journals by means of a style file template, which includes the journal title. However, use of a template does not certify that the paper has been accepted for publication in the named journal. INFORMS journal templates are for the exclusive purpose of submitting to an INFORMS journal and should not be used to distribute the papers in print or online or to submit the papers to another publication.**

# The Post-Disaster Debris Clearance Problem Under Incomplete Information

Melih Çelik
Department of Industrial Engineering, Middle East Technical University, Ankara 06800, Turkey cmelih@metu.edu.tr,

Özlem Ergun
Department of Mechanical and Industrial Engineering, Northeastern University, Boston, Massachusetts 02115, USA
o.ergun@neu.edu,

Pınar Keskinocak
H. Milton School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332, USA
pinar@isye.gatech.edu

Debris management is one of the most time consuming and complicated activities among post-disaster operations. Debris clearance is aimed at pushing the debris to the sides of the roads so that relief distribution and search-and-rescue operations can be maintained in a timely manner. Given the limited resources, uncertainty, and urgency during disaster response, efficient and effective planning of debris clearance to achieve connectivity between relief demand and supply is important. In this paper, we define the stochastic debris clearance problem (SDCP), which captures post-disaster situations where the limited information on the debris amounts along the roads is updated as clearance activities proceed. The main decision in SDCP is to determine a sequence of roads to clear in each period such that benefit accrued by satisfying relief demand is maximized. To solve SDCP to optimality, we develop a partially observable Markov decision process (POMDP) model. We then propose a heuristic based on a continuous-time approximation, and further reduce the computational burden by applying a limited look-ahead on the search tree and heuristic pruning. The performance of these approaches is tested on randomly generated instances that reflect various geographical and information settings, and instances based on a real world earthquake scenario. The results of these experiments underline the importance of applying a stochastic approach and indicate significant improvements over heuristics that mimic the current practice for debris clearance.

*Key words*: Debris clearance, online and stochastic networks, continuous-time approximations, partially observable Markov decision processes

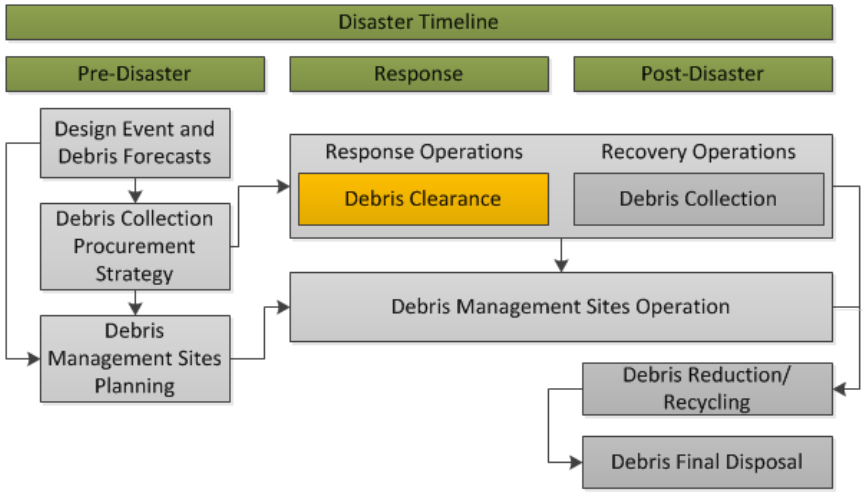


**Figure 1** Elements of a debris management plan during a disaster timeline

## 1. Introduction

In recent years, there has been a significant increase in the number and impact of natural and man-made disasters worldwide (Center for Research on the Epidemiology of Disasters 2013). An important issue faced by local and federal authorities is the management of post-disaster debris, which can be defined as any type of waste generated by disasters, including vegetation (e.g., trees, limbs), construction and demolition waste (e.g., concrete, roof material), municipal solid waste (e.g., household garbage), white goods, motor vehicles, dead animals, and hazardous chemicals. Management of post-disaster debris deserves significant attention due to the size, cost, and complicated structure of the activities involved.

Volume of post-disaster debris may sometimes be equivalent to multiple years of solid waste that would normally be generated. For instance, the debris generated by Hurricane Katrina in 2005 is estimated to be more than 100 million cubic yards (CY), which is enough to fill a football field up to 16 miles high (Congressional Research Service 2011). Similarly, Hurricane Andrew in 1992 generated around 43 million CY of debris, the March 2011 cascading disasters in Japan produced more than 35 million CY (Japan Times 2011), and the 2010 Haiti earthquake resulted in no less than 20 million CY (Washington Post 2010). Debris also contributes significantly to disaster management costs by accounting for an estimated 27% (FEMA 2007).

In the aftermath of a disaster, debris hampers effective search-and-rescue and relief distribution efforts by blocking roads and preventing access to disaster sites, highlighting the significance of timely debris clearance and removal efforts. Motivated by this, FEMA (2007) provides a Public Assistance Debris Management Guide to help public and authorities deal with the impact of post-disaster debris. Adapted from FEMA (2007), Figure 1 gives the elements of a debris management

plan during different phases of the disaster. Before the disaster occurs, a set of potential disaster scenarios are considered to forecast the resulting amount and distribution of debris. Equipment procurement and workforce allocation plans are made based on these expectations. Locations and capacities of facilities such as debris processing sites and landfills are also determined in this stage.

In the first few days following the disaster (response phase), the resources are often extremely limited and the urgency of delivering aid to those in need is very high; hence, the main objective for the clearance activities is to unblock the roads by pushing the debris to the side so that search-and-rescue activities as well as relief transportation can proceed. These activities, on which this paper focuses, are referred to as *debris clearance*. In the recovery phase, *debris collection* activities consist of transportation of the debris from the disaster area to collection sites for further processing or disposal. Lastly, *debris recycling/disposal* consists of processing (e.g., burning, crushing, grinding) debris at predetermined facilities followed by further processing for recycling and/or disposal by transportation to landfills.

FEMA (2007) provides guidelines related to post-disaster debris management outlining the activities and resources as well as assigning responsibilities. However, it takes a descriptive rather than a prescriptive approach. In particular, for the debris clearance stage, it lacks guidance on prioritizing roads in a systematic way in order to complete the clearance activities in a timely manner. The lack of operational and tactical guidance on how to carry out debris management activities motivates our research towards developing mathematical models for decision support.

In the debris clearance stage, road prioritization decisions are generally made by local or federal emergency management authorities. The main decision in this stage is to determine a list (sequence) of roads to clear with the goal of enabling immediate response efforts. (FEMA 2007). More detailed scheduling decisions, such as clearance team assignments and routing of clearance equipment, are made a posteriori based on this sequence.

In determining the clearance schedule, it is important to consider the fact that the resources required to clear the roads depend on the amount of debris on each road segment. Debris amounts on different roads might be hard to estimate. For instance, while Jean-Louis and Hilaire (2010) provided an aggregate estimate of 20-25 million CY of debris following the Haiti earthquake, no estimates on individual roads or regions were available. In certain cases, even obtaining aggregate estimates might be a challenge. As an example, aggregate debris estimations were deemed impossible for the 2013 tornadoes in Oklahoma due to the lack of debris estimation models for tornadoes. Given these challenges in estimating the debris amounts, our model assumes that these amounts are stochastic (for a subset of roads). By considering this uncertainty, we define the *stochastic debris clearance problem* (SDCP), in which the main decision is to dynamically create a clearance

sequence for debris-covered roads in the network so that demand and supply for relief can be connected in a timely manner. We model SDCP as a multi-period problem where clearance decisions are made at the beginning of each period (e.g., a day, or a shift of eight or 12 hours) and there is limited clearance capacity (resources) in each period, determined by workforce and equipment constraints. To the best of our knowledge, our work is the first in the literature that provides a systematic approach for prioritizing road clearance considering the uncertainty in the resources required to clear the debris.

The consideration of uncertainty along with the multi-period nature of the problem add significant complexity and make exact methods impractical for real-sized instances. To overcome this, we propose a heuristic that relaxes the multi-period nature of the decision making process. We also develop solution methods that incorporate regional information updates, in which case the clearance resource requirements on the roads that become reachable and/or observable in a region lead to adjustments of those on the remaining roads in the same region. We test the performances of these approaches and compare them to a set of "benchmark" approaches (motivated by current practice) on grid/ring networks and various instances of a disaster scenario. Our computational results demonstrate the effectiveness of the models and heuristics, and provide insights on how road prioritization decisions are affected by factors such as the structure of the network and the extent of uncertainty. In particular, we show that our approach provides significant improvements over those ignoring uncertainty or assigning road priorities without considering the network structure or debris amounts. We also characterize the situations where employing regional updates improve the solutions.

The remainder of the paper is organized as follows: §2 provides an overview of the literature relevant to the problem and underlines the main contributions of this work. §3 defines SDCP formally and introduces the partially observable Markov decision process (POMDP) model to solve it. §4 describes the developed heuristics to solve larger-sized problems, whereas §5 discusses structural results for the POMDP model and the proposed continuous-time heuristic. §6 provides the results of the computational experiments performed on randomly generated grid and ring networks and instances generated based on an earthquake scenario. We conclude the paper in §7 and discuss potential future research directions.

## 2. Literature Review

Significant contributions have been made by the operations research and management science literature to the field of disaster management. Altay and Green (2006) provide a detailed review and classification of these studies and regard debris removal as a major disaster response/recovery problem and an important future research direction. Caunhye et al. (2012) focus on preparedness

and response stages of disasters, also pointing out debris removal among future research directions. For recent reviews on humanitarian logistics, please see Apte (2009) and Çelik et al. (2012).

The work in this paper is related to a number of different streams in the literature including policy-related qualitative studies on debris management, network construction/repair and relief distribution, online and stochastic networks, and partially observable Markov decision processes (POMDPs).

Brown et al. (2011) review the post-disaster waste management literature and conclude that the literature is mainly comprised of disaster waste management plans or guidelines and isolated case studies. Examples of such guidelines include FEMA (2007) for the United States, Solis et al. (1996) for Canada, and United Nations Office for the Coordination of Humanitarian Affairs (2011) for the United Nations. There are also case studies regarding debris management decisions in specific disasters. These include Mendonça and Hu (2006) and Luther (2008) on Hurricane Katrina and Srinivas and Nakegawa (2009) on the 2004 Indian Ocean Tsunami. Swan (2000) and Ekici et al. (2009) take a location-independent perspective and provide analysis on issues such as allocation of responsibilities and administrative procedures. As Fetter and Falasca (2011) also point out, despite the increasing interest in the policy-related aspects of debris management, very few studies propose decision support systems and quantitative tools. The work in this paper tries to bridge this gap.

The debris clearance problem has been modeled by Stilp et al. (2012) considering deterministic clearance resource requirements. The authors show the NP-hardness of the problem even under very simple network structures, such as a network with a single supply and a single demand node. They formulate a mixed integer programming model and propose heuristic methods. Aksu and Özdamar (2014) present a path-based mixed integer programming formulation for the same problem. The work in this paper considers uncertainty in debris volumes over the network and the corresponding clearance times. Through structural analysis and computational experiments, we show that policies that ignore uncertainty by assuming expected clearance times and solving a deterministic model can lead to poor solution quality.

A problem relevant to debris clearance is that of restoring a disrupted infrastructure network after a disaster. Perrier et al. (2008) consider the problem of routing and scheduling of snow plowing equipment in urban areas where each road segment has a pre-assigned priority level. Similarly, Şahin et al. (2013) consider debris clearance in the response stage using predetermined road priorities. In general, the decision of prioritizing roads is not trivial due to possible differences between type, severity, and location of the disaster as well as the requirements of connectivity among communities and critical facilities. Therefore, in our model, instead of taking the priorities as inputs, we include them as decisions. Using computational experiments, we also show that policies that use pre-assigned road priorities may lead to significantly worse results.

A number of studies consider the network repair and relief distribution decisions simultaneously (Yan and Shih 2009, Matisziw et al. 2010, Chen et al. 2011, Çavdaroğlu et al. 2011, Nurre et al. 2012, Kibar and Salman 2013, Nurre and Sharkey 2014). In these cases, it is possible to repair edges that are not physically reachable by the repair team due to damaged edges. This is not the case in SDCP, as debris clearance teams can only reach those roads (edges in the network) which are connected (i.e., through a sequence of clear roads) to their current location. Furthermore, as opposed to the deterministic repair (clearance) times in these studies, the work in this paper assumes uncertainty in edge repair times.

SDCP is also relevant to other construction/repair problems that aim to establish connectivity on a network, all of which assume deterministic requirements for road clearance (construction/repair) resources. In the incremental network design problem (e.g., Baxter et al. 2014, Kalinowski et al. 2014), where the aim is to clear (construct/repair) roads to establish supply and demand connectivity in a timely manner, the decision in each period is to determine a road to clear, regardless of the resource requirement. The flowtime network construction problem (Averbakh and Pereira 2012), in which the objective is to find a clearance (construction) schedule to minimize the total time to reach all demand nodes, can be considered as a special case of SDCP with deterministic clearance times and unit demand at each node. The emergency path restoration problem (Averbakh 2012) aims to assign multiple teams to roads to clear. However, the literature on this problem is limited to path networks.

In most online/stochastic network problems, edge travel times are known and one or more servers provide service to demand nodes with known demand arrival times; however, some uncertain characteristics of the demand (location, quantity, destination to transport the load, etc.), are revealed only at the time of demand occurrence. Studies on online problems include Jaillet and Wagner (2006, 2008), Angelelli et al. (2007), and Jaillet and Lu (2011) on the traveling salesman problem (TSP), Ausiello et al. (2008) on the prize-collecting TSP, Anagnostopoulos et al. (2004) on the facility location problem, Bent and Hentenryck (2007) on the vehicle routing problem (VRP), and Ascheuer et al. (2000) and Feuerstein and Stougie (2001) on the dial-a-ride problem. While how uncertainty is revealed in these problems is similar to SDCP, the source of uncertainty being nodes rather than edges causes significant differences in the decisions made and, hence, also in the solution methods. The main decisions in these problems involve where and how long to wait, to accept or reject an incoming demand, etc., as opposed to edge sequencing decisions in SDCP.

An aspect of the stochastic network problems relevant to SDCP is the incorporation of learning as uncertainty is revealed. Learning algorithms are proposed by Littlestone and Warmuth (1989), Bent and Hentenryck (2004a,b) Kalai and Vempala (2005), Powell and Frazier (2012), Ryzhov and Powell (2011), and Hentenryck et al. (2010). These approaches are quite general and can be

applied to a wide range of stochastic network problems. However, they either involve problems that are encountered by the decision maker multiple times (therefore learning occurs each time a new problem instance is encountered) or assume that no a priori information exists.

Stochastic network problems in which uncertainty arises from edge lengths (travel times) include robotic mapping and the Canadian traveller problem. In robotic mapping, a robot uses its error-prone sensors to learn about and map the network structure of an unknown environment as quickly and accurately as possible. A detailed review of this problem is given by Thrun (2002). While the learning aspect of the problem bears similarities to SDCP, robotic mapping aims to learn the network as quickly as possible, while SDCP also needs to consider demand satisfaction (by connecting nodes of relief demand and supply).

Various versions of the Canadian traveller problem exist, of which the most relevant to SDCP is studied by Karger and Nikolova (2008), where an agent travels from a source node to a destination node through a shortest path. Edge lengths are uncertain and can have different values, each with an associated probability. Uncertainty is revealed when the agent reaches the edge, which is also the case in SDCP. Despite the similarity of uncertainty and how it is revealed, consideration of a single source and sink node as opposed to the case of multiple supply and demand nodes in SDCP, as well as the focus on minimizing the travel distance instead of relief distribution results in significant differences in the modeling, algorithms, and structure of the optimal solutions.

Partially observable Markov decision processes (POMDPs) provide a general framework to model sequential decision problems in environments where the system state is not completely observable. A review of POMDP models is given by Ross et al. (2008), with a focus on online planning approaches, which find a good local policy for the current belief state, usually based on a limited look-ahead, update information as new observations are received, and repeat the process. In contrast, offline POMDP approaches find optimal actions to perform in every belief state, requiring considerable computation time. While our exact POMDP model uses an offline approach, we make use of online approaches in our heuristics for the SCDP.

The contributions of this paper are three-fold. First, we introduce SDCP, a problem applicable in many real-life post-disaster situations. As in the case of most online and stochastic problems, there is a key trade-off between taking the actions that bring the most immediate benefit and learning the network structure to take more informed actions in the future. In SDCP, the trade-off is between clearing the most beneficial roads in the next period and learning the debris volume throughout the network. Second, to resolve this trade-off and overcome the challenges of uncertainty and the periodic nature of the problem, we propose a heuristic procedure that relaxes the periodicity in the decision making scheme to reduce the number of possible actions. Our approach can also make use of a regional information update scheme, which is motivated by the fact that the disaster is expected

to have a similar impact on geographically close areas. In this update scheme, debris information obtained for roads in a certain region leads to updated belief probabilities for the clearance resource requirements in the same region. The proposed solution approaches are not problem-specific and can be applied to online stochastic problems on networks where decisions need to be made at the beginning of each period. Examples of such problems include power systems repair and post-disaster relief distribution with stochastic demands. Lastly, we compare the proposed approaches to their counterparts, which (i) ignore uncertainty in clearance resource requirements and (ii) mimic current debris clearance practice. Through computational experiments, we classify the cases where the difference between the benefits obtained by a stochastic approach is more significant.

## 3. Problem Description and Model

### 3.1. The Stochastic Debris Clearance Problem (SDCP)

The debris clearance problem aims to establish connectivity between supply and demand points in a disaster area by clearing debris-blocked roads over a fixed planning horizon, so that the flow of relief commodities or services between these points can be established. The planning horizon is divided into discrete time periods (e.g., days or eight-hour shifts), and the main decision at the beginning of each period is to determine an ordered sequence of roads to clear for the upcoming period subject to a capacity on how much debris can be cleared. In practice, detailed scheduling of the clearance teams is made a posteriori based on this sequence.

The notation we use throughout the remainder of this section is given in Table 1. We model SDCP on an undirected network $G = (N, E)$. Clearance activities may start from a number of different locations in the network, depending on where clearance resources are initially stored or prepositioned. The planning horizon is divided into discrete time periods, given by the set $T$. The node set is partitioned into subsets $N_S$ (supply), $N_D$ (demand), and $N_T$ (transshipment). Supply points may consist of critical facilities such as fire stations, hospitals, shelters, distribution centers for relief supplies, etc. Demand points are population centers consisting of people in need of relief commodities or access to services offered by the supply points. At the beginning of each period $t \in T$, the edge set is partitioned into two subsets, $B_t$ and $U_t$, which denote the sets of blocked and unblocked edges, respectively. Without loss of generality, these sets are assumed to be known. We assume that clearance units cannot traverse debris-blocked roads, i.e., for a clearance team to reach a debris-blocked edge, there needs to be a cleared path to that edge from the team's current location. In line with debris clearance practice and the literature on network construction/repair (e.g., Averbakh 2012, Baxter et al. 2014, Nurre and Sharkey 2014), we also assume that the transportation times on the cleared edges are negligible, as these are much smaller than the times required to clear the edges.

**Table 1** Notation

| **Index sets** | |
|---|---|
| $T$ | Periods |
| $N$ | Node set |
| $N_S$ | Supply nodes |
| $N_T$ | Transshipment nodes |
| $N_D$ | Demand nodes |
| $E$ | Edge set |
| $U_t$ | Unblocked edges at beginning of period $t$ |
| $B_t$ | Blocked edges at beginning of period $t$ |
| $RB_t$ | Blocked edges that are reachable by a clearance team at beginning of period $t$ |
| $UB_t$ | Blocked edges that are not reachable by any team at beginning of period $t$ |
| $DL_{ij}$ | Set of possible clearance resource requirement levels for blocked edge $(i,j)$ |

| **Parameters** | |
|---|---|
| $b_i$ | Benefit accrued in node $i \in N_D$ per unit demand satisfied in each period |
| $k_t$ | Total clearance capacity in period $t \in T$ |
| $FC_{ij}$ | Flow capacity on edge $(i,j)$ if unblocked |
| $W_{ij}^t$ | Amount of resources required to clear edge $(i,j) \in B_t$ (Random variable for $(i,j) \in UB_t$, known for $(i,j) \in RB_t$) |
| $\overrightarrow{W_t}$ | Vector of resource requirements on each edge in $B_t$ in period $t \in T$ |
| $\beta_{ij}^{dt}$ | Probability of observing debris level $d \in DL_{ij}$ on edge $(i,j) \in UB_t$ based on belief at beginning of period $t \in T$ |
| $\overrightarrow{Y_t}$ | Vector consisting of probabilities of observing each debris level on each edge in $UB_t$ based on belief at beginning of period $t \in T$ |
| $RC_t$ | Remaining clearance capacity at the beginning of period $t \in T$ after ongoing clearance from period $t-1$ is completed |
| $rs_{it}$ | Remaining supply capacity in node $i \in N_S$ at beginning of period $t \in T$ |
| $\overrightarrow{RS_t}$ | Vector consisting of remaining supply in each supply node at beginning of period $t \in T$ |
| $rd_{it}$ | Remaining unsatisfied demand in node $i \in N_D$ at beginning of period $t \in T$ |
| $\overrightarrow{RD_t}$ | Vector consisting of remaining demand in each demand node at beginning of period $t \in T$ |
| $\gamma^t$ | Remaining resource requirement for the partially cleared edge (if any) at the end of period $t \in T$ |

At the beginning of each period $t \geq 2$ in $T$, the parameters $rd_{it}$ and $rs_{it}$ denote the remaining unsatisfied demand of node $i \in N_D$ and remaining supply capacity of node $i \in N_S$, respectively. $rd_{i1}$ and $rs_{i1}$ denote the initial demand and supply at node $i$, respectively. Once a demand node becomes connected to a supply node, it starts accruing benefits proportional to the amount of demand satisfied in the remainder of the planning horizon. We assume that once resources (e.g., commodities or service) are assigned to satisfy the demand in a certain node, that assignment

remains the same for the remainder of the horizon, i.e., those resources cannot be assigned to a different demand node in a future period. Per each unit of demand satisfied, demand node $i$ accrues $b_i$ units of benefit in each period. The objective in SDCP is to maximize the total (expected) benefit during the planning horizon. The benefits being accrued over time motivates connecting the supply nodes to the demand nodes as quickly as possible. The problem can be equivalently modeled using a penalty for not satisfying the demand in a given period. It is also possible to extend the models to incorporate changes in supply capacity (e.g., due to new commodity arrivals or service capacity increase) and demand (e.g., due to population movements) by introducing additional variables. Throughout this paper, we assume a single relief commodity or service to be offered. However, all the models and solution methods can be extended to the case of multiple commodities by introducing a commodity index and incorporating it into supply and demand parameters.

We incorporate uncertainty into the problem by assuming resource requirements to clear the roads (that is, the debris amounts) to be stochastic for a subset of blocked roads. While the assumption of deterministic clearance times is valid in cases where it is possible to get good debris amount estimations (e.g., smaller hurricanes), estimating debris amounts and corresponding resource requirements require a significant amount of demographic and structural data as well as proper damage forecast models on the disaster area, which may not be easily available. Even when such estimates are available (e.g., FEMA 2012), they are in aggregate form (e.g., for a region) and accuracy might be lost when they are disaggregated to road segments. To model uncertainty, let $RB_t \subseteq B_t$ denote the blocked edges that can be reached by a clearance unit using edges in $U_t$ and $UB_t = B_t \setminus RB_t$. We denote by $W_{ij}^t$ the amount of resources required to clear edge $(i,j) \in B_t$. We assume that $W_{ij}^t$ are known for $(i,j) \in RB_t$ and are random variables following a categorical distribution for $(i,j) \in UB_t$. For $(i,j) \in UB_t$, a discrete belief vector $\beta_{ij}^t = \left\{\beta_{ij}^{lt} : l \in DL_{ij}\right\}$ gives the set of the probabilities under the current belief state for each possible requirement level $l \in DL_{ij}$. The decision maker assumes that resource requirements on blocked edges are independent of each other. Later, we relax this assumption by introducing regional updates, where the dependence between the resource requirements in the same region is handled using an information update mechanism. Due to constraints on workforce and equipment capacity, at most $k_t$ units of resources can be allocated to debris clearance in each period $t \in T$.

Clearance decisions change the sets $RB_t$ and $UB_t$ dynamically, as illustrated on a simple example in Figure 2. Upon the clearance of edges $(0,3)$ and $(3,6)$ in period $t$, edges $(2,6)$, $(3,7)$, and $(6,8)$ become reachable at the beginning of period $t+1$. Similar to current practice, we assume that once the clearance decisions for a period are made, that is, a work order is determined, they are fixed until the beginning of the next period (Markin 2014). Therefore, information updates cannot be incorporated before the period ends.

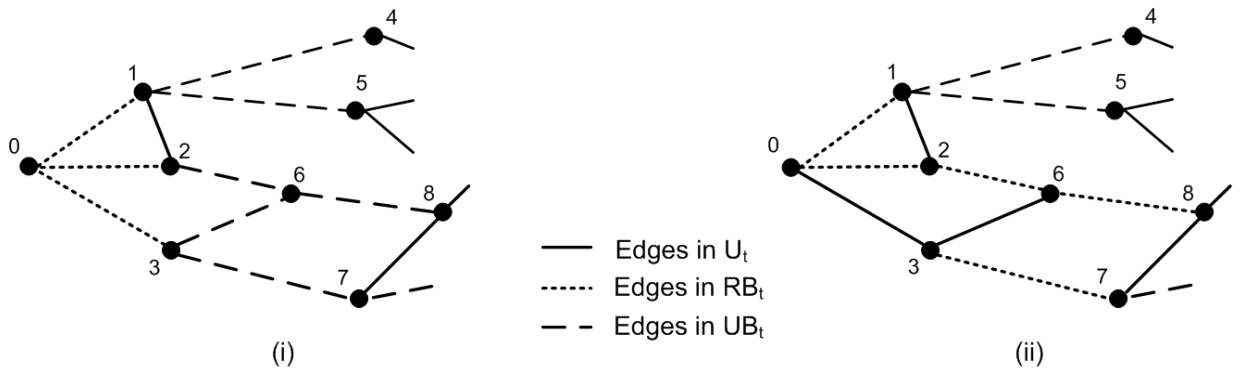


**Figure 2** The unblocked, reachable blocked, and unreachable blocked edges (i) before edges $(0,3)$ and $(0,6)$ are cleared in period $t$, (ii) after the edges are cleared in the period $t$

Since the clearance resource requirements of a subset of edges are not known in advance, this might result in partial clearance of an edge at the end of a period. In this case, the clearance of this edge has to be continued at the beginning of the next period. This implies that if there is ongoing clearance from period $t-1$ in period $t$, part or all of $k_t$ will be spent on this ongoing clearance before any new decisions for period $t$ can be made. To incorporate this into the modeling process, $RC_t$ denotes the remaining clearance capacity in period $t$ after the ongoing clearance is completed. Lastly, only an entirely cleared edge provides information about the amount of resources required to clear its adjacent edges.

### 3.2. The POMDP Model

SDCP can be formulated as a stochastic network problem. The system state $s_t$ in period $t \in T$ is denoted by the quartet $s_t = (\overrightarrow{W_t}, RC_t, \overrightarrow{RS_t}, \overrightarrow{RD_t})$, where $\overrightarrow{W_t} = \{W_{ij}^t : (i,j) \in B_t\}$, $\overrightarrow{RS_t} = \{rs_{it} : i \in N_S\}$, and $\overrightarrow{RD_t} = \{rd_{it} : i \in N_D\}$. The size of the state vector is $|B_t| + |N_S| + |N_D| + 1$.

Throughout the remainder of this section, we describe the details of the POMDP model to solve SDCP.

The state space $S_t$ in period $t \in T$ is given by:

$$S_t = \Bigg\{ (\overrightarrow{W_t}, RC_t, \overrightarrow{RS_t}, \overrightarrow{RD_t}) : W_{ij}^t \in DL_{ij} \ \forall (i,j) \in B_t;\ 0 \le RC_t \le k_t;\ 0 \le rs_{it} \le rs_{i1} \ \forall i \in N_S;$$
$$0 \le rd_{it} \le rd_{i1} \ \forall i \in N_D \Bigg\}. \tag{1}$$

Let $\overrightarrow{Y_t} = \{\beta_{ij}^{dt} : (i,j) \in B_t,\ d \in DL_{ij}\}$ be the probability vector of observing different resource requirement levels on each blocked edge under the current belief. More formally, given the belief probabilities $\beta_{ij}^{dt}$, the belief state can be represented as $bs_t = (\overrightarrow{Y_t}, RC_t, \overrightarrow{RS_t}, \overrightarrow{RD_t})$, where $\beta_{ij}^{lt} \ge 0 \ \forall l \in DL_{ij}$ and $\sum_{l \in DL_{ij}} \beta_{ij}^{lt} = 1$ for all $(i,j)$ in $B_t$, and $|bs_t| = \sum_{ij \in B_t} |DL_{ij}| + |N_S| + |N_D| + 1$.

The set of possible actions $\mathcal{A}_t$ in period $t$ is any potentially feasible sequence of edges in $B_t$ that can be cleared under the current belief state $bs_t$. Let $\mathcal{E}_t$ be the set of sequences of edges that can be cleared without violating connectivity. Then, the action space consists of all sequences $A \subseteq \mathcal{E}_t$ such that when the last edge, $e'(A)$, in the ordered set is excluded, clearance capacity constraint is strictly satisfied under minimum possible clearance requirement levels. Let $W_{ij}^{min,t}$ be the minimum possible amounts of resources required to clear the debris on edge $(i,j)$ based on the current belief, i.e., $W_{ij}^{min,t} = \min\{l : \beta_{ij}^{lt} > 0\}$. In more formal terms:

$$\mathcal{A}_t = \left\{ A \subseteq \mathcal{E}_t : \sum_{(i,j) \in A \setminus e'(A)} W_{ij}^{min,t} < RC_t \right\}. \tag{2}$$

If $RC_t = 0$, the action space consists only of the edge for which there is ongoing clearance from the previous period.

Given the current belief state $bs_t$ and a feasible action $A \in \mathcal{A}_t$, the observation space consists of: (i) all possible $O \subseteq UB_t$ that can be cleared or become reachable as a result of the action, and (ii) all possible combinations of possible resource requirements on each of these subsets. For instance, if $A = \{(0,3),(3,6)\}$ is a feasible action in Figure 2, and if none of the edges is completely cleared at the end of the current period, $O = \emptyset$. Otherwise, if only $(3,6)$ is completely cleared, $O = \{(3,6),(3,7)\}$. If both edges are completely cleared, $O = \{(2,6),(3,6),(3,7),(6,8)\}$. In more formal terms, the observation space $\mathcal{O}(bs_t, A)$ is defined by:

$$\mathcal{O}(bs_t, A) = \Bigg\{ \overrightarrow{w_t} : w_{ij} \geq 0 \ \forall (i,j) \in O,\ O \subseteq UB_t,\ \beta_{ij}^{w_{ij},t} > 0 \ \forall (i,j) \in O; \sum_{(i,j) \in E'} w_{ij} \leq RC_t;\ RB_{t+1}(E') \subseteq O \Bigg\}, \tag{3}$$

where $E' \subseteq E_{B_t}^*$ is the set of edges that are completely cleared and $RB_t'(E')$ is the set of blocked edges that became reachable after the edges in $E'$ have been cleared.

Since the decision maker assumes that clearance resource requirements on different blocked edges are independent, the probability of an observation $\overrightarrow{w_t}$, given current belief $bs_t$ and action $A$ is:

$$P(\overrightarrow{w_t} \,|\, bs_t, A) = \prod_{(i,j) \in O} \beta_{ij}^{w_{ij},t}. \tag{4}$$

Let $U_t'(\overrightarrow{w_t})$ be the set of edges that are observed to be clear of debris at the end of period $t$. When an action $A$ under the current belief $bs_t$ is taken, the resulting reward also depends on the resulting observation $\overrightarrow{w_t} \in \mathcal{O}(bs_t, A)$. Calculating the reward requires solving a maximum weighted flow problem for the subgraph of $G$ induced by $U_{t+1}(\overrightarrow{w_t})$. In more formal terms, the reward is calculated from:

$$R(\overrightarrow{w_t} \,|\, bs_t, A) = \sum_{i \in N_D} b_i \left( rd_{i1} - rd_{it} \right) + z^*(f^*, x^* \,|\, \overrightarrow{w_t}, bs_t, A), \tag{5}$$

Demand satisfied in each period keeps accruing benefits throughout the horizon. Therefore the first term in (5) captures these benefits for demand satisfied in periods prior to $t$. The second term $z^*(f^*, x^* \mid \overrightarrow{w_t}, bs_t, A)$ is determined by solving the linear program (LP1) corresponding to the maximum weighted flow problem on the subgraph of $G$ induced by $U'_t(\overrightarrow{w_t})$, where $f_{ij}$ denotes the amount of flow on edge $(i,j)$ and $x_i$ denotes the amount of demand at node $i$ satisfied in the current period:

$$(\mathbf{LP1}) \qquad \max \sum_{i \in N_D} b_i \, x_i \tag{6}$$

$$s.t. \sum_{j:(j,i)\in U'_t(\overrightarrow{w_t})} f_{ji} - \sum_{j:(i,j)\in U'_t(\overrightarrow{w_t})} f_{ij} = x_i \qquad \forall i \in N_D \tag{7}$$

$$\sum_{j:(i,j)\in U'_t(\overrightarrow{w_t})} f_{ij} - \sum_{j:(j,i)\in U'_t(\overrightarrow{w_t})} f_{ji} \leq rs_{it} \qquad \forall i \in N_S \tag{8}$$

$$\sum_{j:(j,i)\in U'_{t+1}(\overrightarrow{w_t})} f_{ji} - \sum_{j:(i,j)\in U'_t(\overrightarrow{w_t})} f_{ij} = 0 \qquad \forall i \in N_T \tag{9}$$

$$x_i \leq rd_{it} \qquad \forall i \in N_D \tag{10}$$

$$f_{ij} \leq FC_{ij} \qquad \forall (i,j) \in U'_t(\overrightarrow{w_t}) \tag{11}$$

$$x_i \geq 0 \qquad \forall i \in N_D \tag{12}$$

$$f_{ij} \geq 0 \qquad \forall (i,j) \in U'_t(\overrightarrow{w_t}). \tag{13}$$

Objective function (6) maximizes the total weighted flow from supply nodes to demand nodes. Constraints (7) through (9) ensure that the flow balance is conserved for demand, supply, and transshipment nodes, respectively. Constraints (10) bound the net flow into a demand node by the total demand of the node. Constraints (11) are the road capacity constraints, whereas (12) and (13) are the non-negativity constraints.

The expected reward under each action $A$ is the expected value of the possible rewards resulting from the observations emanating from it:

$$ER(bs_t, A) = \sum_{\overrightarrow{w_t} \in \mathcal{O}(bs_t, A)} \left( \prod_{(i,j)\in O} \beta_{ij}^{w_{ij},t} \, R(\overrightarrow{w_t} \mid bs_t, A) \right). \tag{14}$$

Based on the current belief state $bs_t$, under action $A$, and observation $\overrightarrow{w_t}$, the edge sets are updated as follows:

$$U_{t+1} \leftarrow U'_t(\overrightarrow{w_t}), \quad B_{t+1} \leftarrow E \setminus U_{t+1} \tag{15}$$

$$RB_{t+1} \leftarrow RB'_t(E'(A, \overrightarrow{w_t})), \quad UB_{t+1} \leftarrow B_{t+1} \setminus RB_{t+1}, \tag{16}$$

where $E'(A, \overrightarrow{w_t})$ is the set of edges that are completely cleared based on action $A$ and observation $\overrightarrow{w_t}$. Given these updated sets, the updated belief $bs_{t+1}(bs_t, A, \overrightarrow{w_t}) = (\overrightarrow{Y_{t+1}}, RC_{t+1}, \overrightarrow{RS_{t+1}}, \overrightarrow{RD_{t+1}})$ is obtained by:

$$\beta_{ij}^{d,t+1} = \begin{cases} \beta_{ij}^{lt}, & \text{if } (i,j) \in UB_{t+1} \\ 1, & \text{if } (i,j) \in RB_{t+1},\ l = W_{ij}^t \\ 0, & \text{if } (i,j) \in RB_{t+1},\ l \neq W_{ij}^t \end{cases} \tag{17}$$

$$RC_{t+1} = \max\{k_{t+1} - \gamma_t, 0\} \tag{18}$$

$$rs_{i,t+1} = rs_{it} - \left( \sum_{j:(i,j)\in U'_{t+1}(\overrightarrow{w_t})} f_{ij} - \sum_{j:(j,i)\in U'_t(\overrightarrow{W})} f_{ji} \right) \tag{19}$$

$$rd_{i,t+1} = rd_{it} - x_i^*, \tag{20}$$

where $\gamma_t$ is the remaining clearance resource requirement for the partially cleared edge (if any) at the end of period $t$ and $x_i^*$ is the amount of demand satisfied at node $i$, found by solving the linear program defined by (6)-(13). Equation (17) states that if a blocked edge stays unreachable after period $t$, then its belief probabilities do not change. If a blocked edge is reachable, then the belief probability is 1 for its actual amount and 0 for all other amounts. Equation (18) adjusts the starting capacity at the beginning of period $t+1$ by considering potential ongoing clearance from period $t$. Equations (19) and (20) update the remaining supply and demand quantities in the supply and demand nodes, respectively.

We set $RC_1 = k_1$ and $\gamma^0 = 0$. Based on these, $\gamma_t$ is updated as follows:

$$\gamma_t = \begin{cases} 0, & \text{if } RC_t > \sum_{(i,j)\in E'(A,\overrightarrow{w_t})} W_{ij}^t \text{ and } A = E'(A, \overrightarrow{w_t}) \\ \gamma_{t-1} - k_t, & \text{if } RC_t = 0 \\ \sum_{(i,j)\in E''(A,\overrightarrow{w_t})} W_{ij} - RC_t, & \text{otherwise,} \end{cases}$$

where the set $E''(A, \overrightarrow{w_t})$ includes edges in $E'(A, \overrightarrow{w_t})$ and the edge that is partially cleared in period $t$. In the first case, there is no partial clearance at the end of period $t$ and hence $\gamma_t = 0$. In the second case, the partial clearance from period $t-1$ consumes all the clearance capacity in period $t$, therefore $\gamma_t$ equals the remaining requirement on the partially cleared edge at the end of period $t$. In the last case, a number of edges are completely cleared, and $\gamma_t$ equals the remaining resource requirement on the partially cleared edge at the end of the period.

The POMDP model for SDCP can be solved by using Bellman equations and treating the belief states as states of an MDP model. Let $v^*(bs_t)$ denote the optimal benefit-to-go of belief state $bs_t$. Then, the following set of Bellman equations solve the POMDP model for SDCP:

$$v^*(bs_t) = \max_{A\in\mathcal{A}_t} \left\{ \sum_{\overrightarrow{w_t}\in\mathcal{O}(bs_t,A)} \prod_{(i,j)\in O} \beta_{ij}^{w_{ij},t} \left( R(\overrightarrow{w_t} \,|\, bs_t, A) + v^*(bs_{t+1} \,|\, bs_t, A, \overrightarrow{w_t}) \right) \right\}. \tag{21}$$

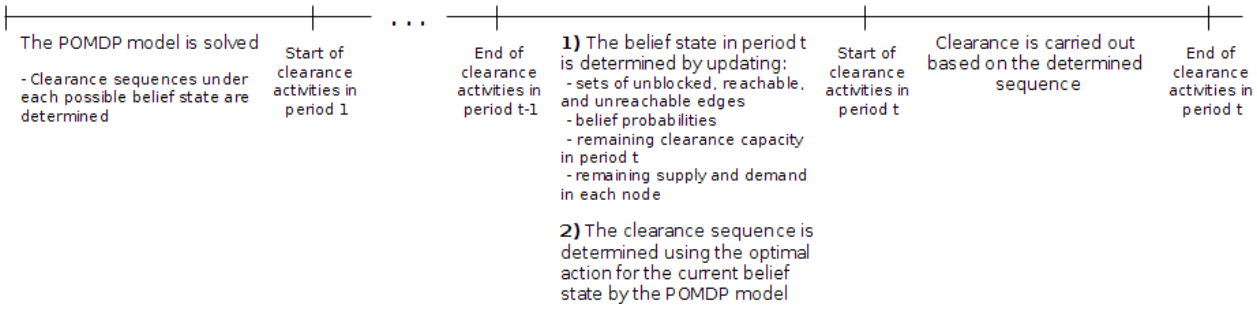


**Figure 3** Timeline of decisions and events in the POMDP model before and during period $t$

The solution process for the POMDP can be represented by a search tree in which each belief is represented by a belief node and connected to these are alternative action nodes with their corresponding rewards. Observations are represented by arcs emanating from the action nodes, each with their corresponding probabilities. Each new observation leads to an updated belief node. The optimal policy can be found by backward recursion, taking expectations over observations and selecting the action with maximum benefit-to-go. Consequently, the POMDP model is an offline (or static) solution method, that is, it needs to be solved once and finds the optimal action for each possible belief state resulting from all possible observations. Figure 3 summarizes the timeline of decisions throughout a given period.

Since the action space consists of subsets of possible edges that can be (partially) cleared in one period and the observation space consists of possible combinations of actual debris amounts on the edges that become reachable, both action and observation spaces are substantially large, even for small-sized instances. Hence, the number of possible belief states increase significantly with the number of alternative actions and possible observations, resulting in a "curse of dimensionality" (Ross et al. 2008). Nevertheless, this exact solution approach provides a framework for solving small- to mid-sized problems to optimality and for developing heuristics.

### 3.3. Extension to Regional Updates

In the above formulation, belief probabilities $\beta_{ij}^{dt}$ for the edges $(i,j)$ in $UB_{t+1}$ remain the same (see Equation (17)). Furthermore, the decision maker assumes that the clearance resource requirements on the roads are independent of each other (Equation (4)). However, in real life disaster scenarios, the clearance resource requirements on the roads in the same region might be correlated due to similar geographic and demographic properties.

To incorporate potential within-region similarities in clearance resource requirements and information updates into our model, we use a Bayesian information update approach (similar to Powell and Ryzhov 2012). The disaster area, i.e., first we assume that the edge set $E$, is partitioned into regions $\{E_1, E_2, \ldots, E_R\}$ and let $U_r^t$, $B_r^t$, and $UB_r^t$ denote the clear, blocked, and unreachable blocked

edges in region $r$ at the beginning of period $t$, respectively. Assume without loss of generality, that all roads in the set $UB_r^1$ in region $r$ have the same resource requirement levels $D=\{1,2,\ldots,N^r\}$. Suppose also that these follow a categorical distribution, that is, each resource requirement level $l\in D$ has an associated probability $\beta_{l,1}^r$ for all $(i,j)\in UB_1^r$. To preserve conjugacy while updating the parameters of the categorical distribution, we apply the Dirichlet distribution as prior (Gelman et al. 2004), where:

$$\beta_{l,1}^r=\frac{\alpha_{l,1}^r}{\alpha_{l,1}^r+\ldots+\alpha_{N^r,1}^r} \text{ for all } l=1,2,\ldots,N^r, \tag{22}$$

and $\alpha_{l,1}^r$ represents the "weight" given to resource requirement level $l$ relative to the other clearance resource requirements in the same region.

Assume that for debris level $l$, we have made $c_{lt}^r$ observations in region $r$ after clearance is complete in period $t$. Then, we let $\alpha_{l,t+1}^r=\alpha_{lt}^r+c_{lt}^r$ for all $l=1,2,\ldots,N^r$. In this update mechanism, the higher the initial $\alpha$ values, the lower the effect of new resource requirement observations on the update of the belief probabilities. Therefore while the relative $\alpha$ values determine the belief probability for each level, the nominal values determine the strength of the initial beliefs.

To incorporate the regional update mechanism into the POMDP model, Equation (17) is replaced by:

$$\alpha_{l,t+1}^r=\alpha_{lt}+c_{lt}^r \qquad \forall l\in\{1,2,\ldots,N^r\} \tag{23}$$

$$\xi_{l,t+1}^r=\frac{\alpha_{l,t}^r}{\alpha_{1,t}^r+\ldots+\alpha_{N^r,t}^r} \qquad \forall l\in\{1,2,\ldots,N^r\} \tag{24}$$

$$\beta_{ij}^{l,t+1}=\begin{cases}\xi_{l,t+1}^r, & \text{if } (i,j)\in UB_{t+1}\\ 1, & \text{if } (i,j)\in RB_{t+1},\ l=W_{ij}^t\\ 0, & \text{if } (i,j)\in RB_{t+1},\ l\neq W_{ij}^t\end{cases} \qquad \forall l\in\{1,2,\ldots,N^r\}. \tag{25}$$

This regional update mechanism can be used both in the POMDP approach described in this section and the heuristics presented in §4.

## 4. Heuristic Approaches

While the POMDP approach described in §3 can solve small to moderate sized instances of SDCP to optimality, the sizes of the action and observation spaces result in a significant computational burden. To solve larger sized problems, we propose three heuristics. The first uses a continuous time approach, ignoring the periodic nature of the clearance process (§4.1). The other two heuristics limit the action and observation spaces by reducing the size and depth of the search tree (§4.2).

### 4.1. A Heuristic Based on a Continuous-Time Approximation

In SDCP, the set of feasible actions is determined by subsets of edges that can be cleared in a period and the observation space consists of all feasible combinations of clearance resource requirements

on the unreachable edges. The size of the action space poses an important challenge on solving the POMDP model. This challenge is exacerbated by the existence of multiple periods.

Appendix EC.2 gives a number of illustrative examples of SDCP on simple networks with unit demand, resulting in the following observations:

**Observation 1** *Ignoring the stochastic aspect of the problem may lead to significant benefit losses unless the initial beliefs of the decision maker accurately estimate the actual debris amounts.*

**Observation 2** *Due to the periodic nature of the problem, optimal policies for simple network and demand structures are not easily generalizable.*

Particularly motivated by Observation 2, we propose a heuristic approach based on a continuous-time approximation of SDCP, which we denote as POMDP-C. The approximation assumes that once a debris-blocked edge is cleared, we select the next edge to be cleared immediately, without any consideration of periods or capacity. The main aim is to find an entire sequence of edges to be cleared. Once the sequence is determined, a transformation process results in a periodic policy that is capacity-feasible for each period. This relaxation reduces the size of the action space drastically, as it now consists only of blocked edges that are immediately reachable. It also shrinks the observation space significantly, since new observations are obtained only from the edges that become reachable, which are now smaller in number.

In the network optimization literature, there are several "continuous space" relaxations for finding near optimal routes for various versions of the TSP and the vehicle routing problem. Examples include Daganzo (1984, 1987a,b), Del Castillo (1998), Daganzo and Erera (1999), Francis and Smilowitz (2006), and Zhong et al. (2007). In this paper, we take a different approach by focusing on time, rather than space, in our approximation.

While definitions of the state space and the belief state space do not change from the original POMDP model (except that index $t$ now indicates that $t-1$ edges have been cleared so far), the action space now consists of blocked edges that are reachable. More formally, Equation (2) becomes:

$$\mathcal{A}_t = RB_t. \tag{26}$$

Since the action does not clear any unreachable blocked edges, new observations only arise from edges that become reachable after an edge is cleared as a consequence of the action. If $A \in \mathcal{A}_t$ is the edge that has just been cleared, then Equation (3) becomes:

$$O(bs_t, A) = \{w_{ij} \ \forall (i,j) \in RB_{t+1}(A)\}. \tag{27}$$

Due to the modification of the observation space, the transition probabilities apply only to the edges that become reachable as a result of the action, that is, Equation (4) is replaced by:

$$P(\overrightarrow{w_t}|bs_t,\, A) = \prod_{(i,j)\in RB_{t+1}(A)} \beta_{ij}^{W_{ij},t}. \tag{28}$$

In order to adjust the reward function to the continuous-time approximation, the benefit accrued must be multiplied by the amount of time that has passed since the beginning of the current period. Let set $CE_t(bs_t,\, A)$ include edge $A$ and the edges cleared until the $t^{th}$ clearance. Then, Equation (14) needs to be multiplied by:

$$\begin{cases} \frac{W_{ij}}{C}, & \text{if } \left\lfloor \frac{\sum_{(i,j)\in CE(bs_t,A)} W_{ij}}{C} \right\rfloor = \left\lfloor \frac{\sum_{(i,j)\in CE(bs_t,A)\setminus A} W_{ij}}{C} \right\rfloor \\ \frac{\sum_{(i,j)\in CE(bs_t,A)\setminus A} W_{ij}}{C} - \left\lfloor \frac{\sum_{(i,j)\in CE(bs_t,A)\setminus A} W_{ij}}{C} \right\rfloor, & \text{if } \left\lfloor \frac{\sum_{(i,j)\in CE(bs_t,A)} W_{ij}}{C} \right\rfloor \neq \left\lfloor \frac{\sum_{(i,j)\in CE(bs_t,A)\setminus A} W_{ij}}{C} \right\rfloor. \end{cases} \tag{29}$$

Furthermore, the term $U'_t(\overrightarrow{w_t})$ in Equations (7)-(9), (11), and (13) can now be replaced by $U_t \cup A$. Due to the change in the observation space, Equation (14) is simplified to:

$$R(bs_t,\, A) = \sum_{\overrightarrow{w_t}\in O(bs_t,\, A)} \left( \prod_{(i,j)\in RB'_t(A)} \beta_{ij}^{w_{ij},t}\, R(\overrightarrow{w_t}\,|\, bs_t, A) \right). \tag{30}$$

The set updates in the belief update equations are also simplified. Update steps in expressions (15) and (16) change to:

$$U_{t+1} \leftarrow U_t \cup A, \quad B_{t+1} \leftarrow B_t \setminus A \tag{31}$$

$$RB_{t+1} \leftarrow RB'_t(A), \quad UB_{t+1} \leftarrow B_{t+1} \setminus RB_{t+1}. \tag{32}$$

The remaining components of the belief state update are unchanged.

Based on the given simplifications, Bellman equations in Equation (21) become:

$$v^*(bs_t) = \max_{A\in RB_t} \left\{ \sum_{\overrightarrow{W}\in O(bs_t,\, A)} \prod_{(i,j)\in O} \beta_{ij}^{w_{ij},t} \left( R(\overrightarrow{w_t}\,|\, bs_t,\, A) + v^*(bs_{t+1}\,|\, bs_t,\, A,\, \overrightarrow{w_t}) \right) \right\}. \tag{33}$$

As stated in Observation 3, this approach provides an upper bound on the optimal benefit. The proofs for this observation and the next are provided in Appendix EC.1.

**Observation 3** *POMDP-C provides an upper bound on the optimal objective value of SDCP.*

To transform the optimal continuous policy into one that is feasible for the original problem, we use a sampling approach. We start from the root node of the search tree and follow the action

sequence resulting from the continuous approximation. To move through observation arcs, we select the observations randomly based on their occurrence probabilities. Once sequencing is complete, we cluster the edges for the first period based on the smallest debris observations until capacity is exhausted. After receiving the actual observations, we move along the search tree according to the these observations and repeat the procedure. The resulting heuristic, is called the *continuous-time heuristic*.

**Observation 4** *The expected reward obtained by POMDP-C yields a lower bound on the optimal objective value of SDCP.*

### 4.2. Heuristics for Approximate Tree Search

While the computational results in §6 show that the run-time of POMDP-C is significantly lower compared to the POMDP model discussed in §3, it still fails to solve large sized problems due to the depth of the tree. Hence, it is not practical to use the continuous-time heuristic on large instances. To overcome this issue and enhance the computational time performance, we propose two additional heuristics, namely *heuristic pruning* (HP) and *limited-lookahead* (LL), which are applicable on the search trees of both the original POMDP model and the continuous-time approximation.

HP is an "upper bounding" scheme on the search tree, which is commonly used for solving POMDP models in the literature (e.g., Paquet et al. 2006, Ross and Chaib-draa 2007). The general approach for a maximization problem, which is analogous to branch-and-bound, is to find an upper bound for the belief node, compare it to the best known lower bound so far, and prune the search node if its upper bound is smaller than the lower bound.

The heuristic pruning approach for SDCP is summarized in Algorithm 2 in Appendix EC.3. In each step, HP expands the search tree for POMDP-C in a depth-first manner. Assuming that all remaining unreachable edges have the lowest possible clearance resource requirements, it solves a deterministic version of SDCP. It then prunes the belief nodes whose upper bounds are lower than the expected benefit from the best solution so far. For larger problems, due to significant computational requirements of solving the deterministic debris clearance problem to optimality, the heuristic by Stilp et al. (2012) is used. While the objective value resulting from the heuristic does not always give a valid upper bound, it can be used to heuristically prune some of the belief nodes. Upon reaching the leaves of the search tree, the expected benefit for all preceding nodes are updated using backward induction. If necessary, the best lower bound is also updated. The heuristic terminates when the search tree is completely expanded.

The main objective in applying the limited-lookahead heuristic is to limit the search to a fixed tree depth while approximating the expected benefit of the leaves of the tree. Algorithm 3 in

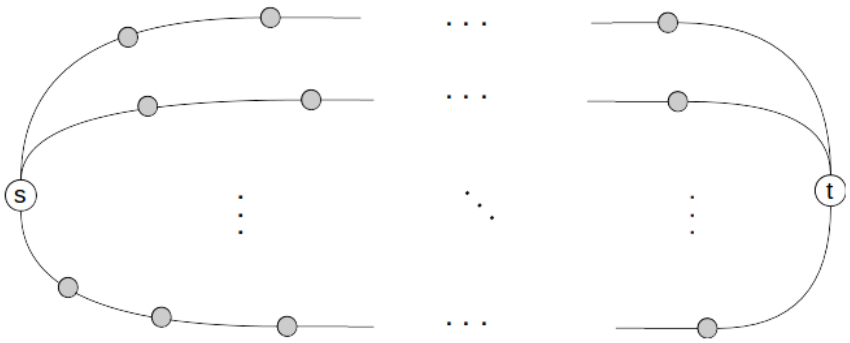


**Figure 4** A node-disjoint path network with $k$ node-disjoint paths between a single supply node $s$ and a single demand node $t$. All edges have either $l_1$ or $l_2$ units of resource requirements. On path $j$, the probability of observing $l_1$ units of debris is $p_j$.

Appendix EC.3 summarizes the limited lookahead procedure. The benefit-to-go values of the leaves are approximated by applying the greedy *demand-per-debris* heuristic described in §6, also summarized in Appendix EC.3. Once the tree is completely expanded, the heuristic finds the best actions using backward induction.

Both HP and LL fall into the category of online algorithms for POMDP (Ross et al. 2008), as they dynamically update the decisions based on new information. We also combine these heuristics and propose a hybrid approach (HP & LL), where heuristic pruning is applied up to a limited depth of the search tree, rather than the complete one.

## 5. Structural Results on Simple Networks

In this section, we consider SDCP on simple network structures and show that under certain settings, the continuous-time heuristic solves SDCP to optimality. Furthermore, under several more general network settings, solutions generated by the heuristic yields structural properties that are very similar to those of the optimal solution.

Consider a network with a single supply node $s$ and a single demand node $t$ connected with $k$ node-disjoint paths (NDP). An example of such a network is shown in Figure 4. Furthermore, assume that for each initially unreachable edge (ones that are not incident to node $s$), there are two possible resource requirement levels, $l_1$ and $l_2$, where $l_1$ and $l_2$ are identical for all edges and $l_1 < l_2$. On each path $j \in \{1, 2, \ldots, k\}$, the probability that an edge has $l_1$ units is Bernoulli distributed with probability $p_j$, denoted by $Bernoulli\big(l_1, l_2; \{p_j\}_{j=1}^{k}\big)$. We focus on two special cases: (i) $l_1 = 0$ and $l_2 > 0$ (without loss of generality, we assume $l_2 = a$), and (ii) $l_2 > l_1 > 0$ (without loss of generality, we assume $l_1 = 1$, $l_2 = K$).

THEOREM 1. *On an NDP network with a single supply and demand node and resource requirements distributed with* $Bernoulli\big(0, a; \{p_j\}_{j=1}^{k}\big)$*, the optimal (continuous and periodic) policy proceeds on the path that has the least expected debris on its blocked edges until the demand node.*

The proof for Theorem 1 and the remaining theorems in this section are provided in Appendix EC.1.

When clearance resource requirements can take two positive values, the optimal policies for the continuous and periodic models are more complicated and not identical, but exhibit similar properties, as stated in Theorem 2. Here, a 1-debris edge and a $K$-debris edge have 1 and $K$ units of resource requirements, respectively.

THEOREM 2. *Consider an NDP network with* $Bernoulli\big(1, K; \{p_j\}_{j=1}^{k}\big)$ *debris levels.*

- *If a* 1*-debris edge is cleared last on a path, the optimal (continuous and periodic) policy is to proceed on that path until a* $K$*-debris edge is reached.*
- *If a* $K$*-debris edge is cleared last on a path,*
  - *the optimal continuous policy is to proceed on that path until the demand node.*
  - *the optimal periodic policy is to proceed on that path until the demand node, unless a* $K$*-debris edge is completely cleared at the end of a period, and another* $K$*-debris edge becomes reachable.*

Theorem 2 partially characterizes the optimal periodic and continuous policies on NDPs with $Bernoulli\big(1, K; \{p_j\}_{j=1}^{k}\big)$ debris levels. Theorem 3 establishes the tractability of the optimal policy characterization for the continuous model.

THEOREM 3. *Consider an NDP network with* $Bernoulli\big(1, K; \{p_j\}_{j=1}^{k}\big)$ *debris levels. The optimal policy for the continuous model can be found in* $O(k|E|^4)$ *time.*

The results in Theorems 1 through 3 show that when potential resource requirements are limited to two fixed levels, the problem becomes substantially easier to solve, and the optimal periodic and continuous policies are either identical or show very similar properties. For comparison, the deterministic version of SDCP on a single path is NP-hard, as shown by Stilp et al. (2012).

Theorems 1, 2, and 3 extend the results by Karger and Nikolova (2008) on the Canadian traveller problem to the case where probabilities are path-dependent.

Next, we direct our attention to the case of a single supply node and multiple demand nodes, where $k_i$ NDPs connecting the supply node to the demand node $i$. We extend the notation as $Bernoulli\big(l_1, l_2; \{p_{ij}\}_{j=1}^{k_i}\big)$ for each demand node $i$. Let $I'$ denote the set of demand nodes not currently reachable, $J(i)$ represent the set of paths leading to node $i$, and $d_i$ denote the demand on node $i$. The results from Theorems 1, 2, and 3 can be extended to these cases as well for sufficiently large number of periods, denoted by $|T|$.

THEOREM 4. *Consider an NDP network with a single supply node, multiple demand nodes with varying per-unit benefits and demands, and* $Bernoulli\big(0, a; \{p_{ij}\}_{j=1}^{k_i}\big)$ *debris levels. If* $|T|$ *is large enough to allow all demand nodes to become reachable when clearance is complete, the optimal (continuous and periodic) policy is as follows:*

1. *Proceed on path* $j^* \in J(i)$ *that maximizes the ratio* $\dfrac{b_i d_i}{n_{J(i)} b_i a}$ *over all* $i \in I'$ *and* $j \in J(i)$*, where* $n_{J(i)}$ *is the number of unreachable edges in path* $J(i)$*. Remove* $i$ *from* $I'$*.*
2. *Repeat until* $I' = \emptyset$*.*

The partial characterization of the optimal policy in Theorem 2 applies to the case with $Bernoulli\big(1, K; \{p_{ij}\}_{j=1}^{k_i}\big)$ debris levels. Furthermore, the optimal policy for the continuous model can be found in polynomial time.

For comparison, the deterministic version of the problem with a single supply node and multiple demand nodes has been shown to be NP-hard, even on trees (Stilp et al. 2012).

The results we present in this section show that on simple network settings, the optimal policy under the continuous approximation and the periodic model are either identical, or show similar structural properties. Hence, the results motivate our development of the continuous-time heuristic and exhibit under which network structures we can expect it to perform well. In the next section, we use computational experiments to investigate whether these results extend to more general settings.

## 6. Computational Experiments

The goals of our computational experiments are three-fold: to (i) determine the largest problems for which the POMDP approach can find optimal solutions, and (ii) analyze the performance of the heuristics discussed in §4, and (iii) to derive policy-based insights. The quality of the heuristic solutions is evaluated by the gaps between the calculated upper bounds and the best objective values obtained during the computational experiments. We also report the run times of the heuristics, since timely decision making is crucial in this context.

Stilp et al. (2012) developed heuristics that run fast and find good quality solutions for the deterministic debris clearance problem, which is NP-hard. The stochastic nature of SDCP presents further computational challenges. The approaches we propose for solving SDCP take considerably more time and are useful only if they provide significant improvements for solution quality over solving the deterministic model with expected debris amounts. We compare the solutions generated by our proposed approaches for SDCP versus the deterministic approach, and determine when there are significant differences.

Current operations for debris clearance use predetermined road prioritization. To the best of our knowledge, a systematic procedure for determining road priorities does not exist in practice.

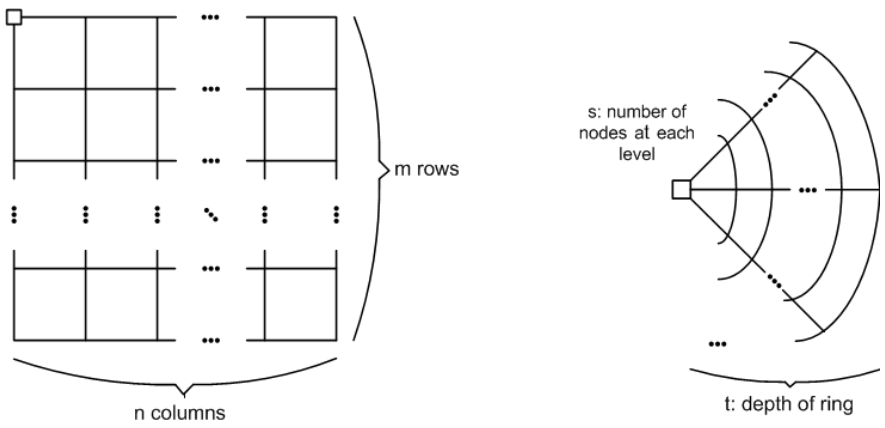


**Figure 5** (a) $G(m,n)$, a grid network with $m$ rows and $n$ columns, (b) $R(s,t)$, a ring network with $t$ levels and $s$ nodes at each level (excluding the center)

We consider a number of "benchmark" heuristics to mimic current practice. We also determine the settings under which performance differences between benchmark heuristics and the SDCP heuristics we proposed in §4 are more significant.

The remainder of this section is organized as follows. We first describe the set of instances used in the experiments and how they are generated. Then, we describe the benchmark heuristics that ignore the stochastic aspect of the problem and/or apply predetermined road prioritization. Finally, we present the results of the experiments and derive conclusions.

## 6.1. Instance Generation

The instance sets we generate can be classified under two main groups: (i) structured grid and ring networks, which resemble urban city road networks, and (ii) a realistic earthquake scenario.

We denote a grid network with $m$ rows and $n$ columns by $G(m \times n)$, whereas $R(s \times t)$ represents a ring network with $t$ levels and $s$ nodes at each level, as illustrated in Figure 5. The sizes of both grid and ring networks in the experiments are $(5 \times 5)$, $(10 \times 10)$, and $(15 \times 15)$. By varying the size of the network and the number of blocked roads, we test the performances of both optimal and heuristic approaches regarding instance size.

**6.1.1. Instances on Grid and Ring Networks.** Each grid/ring instance has a single supply node, from which the clearance process starts, and there is no capacity for supply. For grid networks, the supply node is on the upper left corner, whereas for ring graphs, it is in the center. In further experiments, we randomly vary the location of the supply node, and also conduct experiments with two supply nodes. The results for these cases are given in Appendix EC.5 and are in line with those presented in this section. Demand on each node is randomly generated based on a uniform distribution with parameters $U(0,5)$. Each unit of satisfied demand accrues one unit of benefit per

**Table 2** Number of blocked edges for each instance

| | | % roads blocked | | |
|---|---|---|---|---|
| **Problem set** | | **25** | **50** | **100** |
| Grid/ring instances | $G(5\times5)$ and $R(5\times5)$ | *** | *** | 40 |
| | $G(10\times10)$ and $R(10\times10)$ | 45 | 90 | 180 |
| | $G(15\times15)$ and $R(15\times15)$ | 105 | 210 | 420 |
| Earthquake scenario | | 151 | 302 | 604 |

period. The percentage of roads blocked is set to 25%, 50%, and 100%, resulting in instance sizes given in Table 2, where the blocked edges are chosen randomly and their number varies from 40 to 420.

The clearance resource requirements are generated based on a uniform distribution with parameters $U(0,9)$, $U(0,5)$, and $U(0,1.2)$ for the cases with network sizes of $(5\times5)$, $(10\times10)$, and $(15\times15)$, respectively. These settings yield approximately equal total expected debris amounts for each instance. The clearance capacity is 30 units per period and there are eight, six, and four periods for the cases with 100%, 50%, and 25% blocked roads, respectively.

The initial beliefs of the decision maker on the clearance resource requirements on unreachable roads are constructed as follows: There are two possible beliefs regarding the resource requirement for each road. The first is the actual one, whereas the second is determined by multiplying the actual requirement by a random variable for each road. For half of the instances, the random variable follows a uniform distribution with parameters $U(0.5,2)$, whereas for the remainder it follows a uniform distribution with parameters $U(0,1.5)$. The corresponding initial beliefs, i.e., probabilities corresponding to each possible requirement level, are $p_{ij}$ and $1-p_{ij}$, respectively. To analyze the effect of belief accuracy, we let $p_{ij}$ follow a uniform distribution with one of the following sets of parameters: $U(0,0.5)$, $U(0,1)$, $U(0.25,0.75)$, and $U(0.5,1)$. Note that a higher mean for the distribution implies higher *accuracy* of beliefs. Furthermore, when the mean is closer to zero or one, the *certainty* in beliefs is higher, regardless of the accuracy.

For each network structure and parameter settings, we generate 50 instances by varying the demand, blocked edges, debris amounts, and belief probabilities, leading to a total of 2,800 instances.

**6.1.2. An Earthquake Scenario.** We use a second set of problem instances based on an earthquake scenario with an epicenter around 20 miles off the east coast of Boston, Massachusetts. The scenario is generated using the disaster simulation tool Hazus by FEMA (2012). The earthquake (with epicenter shown in Figure 6), is of magnitude 6.5 on the Richter scale. The disaster area we consider consists of the six census tracts in central Cambridge, MA (see Figure 6). The main objective is to establish connectivity between the hospitals (supply nodes) in the region and population centers (demand nodes) that are in need of access to hospitals.

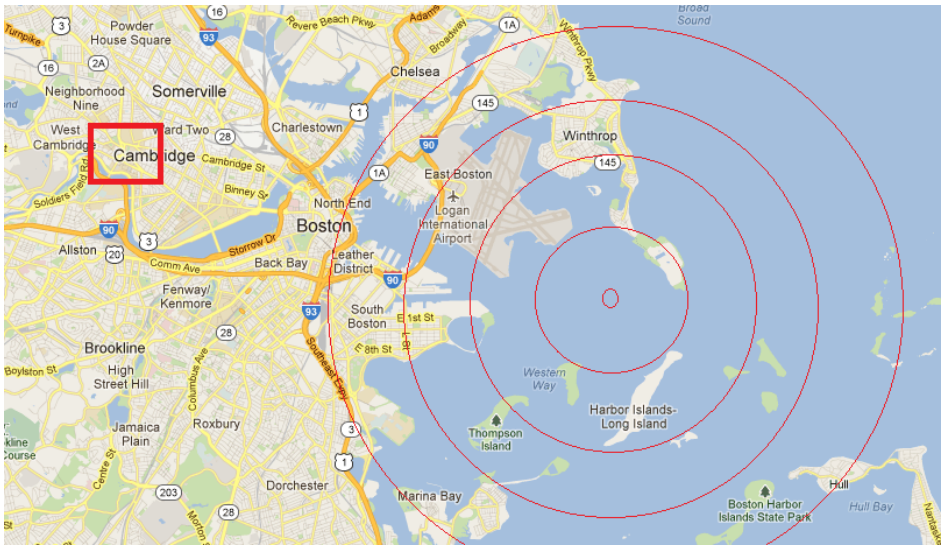


**Figure 6** The epicenter of the earthquake scenario and the focus area

For this set, we assume that the clearance resource requirements are directly proportional to the debris amounts. We use a two-phase approach to set the actual debris amounts on the 604 road segments. First, we use Hazus (FEMA 2012) to estimate the aggregate debris amount for each census tract. Then, we determine the road segments in each census tract and distribute the total debris amount to these segments proportional to road length.

There are two supply nodes (hospitals) in the disaster region: a regional hospital and the health center of a university, with capacities of 3,000 and 415 units, respectively, estimated proportional to their bed capacities. Each non-supply node on the network is considered as a demand node. There are 446 demand nodes, with a total demand normalized to 2,500 units, and the aggregate demand of each census tract is determined proportional to its population. The aggregate demand in each census tract is then distributed to the nodes in proportion to the total length of the roads emanating from the nodes. There is no additional supply/demand occurring throughout the planning horizon. Each unit of satisfied demand accrues one unit of benefit per period.

As in the case of grid/ring instances, the percentage of blocked roads is varied to be 25%, 50%, and 100%. The length of the planning horizon for these instances is three, four, and six periods, respectively. Per period, clearance capacity is 15,000 CY for the first two periods, 23,000 CY for the third and fourth periods, and 31,000 CY in the remainder of the horizon. With these capacities, more than 95% of all demand nodes can be reached in each instance.

The possible debris amounts and initial belief probabilities are determined in an identical way to those for grid/ring instances. When regional updates are considered, each census tract is taken as a separate region. For each case, we generate 50 instances by varying the choice of blocked edges,

**Table 3** A summary and comparison of proposed and benchmark heuristics, with ⋆ indicating problem sizes solvable up to 40 blocked roads; ⋆ ⋆ up to 100 blocked roads, ⋆ ⋆ ⋆ up to 200 blocked roads, and ⋆ ⋆ ⋆ ⋆ more than 200 blocked roads

| | | **Periodic nature** | **Stochasticity** | **Instance size** | **Network structure** | **Regional updates** |
|---|---|---|---|---|---|---|
| Perfect information | Optimal | ✓ | ✓ | ⋆ | ✓ | ✓ |
| | Heuristic | ✓ | ✓ | ⋆ ⋆ ⋆ ⋆ | ✓ | ✓ |
| Deterministic with expected amounts | Optimal | ✓ | | ⋆ | ✓ | |
| | Heuristic | ✓ | | ⋆ ⋆ ⋆ ⋆ | ✓ | |
| Predetermined road prioritization | DPD | | | ⋆ ⋆ ⋆ ⋆ | | |
| | DPDW | | | ⋆ ⋆ ⋆ ⋆ | | |
| | MFP | ✓ | | ⋆ ⋆ ⋆ ⋆ | | |
| Stochastic approach | POMDP-C | ✓ | ✓ | ⋆ | ✓ | ✓ |
| Continuous approximation | POMDP | | ✓ | ⋆ ⋆ | ✓ | ✓ |
| | HP | | ✓ | ⋆ ⋆ ⋆ | ✓ | ✓ |
| | LL | | ✓ | ⋆ ⋆ ⋆ ⋆ | ✓ | ✓ |
| | HP & LL | | ✓ | ⋆ ⋆ ⋆ ⋆ | ✓ | ✓ |

potential debris amounts, and initial belief probabilities, leading to a total of 600 instances in our second set of experiments.

## 6.2. Benchmark Approaches

To assess if and when the approaches we propose for SDCP are beneficial, we compare them to simpler approaches that ignore the uncertainty and/or apply predetermined road prioritization. The proposed and benchmark approaches are summarized in Table 3.

To assess the effect of uncertainty in our instances, we solve each instance under perfect information on the clearance resource requirements. One way to decrease the complexity due to the uncertainty is to solve a deterministic model using expected debris amounts on each road segment. Algorithm 4 in Appendix in Appendix EC.3 summarizes the procedure applied for solving the models under both cases. For small- to medium-sized instances, the deterministic problem can be solved to optimality, whereas for larger instances, we use the heuristics developed by Stilp et al. (2012). All of the benchmark heuristics assume a deterministic environment with expected values as debris amounts.

Currently, to the best of our knowledge, there is no systematic procedure in practice for how the clearance activities are sequenced. Guidelines such as FEMA (2007) suggest clearing roads of higher convenience with regard to disaster response operations (main arteries, evacuation routes, routes for care facilities, shelters, etc.) first, and clearing roads of lesser importance later, possibly during the recovery stage. In the Hazus database (FEMA 2012), there are mainly four types of roads, and each road is assigned a predetermined weight from 1 (most important) to 4 (least important). Figure 7 shows the road network and the pre-assigned road priorities for the earthquake scenario.

We develop and test three benchmark algorithms to calculate road priorities based on these road weights, mimicking the current practice. The first algorithm, *demand-per-debris* (*DPD*) works in

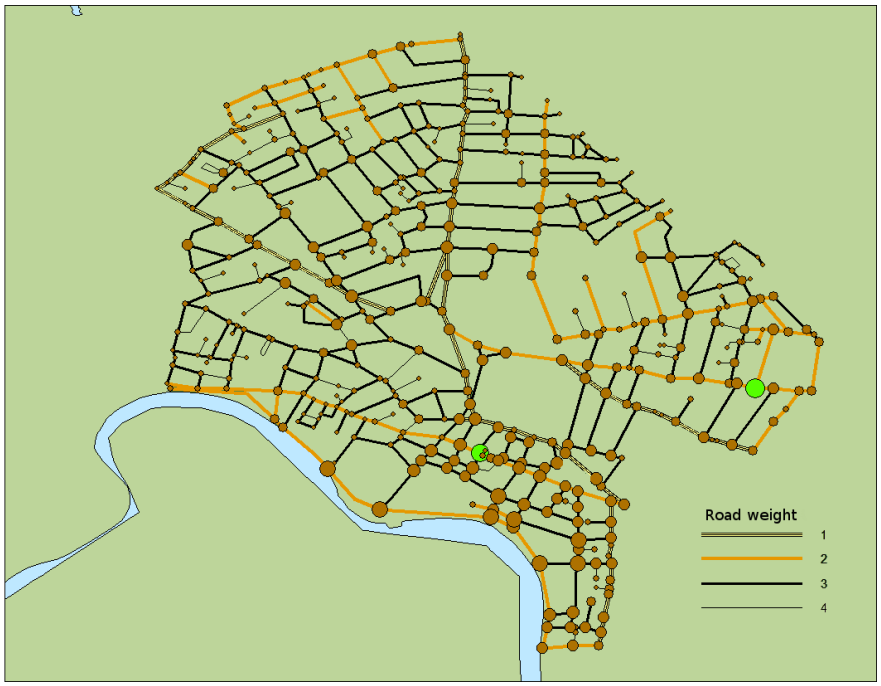


**Figure 7** Road types for the earthquake scenario

a similar way to the continuous approximation in §4, i.e., it greedily chooses the road segment to clear one at a time, based on the highest ratio of demand-reached to debris-amount among the immediately reachable roads with lowest weight (highest importance). Algorithm 5 in Appendix EC.3 summarizes the DPD heuristic more formally. The *demand-per-debris-per-weight* heuristic (*DPDW*) is similar to the DPD, except that it is not limited to roads with maximum priority. Upon each clearance, among immediately reachable roads, it clears the one with the highest ratio of demand-reached to debris-amount-per-road-weight. Hence, DPDW differs from DPD only by Step 3 of Algorithm 5, which now chooses $(i,j) \in RB_t$ with maximum $\frac{rd_{j1}}{w_{ij} * rw_{ij}}$. For both DPD and DPDW, the resulting solution is converted to a periodic solution using the transformation described in Algorithm 5. *Maximum-flow-per-period* (*MFP*) differs from the first two in the sense that it is applied on a periodic basis. In each period, among the roads with highest importance, it applies the heuristic by Stilp et al. (2012) to determine the clearance sequence. A summary of the MFP procedure is given in Algorithm 6 in Appendix EC.3.

### 6.3. Computational Results

In assessing the solution quality of the proposed and benchmark approaches, we consider a number of optimality gaps. Let $z^*$, $z^p$, $z^u$, and $z^h$ denote the objective function values for the optimal POMDP approach, optimal/heuristic solution with perfect information, upper bound obtained by POMDP-C, and any proposed or benchmark heuristic solution, respectively. Then, we define

**Table 4** Average % upper bound gaps and heuristic gaps for POMDP-C along with CPU times (in seconds) on small-sized grid/ring instances

| Distribution of $p_{ij}$ | Network | % roads blocked | Upper bound gap | Optimality gap | CPU time for optimal | CPU time for POMDP-C |
|---|---|---|---|---|---|---|
| $U(0, 0.5)$ | G(5×5) | 100 | 9.14 | 7.34 | 1,275 | 280 |
| | G(10×10) | 25 | 12.65 | 10.18 | 5,256 | 402 |
| | R(5×5) | 100 | 11.08 | 5.35 | 1,423 | 256 |
| | R(10×10) | 25 | 13.09 | 7.61 | 2,994 | 259 |
| $U(0, 1)$ | G(5×5) | 100 | 8.24 | 7.50 | 1,340 | 277 |
| | G(10×10) | 25 | 13.20 | 11.00 | 5,406 | 447 |
| | R(5×5) | 100 | 12.13 | 6.72 | 1,304 | 236 |
| | R(10×10) | 25 | 12.90 | 8.62 | 3,270 | 268 |
| $U(0.25, 0.75)$ | G(5×5) | 100 | 9.70 | 5.79 | 1,449 | 273 |
| | G(10×10) | 25 | 12.54 | 13.04 | 5,530 | 451 |
| | R(5×5) | 100 | 11.44 | 6.90 | 1,271 | 268 |
| | R(10×10) | 25 | 12.09 | 12.58 | 3,035 | 275 |
| $U(0.5, 1)$ | G(5×5) | 100 | 10.12 | 8.27 | 1,430 | 294 |
| | G(10×10) | 25 | 12.56 | 12.07 | 6,103 | 462 |
| | R(5×5) | 100 | 11.69 | 7.22 | 1,083 | 204 |
| | R(10×10) | 25 | 11.46 | 12.74 | 3,197 | 237 |

the *optimality gap*, *upper bound gap*, and *perfect information gap* as $\frac{z^*-z^h}{z^*}$, $\frac{z^u-z^*}{z^*}$, and $\frac{z^p-z^h}{z^p}$, respectively. The experiments throughout this section have been conducted on a computer with 2.10 GHz Intel Core 2 Duo processor and 4 GB RAM.

Our computational results consist two parts: Results on the assessment of heuristic performance (§6.3.1) and policy-related results (§6.3.2).

**6.3.1. Results on Heuristic Performance.** Since all of our heuristics use a continuous-time approximation for SDCP, we compare the performance of POMDP-C to the optimal POMDP approach on grid/ring instances of size up to 45 blocked roads. For larger instances, POMDP fails to find a feasible solution due to reaching the memory limit. The average upper bound gaps and optimality gaps (over 50 instances) are reported in Table 4, along with the average CPU times for each approach.

First, we observe that for grid/ring instances, which are of smaller size, **upper bound and optimality gaps of POMDP-C are at most around 13% on average.** Furthermore, the variation of these gaps with changing belief structures is not statistically significant. Although the number of blocked roads is almost identical in all instances, with a fixed network type (grid or ring) and belief structure, increasing the density of the network increases the gap in almost all instances. Intuitively, when the number of roads to be cleared per period increases, this results in a decrease in the accuracy of a continuous-time approximation (a continuous-time approximation is most accurate when there is a single road to be cleared in each period).

Based on the results on the grid/ring instances given in Table 4, **POMDP-C provides 86% CPU time savings on average, compared to POMDP**. The average time savings range

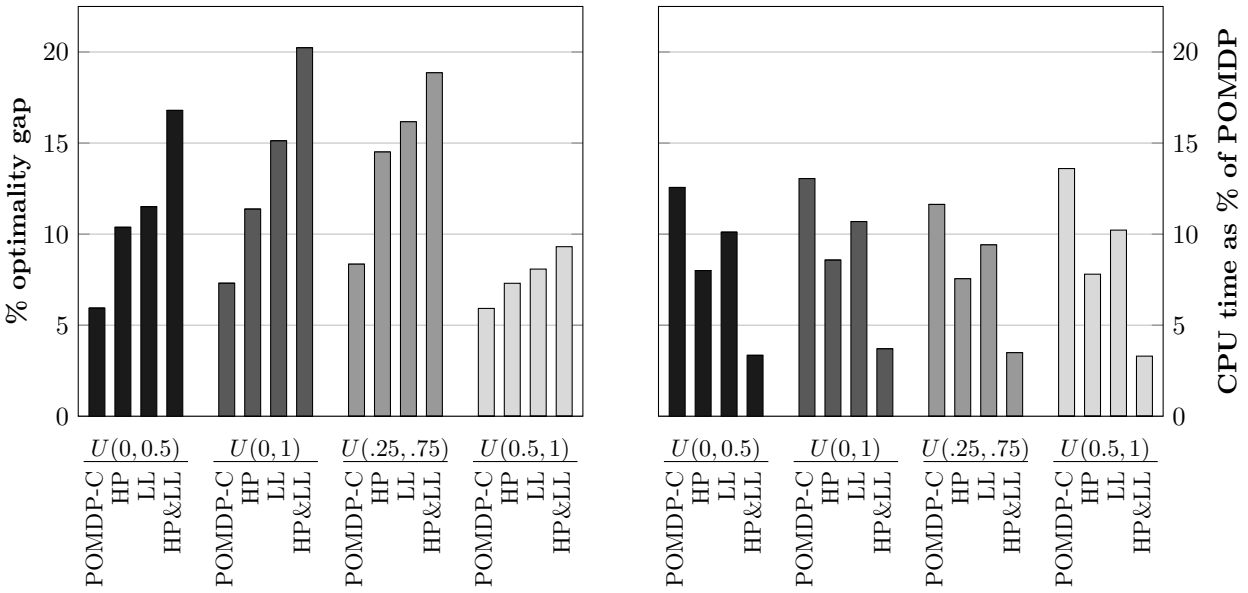


**Figure 8** Average % optimality gaps (left) and CPU times as % of POMDP (right) for the continuous approximation, heuristic pruning, limited lookahead, and hybrid heuristics under varying belief distributions without regional updates on grid/ring instances that can be optimally solved

from 78% to 93%. The level of CPU time savings compared to relatively small gaps from optimal solutions underlines the significance of applying POMDP-C during the disaster response phase, where timely decision making is crucial.

Next, we test the performance of POMDP-C and the related heuristics. Tables EC.1 and EC.2 in Appendix EC.4 present the perfect information gaps for the grid/ring and earthquake scenario instances, respectively. Tables EC.3 and EC.4 in Appendix EC.4 show the average CPU times of each approach under each setting. For brevity, Table EC.3 only gives the CPU times for the case where all $p_{ij}$ follow a uniform distribution with parameters $U(0.5, 1)$. For the remaining scenarios, the CPU times are provided in Table EC.7 in the Appendix.

The largest instances solvable by POMDP-C have 105 blocked roads. HP is able to solve instances with at most 180 blocked roads, while LL and the hybrid heuristic can solve all instances we generated. HP & LL also provides significant time savings, solving the largest instances within 2.5 hours on average, compared to the CPU times of HP or LL, which vary from two to eight hours.

Percent optimality gaps and CPU time savings of the heuristics over the POMDP model on grid/ring networks are shown in Figure 8. **Without regional updates, the proposed heuristics provide an average of at least 86% CPU time savings at the expense of at most 20% solution quality loss.** For grid/ring instances that can be solved to optimality, the average optimality gaps of POMDP-C, HP, LL, and hybrid approaches are 7.6%, 12.6%, 15.1%, and 19.4%, respectively, with average computational time savings of 86.5%, 92.2%, 89.8%, and 96.7%,

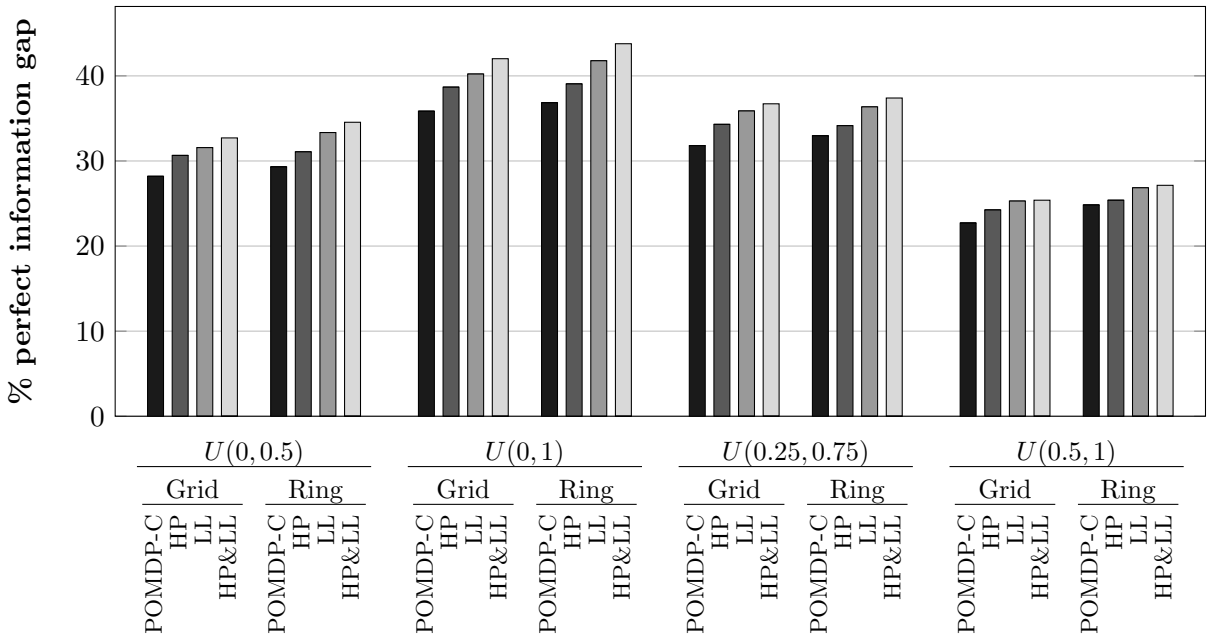


**Figure 9** Average % perfect information gaps for the continuous-time approximation, heuristic pruning, limited-lookahead and the hybrid heuristics on grid/ring networks under varying belief distributions without regional updates

respectively over the optimal POMDP model. In particular, whereas HP & LL takes less than 93 seconds on average, POMDP takes around 102 minutes. Furthermore, the performance of the proposed heuristics are robust with respect to the uncertainty in the environment, as the heuristic performance does not depend on the distribution of the belief parameters.

For grid/ring instances that cannot be solved to optimality, perfect information gaps of the continuous approximation and the approximate tree search heuristics are given in Figure 10. For any initial belief setting, the perfect information gaps for all tree search heuristics are within 19% of that for POMDP-C. Based on the results presented in Tables EC.3 and EC.7, applying the hybrid heuristic provides a minimum of 65% CPU time savings over the continuous approximation for these instances, which underlines the enhancement of the tree search resulting from the heuristics.

**Regional updates improve solution quality when beliefs are uncertain (belief distribution mean is around 0.5).** For the grid/ring and earthquake instances, Figures 10 and 11 compare the perfect information gaps under deterministic approximation and the hybrid heuristic with and without regional updates. When beliefs are certain ($p_{ij}$ follow a uniform distribution with mean closer to 0 or 1), the results with and without regional updates are comparable to each other. However, when beliefs are uncertain, applying regional updates improves the perfect information gaps of the hybrid heuristic solutions by an average of 7.5% and 16.0% for the grid/ring and earthquake instances, respectively.

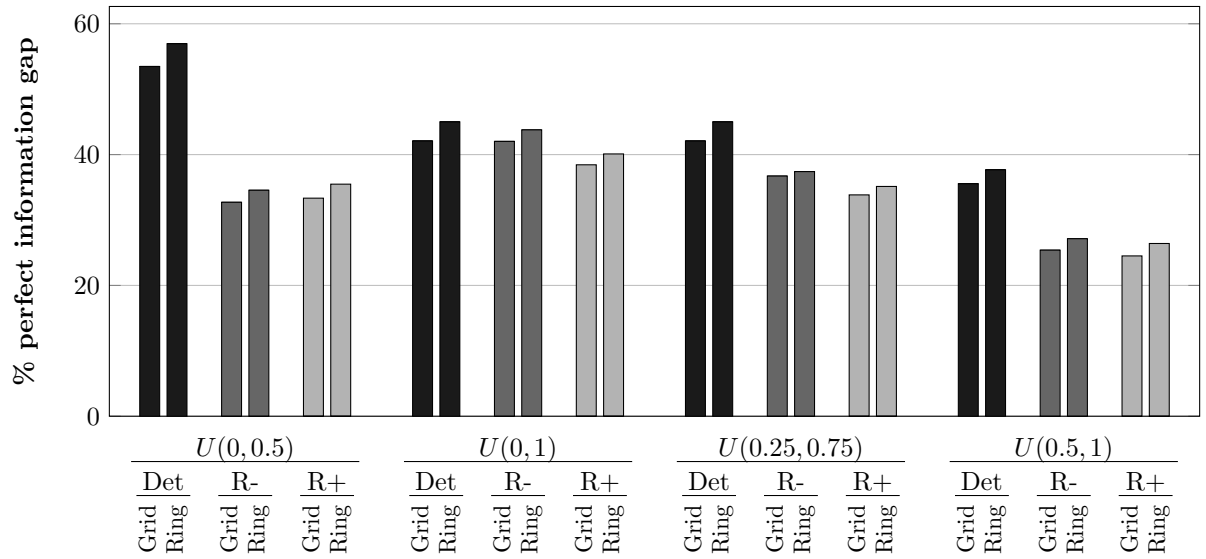


**Figure 10** Average % perfect information gaps on grid/ring networks under varying belief distributions, where *Det*, *R-*, and *R+* denote the deterministic approximation, and the hybrid heuristic with and without regional updates, respectively

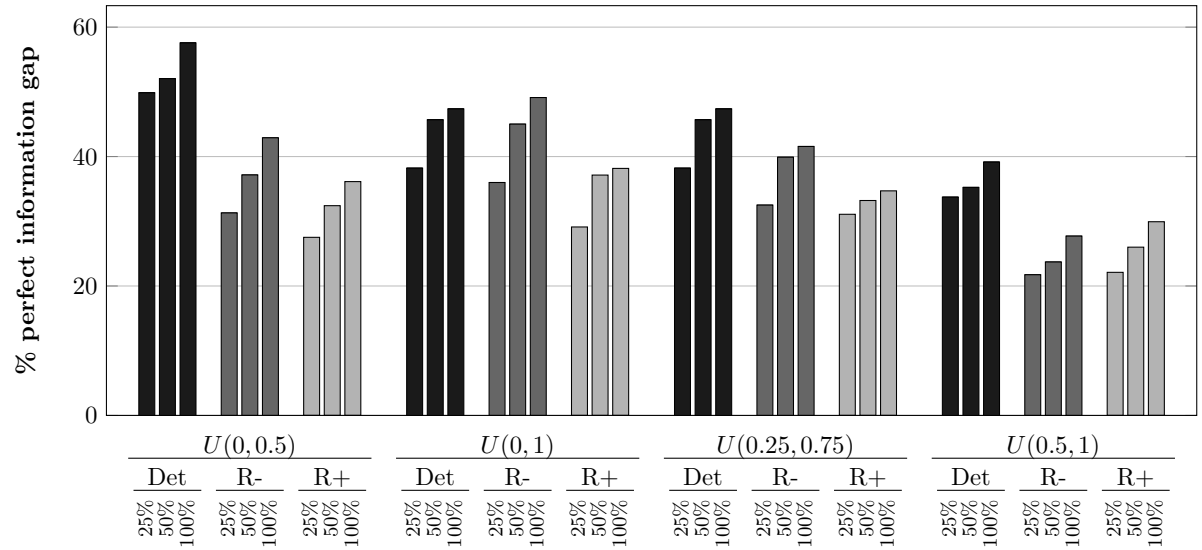


**Figure 11** Average % perfect information gaps for the earthquake scenario under varying belief distributions and % edges blocked, where *Det*, *R-*, and *R+* denote the deterministic approximation, and the hybrid heuristic with and without regional updates, respectively

**6.3.2. Policy-Related Results.** In addition to assessing heuristic performance, a second aim of our computational experiments is to derive policy-related implications. In particular, we are interested in when to apply a stochastic approach based on the structure of the initial beliefs and how the proposed approaches compare with respect to those that ignore uncertainty and those that mimic practice.

**Table 5** Summary of the effects of belief structure on solution quality

| | **Belief variance** | |
|---|---|---|
| **Beliefs** | **Low** | **High** |
| Accurate, certain | Accuracy and certainty of information yields good solution quality | High variance results in more need for exploration and therefore a loss of quality in solutions |
| Uncertain | Significant time spent for exploration, resulting in loss of benefits | |
| Inaccurate, certain | Inaccuracy corrected as new information is obtained | |

When regional updates are not available, from the comparison of the perfect information gaps under different belief distributions in Figures 10 and 11, we observe that **proposed heuristics perform better when beliefs are more certain (and not necessarily accurate).** For the grid/ring instances, the average perfect information gaps for $p_{ij}$ uniformly distributed with parameters $U(0,5)$, $U(0,1)$, $U(0.25,0.75)$, and $U(0.5,1)$ are 33.6%, 42.9%, 37.1%, and 26.3%, respectively. For the earthquake scenario instances, the gaps are 37.1%, 43.4%, 38.0%, and 24.4%, respectively. Hence, the best results are obtained when beliefs are accurate, i.e., distribution mean is higher. When beliefs are inaccurate but certain (distribution mean close to zero), the gaps are better compared to the cases where they are uncertain (distribution mean centered around 0.5). A summary of findings related to the effects of belief structure on solution quality is given in Table 5.

**Without regional updates, when belief certainty is higher, proposed heuristics outperform the deterministic approach.** The deterministic approximation using the heuristic approach by Stilp et al. (2012) solves all the instances in no more than 13 minutes on average. However, from Figures 10 and 11, the average perfect information gaps with $p_{ij}$ following a uniform distribution with parameters $U(0,5)$, $U(0,1)$, $U(0.25,0.75)$, and $U(0.5,1)$ for the grid/ring instances are 54.2%, 43.4%, 43.4%, and 36.0%, respectively, whereas the gaps are 53.2%, 43.7%, 43.7%, and 36.0%, respectively for the earthquake scenario instances. The hybrid heuristic outperforms the deterministic approximation for the first and last cases, that is, when beliefs are certain. For the first case, the deterministic approximation performs poorly due to inaccurate beliefs. For the last, while both approaches perform well due to accurate beliefs, the stochasticity in the problem environment (e.g., roads that actually have the less likely amount of debris) is captured better by the proposed heuristics. For the remaining belief settings, the average performances of the deterministic approach and the proposed heuristics are close, and the deterministic approach can find better solutions. However, **when regional updates are incorporated, proposed heuristics always outperform the deterministic approximation**, as can also be inferred from Figures 10 and 11. A practical implication of these results is the insight that it is worth using a stochastic approach when beliefs are certain. **If there is regional information, using a stochastic approach**

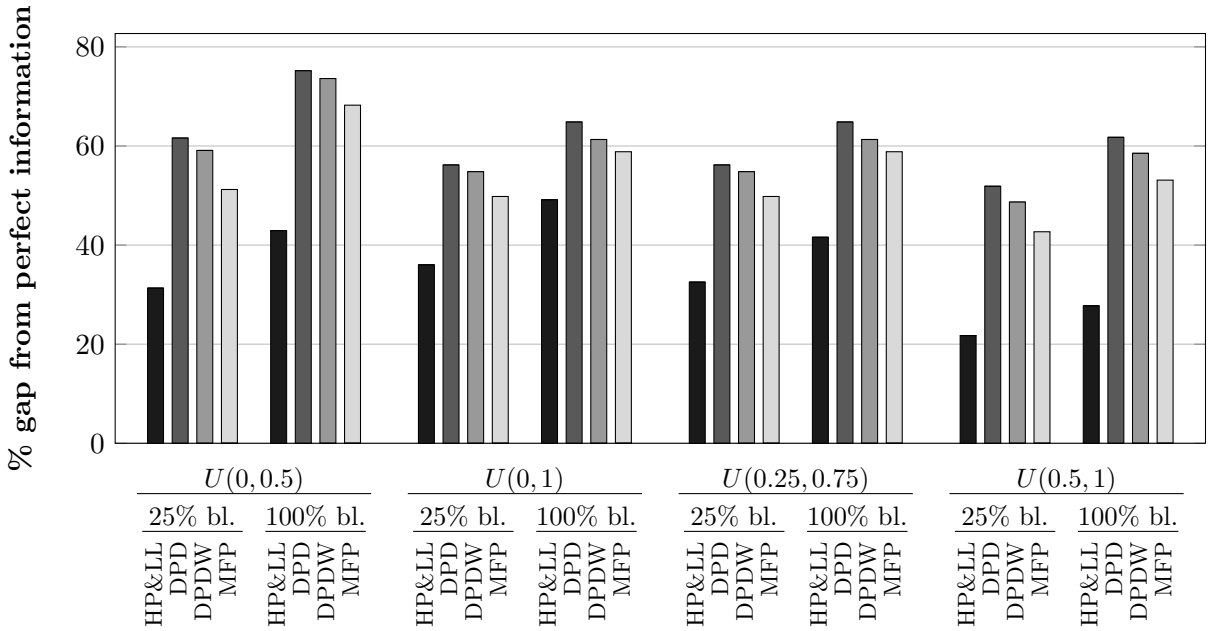


**Figure 12** Performance comparison of the proposed hybrid HP & LL heuristic to DPD, DPDW, and MFP, which mimic current practice, for varying initial belief distributions and % edges initially blocked

**results in significant improvement. When beliefs are uncertain and regional information is not available, the deterministic approach with expected resource requirements can be used.**

**Proposed heuristics outperform predetermined road prioritization.** The results in Table EC.4 in Appendix EC.4 indicate that for our instances, proposed approaches provide significant savings in solution quality over those in practice. Figure 12 summarizes and compares the average perfect information gaps of the proposed hybrid HP & LL heuristic with those under the DPD, DPDW, and MFP heuristics for the earthquake scenario instances. The largest improvement in the perfect information gap by taking road priorities as part of the decisions is observed when the initial beliefs are certain. For $p_{ij} \sim U(0, 0.5)$ and $p_{ij} \sim U(0.5, 1)$, using the hybrid heuristic instead of MFP decreases the perfect information gap from 68.24% to 42.90% and from 53.11% to 27.71%, respectively. The improvements are significant even under uncertain beliefs; the gap decreases from 58.84% to 49.12% and 41.56% under $p_{ij} \sim U(0, 1)$ and $p_{ij} \sim U(0.25, 0.75)$, respectively. **These results underline the significance of considering road priorities as dynamic decisions rather than taking them as predetermined inputs.**

Due to its myopic nature, DPD prioritizes the clearance of high-importance roads incident to low-demand nodes over low-importance roads incident to high-demand nodes. This may lead to significant loss of benefits. An example for this is given in Figure 13, where the high demand nodes in the south of the region are reached later than the low demand ones located north and




**Figure 13** Clearance sequence for DPD (left) and DPDW (right) on an earthquake scenario instance with 604 blocked roads. For DPD, high demand nodes on low priority roads and low demand nodes on main roads decrease the performance of the approach. For DPDW, clusters of high and low demand nodes and secluded high demand nodes decrease the performance of the approach

west of the region. Similarly, the myopic nature of DPDW may require significant time to reach high-demand nodes that are not easily reachable from roads of high importance. Such a case is shown in Figure 13, where the low demand nodes located close to high demand nodes in the west are favored over high demand nodes in the south, which are secluded as they can only be reached using low importance roads.

## 7. Conclusions

In this paper, we studied the debris clearance problem under incomplete information on clearance resource requirements, a practical problem faced especially in post-disaster situations where information on the disaster region is scarce. While a POMDP approach can solve the problem optimally, it does so with significant computational burden. Our proposed heuristics reduce the sizes of the action and observation spaces by employing a continuous-time approximation for the POMDP and applying heuristic pruning and/or limited look-ahead on the search tree. All of our approaches allow for updating information on a regional basis.

Our computational experiments on grid/ring instances as well as an earthquake scenario reveal that considering the stochasticity of the problem is useful especially when beliefs on the possible debris amounts are certain, regardless of accuracy. For the remaining cases, application of regional updates always improves the quality of the solutions. Furthermore, the current practice of predetermined road prioritization or ignoring stochasticity of the disaster environment may significantly reduce solution quality.

While deciding on road clearance prioritization, one can also consider the travel times of clearance units. This requires incorporating routing decisions and hence significantly increases the complexity of the problem. When travel times are considered, road clearance sequences for individual clearance teams might also need to determined.

SDCP addresses decisions in an environment where computational resources may be limited and decisions need to be made quickly. An important research direction is to further improve the computational time required for the solution of the problem. Another practical concern is the usability of the models by debris management authorities by means of user-friendly decision tools, which is addressed in a parallel work (Lorca et al. 2013).

Finally, future work could focus on the problems of clearance, collection, disposal, and recycling in a more holistic way, although this could lead to substantial complexity.

## Acknowledgments

The authors gratefully acknowledge the associate editor and two anonymous referees for their valuable comments that improved the quality of the work in this paper. Support for this work was provided by the National Science Foundation Grant CMMI-1000085, Andrea Laliberte, and Richard E. "Rick" and Charlene Zalesky.

## References

Aksu, D. T., L. Özdamar. 2014. A mathematical model for post-disaster road restoration: Enabling accessibility and evacuation. *Transportation Research Part E: Logistics and Transportation Review* **61**(1) 56–67.

Altay, N., W. G. Green. 2006. OR/MS research in disaster operations management. *European Journal of Operational Research* **175**(1) 475–493.

Anagnostopoulos, A., R. Bent, E. Upfal, P. Van Hentenryck. 2004. A simple and deterministic competitive algorithm for online facility location. *Information and Computation* **194**(2) 175 – 202.

Angelelli, E., M. G. Speranza, M. W. P. Savelsbergh. 2007. Competitive analysis for dynamic multiperiod uncapacitated routing problems. *Networks* **49**(4) 308–317.

Apte, A. 2009. Humanitarian logistics: A new field of research and action. *Foundations and Trends in Technology, Information and Operations Management* **3**(1) 1–100.

Ascheuer, N., S. Krumke, J. Rambau. 2000. Online dial-a-ride problems: Minimizing the completion time. H. Reichel, S. Tison, eds., *STACS 2000*, *Lecture Notes in Computer Science*, vol. 1770. Springer Berlin Heidelberg, 639–650.

Ausiello, G., V. Bonifaci, L. Laura. 2008. The online prize-collecting traveling salesman problem. *Information Processing Letters* **107**(6) 199 – 204.

Averbakh, I. 2012. Emergency path restoration problems. *Discrete Optimization* **9**(1) 58–64.

Averbakh, I., J. Pereira. 2012. The flowtime network construction problem. *IIE Transactions* **44**(8) 681–694.

Baxter, M., T. Elgindy, A. T. Ernst, T. Kalinowski, M. W. P. Savelsbergh. 2014. Incremental network design with shortest paths. *European Journal of Operational Research* **238**(3) 675–684.

Bent, R., P. Van Hentenryck. 2007. Waiting and relocation strategies in online stochastic vehicle routing. *Proceedings of the 20th international joint conference on Artifical intelligence*. IJCAI'07, 1816–1821.

Bent, R. W., P. Van Hentenryck. 2004a. Online stochastic and robust optimization. M. J. Maher, ed., *Advances in Computer Science - ASIAN 2004. Higher-Level Decision Making*, *Lecture Notes in Computer Science*, vol. 3321. Springer Berlin Heidelberg, 286–300.

Bent, R. W., P. Van Hentenryck. 2004b. Scenario-based planning for partially dynamic vehicle routing with stochastic customers. *Operations Research* **52**(6) 977–987.

Brown, C., M. Milke, E. Seville. 2011. Disaster waste management: A review article. *International Journal of Logistics: Research and Applications* **31**(6) 1085–1098.

Caunhye, A. M., X. Nie, S. Pokharel. 2012. Optimization models in emergency logistics: A literature review. *Socio-Economic Planning Sciences* **46**(1) 4–13.

Çavdaroğlu, B., E. Hammel, J. E. Mitchell, T. C. Sharkey, W. A. Wallace. 2011. Integrating restoration and scheduling decisions for disrupted interdependent infrastructure systems. *Annals of Operations Research* **203**(1) 279–294.

Çelik, M., Ö. Ergun, B. Johnson, P. Keskinocak, Á. Lorca, P. Pekgün, J. Swann. 2012. Humanitarian logistics. P. Mirchandani, ed., *Tutorials in Operations Research*, vol. 9. INFORMS, Hanover, MD, 18–49.

Center for Research on the Epidemiology of Disasters. 2013. EM-DAT: The international disaster database. `http://www.emdat.be`. Accessed: February 2013.

Chen, S., Y. Chen, Y. Long, Y. Shen. 2011. Optimized road network emergency restoration scheduling for relief distribution. *International Conference of E-Business and E-Government (ICEE)*. Shanghai, China, 1–4.

Congressional Research Service. 2011. Managing disaster debris: Overview of regulatory requirements, agency roles, and selected challenges. `http://www.fas.org/sgp/crs/misc/RL34576.pdf`. Accessed: January 2013.

Daganzo, C. F. 1984. The distance traveled to visit n points with a maximum of c stops per vehicle: An analytic model and an application. *Transportation Science* **18**(4) 331–350.

Daganzo, C. F. 1987a. Modeling distribution problems with time windows: Part I. *Transportation Science* **21**(3) 171–179.

Daganzo, C. F. 1987b. Modeling distribution problems with time windows. Part II: Two customer types. *Transportation Science* **21**(3) 180–187.

Daganzo, C. F., A. L. Erera. 1999. On planning and design of logistics systems for uncertain environments. M. G. Speranza, P. Stahly, eds., *Lecture Notes in Economics and Mathematical Systems: New Trends in Distribution Logistics*, vol. 480. Springer-Verlag, Berlin, 3–21.

Del Castillo, J. M. 1998. A heuristic for the traveling salesman problem based on a continuous approximation. *Transportation Research Part B: Methodological* **33**(2) 123 – 152.

Ekici, S., D. A. McEntire, R. Afedzie. 2009. Transforming debris management: Considering new essentials. *Disaster Prevention and Management* **18**(5) 511–522.

FEMA, Federal Emergency Management Agency. 2007. Public assistance debris management guide. `http://www.fema.gov/pdf/government/grant/pa/demagde.pdf`. Accessed: January 2013.

FEMA, Federal Emergency Management Agency. 2012. Hazus: Fema's methodology for estimating potential losses from disasters. `http://www.fema.gov/hazus`. Accessed: January 2013.

Fetter, G., M. Falasca. 2011. Establishing the need for decision support in disaster debris disposal. *Proceedings of the 8th International ISCRAM Conference*. ISCRAM, Lisbon, Portugal, 1–4.

Feuerstein, E., L. Stougie. 2001. On-line single-server dial-a-ride problems. *Theoretical Computer Science* **268**(1) 91 – 105.

Francis, P., K. Smilowitz. 2006. Modeling techniques for periodic vehicle routing problems. *Transportation Research Part B: Methodological* **40**(10) 872–884.

Gelman, A., J. B. Carlin, H. S. Stern, D. B. Rubin. 2004. *Bayesian Data Analysis*. Chapman & Hall, New York, NY.

Hentenryck, P. Van, R. Bent, E. Upfal. 2010. Online stochastic optimization under time constraints. *Annals of Operations Research* **177** 151–183.

Jaillet, P., X. Lu. 2011. Online traveling salesman problems with service flexibility. *Networks* **58**(2) 137–146.

Jaillet, P., M. R. Wagner. 2006. Online routing problems: Value of advanced information as improved competitive ratios. *Transportation Science* **40**(2) 200–210.

Jaillet, P., M. R. Wagner. 2008. Generalized online routing: New competitive ratios, resource augmentation, and asymptotic analyses. *Operations Research* **56**(3) 745–757.

Japan Times. 2011. Radiation, legalities complicate cleanup efforts. `http://english.kyodonews.jp/news/2011/04/85506.html`. Accessed: January 2013.

Jean-Louis, M., E. Hilaire. 2010. Debris removal situation, context and recommendations. `http://haiti-ecap.com/Websites/dig/images/H-Debris-SF3.pdf`. Accessed: January 2013.

Kalai, A., S. Vempala. 2005. Efficient algorithms for online decision problems. *Journal of Computer and System Sciences* **71**(3) 291–307.

Kalinowski, T., D. Matsypura, M. W. P. Savelsbergh. 2014. Icremental network design with maximum flows. `http://arxiv.org/abs/1312.6447`. Accessed: July 2014.

Karger, D., E. Nikolova. 2008. Exact algorithms for the Canadian traveller problem on paths and trees. Technical report, Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA.

Kibar, A. N., S. Salman. 2013. Logistics planning for restoration of network connectivity after a disaster. Working paper, Department of Industrial Engineering, Koç University, Ankara, Turkey.

Littlestone, N., M. K. Warmuth. 1989. The weighted majority algorithm. *30th Annual Symposium on Foundations of Computer Science*. Research Triangle Park, NC, 256 –261.

Lorca, Á., M. Çelik, Ö. Ergun, P. Keskinocak. 2013. A decision support tool for post-disaster debris operations. Working paper, H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA.

Luther, L. 2008. Disaster debris removal after Hurricane Katrina: Status and associated issues. `http://gactaern.org/Unit%20Plan/Healthcare%20Science_JM/Emergency_and_Disaster_Preparedness_25.564/EDP-17_Katrina_Debris_Report_JM.pdf`. Accessed: April 2013.

Markin, C. 2014. US Army Corps of Engineers, personal communication.

Matisziw, T. C., A. T. Murray, T. H. Grubesic. 2010. Strategic network restoration. *Networks and Spatial Economics* **10**(3) 345–361.

Mendonça, D., Y. Hu. 2006. Hurricane Katrina debris removal operations: The role of communication and computing technologies. *Learning from Catastrophe: Quick Response Research in the Wake of Hurricane Katrina* **40** 283–306.

Nurre, S. G., B. Çavdaroglu, J. E. Mitchell, T. C. Sharkey, W. A. Wallace. 2012. Restoring infrastructure systems: An integrated network design and scheduling problem. *European Journal of Operational Research* **223**(3) 794–806.

Nurre, S. G., T. C. Sharkey. 2014. Integrated network design and scheduling problems with parallel identical machines: Complexity results and dispatching rules. *Networks* **63**(4) 306–326.

Paquet, S., B. Chaib-draa, S. Ross. 2006. Hybrid POMDP algorithms. *Proceedings of the Workshop on Multi-Agent Sequential Decision Making in Uncertain Domains (MSDM-06)*. MSDM, Hakodate, Japan, 133–147.

Perrier, N., A. Langevin, C. A. Amaya. 2008. Vehicle routing for urban snow plowing operations. *Transportation Science* **42**(1) 44–56.

Powell, W. B., P. Frazier. 2012. Optimal learning. P. Mirchandani, ed., *Tutorials in Operations Research*, vol. 5. INFORMS, Hanover, MD, 213–246.

Powell, W. B., I. O. Ryzhov. 2012. *Optimal Learning*. John Wiley & Sons, Hoboken, NJ.

Ross, S., B. Chaib-draa. 2007. AEMS: An anytime search algorithm for approximate policy refinement in large POMDPs. *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07)*. Hyderabad, India, 2592–2598.

Ross, S., J. Pineau, S. Paquet, B. Chaib-draa. 2008. Online planning algorithms for POMDPs. *Journal of Artificial Intelligence Research* **32**(1) 663–704.

Rothkopf, M. S. 1966. Scheduling with random service times. *Management Science* **12**(1) 707–713.

Ryzhov, I. O., W. B. Powell. 2011. Information collection on a graph. *Operations Research* **59**(1) 188–201.

Şahin, H., B. Y. Kara, O. Karaşan. 2013. On debris removal during the response phase. Working paper, Department of Industrial Engineering, Bilkent University, Ankara, Turkey.

Solis, G. Y., H. C. Hightower, J. Sussex, J. Kawaguchi. 1996. *Disaster Debris Management*. Disaster Preparedness Resources Centre, University of British Columbia, Vancouver, BC.

Srinivas, H., Y. Nakegawa. 2009. Environmental aspects of the Indian Ocean tsunami recovery. *Journal of Environmental Management* **89**(1) 4–13.

Stilp, M., Ö. Ergun, P. Keskinocak. 2012. Management of debris operations. Working paper, H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA.

Swan, R. C. 2000. Debris management planning for the 21st century. *Natural Hazards Review* **1**(4) 222–225.

Thrun, S. 2002. Robotic mapping: A survey. G. Lakemeyer, B. Nebel, eds., *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, Burlington, MA, 1–36.

United Nations Office for the Coordination of Humanitarian Affairs. 2011. Disaster waste management guidelines. `http://ochanet.unocha.org/p/Documents/DWG%20Annex%20XII.Disaster%20waste%20management%20contingency%20planning.pdf`. Accessed: January 2013.

Washington Post. 2010. Haiti faces colossal and costly cleanup before it can rebuild. `http://www.washingtonpost.com/wp-dyn/content/article/2010/03/06/AR2010030602544.html`. Accessed: January 2013.

Yan, S., Y. Shih. 2009. Optimal scheduling of emergency roadway repair and subsequent relief distribution. *Computers & Operations Research* **36**(6) 2049–2065.

Zhong, H., R. W. Hall, M. Dessouky. 2007. Territory planning and vehicle dispatching with driver learning. *Transportation Science* **41**(1) 74–89.

# Appendices

## EC.1. Proofs of Statements

*Proof of Observation 3.* Consider the optimal policy of the original problem. Suppose first that it does not involve idling at the end of any period. This optimal policy gives a sequencing of edges under each possible debris amount observation. Each sequence under each possible observation can be transformed into a continuous-time policy by considering the edges in the sequence individually. By construction, the rewards obtained in both cases are identical. Suppose now that the optimal policy in the original problem involves idling. Then the transformed policy in the continuous model gives an expected reward that is strictly higher. Hence in both cases, the optimal policy of the original problem can be transformed to a feasible policy in the continuous approximation with an objective value that is no smaller than that in the original problem. □

*Proof of Observation 4.* By construction, the transformation results in a policy whose actions are in the action space defined by Equation (2). Furthermore, the observations are selected from the search tree for the original problem. Thus, the resulting policy is feasible for SDCP, and the expected reward of the policy is therefore a lower bound on the optimal objective value. □

*Proof of Theorem 1.* For the sake of brevity, we only provide the proof for the continuous case. The proof is very similar for the periodic case.

At any step of the optimal algorithm, let $v^*\left(\{w_j, n_j, p_j\}_{j=1}^k\right)$ denote the expected clearance resource requirement until reaching the demand node, where $w_j$, $n_j$, and $p_j$ denote the resource requirement in the first reachable edge, the number of unreachable edges until the demand node, and the probability of observing no debris on an unreachable edge on path $j$. Following Karger and Nikolova (2008), we first note that the optimal expected resource requirement is symmetric with regard to the paths, that is, $v^*\left(w_j, n_j, p_j, a_l, n_l, p_l, \{w_m, n_m, p_m\}_{m\neq j,l}\right)$ $= v^*\left(w_l, n_l, p_l, w_j, n_j, p_j, \{w_m, n_m, p_m\}_{m\neq j,l}\right)$. Next, we show that the expected clearance resource requirement is monotone nondecreasing in $n_j$ and monotone nonincreasing in $p_j$, $j = 1, \ldots, k$.

In order to show the monotone nondecreasing property for $n_j$, we make use of the proof by Karger and Nikolova (2008). By symmetry, it suffices to show the property for $n_1$. Let $\mathcal{C}_{n_1} = (w_1, n_1, p_1, \{w_j, n_j, p_j\}_{j>1})$, $\mathcal{C}_{n_1+n} = (w_1, n_1 + n, p_1, \{w_j, n_j, p_j\}_{j>1})$ for some $n > 0$, and $\tilde{\mathcal{C}}_{n_1+n} = \mathcal{C}_{n_1+n}$ with the last $n$ edges on the first path having zero debris. Let $\mathcal{A}$ be the optimal algorithm on $\mathcal{C}_{n_1+n}$. Define the algorithm $\mathcal{A}^*$ as follows: Run $\mathcal{A}$ until either the demand node or the last node before the zero-debris edges on the first path. If it reaches the former first, stop. Else, proceed to the demand node from the latter node. If $v_A^*(\mathcal{C})$ denotes the expected debris amount to be cleared on network configuration $\mathcal{C}$, we have:

$$v^*\left(w_1, n_1, p_1, \{w_j, n_j, p_j\}_{j>1}\right) \leq v_{\mathcal{A}}^*\left(w_1, n_1, p_1, \{w_j, n_j, p_j\}_{j>1}\right)$$

$$
\begin{aligned}
&= v^*_{\mathcal{A}^*}(\tilde{\mathcal{C}}_{n_1+n}) \\
&\leq v^*\left(w_1, n_1+n, p_1, \{w_j, n_j, p_j\}_{j>1}\right).
\end{aligned}
$$

For $p_1$, let $\mathcal{C}_{p_1} = (w_1, n_1, p_1, \{w_j, n_j, p_j\}_{j>1})$ and $\mathcal{C}_{p_1-\epsilon} = (w_1, n_1, p_1-\epsilon, \{w_j, n_j, p_j\}_{j>1})$. Let $\mathcal{A}$ be the optimal algorithm on $\mathcal{C}_{p_1-\epsilon}$. Then:

$$
\begin{aligned}
v^*\left(w_1, n_1, p_1, \{w_j, n_j, p_j\}_{j>1}\right) &\leq v^*_{\mathcal{A}}(\mathcal{C}_{p_1}) \\
&\leq v^*_{\mathcal{A}}(\mathcal{C}_{p_1-\epsilon}) \\
&= v^*\left(a+a_1, n_1, p_1-\epsilon, \{w_j, n_j, p_j\}_{j>1}\right).
\end{aligned}
$$

At any step of the optimal algorithm, let $ED_j$ be the total expected requirement on the unreachable edges on path $j$. Since the optimal amount of resources used until the demand node is nondecreasing in $n_j$ and nonincreasing in $p_j$, it is also nondecreasing in $ED_j$. As the next step, we show that if $w_1 = \min_j w_j$ and $ED_1 = \min_j ED_j$, it is optimal to proceed along the first path until the first unreachable edge. If $w_1 = 0$, the claim is obviously true. Therefore assume that $w_1 = 1$, which requires that $w_j = 1$ for all paths $j$. Assume the optimal algorithm chooses to proceed along a path $l \neq 1$. If a zero-debris edge is encountered next, the optimal policy is to keep on this path. Otherwise, by the monotonicity of the expected debris volume function, it is still optimal to proceed on this path, since $v^*\left(\{1, ED_j\}_{j=1}^k\right)$ being minimized by choosing path $j$ implies that $v^*\left(1, ED_l\frac{n_l-2}{n_l-1}, \{1, ED_j\}_{j\neq l}\right)$ is still minimized by choosing path $j$. The expected cost of this policy is $1 + ED_j$. However, the expected cost of proceeding on path 1 from the beginning is $1 + ED_1$, which is smaller.

This result implies that when all paths initially have 1 unit of requirement, it is optimal to proceed with the one that has the least expected requirement remaining. Hence, the optimal policy starts by traversing through all the zero-debris edges that are reachable and when all such edges are explored, proceeding with the path with the lowest remaining expected requirement. □

*Proof of Theorem 2.* (Outline) Suppose the optimal policy traverses an edge with 1 unit of requirement. This indicates that the sum of the cost of traversing this edge and the expected cost of optimal policy at the next node on this path is no more than that of any other action. Since 1 unit of requirement at the next edge is the lowest possible amount, the expected requirement on the remainder of the path decreases after encountering a 1-debris edge, and hence proceeding on this path still dominates switching to another path. Thus, if a 1-debris edge is traversed, it is optimal to keep on the same path until a $K$-debris edge is encountered.

Suppose now that the optimal continuous policy traverses a $K$-edge on a path. By the above argument, if the next edge on this path has 1 unit of requirement, it is optimal to remain on the

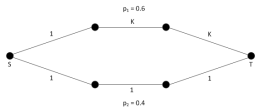


**Figure EC.1** A counterexample with $C = K + 1$ on a network with two node-disjoint parallel paths. The periodic model proceeds with the three edges on the second path after clearing the first K-debris edge on the first path.

same path. Suppose the next encountered edge is a $K$-debris edge. Since the action to choose this path with one more unreachable edge and a $K$-debris edge as its reachable edge dominated all remaining actions, with one fewer unreachable edge, staying on this path still dominates switching to any of the remaining paths. Hence if a $K$-debris edge is traversed, it is optimal to keep on the same path until the demand node.

Suppose that under the optimal periodic policy, a $K$-debris edge is cleared in period $t$ and the period is over at the same time the clearance is complete. If a 1-debris edge becomes reachable, by the above argument, it is optimal to proceed with it. Assume that a $K$-debris edge becomes reachable. An example showing that it may be optimal to switch to a different path is given in Figure EC.1. The network consists of two node-disjoint paths with three edges each, $p_1 = 0.6$, $p_2 = 0.4$, and $C = K + 1$. In the first period, the optimal periodic policy clears the first two edges on the first path and encounters another $K$-debris edge on the first path at the end of the period. In this situation, it is optimal to switch to the second path and traverse it until the demand node. □

*Proof of Theorem 3.* As in Karger and Nikolova (2008), we start by showing that the continuous approximation with two paths is polynomially solvable. At any instant before committing to a path, let the graph knowledge (state) be represented by $(x_1, n_1; x_2, n_2)$, where $x_j$ is the clearance resource requirement on the first reachable edge on path $j$ and $n_j$ is the number of unreachable edges in that instant. Unlike Karger and Nikolova (2008), our state definition does not need to consider the number of 1-debris edges encountered so far on a path and the current location, as there is no cost of traversing an edge on the way back. Under this state definition, there are $O(|E|^2)$ states, as opposed to $O(|E|^4)$ states for CTP. Since the only actions are to proceed with the current path or to switch to the other, the optimal policy can be found in polynomial time.

Next, we show that the general problem is polynomially solvable by using the 2-path algorithm as a subroutine. First, order the paths starting with a 1-debris edge in increasing expected debris. Out of the ones starting with a $K$-debris edge, only keep the one with the least expected debris (committing to this path dominates committing to other paths starting with a $K$-debris edge, and once we commit to this path, the optimal policy is to traverse it until the demand node).

If all paths start with a $K$-debris edge, it is optimal to traverse the one with the least expected debris. Otherwise, formulate an MDP with the same state definition as defined for two paths, except that the index $j = 1$ for the path that has been traversed until its first $K$-debris edge and has the least expected resource requirement remaining (the "best" path so far), and the index $j = 2$ for the current path. Since there is no need to store the number of 1-debris edges encountered so far on the paths and the current path, the number of states is $O(|E|^2)$. The only actions are to traverse the best path so far until the demand node or to start exploring the next best unexplored path. In the worst case, one needs to explore all paths before choosing the best one to proceed with. Hence, the number of decision epochs is at most $k$. With $O(|E|^2)$ states, a constant number of actions, and $k$ decision epochs, the Markov decision process can be solved using dynamic programming in $O(k|E|^4)$ time. □

*Proof of Theorem 4.* When $T$ is sufficiently large, the problem is one of minimizing the time to reach the last demand node, which is identical to minimizing the makespan on a single machine where jobs have stochastic task times. As shown by Rothkopf (1966), the optimal solution for this problem is to sequence the jobs (demand nodes) in the order of decreasing ratio of job weights (per-unit benefit times demand on the nodes) to expected task times (expected debris on the remaining unreachable edges on the path). □

## EC.2. Illustrative Examples

While the curse of dimensionality prevents finding optimal solution for large-sized instances, availability of an exact solution procedure may facilitate deriving structural results from smaller instances. In this section, we show that even on simple networks with unit demand, it is hard to derive structural results due to the periodic nature of the problem. We also use an example on a small illustrative network to show that solving SDCP using the POMDP model may provide significant benefits compared to solving a deterministic model with expected debris amounts, and that making regional updates may bring additional benefits in certain cases. The last part of this section introduces a continuous-time approximation, which provides the baseline for the heuristic approach discussed in the paper.

### EC.2.1. An Illustrative Example on Path Networks with Unit Demand

In this part, we will show that even for a simple graph structure (a path network) simple assumptions on the supply, demand, and beliefs on the resource requirements, optimal policies do not yield an easily generalizable structure. The insights on the difficulty of deriving structural properties even under very restrictive and simple conditions motivates us to use a continuous-time approximation and develop heuristic procedures based on it in §4.

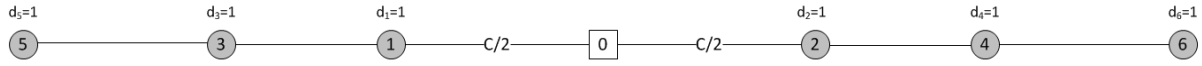


**Figure EC.2** An instance on a path with six demand nodes, each with unit demand

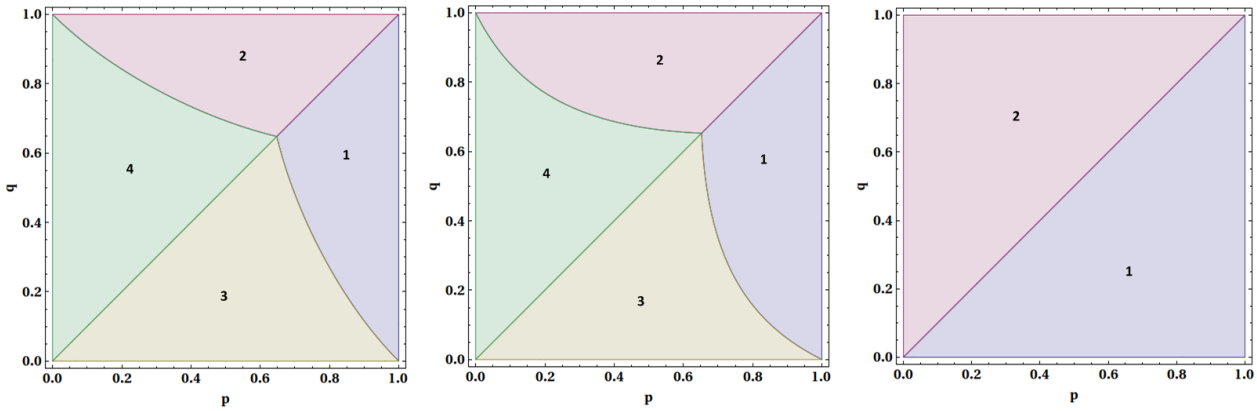


**Figure EC.3** Optimal policies for the instance in Figure EC.2 for different $\beta$ values and combinations of $p$ and $q$

Figure EC.2 presents an example with six demand nodes, each with unit demand, a single supply node with unlimited supply (node 0), and a constant clearance capacity of $C$ units per period. Clearance starts from the supply node. The roads incident to the supply node have $\frac{C}{2}$ units of debris and for each unreachable road, the two possible debris amounts are $\beta C$ and $(1-\beta)C$, where $0<\beta<\frac{1}{2}$. For the unreachable roads to the left and right of the supply node, corresponding belief probabilities for $\beta C$ are $p$ and $q$ respectively, and the planning horizon is four periods. Our aim is to determine the optimal clearance policy based on the combinations of values of $\beta$, $p$, and $q$.

In Figure EC.3, plots of the optimal policies based on varying values of the parameters are given. When $\beta \leq \frac{1}{4}$, four possible policies can be optimal:

- Policy 1 first clears the edges to the right of the supply node, then clears the edges to the left.
- Policy 2 first clears the edges to the left of the supply node, then clears the edges to the right.
- Policy 3 starts by clearing edges $(0,1)$ and $(0,2)$ first. Then it proceeds left unless there is a reachable low-debris edge on the right, in which case it clears this low-debris edge before proceeding to the left.
- Policy 4 starts by clearing edges $(0,1)$ and $(0,2)$ first. Then it proceeds right unless there is a reachable low-debris edge on the left, in which case it clears this low-debris edge before proceeding to the right.

The results in Figure EC.3 demonstrate that when $\beta$ is small and $|p-q|$ is large, the optimal policy favors clearing the roads on the side with less expected debris. When the possible debris amount variability is high and belief probabilities are closer to 0.5, the optimal policy favors

learning the debris structure over taking the actions that bring the most immediate benefit. While this behavior is intuitive, an interesting observation is the behavior of the optimal policy when $\beta$ changes. Until the threshold value of $\beta = \frac{1}{4}$, policies 1 and 3 co-exist for different values of $p$ and $q$, whereas after this threshold is passed, only policy 1 can be optimal.

The length of the path also determines the possible optimal policies. When there are two demand nodes on each side, optimal policies resemble policies 3 and 4 only, regardless of the value of $\beta$, whereas with four demand nodes on each side, more complicated policies concurrently involving both learning and switching to the side with less expected debris to reach immediate benefits can be optimal.

The main difficulty in generalizing the optimal policies for more complicated network, supply, demand, and belief structures arises from the periodic nature of the problem, as this leads to a substantially large action space, making many of these actions candidates to be included in the optimal policy. Furthermore, finiteness of the network prevents stationary policies from becoming optimal. The example given in this section has many restrictive simplifications such as unit demand, path network structure, and same belief amounts and probabilities for each side. In more general instances, the interdependency between these components of the problem make it even harder to generalize the results obtained in this section, adding another dimension of complexity to the problem.

### EC.2.2. An Illustrative Example to Compare Different Solution Approaches to SDCP

Figure EC.4 gives the network settings for a small illustrative example that we use to compare three different approaches for solving SDCP. The first of these approaches is to solve the problem using a deterministic algorithm in which we assume the resource requirements on all road segments are set to their expectations. The second approach solves SDCP using the POMDP model without the regional update mechanism described in §3.3, whereas the last approach adds on regional updates.

The example consists of a single supply node (given by node 0 in Figure EC.4), from which the clearance starts, and assumes unlimited supply capacity. There are 12 demand nodes for a single commodity, each with unit demand at the beginning of the first period and no additional demand in the coming periods. Debris amounts on each unreachable road segment can be either 5 or 15 units (actual debris amounts are given in Figure EC.4) and the clearance capacity in each period is 20 units. For each initially unreachable road to the left and right of the supply node, corresponding belief probabilities for 5 units of debris are 0.5 and 0.475, respectively. Planning horizon consists of seven periods. The regional update approach considers two regions (roads on the left and right of the supply node, respectively).

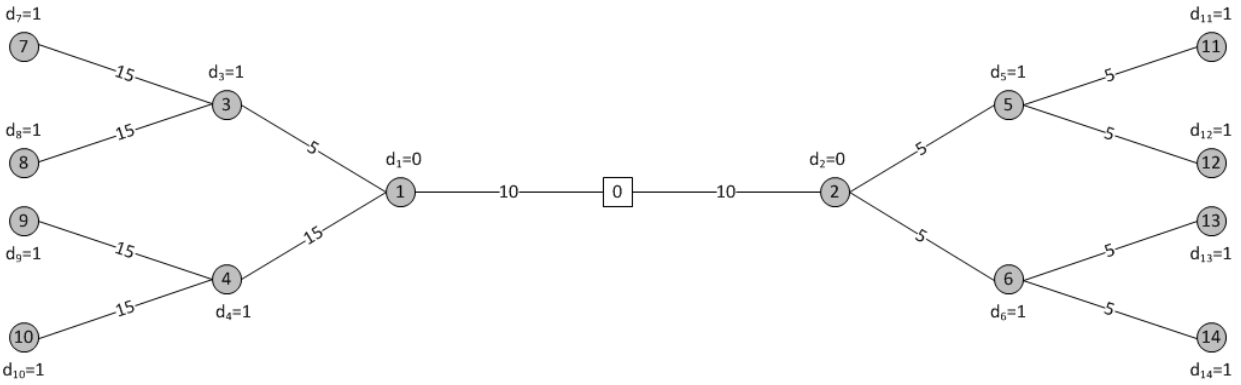


**Figure EC.4** An example network with 12 demand nodes, each with unit demand. Quantities on the road segments show actual debris amounts.

Figure EC.5 gives the clearance sequences found by solving the problem using the three approaches. Because of the homogeneous unit demand as well as identical potential debris amounts and belief probabilities within each side of the network, the expected value approach clears the roads on the left side first and proceeds with the roads on the right once the clearance on the left side is complete. This approach results in a total benefit of 40 units. The POMDP approach without regional updates starts by clearing the roads adjacent to the supply node, proceeding to clear the roads on the top-left, on the right, and lastly the ones on the bottom-left, resulting in a total benefit of 50 units.

The comparison between the first two approaches in this small example illustrates the trade-off between taking the actions that bring the most immediate benefit versus learning the debris structure. While the deterministic approach favors the former by clearing first the roads that seemingly have less debris, the stochastic approach first collects information about the surrounding blocked roads and takes actions based on the results of this learning process. The difference between these two approaches is prominent when, as in this example, belief distribution is close to uniform and actual amounts have lower belief probability.

When regional updates are also taken into account, the POMDP approach clears the roads on the right side of the supply node after observing the debris amounts on edges $(1,3)$, $(1,4)$, $(2,5)$, and $(2,6)$. In this way, it avoids clearing the high-debris roads on the top-left, resulting in a total benefit of 52 units. In general, making regional updates tends to be beneficial compared to the POMDP approach without updates when the belief probabilities are close to uniform and within-region variability is low, as in this small example (The results in Tables EC.1 and EC.2 in §6 provide a more detailed analysis of the effects of initial beliefs on the difference between these two approaches).

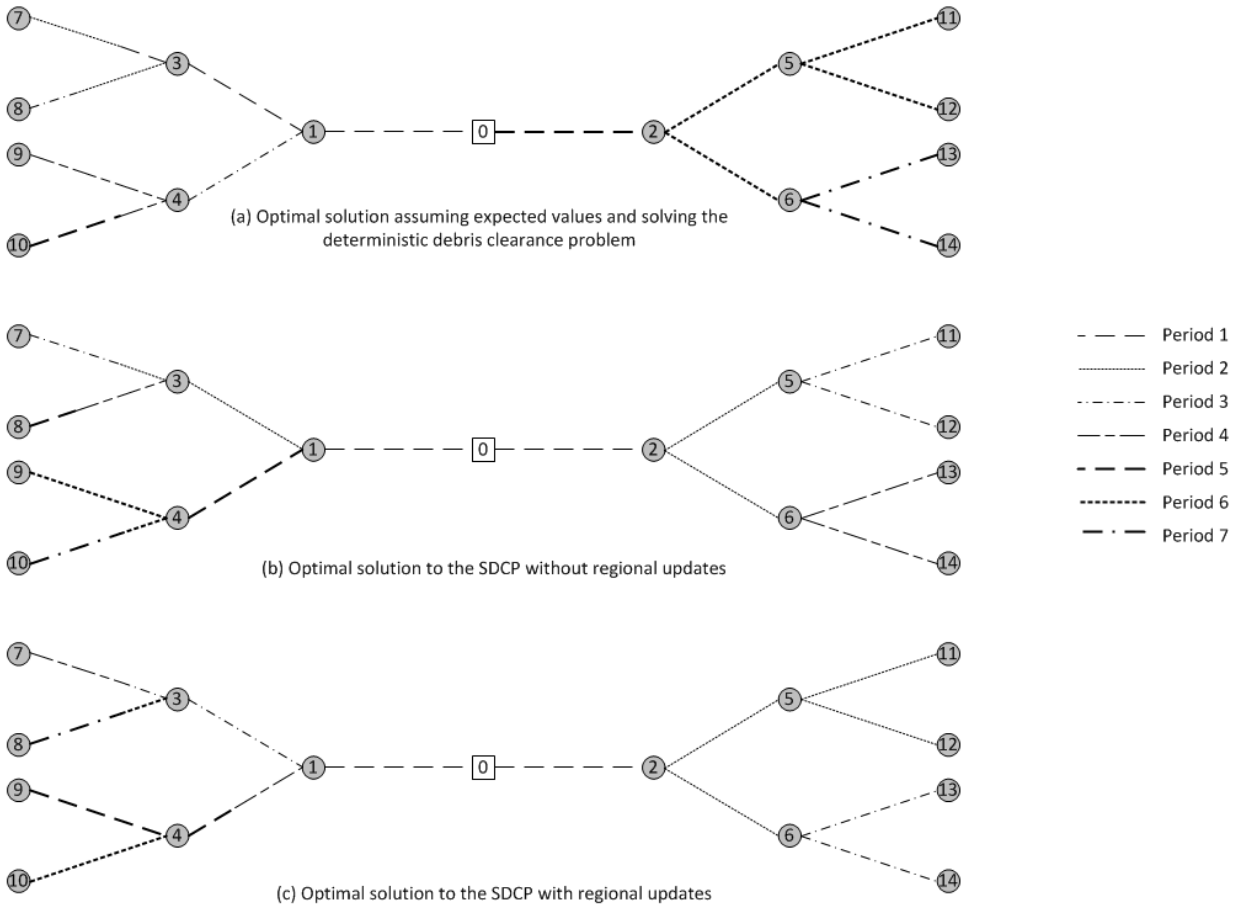


**Figure EC.5** Optimal solutions for each of the three approaches to solving the SDCP instance in Figure EC.4

This small example illustrates that although solving the POMDP model has more computational burden compared to the deterministic model with expected debris amounts, it improves the accrued benefit. Furthermore, under certain conditions, update of beliefs based on regional information gained as clearance proceeds may lead to additional benefits. A more detailed analysis of the results from applying different approaches to SDCP is made in §6.

### EC.2.3. Examples on Infinite Paths with Unit Demand

In the previous example, the main difficulty in deriving structural results for SDCP arises from the periodic structure of the problem, which results in the action space to be substantially large. In this part, we analyze SDCP on infinite path networks and attempt to observe the effect of stationary policies on the structure of the optimal solutions. The following analysis shows that for the original problem, the computational burden is still significant and the analytical results are not simple. We also observe that a continuous-time approximation decreases the computational burden significantly, allowing us to solve more general problems in shorter time. The results obtained in this part extend immediately to infinite stars and trees of the same demand and belief structure.

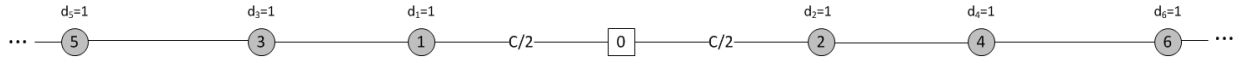


**Figure EC.6** An instance on an infinite path with unit demand on all demand nodes

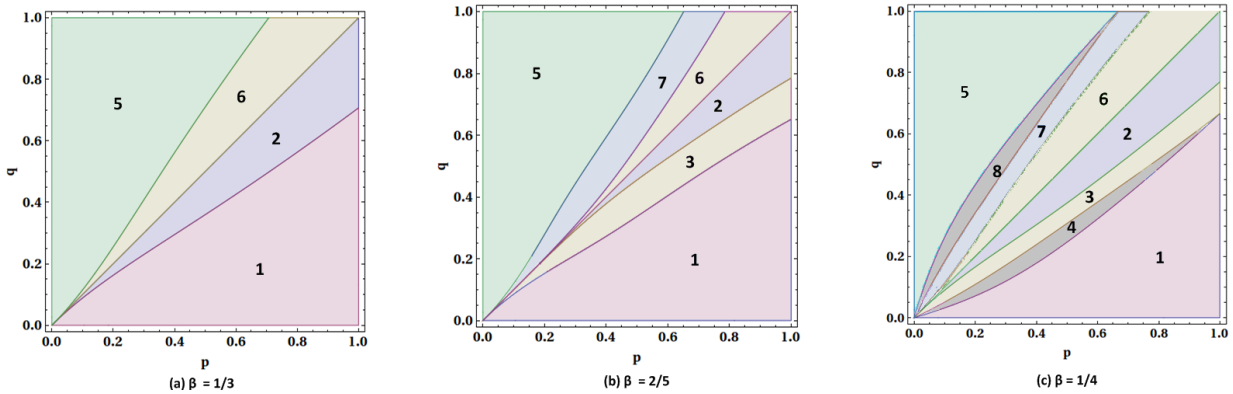


**Figure EC.7** Values of $p$ and $q$ for which each policy becomes optimal for different $\beta$ values when $\lambda = 0.5$

Figure EC.6 shows the infinite extension of the path network in Figure EC.2. As before, all nodes have unit demand, there is unlimited supply, and clearance starts from the supply node. There is a clearance capacity of $C$ units per period, roads incident to the supply node have $\frac{C}{2}$ units of debris (without loss of generality), whereas the unreachable roads on the left and right of the supply node have $\beta C$ units of debris with probability $p$ and $q$, respectively. Due to the infinite nature of the problem, the objective is modified to maximize the discounted benefit obtained. Each demand node reached accrues a single unit of benefit and there is a per-period discount factor of $\lambda$.

The problem can be modeled as a MDP with the state definition consisting of four components: The first two components state the side (left or right) of ongoing clearance from the previous period at the beginning of the period and remaining capacity after ongoing clearance is complete. The last two components give the amount of debris on the nearest reachable road on both sides, unless there is ongoing clearance from previous period on that side. Action definition is identical to that given in the original POMDP model. The reward of each action is the expectation over all possible debris observations of the number of nodes reached in that period.

Our aim in solving the MDP is to find the set of parameters for which each stationary policy is optimal. To do so, we first pick a candidate stationary policy to be optimal for a subset of parameters and apply a policy iteration approach. That is, we first find the expected discounted reward under the given policy, and check if switching to another policy improves the reward or not.

Figure EC.7 gives the values of $p$ and $q$ that make different policies optimal for varying values of $\beta$ when $\lambda$ is 0.5. A common characteristic of all optimal policies is that in any given state,

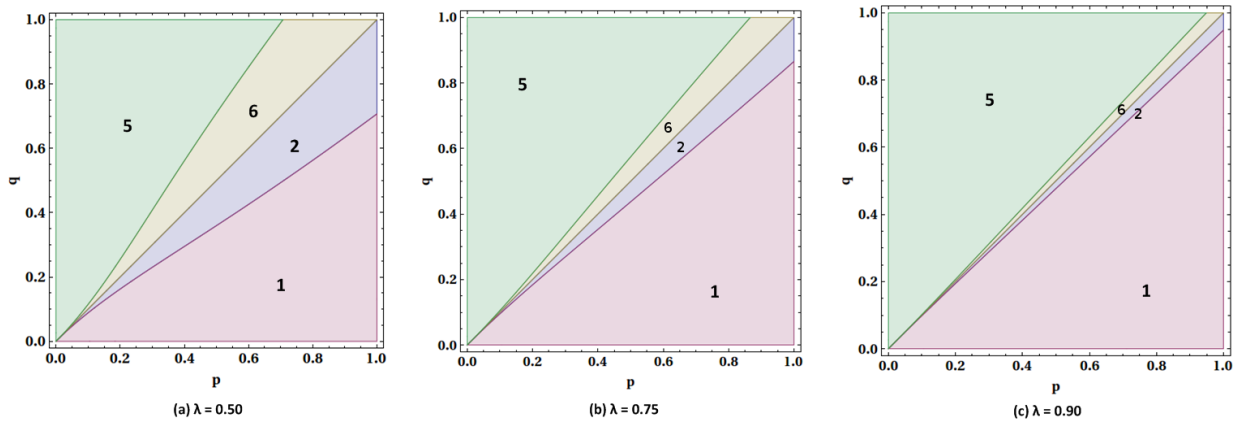


**Figure EC.8** Values of $p$ and $q$ for which each policy becomes optimal for different $\lambda$ values when $\beta = \frac{1}{3}$

the optimal clearance sequence in each period starts by finishing the ongoing clearance from the previous period and clearing any low-debris roads that are reachable at the beginning of the period. After these initial clearance actions, Policy 1 consists of persistently clearing the roads on the left. Policy 2 persistently clears the roads on the right if there is high debris on the left and low debris on the right at the beginning of the period. Policies 3 and 4 first clear one and two road segments on the right, respectively, before going left, if the initial debris on the right is low (otherwise they keep going left). Policies 5 through 8 are symmetric versions of policies 1 through 4 when $p \leq q$. The results show that as the values of $p$ and $q$ are closer to each other, learning the debris structure is a good strategy, whereas when these values are farther apart, greedily moving to the side with less expected debris gives better expected benefit.

An interesting observation from Figure EC.7 is the fact that the number of potentially optimal policies is not monotonic with respect to the value of $\beta$. With $\beta$ values of $\frac{1}{4}$, $\frac{1}{3}$, and $\frac{2}{5}$, the number of potentially optimal policies is eight, four, and six, respectively. The number of possible states is determined by where a partial clearance might start in the beginning of a period, which is not monotonic with respect to $\beta$. Since the number of potentially optimal policies is mainly determined by the number of possible states, the former is not monotonic with respect to $\beta$.

In Figure EC.8, the optimal policies for varying values of $p$ and $q$ are shown for different levels of $\lambda$ when $\beta$ is $\frac{1}{3}$. Changing the values of $\lambda$ does not change the number of potentially optimal policies, since the difference between potential debris levels is constant, which results in the same set of potential states, and consequently, same set of potentially optimal policies. The only change is in the values of $p$ and $q$ for which the policies can be optimal. As expected, increasing the value of $\lambda$ favors taking the actions that bring the most immediate benefit, since in the long run the side with less expected debris accrues more benefits. Consequently, letting $\lambda = 1$ leads to only policies 1 or 8 being potentially optimal, as the need to reach a demand node in a timely manner is completely ignored.

### EC.2.4. A Continuous Approximation

As in the case with finite networks, applying a periodic policy produces results that are hard to compute and generalize when the network structure is more complicated. We next study the case in which instead of making the clearance sequencing decisions at the beginning of each period, we can update the clearance resource requirement information and decide which road to clear next after clearing each blocked edge. Since this implies a continuous-time model, we replace $\lambda$ by a continuous discount factor $\alpha = -ln\,\lambda$. We also allow the number of possible requirement levels on each side to be more than two. These possible debris levels are also allowed to be different on the left and right side. Let $W_{sl}$ denote the debris volume corresponding to debris level $l$ on side $s \in \{L, R\}$ and, without loss of generality, let $W_{s1} \leq \ldots \leq W_{N_S}$, where $n$ denotes the number of requirement levels on side $s$. Let $p_s(l)$ be the corresponding probability of observing debris level $l$ on side $s$.

Applying such an approximation scheme reduces the sizes of the state and action spaces drastically. In this case, state space $S$ can be defined by $(W_L, W_R)$, the resource requirements on the left and right at the beginning of each clearance epoch. Action space reduces to going to either left or right. The corresponding Bellman equation is given by:

$$v^*(W_L, W_R) = max\Bigg\{1 + e^{-\alpha W_L} \sum_{(u, W_R \in S)} p_L(u)\, v^*(u, W_R),\ 1 + e^{-\alpha W_R} \sum_{(W_L, u \in S)} p_R(u)\, v^*(W_L, u)\Bigg\}, \tag{EC.1}$$

where $v^*(W_L, W_R)$ is the optimal value of the state $(W_L, W_R)$. This set of equations can be easily solved by value or policy iteration. Since the number of possible states is significantly reduced and there are only two possible actions, significant computational burden is avoided.

For the special case of two potential requirement levels on each side ($W_{L1}$ and $W_{L2}$ on the left and $W_{R1}$ and $W_{R2}$ on the right), the optimal policy is given by Algorithm 1. While the algorithm does not give insight into how the optimal policies compare to those for the above examples, it does illustrate the fact that the continuous approximation can handle any number of possible debris amounts in a more efficient way.

**Algorithm 1** The algorithm to find the optimal policy with two possible requirement levels on each side

**if** $P_L(W_{L1}) + P_L(W_{L2}) \leq P_R(W_{R1}) + P_R(W_{R2})$ **then**
  **if** $\frac{1}{1-e^{-\alpha W_{R1}}} \geq 1 + e^{-\alpha W_{R2}} \frac{P_L(W_{L1}+P_L(W_{L2}))}{1-P_L(W_{L1})e^{-\alpha W_{L1}}-P_L(W_{L2})e^{-\alpha W_{L2}}}$ **then**
    Always go left
  **else**
    Go right on $(W_{L2}, W_{R1})$ and go left otherwise
  **end if**
**else**
  **if** $\frac{1}{1-e^{-\alpha W_{L1}}} \geq 1 + e^{-\alpha W_{L2}} \frac{P_R(W_{R1}+P_R(W_{R2}))}{1-P_R(W_{R1})e^{-\alpha W_{R1}}-P_R(W_{R2})e^{-\alpha W_{R2}}}$ **then**
    Always go right
  **else**
    Go left on $(W_{L1}, W_{R2})$ and go right otherwise
  **end if**
**end if**

## EC.3. Summaries of Heuristic and Benchmark Approaches

Algorithms 2 through 5 present summaries of the heuristic pruning, limited lookahead, perfect information, and demand-per-debris approaches, respectively.

**Algorithm 2** Summary of the generic *heuristic pruning* procedure

**Inputs:**
  $bs_1$ The initial belief state
  $\underline{v}^*$ An initial lower bound on the objective value
  $\tau$ Threshold for $|UB_t|$
**while** $\exists$ belief nodes with $UB_t \neq \emptyset$ **do**
  Expand the next belief node with $|UB_t| \neq \emptyset$
  **for** each new belief node $i$ **do**
    **if** $|UB_t| \leq \tau$ under $i$ **then**
      Find an upper bound $\bar{v}_i$ for belief node $i$
      **if** $\bar{v}_i \leq \underline{v}^*$ **then**
        Prune node $i$
      **end if**
    **end if**
    **if** $UB_t = \emptyset$ for $i$ **then**
      Update expected benefit for each parent belief node using backward induction
      **if** $v(bs_1) > \underline{v}^*$ after the update **then**
        $\underline{v}^* \leftarrow v(bs_1)$
      **end if**
    **end if**
  **end for**
**end while**

**Algorithm 3** Summary of the generic *limited lookahead* procedure

**Inputs:**
  $bs_1$ The initial belief state
  $\delta$ Limit on search tree depth
  $j = 1$ Counter for tree depth
**while** $j < \delta$ **do**
  Expand all belief nodes on level $j$
  $j \leftarrow j + 1$
**end while**
**for** all belief nodes $i$ at depth $\delta$ **do**
  Approximate the benefit-to-go for node $i$ using a heuristic procedure
**end for**
Find best action under each belief node using backward induction and Bellman equation (21)

**Algorithm 4** Procedure to solve instances under perfect information (or expected amounts)

**Additional inputs:**
  Actual (or expected) debris amounts on each edge in $B_t$
**if** *optimal* **then**
  Solve the MIP model by Stilp et al. (2012) corresponding to the instance
**else if** *heuristic* **then**
  Use the *Push M* heuristic by Stilp et al. (2012) to obtain the clearance schedule
**end if**

**Algorithm 5** Summary of the *demand-per-debris* (DPD) procedure

**Additional inputs:**
  Expected debris amounts $w_{ij}$ $\forall (i,j) \in B_t$
  Edge weights $rw_{ij}$ $\forall (i,j) \in E$
**while** $\exists$ nodes in $N_D$ not connected to a node in $N_S$ **do**
  Among edges $(i,j)$ in $RB_t$ with minimum $rw_{ij}$, clear $(i,j)$ with maximum $\frac{rd_{j1}}{w_{ij}}$
  Update $U_t$, $RB_t$, and $UB_t$
  $t \leftarrow t + 1$
**end while**
Let $\pi = \{e_{(1)}, e_{(2)}, \ldots, e_{(|\pi|)}\}$ be the resulting sequence of edges. Convert to a feasible schedule:
$s \leftarrow 1$
**for** $t = 1..|T|$ **do**
  Schedule the first $k$ edges in $\pi$ such that $\sum_{l=s}^{s+k-1} w_{e_{(l)}} \leq k_t$ and $\sum_{l=s}^{s+k} w_{e_{(l)}} > k_t$ to clear in period $t$
  Remove the first $k$ edges from $\pi$
  $s \leftarrow s + k$
**end for**

## EC.4. Detailed Computational Results

Tables EC.1 and EC.2 give the perfect information gaps for the grid/ring and earthquake scenario instances, respectively. Since only small sized instances can be solved to optimality under perfect

**Algorithm 6** Summary of the *maximum flow per period* (MFP) procedure

**Additional inputs:**
Expected debris amounts $w_{ij}$ $\forall (i,j) \in B_t$
Edge weights $rw_{ij}$ $\forall (i,j) \in E$
**for** $t = 1..|T|$ **do**
  $\pi_t = \emptyset$: Sequence of edges to clear in period $t$
  $rc = k_t$
  **while** $rc > 0$ **do**
    **for** w=1..4 **do**
      Determine a sequence of edges $\{e(1), \ldots, e(l)\}$ using the heuristic by Stilp et al. (2012) subject to a clearance capacity of $rc$ on the subgraph consisting of edges $(i,j)$ in $U_t$ with $rw_{ij} = w$
      **for** s=1..l **do**
        $rc \leftarrow rc - w_{e(s)}$
        Add $e(s)$ to $\pi_t$
      **end for**
    **end for**
  **end while**
**end for**

information, for larger instances we use the solutions obtained by the heuristics in Stilp et al. (2012) to calculate the gaps. In Tables EC.1 and EC.2, the optimal and heuristic results are denoted by $O$ or $H$, respectively. Tables EC.3 and EC.4 show the average CPU times of each approach under each setting. For brevity, Table EC.3 only gives the CPU times for the case where all $p_{ij}$ follow a uniform distribution with parameters $U(0.5, 1)$. For the remaining scenarios, the CPU times are provided in Table EC.7 in the Appendix. A blank entry in each table indicates that the instances under the given setting cannot be solved due to insufficient memory.

**Table EC.1** Average perfect information gaps of the proposed and benchmark approaches on grid/ring instances

| | | | Perfect information | Deterministic with expected amounts | | No regional updates | | | | | With regional updates | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | Periodic | Continuous approximation | | | | Periodic | Continuous approximation | | | |
| Distribution of $p_{ij}$ | Instance | % roads blocked | O - optimal H - heuristic | Optimal | Heuristic | POMDP | POMDP | HP | LL | HP & LL | POMDP | POMDP | HP | LL | HP & LL |
| $U(0,0.5)$ | $G(5\times5)$ | 100 | O | 41.41 | 44.30 | 22.75 | 24.43 | 25.14 | 25.46 | 26.37 | 23.77 | 24.45 | 25.34 | 26.40 | 28.57 |
| | $G(10\times10)$ | 25 | O | 45.47 | 49.99 | 26.17 | 28.77 | 30.16 | 30.35 | 32.19 | 26.84 | 28.91 | 29.88 | 31.43 | 33.00 |
| | $G(10\times10)$ | 50 | H | *** | 56.22 | *** | 31.41 | 33.09 | 33.23 | 33.94 | *** | 31.30 | 32.28 | 33.07 | 33.74 |
| | $G(10\times10)$ | 100 | H | *** | 57.21 | *** | *** | 34.52 | 35.08 | 35.51 | *** | *** | 32.88 | 34.98 | 36.05 |
| | $G(15\times15)$ | 25 | H | *** | 48.29 | *** | 28.15 | 30.35 | 30.92 | 32.84 | *** | 28.27 | 30.97 | 31.20 | 32.58 |
| | $G(15\times15)$ | 50 | H | *** | 55.07 | *** | *** | *** | 32.27 | 33.19 | *** | *** | *** | 31.72 | 34.61 |
| | $G(15\times15)$ | 100 | H | *** | 63.23 | *** | *** | *** | 33.68 | 34.93 | *** | *** | *** | 33.60 | 34.83 |
| | $R(5\times5)$ | 100 | O | 39.22 | 42.92 | 25.76 | 27.02 | 29.01 | 29.11 | 30.31 | 26.03 | 27.36 | 28.22 | 28.96 | 32.41 |
| | $R(10\times10)$ | 25 | O | 45.52 | 48.62 | 28.48 | 28.91 | 29.37 | 29.93 | 31.49 | 29.40 | 30.54 | 32.47 | 32.57 | 33.81 |
| | $R(10\times10)$ | 50 | H | *** | 61.03 | *** | 30.89 | 31.30 | 32.47 | 33.01 | *** | 31.18 | 33.17 | 33.78 | 34.71 |
| | $R(10\times10)$ | 100 | H | *** | 64.06 | *** | *** | 33.07 | 34.20 | 35.63 | *** | *** | 32.23 | 34.47 | 35.60 |
| | $R(15\times15)$ | 25 | H | *** | 57.12 | *** | 30.38 | 32.59 | 33.50 | 34.00 | *** | 30.28 | 33.10 | 33.82 | 35.20 |
| | $R(15\times15)$ | 50 | H | *** | 60.96 | *** | *** | *** | 35.01 | 36.22 | *** | *** | *** | 35.78 | 36.93 |
| | $R(15\times15)$ | 100 | H | *** | 63.93 | *** | *** | *** | 39.14 | 41.17 | *** | *** | *** | 37.82 | 39.68 |
| $U(0,1)$ | $G(5\times5)$ | 100 | O | 33.09 | 34.32 | 29.16 | 31.08 | 31.63 | 33.38 | 34.78 | 26.83 | 26.93 | 28.41 | 29.66 | 29.82 |
| | $G(10\times10)$ | 25 | O | 37.38 | 38.27 | 33.03 | 36.64 | 37.87 | 39.86 | 41.68 | 30.33 | 32.83 | 33.91 | 34.72 | 36.54 |
| | $G(10\times10)$ | 50 | H | *** | 43.31 | *** | 39.53 | 41.74 | 41.86 | 43.75 | *** | 35.65 | 38.27 | 39.46 | 41.34 |
| | $G(10\times10)$ | 100 | H | *** | 46.05 | *** | *** | 43.19 | 42.47 | 45.89 | *** | *** | 38.80 | 39.29 | 42.37 |
| | $G(15\times15)$ | 25 | H | *** | 38.94 | *** | 36.09 | 38.92 | 39.70 | 41.53 | *** | 34.62 | 35.35 | 34.78 | 37.65 |
| | $G(15\times15)$ | 50 | H | *** | 43.29 | *** | *** | *** | 40.92 | 42.65 | *** | *** | *** | 37.51 | 40.01 |
| | $G(15\times15)$ | 100 | H | *** | 50.40 | *** | *** | *** | 43.37 | 43.86 | *** | *** | *** | 39.50 | 41.38 |
| | $R(5\times5)$ | 100 | O | 33.36 | 34.11 | 30.69 | 33.06 | 34.95 | 34.60 | 35.28 | 29.19 | 30.79 | 32.00 | 34.07 | 35.32 |
| | $R(10\times10)$ | 25 | O | 38.20 | 39.36 | 35.23 | 36.63 | 38.21 | 39.67 | 42.63 | 33.00 | 32.80 | 34.80 | 36.26 | 38.78 |
| | $R(10\times10)$ | 50 | H | *** | 48.01 | *** | 37.84 | 39.51 | 41.53 | 42.53 | *** | 34.67 | 35.51 | 37.37 | 39.78 |
| | $R(10\times10)$ | 100 | H | *** | 50.04 | *** | *** | 41.13 | 41.63 | 44.23 | *** | *** | 36.94 | 37.64 | 41.51 |
| | $R(15\times15)$ | 25 | H | *** | 45.40 | *** | 39.81 | 41.43 | 42.71 | 45.81 | *** | 36.19 | 37.49 | 38.24 | 39.97 |
| | $R(15\times15)$ | 50 | H | *** | 48.73 | *** | *** | *** | 44.61 | 46.07 | *** | *** | *** | 38.53 | 41.57 |
| | $R(15\times15)$ | 100 | H | *** | 49.38 | *** | *** | *** | 47.70 | 49.93 | *** | *** | *** | 43.33 | 43.75 |
| $U(0.25,0.75)$ | $G(5\times5)$ | 100 | O | 33.09 | 34.32 | 25.43 | 27.19 | 29.84 | 30.94 | 31.24 | 23.20 | 25.16 | 26.73 | 27.92 | 30.19 |
| | $G(10\times10)$ | 25 | O | 37.38 | 38.27 | 28.91 | 32.78 | 34.70 | 35.51 | 36.79 | 26.31 | 28.88 | 31.17 | 33.48 | 33.99 |
| | $G(10\times10)$ | 50 | H | *** | 43.31 | *** | 35.52 | 35.88 | 37.45 | 37.64 | *** | 32.78 | 33.79 | 33.52 | 34.21 |
| | $G(10\times10)$ | 100 | H | *** | 46.05 | *** | *** | 37.87 | 38.22 | 39.09 | *** | *** | 34.45 | 35.19 | 35.48 |
| | $G(15\times15)$ | 25 | H | *** | 38.94 | *** | 31.70 | 33.22 | 34.01 | 36.23 | *** | 29.59 | 30.64 | 31.15 | 32.63 |
| | $G(15\times15)$ | 50 | H | *** | 43.29 | *** | *** | *** | 37.65 | 38.07 | *** | *** | *** | 34.10 | 34.41 |
| | $G(15\times15)$ | 100 | H | *** | 50.40 | *** | *** | *** | 37.40 | 37.97 | *** | *** | *** | 33.90 | 35.94 |
| | $R(5\times5)$ | 100 | O | 33.36 | 34.11 | 28.03 | 29.87 | 31.60 | 30.61 | 31.80 | 25.67 | 28.12 | 27.91 | 29.55 | 31.43 |
| | $R(10\times10)$ | 25 | O | 38.20 | 39.36 | 30.99 | 33.01 | 33.45 | 34.39 | 34.67 | 30.31 | 31.41 | 31.72 | 31.87 | 33.07 |
| | $R(10\times10)$ | 50 | H | *** | 48.01 | *** | 33.69 | 34.21 | 34.44 | 35.94 | *** | 32.47 | 32.60 | 32.27 | 34.03 |
| | $R(10\times10)$ | 100 | H | *** | 50.04 | *** | *** | 34.59 | 35.86 | 35.99 | *** | *** | 32.76 | 33.31 | 34.45 |
| | $R(15\times15)$ | 25 | H | *** | 45.40 | *** | 35.25 | 36.89 | 37.02 | 38.09 | *** | 32.50 | 33.68 | 34.54 | 35.82 |
| | $R(15\times15)$ | 50 | H | *** | 48.73 | *** | *** | *** | 37.93 | 39.44 | *** | *** | *** | 34.84 | 37.21 |
| | $R(15\times15)$ | 100 | H | *** | 49.38 | *** | *** | *** | 44.28 | 45.89 | *** | *** | *** | 37.69 | 39.87 |
| $U(0.5,1)$ | $G(5\times5)$ | 100 | O | 27.02 | 29.60 | 19.37 | 20.80 | 21.35 | 21.46 | 21.80 | 18.95 | 19.25 | 20.14 | 20.99 | 20.58 |
| | $G(10\times10)$ | 25 | O | 30.96 | 34.14 | 21.13 | 21.52 | 22.02 | 22.27 | 22.48 | 21.04 | 22.03 | 22.92 | 22.49 | 23.35 |
| | $G(10\times10)$ | 50 | H | *** | 37.25 | *** | 24.52 | 25.07 | 25.73 | 25.60 | *** | 23.49 | 23.77 | 24.33 | 24.35 |
| | $G(10\times10)$ | 100 | H | *** | 38.69 | *** | *** | 27.25 | 27.36 | 27.51 | *** | *** | 25.39 | 25.68 | 26.41 |
| | $G(15\times15)$ | 25 | H | *** | 33.55 | *** | 23.95 | 25.49 | 25.62 | 25.62 | *** | 22.80 | 24.16 | 24.36 | 24.66 |
| | $G(15\times15)$ | 50 | H | *** | 35.74 | *** | *** | *** | 26.77 | 26.86 | *** | *** | *** | 25.20 | 25.50 |
| | $G(15\times15)$ | 100 | H | *** | 39.84 | *** | *** | *** | 27.82 | 27.83 | *** | *** | *** | 25.96 | 26.69 |
| | $R(5\times5)$ | 100 | O | 28.74 | 30.37 | 21.88 | 23.30 | 23.19 | 23.29 | 23.45 | 20.94 | 22.19 | 23.58 | 23.52 | 23.56 |
| | $R(10\times10)$ | 25 | O | 31.58 | 33.15 | 22.98 | 24.81 | 24.99 | 25.21 | 25.54 | 22.72 | 24.58 | 25.06 | 25.74 | 25.69 |
| | $R(10\times10)$ | 50 | H | *** | 39.12 | *** | 24.90 | 25.00 | 25.77 | 26.43 | *** | 24.48 | 25.70 | 25.94 | 26.12 |
| | $R(10\times10)$ | 100 | H | *** | 40.72 | *** | *** | 26.12 | 26.44 | 26.70 | *** | *** | 26.33 | 26.67 | 26.56 |
| | $R(15\times15)$ | 25 | H | *** | 37.13 | *** | 26.27 | 27.61 | 28.02 | 28.03 | *** | 24.84 | 26.09 | 26.69 | 26.49 |
| | $R(15\times15)$ | 50 | H | *** | 40.86 | *** | *** | *** | 28.70 | 29.01 | *** | *** | *** | 26.75 | 27.24 |
| | $R(15\times15)$ | 100 | H | *** | 42.32 | *** | *** | *** | 30.51 | 30.81 | *** | *** | *** | 28.96 | 29.18 |

**Table EC.2** Average perfect information gaps of the proposed and benchmark approaches on earthquake scenario instances

| | | Perfect information | Deterministic with expected amounts | | Predetermined road priorities | | | No regional updates | | | | | With regional updates | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | Periodic | Continuous approximation | | | | Periodic | Continuous approximation | | | |
| Distribution of $p_{ij}$ | % roads blocked | O - optimal H - heuristic | Optimal | Heuristic | DPD | DPDW | MFP | POMDP | POMDP | HP | LL | HP & LL | POMDP | POMDP | HP | LL | HP & LL |
| $U(0,0.5)$ | 25 | H | *** | 49.85 | 61.58 | 59.10 | 51.22 | *** | *** | 25.47 | 28.38 | 31.29 | *** | *** | 23.82 | 25.58 | 27.52 |
| | 50 | H | *** | 52.04 | 66.69 | 64.54 | 60.46 | *** | *** | *** | 33.52 | 37.16 | *** | *** | *** | 29.57 | 32.41 |
| | 100 | H | *** | 57.56 | 75.18 | 73.61 | 68.24 | *** | *** | *** | 39.24 | 42.90 | *** | *** | *** | 32.65 | 36.13 |
| $U(0,1)$ | 25 | H | *** | 38.20 | 56.15 | 54.81 | 49.81 | *** | *** | 31.35 | 33.75 | 35.99 | *** | *** | 25.45 | 27.78 | 29.12 |
| | 50 | H | *** | 45.68 | 58.93 | 56.65 | 52.15 | *** | *** | *** | 42.46 | 45.01 | *** | *** | *** | 34.67 | 37.14 |
| | 100 | H | *** | 47.36 | 64.82 | 61.31 | 58.84 | *** | *** | *** | 47.51 | 49.12 | *** | *** | *** | 36.30 | 38.17 |
| $U(0.25,0.75)$ | 25 | H | *** | 38.20 | 56.15 | 54.81 | 49.81 | *** | *** | 26.86 | 29.53 | 32.50 | *** | *** | 26.38 | 28.35 | 31.08 |
| | 50 | H | *** | 45.68 | 58.93 | 56.65 | 52.15 | *** | *** | *** | 36.76 | 39.90 | *** | *** | *** | 30.18 | 33.21 |
| | 100 | H | *** | 47.36 | 64.82 | 61.31 | 58.84 | *** | *** | *** | 38.90 | 41.56 | *** | *** | *** | 31.62 | 34.70 |
| $U(0.5,1)$ | 25 | H | *** | 33.72 | 51.86 | 48.68 | 42.69 | *** | *** | 20.13 | 21.10 | 21.74 | *** | *** | 20.19 | 21.34 | 22.11 |
| | 50 | H | *** | 35.23 | 58.55 | 55.02 | 44.30 | *** | *** | *** | 21.30 | 23.71 | *** | *** | *** | 23.40 | 26.00 |
| | 100 | H | *** | 39.14 | 61.74 | 58.52 | 53.11 | *** | *** | *** | 25.12 | 27.71 | *** | *** | *** | 26.67 | 29.93 |

**Table EC.3** Average CPU times (in seconds) for the proposed and benchmark approaches on grid/ring instances when $p_{ij} \sim U(0.5,1)$

| | | Deterministic with perfect information | | Deterministic with expected amounts | | No regional updates | | | | | With regional updates | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | Periodic | Continuous approximation | | | | Periodic | Continuous approximation | | | |
| Instance | % roads blocked | Optimal | Heuristic | Optimal | Heuristic | POMDP | POMDP | HP | LL | HP & LL | POMDP | POMDP | HP | LL | HP & LL |
| $G(5\times5)$ | 100 | 26.9 | 5.6 | 25.8 | 3.9 | 1,430.1 | 294.4 | 175.1 | 229.4 | 69.3 | 1,537.7 | 313.7 | 188.5 | 241.1 | 72.4 |
| $G(10\times10)$ | 25 | 78.6 | 12.9 | 72.8 | 11.5 | 6,103.2 | 462.3 | 238.2 | 314.4 | 92.7 | 6,221.5 | 481.5 | 249.6 | 323.8 | 97.3 |
| $G(10\times10)$ | 50 | *** | 36.8 | *** | 20.8 | *** | 2,262.4 | 1,119.8 | 1,433.7 | 429.6 | *** | 2,326.4 | 1,173.4 | 1,513.8 | 446.6 |
| $G(10\times10)$ | 100 | *** | 101.4 | *** | 76.7 | *** | *** | 1,990.9 | 2,645.7 | 783.2 | *** | *** | 2,114.9 | 2,786.9 | 817.7 |
| $G(15\times15)$ | 25 | *** | 25.9 | *** | 25.1 | *** | 482.4 | 241.2 | 321.7 | 96.0 | *** | 499.1 | 250.7 | 339.7 | 101.6 |
| $G(15\times15)$ | 50 | *** | 66.8 | *** | 28.6 | *** | *** | *** | 2,095.7 | 627.7 | *** | *** | *** | 2,236.5 | 674.2 |
| $G(15\times15)$ | 100 | *** | 132.5 | *** | 129.5 | *** | *** | *** | 3,900.5 | 1,175.2 | *** | *** | *** | 4,097.9 | 1,200.5 |
| $R(5\times5)$ | 100 | 34.8 | 6.1 | 29.5 | 4.2 | 1,083.2 | 203.8 | 118.4 | 152.7 | 46.0 | 1,166.4 | 212.2 | 123.6 | 159.3 | 47.9 |
| $R(10\times10)$ | 25 | 72.4 | 14.0 | 63.9 | 11.2 | 3,196.5 | 237.3 | 132.4 | 178.5 | 82.3 | 3,294.5 | 252.4 | 138.1 | 188.4 | 85.6 |
| $R(10\times10)$ | 50 | *** | 42.4 | *** | 26.6 | *** | 2,017.8 | 950.9 | 1,256.8 | 372.8 | *** | 2,096.2 | 1,008.9 | 1,335.5 | 387.5 |
| $R(10\times10)$ | 100 | *** | 116.1 | *** | 88.4 | *** | *** | 1,800.4 | 2,345.2 | 701.4 | *** | *** | 1,850.9 | 2,437.2 | 732.9 |
| $R(15\times15)$ | 25 | *** | 28.0 | *** | 13.9 | *** | 394.7 | 202.7 | 274.0 | 81.3 | *** | 409.7 | 213.7 | 288.0 | 83.9 |
| $R(15\times15)$ | 50 | *** | 70.5 | *** | 70.8 | *** | *** | *** | 2,009.7 | 604.5 | *** | *** | *** | 2,099.7 | 652.6 |
| $R(15\times15)$ | 100 | *** | 147.8 | *** | 129.3 | *** | *** | *** | 3,467.8 | 1,049.7 | *** | *** | *** | 3,639.5 | 1,116.3 |

**Table EC.4** Average CPU times (in seconds) for the proposed and benchmark approaches on earthquake scenario instances

| | | Deterministic with perfect information | | Deterministic with expected amounts | | Predetermined road priorities | | | No regional updates | | | | | With regional updates | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | Periodic | Continuous approximation | | | | Periodic | Continuous approximation | | | |
| Distribution of $p_{ij}$ | % roads blocked | Optimal | Heuristic | Optimal | Heuristic | DPD | DPDW | MFP | POMDP | POMDP | HP | LL | HP & LL | POMDP | POMDP | HP | LL | HP & LL |
| $U(0,0.5)$ | 25 | *** | 126 | *** | 139 | 2.6 | 2.7 | 104 | *** | *** | 20,431 | 8,503 | 1,980 | *** | *** | *** | 8,612 | 2,088 |
| | 50 | *** | 249 | *** | 274 | 3.8 | 3.6 | 205 | *** | *** | *** | 16,657 | 3,529 | *** | *** | *** | 16,742 | 3,663 |
| | 100 | *** | 333 | *** | 366 | 6.5 | 6.5 | 285 | *** | *** | *** | 23,980 | 7,731 | *** | *** | *** | 24,097 | 7,832 |
| $U(0,1)$ | 25 | *** | 113 | *** | 124 | 2.6 | 2.3 | 102 | *** | *** | 19,931 | 8,203 | 1,942 | *** | *** | *** | 8,257 | 2,018 |
| | 50 | *** | 204 | *** | 224 | 3.9 | 3.9 | 160 | *** | *** | *** | 15,294 | 3,971 | *** | *** | *** | 15,374 | 4,044 |
| | 100 | *** | 340 | *** | 375 | 6.4 | 6.1 | 160 | *** | *** | *** | 25,302 | 7,712 | *** | *** | *** | 25,426 | 7,786 |
| $U(0.25,0.75)$ | 25 | *** | 138 | *** | 152 | 2.6 | 2.5 | 114 | *** | *** | 20,317 | 7,709 | 2,171 | *** | *** | *** | 7,784 | 2,293 |
| | 50 | *** | 303 | *** | 333 | 4.2 | 3.9 | 244 | *** | *** | *** | 16,256 | 4,203 | *** | *** | *** | 16,350 | 4,327 |
| | 100 | *** | 472 | *** | 519 | 6.3 | 6.4 | 437 | *** | *** | *** | 26,046 | 7,803 | *** | *** | *** | 26,158 | 7,915 |
| $U(0.5,1)$ | 25 | *** | 173 | *** | 190 | 2.7 | 2.7 | 144 | *** | *** | 20,481 | 7,439 | 2,350 | *** | *** | *** | 7,549 | 2,415 |
| | 50 | *** | 365 | *** | 401 | 4.2 | 4.3 | 293 | *** | *** | *** | 15,540 | 4,753 | *** | *** | *** | 15,622 | 4,877 |
| | 100 | *** | 704 | *** | 774 | 6.3 | 6.4 | 625 | *** | *** | *** | 28,952 | 8,781 | *** | *** | *** | 29,059 | 8,258 |

## EC.5. Further Computational Results for Grid/Ring Instances

Table EC.5 gives the average percent gaps from perfect information solutions for grid/ring instances when $p_{ij}$ follows a uniform distribution with parameters $U(0.5, 1)$ and there are two supply nodes. Similarly, Table EC.6 shows the average gaps for the same belief settings with a single supply node, where clearance starts from outside the disaster area, instead of the supply node. For both cases, the results are in parallel with the baseline case we present in the main body of the paper, where there is a single supply node and the clearance starts from this supply node.

Table EC.7 extends the results given in Table EC.3 to the cases where $p_{ij}$ follows a uniform distribution with parameters $U(0, 0.5)$, $U(0, 1)$, $U(0.25, 0.75)$, and $U(0.5, 1)$.

**Table EC.5** Average perfect information gaps for grid/ring instances under $p_{ij}$ following a uniform distribution with parameters $U(0.5, 1)$ and two supply nodes, a blank entry indicates insufficient memory

| | | Deterministic with perfect information | | Deterministic with expected amounts | | No regional updates | | | | | With regional updates | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | Periodic | Continuous approximation | | | | Periodic | Continuous approximation | | | |
| Instance | % roads blocked | Optimal | Heuristic | Optimal | Heuristic | POMDP | POMDP | HP | LL | HP & LL | POMDP | POMDP | HP | LL | HP & LL |
| $G(5 \times 5)$ | 100 | 0.00 | 1.61 | 23.13 | 22.33 | 15.16 | 15.29 | 15.53 | 15.49 | 16.11 | 14.12 | 14.49 | 16.11 | 16.44 | 15.41 |
| $G(10 \times 10)$ | 25 | 0.00 | 3.12 | 25.55 | 25.98 | 16.01 | 17.79 | 18.57 | 18.97 | 18.62 | 15.86 | 17.86 | 18.00 | 18.95 | 17.78 |
| $G(10 \times 10)$ | 50 | *** | 4.01 | *** | 31.35 | *** | 20.02 | 19.82 | 21.25 | 20.77 | *** | 17.69 | 18.66 | 19.88 | 20.72 |
| $G(10 \times 10)$ | 100 | *** | 4.64 | *** | 33.09 | *** | *** | 21.10 | 20.81 | 21.38 | *** | *** | 20.30 | 21.13 | 19.76 |
| $G(15 \times 15)$ | 25 | *** | 4.45 | *** | 27.70 | *** | 18.30 | 19.10 | 19.83 | 19.21 | *** | 16.57 | 18.01 | 17.87 | 19.65 |
| $G(15 \times 15)$ | 50 | *** | 5.33 | *** | 29.23 | *** | 19.46 | 19.81 | 20.82 | 20.47 | *** | 18.17 | 19.99 | 18.79 | 19.79 |
| $G(15 \times 15)$ | 100 | *** | 6.10 | *** | 34.99 | *** | *** | 19.90 | 21.87 | 21.79 | *** | *** | 20.49 | 20.56 | 20.73 |
| $R(5 \times 5)$ | 100 | 0.00 | 2.46 | 25.14 | 25.32 | 15.86 | 16.37 | 17.78 | 18.18 | 18.65 | 15.29 | 15.95 | 16.93 | 17.17 | 16.58 |
| $R(10 \times 10)$ | 25 | 0.00 | 3.74 | 27.78 | 28.23 | 18.98 | 19.71 | 18.88 | 19.28 | 18.66 | 16.06 | 16.98 | 17.25 | 18.10 | 18.54 |
| $R(10 \times 10)$ | 50 | *** | 3.78 | *** | 33.07 | *** | 19.90 | 19.40 | 19.52 | 20.46 | *** | 18.47 | 17.85 | 18.18 | 19.49 |
| $R(10 \times 10)$ | 100 | *** | 4.09 | *** | 34.71 | *** | *** | 20.05 | 20.34 | 20.86 | *** | *** | 19.08 | 18.90 | 20.29 |
| $R(15 \times 15)$ | 25 | *** | 4.66 | *** | 31.88 | *** | 19.19 | 20.18 | 20.39 | 22.43 | *** | 19.45 | 20.32 | 20.60 | 19.59 |
| $R(15 \times 15)$ | 50 | *** | 5.01 | *** | 35.80 | *** | 19.64 | 21.90 | 20.65 | 20.71 | *** | 19.55 | 20.61 | 21.00 | 20.47 |
| $R(15 \times 15)$ | 100 | *** | 6.27 | *** | 35.87 | *** | *** | 22.13 | 22.64 | 24.15 | *** | *** | 21.39 | 21.29 | 21.85 |

**Table EC.6** Average perfect information gaps for grid/ring instances under $p_{ij}$ following a uniform distribution with parameters $U(0.5, 1)$ and clearance starting from outside the disaster area, a blank entry indicates insufficient memory

| | | Deterministic with perfect information | | Deterministic with expected amounts | | No regional updates | | | | | With regional updates | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | Periodic | Continuous approximation | | | | Periodic | Continuous approximation | | | |
| Instance | % roads blocked | Optimal | Heuristic | Optimal | Heuristic | POMDP | POMDP | HP | LL | HP & LL | POMDP | POMDP | HP | LL | HP & LL |
| $G(5 \times 5)$ | 100 | 0.00 | 1.58 | 43.03 | 44.60 | 23.45 | 23.77 | 24.46 | 25.67 | 25.99 | 22.84 | 24.30 | 26.49 | 27.31 | 29.14 |
| $G(10 \times 10)$ | 25 | 0.00 | 3.61 | 46.05 | 50.46 | 26.41 | 27.49 | 30.65 | 31.44 | 31.61 | 26.95 | 28.23 | 29.02 | 30.71 | 33.00 |
| $G(10 \times 10)$ | 50 | *** | 5.18 | *** | 55.37 | *** | 31.23 | 31.37 | 34.04 | 36.24 | *** | 29.49 | 31.67 | 32.25 | 32.19 |
| $G(10 \times 10)$ | 100 | *** | 3.64 | *** | 56.55 | *** | *** | 35.69 | 34.82 | 35.85 | *** | *** | 32.55 | 34.97 | 38.47 |
| $G(15 \times 15)$ | 25 | *** | 4.89 | *** | 49.06 | *** | 27.12 | 29.44 | 31.07 | 33.35 | *** | 29.11 | 29.27 | 31.65 | 32.49 |
| $G(15 \times 15)$ | 50 | *** | 5.60 | *** | 54.41 | *** | *** | *** | 33.28 | 31.94 | *** | *** | *** | 33.19 | 35.01 |
| $G(15 \times 15)$ | 100 | *** | 6.06 | *** | 65.81 | *** | *** | *** | 32.55 | 36.15 | *** | *** | *** | 32.84 | 36.76 |
| $R(5 \times 5)$ | 100 | 0.00 | 2.68 | 36.78 | 41.27 | 25.38 | 26.63 | 29.14 | 29.69 | 29.98 | 25.60 | 27.13 | 28.82 | 28.03 | 32.89 |
| $R(10 \times 10)$ | 25 | 0.00 | 3.25 | 43.49 | 49.90 | 27.92 | 29.11 | 28.55 | 29.25 | 29.64 | 29.12 | 30.20 | 32.39 | 33.29 | 33.54 |
| $R(10 \times 10)$ | 50 | *** | 3.82 | *** | 58.93 | *** | 31.73 | 32.69 | 33.25 | 32.53 | *** | 31.16 | 32.84 | 32.35 | 35.98 |
| $R(10 \times 10)$ | 100 | *** | 4.62 | *** | 64.04 | *** | *** | 33.23 | 36.57 | 34.66 | *** | *** | 31.17 | 34.03 | 36.24 |
| $R(15 \times 15)$ | 25 | *** | 4.91 | *** | 57.23 | *** | 29.15 | 32.51 | 33.23 | 34.88 | *** | 30.63 | 33.81 | 32.94 | 35.26 |
| $R(15 \times 15)$ | 50 | *** | 4.79 | *** | 58.49 | *** | *** | *** | 33.14 | 36.18 | *** | *** | *** | 34.48 | 36.80 |
| $R(15 \times 15)$ | 100 | *** | 6.16 | *** | 61.33 | *** | *** | *** | 39.27 | 39.30 | *** | *** | *** | 39.51 | 39.37 |

**Table EC.7** Average CPU times (in seconds) for the proposed and benchmark approaches on grid/ring instances, a blank entry indicates insufficient memory

| | | | Deterministic with perfect information | | Deterministic with expected amounts | | No regional updates | | | | | With regional updates | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | Periodic | Continuous approximation | | | | Periodic | Continuous approximation | | | |
| Distribution of $p_{ij}$ | Instance | % roads blocked | Optimal | Heuristic | Optimal | Heuristic | POMDP | POMDP | HP | LL | HP & LL | POMDP | POMDP | HP | LL | HP & LL |
| $U(0, 0.5)$ | $G(5\times5)$ | 100 | 24.89 | 6.10 | 27.50 | 3.55 | 1479.61 | 271.72 | 193.15 | 226.33 | 72.14 | 1343.72 | 294.66 | 190.78 | 212.89 | 66.38 |
| | $G(10\times10)$ | 25 | 81.77 | 13.89 | 74.31 | 11.83 | 6284.43 | 523.71 | 262.86 | 273.89 | 88.99 | 6181.25 | 453.48 | 275.34 | 312.10 | 86.57 |
| | $G(10\times10)$ | 50 | *** | 36.21 | *** | 21.47 | *** | 2173.06 | 1117.96 | 1320.89 | 407.74 | *** | 1978.03 | 1067.74 | 1770.20 | 389.96 |
| | $G(10\times10)$ | 100 | *** | 99.38 | *** | 78.30 | *** | *** | 2217.19 | 2404.34 | 900.66 | *** | *** | 2132.19 | 2854.77 | 870.75 |
| | $G(15\times15)$ | 25 | *** | 29.19 | *** | 20.43 | *** | 430.59 | 214.16 | 304.70 | 100.35 | *** | 542.03 | 274.36 | 356.63 | 99.71 |
| | $G(15\times15)$ | 50 | *** | 71.34 | *** | 28.93 | *** | *** | *** | 1819.47 | 525.24 | *** | *** | *** | 2215.93 | 583.79 |
| | $G(15\times15)$ | 100 | *** | 147.71 | *** | 135.58 | *** | *** | *** | 4104.47 | 1136.02 | *** | *** | *** | 3710.65 | 1397.90 |
| | $R(5\times5)$ | 100 | 32.53 | 6.94 | 31.79 | 3.80 | 1120.56 | 173.04 | 119.53 | 166.68 | 52.02 | 1210.25 | 225.86 | 136.07 | 154.87 | 48.25 |
| | $R(10\times10)$ | 25 | 83.21 | 13.64 | 64.03 | 12.24 | 3077.16 | 249.37 | 125.54 | 180.15 | 75.07 | 3462.87 | 263.68 | 129.87 | 154.24 | 94.95 |
| | $R(10\times10)$ | 50 | *** | 45.31 | *** | 28.22 | *** | 1957.23 | 1094.72 | 1147.77 | 417.82 | *** | 2145.04 | 1085.43 | 1331.45 | 456.09 |
| | $R(10\times10)$ | 100 | *** | 116.86 | *** | 93.91 | *** | *** | 1579.95 | 2784.09 | 822.13 | *** | *** | 1713.58 | 2665.49 | 669.99 |
| | $R(15\times15)$ | 25 | *** | 25.38 | *** | 13.36 | *** | 422.78 | 191.89 | 267.59 | 81.61 | *** | 435.98 | 185.66 | 278.40 | 95.08 |
| | $R(15\times15)$ | 50 | *** | 67.04 | *** | 77.15 | *** | *** | *** | 1735.73 | 616.41 | *** | *** | *** | 1845.61 | 623.39 |
| | $R(15\times15)$ | 100 | *** | 163.47 | *** | 138.77 | *** | *** | *** | 3615.20 | 896.01 | *** | *** | *** | 3822.38 | 913.98 |
| $U(0, 1)$ | $G(5\times5)$ | 100 | 24.00 | 5.95 | 29.22 | 3.85 | 1387.71 | 237.10 | 178.11 | 211.97 | 68.47 | 1374.70 | 260.85 | 167.14 | 230.64 | 65.20 |
| | $G(10\times10)$ | 25 | 87.41 | 14.76 | 72.45 | 12.54 | 5773.54 | 437.68 | 290.58 | 303.68 | 104.22 | 7190.76 | 464.54 | 254.20 | 286.27 | 81.49 |
| | $G(10\times10)$ | 50 | *** | 33.83 | *** | 23.06 | *** | 2036.47 | 1208.23 | 1334.54 | 440.85 | *** | 1907.62 | 1038.68 | 1705.58 | 350.35 |
| | $G(10\times10)$ | 100 | *** | 107.47 | *** | 71.14 | *** | *** | 2033.88 | 2408.62 | 988.70 | *** | *** | 2198.97 | 2783.91 | 770.08 |
| | $G(15\times15)$ | 25 | *** | 29.76 | *** | 17.63 | *** | 462.49 | 204.33 | 297.90 | 99.02 | *** | 537.46 | 236.89 | 375.10 | 101.54 |
| | $G(15\times15)$ | 50 | *** | 68.89 | *** | 28.76 | *** | *** | *** | 1696.07 | 541.06 | *** | *** | *** | 2165.10 | 585.89 |
| | $G(15\times15)$ | 100 | *** | 155.96 | *** | 129.78 | *** | *** | *** | 4494.28 | 1114.35 | *** | *** | *** | 3697.79 | 1268.11 |
| | $R(5\times5)$ | 100 | 35.58 | 6.95 | 36.01 | 3.46 | 985.74 | 184.76 | 121.97 | 163.12 | 51.95 | 1048.04 | 248.61 | 134.81 | 179.42 | 41.69 |
| | $R(10\times10)$ | 25 | 95.52 | 12.66 | 69.28 | 12.18 | 3044.39 | 267.35 | 122.53 | 172.08 | 84.52 | 3551.31 | 265.64 | 117.45 | 150.67 | 108.80 |
| | $R(10\times10)$ | 50 | *** | 49.40 | *** | 25.50 | *** | 1718.12 | 1101.08 | 1157.62 | 371.76 | *** | 2354.54 | 1020.73 | 1192.11 | 462.15 |
| | $R(10\times10)$ | 100 | *** | 117.74 | *** | 104.74 | *** | *** | 1449.64 | 2802.91 | 856.10 | *** | *** | 1622.34 | 2898.14 | 657.88 |
| | $R(15\times15)$ | 25 | *** | 26.77 | *** | 15.03 | *** | 454.08 | 203.30 | 252.75 | 79.89 | *** | 421.44 | 176.62 | 303.16 | 78.21 |
| | $R(15\times15)$ | 50 | *** | 64.28 | *** | 76.30 | *** | *** | *** | 1932.85 | 640.71 | *** | *** | *** | 1656.13 | 699.35 |
| | $R(15\times15)$ | 100 | *** | 190.02 | *** | 137.42 | *** | *** | *** | 3062.59 | 1014.86 | *** | *** | *** | 3617.74 | 917.88 |
| $U(0.25, 0.75)$ | $G(5\times5)$ | 100 | 24.46 | 6.13 | 25.44 | 3.77 | 1545.64 | 229.58 | 169.12 | 182.22 | 72.04 | 1500.60 | 278.45 | 159.91 | 232.78 | 59.63 |
| | $G(10\times10)$ | 25 | 85.33 | 14.76 | 75.71 | 11.31 | 5901.14 | 409.24 | 247.70 | 299.01 | 101.57 | 7514.09 | 441.79 | 257.08 | 312.39 | 73.73 |
| | $G(10\times10)$ | 50 | *** | 35.73 | *** | 25.06 | *** | 1829.71 | 1052.16 | 1392.42 | 501.18 | *** | 2045.46 | 974.34 | 1887.58 | 364.99 |
| | $G(10\times10)$ | 100 | *** | 118.05 | *** | 65.75 | *** | *** | 2033.24 | 2416.52 | 1007.44 | *** | *** | 1907.02 | 2475.25 | 747.13 |
| | $G(15\times15)$ | 25 | *** | 29.57 | *** | 15.95 | *** | 475.63 | 212.53 | 283.93 | 109.00 | *** | 539.11 | 254.80 | 383.75 | 83.73 |
| | $G(15\times15)$ | 50 | *** | 76.33 | *** | 28.04 | *** | *** | *** | 1523.40 | 559.23 | *** | *** | *** | 2172.45 | 520.58 |
| | $G(15\times15)$ | 100 | *** | 177.58 | *** | 119.87 | *** | *** | *** | 4419.80 | 1212.60 | *** | *** | *** | 3419.55 | 1206.58 |
| | $R(5\times5)$ | 100 | 37.02 | 6.64 | 41.52 | 3.93 | 1123.76 | 180.33 | 116.59 | 168.19 | 53.31 | 1070.69 | 272.51 | 139.68 | 194.41 | 40.43 |
| | $R(10\times10)$ | 25 | 95.77 | 11.86 | 66.60 | 11.65 | 2996.91 | 260.83 | 140.53 | 174.32 | 85.19 | 3682.58 | 228.08 | 111.39 | 138.12 | 107.05 |
| | $R(10\times10)$ | 50 | *** | 49.83 | *** | 28.62 | *** | 1633.16 | 1065.56 | 1093.98 | 400.69 | *** | 2359.42 | 1142.92 | 1088.27 | 469.28 |
| | $R(10\times10)$ | 100 | *** | 121.26 | *** | 105.65 | *** | *** | 1463.05 | 2916.12 | 805.22 | *** | *** | 1466.72 | 3371.43 | 661.93 |
| | $R(15\times15)$ | 25 | *** | 29.60 | *** | 16.14 | *** | 454.73 | 206.06 | 247.07 | 70.27 | *** | 422.84 | 148.12 | 303.27 | 69.78 |
| | $R(15\times15)$ | 50 | *** | 66.85 | *** | 76.51 | *** | *** | *** | 1958.45 | 628.68 | *** | *** | *** | 1799.60 | 670.48 |
| | $R(15\times15)$ | 100 | *** | 199.66 | *** | 121.94 | *** | *** | *** | 2980.52 | 1056.07 | *** | *** | *** | 3940.47 | 915.90 |