1 Appendix

The small example is intended to be solved manually to aid readers in gaining a better understanding of both the proposed model and the three algorithms. Additionally, we provide detailed explanations of each step of the algorithmic process and their corresponding results, aiming to facilitate reader comprehension. As explained in **Remark 1**, the proposed model can be extended to other problems, such as protecting road links and restoring road links, among others. In the small example, we applied the proposed model and algorithm to address the protection of road links. The model for the protection link problem is formulated as follows.

$$\textbf{First stage:}\quad \max_{\mathbf{y}} f(\mathbf{y}) = \mathbb{E}_{\mathbf{T}}[Q(\mathbf{y},t)] = \sum_{t\in\mathbf{T}} P(t)Q(\mathbf{y},t) \tag{1}$$

$$= \sum_{t\in\mathbf{T}} P(t)\left(\sum_{od\in OD(t)} Q^{od}(\mathbf{y},t)\right) \tag{2}$$

$$\textbf{s.t.} \sum_{(i,j)\in\bar{A}} c_{ij}y_{ij} \le b \tag{3}$$

$$y_{ij}\in\{0,1\}\quad \forall(i,j)\in\bar{A};\quad y_{ij}=1\quad \forall(i,j)\in A\setminus\bar{A} \tag{4}$$

$$\textbf{Second stage:}\quad Q^{od}(\mathbf{y},t) = \max w_{od}h^{od}(t)z_{od}(t) \tag{5}$$

$$\textbf{s.t.} -M\left(1-z_{od}\right)+\varepsilon \le n_{od}-\pi_{od} \le Mz_{od} \tag{6}$$

$$\frac{1}{n_{od}(t)}\left(\sum_{(i,j)\in A\cup\bar{A}} \big(r^0_{ij}+(1-y_{ij})r_{ij}(t)\big)x^*_{ij}(t)\right) \le \alpha^{od}\tilde{r}^{od}(t) \tag{7}$$

$$\mathbf{x}^*(t) = \arg\min \sum_{(i,j)\in A\cup\bar{A}} \big(r^0_{ij}+(1-y_{ij})r_{ij}(t)\big)x_{ij}(t) \tag{8}$$

$$\textbf{s.t.} \sum_{(o,j)\in\Gamma(o)} x_{oj}(t) = \sum_{(i,d)\in\Gamma^{-1}(d)} x_{id}(t) = n_{od}(t) \tag{9}$$

$$\sum_{(i,o)\in\Gamma^{-1}(o)} x_{io}(t) = \sum_{(d,j)\in\Gamma(d)} x_{dj}(t) = 0 \tag{10}$$

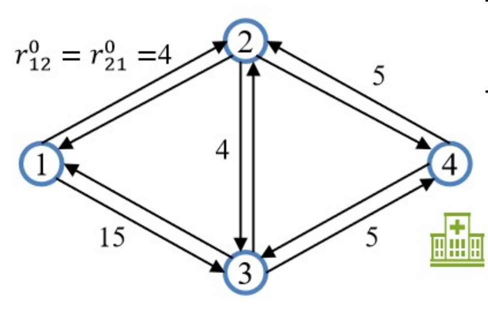
$$\sum_{(i,j)\in\Gamma(i)} x_{ij}(t) - \sum_{(h,i)\in\Gamma^{-1}(i)} x_{hi}(t) = 0\quad \forall i\in N, i\ne o,d \tag{11}$$

$$z_{od}(t)\in\{0,1\}; n_{od}(t)\in\mathbb{Z}; x_{ij}(t)\in\{0,1\}\quad \forall(i,j)\in A\cup\bar{A} \tag{12}$$

where r^0_{ij} denotes the free-flow traffic times for each link, while $r_{ij}(t)$ represents the incremental travel times for each link under a given disaster size and time slot. M is a sufficiently large positive constant, and ε is a sufficiently small positive constant. Other parameters maintain the same definitions as in the original model. The model remains solvable using the three proposed solution methods.

Next, let's introduce this example. Figure 1 illustrates a small network, employed to explain the proposed model and the three algorithms. In this illustrative case study, we consider a single disaster size and three stochastic time slots, each with an equal probability of occurrence (0.33). Time_1, Time_2, and Time_3 represent the three time slots. During Time_1, the only OD pair is (1,4); during Time_2, the sole OD pair is (2,4); and during Time_3, the exclusive OD pair is (3,4). To provide a real-world context, consider that Time_1 corresponds to resting time, such as late at night, with node 1 representing a residential area. Time_2 aligns with working hours, such as daytime, where node 2 represents a working area. Lastly, Time_3 signifies leisure time, such as evening, with node 3 representing a recreational area. Node 4 serves as an urban emergency facility, such as a hospital. Consequently, the OD pairs vary during different time slots.

Figure 1 displays the free-flow traffic times r^0_{ij} for each link indicated above their respective positions on the network. Additionally, Figure 1 illustrates the incremental travel times $r_{ij}(t)$ for each link under the disaster size and each time slot. The total available funding b is limited to 2 million dollars, with a cost of 1 million dollars incurred for protecting each link c_{ij} on the network. Given this budget constraint, traffic



Link (i,j)	Time_1 $r_{ij}(t_1)$	Time_2 $r_{ij}(t_2)$	Time_3 $r_{ij}(t_3)$
(1,2); (2,1)	3	3	3
(1,3); (3,1)	1	2	3
(2,3); (3,2)	1	2	3
(2,4); (4,2)	0	0	0
(3,4); (4,3)	1	2	3

Figure 1: A small example.

managers expect that the minimum number of DPATT for each OD pair, is as follows: $\pi_{14} = 1$; $\pi_{24} = 0$; and $\pi_{34} = 1$. For simplicity and without sacrificing generality, the weight w_{od} and travel demand $h^{od}(t)$ are consistently set to 1 across all scenarios. The acceptable level, α^{od}, is fixed at 1.7. The shortest travel times, $\tilde{r}^{od}(t)$, are specified as follows: $\tilde{r}^{14}(t_1) = 9$; $\tilde{r}^{24}(t_2) = 5$; and $\tilde{r}^{34}(t_3) = 5$. Consequently, with these simplified parameters, this illustrative example can be manually solved, facilitating a more comprehensive understanding of the solution methodologies employed by the three algorithms.

Deterministic equivalent program

The proposed model, as discussed in Section 4.3, can be reformulated as an integer programming, solvable with tools like Gurobi. With the given parameters, the optimal solution involves protecting two links, (1,2) and (3,4), resulting in an optimal objective value of 0.99. In this solution, all the OD pairs have two DPATT.

Algorithm 1 and Algorithm 2

Building upon the analysis provided in Section 4.4 and 4.5, we can employ the integer L-shaped algorithm with KKT conditions and the integer L-shaped algorithm with minimum-cost maximum-flow method to solve the proposed model. To provide insight into the workings of this algorithm, we will illustrate its first iteration in solving the proposed model. In this particular iteration, we set LB to be equal to -1. This decision is grounded on the observation that in the given example, protecting all road links results in every OD pair having two DPATT. Consequently, the z_{od} values for all OD pairs become equal to 1, resulting in the value of the first-stage objective function being equal to 1. It's worth noting that since the proposed model is a maximization problem, we set L to -1 accordingly.

Step 1: Initialize UB $\leftarrow +\infty$, $\theta \leftarrow -\infty$, and $\mathbf{y} = 0$ as the root node.

Step 2: Choose the pendant node.

Step 3: As this is the first iteration, skip the step.

Step 4: Go to step 5.

Step 5: Solve the subproblem by **KKT conditions with relaxation**. Obtain $G_{LP}(\mathbf{y}) = -0.98617$. Generate the corresponding continuous optimality cut: $-0.00148y_{12} - 0.00024y_{13} - 0.00040y_{32} - 0.00307y_{34} - 0.98616 <= \theta$. Return to step 3.

Step 3: Solve the master problem. The solution is as follows: **y:** $y_{12} = 1, y_{34} = 1$, and all other y values are 0. Additionally, $\theta = -0.99071$.

Step 4: Go to step 5.

Step 5: Solve the subproblem by **KKT conditions with relaxation**. Obtain $G_{LP}(\mathbf{y}) = -0.99$. Generate the corresponding continuous optimality cut: $-0.99 <= \theta$. Return to step 3.

Step 3: Upon solving the master problem, the outcome is as follows: **y:** $y_{12} = 1, y_{34} = 1$, and all other y values are 0. Additionally, $\theta = -0.99$.

Step 4: Go to step 5.

Step 5: Solve the subproblem by **KKT conditions with relaxation**. Obtain $G_{LP}(\mathbf{y}) = -0.99$. Based on $G_{LP}(\mathbf{y}) = \theta$, Go to step 6.

Step 6: Solve the subproblem by **KKT conditions without relaxation or by Algorithm ??**. Obtain $G(\mathbf{y}) = -0.99$, and update UB $= -0.99$. Based on $G(\mathbf{y}) = \theta$, fathom this node and return to step 2.

Step 2: Stop. We obtain the optimal value of -0.99. It's worth noting that since the proposed model is a maximization problem, the final optimal value is 0.99.