Submitted to *INFORMS Journal on Computing*
Manuscript JOC-2023-08-OA-0286

Online Supplements for "An Exact Algorithm for Multicommodity Network Design under Stochastic Interdictions"

Shabnam Mahmoudzadeh Vaziri[a,c], Onur Kuzgunkaya[a], Navneet Vidyarthi[b,c]
[a]Gina Cody School of Engineering and Computer Science, Concordia University, Montreal, Canada
[b]John Molson School of Business, Concordia University, Montreal, Canada
[c]Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT), Montreal, Canada

Appendix A: Proofs of the lemma and propositions

In this appendix, we first prove the equivalence of MCF_s and $PMCF_s$ formulations.

LEMMA 1. *Let M be a large number and $\overline{v}(\hat{x})$ be an optimal solution of $PMCF_s$ for any $\hat{x} \in X$. Then we have $\overline{v}(\hat{x})^k_{hs} = 0, \forall k \in K$ if $\hat{x}_h = 1$ for any $\hat{x} \in X$.*

Proof. Since M is a large number if $\hat{x} = 1$ and $v(\hat{x}) > 0$, a strictly better solution exists in which $v(\hat{x}) = 0$, achieved by reallocating flows from paths containing arc h to the dummy arcs in the network, resulting in lower costs. Therefore, if $\hat{x} = 1$, $v(\hat{x})$ should be zero. This completes the proof. □

PROPOSITION 1. *Given any $\hat{x} \in X$, let $MCF_s(\hat{x})$ and $PMCF_s(\hat{x})$ denote the designer's solutions to MCF_s and $PMCF_s$, respectively. Then $MCF_s(\hat{x})$ has an optimal solution $v(\hat{x})$ if and only if $v(\hat{x})$ is optimal to $PMCF_s(\hat{x})$. Furthermore, the optimal objective function value of $MCF_s(\hat{x})$ is equal to the objective function value of $PMCF_s(\hat{x})$.*

Proof. We first show that the objective function value of $PMCF_s$ remains less than or equal to the objective function value of MCF_s. Consider an optimal solution $v(\hat{x})$ to MCF_s with an objective function value of v^*. Since $PMCF_s(\hat{x})$ is a relaxation of $MCF_s(\hat{x})$, $v(\hat{x})$ is also feasible for $PMCF_s(\hat{x})$. Moreover, we have $\sum_{h\in A}\sum_{k\in K}(r^k_h + M\hat{x}_{hs})v^k_{hs} = v^*$ because $\sum_{h\in A}\sum_{k\in K} r^k_h v^k_{hs} = v^*$ and Lemma 1 ensures that $\sum_{h\in A}\sum_{k\in K} M\hat{x}_{hs}v^k_{hs} = 0$. Therefore, the optimal objective function value of $PMCF_s(\hat{x})$ does not exceed v^*.

Furthermore, we show that the optimal objective function value of $MCF_s(\hat{x})$ does not exceed the optimal objective function value of $PMCF_s(\hat{x})$, leading to the equivalence of these two values. Let $v(\hat{x})$ be an optimal solution to $PMCF_s$ with an objective function value of v^*. Since Lemma 1 guarantees that $v^k_{hs}(\hat{x}) = 0$ for $\hat{x}_{hs} = 1$, $v(\hat{x})$ is a feasible solution for $MCF_s(\hat{x})$ with the objective function value $\sum_{h\in A}\sum_{k\in K} r^k_h v^k_{hs} = v^*$, which provides a lower bound on the optimal objective function value of $MCF_s(\hat{x})$. This completes the proof. □

Next, we prove the validity of the proposed supervalid inequality.

PROPOSITION 2. *Consider the Benders cut $\eta_s + \sum_{h\in A} q_h w_h(\hat{\alpha}_{hs} - \hat{\varphi}_{hs}) \geq \sum_{k\in K} d^k(\hat{\pi}^k_{O_k s} - \hat{\pi}^k_{D_k s})$ is added to the MP. The following inequality is supervalid inequality:*

$$\sum_{h\in A} I(\hat{\alpha}_{hs} - \hat{\varphi}_{hs})w_h \geq 1, \qquad \textit{where} \;\; I(\hat{\alpha}_{hs} - \hat{\varphi}_{hs}) = \begin{cases} 1, & \textit{if } \hat{\alpha}_{hs} - \hat{\varphi}_{hs} > 0 \\ 0, & \textit{otherwise} \end{cases}$$

Proof. We prove this by contradiction. It demonstrates that if we assume that the incumbent solution is not optimal, it leads to a contradiction, thus confirming the necessity of the inequality being at least 1 in an optimal solution. This establishes the supervalidity of the stated inequality in the given context of the Benders cut. The following steps prove the Proposition:

The statement supposes that a feasible solution $(\hat{w}, \hat{\pi}, \hat{\alpha}, \hat{\varphi})$ generates the Benders cut $\eta_s + \sum_{h\in A} q_h w_h(\hat{\alpha}_{hs} - \hat{\varphi}_{hs}) \geq \sum_{k\in K} d^k(\hat{\pi}^k_{O_k s} - \hat{\pi}^k_{D_k s})$. Let $\hat{\eta}_s = \sum_{k\in K} d^k(\hat{\pi}^k_{O_k s} - \hat{\pi}^k_{D_k s}) - \sum_{h\in A} q_h \hat{w}_h(\hat{\alpha}_{hs} - \hat{\varphi}_{hs})$ and let (w^*, y^*, η^*) denote the optimal solution to the MP. Note that $\sum_{h\in A} I(\hat{\alpha}_{hs} - \hat{\varphi}_{hs})w^*_h = 0$ or ≥ 1 holds. $\sum_{h\in A} I(\hat{\alpha}_{hs} - \hat{\varphi}_{hs})w^*_h = 0$ implies that (i) $w^*_h = 0$ or (ii) $I(\hat{\alpha}_{hs} - \hat{\varphi}_{hs}) = 0$ hold for all $h \in A$. In the latter case, based on the definition of $I(\hat{\alpha}_{hs} - \hat{\varphi}_{hs})$, $\hat{\alpha}_{hs} - \hat{\varphi}_{hs} \leq 0$ holds. Since the maximum value of $\hat{\varphi}_{hs}$ is $\hat{\alpha}_{hs}$, $\hat{\alpha}_{hs} - \hat{\varphi}_{hs}$ would be zero, implying that $\sum_{h\in A} q_h w^*_h(\hat{\alpha}_{hs} - \hat{\varphi}_{hs}) = 0$. In general, we can thus conclude from $\sum_{h\in A} I(\hat{\alpha}_{hs} - \hat{\varphi}_{hs})w^*_h = 0$ that $\sum_{h\in A} q_h w^*_h(\hat{\alpha}_{hs} - \hat{\varphi}_{hs})w_h = 0$.
When $\sum_{h\in A} I(\hat{\alpha}_{hs} - \hat{\varphi}_{hs})w^*_h = 0$,
$\eta^*_s \geq \sum_{k\in K} d^k(\hat{\pi}^k_{O_k s} - \hat{\pi}^k_{D_k s}) - \sum_{h\in A} q_h w^*_h(\hat{\alpha}_{hs} - \hat{\varphi}_{hs})$ is true for any $(\hat{\pi}, \hat{\alpha}, \hat{\varphi})$,
$= \hat{\eta}_s - \sum_{h\in A} q_h w^*_h(\hat{\alpha}_{hs} - \hat{\varphi}_{hs}) + \sum_{h\in A} q_h \hat{w}_h(\hat{\alpha}_{hs} - \hat{\varphi}_{hs})$
$\geq \hat{\eta}_s$ because $\sum_{h\in A} q_h w^*_h(\hat{\alpha}_{hs} - \hat{\varphi}_{hs}) = 0$
$\geq \overline{\eta}_s$ because $\hat{w}$ does not need to be the incumbent solution and
$> \eta^*_s$ by assumption; but this is a contradiction.

Therefore, if the incumbent solution is not optimal, $\sum_{h\in A} I(\hat{\alpha}_{hs} - \hat{\varphi}_{hs})w^*_h \geq 1$ must be true for every optimal solution (w^*, y^*, η^*). Hence, the inequality $\sum_{h\in A} I(\hat{\alpha}_{hs} - \hat{\varphi}_{hs})w_h \geq 1$ is supervalid. □

Appendix B: Smith et al. (2007)'s Algorithm

In this appendix, we present the pseudo-code of the algorithm presented by Smith et al. (2007)'s Algorithm. We define UB as an upper bound on the optimal objective value obtained by solving the scenario SPs, LB as a lower bound on the optimal objective value obtained by solving the MP, and t as the current iteration. Let Υ^t be the set of extreme points generated up to iteration t, and ϵ be the desirable optimality gap. The algorithm is detailed as follows:

Algorithm 1 Smith et al. (2007)'s Algorithm's

$UB \leftarrow \infty$, $LB \leftarrow -\infty$, $t \leftarrow 0$, $\Upsilon^t \leftarrow \varnothing$

While $(\frac{UB - LB}{UB} > \epsilon)$ **do**

  Solve the MP to obtain w

  Update LB

  For $s \in S$, Solve the SP_s for a given $\hat{w}$ and obtain dual variables (π, α, φ)

  End for

  Update UB

  $\Upsilon^{t+1} \leftarrow \Upsilon^t \cup \{\pi, \alpha, \varphi\}$

  $t \leftarrow t + 1$

End while

Appendix C: Test Instances

In this Appendix, we present the details of publicly available Set "R" instances in Table S1 (Frangioni, 2024), used in several papers on the MCNDP and described in detail by Crianic et al. (2001). There are 18 sets of nine problem instances, denoted by "R01" to "R18" where each instance has different arc capacities and fixed design cost.

Table S1 Details of Test Instances (Set R01 to R18)

Set	Node ($\lvert N\rvert$)	Arc ($\lvert A\rvert$)	Commodity ($\lvert K\rvert$)	No. of scenarios	Interdiction budget
R01	10	35	10	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R02	10	35	25	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R03	10	35	50	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R04	10	60	10	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R05	10	60	25	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R06	10	60	50	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R07	10	82	10	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R08	10	83	25	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R09	10	83	50	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R10	20	120	40	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R11	20	120	100	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R12	20	120	200	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R13	20	220	40	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R14	20	220	100	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R15	20	220	200	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R16	20	314	40	2	1, 2
				3	2, 3, 4
				4	1, 2, 3, 4
R17	20	318	100	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4
R18	20	315	200	2	1, 2
				3	1, 2, 3
				4	1, 2, 3, 4

Appendix D: Detailed Computational Results

The detailed results of Set R01 to R09 using the five variants of the BBC algorithm are presented in Table S2. We also present the detailed results of Set R10 to R18 for the three best variants of the BBC algorithm in Table S3. The comparison of the performance of the MibS Solver and BBC2 algorithm is presented in Table S4. The results of the performance of the BBC2 algorithm for the model with uncertain demand and interdiction budget are presented in Table S5.

Table S2 Performance of Variants of BBC Algorithm on Set R01-R09 Instances

Ins.	Set	$\lvert N\rvert$	$\lvert A\rvert$	$\lvert K\rvert$	CPU Time (s) for $\lvert S\rvert=2$					CPU Time (s) for $\lvert S\rvert=3$					CPU Time (s) for $\lvert S\rvert=4$				
					BBC1	BBC2	BBC3	BBC4	BBC5	BBC1	BBC2	BBC3	BBC4	BBC5	BBC1	BBC2	BBC3	BBC4	BBC5
1	R01	10	35	10	17	5	14	17	5	21	6	20	23	8	40	8	46	44	12
2		10	35	10	21	7	19	21	6	29	9	26	30	8	40	11	32	45	12
3		10	35	10	21	8	23	21	7	38	9	27	40	8	58	12	50	62	14
4		10	35	10	14	5	17	14	7	23	6	23	24	8	42	9	39	46	9
5		10	35	10	16	7	17	17	7	28	9	26	28	7	47	13	50	57	9
6		10	35	10	16	6	21	17	6	28	8	27	29	9	40	9	44	44	11
7		10	35	10	3	1	3	3	1	4	1	5	5	1	7	1	8	8	1
8		10	35	10	5	1	5	6	1	10	2	13	10	3	15	3	13	17	3
9		10	35	10	8	2	7	8	2	12	3	12	13	3	20	4	22	23	4
10	R02	10	35	25	21	5	19	20	5	28	7	41	29	7	36	9	44	40	9
11		10	35	25	51	14	57	40	19	47	15	71	50	17	82	18	111	87	15
12		10	35	25	60	17	50	64	17	88	29	91	86	26	100	29	144	148	28
13		10	35	25	26	4	20	25	5	37	8	31	36	7	52	10	49	54	11
14		10	35	25	23	7	29	25	7	58	9	54	56	12	90	15	75	65	15
15		10	35	25	29	5	25	28	6	55	12	56	45	13	84	12	73	87	15
16		10	35	25	18	7	17	17	7	29	7	22	28	7	31	8	27	35	8
17		10	35	25	8	2	8	9	3	13	3	12	15	4	11	4	15	14	5
18		10	35	25	18	8	23	17	6	34	11	34	37	14	47	14	46	48	13
19	R03	10	35	50	110	31	100	110	28	164	34	135	160	39	181	41	151	253	48
20		10	35	50	122	30	140	118	33	230	61	216	170	50	220	49	197	233	56
21		10	35	50	146	29	119	125	36	235	52	186	222	45	407	70	291	314	65
22		10	35	50	39	9	32	42	11	55	13	49	56	18	71	15	57	79	21
23		10	35	50	89	22	69	86	16	99	25	94	97	27	104	26	100	107	28
24		10	35	50	89	21	69	85	16	99	25	94	97	27	104	27	100	107	28
25		10	35	50	25	14	26	27	14	32	16	36	33	16	32	17	38	37	19
26		10	35	50	24	12	25	26	15	33	19	32	33	17	33	16	36	38	20
27		10	35	50	24	12	25	25	14	32	15	34	33	17	32	18	35	36	18
28	R04	10	60	10	109	39	72	135	41	642	102	224	780	118	619	151	351	782	169
29		10	60	10	105	32	75	107	43	461	129	253	518	100	558	149	385	632	228
30		10	60	10	113	36	60	107	52	268	115	234	300	96	573	161	278	573	148
31		10	60	10	298	52	112	260	77	287	52	176	274	68	310	95	200	365	91
32		10	60	10	150	60	115	142	65	271	70	201	252	89	439	117	364	500	124
33		10	60	10	143	60	96	139	70	185	85	180	171	82	290	77	347	304	94
34		10	60	10	29	5	17	35	7	68	12	38	64	7	63	14	49	59	11
35		10	60	10	17	5	14	17	5	31	8	28	27	7	46	11	51	44	9
36		10	60	10	19	6	19	22	5	43	8	32	35	8	63	10	52	65	11
37	R05	10	60	25	1,353	581	883	1,355	596	10,594	3,572	8,702	10,428	4,528	6,555	3,429	4,983	7,588	3,033
38		10	60	25	1,095	543	966	1,153	583	6,958	2,872	5,577	6,191	3,419	5,881	2,454	4,468	6,420	2,779
39		10	60	25	1,191	542	954	1,076	577	6,327	3,351	5,351	6,883	3,529	5,218	2,316	5,974	8,437	3,424
40		10	60	25	2,087	664	1,374	1,836	798	1,693	602	1,352	1,360	810	1,776	796	1,014	1,410	842
41		10	60	25	1,265	553	1,105	1,106	743	1,929	610	1,300	1,815	911	1,855	793	1,843	1,957	1,160
42		10	60	25	1,599	541	1,311	1,625	736	1,910	874	1,647	1,903	876	2,698	751	1,957	2,815	1,209
43		10	60	25	51	15	56	51	13	84	22	77	80	19	98	24	83	107	20
44		10	60	25	71	22	78	71	18	96	30	104	96	21	111	31	100	126	24
45		10	60	25	88	23	75	97	24	124	31	121	114	25	120	31	123	147	31
46	R06	10	60	50	1,339	312	748	1,149	337	4,054	904	2,720	4,372	1,033	4,215	1,542	2,128	4,338	1,737
47		10	60	50	1,299	499	800	1,226	381	4,128	1,459	3,681	4,930	1,604	8,410	3,239	4,079	7,545	3,537
48		10	60	50	1,067	584	1,046	1,152	342	28,691	5,085	10,608	28,060	7,573	49,501	18,956	36,189	39,608	30,555
49		10	60	50	2,342	446	1,489	2,028	563	3,511	1,007	1,891	3,454	987	5,075	1,490	2,953	6,230	1,598
50		10	60	50	3,515	1,532	2,797	3,622	1,746	14,871	5,061	7,050	13,836	6,446	13,825	6,601	11,931	16,541	7,441
51		10	60	50	4,477	1,235	3,479	5,311	1,904	16,371	4,917	10,630	18,755	7,309	42,988	16,932	28,730	41,414	26,956
52		10	60	50	57	16	45	61	15	89	48	81	97	34	110	52	86	122	40
53		10	60	50	131	39	99	126	35	174	70	168	182	59	209	59	170	241	63
54		10	60	50	1,713	693	978	1,734	783	3,394	1,408	3,011	3,116	1,741	5,953	2,495	3,554	4,826	2,403

Table S2 Performance of Variants of BBC Algorithm on Set R01-R09 Instances (continued)

Ins.	Set	$\|N\|$	$\|A\|$	$\|K\|$	CPU Time (s) for $\|S\|=2$					CPU Time (s) for $\|S\|=3$					CPU Time (s) for $\|S\|=4$				
					BBC1	BBC2	BBC3	BBC4	BBC5	BBC1	BBC2	BBC3	BBC4	BBC5	BBC1	BBC2	BBC3	BBC4	BBC5
55	R07	10	82	10	110	28	43	108	26	1,043	368	435	1,145	289	5,124	1,899	3,624	5,797	2,574
56		10	82	10	81	22	66	90	34	765	216	460	729	289	4,551	2,090	3,266	4,944	1,920
57		10	82	10	107	27	70	135	39	640	199	426	638	228	5,774	1,592	3,137	5,943	2,229
58		10	82	10	231	63	111	242	59	1,995	689	1,341	2,481	734	3,358	1,070	3,151	4,059	1,388
59		10	82	10	140	36	123	147	59	1,014	332	604	1,058	460	4,069	1,129	2,425	4,209	1,493
60		10	82	10	95	27	76	96	38	1,180	356	579	1,292	402	2,874	1,335	2,062	2,771	1,474
61		10	82	10	248	50	178	329	65	333	97	177	420	107	279	73	165	331	70
62		10	82	10	169	48	125	129	57	295	71	213	289	89	335	102	246	274	98
63		10	82	10	132	40	109	154	54	298	92	212	218	97	380	88	281	442	115
64	R08	10	83	25	776	181	493	786	252	21,775	5,506	8,815	23,685	6,028	83,289	36,798	65,651	80,367	50,005
65		10	83	25	808	409	541	780	341	8,788	3,387	7,019	10,818	4,152	76,940	21,922	64,290	64,934	23,498
66		10	83	25	518	210	339	571	263	4,552	2,365	5,453	5,390	3,044	71,651	20,428	46,083	60,988	21,950
67		10	83	25	2,496	664	1,476	2,821	1,059	31,663	10,130	16,178	31,072	9,742	72,846	49,036	59,412	71,716	68,955
68		10	83	25	1,909	701	1,085	1,921	721	12,246	5,314	11,895	13,821	6,410	78,099	45,106	70,694	68,040	47,403
69		10	83	25	1,227	348	715	1,190	537	11,250	5,620	8,037	11,230	6,492	76,033	14,511	48,714	49,121	28,251
70		10	83	25	2,906	1,159	2,027	3,172	1,121	3,554	1,093	2,680	3,453	1,752	3,978	1,769	3,985	4,951	2,120
71		10	83	25	1,432	593	1,275	1,505	538	2,184	1,124	2,718	2,678	1,021	4,814	1,752	4,634	5,471	1,901
72		10	83	25	1,519	829	1,445	1,458	893	3,195	1,391	3,048	2,701	1,372	4,162	1,651	3,839	3,630	2,064
73	R09	10	83	50	15,014	6,582	9,678	16,578	6,969	79,527	33,700	38,934	77,392	36,774	(7.4)*	42,937	58,795	(5.3)	54,674
74		10	83	50	8,002	3,614	5,782	7,624	3,923	59,176	18,500	40,634	60,106	14,745	85,793	35,880	62,793	83,753	41,470
75		10	83	50	4,982	2,403	3,691	4,895	3,740	27,189	12,639	29,868	26,267	15,297	72,165	42,978	52,165	69,164	40,241
76		10	83	50	25,595	10,616	11,931	22,529	10,957	74,382	45,797	56,467	73,277	56,478	(11.6)	70,164	80,521	(9.2)	76,664
77		10	83	50	20,354	13,423	20,382	22,460	17,669	64,073	33,703	54,281	70,759	35,999	(3.7)	62,608	71,598	(2.9)	61,949
78		10	83	50	13,660	10,714	11,417	13,960	10,177	33,708	16,066	28,605	30,283	24,483	(1.3)	55,606	64,351	86,357	55,739
79		10	83	50	606	189	520	604	195	819	208	620	767	249	1,013	264	880	1,136	303
80		10	83	50	746	246	664	713	313	1,509	269	763	1,135	292	1,067	415	1,170	1,302	506
81		10	83	50	1,503	440	1,188	1,293	562	1,883	597	1,459	1,927	826	2,520	727	2,165	2,971	770

* () indicates the gap at the time limit

Table S3 Performance of Variants of BBC Algorithm on Set R10 to R18 Instances

Ins.	Set	$\lvert N\rvert$	$\lvert A\rvert$	$\lvert K\rvert$	CPU Time (s) for $\lvert S\rvert=2$			CPU Time (s) for $\lvert S\rvert=3$			CPU Time (s) for $\lvert S\rvert=4$		
					BBC2	BBC3	BBC5	BBC2	BBC3	BBC5	BBC2	BBC3	BBC5
82	R10	20	120	40	16,358	21,935	16,997	41,430	73,387	59,250	81,474	(4.8)*	83,208
83		20	120	40	10,526	19,755	13,994	32,679	66,082	38,693	(11)	(15.8)	(11.1)
84		20	120	40	8,235	9,782	8,392	24,862	51,325	31,439	35,079	66,078	52,316
85		20	120	40	51,360	69,034	55,916	(1.1)	(4.6)	(1.1)	85,039	(17.4)	(10.5)
86		20	120	40	24,454	33,990	28,850	34,090	(0.8)	50,767	(9.1)	(20.3)	(19.2)
87		20	120	40	14,274	20,590	16,656	32,802	67,807	29,639	53,183	(9)	76,177
88		20	120	40	1,494	5,675	1,552	1,647	7,598	1,689	1,962	9,041	2,397
89		20	120	40	1,718	9,055	1,723	3,845	11,549	3,378	5,905	13,346	4,498
90		20	120	40	2,155	6,983	2,958	4,717	14,590	5,554	6,446	17,118	6,770
91	R11	20	120	100	25,138	(1.2)	34,932	(9.2)	(12.2)	(9.5)	(16.2)	(26.5)	(15.8)
92		20	120	100	11,834	21,411	17,038	18,757	43,432	18,138	63,109	73,041	66,462
93		20	120	100	8,779	14,065	9,736	17,423	59,353	24,563	54,008	82,929	67,976
94		20	120	100	38,504	57,015	39,438	34,178	(17.1)	58,690	62,532	(100)	79,739
95		20	120	100	54,953	(1.4)	70,313	(6.7)	(100)	(6.8)	(8.5)	(100)	(13.2)
96		20	120	100	59,696	79,218	58,872	(9.1)	(18.2)	(13.7)	(11.9)	(100)	(22.9)
97		20	120	100	4,229	5,048	3,696	5,224	8,452	4,669	4,396	12,796	5,044
98		20	120	100	5,983	5,715	4,770	4,889	11,314	4,813	6,581	15,454	6,351
99		20	120	100	3,559	5,530	4,166	10,232	19,047	13,184	12,986	21,144	15,992
100	R12	20	120	200	(13.3)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
101		20	120	200	(9.7)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
102		20	120	200	(17.4)	(30.5)	(25.8)	(31.2)	(100)	(39.2)	(100)	(100)	(100)
103		20	120	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
104		20	120	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
105		20	120	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
106		20	120	200	18,459	25,471	23,340	25,043	32,750	26,570	27,877	38,452	27,061
107		20	120	200	16,151	21,689	19,476	28,790	36,456	32,768	30,271	51,201	32,166
108		20	120	200	15,065	19,093	16,256	22,139	35,214	31,197	32,604	48,959	36,311
109	R13	20	220	40	23,257	26,391	25,174	(10.4)	(11.8)	(10.1)	(12.2)	(18.3)	(17.2)
110		20	220	40	26,565	31,405	33,794	60,421	(14.1)	(8.4)	75,341	(5.7)	79,368
111		20	220	40	29,617	33,325	37,754	82,657	86,262	83,048	(9.3)	(12.8)	(10.5)
112		20	220	40	58,635	60,821	56,756	(9.7)	(12.4)	(12.3)	(14.2)	(18.4)	(16.5)
113		20	220	40	36,554	51,869	43,935	62,710	75,193	65,010	78,052	83,219	80,058
114		20	220	40	38,261	48,449	46,167	67,447	67,081	65,258	80,359	(6.9)	83,920
115		20	220	40	52,815	72,022	69,920	(6.7)	(11.1)	(8.4)	(12.3)	(17.7)	(16.8)
116		20	220	40	48,210	(8.7)	(4.5)	(15)	(23.2)	(19.2)	(21.4)	(28.7)	(26.4)
117		20	220	40	45,426	(4.7)	71,231	69,236	(12.2)	(7.2)	(17.4)	(24.8)	(21.1)
118	R14	20	220	100	84,466	(9.8)	85,048	(4.4)	(9.8)	(5.2)	(7.3)	(13.8)	(9.5)
119		20	220	100	(3.9)	(14.7)	(3.6)	(10.7)	(17.8)	(11.6)	(14.3)	(27.8)	(17.7)
120		20	220	100	(5.2)	(11.1)	(9.6)	(100)	(100)	(100)	(100)	(100)	(100)
121		20	220	100	(10.9)	(26)	(14.7)	(33.9)	(100)	(100)	(38.4)	(100)	(100)
122		20	220	100	(21.5)	(100)	(100)	(21.3)	(100)	(27.4)	(27.3)	(100)	(32.6)
123		20	220	100	(6.3)	(16)	(8.1)	(9.4)	(22.6)	(13.5)	(17.2)	(32.7)	(21.2)
124		20	220	100	(100)	(100)	(100)	(25.2)	(36.2)	(31.3)	(100)	(100)	(100)
125		20	220	100	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
126		20	220	100	(1.7)	(9.6)	(1.5)	(9.7)	(20.7)	(16.3)	(15.8)	(38.3)	(24.2)
127	R15	20	220	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
128		20	220	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
129		20	220	200	(52.1)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
130		20	220	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
131		20	220	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
132		20	220	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
133		20	220	200	(32.2)	(100)	(48.2)	(17.3)	(100)	(100)	(100)	(100)	(100)
134		20	220	200	(20.7)	(33)	(26.1)	(30.4)	(100)	(38.5)	(100)	(100)	(100)
135		20	220	200	(24.1)	(38)	(31.4)	(34.4)	(100)	(39.4)	(100)	(100)	(100)
136	R16	20	314	40	32,129	41,964	38,287	53,525	66,086	63,643	65,286	83,205	77,614
137		20	314	40	36,011	53,860	51,528	56,705	74,485	70,239	73,897	(18.3)	(14.6)
138		20	314	40	46,403	58,601	55,054	60,059	(10.3)	(8.8)	(13.3)	(19.3)	(16.2)
139		20	314	40	61,930	66,893	65,106	(13.5)	(18.4)	(14.2)	(16.5)	(22.6)	(19.4)
140		20	314	40	48,703	59,621	47,479	61,055	(9.2)	64,037	72,637	(9.6)	78,356
141		20	314	40	77,614	82,896	79,040	(8.2)	(16.2)	(10.2)	(13.8)	(20.6)	(17.3)
142		20	314	40	67,043	80,677	63,647	(10.9)	(18)	(13.2)	(16.5)	(24.9)	(21.5)
143		20	314	40	73,190	86,317	72,483	(19.8)	(26.7)	(23.7)	(23.6)	(28.6)	(25.5)
144		20	314	40	59,666	(9.1)	(7.4)	(17.6)	(18.9)	(13)	(15.5)	(19.4)	(15.4)

* () indicates the gap at the time limit

Table S3 Performance of Variants of BBC Algorithm on Set R10 to R18 Instances (continued)

Ins.	Set	$\|N\|$	$\|A\|$	$\|K\|$	CPU Time(s) for $\|S\|=2$			CPU Time (s) for $\|S\|=3$			CPU Time (s) for $\|S\|=4$		
					BBC2	BBC3	BBC5	BBC2	BBC3	BBC5	BBC2	BBC3	BBC5
145	R17	20	318	100	85,679	(16.3)	(11.5)	(7.5)	(18.3)	(14.5)	(11.3)	(100)	(20.3)
146		20	318	100	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
147		20	318	100	(23)	(100)	(100)	(27.6)	(100)	(100)	(33.2)	(100)	(100)
148		20	318	100	(25.3)	(28.1)	(25.1)	(30.1)	(100)	(29.8)	(36.8)	(100)	(43.6)
149		20	318	100	(35.1)	(100)	(37.8)	(39.6)	(100)	(39.3)	(100)	(100)	(100)
150		20	318	100	(26.1)	(100)	(100)	(30.6)	(100)	(100)	(38.8)	(100)	(100)
151		20	318	100	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
152		20	318	100	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
153		20	318	100	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
154	R18	20	315	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
155		20	315	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
156		20	315	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
157		20	315	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
158		20	315	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
159		20	315	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
160		20	315	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
161		20	315	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
162		20	315	200	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)

Table S4 Comparison of the Performance of MibS Solver and BBC2 Algorithm

$\|N\|, \|A\|, \|K\|$	B	MibS		BBC2	
		Time (s)	Obj. Fun. Value	Time (s)	Obj. Fun. Value
10, 15, 5	1	11	138,749	< 1	138,749
	2	14	292,166	< 1	292,166
	3	30	444,844	< 1	444,844
	4	52	556,297	< 1	556,297
	5	54	635,497	< 1	635,497
10, 15, 10	1	6.6	898,918	< 1	898,918
	2	24	1,012,154	< 1	1,012,154
	3	65	1,123,046	< 1	1,123,046
	4	105	1,228,028	< 1	1,228,028
	5	101	1,299,546	< 1	1,299,546
10, 20, 5	1	260	331,105	< 1	331,105
	2	519	469,584	< 1	469,584
	3	648	573,549	< 1	573,549
	4	671	576,106	< 1	576,106
	5	851	637,286	< 1	637,286
10, 20, 10	1	176	592,579	< 1	592,579
	2	655	1,038,001	< 1	1,038,001
	3	797	1,164,033	< 1	1,164,033
	4	907	1,270,589	< 1	1,270,589
	5	1,260	1,305,904	< 1	1,305,904

Table S5 Performance of BBC2 Algorithm for Uncertain Demand and Interdiction Budget

Set	$\|\Omega\|=16$						$\|\Omega\|=32$						$\|\Omega\|=64$					
	$\|S\|=2$		$\|S\|=3$		$\|S\|=4$		$\|S\|=2$		$\|S\|=3$		$\|S\|=4$		$\|S\|=2$		$\|S\|=3$		$\|S\|=4$	
	Time	Gap	Time	Gap	Time	Gap	Time	Gap	Time	Gap	Time	Gap	Time	Gap	Time	Gap	Time	Gap
R04	100	0	253	0	291	0	52	0	293	0	369	0	147	0	660	0	649	0
	903	0	4,692	0	4,740	0	117	0	364	0	18,872	0	161	0	461	0	5,828	0
	302	0	12,622	0	27,556	0	217	0	1,586	0	1,534	0	532	0	6,975	0	1,416	0
	174	0	162	0	117	0	543	0	394	0	185	0	1,252	0	709	0	233	0
	1,531	0	1,567	0	897	0	3,716	0	1,638	0	8,560	0	9,989	0	17,312	0	18,043	0
	1,980	0	2,356	0	4,500	0	1,331	0	4,112	0	6,482	0	4,105	0	37,898	0	19,502	0
	12	0	12	0	14	0	38	0	24	0	32	0	155	0	144	0	136	0
	66	0	37	0	42	0	153	0	104	0	109	0	352	0	513	0	299	0
	208	0	132	0	116	0	215	0	391	0	244	0	740	0	995	0	606	0
R05	2,272	0	3,122	0	2,218	0	3,630	0	3,938	0	2,144	0	4,891	0	4,636	0	2,582	0
	21,001	0	49,484	0	10,466	0	34,630	0	86,400	3.1	34,630	0	86,400	6.4	86,400	8.5	86,400	5.9
	49,639	0	86,400	5.5	86,400	9.5	86,400	6.3	86,400	12.4	86,400	13.8	86,400	15.4	86,400	20.0	86,400	22.3
	831	0	1,270	0	1,046	0	3,082	0	1,265	0	1,106	0	3,655	0	1,103	0	1,713	0
	23,383	0	10,687	0	13,536	0	65,814	0	38,884	0	78,051	0	86,400	3.5	70,938	0	86,400	2.2
	86,400	2.6	86,400	4.0	86,400	5.0	86,400	10.8	86,400	9.9	86,400	11.2	86,400	34.2	86,400	25.8	86,400	29.6
	14	0	18	0	22	0	28	0	28	0	36	0	71	0	71	0	81	0
	45	0	65	0	53	0	90	0	115	0	149	0	581	0	523	0	462	0
	83	0	72	0	76	0	171	0	163	0	178	0	567	0	682	0	615	0
R06	1,814	0	8,486	0	4,386	0	11,387	0	23,003	0	6,660	0	21,557	0	68,702	0	41,603	0
	86,400	3.1	86,400	6.6	86,400	8.3	86,400	6.5	86,400	15.8	86,400	14.0	86,400	9.2	86,400	18.1	86,400	27.2
	86,400	7.3	86,400	14.7	86,400	18.1	86,400	16.3	86,400	28.4	86,400	31.6	86,400	11.4	86,400	32.6	86,400	36.9
	3,129	0	3,873	0	3,477	0	12,406	0	21,375	0	12,993	0	56,036	0	67,541	0	85,170	0
	86,400	2.9	86,400	7.1	86,400	6.6	86,400	8.7	86,400	11.9	86,400	15.7	86,400	10.7	86,400	15.3	86,400	17.1
	86,400	6.4	86,400	13.3	86,400	13.7	86,400	12.7	86,400	20.7	86,400	15.4	86,400	10.1	86,400	22.6	86,400	22.4
	19	0	24	0	39	0	25	0	33	0	53	0	46	0	48	0	70	0
	72	0	84	0	122	0	103	0	125	0	190	0	320	0	365	0	362	0
	14,062	0	45,776	0	86,400	1.9	86,400	2.5	86,400	2.9	86,400	7.0	86,400	8.0	86,400	6.0	86,400	30.2
R07	56	0	1,908	0	86,400	1.2	71	0	3,363	0	86,400	2.0	73	0	5,187	0	86,400	2.1
	43	0	3,657	0	86,400	6.1	54	0	17,876	0	86,400	8.8	70	0	49,251	0	86,400	11.3
	53	0	19,114	0	86,400	11.5	76	0	18,015	0	86,400	12.4	80	0	86,400	2.2	86,400	17.5
	60	0	8,639	0	26,299	0	84	0	66,495	0	35,190	0	95	0	86,400	0.8	86,400	1.0
	62	0	86,400	1.1	86,400	10.4	132	0	86,400	1.3	86,400	8.0	339	0	86,400	2.5	86,400	10.5
	65	0	86,400	2.9	86,400	11.8	141	0	86,400	2.0	86,400	14.4	268	0	86,400	4.0	86,400	16.4
	159	0	170	0	169	0	310	0	514	0	499	0	532	0	575	0	854	0
	3,677	0	2,947	0	4,401	0	26,054	0	23,274	0	20,791	0	23,364	0	29,665	0	30,103	0
	7,821	0	10,879	0	9,645	0	41,583	0	66,909	0	43,403	0	54,813	0	71,027	0	86,400	1.1
R08	695	0	86,400	2.2	86,400	8	586	0	86,400	2.4	86,400	8.2	684	0	86,400	3.5	86,400	10.0
	2,418	0	86,400	10.7	86,400	30.0	3,185	0	86,400	12.3	86,400	28.4	9,329	0	86,400	12.0	86,400	35.8
	1,294	0	86,400	14.6	86,400	30.3	1,531	0	86,400	13.5	86,400	30.7	4,379	0	86,400	14.7	86,400	35.6
	5,333	0	86,400	2.4	86,400	5.9	4,218	0	86,400	2.9	86,400	5.6	28,956	0	86,400	2.4	86,400	12.6
	10,829	0	86,400	15.4	86,400	32.0	15,928	0	86,400	16.0	86,400	30.7	56,127	0	86,400	19.8	86,400	40.7
	4,057	0	86,400	16.8	86,400	30.6	21,422	0	86,400	20.4	86,400	32.3	67,081	0	86,400	26.2	86,400	36.4
	2,836	0	2,239	0	2,877	0	5,643	0	3,870	0	2,381	0	33,591	0	9,979	0	18,250	0
	41,357	0	78,196	0	86,400	2.4	86,400	3.9	86,400	2.3	86,400	5.8	86,400	13.0	86,400	14.4	86,400	16.6
	76,853	0	86,400	3.7	86,400	5.7	86,400	4.3	86,400	8.6	86,400	10.2	86,400	0.1	86,400	13.8	86,400	21.3
R09	86,400	1.3	86,400	5.2	86,400	100.0	86,400	2.3	86,400	8.7	86,400	46.9	86,400	3.0	86,400	14.3	86,400	100
	86,400	4.5	86,400	26.7	86,400	48.3	86,400	9.6	86,400	34.7	86,400	100	86,400	15.5	86,400	41.9	86,400	55.1
	86,400	10.3	86,400	29.4	86,400	44.0	86,400	12.3	86,400	35.1	86,400	51.3	86,400	18.4	86,400	42.8	86,400	49.7
	42,605	0	86,400	4.7	86,400	100.0	27,843	0	86,400	7.3	86,400	100	86,400	3.4	86,400	10.7	86,400	100
	86,400	16.7	86,400	35.7	86,400	45.0	86,400	21.9	86,400	41.9	86,400	43.9	86,400	33.8	86,400	39.0	86,400	44.8
	86,400	21.2	86,400	38.8	86,400	41.6	86,400	30.1	86,400	39.1	86,400	45.8	86,400	35.5	86,400	40.2	86,400	48.7
	1,958	0	1,920	0	1,073	0	3,709	0	3,776	0	2,008	0	16,214	0	14,743	0	10,339	0
	25,900	0	20,015	0	24,545	0	59,098	0	48,102	0	31,768	0	86,400	1.1	86,400	1.9	86,400	3.2
	58,683	0	49,099	0	44,406	0	78,513	0	86,400	1.0	86,400		86,400	4.9	86,400	8.7	86,400	2.5
Average	23,564	1.4	39,963	4.8	45,065	11.6	30,325	2.7	46,482	6.6	48,915	13.1	37,829	4.4	53,332	8.6	55,624	16.1

Appendix E: Sensitivity Analysis

Figure S1 shows the effects of changing the number of scenarios on the optimal network configuration. Table S6 reports the results showing the effects of varying the weight of pre-interdiction cost (Φ) on the number of arcs installed in the network. The results showing the effects of varying the weight of pre-interdiction cost on the CPU time of BBC2 algorithm is presented in Table S7.

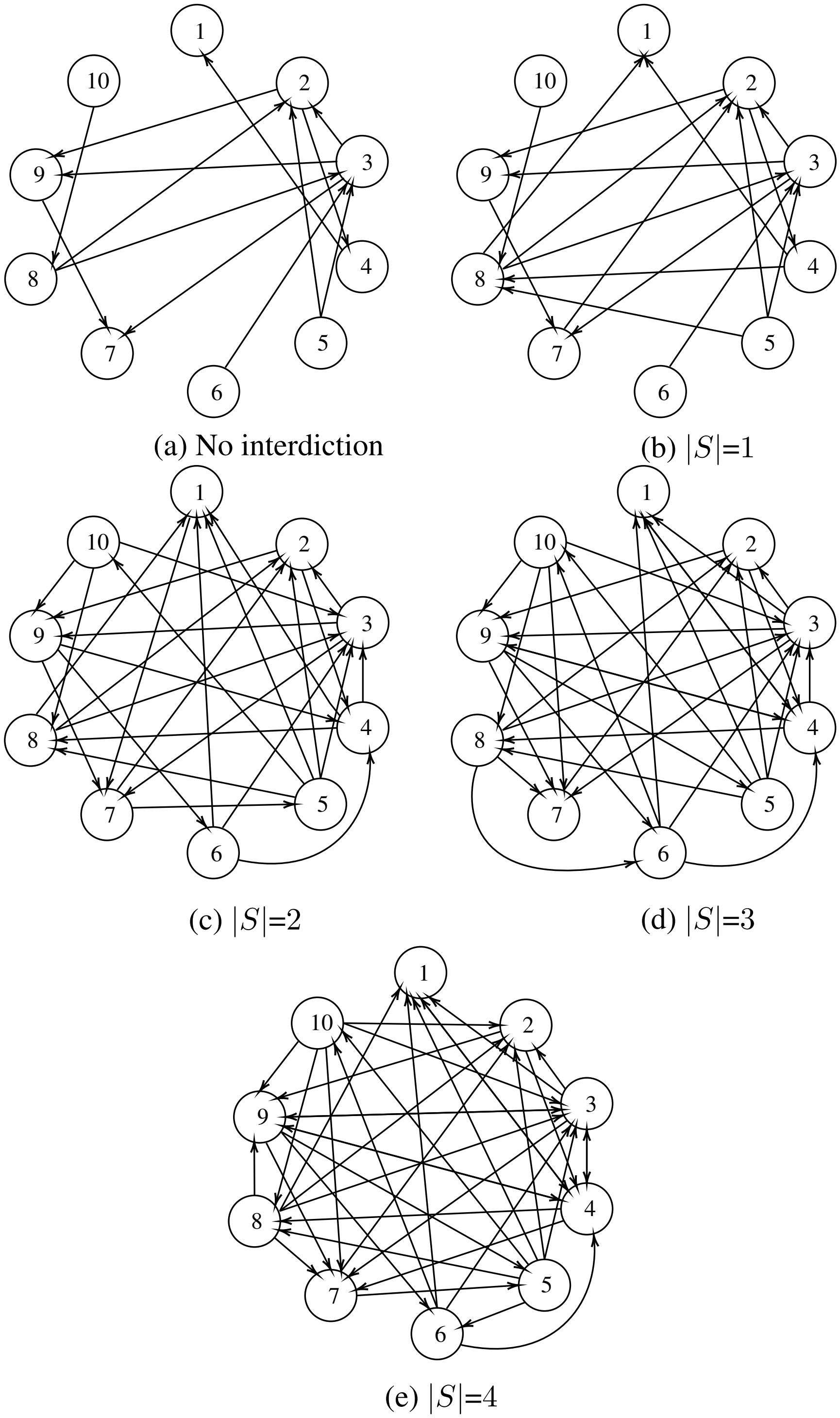

Figure S1 **Effects of Changing the Number of Scenarios on Optimal Network Configuration**

Table S6 Effects of Varying Weight of Pre-interdiction Cost (Φ) on the Number of Arcs Installed in Network

$\|N\|$	$\|A\|$	$\|K\|$	$\Phi=1$	No. of Arcs for $\|S\|=2$				No. of Arcs for $\|S\|=3$				No. of Arcs for $\|S\|=4$			
				$\Phi=0.8$	$\Phi=0.5$	$\Phi=0.2$	$\Phi=0$	$\Phi=0.8$	$\Phi=0.5$	$\Phi=0.2$	$\Phi=0$	$\Phi=0.8$	$\Phi=0.5$	$\Phi=0.2$	$\Phi=0$
10	82	10	29	36	36	36	36	36	37	37	37	36	37	37	37
10	83	25	24	41	41	42	44	49	50	50	51	58	59	59	60
10	83	25	39	48	48	49	50	51	53	53	53	51	55	55	55
10	83	50	41	59	61	63	63	63	64	64	64	64	64	65	66
20	120	40	102	103	105	105	106	104	105	106	107	104	106	106	108
20	120	40	79	84	85	85	86	87	88	90	91	89	91	91	92

Table S7 Effects of Varying the Weight of Pre-interdiction Cost (Φ) on CPU Time (s)

Set	Instance	CPU Time (s) for $\|S\|=2$			CPU Time (s) for $\|S\|=3$			CPU Time (s) for $\|S\|=4$		
		$\Phi=0.8$	$\Phi=0.5$	$\Phi=0.2$	$\Phi=0.8$	$\Phi=0.5$	$\Phi=0.2$	$\Phi=0.8$	$\Phi=0.5$	$\Phi=0.2$
R07	55	28	27	33	368	380	425	1,899	3,025	8,447
	56	22	23	34	216	288	432	2,090	2,245	2,891
	57	27	32	38	199	251	282	1,592	2,269	2,474
	58	63	72	70	689	794	966	1,070	1,726	1,900
	59	36	47	68	332	434	982	1,129	1,321	2,304
	60	27	65	87	356	415	460	1,335	1,452	1,472
	61	50	103	150	97	124	132	73	119	168
	62	48	96	113	71	100	172	102	124	185
	63	40	97	123	92	102	118	88	92	125
Arith. Mean		38	62	79	269	321	441	1,042	1,375	2,218
Geo. Mean		36	54	69	211	257	340	573	743	1,047