Supplementary results

Víctor Bucarey[a], Bernard Fortz[c,d], Natividad González-Blanco [b,*], Martine Labbé[c,d], Juan A. Mesa[b]

[a] *Data Analytics Laboratory, Vrije Universiteit Brussel, Brussels, Belgium.*
[b] *Departamento de Matemática Aplicada II de la Universidad de Sevilla, Seville, Spain.*
[c] *Département d'Informatique, Université Libre de Bruxelles, Brussels, Belgium.*
[d] *Inria Lille-Nord Europe, Villeneuve d'Ascq, France.*

With this supplementary material we provide some extra results to compare the performance of the different families of *Benders cuts* and some preliminary experiments. In Section 1, we show you results for the preliminary experiments described in Section 4.2 of the paper for the benchmarks instances, Sevilla and Sioux Falls networks. In Section 2, we show you the results of Table 6 of the paper in a disaggregated manner. Finally, in Section 3, we have performed the same set of experiments from Section 4.4 but for the benchmark instances.

1. Preliminary experiments for Sevilla and Sioux Falls networks

Network	Formulation using (2.5)-(2.6)		Formulation using (2.11)-(2.12)	
	t	LP gap	t	LP gap
Sioux	48.81	2.96	177.39	7.4
	gap	LP gap	gap	LP gap
Sevilla	3.02	1.26	$+\infty$	2.31

Table 1: Comparing the performance of the two different types of mode choice and capacity constraints for (MC) within a time limit of 1 hour for Sevilla and Sioux Falls instances.

Network	BD_Norm1		BD_Norm2		BD_Norm3	
	t	cuts	t	cuts	t	cuts
Sioux	35.43	3204	1186.46	4761	53.06	19480
Sevilla	490.96	3898	-		526.56	68203

Table 2: Comparing the performance of the three dual normalizations within a time limit of 1 hour for (MC) for Sevilla and Sioux Falls instances.

*Corresponding author
Email addresses: vbucarey@vub.be (Víctor Bucarey), bernard.fortz@ulb.ac.be (Bernard Fortz), ngonzalez2@us.es (Natividad González-Blanco), mlabbe@ulb.ac.be (Martine Labbé), jmesa@us.es (Juan A. Mesa)

Network	BD_CW		Algorithm 3+BD_CW	
	t	cuts	t	cuts
Sioux	30.22	2959	34.08	3254
Sevilla	438.39	6322	706.78	8824

Table 3: Comparing the performance of the Algorithm 3 for (MC) for Sevilla and Sioux Falls instances.

2. Time performance for random instances N40

		BD_Trd		BD_Norm		BD_CW	
		t_master	t_subpbs	t_master	t_subpbs	t_master	t_subpbs
MC	without CS	1040.49	54.76	501.53	39.5	445.91	11.9
	+CS	607.52	29.97	546.50	29.37	267.76	4.63
PC	without CS	467.26	36.8	485.61	28.81	824.85	12.56
	+CS	223.79	37.95	278.61	44.6	187.08	10.47

Table 4: Comparing the performance of the master and the subproblems in the three algorithms for (MC) and (PC).

3. Branch-and-Benders-cut performance for Germany instances

In this section, we show you the same set of experiments from Section 4.4 of the paper for some benchmark instances. G50 (germany50) and Ta2 are two instances taken from the repository `http://sndlib.zib.de/home.action?show=/problem.details.action%3FproblemName%3Dgermany50--D-B-L-N-C-A-N-N%26frameset`. From data of G50 instance we have used the topology of the underlying network, the cost and the distance for each arc and the demand matrix. Its O/D pairs set has 73% of them with zero demand. From Ta2 instance we have only used the topology of the underlying network, the cost vector for the set of arcs and the demand matrix. Its O/D pairs set has 61% of them with zero demand. The rest of the parameters have been chosen in the same manner as for the *random instances*. Note that cells with symbol $*$ is because the algorithm did not get the optimal solution in one hour.

	Network	CPLEX	Auto_BD		BD_Trd		BD_Norm		BD_CW	
w.o. CS	G50	gap	gap	cuts	gap	cuts	gap	cuts	gap	cuts
		$+\infty$	26.31	4274	37.34	9788	30.06	11842	31.51	11472
		t	t	cuts	t	cuts	t	cuts	t	cuts
	Ta2	$*$	$*$		1109.98	12969	471.03	13560	215.13	12628
	Sevilla	$*$	870.56	923	720.64	8192	490.96	3898	438.39	6322
	Sioux	50.91	20.07	1547	42.89	3197	35.43	3204	30.22	2959
+CS	G50	gap	gap	cuts	gap	cuts	gap	cuts	gap	cuts
			20.85	3176	36.27	11856	40.74	15847	19.22	8836
		t	t	cuts	t	cuts	t	cuts	t	cuts
	Ta2	-	$*$		486.52	8890	280.52	7704	156.11	11117
	Sevilla	-	738.28	593	384.42	5992	205.59	6125	313.07	7149
	Sioux	-	17.19	1410	25.48	2588	30.53	2974	22.85	3496

Table 5: Comparing the performance of the three algorithms for (MC).

	Network	`CPLEX`	`Auto_BD`		`BD_Trd`		`BD_Norm`		`BD_CW`	
w.o. `CS`	G50	`gap`	`gap`	`cuts`	`gap`	`cuts`	`gap`	`cuts`	`gap`	`cuts`
		72.44	11.38	7450	28.57	14176	21.78	9977	20.68	7602
		`t`	`t`	`cuts`	`t`	`cuts`	`t`	`cuts`	`t`	`cuts`
	Ta2	$*$	$*$		257.44	6950	$*$		190.80	6386
	Sevilla	$*$	970.02	1039	512.08	4342	981.47	4698	516.30	3549
	Sioux	1998.19	199.09	1708	513.11	3986	548.05	3849	1043.59	4118
+`CS`	G50	`gap`	`gap`	`cuts`	`gap`	`cuts`	`gap`	`cuts`	`gap`	`cuts`
			10.04	5459	18.76	8185	23.96	11915	14.22	9819
		`t`	`t`	`cuts`	`t`	`cuts`	`t`	`cuts`	`t`	`cuts`
	Ta2	-	$*$		138.94	2763	127.19	2645	75.78	2702
	Sevilla	-	759.69	744	466.20	5776	665.26	7622	463.45	3934
	Sioux	-	185.94	1390	1094.30	3680	551.5	3668	429.85	3306

Table 6: Comparing the performance of the three algorithms for (PC).

	`Auto_BD`		`BD_Trd`		`BD_Norm`		`BD_CW`	
	`t`	`cuts`	`t`	`cuts`	`t`	`cuts`	`t`	`cuts`
without{`CS`, `IS`, `RNC`}	870.56	923	720.64	8192	490.96	3898	438.39	6322
+`CS`	738.28	593	384.42	5992	205.59	6125	313.07	7149
+`CS`+`IS`	442.92	1647	374.52	7531	614.75	8588	240.75	7457
+`CS`+`IS`+`RNC`	-		429.31	6627	528.06	6493	200.81	6369

Table 7: Computing gaps to solve (MC) for Sevilla instance comparing the performance of three families of Benders cuts.

	`Auto_BD`		`BD_Trd`		`BD_Norm`		`BD_CW`	
	`t`	`cuts`	`t`	`cuts`	`t`	`cuts`	`t`	`cuts`
without{`CS`, `IS`, `RNC`}	970.02	1039	512.08	4342	981.47	4698	516.30	3549
+`CS`	759.69	744	466.20	5776	665.26	7622	463.45	3934
+`CS`+`IS`	723.45	398	477.65	4957	1507.48	4756	681.03	4919
+`CS`+`IS`+`RNC`	-		548.86	4340	887.44	3682	547.64	4383

Table 8: Computing gaps to solve (PC) for Sevilla instance comparing the performance of three families of Benders cuts.

	`Auto_BD`		`BD_Trd`		`BD_Norm`		`BD_CW`	
	`t`	`cuts`	`t`	`cuts`	`t`	`cuts`	`t`	`cuts`
without{`CS`, `IS`, `RNC`}	20.07	1547	42.89	3197	35.43	3204	30.22	2959
+`CS`	17.19	1410	25.48	2588	30.53	2974	22.85	3496
+`CS`+`IS`	19.80	1211	36.47	3088	39.44	3068	15.75	2853
+`CS`+`IS`+`RNC`	-		50.06	2723	30.08	2355	17.5	2417

Table 9: Computing gaps to solve (MC) for Sioux Falls instance comparing the performance of three families of Benders cuts.

	Auto_BD		BD_Trd		BD_Norm		BD_CW	
	t	cuts	t	cuts	t	cuts	t	cuts
without{CS, IS, RNC}	199.09	1708	513.11	3986	548.05	3849	1043.59	4118
+CS	185.94	1390	1094.30	3680	551.5	3668	429.85	3306
+CS+IS	388.28	1558	664.44	4142	1279.44	3447	640.14	3558
+CS+IS+RNC	-		602.34	3104	467.22	3695	442.91	3242

Table 10: Computing gaps to solve (PC) for Sioux Falls instance comparing the performance of three families of Benders cuts.

	Auto_BD		BD_Trd		BD_Norm		BD_CW	
	gap	cuts	gap	cuts	gap	cuts	gap	cuts
without{CS, IS, RNC}	26.31	4274	37.34	9788	30.06	11842	31.51	11472
+CS	20.85	3176	36.27	11856	40.74	15847	19.22	8836
+CS+IS	56.12	3492	27.28	11635	29.74	11707	17.83	8206
+CS+IS+RNC	-		31.33	10150	32.27	11538	14.14	9064

Table 11: Computing gaps to solve (MC) for Germany50 instance comparing the performance of three families of Benders cuts.

	Auto_BD		BD_Trd		BD_Norm		BD_CW	
	gap	cuts	gap	cuts	gap	cuts	gap	cuts
without{CS, IS, RNC}	11.38	7450	28.57	14176	21.78	9977	20.68	7602
+CS	10.04	5459	18.76	8185	23.96	11915	14.22	9819
+CS+IS	11.65	3446	21.32	9552	21.14	9087	18.54	5075
+CS+IS+RNC	-		23.85	9771	19.93	8326	13.87	7955

Table 12: Computing gaps to solve (PC) for Germany50 instance comparing the performance of three families of Benders cuts.

	Auto_BD		BD_Trd		BD_Norm		BD_CW	
	t	cuts	t	cuts	t	cuts	t	cuts
without{CS, IS, RNC}	$*$		1109.98	12969	471.03	13560	215.13	12628
+CS	$*$		486.52	8890	280.52	7704	156.11	11117
+CS+IS	$*$		587.41	13452	884.95	12530	161.14	8893
+CS+IS+RNC	-		500.14	10302	530.34	8238	136.13	8307

Table 13: Computing times to solve (MC) for Ta2 instance comparing the performance of three families of Benders cuts.

	Auto_BD		BD_Trd		BD_Norm		BD_CW	
	t	cuts	t	cuts	t	cuts	t	cuts
without{CS, IS, RNC}	$*$		257.44	6950	$*$		190.80	6386
+CS	$*$		138.94	2763	127.19	2645	75.78	2702
+CS+IS	1456.67	254	153.48	3322	141.69	3535	57.88	2411
+CS+IS+RNC	-		196.55	3286	136.94	2096	68.39	2512

Table 14: Computing times to solve (PC) for Ta2 instance comparing the performance of three families of Benders cuts.

C_{max}	BD_Norm+CS		BD_CW+CS	
	gap	cuts	gap	cuts
$0.2\,TC$	74.40	5026	48.01	5806
$0.3\,TC$	40.74	15847	19.22	8836
$0.4\,TC$	19.56	19193	11.53	14858

a.

u	BD_Norm+CS		BD_CW+CS	
	gap	cuts	gap	cuts
$1.5\,SPath$	67.57	13531	11.95	7545
$2\,SPath$	40.74	15847	19.22	8836
$3\,SPath$	29.53	11875	18.22	8559

b.

Table 15: Sensitivity analysis for (MC) with G50 instance.

β	BD_Norm+CS		BD_CW+CS	
	gap	cuts	gap	cuts
0.3	21.14	9174	18.22	4154
0.5	23.96	11915	14.22	9819
0.7	17.47	15498	17.80	12736

a.

u	BD_Norm+CS		BD_CW+CS	
	t	cuts	t	cuts
$1.5\,SPath$	23.91	12491	10.14	6651
$2\,SPath$	23.96	11915	14.22	9819
$3\,SPath$	15.95	11761	13.02	6744

b.

Table 16: Sensitivity analysis for (PC) with G50 instance.

C_{max}	BD_Norm+CS		BD_CW+CS	
	t	cuts	t	cuts
$0.2\,TC$	1124.30	7746	329.89	7124
$0.3\,TC$	280.52	7704	156.11	11117
$0.4\,TC$	84.25	3220	23.30	3936

a.

u	BD_Norm+CS		BD_CW+CS	
	t	cuts	t	cuts
$1.5\,SPath$	$*$		457.70	11266
$2\,SPath$	280.52	7704	156.11	11117
$3\,SPath$	458.73	8775	109.36	9604

b.

Table 17: Sensitivity analysis for (MC) with Ta2 instance.

β	BD_Norm+CS		BD_CW+CS	
	t	cuts	t	cuts
0.3	214.30	1177	162.83	2715
0.5	127.19	2645	75.78	2702
0.7	2689.41	6668	461.44	5573

a.

u	BD_Norm+CS		BD_CW+CS	
	t	cuts	t	cuts
$1.5\,SPath$	327.73	4821	130.81	3251
$2\,SPath$	127.19	2645	75.78	2702
$3\,SPath$	130.03	2745	65.98	2277

b.

Table 18: Sensitivity analysis for (PC) with Ta2 instance.

C_{max}	BD_Norm+CS		BD_CW+CS	
	t	cuts	t	cuts
$0.3\,TC$	$*$		2235.92	13429
$0.5\,TC$	205.59	6125	313.07	7149
$0.7\,TC$	83.56	2244	18.95	2259

a.

u	BD_Norm+CS		BD_CW+CS	
	t	cuts	t	cuts
$1.5\,SPath$	$*$		1719.39	7187
$2\,SPath$	205.59	6125	313.07	7149
$3\,SPath$	306.01	5698	91.11	5514

b.

Table 19: Sensitivity analysis for (MC) with Sevilla instance.

β	BD_Norm+CS		BD_CW+CS	
	t	cuts	t	cuts
0.3	383.03	2596	246.34	4135
0.5	665.26	7622	463.45	3934
0.7	1050.97	6613	282.01	6184

a.

u	BD_Norm+CS		BD_CW+CS	
	t	cuts	t	cuts
$1.5\,SPath$	3067.88	4776	1796.89	4255
$2\,SPath$	665.26	7622	463.45	3934
$3\,SPath$	453.19	4603	175.78	3479

b.

Table 20: Sensitivity analysis for (PC) with Sevilla instance.

C_{max}	BD_Norm+CS		BD_CW+CS	
	t	cuts	t	cuts
$0.3\,TC$	353.04	3440	460.61	3077
$0.5\,TC$	30.53	2974	22.85	3496
$0.7\,TC$	6.86	747	2.16	792

a.

u	BD_Norm+CS		BD_CW+CS	
	t	cuts	t	cuts
$1.5\,SPath$	152.42	4232	31.84	2876
$2\,SPath$	30.53	2974	22.85	3496
$3\,SPath$	37.33	2712	22.20	3075

b.

Table 21: Sensitivity analysis for (MC) with Sioux Falls instance.

β	BD_Norm+CS		BD_CW+CS	
	t	cuts	t	cuts
0.3	540.92	3466	808.05	2679
0.5	551.5	3668	429.85	3306
0.7	314.16	3602	105.89	3292

a.

u	BD_Norm+CS		BD_CW+CS	
	t	cuts	t	cuts
$1.5\,SPath$	969.84	4179	2160.27	2557
$2\,SPath$	551.5	3668	429.85	3306
$3\,SPath$	599.52	3362	533.09	3169

b.

Table 22: Sensitivity analysis for (PC) with Sioux Falls instance.