Boosting an Exact Logic-Based Benders Decomposition Approach by Variable Neighborhood Search – Full Result Table

Günther R. Raidl, Thomas Baumhauer, and Bin Hu
{raidl|hu}@ads.tuwien.ac.at, thomas.baumhauer@gmail.com

Institute of Computer Graphics and Algorithms
Vienna University of Technology
Favoritenstraße 9-11/1861, 1040 Vienna, Austria

This document contains the full result tables of the tests performed with our hybrid algorithm in the paper "Boosting an Exact Logic-Based Benders Decomposition Approach by Variable Neighborhood Search".

We use instances where the first-level VRP and the second-level VRPs are equally large. For the first-level VRP, satellites are randomly placed on a 201×201 grid with the depot node being located at the center. Each second-level VRP is constructed essentially in the same way considering a separate grid of the same size: The satellite is assumed to be at the center, and all customers are placed randomly at the grid. Traveling times are rounded Euclidean distances, and traveling costs are derived from these times by adding uniform random perturbations of 20%. Demands are chosen randomly from $\{1, \dots, 100\}$. The vehicle capacity and the global time limit were selected manually in a way that the instances are non-trivial.

The algorithms have been implemented with GCC 4.6 and were performed on single cores of an Intel Xeon E5540 machine with 2.53GHz. CPLEX version 12.1 was used for solving the MIPs.

The columns in Table 1 list the following data:

- instance name
- size of the master problem $|V_0|$ and the subproblems $|V_s|$
- global time budget T
- algorithm variant
- number of subproblems solved
- total CPU-time for solving the subproblems
- number of times the master problem is (re)solved
- total CPU-time for solving the master problem
- number of times where intervals derived from cuts have been merged
- number of times heuristic cuts have been corrected
- total CPU-time for obtaining (and proving) the optimal solution

Table 1: Results of the pure MIP-based BD, heuristically boosted BD variants A and B.

	size	T	Algorithm	subproblems #	subproblems time [s]	master problem #	master problem time [s]	cuts merg.	cuts corr.	total time [s]
Instance 1	15	260	MIP-based BD	28	27.9	6	11.5	-	-	85.8
Instance 1	15	260	Variant A	27	18.0	2	2.3	6	21	65.5
Instance 1	15	260	Variant B	27	17.9	2	3.8	6	21	68.7
Instance 1	15	290	MIP-based BD	48	58.1	10	31.7	-	-	137.1
Instance 1	15	290	Variant A	32	47.6	4	12.2	7	19	100.1
Instance 1	15	290	Variant B	30	48.7	2	5.2	7	24	96.8
Instance 1	15	320	MIP-based BD	29	65.1	7	13.4	-	-	136.5
Instance 1	15	320	Variant A	30	51.7	3	7.0	8	26	110.2
Instance 1	15	320	Variant B	39	62.5	2	4.1	8	31	121.4
Instance 2	15	270	MIP-based BD	33	13.0	7	12.7	-	-	54.3
Instance 2	15	270	Variant A	33	9.9	5	14.7	8	15	48.3
Instance 2	15	270	Variant B	26	9.2	2	5.3	8	17	39.5
Instance 2	15	300	MIP-based BD	37	35.6	8	23.0	-	-	92.4
Instance 2	15	300	Variant A	42	31.0	7	23.6	3	20	79.2
Instance 2	15	300	Variant B	46	30.1	2	5.6	3	30	63.6
Instance 2	15	330	MIP-based BD	26	28.6	8	21.9	-	-	86.2
Instance 2	15	330	Variant A	24	23.9	4	12.6	1	15	62.7
Instance 2	15	330	Variant B	36	29.9	3	15.2	1	24	73.4
Instance 3	16	260	MIP-based BD	33	29.9	10	8.78	-	-	113.0
Instance 3	16	260	Variant A	35	36.3	4	2.56	9	24	107.9
Instance 3	16	260	Variant B	38	36.2	2	1.02	9	26	108.3
Instance 3	16	290	MIP-based BD	37	51.7	10	27.3	-	-	214.9
Instance 3	16	290	Variant A	34	37.3	5	14.3	3	23	162.8
Instance 3	16	290	Variant B	38	48.4	2	3.6	3	26	165.6
Instance 4	17	270	MIP-based BD	40	342.9	6	3.3	-	-	718.7
Instance 4	17	270	Variant A	38	288.2	4	6.8	5	27	654.3
Instance 4	17	270	Variant B	39	290.0	3	4.1	5	28	656.3