Boosting an Exact Logic-Based Benders Decomposition Approach by Variable Neighborhood Search

Günther R. Raidl, Thomas Baumhauer, and Bin Hu [1,2]

Institute of Computer Graphics and Algorithms
Vienna University of Technology, Vienna, Austria

Abstract

Logic-based Benders decomposition (BD) extends classic BD by allowing more complex subproblems with integral variables. Metaheuristics like variable neighborhood search are becoming useful here for faster solving the subproblems' inference duals in order to separate approximate Benders cuts. After performing such a purely heuristic BD approach, we continue by exactly verifying and possibly correcting each heuristic cut to finally obtain a proven optimal solution. On a bi-level vehicle routing problem, this new hybrid approach exhibits shorter overall runtimes and yields excellent intermediate solutions much earlier than the classical exact method.

Keywords: Benders decomposition, variable neighborhood search, vehicle routing

1 Introduction

In mathematical programming *Benders Decomposition* (BD) [1] is a prominent technique for approaching large *Mixed Integer Linear Programming* (MIP) prob-

[1] This work is supported by the Austrian Science Fund (FWF) under grant P24660.
[2] Email: {raidl|hu}@ads.tuwien.ac.at, thomas.baumhauer@gmail.com

lems having a special block-diagonal structure with complicating variables. The problem is reformulated by expressing it as a master problem on only a subset of the original variables – usually the complicating integer variables – and considering the contributions of all further variables by additional inequalities, so-called *Benders cuts*. Following the concept of the cutting plane method, a restricted master problem is first solved in which none or only a small set of initial Benders cuts is considered. Obtained master problem variables are then temporarily fixed and a subproblem is solved on all remaining – typically continuous – problem variables. As the obtained solution is in general not necessarily feasible and optimal, dual solution information is used to derive one or more new, violated Benders cuts, which are added to the master problem. The master problem is then resolved and the process iterated until no further violated cuts can be derived, in which case an obtained feasible solution is optimal w.r.t. the original problem.

As pointed out, classical BD keeps integral variables in the master problem so that the resulting subproblem is an efficiently solvable *linear programming* (LP) problem and LP-duality can be exploited. Hooker and Ottosson [4,3] extended this concept to *Logic-Based BD*, in which the more general *inference duality* replaces LP-duality and the subproblem is not restricted to an LP; in particular, subproblem variables may also be integral and the objective function does not necessarily have to be linear. Constraint programming techniques turned out to be especially useful for approaching the inference dual. Logic-based BD has already been successfully applied to a variety of problems, in particular in scheduling and planning.

Other work has shown that metaheuristics can be useful to speed up classical BD for MIP problems. While the subproblems are typically solved by an efficient LP-solver, the master problem, although smaller than the original problem, remains a MIP and in general needs to be resolved with newly added cuts many times. Consequently, metaheuristics have been applied to the master problem instead of an exact method, see e.g. [7,2,5]. Substantial speedups could be achieved in this way, although proven optimal solutions are only obtained if in the end the master problem is solved to optimality.

But also BD subproblems need not necessarily always to be solved to optimality in order to obtain useful Benders cuts. This aspect in particular becomes interesting in logic-based BD with its typically more complex subproblems. With the exception of our recent study [8], however, we are not aware of any work so far where metaheuristics have been applied to discrete BD subproblems for deriving Benders cuts. A reason may lie in the difficulty to exploit duality for identifying cuts that are valid for any master solution.

In our recent work [8] we considered a logic-based BD for a bi-level vehicle routing problem and solved the master problem as well as all subproblems by means of *Variable Neighborhood Search* (VNS) [6]. A dramatic speedup and good solutions could be achieved, but the overall approach is purely heuristic. In particular, determined Benders cuts may be invalid as they may cut off feasible solutions, possibly including global optima. Once such a bad Benders cut has been added, the whole approach has no chance of finding the excluded solutions anymore, independently on how much effort is further invested.

In the current work we build upon this previous approach and remedy the conceptual disadvantage. By *exactly verifying and possibly correcting* heuristically generated Benders cuts in the course of the optimization, we undo earlier introduced invalid cuts and finally obtain a proven optimal solution, while still having the advantage of obtaining good heuristic solutions very early.

2 Bi-Level Capacitated Vehicle Routing Problem with Time Limits

We study the new approach exemplarily on the *Bi-Level Capacitated Vehicle Routing Problem with Time Limits* (2L-VRP-TL), in which goods shall be transported from a main depot via satellite depots to customers [8]. A global time limit is imposed on all deliveries, and the assignment of customers to satellite depots is prespecified. In practice this problem appears in the distribution of newspapers from a printing shop to subscribers. An assignment of subscribers to satellite depots is here naturally given as region-specific newspapers with individual supplements must be considered and each region-specific version is to be distributed via its dedicated satellite depot.

Formally, we are given

- a complete, directed graph $G_0 = (V_0, A_0)$ with node set $V_0 = \{0\} \cup V_0'$ and arc set $A_0 = V_0 \times V_0$, where the special node 0 represents the main depot and V_0' the set of satellite depots;
- for each satellite depot $s \in V_0'$ an individual, complete, directed graph $G_s = (V_s, A_s)$ with node set $V_s = \{s\} \cup V_s'$, where V_s' represents the set of customers that receive their deliveries via satellite depot s;
- a demand $q_v \geq 0$ for each customer $v \in V_s'$ and a resulting total demand $q_s = \sum_{v \in V_s'} q_v$ for each satellite depot $s \in V_0'$;
- travel cost $c_{u,v} \geq 0$ and a travel time $t_{u,v} \geq 0$ for each arc $(u, v) \in \bigcup_{s \in V_0} A_s$ representing the fastest way to go from u to v; fixed cost for the usage of a vehicle can be incorportated in arc costs $c_{0,v}$;
- vehicle capacities $Q_s \geq 0,\ s \in V_0$; we assume a homogeneous vehicle fleet for each depot and the number of vehicles is not limited;

- and the global time limit T (due time) within which all deliveries at the customers have to take place.

A solution R consists of a set of routes R_s in each subgraph $G_s,\ \forall s \in V_0$, with a route $r \in R_s$ being an ordered sequence of nodes $r = (r_i)_{i=1,\dots,|r|}$ with $r_i \in V'_s$. Each route starts at the depot s, visits the nodes as specified by r and finally ends at s again; for convenience, we define $r_0 = r_{|r|+1} = s$. Each node except the main depot 0 has to be visited exactly once.

By $c(r)$ and $q(r)$ we denote the total cost and total demand of route $r \in R_s,\ \forall s \in V_0$, respectively. Note that $q(r) \leq Q_s$ must hold. Furthermore, let $t(r_i)$ be the time needed to reach node r_i from the depot s, $\forall i = 1, \dots, |r|$. As second-level tours may only start after the goods have been delivered to the respective satellite depots by the first-level tours and all deliveries have to be performed within the due time, $t(s) + t(v) \leq T$ must hold $\forall v \in V'_s,\ s \in V'_0$. The objective is to minimize a solution's total cost $c(R) = \sum_{s \in V_0} \sum_{r \in R_s} c(r)$.

3 Logic-Based BD for 2L-VRP-TL

We decompose 2L-VRP-TL into a master problem essentially corresponding to the first-level of transporting the goods from the main depot to the satellite depots and $|V'_0|$ subproblems corresponding to the transportation from each satellite depot to the respective customers. In the master problem, the cost contributions of the subproblems are considered by Benders cuts.

We use variables $x_{u,v} \in \{0,1\},\ \forall (u,v) \in A_s,\ s \in V_0$ for indicating the arcs used in the routes, $t_v \geq 0,\ \forall v \in V_s,\ s \in V_0$ for the times $t(v)$, and $c_s \geq 0$ for the total cost of the subproblem tours in G_s for each satellite depot $s \in V_0$. The master problem can then be expressed as follows.

$$
\begin{aligned}
\text{(MP)} \quad \text{minimize} \quad & \sum_{(u,v) \in A_0} c_{u,v}\, x_{u,v} + \sum_{s \in V'_0} c_s && && (1)\\
\text{s.t.} \quad & (x(A_0), t(V'_0)) \in \mathrm{VRP}(G_0) && && (2)\\
& c_s \geq \beta_{t_s^k}(t_s) && k \in K_s,\ s \in V'_0 && (3)\\
& 0 \leq t_s \leq t_s^{\mathrm{ub}} && \forall s \in V'_0 && (4)\\
& 0 \leq c_s && \forall s \in V'_0 && (5)\\
& x_{u,v} \in \{0,1\} && \forall (u,v) \in A_0 && (6)
\end{aligned}
$$

$\mathrm{VRP}(G_s)$ (in Eq. (2) with $s = 0$) represents a valid formulation for the classical capacitated vehicle routing problem including the calculation of the corresponding traveling times $t(v)$ on graph G_s expressed on the variables $x(A_s)$ and $t(V'_s)$. In our proof-of-concept implementation, we used for this purpose a rather simple, compact MIP formulation based on Miller-Tucker-Zemlin inequalities for avoiding subtours, see [8] for details. Inequalities (3) are the Benders cuts relating c_s with t_s in order to ensure optimality. Feasibility w.r.t. the time limit T is ensured by calculating an upper bound t_s^{ub} for each satellite depot $s \in V_0'$ by assuming that in the second level each customer is directly served from its satellite depot with an own vehicle (4).

The associated subproblems for finding second-level routes assume the above master problem variables t_s to be fixed to some current values t_s^k and become for each $s \in V_0'$

$$
\begin{aligned}
(\mathrm{SP}_s(t_s^k)) \quad & \text{minimize} \quad && \sum_{(u,v)\in A_s} c_{u,v}\, x_{u,v} && && (7)\\
& \text{s.t.} && (x(A_s), t(V_s')) \in \mathrm{VRP}(G_s) && && (8)\\
& && 0 \le t_v \le T - t_s^k && \forall v \in V_s' && (9)\\
& && x_{u,v} \in \{0,1\} && \forall (u,v) \in A_s && (10)
\end{aligned}
$$

Thus, a minimum cost VRP-solution on G_s with delivery times not exceeding $T - t_s^k$ needs to be found for each $s \in V_0'$.

The Benders algorithm starts by solving a restricted MP with only a small set of initial Benders cuts derived from simple lower bounds on subproblem costs and yields values for the MP variables, i.e., t_s^k. For them, the subproblems $\mathrm{SP}_s(t_s^k)$ are solved, yielding respective solutions with costs $\hat{c}_s^k$ and latest starting times $\hat{t}_s^k = \max_{v \in V_s'} t_v \ge t_s^k$. The fixed times t_s^k and determined subproblem costs $\hat{c}_s^k$ allow for deriving logic-based Benders cuts of the form

$$
c_s \ge \beta_{t_s^k}(t_s) = \begin{cases} \hat{c}_s^k & \text{if } t_s \ge t_s^k \\ 0 & \text{else.} \end{cases} \tag{11}
$$

New cuts are added to the MP and the whole process is iterated until no further violated cuts exist. An important aspect is to store the data $(t_s^k, \hat{c}_s^k, \hat{t}_s^k)$ together with the respective solutions for all solved subproblems in order to avoid unnecessary future recalculations of subproblems $\mathrm{SP}_s(t_s)$ with $t_s^k \le t_s \le \hat{t}_s^k$. Occasionally, a merging of entries with overlapping valid time intervals is possible and corresponding dominated cuts can be removed.

Exact MIP-based BD

In the exact variant of the approach, all master problem instances as well as all subproblems are solved to optimality via a MIP-solver. For this purpose, the non-linear Benders cuts (11) are further translated into a pair of linear inequalities that also make use of additional Boolean variables; for details we again refer to [8]. Theory of logic-based BD proves that an optimal overall solution is obtained once no further violated Benders cut can be found.

Heuristic VNS-based BD

For speeding up above method we replace the exact solving of the master and subproblem instances by a VNS heuristic. This VNS is based on well-known ingredients: Clarke and Wright's savings algorithm is adapted in a straight-forward way to only merge routes when the result is feasible w.r.t. the time limits. This construction heuristic is further randomized to be able to obtain different promising solutions for performing restarts. The VNS considers the neighborhood structures intra-route 2-opt, intra-route or-opt, and inter-route 2-opt*, searching them in this order and following a first-improvement strategy. Infeasible solutions are always discarded.

4 Heuristically Boosted Exact Logic-Based BD

The VNS-based BD, although often yielding reasonably good solutions, has in general the disadvantage that Benders cuts derived from not necessarily optimal subproblem solutions may cut away actually feasible regions of the search space. Solutions lying in these wrongly excluded regions have no chance of being found in any future step. We address this limitation by continuing after performing the VNS-based BD with a *resolving phase*, in which all heuristically generated cuts are exactly verified and possibly replaced by correct counterparts. In more detail, we consider the following two variants.

Variant A – Resolve master after verifying all heuristic cuts

We iterate over all heuristically derived Benders cuts in the order as they were introduced in the VNS-based BD phase and for each exactly resolve the corresponding subproblem $\mathrm{SP}_s(t_s^k)$ by the MIP-approach. If the obtained exact solution value differs from the previously found heuristic one, the cut is corrected by setting $\hat{c}_s^k$ to the new, optimal value. Only after verifying/correcting all heuristic cuts, we exactly resolve the master problem and follow the exact BD approach, possibly performing a few more master iterations, until a proven optimal solution is reached.

Variant B – Resolve master after every corrected cut

As before we iterate over all heuristic Benders cuts and verify each by resolving the corresponding subproblem. If a cut is corrected due to a better optimal solution value, however, we now immediately go back to the VNS-based BD phase, resolving the master problem as well as possibly separating further Benders cuts heuristically. When this heuristic BD phase terminates again, we continue with exactly verifying/correcting the remaining heuristic cuts. Only after all cuts are known to be exact, we finally follow the exact BD approach again until a proven

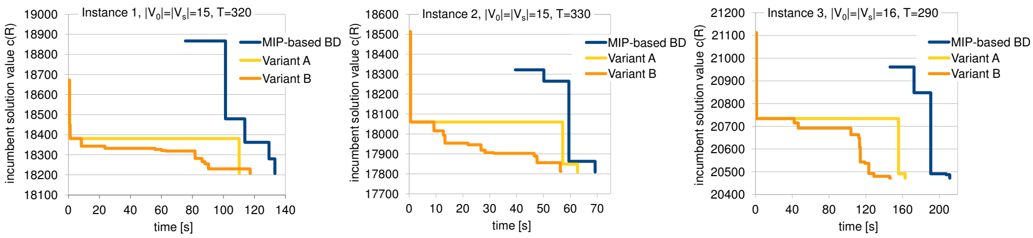

Fig. 1. Performance charts for three instances.

optimal solution is reached. We do not expect this variant to have an overall shorter running time than variant A. However, a significant practical advantage is that the incumbent solution is more continuously improved over time, as benefits gained by corrected heuristic cuts are immediately realized by resolving the master problem.

5 Experimental Results

Benchmark instances have been created as described in [8]. We present here representative snapshots of computational results, a more detailed table is available at https://www.ads.tuwien.ac.at/w/Research/Problem_Instances. CPLEX in version 12.1 was used for solving the MIPs. We compare the pure MIP-based BD to the above two heuristically boosted variants A and B. Remember that the pure heuristic BD from [8] corresponds to the first phase of the latter.

Figure 1 displays the objective values of obtained so far best feasible solutions over the whole optimization time for three different random Euclidean instances with $|V_s| \in \{15, 16\}$, $s \in V_0$ nodes. All approaches were performed until guaranteed optimality had been achieved.

First of all, we observe that in all cases, variants A and B required significantly less total time than the pure MIP-based BD. Surprisingly, variant B is frequently even faster than variant A, which can be explained by the smaller number of exactly solved master problem and subproblem instances, partly due to merged heuristic cuts. Over all our tests, we observed a median runtime reductions of 17.8% (variant A) and 18.9% (variant B). But even more importantly, the figures show clearly that good heuristic solutions could be obtained almost immediately, and especially in variant B, they were indeed continuously improved over time until reaching an optimum. In contrast, the pure MIP-based BD already required a long time to come up with any feasible solution. For larger instances, an exact solving soon becomes too time-demanding. Nevertheless, our hybrid variants still deliver good heuristic solutions relatively quickly, and variant B is usually able to continue with finding further improved solutions in short time intervals.

6 Conclusions

Solving in a logic-based BD the master problem as well as all subproblems heuristically, e.g., by a VNS, can speed up the whole approach dramatically, however at the expense of losing completeness. Here we have extended this pure heuristic approach by continuing with exactly verifying and possibly correcting heuristic cuts and obtain in the end a proven optimal solution again. This hybrid led to overall shorter running times than the pure exact BD and in particular yields excellent heuristic solutions during the course of the optimization much earlier, which are especially in variant B more continuously improved. The approach is generic in the sense that it is promising also for other problems where logic-based BD is applicable and effective (meta-)heuristics like VNS can be devised for separating approximate Benders cuts.

References

[1] J. F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4:238–252, 1962.

[2] M. Boschetti and V. Maniezzo. Benders decomposition, Lagrangian relaxation and metaheuristic design. *Journal of Heuristics*, 15:283–312, 2009.

[3] J. Hooker. *Integrated Methods for Optimization*, volume 170 of *Int. Series in Operations Research & Management Science*. Springer, 2nd edition, 2012.

[4] J. N. Hooker and G. Ottosson. Logic-based Benders decomposition. *Mathematical Programming*, 96:33–60, 2003.

[5] M.-C. Lai, H.-S. Sohn, T.-L. Tseng, and D. L. Bricker. A hybrid Benders/genetic algorithm for vehicle routing and scheduling problem. *International Journal of Industrial Engineering*, 19(1):33–46, 2012.

[6] N. Mladenović and P. Hansen. Variable neighborhood search. *Computers and Operations Research*, 24(11):1097–1100, 1997.

[7] C. Poojari and J. E. Beasley. Improving Benders decomposition using a genetic algorithm. *European Journal of Operational Research*, 199(1):89–97, 2009.

[8] G. Raidl, T. Baumhauer, and B. Hu. Speeding up logic-based Benders' decomposition by a metaheuristic for a bi-level capacitated vehicle routing problem. In C. Blum et al., editors, *Hybrid Metaheuristics, 9th Int. Workshop, HM 2014*, LNCS. Springer, to appear 2014.