A Heuristic Logic-Based Benders Method for the Home Health Care Problem

Andre A. Cire, J. N. Hooker

Tepper School of Business, Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213, U.S.A.
andrecire@cmu.edu, jh38@andrew.cmu.edu

Revised December 2012

Abstract

We propose a heuristic adaptation of logic-based Benders decomposition to the home health care problem. The objective is design routes and schedules for health care workers who visit patient homes, so as to minimize cost while meeting all patient needs and work requirements. We solve the Benders master problem by a greedy heuristic that is enhanced by the propagation facilities of constraint programming (CP). We solve the subproblems entirely by CP and generate logic-based Benders cuts that exploit problem structure. Many of the subproblem constraints are included in the master problem to compensate for the difficulty of designing strong Benders cuts, but they serve as guidance for the heuristic rather than constraints to be satisfied. We also experiment with local search in the master problem to hasten the discovery of a feasible solution. We report preliminary computational results for realistic problem instances.

1 Introduction

Home health care is a fast-growing industry in North America, Europe, and Japan, due primarily its potentially lower cost as well as aging populations. Many patients can receive basic medical care in the comfort and familiar surroundings of home, rather than being transported or admitted to treatment facilities that are expensive to operate. Home care also reduces the risk of acquiring drug-resistant infections that may spread in hospitals and nursing homes.

The cost-effectiveness of home health care depends critically on the efficient dispatch of health care workers, whom we call *nurses* for short. This poses the *home health care problem* (HHCP), which asks how home visits can be scheduled and staffed so as to make the best use of nurses while meeting patient needs.

Nurses typically start their work shift at a central office, travel directly from one patient to the next, and return to the central office at the end of the shift. The shift may be subject to a number of legal or contractual restrictions, such as a maximum work time and the need for lunch/dinner breaks. Each medical service must be performed by a nurse with the proper qualifications, and some services may be restricted to specified time windows (e.g., blood transfusion).

The HHCP is a difficult problem that adds nurse rostering elements to the vehicle routing problem with time windows [5] (VRPTW). Its natural decomposition into an assignment component (allocation of jobs to nurses) and a scheduling component (nurse routing) makes it particularly suitable to hybrid approaches, and nearly all existing optimization techniques for the HHCP rely to some extent on this decomposition.

We propose a novel solution technique based on a heuristic adaptation of *logic-based Benders decomposition* [8]. We exploit the natural decomposition by making job assignments in a *master problem* and routing/scheduling nurses in separable *subproblems*. The master problem and subproblems communicate through *logic-based Benders cuts*, which are generated in the subproblems and transmit information to master problem about the routing and scheduling consequences of previous assignments.

Logic-based Benders methods have yielded orders-of-magnitude speedups in the solution of assignment and scheduling problems, as surveyed in [11]. However, the Benders cuts tend to be weak when subproblems have sequence-dependent costs, as they do in the HHCP. Weak cuts transmit limited information to the master problem, resulting in slow convergence. We address this issue by including most of the subproblem variables and constraints in the master problem, so that assignments are more likely to take account of routing and scheduling constraints. Normally this would sacrifice the advantage of decomposition by making the master problem hard to solve. However, we do not actually satisfy the subproblem constraints when solving the master. Rather, we allow these constraints to guide job assignments, by combining a heuristic assignment algorithm with the propagation techniques of constraint programming (CP).

We accomplish this as follows. The heuristic algorithm assigns jobs to nurses in greedy fashion, using restarts as necessary. After assigning each job, we allow a CP solver to deduce some of the consequences of the assignments made so far, using propagation techniques. These consequences may include restrictions on the remaining assignments. When a complete assignment is obtained, we send it to the subproblems, which are solved to optimality by CP. The subproblems generate new Benders cuts that are returned to the master problem. We cycle through this process until an improving solution is not found. Although a logic-based Benders method normally terminates with an optimal solution, in this case the solution may not be optimal because the master problem is solved heuristically.

Because it is frequently difficult to find an initial feasible assignment for the HHCP, we also experimented with a local search heuristic in the master problem. When a greedy assignment proves to be infeasible in one or more subproblems, the process of generating Benders cuts in these subproblems generates a partial assignment that satisfies all constraints. We then use local search to try to complete this partial assigment while observing all constraints.

After a brief review of previous work, we begin below with a description of the master problem and its solution by a CP-enhanced constructive greedy heuristic. We then describe logic-based Benders cuts generated by the routing and scheduling subproblems. Finally, we report preliminary computational tests on a real-world HHC problem first introduced in [14].

2 Previous Work

An early statement of the HHCP appears in [6], which proposes two alternative mixed-integer linear programming (MILP) formulations and a two-phase constructive heuristic to solve small instances. In [17], the authors propose combining a periodic vehicle routing model with a nurse rostering model for a weekly assignment. In [3], the authors develop an hybrid approach using MILP, CP, and a tabu search metaheuristic for a weekly nurse assignment, under the assumption that job time windows are generally very tight. More recently, the work in [15] tackles large real-world HHCP instances by using an hybrid approach composed of two steps, where a Constraint Programming model is used to generate a feasible solution, and metaheuristics are applied to improve the solution.

The work in [14] was the first attempt to develop an optimization method for the German HHCP problem considered here. Three different column generation models are presented to solve the nurse assignment for a single day, considering only maximum shift length constraints and some of the break time conditions. The pricing problem is solved by a dynamic programming approach. The author reports that the proposed method takes an excessive amount of computation time for instances containing more than 6 nurses and 60 patients, for which they were unable to provide feasible solutions.

Combinations of CP with heuristic methods have received a good deal of research attention, surveyed in [4, 16, 18]. Generally they take the form of local search or large-neighborhood search [1], rather than a greedy heuristic with propagation, as described here.

Logic-based Benders decomposition, introduced in [12, 13], generalizes classical Benders decomposition [2]. It allows the subproblem to be any type of optimization problem, as opposed to a linear programming problem as in the classical method. Benders cuts are obtained by solving an *inference dual* of the subproblem, rather than the linear programming dual. The cuts must be specifically designed for each type of subproblem, but this allows them to exploit problem structure [11].

3 Problem Formulation

In the HHCP we schedule a set of medical services to be provided in patient homes over a single-day horizon. The services are represented by a set of *jobs* $\mathcal{J} = \{1, ..., J\}$. Each job $j \in \mathcal{J}$ has a fixed duration p_j, a release time r_j, and a deadline d_j. Each job j requires a *qualification level* $q_j \in \{1, ..., Q\}$, which specifies the minimum skill a nurse must have to perform the service. With each pair of jobs $(j, k) \in \mathcal{J} \times \mathcal{J}$ we also associate a nonnegative *travel distance* t_{jk}. We assume the distances are symmetric and satisfy the triangle inequality.

Each job must be assigned to a nurse in the set $\mathcal{N} = \{1, ..., N\}$. The shift of each nurse i must start within the interval $[a_i, b_i]$. There are lower and upper bounds, ℓ_i and u_i, on the total duration of the shift (end time minus start time). The qualification level of nurse i is $\bar{q}_i \in \{1, ..., Q\}$, which means that nurse i can perform any job j for which $q_j \leq \bar{q}_i$. We assume without loss of generality that the jobs never require more than one nurse.

A shift with length above threshold T_B must contain a *break* of length t_B. In addition, the shift time before the break should not exceed T_B, and similarly for the shift time after the break. For instance, suppose $T_B = 6$ hours and $t_B = 15$ minutes. If a certain nurse works for more than 6 hours, then there must be a 15 minute break between some pair of jobs, and the work time before or after this break should not exceed 6 hours. A shift must always start and end at a *central office*, where the nurse must spend p_0 time units both at the beginning and at the end of the shoft. We let t_{0j} be the travel time between patient j and the central office.

With each nurse $i \in \mathcal{N}$ we associate a cost rate c_i, so that the total cost for a nurse i is c_i times the shift duration. Because nurse i must work a period of at least ℓ_i, the minimum cost for the nurse is $c_i \ell_i$, even if the nurse visits no patients. In practice, the nurse may spend extra shift time in the central office performing administrative tasks. The problem may be formulated

$$\text{minimize} \sum_{i \in \mathcal{N}} c_i (w_i^e - w_i^s) \tag{1}$$

$$\text{subject to} \sum_{i \in \mathcal{N}} x_{ij} = 1, \ \ \forall j \in \mathcal{J} \tag{2}$$

$$w_i^s \leq \min_{j \in \mathcal{J}} \{s_j - t_{0j} \mid x_{ij} = 1\} - p_0, \ \ \forall i \in \mathcal{N} \tag{3}$$

$$w_i^e \geq \max_{j \in \mathcal{J}} \{s_j + p_j + t_{0j} \mid x_{ij} = 1\} + p_0, \ \ \forall i \in \mathcal{N} \tag{4}$$

$$w_i^s \in [a_i, b_i], \quad w_i^e \in [a_i + \ell_i, b_i + u_i], \ \ \forall i \in \mathcal{N} \tag{5}$$

$$s_j \in [r_j, d_j], \quad \forall j \in \mathcal{J} \tag{6}$$

$$\texttt{disjunctive}(\{s_j | x_{ij} = 1\}, \{p_j | x_{ij} = 1\}, \{t_{jk}\}), \ \ \forall i \in \mathcal{N} \tag{7}$$

$$w_i^e - w_i^s \geq T_B \Rightarrow \tag{8}$$

$$\left\{ \begin{array}{l} \max\{s_{iB} - w_i^s, \ w_i^e - s_{iB}\} \leq T_B \\ s_{iB} \in [s_j, s_k] \Rightarrow s_k - s_j \geq p_j + t_{jk} + t_B, \\ \quad \forall j, k \in \mathcal{J} \text{ with } x_{ij} = 1 \end{array} \right\}, \ \ \forall i \in \mathcal{N}$$

$$x_{ij} = 1 \Rightarrow \bar{q}_i \geq q_j, \ \ \forall i \in \mathcal{N}, \forall j \in \mathcal{J} \tag{9}$$

$$x_{ij} \in \{0, 1\}, \ \ \forall i \in \mathcal{N}, \forall j \in \mathcal{J} \tag{10}$$

The objective function (1) computes the total cost, where binary variable x_{ij} takes the value 1 when nurse i is assigned to job j. Constraints (3)–(4) define the start time w_i^s and end time w_i^e of each nurse i's shift, where variable s_j is the start time of job j. The formulation of these constraints assumes that travel times satisfy the triangle inequality. Constraints (5) and (6) restrict the shift times and job start times, respectively. Constraint (7) imposes a CP global constraint `disjunctive` which ensures that the jobs assigned to each nurse i do not overlap, given processing times p_j and travel times t_{jk}. If nurse i's shift is long enough to require a break, constraint (8) allows time for a break, where variable s_{iB} is the break start time. It also ensures that the shift time before and after the break is not too long. Constraint (9) makes sure that nurses are qualified to performed the assigned service.

4 The Master Problem

The master problem is solved for the assignment variables x_{ij} using a greedy heuristic. The solution is passed to the scheduling subproblems, which generate a route and schedule for each nurse. Each subproblem then generates a Benders cut as described in the next section. These cuts are added to the master problem, which is then re-solved. The process continues until an improving solution is not found in the master problem.

The master problem is as follows:

$$\text{minimize} \sum_{i\in\mathcal{N}} c_i w_i \tag{11}$$

$$\text{subject to} \sum_{i\in\mathcal{N}} x_{ij} = 1, \ \ \forall j \in \mathcal{J} \tag{12}$$

$$\text{[relaxation of the subproblems]} \tag{13}$$

$$\text{[Benders cuts]}, \tag{14}$$

$$x_{ij} = 1 \Rightarrow \bar{q}_i \geq q_j, \ \ \forall i \in \mathcal{N}, \forall j \in \mathcal{J} \tag{15}$$

$$x_{ij} \in \{0,1\}, \ \ \forall i \in \mathcal{N}, \forall j \in \mathcal{J} \tag{16}$$

Constraints (12) and (15)–(16) are as before. The subproblem relaxation is described below, and the Benders cuts in the next section.

Subproblem Relaxation Constraints (13) are motivated by practical experience [7, 8, 9, 10], which indicates that including a subproblem relaxation in the master problem can significantly reduce the number of Benders iterations. Normally, the relaxation contaions only master problemn variables, but here we depart from that practice.

The relaxation represents valid constraints that take into account the job time windows, travel distances, and the nurse shift lengths. We observed that if the master problem is formulated as an MILP, it is usually difficult to find feasible tours due to the presence of sequence-dependent travel times, which are are not well reflected in Benders cuts. Relaxations based on the VRPTW can be applied, but the resulting MILP model size is prohibitively large, as hundreds of Benders iterations may be required to find a good solution.

We therefore formulated a subproblem relaxation as a CP model, to take advantage of sophisticated propagation algorithms that can rule out non-trivial infeasible assignments. We used the following relaxation, which introduces the start time variables s_i into the master:

$$\texttt{disjunctive}(\{s_j \mid x_{ij} = 1\}, \{p_j | x_{ij} = 1\}, \{t_{jk}\}), \ \ \forall i \in \mathcal{N} \tag{17}$$

$$w_i \geq \max\{s_j + p_j \mid x_{ij} = 1\} - \min\{s_j - p_j \mid x_{ij} = 1\}, \forall i \in \mathcal{N} \tag{18}$$

$$w_i \in [l_i, u_i], \quad \forall i \in \mathcal{N} \tag{19}$$

$$s_j \in [r_j, d_j], \quad \forall j \in \mathcal{J} \tag{20}$$

The shift duration variable w_i represents $w_i^s + w_i^e$. In addition to these constraints, we also use bounds on the variables s_j derived from the nurse start times and lengths.

Greedy Constructive Heuristic To solve the master problem (11)–(16), we applied heuristics based on our specific knowledge of the problem. In particular, our heuristic receives as input a partial nurse-job assignment, and decides which job to allocate next based on the variable domains. This decision is then communicated to the CP solver, which propagates constraints (17)–(20) and Benders cuts so as to reduce the variable domains. The process is repeated until a complete assignment is found, or the instance is proved to be infeasible.

The specifics of the greedy procedure are as follows. We first select the unallocated job with the highest *priority*, which is the one that requires the highest qualification and has the least number of nurses that can serve it. We then rank the nurses based on the work shift slack, which is the maximum shift time (u_i) minus the sum of their current job durations. The job is then assigned to the nurse with minimum work shift slack, and this decision is communicated to the CP solver.

Once a complete assignment for the variables x_{ij} is found, the search is terminated, even though the solution may not be optimal. The variables w_i and s_j in the relaxation (17)–(20) may be left unfixed, but their domains will be updated due to constraint propagation. Finally, if a certain number of partial assignments performed by the algorithm are detected to be infeasible, the search is *restarted* so that different assignments can be tried.

5 Subproblems

Given an assignment solution of the master problem, we generate a separate problem for each nurse to schedule their tours. These problems are formulated and solved here using a pure CP approach.

Let $J_i \subseteq \mathcal{J}$ be the assignment of jobs to a nurse $i \in \mathcal{N}$ fixed by a master problem solution $\bar{x}$; that is, $J_i := \{j \in \mathcal{J} : \bar{x}_{ij} = 1\}$. The subproblem for nurse i is

$$\text{minimize } w_i^e - w_i^s \tag{21}$$

$$\text{subject to } w_i^s \leq s_j - p_0 - t_{j,0}, \quad \forall j \in J_i \tag{22}$$

$$w_i^e \geq s_j + p_j + t_{j,0} + p_0, \quad \forall j \in J_i \tag{23}$$

$$w_i^e - w_i^s \in [l_i, u_i] \tag{24}$$

$$\texttt{disjunctive}(\{s_j |\ j \in J_i\}, \{p_j |\ j \in J_i\}, \{t_{jk} |\ j,k \in J_i\}) \tag{25}$$

$$w_i^e - w_i^s \geq T_B \Rightarrow \tag{26}$$

$$\left\{ \begin{array}{l} \max\{s_{iB} - w_i^s,\ w_i^e - s_{iB}\} \leq T_B \\ s_{iB} \in [s_j, s_k] \Rightarrow s_k - s_j \geq p_j + t_{jk} + t_B, \quad \forall j,k \in J_i \end{array} \right\}, \quad \forall i \in \mathcal{N}$$

$$s_j \in [r_j, d_j]\,, \quad \forall j \in J_i \tag{27}$$

Benders cuts for infeasible subproblems. If subproblem (21)–(27) for an assignment J_i is infeasible, Benders cuts in the form of *nogoods* are generated to exclude this and other infeasible assignments from the master problem. The simplest nogood cut would simply rule out assigining all the jobs in J_i to nurse i. However, as in [7, 8], this is an unnecessarily weak cut, because a smaller subset of jobs is usually responsible for the infeasibility. The jobs in J_i are therefore removed one at a time, and the subproblem re-solved, until the subproblem becomes feasible. Let $\bar{J}_i$ be the smallest assignment for which the subproblem is still infeasible. We write the nogood cut

$$\sum_{j \in \bar{J}_i} (1 - x_{ij}) \geq 1. \tag{28}$$

Jobs are removed from J_i in increasing order of

$$\alpha_j = \sum_{k \in J_i \setminus \{j\}} t_{jk} + p_j$$

which attempts to measure the nurse time "consumed" by job j. The cut (28) is added for nurse i as well as for all nurses i' that share the same shift parameters; that is, nurses for which $l_i = l_{i'}$, $u_i = u_{i'}$, $a_i = a_{i'}$, and $b_i = b_{i'}$.

For any nogood cut found, new ones can be derived by applying a simple and very efficient procedure, directly derived from the following property.

Property 1 *Let J be an infeasible assignment for $S \subset \mathcal{N}$ and take any $j \in J$. For any job $k \in \mathcal{J} \setminus J$ such that $[r_k, p_k] \subseteq [r_j, p_j]$, $p_k \geq p_j$, $q_k \leq q_j$, and $t_{k\ell} \geq t_{j\ell}$ for all $\ell \in J \setminus \{j\}$, we have that the assignment $J' = (J \setminus \{j\}) \cup \{k\}$ is also infeasible for S.*

Proof Observe that, if assignment J' is feasible, then we can replace job k by j in J' to get a feasible schedule for the assignment J. □

Many of the subproblem infeasibilities are due to job time windows, rather than the shift or break conditions. In such cases, we can re-solve the subproblem without any particular nurse constraints to obtain nogood cuts that are valid for all nurses in $\mathcal{N}$.

Benders cuts for feasible subproblems. For a nurse $i \in \mathcal{N}$ and respective assignment J_i, assume now that the problem defined by (21)–(27) is feasible, and let w_i^* be its optimal solution. Hence, assigning jobs J_i to i incurs shift length w_i^*, and the following Benders cut can be added to the master problem.

$$w_i \geq w_i^* \left(1 - \sum_{j \in J_i} (1 - x_{ij}) \right) \tag{29}$$

This cut can be often strengthened by observing that some jobs do not affect the minimum work time w_i^*. These jobs are identified by taking advantage of the following property.

Property 2 *Let s_j^* be the start time of a job $j \in J_i$ in some optimal subproblem solution. Suppose that the solution has no breaks, and that j is not the first or last jobs in the optimal tour. If one of the conditions $d_k + p_k + t_{kj} \leq s_j^*$ or $s_j^* + p_j + t_{jk} \leq r_k$ hold for each $k \in J_i \setminus \{j\}$, then the optimal work time is the same for J_i and $J_i \setminus \{j\}$.*

Proof For any optimal subproblem solution of the assignment $J_i \setminus \{j\}$, the above conditions ensure that job j can be injected into the optimal tour with the same time s_j^*, resulting in a valid solution and not affecting the time worked. □

Property 2 can be checked for all $j \in J_i$. The complexity of this operation is $O(|J_i| \log |J_i|)$ in the worst case. The resulting cut is then added for nurse i and all nurses i' that share the same shift parameters.

It is worthwhile in practice to deduce still tighter bounds by removing some jobs and re-solving the subproblem, which is faster than generating a new solution by solving the master problem. Let J_i' be the set of jobs after property 2 is applied. The procedure consists of removing jobs one by one from J_i' using the α_j-based criteria again, recalculating the optimal worked time w_i^* for each job removed. The procedure stops when $w_i^* = l_i$, which means that deleting any jobs will not change the nurse total worked time.

6 Local Search

If subproblem i is infeasible, the cut strengthening procedure nonetheless generates a partial solution by removing jobs from J_i one at a time until a feasible assignment is found. Hence, we find a set $J_i' \subset J_i$ for which there is a feasible schedule that fails to cover the jobs in $L_i = J_i \setminus J_i'$. Repeating this operation for all subproblems, we have a solution that satisfies all operational constraints but fails to cover jobs in $L = \bigcup_{i \in \mathcal{N}} L_i$.

This permits the application of several *local search* techniques. In our case, we define a simple but effective technique in which L defines a search neighborhood. We select an arbitrary job j in L and try to reassign it to each nurse in turn, re-solving the subproblems and generating new Benders cuts for each trial. Among the nurses for which the insertion allows a feasible schedule, we add j to the one for which the insertion results in the smallest increase in total work time. We then repeat the procedure for the remaining jobs in L. If a job in L cannot be inserted in any nurse, we restart the procedure with a different order of jobs from L. This is repeated until a predefined number of feasible insertions are discovered.

This heuristic is only applied if L is small ($|L| \leq 10$). We observed, however, that $|L|$ typically tends to decrease after a certain number of iterations of the Benders approach, as more cuts are being inferred.

7 Computational Results

In this section we report preliminary results of our Benders approach on some of the real-world instances first studied in [14]. They are obtained from two home health care offices in Hamburg, Germany, a smaller one in the Nienstedten area and a larger one in the Barmbek area of the city. The instances are based on actual scenarios that planners face every day. Tests were run on an Intel Core 2 Quad with 8 GB RAM.

Table 1: Computational results for real-world instances from [14]. The symbol ∞ indicates that no solution was found within the time limit of 3600 seconds.

Instance			*CP*		*Benders*		*Benders + LS*	
N	P	Day	BestSol	Time (s)	BestSol	Time (s)	BestSol	Time (s)
6	28	Mon	2468160	6.98	2468160	0.36	2468160	0.31
6	25	Tue	2468160	6.42	2468160	0.12	2468160	0.05
8	26	Thu	3990420	14.5	3990420	1.12	3990420	0.72
8	25	Fri	3990420	18.0	3990420	0.40	3990420	0.89
6	63	Sat	∞	3600	2772717	683	2772717	51.8
6	63	Sun	∞	3600	2775343	74.3	2775343	43.2
11	82	Wed	6128424	3593	6362094	453	6362094	300
11	92	Mon	∞	3600	∞	3600	7501288	160

The CP models for the master problem and subproblems were implemented in the ILOG CP Optimizer with extended inference level for global constraint propagation. The CP subproblems were solved with depth-first search. For comparison purposes, the full version of the complete problem was also implemented using a pure CP approach. The parameters were set for extended inference level, default search strategy, and at most 2 cores for parallel processing. The time limit for all methods was set to 3600 seconds.

A summary of the results is presented in Table 1. The first 4 instances are from Nienstedten, and the last 4 from Barmbek. Each instance corresponds to a particular day of the week, as shown. The number N of nurses and the number P of patients are indicated. The *CP* columns corresponds to the pure CP model. Two versions of the Benders approach were tested: *Benders* and *Benders + LS* correspond to versions with local search and without local search, respectively. For each method, the column BestSol registers the cost of the best solution found within 3600 seconds, and the time shown is the time that it took for the methods to reach this best solution.

The Nienstedten instances were solved to optimality by all the methods. Even though they are relatively small, the CP approach required much more time to prove optimality, which is due to the large number of operational constraints. In the Barmbek instances for Saturday, Sunday and Monday, the average number of jobs per nurse is relatively high when one observes the qualification requirements involved in such instances, and hence nurse tours are expected to be close to the limit of their maximum shift times. In those cases, only the Benders approach could obtain a feasible solution in the given time limit. In the Wednesday instance, we have the opposite scenario, with more nurses than we actually require to solve the problem, and CP is able to find a less costly solution to the instance.

We observe that *Benders + LS* approach finds the best solution faster than *Benders*, but this happens early in the procedure, and no better solution is found afterwards in both techniques. This is because both the local search and the

Table 2: Comparison with a heuristic method in [14] for wide and narrow time windows on shift start times. The instances solved in [14] are a simplified version of those solved here.

Instance			*From* [14], *Wide*		*From* [14], *Narrow*		*Benders + LS*	
N	P	Day	% gap[1]	Time[2] (s)	% gap[1]	Time[2] (s)	% gap[3]	Time (s)
6	28	Mon	0	< 600	0	< 600	0	0.31
6	25	Tue	0.68	< 600	0	< 600	0	0.05
8	26	Thu	0	< 600	9.73	< 600	0	0.72
8	25	Fri	0	< 600	0	< 600	0	0.89
6	63	Sat	1.26	< 600	7.54	< 600	?	51.8
6	63	Sun	4.31	< 600	6.35	< 600	?	43.2
11	82	Wed	0.85	< 600	3.20	< 600	3.81	300
11	92	Mon	1.82[4]	< 7200	1.88[5]	< 7200	?	160

[1]Gap between upper and lower bounds computed by the heuristic.

[2]Best solution was obtained within the time shown. Time to best solution is not reported.

[3]Gap between best value found and optimal valued computed by CP.

[4]Gap of 2.26% obtained in first 600 seconds.

[5]Gap of 7.65% obtained in first 600 seconds.

greedy construction heuristic of the master problem guide the search towards finding feasible solutions instead of improving solution cost. The motivation for this approach was that no feasible solution was known for the last 3 instances previous to this work.

Table 2 attempts a rough comparison of the Benders method with the reported performance of the heuristic developed for these instances in [14]. The comparison is inexact because [14] solves simplified versions of the instances, for which the objective function values are different from those we obtained. Each simplified instance has two versions, distinguished by the width of time windows for the shift start times. The tests in [14] check the quality of solutions after 10, 20, 30, 60 and 120 minutes, and the time required to reach the best solution found is not reported.

The heuristic in [14] found optimal solutions for 3 of the 4 Nienstedten instances, while the Benders method found optimal solutions for all four in less than a second. Because an exact version of the heuristic in [14] solved instances of this size in less than a second, we may suppose that the heuristic found its best solution very quickly. We know the optimal value for only one of the 4 Barmbek instances we solved, which makes it hard to compare the two methods on these instances with respect to solution quality. We note, however, that for 3 of the 4 instances, the heuristic in [14] could not reduce the gap that it obtained in the first 10 minutes, even if it is allowed to run 2 hours.

8 Conclusion and Future Work

We applied a heuristic adaptation of logic-based Benders decomposition to the home health care problem. Rather than solve the master problem to optimality, we used a constructive greedy heuristic that is guided by CP-based propagation of subproblem constraints and Benders cuts. In prelimninary tests, we found that the method can obtain good and sometimes optimal solutions for problems of realistic size. The method is substantially faster than CP and obtains reasonable solutions for some instances that are intractable for CP.

Although we focused on the HHCP, we envision that this technique may be applicable to VRPTW or machine scheduling problems with difficult side constraints.

References

[1] R. K. Ahuja, Ö. Ergun, J. B. Orlin, and A. P. Punnen. A survey of very large scale neighborhood search techniques. *Discrete Applied Mathematics*, 123:75–102, 2002.

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

[3] Stefan Bertels and Torsten Fahle. A hybrid setup for a hybrid scenario: combining heuristics for the home health care problem. *Computers and Operations Research*, 33(10):2866–2890, 2006.

[4] C. Blum, J. Puchinger, G. Raidl, and A. Roli. Hybrid metaheuristics. In P. van Hentenryck and M. Milano, editors, *Hybrid Optimization: The Ten Years of CPAIOR*, pages 305–336. Springer, 2011.

[5] Olli Braysy and Michel Gendreau. Vehicle Routing Problem with Time Windows, Part I: Route Construction and Local Search Algorithms. *Transportation Science*, 39(1):104–118, 2005.

[6] Eddie Cheng and Jennifer Lynn Rich. A home health care routing and scheduling problem. Technical Report TR98-04, Department of CAMM, Rice University, USA, 1998.

[7] Elvin Coban and John N. Hooker. Single-facility scheduling over long time horizons by logic-based benders decomposition. In A. Lodi, M. Milano, and P. Toth, editors, *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*, LNCS 6140, pages 87–91, 2010.

[8] J. N. Hooker. A hybrid method for planning and scheduling. In *In Procs. of the 10th Intern. Conference on Principles and Practice of Constraint Programming - CP 2004*, pages 305–316. Springer, 2004.

[9] J. N. Hooker. An integrated method for planning and scheduling to minimize tardiness. *Constraints*, 11(2-3):139–157, 2006.

[10] J. N. Hooker. Planning and scheduling by logic-based Benders decomposition. *Operations Research*, 55:588–602, 2007.

[11] J. N. Hooker. *Integrated Methods for Optimization, 2nd edition*. Springer, 2012.

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

[13] J. N. Hooker and H. Yan. Logic circuit verification by benders decomposition. In *Principles and Practice of Constraint Programming: The Newport Papers*, 1995.

[14] Arne Nielsen. Optimization of a health care system. Master's thesis, Zuse Institute Berlin, 2006.

[15] Andrea Rendl, Matthias Prandtstetter, Gerhard Hiermann, Jakob Puchinger, and Günther Raidl. Hybrid heuristics for multimodal homecare scheduling. In Nicolas Beldiceanu, Narendra Jussien, and Éric Pinson, editors, *9th International Conference on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems (CPAIOR'12)*, volume 7298 of *Lectures Notes in Computer Science*, pages 339–355, Nantes, France, June 2012. Springer Verlag.

[16] P. Shaw. Constraint programming and local search hybrids. In P. van Hentenryck and M. Milano, editors, *Hybrid Optimization: The Ten Years of CPAIOR*, pages 271–304. Springer, 2011.

[17] Jörg Steeg and Michael Schröder. A hybrid approach to solve the periodic home health care problem. In *Operations Research Proceedings 2007 - Selected Papers of the Annual International Conference of the German Operations Research Society (GOR)*, pages 297–302, 2007.

[18] Pascal Van Hentenryck and Laurent Michel. *Constraint-Based Local Search*. The MIT Press, 2005.