

A Gossip Algorithm for Home Healthcare Scheduling and Routing Problems^{*}

Sarmad Riazi^{*} Payam Chehrazi^{*} Oskar Wigström^{*}
Kristofer Bengtsson^{**} Bengt Lennartson^{*}

^{*} *Dep. of Signals and Systems, Chalmers University of Technology,
Gothenburg, Sweden; e-mail: cherazi@student.chalmers.se,
{sarmad.riazi, oskar.wigstrom, bengt.lennartson}@chalmers.se.*
^{**} *Sekvensa AB, Gothenburg, Sweden; e-mail: kristofer@sekvensa.se*

Abstract: Many people in need of care still live in their homes, requiring the caretakers to travel to them. Assigning tasks to nurses (or caretakers) and scheduling their work plans is an NP-hard problem, which can be expressed as a vehicle routing problem with time windows (VRPTW) that includes additional problem-specific constraints. In this paper, we propose to solve the Home Healthcare Scheduling and Routing Problem (HHCSP) by a distributed Gossip algorithm. We also apply an extended version called n-Gossip, which provides the required flexibility to balance optimality versus computational speed. We also introduce a relaxation on the optimality of the underlying solver in the Gossip, which speeds up the iterations of the Gossip algorithm, and helps to quickly obtain solutions with good quality.

1. INTRODUCTION

Healthcare services provided at home for patients, who do not require in-house treatment, can often be more suitable (Shyu et al. [2002]), and cost efficient (Ahlner-Elmqvist et al. [2008]) compared to in-house treatment. The demand for home health care solutions is growing according to a study conducted in Woodward et al (2004). The authors suggest a possible explanation in that higher life expectancy in industrial countries has led to a growing number of frail and elderly in need of care. Another study by Ian M. et al (2004), reaches the same conclusion. Also a broader study by the World Health Organization has announced the growth of care-dependent elderly in all of Europe (Tarricone and Tsouros [2008]). Several other studies attribute the growing demand of home healthcare to economic factors and patient preference, (Kergosien et al. [2009]). This paper suggest a scheduling and planning algorithm to aid caretakers in handling this increase in demand.

A problem arises in the scheduling and organization of work plans for caretakers when demand exceeds the number of caretakers available. Manual planning of work schedules is difficult due to the large number of variables involved and the factorial dependence of the problem. Examples of dependencies are when some caretakers are qualified to administer drugs, while others are not, the caretaker method of travelling, time constraints such as earliest to latest time for doing a job, time of administering medication or care, and the time of day a caretaker is available. Furthermore, when there are more patients than caretakers for a specific time slot, patients with important

needs are to be prioritized. These are just some examples of the many constraints needed to be taken into account when scheduling a work plan. The Home Healthcare Scheduling and Routing Problem (HHCSP) concerns the optimization of the route of each caretaker in a way that minimizes an objective specified by the service provider. These objectives may vary, but usually include travel time and customer dissatisfaction.

There are two main approaches to solving the HHCSP. In the first approach, the problem is solved sequentially, i.e. the assignments of caretaker to jobs is solved independently of the routing problem. This has shown to be viable with shorter travel times and fewer caretakers to assign (Yalcindag et al. [2012]). In the second approach, the problems are solved simultaneously, using one single model. In general, the second approach involves defining the problem as a vehicle routing problem with time windows (VRPTW). This method, while having the drawback of increased problem complexity, provides more accurate solutions.

A classical vehicle routing problem (VRP) can be stated as the problem of determining an optimal set of routes for a set of vehicles such that given demands at the customers are satisfied. Each route starts and ends in a given depot. Each customer should be visited by one vehicle and each vehicle has a capacity that cannot be exceeded. The routes are to be selected such that the total cost is minimized (Laporte [1992]). In the time-windowed version of VRP (VRPTW), additional time constraints are present that force the vehicles to visit the tasks within a specified time frame. Solomon [1987] provides a detailed survey on algorithms for VRPTW, and in Solomon and Desrosiers [1988], the authors present a survey on variations of this problem.

In an HHCSP, which essentially is a VRPTW, vehicles are replaced by the caretakers who travel and serve the

^{*} This work was carried out at the Wingquist Laboratory VINN Excellence Center within the Area of Advance – Production at Chalmers, supported by the Swedish Governmental Agency for Innovation Systems (VINNOVA) and the Swedish Research Council. The support is gratefully acknowledged.

healthcare customers. The objective function which should be minimized, may include terms such as for example the total distance traveled by the caretakers, subject to the following healthcare specific constraints: (i) each caretaker has a level of qualification, and each job a level of requirement and caretakers will only be assigned to routes where they are qualified to perform all jobs on that route; (ii) All jobs have durations, earliest possible starting times and deadlines, caretakers have a start and an end to their work day.

In this paper, a mixed integer linear programming formulation of the HHCRSP, based on the VRPTW is derived. This is a simplified theoretical model and includes only a subset of the real-world constraints. While the model works well for small to medium problems, an exact deterministic solution quickly becomes intractable for larger problem instances. This is overcome in this paper by employing a heuristic Gossip method to solve the problem. Such an algorithm has been successfully applied to a class of VRPs known as the heterogeneous multi-vehicle routing problem (HMVRP) (see Franceschelli et al. [2013], and Riazi et al. [2013]). The Gossip algorithm iteratively solves local problems that only consider a subset of caretakers and customers in each iteration. The results of these local optimizations propagate through the entire search space as iterations continue. This is in contrast to a *Centralized* approach, where the entire problem is considered at once.

In the next section, related work to HHCRSP is discussed. In Section 3, the problem formulation is given and the algorithms are presented in Section 4. Besides using the standard Gossip algorithm, which works on nurses/vehicles in a pairwise fashion, an extended version is also used. The extended algorithm, n-Gossip, presented in Riazi et al. [2013], is used to obtain the required flexibility to balance optimality versus computational speed. Finally, Section 5 provides numerical simulations and complexity evaluations of both Gossip algorithms and the Centralized approach.

2. PREVIOUS WORK

Bredström and Rönnqvist [2008] presents a combined scheduling and vehicle routing problem with time windows. The model uses a three part weighted-criterion objective function for modeling: (i) the preference measure for a particular vehicle when visiting a node, (ii) the traveling cost between nodes for a particular vehicle; (iii) the difference in workload. Besides the standard VRPTW constraints, a constraint to express the workload variable included in the third objective term is added to the problem. The workload is defined as the maximal difference in travel time, or service duration for any two vehicles on their respective routes.

In Rasmussen et al. [2012], the VRPTW formulations in Bredström and Rönnqvist [2008] were extended into a HHCRSP formulation and is expanded to include uncovered visits. Also here, weighted terms in the objective function are used. These terms include uncovered visits, caretaker preferences, and travel cost. The main difference from other methods is that during assignment, jobs which fall outside of the given restrictions are considered uncovered. In their paper, the priority of a job is included into the objective function, rather than added as a constraint.

Besides the mentioned additions to the objective function, constraints ensuring that the starting times of any job lies between the start and end time of the work day for any caretaker are included. They solved the problem using a branch and price framework, a two step approach where an initial set of feasible schedules were generated and then combined in the second step to find the minimum cost combination of schedules where all constraints were fulfilled

Eveborn et al. [2006] formulated the problem using a set-partitioning model, and for solution, they utilized a repeated matching algorithm, and they reported a significant reduction in operational planning time. In Bredström and Rönnqvist [2008], the authors have generalized and combined the vehicle routing and scheduling model by including temporal precedence and synchronization constraints. They reported that for home care applications, adding extra constraints had a positive effect on the quality of the schedules, without making it more difficult to find a feasible solution.

Trautsamwieser and Hirsch [2011] have proposed a two step solution for real life scenarios using metaheuristics, based on the Variable Neighborhood Search (VNS). The first step generates initial schedules for the caretakers by assigning the jobs without violating the constraints. The second step is applying VNS, to refine the schedules obtained by the initial solution and find the combination that minimizes the total cost. The benefit of this approach is that the many criteria of the formulation makes the solution more applicable in a real life scenario, e.g. ensuring that breaks follow work regulations. Furthermore, by using soft time window constraints for the majority of jobs and minimizing with regards to time window violations, some slack is allowed in finding a feasible solution. The downside to this approach is that there is no way of finding a global optimum, or even a feasible solution using standard solver software for real life situations (i.e. larger problems), without applying heuristics.

3. PROBLEM FORMULATION

For our mixed integer linear programming model (MILP), we have used the VRPTW model from Kallehauge et al. [2005], and modified it to fit the HHCRSP context. The altered version is also more suitable for the Gossip algorithm, or a partitioning method like Benders decomposition (Hooker and Ottosson [2003]). This is because the modified model has a clear distinction between job assignment variables and route variables.

The problem is defined by a set of n nurses (or caretakers) \mathcal{N} , a set of c customers (or jobs) \mathcal{C} , and a graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$, where \mathcal{V} is the set of nodes, and \mathcal{A} is the set of arcs. All caretakers are employees of Home HealthCare Service, and hence they all start from the same depot, and return to it after completing their jobs. For the sake of convenience, the depot is modeled by two identical nodes in the Graph, one for starting from, and the other for ending to. Therefore, $|\mathcal{V}| = |\mathcal{C}| + 2$, where nodes 0 and $c + 1$ represent the starting and the ending nodes respectively.

The MILP model in Kallehauge et al. [2005] includes the following sets of variables

- \mathcal{X} : The set of binary sequencing variables x_{ijn} , which are defined for arcs of the graph. If $x_{ijn} = 1$, the caretaker n serves client (node) j directly after client i , and therefore the corresponding arc in the graph exists. No arc enters the first depot (node 0), and no arc leaves the last one (node $c + 1$).
- \mathcal{T} : The set of continuous time variables t_{in} defined for each node i and each caretaker n . If client i is to be visited by caretaker n , the value of t_{in} determines when this should happen, otherwise, the value of t_{in} bears no meaning.

We introduce an additional set of variables \mathcal{Y} including binary job assignment variables y_{in} , to facilitate implementing the Gossip algorithm. If caretaker n serves customer i , then y_{in} is one, otherwise, it is zero.

The costs and other constants in the model are as follows:

- Caretaker's time window $\{\bar{a}_n, \bar{b}_n\}$: \bar{a}_n is the starting time for caretaker n and \bar{b}_n is the finishing time for caretaker n .
- Customer's time window $\{\bar{a}_i, \bar{b}_i\}$: \bar{a}_i and \bar{b}_i are the earliest starting time, and the deadline of task i .
- Caretaker's qualification level Q_n for caretaker n .
- Client's qualification requirement r_i for job i .
- Client's job duration \bar{t}_i : The duration of the job i that a caretaker has to perform
- Symmetric cost matrix C_{ij} : This is the Euclidean distance between clients i and j .
- Travel time \bar{t}_{ij} : This is calculated by dividing the euclidean distance between nodes i and j by caretaker's speed.

Hence, the *Centralized* (or complete) MILP is as follows:

$$\min z = \sum_{i \in \mathcal{C}} \sum_{j \in \mathcal{C}} \sum_{n \in \mathcal{N}} C_{ij} x_{ijn} \quad (1)$$

$$\text{s.t.} \quad \sum_{n \in \mathcal{N}} y_{in} = 1 \quad \forall i \in \mathcal{C} \quad (2)$$

$$y_{0n} = y_{(c+1)n} = 1 \quad \forall n \in \mathcal{N} \quad (3)$$

$$\sum_{j \in \mathcal{V}} x_{ijn} = y_{in} \quad \forall i \in \mathcal{C}, \forall n \in \mathcal{N} \quad (4)$$

$$\sum_{j \in \mathcal{V}} x_{ijn} = y_{in} \quad \forall i \in \mathcal{V} \setminus \{0\}, \forall n \in \mathcal{N} \quad (5)$$

$$t_{in} + \bar{t}_{ij} + \bar{t}_i - M_{ij}(1 - x_{ijn}) \leq t_{jn} \quad \forall (i, j) \in \mathcal{A}, \forall n \in \mathcal{N} \quad (6)$$

$$\bar{a}_i \leq t_{in} + \bar{t}_i \leq \bar{b}_i \quad \forall i \in \mathcal{V}, \forall n \in \mathcal{N} \quad (7)$$

$$\bar{a}_n \leq t_{in} \leq \bar{b}_n \quad \forall i \in \mathcal{V}, \forall n \in \mathcal{N} \quad (8)$$

$$r_i y_{in} \leq Q_n \quad \forall i \in \mathcal{C}, \forall n \in \mathcal{N} \quad (9)$$

$$x_{ijn} \in \{0, 1\} \quad \forall i \in \mathcal{V} \setminus \{c + 1\}, j \in \mathcal{V} \setminus \{0\}, \forall n \in \mathcal{N} \quad (10)$$

$$y_{in} \in \{0, 1\} \quad \forall i \in \mathcal{V}, \forall n \in \mathcal{N} \quad (11)$$

$$t_{in} \geq 0 \quad \forall i \in \mathcal{V}, \forall n \in \mathcal{N} \quad (12)$$

The objective function given by (1) minimizes the total travelling distance for all of the caretakers. Constraints in (2) assure that any job is visited once, by only one caretaker. With (3), it is ensured that every caretaker leaves the depot, and comes back to it, whether it does anything or not. By (4) and (5) it is guaranteed that if a job is to be performed by a caretaker, there is an arc (path) entering the corresponding node, as well as another one leaving it. Constraints in (6) state the relationship between the total spent time for a customer, and its immediate successor. Notice that the large constant M_{ij} can be reduced to $\max\{\bar{b}_i + \bar{t}_{ij} - \bar{a}_j\}$, $(i, j) \in \mathcal{A}$. Furthermore, (7) affirms that a job is performed within its time window, while (8) assures that any caretaker works within the specified work hours. Constraints in (9) assures that only a qualified enough caretaker is assigned to any job. Finally, (10) through (12) define the variable domains.

An important remark on the problem formulation is the absence of the notorious subtour elimination constraints (SECs). In the classical VRP, SECs guarantee that no tour not including the depot node exists in the solution. SECs induce combinatorial explosion, and they cannot be handled efficiently by a general branching scheme (see Bektas [2006] for more information on SECs). In a time windowed version of VRP, like HHCRSP, the service start variables t_{in} in (6) impose a unique route direction that necessarily includes the depot, hence, SECs become redundant. (see more on Solomon and Desrosiers [1988]).

4. THE ALGORITHMS

We have implemented a number of algorithms to solve the HHCRSP: A *Centralized* method, which solves the whole block of the constraints and objective function using a MILP solver; A logic-based Benders decomposition, which breaks down the HHCRSP into a master problem that assigns tasks to the caretakers, and a set of subproblems that schedule the assigned tasks for the caretaker; A standard *Gossip* algorithm that solves the problem locally and works on pairs of caretakers; An extended Gossip, or *n-Gossip* that includes more than two caretakers in the local problems, and finally, a relaxed gossip that relaxes the optimality condition on the subproblems to speed up the search.

4.1 The Centralized Method

The results of the Centralized method was used as a reference for comparing the results of the other algorithms. Riazi et al. [2013] reported that the Centralized method was ineffective for problems with 14 tasks and beyond, mainly due to explosion of the sub-tour elimination constraints (SECs). However, in HHCRSP, we were able to solve instances as large as 35 tasks within a time limit of 2 hours. This is both because the time-window constraints in (6) removed SECs, and also because the time-window constraint significantly reduced the search space as solutions could become infeasible because of timing concerns.

4.2 The Logic-based Benders Decomposition

Logic-based Benders decomposition is a partitioning technique that decomposes a problem into a master problem,

and one (or more) subproblems, such that the subproblems become easier to solve, or they take form of a known problem for which efficient algorithms exist. Applying this method to a problem like HHCRSP results in a master problem that handles the tasks assignment, and a series of subproblems (one for each caretaker) that schedule the assigned tasks on their corresponding caretaker.

In Riazi et al. [2013], Logic-based Benders decomposition was used for the HMVRP to decompose the problem into a task assignment problem (master), and a series of scheduling subproblems in form of traveling salesman problems (TSPs). Integrating the Benders algorithm with a TSP solver for handling the subproblems led to significant speed ups that made the Benders method a viable alternative compared to the Centralized one. In the case of HHCRSP, however, we found out that Benders algorithm was many times slower than the Centralized method, for any problem instance that we tested. One reason could be the fact that in HMVRP, the main computational burden was to handle the huge number of SECs that paralyzed the MILP solver. In contrast, the use of a dedicated TSP solver in Benders method greatly alleviated this issue, and resulted in a more successful method compared to the Centralized one.

Another issue affecting the Benders method in both HMVRP and HHCRSP was that the master problem had difficulty finding new incumbent solutions. The reason was that the implemented Benders' cuts were not strong enough to rule out large portions of the huge search space. We attempted to speed up the Benders by generating stronger infeasibility cuts. To do this, a relaxed Constraint Programming (CP) version of each subproblem was constructed. Then, CP methods were used to detect the minimal conflicting set of tasks on a caretaker's schedule. Generating the stronger infeasibility cuts was effective, but it was not enough to make the Benders algorithm a better alternative to the Centralized method.

4.3 The Gossip Algorithm

The Gossip algorithm for HMVRP was first proposed in Franceschelli et al. [2013], where instead of caretaker and customers, the context involved robots and tasks. According to the Gossip rule, after an initial task assignment, two robots and their tasks are randomly picked to form a local optimization problem. The solution to this problem yields either an equal or a better objective function value by retaining or changing the task assignments among the robots. Then, the same process is repeated for another randomly chosen couple of robots, until no further improvement is achieved. Notice that the method has no specific criterion for stopping condition.

The important rationale behind the use of Gossip for HHCRSP is that it allows us to solve local problems that are smaller in number of both caretakers, and customers. For instance, consider a case of 4 caretakers and 36 tasks. If the algorithm starts with an initial assignment of 9 tasks for each customer, the first iteration involves solving a problem of 2 caretakers and 18 clients which is easier to solve and is handled more efficiently by a local solver.

4.4 The n-Gossip Algorithm

The n-Gossip algorithm was introduced in Riazi et al. [2013]. In n-Gossip, more than two robots (or caretakers) are involved in the local optimizations. In fact, for $|\mathcal{N}|$ robots, n can grow up to $|\mathcal{N}| - 1$. This implies higher algorithmic complexity, but the benefit is the higher quality of the solution, because larger local problems give a better approximation of the original problem. Therefore, one can decide to increase the solution quality by increasing n such that the CPU time is still smaller than an exact method. According to Riazi et al., increasing the n to its maximum value resulted in significant reduction of the optimality gap in most of the tested problem instances, while the solution time remained considerably less than their fastest exact method (Benders with TSP solver).

4.5 The Relaxed Gossip Algorithm

The performance of the Gossip algorithm is very dependant on that of its local solver. When the problem size grows large, the local MILP solver starts losing its efficiency. To overcome this issue, we relax the optimality requirement and let the solver terminate the search when the optimality gap reaches 5%. Moreover, if the time spent on a local problem exceeds a certain threshold, the search is halted and the best incumbent solution is returned. This method proved to alleviate the bottlenecking issues arisen in difficult local problems.

5. NUMERICAL SIMULATIONS

The aim of the numerical experiments is to examine the behavior of the 3 versions of the Gossip algorithm; standard, n-Gossip, and the relaxed one. The Centralized method was used to provide the optimal values of the test instances, which was used as a measure of the solution quality and computational efficiency of the Gossip methods. The MILP solver used was IBM ILOG CPLEX 12.5 (64-bit) in Microsoft Windows 7 Enterprise environment. The hardware used was an Intel Core2 Quad CPU (2.66 GHz), with 4 GBs of RAM.

5.1 The Test Scenarios

We have evaluated the Centralized and Gossip methods for different configurations of caretakers and jobs, and divided the experiments into two sets. For the first set, the number of caretakers $|\mathcal{N}|$ was set to 3, 4, and 5, while the number of jobs $|\mathcal{C}|$ was set to 20, 25, and 30 respectively. For each of these combinations, three instances were generated randomly, and each instance was solved 10 times for the Gossip method to obtain a better picture of its average performance in terms of CPU time and gap. In the second set of the experiments, $|\mathcal{N}|$ was 6, 7 and 8, with $|\mathcal{C}|$ being 40, 50, and 60. Each combination of the second set was tested for two random instances. The randomly generated costs were based on real-life conditions as seen in Eveborn et al. [2006]. For example, $\{a_n, b_n\}$ was chosen to be $\{8, 16\}$ to reflect the working hours of a typical caretaker. The durations of tasks in the first set are long (10 to 45 minutes), while in the second set the durations are short (10 to 20 minutes). By using the short version we meant

to fill-up the schedule of each caretaker with more tasks. For the first part of the experiments, we selected instances from the Centralized method with solution time of less than two hours, and for the Gossip, those instances were solved subject to a time-limit of 1200 [s]. For the second part of the tests, the best integer solutions found from the Centralized and Gossip were reported after spending, in turn, two hours, and 1200 [s] on each of the algorithms.

5.2 The Standard Gossip

In the standard Gossip, local problems include only two caretakers, which is denoted by $LP = 2$. To measure the quality of the Gossip solution, an artificial gap is defined as

$$\text{Gap } \% = \frac{z_{Gossip}^* - z_{Centralized}^*}{z_{Centralized}^*} * 100$$

where z_{Gossip}^* is the best solution obtained from Gossip, and $z_{Centralized}^*$ is the optimal solution obtained from the Centralized method. The solution time for the Gossip is the time at which z_{Gossip}^* first happened.

In Table 1, the *Centralized* and *Gossip (LP=2)* columns present the result of the numerical experiments. As seen in the table, for the tested instances, the standard Gossip is generally many times faster than the Centralized, and the gap varies from 0.02% to 7.02%. Note that the reported time and gap for each instance are the average of 10 experiments. Note also that in the first iteration of the Gossip algorithms, the tasks were distributed among the caretakers such that the workload of the caretakers would be almost the same, while the qualification requirement of the assigned tasks were smaller than or equal to the qualification level of the corresponding caretaker.

Distributing the tasks in a load-balancing fashion in some cases resulted in infeasible schedules in the first iteration. In several cases, the algorithm was not able to find a feasible solution within the specified time limit. For example, for the case of $(|\mathcal{N}|, |\mathcal{C}|, i) = (5, 30, 3)$, one of the 10 experiments suffered from this issue. As we will see in the next section, for larger problems, it is cost-effective to spend some time on finding a feasible first assignment. However, as we tested, for smaller problems, finding a feasible initial solution sometimes resulted in worse computational performance.

5.3 The n-Gossip Method

As stated in Riazi et al. [2013], the purpose of the n-Gossip algorithm is to provide the means to balance solution quality versus computational speed. This trade-off is achieved by including more than two caretakers in the local problems, which improves the solution quality at the cost of computational speed.

Table 1 (columns *Gossip (LP=3)*, *Gossip (LP=4)*) provides the performance analyses of the n-Gossip method for the long task duration. By moving from left ($LP=2$) to the right ($LP=4$), we observe that in most of the instances the solution quality improves (the gap decreases), and the CPU time worsens. For the case of $(|\mathcal{N}|, |\mathcal{C}|, i, LP) =$

$(5, 30, 1, 4)$, we can see that the gap is completely eliminated, but the solution time is even worse than the Centralized. Hence, depending on the problem instance, it may not be wise to increase n in n-Gossip method to its maximum value $(|\mathcal{N}| - 1)$, since the Centralized may then yield the optimal solution more quickly. Nevertheless, we observe that for most of the instances, the n-Gossip method yields solutions with better quality than standard Gossip, and less CPU time than the Centralized method. One issue with the n-Gossip is that if the initial step of the algorithm starts with an infeasible assignment, the algorithm sometimes cannot find a feasible solution before the time-limit, as was the case for standard Gossip. However, due to the higher computational complexity of n-Gossip, it is more prone to the mentioned problem. The remedy, as it will come, is to start the algorithm with a feasible solution.

Another problem that generally arises with both versions of Gossip, is that the underlying CPLEX solver spends too much time on proving the optimality of the local problem. This can be alleviated by adjusting the optimality tolerance of the subproblems, without compromising the solution quality significantly.

5.4 Gossip With Relaxed Optimality Tolerance

The Centralized method was not able to yield optimal solution for problems including 40 nodes or beyond, yet it was used to provide upper bounds on the optimal objective function value. Such bounds could be compared to those found by the Gossip algorithm, as a measure of quality. The relaxed Gossip was started with feasible solutions obtained from CPLEX using heuristics, and it was halted after 20 minutes. To perform numerical experiments for this section, the tasks were generated with shorter durations (10 to 20 minutes) such that more tasks could be assigned to each set of caretakers. Table 2 gives the simulation results of this part of the experiments.

It is seen from the table that, the relaxed Gossip yields significantly better integer solutions compared to the upper bounds obtained from the Centralized method. Note that the Gossip algorithm was subject to a time limit of 1200[s] (20 minutes), and the Centralized to two hours. We observed that relaxation strategy in Gossip effectively prevented the algorithm from spending too much time on difficult local problems. Moreover, starting the Gossip with any feasible solution removed the problem of unsolved instances reported in Table 1.

6. CONCLUSION

The Home Healthcare Scheduling and Routing problem could be modeled a variation of the vehicle routing problem with time windows. We have applied standard Gossip, and n-Gossip algorithms to solve this problem. The standard Gossip yields quick solutions with reasonable quality, while the n-Gossip offers a trade-off between optimality and computational complexity. The solution quality of both algorithms can further be improved by initializing them with a feasible solution, and by adjusting the optimality tolerance of the Gossip local problems. This adjustment will allow the Gossip to quickly find integer solutions

size			Centralized	Gossip (LP=2)			Gossip (LP=3)			Gossip (LP=4)		
$ \mathcal{N} $	$ \mathcal{C} $	i	Seconds	Seconds	Gap (%)	#unsolved	Seconds	Gap (%)	#unsolved	Seconds	Gap (%)	#unsolved
3	20	1	8.68	1.44	0.02	0						
		2	3.94	0.84	1.60	0						
		3	1.90	2.58	0.00	0						
4	25	1	107.84	8.39	4.70	0	40.52	0.80	0			
		2	123.82	3.22	2.61	0	34.46	0.00	2			
		3	24.04	2.65	0.00	0	15.08	1.81	0			
5	30	1	99.88	14.18	2.04	0	148.37	1.06	0	132.99	0	0
		2	401.19	151.11	7.02	0	240.40	5.85	4	345.72	4.93	2
		3	2923.25	120.29	3.31	1	266.99	0.45	1	335.99	3.12	2

Table 1. Comparison of Centralized and Gossip methods: Average CPU time in seconds, Average gap, and number of unsolved instances (#unsolved); $|\mathcal{N}|$, $|\mathcal{C}|$, and i indicate number of caretakers, jobs, and the instance number; LP=2, LP=3, and LP=4 mean the number of caretakers in the local problem.

size			Centralized	Gossip (LP=2)
\mathcal{N}	$ \mathcal{C} $	i	Best integer solution	Best integer solution
6	40	1	47.44	41.66
		2	45.11	43.19
7	50	1	53.88	48.01
		2	72.88	49.27
8	60	1	83.67	61.98
		2	81.97	70.36

Table 2. Comparison of Centralized and relaxed Gossip: Best integer solution after two hours (Centralized), and after 20 minutes (Gossip); $|\mathcal{N}|$, $|\mathcal{C}|$, and i indicate number of caretakers, jobs, and the instance number.

with good quality compared to those obtained from the Centralized method.

REFERENCES

- Marianne Ahlner-Elmqvist, Marit S. Jordhy, Kristin Bjordal, Magnus Jannert, and Stein Kaasa. Characteristics and quality of life of patients who choose home care at the end of life. *Journal of pain and symptom management*, 36(3):217–227, 2008. doi: 10.1016/j.jpainsymman.2007.10.010.
- Tolga Bektas. The multiple traveling salesman problem: an overview of formulations and solution procedures. *Omega*, 34(3):209–219, June 2006.
- David Bredström and Mikael Rönnqvist. Combined vehicle routing and scheduling with temporal precedence and synchronization constraints. *European Journal of Operational Research*, 191(1):19–31, 2008. doi: http://dx.doi.org/10.1016/j.ejor.2007.07.033.
- Patrik Eveborn, Patrik Flisberg, and Mikael Rönnqvist. Laps carean operational system for staff planning of home care. *European Journal of Operational Research*, 171(3):962–976, 2006.
- Mauro Franceschelli, Daniele Rosa, Carla Seatzu, and Francesco Bullo. Gossip algorithms for heterogeneous multi-vehicle routing problems. *Nonlinear Analysis: Hybrid Systems*, 2013. http://dx.doi.org/10.1016/j.nahs.2013.03.001.
- J.N. Hooker and G. Ottosson. Logic-based benders decomposition. *Math. Program.*, 96:33–60, 2003.
- Brian Kallehauge, Jesper Larsen, Oli B.G. Madsen, and Marius M. Solomon. Vehicle routing problem with time windows. In Guy Desaulniers, Jacques Desrosiers, and Marius M. Solomon, editors, *Column generation*, pages 67–99. Springer, 2005.
- Yannick Kergosien, Christophe Lent, and Jean-Charles Billaut. Home health care problem: An extended multiple traveling salesman problem. *Multidisciplinary International Conference on Scheduling : Theory and Applications (MISTA)*, August 2009.
- Gilbert Laporte. The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59(3):345–358, 1992.
- Matias Sevel Rasmussen, Tor Justesen, Anders Dohn, and Jesper Larsen. The home care crew scheduling problem: Preference-based visit clustering and temporal dependencies. *European Journal of Operational Research*, 219(3):598–610, 2012.
- Sarmad Riazi, Carla Seatzu, Oskar Wigstrom, and Bengt Lennartson. Benders/gossip methods for heterogeneous multi-vehicle routing problems. In *Emerging Technologies Factory Automation (ETFA), 2013 IEEE 18th Conference on*, pages 1–6, 2013. doi: 10.1109/ETFA.2013.6647983.
- Lotus Shyu, Yea-Ing, and Hsiao-Chin Lee. Predictors of nursing home placement and home nursing services utilization by elderly patients after hospital discharge in taiwan. *Journal of Advanced Nursing*, 2002.
- Marius M. Solomon. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35:254–265, 1987.
- Marius M. Solomon and Jacques Desrosiers. Time window constrained routing and scheduling problems. *Transportation Science*, 22(1):962–976, 1988.
- Rosanna Tarricone and Agis D. Tsouros. Home care in europe: the solid facts. 2008. ISSN 978 92 890 4281 9.
- Andrea Trautsamwieser and Patrick Hirsch. Optimization of daily scheduling for home health care services. *J Appl Oper Res*, 3(3):124–136, 2011.
- S. Yalcindag, A. Matta, and E. Sahin. Operator assignment and routing problems in home health care services. In *Automation Science and Engineering (CASE), 2012 IEEE International Conference on*, pages 329–334, 2012. doi: 10.1109/CoASE.2012.6386478.