Appendix of: Metaheuristics for Solving a Multimodal Home-Healthcare Scheduling Problem*

Gerhard Hiermann, Matthias Prandtstetter, Andrea Rendl, Jakob Puchinger, and Günther R. Raidl

[1] M. Prandtstetter, A. Rendl, J. Puchinger
AIT Austrian Institute of Technology GmbH
Mobility Department, Dynamic Transportation Systems
Giefinggasse 2, 1210 Vienna, Austria
{matthias.prandtstetter | andrea.rendl | jakob.puchinger}@ait.ac.at

[2] G. Hiermann, G. R. Raidl
Institute of Computer Graphics and Algorithms
Vienna University of Technology
Favoritenstraße 9–11/186, 1040 Vienna, Austria
raidl@ads.tuwien.ac.at

A Features of the real-world instances

As we are not allowed to make our real-world instances public available due to legal issues, this section provides an overview of the specific features of our instances. Table 1 summarizes the properties of the instances used in our experiments in our work [1].

Time values shown in this table represents one time unit of our problem formulation. In our case, one unit represents five minutes, where numbers in between are rounded down to the previous time unit (e.g., 9 minutes = 1 time unit). Average values and deviations for all instances are the average sum of the individual values.

B Random MHC problem instances

For reproducibility purposes, we generated additional instances using distributions extracted from the real world settings. These instances can be found at https://www.ads.tuwien.ac.at/w/Research/Problem_Instances.

B.1 Generation of random instances

We derived distributions for each feature of real-world instances, and created four random instances (random01, random02, random03 and random04). To simulate

* This work is partially funded by the Austrian Federal Ministry of Transport, Innovation and Technology (BMVIT) within the strategic programme I2VSplus under grant 826153 (CareLog).

the spatial distribution of the real-world, we generated uniform distributed GPS locations in the region of Vienna.

To generate meaningful distance matrices, we use the data from OpenStreetMap (`http://www.openstreetmap.org/`) to calculate the travel times using a car. To simulate the closed-source public transport travel times we decided to use the data from OpenStreetMap as well, but using the times calculated for bike routes. As bikes can often use similar routes as public transports (e.g, bike lanes on bridges for train tracks), the use of these times seemed appropriate for our needs.

To both modes of transport we added a fixed amount of additional time. For the car distances, we added ten minutes to each distance, seven for finding a parking slot and for walking to the destination, as well as three minutes to get back to the car after finishing. For the public transport we first scale the travel time by 1.1 to count for the additional relative waiting time for all switches of transports (e.g, bus to train) and added additional five minutes as a constant waiting time. These values have been found suitable to approximate the real-world travel times for both modes of transport.

Further changes have been made concerning the shifts of nurses and the fixed jobs: As the number of fixed jobs is relative low (around 20 per day out of approx. 700 jobs), we decided not to generate these in our new instances. Furthermore, we omitted the generation of multiple shifts per day - we found only multiple shifts of maximal two - to generalize the problem such that contract specific properties of the objective function (e.g, payment of the traveltime from/to the home location based on the shift durations and more) are not considered.

Tables 2-5 provide the distributions of the random generated instances in the same form as in the previous Section A.

B.2 Experiments

To show how the approaches from [1] perform on these random generated instances, we run them ten times per instance and per initial solution generation with the same settings used in [1]. The results are presented in Table 6 These show that the overall performance is similar to the one described in our work. Again, the Memetic Algorithm (MA) outperforms the other approaches, followed by the Variable Neighbourhood Search (VNS), which performs better than the Simulated Annealing Hyper Heuristic (SAHH) and Scatter Search (SS) for these instances. The SS performs worst, especially for the first instance (random01) for which it could find no improvement at all when initialized with an CP solution.

B.3 Instance format

For each instance we create three separate files: one is the configuration file (with the extension .conf), holding the general information of the problem instance. The file format used for the configuration of a hierarchical structure based on the .ini file format frequently used in windows-based systems. The information is stored using following pattern as shown in Table 7. The first section (i.e.,

[normalizations]) holds the corresponding γ_i weights for the objective value calculation (see [1]), followed by sections describing nurses, jobs and customers.

The other two files hold the distances between every nurse and customer for both transportation modes. The format of the files are simple comma separated values representing a matrix, where the first line holds the identifier for the column (e.g., nurse-03, customer-12, ...), followed by lines starting with an identifier for the row and distance values. A cell in this matrix holds the travel distance from the location of the row-identifier to the column-identifier.

Both distance files starts with the name of the instance (e.g., random01) followed by '.CAR.dm', '.PUBLIC.dm' respectively. The entries in these two distance matrices are already normalized to the five minute interval and used as-is in our approaches.

Acknowledgments This work is part of the project CareLog, partially funded by the Austrian Federal Ministry for Transport, Innovation and Technology (BMVIT) within the strategic programme I2VSplus under grant 826153. We thankfully acknowledge the CareLog project partners Verkehrsverbund Ost-Region GmbH (ITS Vienna Region), Sozial Global AG, and ilogs mobile software GmbH. We also thank the reviewers of our revised work [1] for their helpful comments.

References

1. Hiermann, G., Prandtstetter, M., Rendl, A., Puchinger, J., Raidl, G.R.: Metaheuristics for solving a multimodal home-healthcare scheduling problem. In Revision at: Central European Journal of Operations Research

Table 1: Properties and distributions over all real-world instances

# of nurses (total)		4000
# of jobs (total)		5333
# of customers (total)		3342
		csw / vn / hn / ahn / mn
Qualification of nurses		40 / 178 / 3106 / 333 / 343
Qualification required for jobs		1 / 272 / 4666 / 353 / 41
		female / male / smoker / dog / cat
Aspects of nurses		3625 / 375 / 0 / 104 / 73
Aspects unwanted by customers		4 / 482 / 0 / 67 / 156
Job Duration (min)		3
(max)		33
(avg)		10.85
(dev)		5.31
		car / public
Preferred MOT		1012 / 2988
Job startTW Size (min)		0
(max)		77
(avg)		40.48
(dev)		18.93
Desired starttime position (relative) (avg)		0.3357
(dev)		0.2404
Shift [start,end]	[72,276]	2475
	[72,(72,95)]	38
	[72,96]	1022
	[72,(96,192)]	50
	[72,192]	211
	[[108,162),276]	255
	[162,276]	183
	[168,276]	368
	[174,276]	176
	[(174,240),276]	160
	[240,276]	204
	[(240,264),276]	9
	[264,276]	167

Table 2: Properties and distributions of instance random01

# of nurses (total)		499
# of jobs (total)		744
# of customers (total)		420
		csw / vn / hn / ahn / mn
Qualification of nurses		4 / 23 / 384 / 37 / 51
Qualification required for jobs		1 / 34 / 652 / 49 / 8
		female / male / smoker / dog / cat
Aspects of nurses		449 / 50 / 0 / 16 / 873
Aspects unwanted by customers		1 / 61 / 0 / 13 / 18
Job Duration (min)		3
(max)		30
(avg)		10.73
(dev)		4.94
		car / public
Preferred MOT		109 / 390
Job startTW Size (min)		5
(max)		77
(avg)		39.36
(dev)		18.05
Desired starttime position (relative) (avg)		0.3342
(dev)		0.2231
Shift [start,end]	[72,276]	221
	[72,(72,95)]	0
	[72,96]	9
	[72,(96,192)]	101
	[72,192]	20
	[[108,162),276]	19
	[162,276]	23
	[168,276]	34
	[174,276]	20
	[(174,240),276]	13
	[240,276]	17
	[(240,264),276]	1
	[264,276]	21

Table 3: Properties and distributions of instance random02

# of nurses (total)		518
# of jobs (total)		669
# of customers (total)		420
		csw / vn / hn / ahn / mn
Qualification of nurses		2 / 27 / 399 / 41 / 49
Qualification required for jobs		0 / 36 / 581 / 50 / 2
		female / male / smoker / dog / cat
Aspects of nurses		465 / 53 / 0 / 13 / 11
Aspects unwanted by customers		0 / 71 / 0 / 6 / 18
Job Duration (min)		3
(max)		25
(avg)		10.55
(dev)		5.17
		car / public
Preferred MOT		145 / 373
Job startTW Size (min)		5
(max)		77
(avg)		39.51
(dev)		18.22
Desired starttime position (relative) (avg)		0.3319
(dev)		0.2233
Shift [start,end]	[72,276]	238
	[72,(72,95)]	0
	[72,96]	5
	[72,(96,192)]	103
	[72,192]	22
	[[108,162),276]	23
	[162,276]	19
	[168,276]	35
	[174,276]	27
	[(174,240),276]	17
	[240,276]	18
	[(240,264),276]	2
	[264,276]	9

Table 4: Properties and distributions of instance random03

# of nurses (total)		493
# of jobs (total)		742
# of customers (total)		424
		csw / vn / hn / ahn / mn
Qualification of nurses		5 / 24 / 382 / 39 / 43
Qualification required for jobs		0 / 28 / 667 / 44 / 3
		female / male / smoker / dog / cat
Aspects of nurses		438 / 55 / 0 / 10 / 7
Aspects unwanted by customers		0 / 71 / 0 / 10 / 16
Job Duration (min)		3
(max)		27
(avg)		10.77
(dev)		5.03
		car / public
Preferred MOT		113 / 380
Job startTW Size (min)		5
(max)		77
(avg)		39.86
(dev)		17.80
Desired starttime position (relative) (avg)		0.3253
(dev)		0.2215
Shift [start,end]	[72,276]	233
	[72,(72,95)]	0
	[72,96]	6
	[72,(96,192)]	93
	[72,192]	22
	[[108,162),276]	19
	[162,276]	15
	[168,276]	40
	[174,276]	19
	[(174,240),276]	11
	[240,276]	16
	[(240,264),276]	4
	[264,276]	15

Table 5: Properties and distributions of instance random04

# of nurses (total)		500
# of jobs (total)		744
# of customers (total)		419
		csw / vn / hn / ahn / mn
Qualification of nurses		5 / 30 / 375 / 43 / 56
Qualification required for jobs		0 / 41 / 644 / 54 / 5
		female / male / smoker / dog / cat
Aspects of nurses		465 / 44 / 0 / 15 / 7
Aspects unwanted by customers		1 / 65 / 0 / 14 / 18
Job Duration (min)		3
(max)		29
(avg)		10.14
(dev)		4.94
		car / public
Preferred MOT		117 / 392
Job startTW Size (min)		5
(max)		77
(avg)		39.97
(dev)		17.51
Desired starttime position (relative) (avg)		0.3186
(dev)		0.2219
Shift [start,end]	[72,276]	245
	[72,(72,95)]	0
	[72,96]	8
	[72,(96,192)]	100
	[72,192]	13
	[[108,162),276]	19
	[162,276]	13
	[168,276]	43
	[174,276]	26
	[(174,240),276]	11
	[240,276]	18
	[(240,264),276]	2
	[264,276]	11

Table 6: Final objective values for each metaheuristic, averaged over 10 runs.

	constr.	init.	VNS	MA	SAHH	SS
random01	cp	0.1085	0.04343 ±0.00155	**0.04148** ±0.00075	0.04589 ±0.00117	0.10853 ±**0.00000**
	rand.	246.1894	0.04434 ±0.90143	**0.04199** ±**0.00063**	0.04652 ±0.00101	0.06028 ±1.50045
random02	cp	0.0984	0.03993 ±0.00185	**0.03649** ±**0.00092**	0.04141 ±0.00168	0.05416 ±0.03808
	rand.	222.1736	0.04049 ±0.00096	**0.03676** ±0.00081	0.04073 ±0.00212	0.05384 ±**0.00048**
random03	cp	0.1059	0.04206 ±0.00193	**0.03995** ±**0.00083**	0.04477 ±0.00162	0.06105 ±0.02750
	rand.	222.1874	0.04304 ±**0.00080**	**0.03923** ±0.00119	0.04503 ±0.00112	0.05848 ±2.70338
random04	cp	0.1053	0.04131 ±**0.00114**	**0.03806** ±0.00150	0.04293 ±0.00135	0.05893 ±0.03298
	rand.	247.1836	0.04139 ±**0.00114**	**0.03798** ±0.00155	0.04306 ±0.00138	0.05823 ±0.90252

Table 7: File format of the instances generated.

[normalizations]	
tw_max_dev_in_time	= **number**
tw_max_dev_in_desired_time	= **number**
working_time	= **number**
overtime	= **number**
travel_time	= **number**
aspects	= **number**
qualification	= **number**
[nurse-**number**]	
qualification	= **(CSW|VN|HN|AHN|MN)**
home	= **longitude,latitude**
timewindow	= **[number, number]**
mode	= **(car|public)**
aspects	= **(female|male|smoker|dog|cat)***
⋮	
[customer-**number**]	
home	= **longitude,latitude**
mode	= (job-**number**)+
aspects	= **(female|male|smoker|dog|cat)***
⋮	
[job-**number**]	
timewindow	= **[number, number]**
duration	= **number**
preferredStartTime	= **number**
qualification	= **(CSW|VN|HN|AHN|MN)**
customer	= customer-**number**
⋮	