Improved MILP Formulation for Home Healthcare Scheduling and Routing with Multiple Depots

Ifeoma Ogbodo[1][a], Wasakorn Laesanklang[2][b] and Dario Landa-Silva[1][c]
[1]*School of Computer Science, University of Nottingham, Nottingham, United Kingdom*
[2]*Department of Mathematics, Faculty of Science, Mahidol University, Bangkok, Thailand*
{Ifeoma.ogbodo, Dario.landasilva}@nottingham.ac.uk, wasakorn.lae@mahidol.edu

Keywords: Home Healthcare, Scheduling and Routing, Mixed-Integer Linear Programming, Operational Feasibility, Constraint Validation, Multi-Depot Routing.

Abstract: Despite extensive research on home healthcare scheduling and routing problems (HHCSRP), a critical gap persists between mathematically optimal solutions and operationally feasible implementations. This paper demonstrates that constraints often considered redundant in vehicle routing formulations are essential for HHCSRP, where worker-depot assignments cannot be arbitrarily changed. We validate a widely cited multi-depot HHCSRP MILP formulation using 42 real-world instances, revealing that 77.8% of optimal solutions contain operational violations, workers incorrectly assigned to arbitrary depots and unproductive direct depot-to-depot routes without patient visits. Our main contribution is a refined formulation with explicit operational feasibility constraints that eliminate these violations, while improving computational efficiency on average by 40%. Comparative analysis using GUROBI and CPLEX solvers reveals instance-dependent performance patterns, with GUROBI achieving faster solving times for small to medium resource-constrained instances, and CPLEX producing superior solutions for large-scale, over-resourced problems. These findings underscore that operational validation must extend beyond standard optimisation metrics to verify real-world practicability, a persistent gap contributing to the scarcity of successful HHCSRP deployments in practice.

1 INTRODUCTION

Home healthcare (HHC) has grown globally due to ageing populations, rising chronic diseases, and cost-reduction pressures on hospitals, necessitating efficient operational methods for healthcare workforce management. Home healthcare scheduling and routing problems (HHCSRP) involve assigning workers to visit geographically dispersed patients while satisfying constraints, including time windows, regulatory compliance, skill requirements, and service task restrictions. Mixed-Integer Linear Programming (MILP) remains the dominant modelling approach for HHCSRP, offering exact solution guarantees and flexible constraint modelling (Euchi et al., 2022). However, models can converge to optimal solutions according to their objective function and constraints, yet deliver unimplementable schedules when constraints are inadequate for capturing real-world requirements. An operationally infeasible solution could have workers using depots other than their own and having a depot-to-depot route with no visited patients, despite the solution being deemed feasible and even optimal with respect to the modelled constraints and objective function.

This paper examines a MILP formulation for multi-depot HHCSRP by Laesanklang et al. (2015). Computational validation reveals that 77.8% of optimal solutions violate operational requirements, such that workers utilise incorrect depots and perform empty depot-to-depot trips without patient visits. These violations represent operational failures for the practical implementation of the mathematically feasible solutions.

We contribute: (1) a systematic analysis demonstrating that formulation gaps can produce operationally infeasible solutions despite mathematical optimality, (2) a refined formulation

[a] https://orcid.org/0009-0002-5406-4467
[b] https://orcid.org/0000-0003-4203-3452
[c] https://orcid.org/0000-0002-9499-6827

with explicit feasibility constraints eliminating violations with faster convergence (40% average reduction in computation time) and minimal cost impact (3.1% average objective function increase), and (3) a computational analysis comparing CPLEX and GUROBI revealing solver-specific performance patterns depending on problem scale and worker-to-task ratios.

Section 2 reviews HHCSRP formulations and constraint validation. Section 3 presents the problem, original model, identified issues and refinements. Section 4 analyses experimental results and solver performance. Section 5 concludes the paper and outlines future research directions.

2 LITERATURE REVIEW

Mathematical programming has been extensively applied to solve the HHCSRP. Early MILP models established foundational formulations minimising travel time and time-related costs, while addressing time windows and worker classifications (Begur et al., 1997; Cheng & Rich, 1998). Subsequent research addresses increasingly complex realities such as temporal dependencies and synchronisation (Bredström & Rönnqvist, 2008), preference (Rasmussen et al., 2012), depot assignment and working area constraints (Laesanklang et al., 2015), and skill-based assignments (Zhang et al., 2023). Recent advances extend to flexible departures (Qiu et al., 2025), stochastic events and dynamic scheduling (Du & Zhang, 2022; Khorasanian et al., 2024). These models demonstrated the indispensability of MILP models in defining rigorous problem structures and their versatility for capturing healthcare-specific requirements beyond routing constraints.

A significant evolution has been the shift from single-depot to multi-depot structures, reflecting real-world complexity (Yazır et al., 2023). Unlike classical vehicle routing problems (VRP), where vehicles can be dispatched from any depot, HHCSRP requires worker-specific depot assignments based on home bases or designated offices (Decerle et al., 2021; Nabavizadeh et al., 2024). This distinction is critical because workers in real-world operational scenarios might not be permitted to change bases arbitrarily, and deploying workers without assigned tasks (considered unused vehicles in VRP) wastes resources and incurs unnecessary costs. Moreover, some multi-depot VRPs assume these operational requirements to be redundant (Lalla-Ruiz & Mes, 2021). However, in healthcare situations, where there is a fundamental variation from classical routing problems, these assumptions may not hold.

Despite extensive HHCSRP research, systematic validation of constraint adequacy remains limited. Studies from Harvey (1970) to Grieco et al. (2020) have reported persistent gaps between mathematical optimality and practical implementation, often due to inadequate problem formulation or overlooking operational realities. This paper addresses this gap by providing empirical evidence that operational validation must extend beyond model feasibility to verify explicit constraints and examine the effects on problem structure and resource ratios.

3 PROBLEM DESCRIPTION AND FORMULATION

3.1 Formal Problem Description

The multi-depot HHCSRP assigns health workers to visit patients at their homes while satisfying resource, temporal, regional, and operational constraints. The problem is defined on a directed graph $G = (V, E)$ where $V = D \cup T \cup D'$ represents departure depots (workers' homes), task locations, and arrival depots (copies of workers' homes), respectively. For clarity, $V^d = D \cup T$ defines the set of all departure locations and $V^a = T \cup D'$ the set of all arrival locations.

Each worker, $k \in K$, is characterised by skill qualifications $\mu_{k,s}$ (1 if worker k has skill, $s \in S$, to fulfil the task $j \in T$, and 0 otherwise, S is the set of skills), work schedule availability given by a time window with start and end times (α_L^k, α_U^k), regional availabilities γ_j^k (1 if worker k is available in the region of task j, 0 otherwise) and designated departure/arrival depots $(D_k \in D, D'_k \in D')$. Each task $j \in T$ has a lower and upper time window (w_j^L, w_j^U), service duration δ_j, skill requirements $\eta_{s,j}$ (1 if a task, j requires skill, s), and required visits r_j. The primary decision variable $x_{i,j}^k$ equals 1 if worker k travels from location i to j, and 0 otherwise. Additional variables include y_j (integer unassigned visits), ω_j (binary time availability violations), and ψ_j (binary regional preference violations). Figure 1 illustrates some aspects of the problem.

3.2 Original MILP Formulation

The original MILP formulation in (Laesanklang et al., 2015) includes standard routing constraints: assignment (1), flow conservation (2), depot

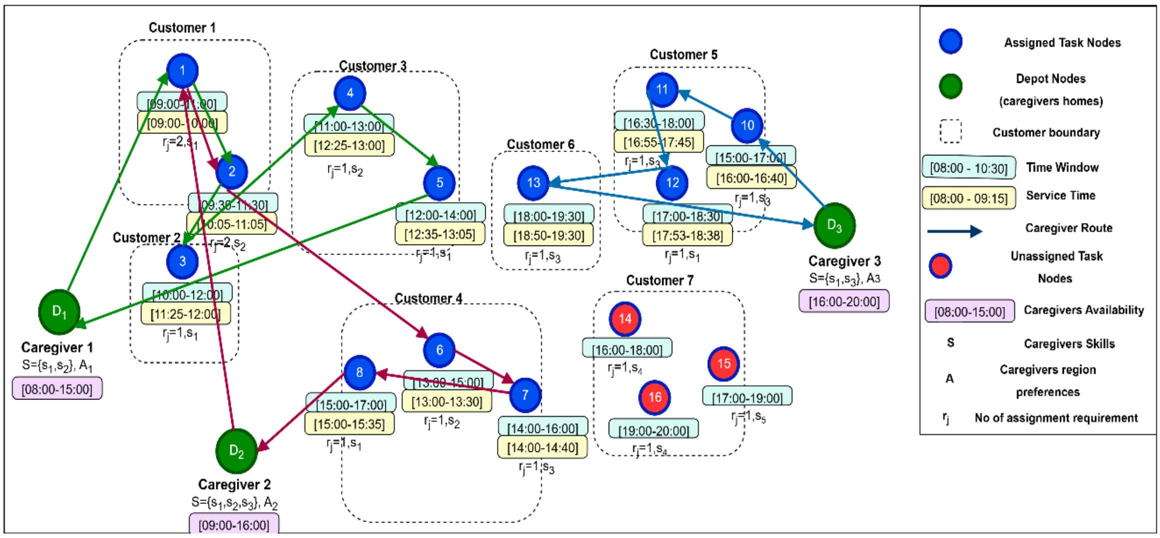

Figure 1: Illustrative example of HHCSRP, with seven customers requesting a total of 15 care tasks. Three tasks requested by customer 7 are unvisited, while tasks 1 and 2 require multiple visits and are performed by workers 1 and 2.

operations (3-6), capacity (7), time sequencing (8), time windows (9), and skill matching (11). Soft constraints enforce regional preferences (10) and worker availability (12-13). The objective function (14) minimises weighted costs: unassigned visits (λ_4), soft constraint violations of availability and region of workers (λ_3), preference dissatisfaction (λ_2), and operational costs of salary and travel (λ_1), where $\lambda_4 > \lambda_3 > \lambda_2 > \lambda_1$.

$$\sum_{k\in K}\sum_{i\in V^d} x_{i,j}^k + y_j = r_j \quad ,\forall j \in T \qquad (1)$$

$$\sum_{i\in V^d} x_{i,j}^k = \sum_{n\in V^a} x_{j,n}^k \quad ,\forall j \in T ,\forall k \in K \qquad (2)$$

$$\sum_{j\in V^a} x_{n,j}^k \geq \sum_{j\in V^a} x_{i,j}^k \quad ,\forall k \in K, \forall i \in T, \exists n \in D \qquad (3)$$

$$\sum_{j\in V^d} x_{i,n}^k \geq \sum_{j\in V^d} x_{i,j}^k \quad ,\forall k \in K, \forall i \in T, \exists n \in D' \qquad (4)$$

$$\sum_{j\in V^a} x_{i,j}^k \leq 1 \quad ,\forall i \in D, \forall k \in K \qquad (5)$$

$$\sum_{i\in V^d} x_{i,j}^k \leq 1 \quad ,\forall j \in D' ,\forall k \in K \qquad (6)$$

$$\sum_{i\in V^d}\sum_{j\in T} x_{i,j}^k * \delta_j \leq H^k \quad ,\forall k \in K \qquad (7)$$

$$a_j^k + M\left(1 - x_{i,j}^k\right) \geq a_i^k + \delta_i + t_{i,j} \quad ,\forall k \in K, \forall i \in V^d ,\forall j \in T \qquad (8)$$

$$w_j^L \leq a_j^k \leq w_j^U \quad ,\forall j \in T, \forall k \in K \qquad (9)$$

$$\sum_{i\in V^d} x_{i,j}^k - \psi_j \leq \gamma_j^k \quad ,\forall k \in K, \forall j \in T \qquad (10)$$

$$x_{i,j}^k . \eta_{s,j} \leq \mu_{k,s} \quad ,\forall k \in K, \forall i \in V^d ,\forall j \in T, \forall s \in S \qquad (11)$$

$$\alpha_L^k - a_j^k \leq M\left(1 - x_{i,j}^k + \omega_j\right) \quad \forall k \in K, \forall i \in V^s, \forall j \in T \qquad (12)$$

$$a_j^k + \delta_j - \alpha_U^k \leq M\left(1 - x_{i,j}^k + \omega_j\right) \quad \forall k \in K, \forall i \in V^s ,\forall j \in T \qquad (13)$$

$$\begin{aligned} Minimise \quad & \sum_{j\in T} \lambda_4 \, y_j + \sum_{j\in T} \lambda_3\left(\omega_j + \psi_j\right) \\ & + \sum_{k\in K}\sum_{i\in V^d}\sum_{j\in V^a} \lambda_2(\rho_{k,j}) x_{i,j}^k \\ & + \lambda_1\left(\sum_{i\in V^d}\sum_{j\in V^a} (d_{i,j}) x_{i,j}^k \right. \\ & \left. + \sum_{k\in K}\sum_{j\in V^a} (p_{k,j}) x_{i,j}^k \right) \end{aligned} \qquad (14)$$

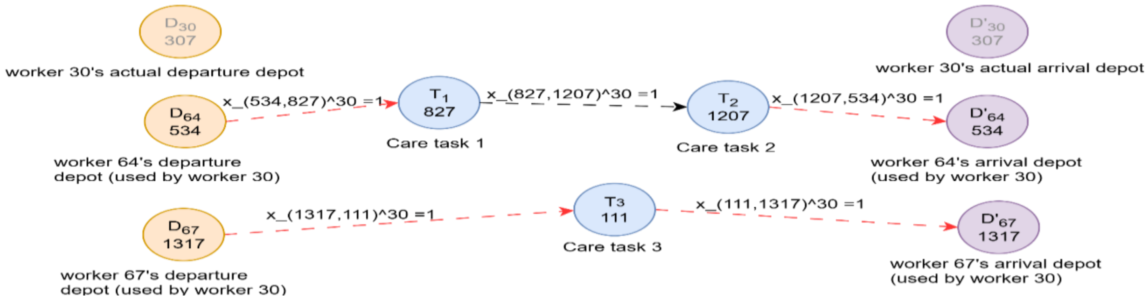

Figure 2: An example of an operational movement violation in the route of worker 30, where a worker's route is allowed to start from and end at two departure depots (location 532 belonging to worker 64 and location 1317 belonging to worker 67). The correct depot for worker 30 (greyed out location 307) is not utilised here.

3.3 Operational Infeasibility Issues

Validating the feasible solution of the original model revealed two critical violations rendering optimal solutions unimplementable:

Issue 1: Weak Depot Enforcement. Constraints (3-4) failed to bind workers to their designated depots, allowing workers to depart from/arrive at different depots other than their own base depot. Figure 2 illustrates worker 30 (designated depot "307") incorrectly starting from/ending at depots "534" and "1317", which belong to other workers, creating routing conflicts while the correct depot remains unused. This relaxed use of different depots is allowed by the above MILP and in VRPs because vehicles are interchangeable, which is not the case for workers in HHCSRP.

Issue 2: Invalid Arcs. Workers received active routes but zero task assignments, thereby reporting to work without productive tasks. Figure 3 illustrates a direct depot-to-depot arc where worker 30 performs no tasks but incurs full salary costs, wasting scarce resources. These violations arose from inadequate constraint formulation combined with the exclusion of salary costs (see Section 4.1), making worker activation free. With zero depot-to-depot distance (same location), empty routes cost nothing while satisfying constraints (2-4). The solver rationally prefers this (cost = 0) over assigning workers to constrained tasks (costly violations) or leaving tasks unassigned (high penalties). Prohibiting these depot-to-depot arcs is essential, as cost set-ups alone cannot prevent this behaviour.

3.4 Refined Model with Included Constraints

We utilise explicit simple constraints, formulated in different ways and used in some VRP literature, to address both issues:

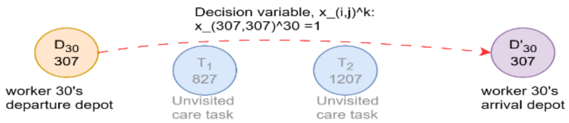

Figure 3: Illustration of the direct routing arc from the departure depot to the arrival depot, where assigned worker 30 performs no productive work by completing zero tasks but still receives the full route salary.

1: Depot Designation Enforcement.

$$\sum_{i \in D \setminus D_k} \sum_{j \in T} x_{i,j}^k = 0 \quad , \forall k \in K \tag{15}$$

$$\sum_{i \in D' \setminus D'_k} \sum_{j \in T} x_{j,i}^k = 0 \quad , \forall k \in K \tag{16}$$

Constraints (15-16) bind each worker to their specific departure/arrival depots by zeroing assignments from/to any other depot, replacing the weak inequalities (3-4) that merely required some depot to have sufficient flows.

2: Arc Feasibility.

$$\sum_{i \in D} \sum_{j \in D'} x_{i,j}^k = 0 \quad , \forall k \in K \tag{17}$$

Constraint (17) eliminates direct depot-to-depot movements without intermediate tasks.

Figure 4 shows the improved routing for worker 30 after adding the constraints. Table 1 shows the formulation differences between the original model's weak depot constraints (3-4) and the refined model's strict depot constraints (15-16).

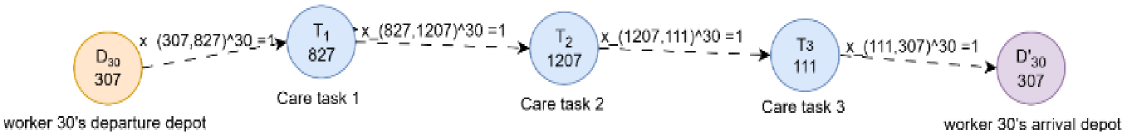

Figure 4: Illustration of the correct routing for worker 30 after including the constraints in the original model. Worker 30 now departs from and arrives at designated departure locations (307) while having three care tasks assigned.

Table 1: Algebraic differences between the weak depot constraints in the original model and the strictly enforced depot constraints in the refined model.

Weak Constraint (3 and 4)	**Strict Constraints (15 and 16)**	**Impact**
Utilises $\exists n \in D$, and $n \in D'$ to specify some departure or arrival depots.	Uses D_k and D'_k to specify explicit departure or arrival depots.	Eliminates ambiguity by precisely enforcing the use of a worker's depots
Uses ≥ operator (inequality) to signify flexible satisfaction.	Uses = operator (equality) to signify a precise requirement	Removes arbitrary depot outflows/inflows and sets an upper bound on depot utilisation
Allows any worker-departure or arrival depot combinations	Establishes a direct relationship between the worker and departure/arrival depots.	Workers are directly bound to their specific departure/arrival depots.

4 EXPERIMENTS AND RESULTS

4.1 Experimental Setup

Both models were solved using GUROBI Optimiser 11.0.0 on an AMD Ryzen 7 PRO 7730U (2.00 GHz, 16GB RAM, Windows 10) with a 2-hour CPU time limit. We evaluate six daily scenarios (A-F) from a UK home healthcare company (Laesanklang et al., 2015). Table 2 presents worker-to-task ratios (K: T), which are necessary for understanding how resource availability affects model difficulty. Note that salary costs were excluded in equation (14) due to data unavailability, a limitation that may affect absolute objective values but not comparative analysis.

Table 2: Summary of worker-to-task ratio, expressed as the K: T ratio and obtained by dividing the number of workers, K, by the number of tasks, T, in each test instance.

Instance	K: T Ratio	Instance	K: T Ratio
A-01	0.7:1	D-01	0.3:1
A-02	0.7:1	D-02	0.4:1
A-03	0.5:1	D-03	0.3:1
A-04	0.7:1	D-04	0.3:1
A-05	1.5:1	D-05	0.3:1
A-06	0.8:1	D-06	0.3:1
A-07	1.6:1	D-07	0.3:1
B-01	0.7:1	E-01	0.6:1
B-02	2.1:1	E-02	0.6:1
B-03	0.5:1	E-03	0.6:1
B-04	1.1:1	E-04	0.8:1
B-05	0.5:1	E-05	0.6:1
B-06	0.6:1	E-06	0.9:1
B-07	0.5:1	E-07	0.6:1
C-01	5.9:1	F-01	0.7:1
C-02	88.3:1	F-02	0.6:1
C-03	7.2:1	F-03	0.6:1
C-04	30.6:1	F-04	0.5:1
C-05	28.3:1	F-05	0.6:1
C-06	5.2:1	F-06	0.5:1
C-07	58.2:1	F-07	0.6:1

4.2 Model Practicality Assessment

Table 3 presents the results for 18 solvable instances (24 instances exceeded memory limits). Practicality cost (PC) is the price of ensuring the realistic implementation of the original model (OM), calculated as the difference between the objective function (OBJ) values of the refined model (RM) and the OM. The OM produced operationally infeasible solutions in 14 of 18 cases (77.8%), with practicality costs (PC) ranging from 0.3 to 33.5. Instance A exhibited the highest violations (A-02: 33.5, A-06: 30.7, A-03: 28.2), while B instances showed moderate violations (0.3-9.9).

Category C instances yielded zero PC despite the added constraints, revealing these are single-depot problems where workers share a depot, structurally equivalent to VRPs. This explains why constraints (3-4) were adequate for C instances but failed in multi-depot scenarios (A, B). This instance-specific behaviour highlights a key limitation: the necessity of a constraint depends on problem structure (Bektaş et al., 2017), suggesting practitioners must validate formulations against their specific operational contexts rather than assuming universal applicability.

The refined model (RM) achieved 16 optimal solutions compared to 15 for OM, with an average increase of 3.1% in OBJ, representing the true cost of eliminating operational infeasibility. However, this modest increase masks a fundamental issue: OM's lower costs resulted from exploiting formulation gaps (generating invalid empty routes) rather than genuine optimisation superiority, a clear distinction between mathematical and practical optimality.

Table 3: Computational results of the original model (OM) across solvable instances by GUROBI, where OBJ is the objective function value and practicality cost (PC) equals the difference between the OBJ of the refined model (RM) and the OM.

Instance	OM OBJ	RM OBJ	PC	OM gap (%)	RM gap (%)
A-01	145.6	149.3	3.7	0.0	0.0
A-02	90.4	123.9	33.5	0.0	4E-3
A-03	135.1	163.4	28.2	0.0	0.0
A-04	59.9	65.05	5.2	0.0	0.0
A-05	36.0	44.5	8.5	1E-2	0.0
A-06	117.1	147.7	30.7	1E-2	1E-4
A-07	44.2	55.8	11.6	0.00	0.0
B-01	94.9	97.3	2.5	0.0	0.0
B-02	32.7	33.1	0.3	0.0	0.0
B-03	187.4	193.8	6.4	0.0	0.0
B-04	87.2	92.4	5.1	0.0	0.0
B-05	167.1	175.0	7.9	0.0	0.0
B-06	146.2	150.1	3.9	2E-3	0.0
B-07	160.9	170.8	9.9	0.0	0.0
C-01	-	-	-	-	-
C-02	36.5	36.5	0.0	0.0	0.0
C-03	-	-	-	-	-
C-04	2369.6	2369.6	0.0	0.0	0.0
C-05	1142.7	1142.7	0.0	0.0	0.0
C-06	-	-	-	-	-
C-07	27.0	27.0	0.0	0.0	0.0
AVG	**282.3**	**291.0**	**8.7**		

4.3 Computational Efficiency

Table 4 compares computational efficiency between the OM and RM formulations. Remarkably, RM improved computational efficiency in 83.3% of instances (15/18), reducing the average solve time by 40.1% (from 14.1 seconds to 8.4 seconds) despite the additional constraints. For example, B-04 was solved in 6.2 seconds for RM versus 28.5 seconds for OM. This demonstrates that tighter formulations reduce feasible regions, enabling efficient solver exploration (Lahyani et al., 2018). While efficiency gains are substantial for small to medium instances, both models encountered memory limits on large instances (D, E, F, C-01, C-03, C-06), indicating scalability challenges and the need for methods addressing real-world large-scale implementations.

Table 4: Computational efficiency comparison, where CPT is CPU computation time and OM and RM are respectively the original and refined models.

Instance	OM CPT (s)	RM CPT (s)
A-01	5.959	2.071
A-02	4.103	1.880
A-03	5.601	8.324
A-04	2.767	1.374
A-05	3.727	0.426
A-06	3.921	1.861
A-07	3.239	0.385
B-01	6.330	3.509
B-02	2.187	0.542
B-03	41.160	29.757
B-04	28.531	6.283
B-05	30.581	25.192
B-06	26.544	14.146
B-07	41.230	18.734
C-01	-	-
C-02	0.371	0.405
C-03	-	-
C-04	21.297	22.254
C-05	25.694	14.528
C-06	-	-
C-07	0.200	0.140
AVG	**14.1**	**8.4**

4.4 Solver Performance Comparison

We compared the performance of GUROBI and CPLEX on RM to assess solver-specific behaviours across problem structures (see Table 5). We analyse the OBJ values obtained by the solvers, and their computation time, CPT. The solver gap on the RM is calculated as (OBJ GUROBI − OBJ CPLEX)/OBJ CPLEX × 100. The computation time is obtained from the literature.

Solution Quality: GUROBI and CPLEX produced comparable solutions for A and B instances (gaps<15%), indicating both solvers operate effectively within RM's tighter feasible region for balanced workforce scenarios (K:T ratios 0.5:1 to 2:1). However, CPLEX dramatically outperformed GUROBI in C instances with high K:T ratios (5.9:1 to 88.3:1), achieving solutions approximately six times better (C-04: 509.5% gap, C-05: 195.4% gap). This is because GUROBI left multiple tasks unassigned in these over-resourced scenarios, incurring high penalty costs. This performance

Table 5: Results of the original MILP model (OM) and refined model (RM) solved by GUROBI and CPLEX solvers, where OBJ is the objective function value, CPT is the CPU Processing Time from the solvers, and the solver gap of the RM is calculated as (OBJ GUROBI – OBJ CPLEX)/OBJ CPLEX x 100.

Instance	OBJ_ GUROBI (OM)	OBJ CPLEX (RM)	OBJ GUROBI (RM)	Solver Gap (%)	CPT GUROBI (OM) (s)	CPT CPLEX (RM) (s)	CPT GUROBI (RM) (s)
A-01	145.6	143.6	149.3	4.0	6.0	7.0	2.1
A-02	90.4	108.2	123.9	14.5	4.1	8.0	1.9
A-03	135.1	151.8	163.4	7.6	5.6	14.0	8.3
A-04	59.9	67.8	65.1	-4.0	2.8	5.0	1.4
A-05	36.0	44.5	44.5	0.0	3.7	1.0	0.4
A-06	117.1	127.4	147.7	15.9	3.9	5.0	1.9
A-07	44.2	61.3	55.8	-9.0	3.2	1.0	0.4
B-01	94.9	97.3	97.3	0.0	6.3	21.0	3.5
B-02	32.7	33.1	33.1	0.0	2.2	2.0	0.5
B-03	187.4	193.8	193.8	0.0	41.2	6003.0	29.8
B-04	87.2	92.4	92.4	0.0	28.5	25.0	6.3
B-05	167.1	175.0	175.0	0.0	30.6	585.0	25.2
B-06	146.2	149.3	150.1	0.5	26.5	184.0	14.1
B-07	160.9	170.2	170.8	0.4	41.2	300.0	18.7
C-01	-	-	-	-	-	-	-
C-02	36.5	29.1	36.5	25.4	0.4	6.0	0.4
C-03	-	-	-	-	-	-	-
C-04	2369.6	388.8	2369.6	509.5	21.3	90.0	22.3
C-05	1142.7	386.9	1142.7	195.3	25.7	55.0	14.5
C-06	-	-	-	-	-	-	-
C-07	27.0	31.8	27.0	-15.1	0.2	1.0	0.1

divergence reveals dependence on solver configurations. High worker-to-task ratios fundamentally change the optimisation problem from routing efficiency (A, B) to worker selection from over-resourced pools (C). CPLEX's robustness in these scenarios suggests its superiority in handling discrete choice problems under resource abundance, while GUROBI excels in resource-constrained routing.

Computational Efficiency: GUROBI consistently solved faster than CPLEX for small-medium cases (average: A instances 4.2s vs 5.9s; B instances 25s vs 1017s). However, CPLEX's longer computation reflects more thorough solution space exploration, explaining its superior solution quality in difficult instances, suggesting a speed-quality trade-off.

5 CONCLUSION AND FUTURE WORK

This study demonstrates that mathematically optimal solutions may be operationally infeasible when critical constraints are omitted or weakly formulated. Validating an existing MILP model for HHCSRP across 42 real-world instances revealed that 77.8% of solvable cases violated essential requirements: workers incorrectly using other depots and invalid depot-to-depot arcs, creating unproductive routes. While mathematically sound with respect to the MILP model, these solutions were unimplementable.

The refined model resolved both violations through explicit depot designation constraints and arc feasibility enforcement, ensuring operational practicality with only a 3.1% average increase in objective function, demonstrating that practical feasibility incurs minimal cost when properly formulated. Tighter constraints improved computational efficiency in 83.3% of instances, reducing average solve time by 40.1% through more efficient solution space exploration.

Key insights emerged regarding context-dependent necessity of constraints: operational constraints proved essential for multi-depot problems (categories A and B) but redundant for single-depot scenarios (category C), challenging assumptions about universal applicability of constraints. This highlights a distinction between VRP and HHCSRP contexts: workers are not commodities and cannot arbitrarily change their home bases, necessitating stricter formulations than the VRP equivalents.

Model validation must extend beyond standard Operations Research (OR) metrics (computation time, objective values) to verify real-world feasibility, a gap this work addresses but remains underexplored in optimisation literature. This validation gap contributes to the scarcity of successful real-world OR deployments.

Future research should: (1) test whether the added constraints generalise across workforce routing domains (for example in field services, emergency response, technician scheduling) where similar feasibility issues may arise, (2) develop decomposition methods addressing scalability limitations, (3) investigate worker-to-task ratio thresholds triggering solver performance divergence, and (4) validate findings across diverse geographical and organisational contexts to establish broader applicability.

ACKNOWLEDGEMENTS

Full financial support was received from the Commonwealth Scholarship Commission, United Kingdom. We are grateful to the University of Nottingham for providing computational resources.

REFERENCES

Begur, S. V., Miller, D. M., & Weaver, J. R. (1997). An Integrated Spatial DSS for Scheduling and Routing Home-Health-Care Nurses. *Interfaces*, *27*(4), 35–48.

Bektaş, T., Gouveia, L., & Santos, D. (2017). New path elimination constraints for multi-depot routing problems. *Networks*, *70*(3), 246–261. https://doi.org/10.1002/net.21760

Bredström, D., & Rönnqvist, M. (2008). Combined vehicle routing and scheduling with temporal precedence and synchronization constraints. *European Journal of Operational Research*, *191*(1), 19–31. https://doi.org/10.1016/j.ejor.2007.07.033

Cheng, E., & Rich, J. (1998). *A Home Health Care Routing and Scheduling Problem* (No. CAAM TR98-04). Rice University, Houston.

Decerle, J., Grunder, O., El Hassani, A. H., & Barakat, O. (2021). A matheuristic-based approach for the multi-depot home health care assignment, routing and scheduling problem. *RAIRO - Operations Research*, *55*, S1013–S1036. https://doi.org/10.1051/ro/2020057

Du, G., & Zhang, J. (2022). Cross-regional manpower scheduling and routing problem with stochastic service times in home health care. *Computers & Industrial Engineering*, *173*, 108668. https://doi.org/10.1016/j.cie.2022.108668

Euchi, J., Masmoudi, M., & Siarry, P. (2022). Home health care routing and scheduling problems: A literature review. *4OR*, *20*(3), 351–389. https://doi.org/10.1007/s10288-022-00516-2

Khorasanian, D., Patrick, J., & Sauré, A. (2024). Dynamic Home Care Routing and Scheduling with Uncertain Number of Visits per Referral. *Transportation Science*, *58*(4), 841–859. https://doi.org/10.1287/trsc.2023.0120

Laesanklang, W., & Landa-Silva, D. (2017). Decomposition techniques with mixed integer programming and heuristics for home healthcare planning. *Annals of Operations Research*, *256*(1), 93–127. https://doi.org/10.1007/s10479-016-2352-8

Laesanklang, W., Landa-Silva, D., & Arturo Castillo Salazar, J. (2015). Mixed Integer Programming with Decomposition to Solve a Workforce Scheduling and Routing Problem: *Proceedings of the International Conference on Operations Research and Enterprise Systems*, 283–293. https://doi.org/10.5220/0005223602830293

Lahyani, R., Coelho, L. C., & Renaud, J. (2018). Alternative formulations and improved bounds for the multi-depot fleet size and mix vehicle routing problem. *Or Spectrum*, *40*(1), 125–157. https://doi.org/10.1007/s00291-017-0494-y

Lalla-Ruiz, E., & Mes, M. (2021). Mathematical formulations and improvements for the multi-depot open vehicle routing problem. *Optimization Letters*, *15*(1), 271–286. https://doi.org/10.1007/s11590-020-01594-z

Nabavizadeh, N., Kayvanfar, V., & Rafiee, M. (2024). A mixed integer linear programming model for quarantine-based home healthcare scheduling under uncertainty. *Healthcare Analytics*, *6*, 100356. https://doi.org/10.1016/j.health.2024.100356

Qiu, H., Wang, D., Yin, Y., Cheng, T. C. E., & Wei, X. (2025). An exact algorithm for the home health care routing and scheduling problem with variable departure times. *Annals of Operations Research*. https://doi.org/10.1007/s10479-025-06690-w

Rasmussen, M. S., Justesen, T., Dohn, A., & Larsen, J. (2012). The Home Care Crew Scheduling Problem: Preference-based visit clustering and temporal dependencies. *European Journal of Operational Research*, *219*(3), 598–610. https://doi.org/10.1016/j.ejor.2011.10.048

Yazır, O. A., Koç, Ç., & Yücel, E. (2023). The multi-period home healthcare routing and scheduling problem with electric vehicles. *OR Spectrum*, *45*(3), 853–901. https://doi.org/10.1007/s00291-023-00721-3

Zhang, T., Liu, Y., Yang, X., Chen, J., & Huang, J. (2023). Home health care routing and scheduling in densely populated communities considering complex human behaviours. *Computers & Industrial Engineering*, *182*, 109332. https://doi.org/10.1016/j.cie.2023.109332