Socio-Economic Planning Sciences 99 (2025) 102204

Contents lists available at ScienceDirect

Socio-Economic Planning Sciences

journal homepage: www.elsevier.com/locate/seps

Integrating logistics planning and population sampling decisions for humanitarian needs assessment

Duygu Pamukcu [a], Melih Çelik [b,*], Burcu Balçık [c]

[a] School of Business, Providence College, 1 Cunningham Square, Providence, RI 02918, United States of America
[b] School of Management, University of Bath, Claverton Down Rd, Combe Down, Bath BA2 7AY, United Kingdom
[c] Department of Industrial Engineering, Ozyegin University, Nisantepe Mah., Orman Sok. No:34-36, 34794, Cekmekoy, Istanbul, Turkey

ARTICLE INFO

Keywords:
Humanitarian logistics
Needs assessment
Routing
Sampling accuracy
Metaheuristics

ABSTRACT

After a disaster, humanitarian organizations must quickly assess the needs of affected populations spread across a large area to inform effective relief distribution. This process involves site visits and collecting information through interviews, where balancing travel time and assessment accuracy is critical. This paper introduces the *Assessment Routing Problem with Coverage and Accuracy*, which addresses the logistical planning and sampling decisions of a post-disaster needs assessment team. The problem involves determining which sites to visit, in what sequence, and how much time to spend at each site to collect information within a total assessment time limit. We present a mathematical model that balances site coverage and sampling accuracy, and propose a constructive heuristic and a tabu search algorithm to efficiently solve large-scale instances. Computational experiments demonstrate that the tabu search algorithm can quickly obtain optimal or near-optimal solutions for large instances, while the performance of the heuristic is influenced by network topology. We illustrate the effectiveness of our approach through a case study based on the 2011 Van Earthquake in Turkey. Our findings emphasize the trade-offs involved in prioritizing certain areas and the advantages of leveraging reliable prior information in the assessment process to achieve better site coverage and sampling accuracy.

1. Introduction

Natural disasters or conflicts can profoundly impact communities, causing fatalities, injuries, physical damage, and economic losses. In these events, affected communities often lack access to essential resources needed for survival and well-being. However, predicting post-disaster needs in advance is challenging due to the unique nature of each event. Therefore, humanitarian agencies conduct immediate needs assessments by deploying teams to affected regions after disasters. These teams make observations, conduct interviews with key informants and potential beneficiaries, and identify the needs of the communities [1]. While relief efforts typically commence immediately using available secondary data, needs assessments also begin promptly and continue throughout the relief horizon, playing a critical role in refining and adapting operations to effectively address evolving post-disaster conditions.

Estimating the prevalence of specific needs in a given population [2], such as the proportion of individuals or households in need of relief items like shelter, food, or hygiene kits, is crucial for guiding disaster response efforts. In our study, without loss of generality, we focus on a humanitarian agency interested in estimating the proportion of people in need of emergency relief supplies across multiple *sites* (e.g., villages, towns, census blocks) within an affected region. As the impact of the disaster on communities can vary between sites, it is often desirable to gather information from each affected site. However, time and resource constraints usually make it impossible to visit all sites and interview a large number of people at each site [3]. Therefore, humanitarian agencies visit a subset of affected sites and conduct interviews with sampled individuals. For example, following Hurricane Ike in 2008, only 68 of 159 sites were assessed for needs [4]. Similarly, after the 2009 tsunami in American Samoa, only 30 out of 244 census blocks were assessed [5].

In conducting post-disaster needs assessments, the choice of sampling method is critical for collecting accurate and representative information at affected sites. Various sampling methods are employed, including probability-based approaches such as random sampling and cluster sampling, and non-probability methods like purposive and convenience sampling [1,6,7]. Probability sampling methods are preferred for their statistical rigor and ability to generalize findings, while non-probability methods are often used when time constraints or access limitations make probability sampling impractical. Humanitarian and

* Corresponding author.
E-mail address: M.Celik@bath.ac.uk (M. Çelik).

https://doi.org/10.1016/j.seps.2025.102204
Received 23 July 2024; Received in revised form 9 March 2025; Accepted 12 March 2025
Available online 21 March 2025
0038-0121/© 2025 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

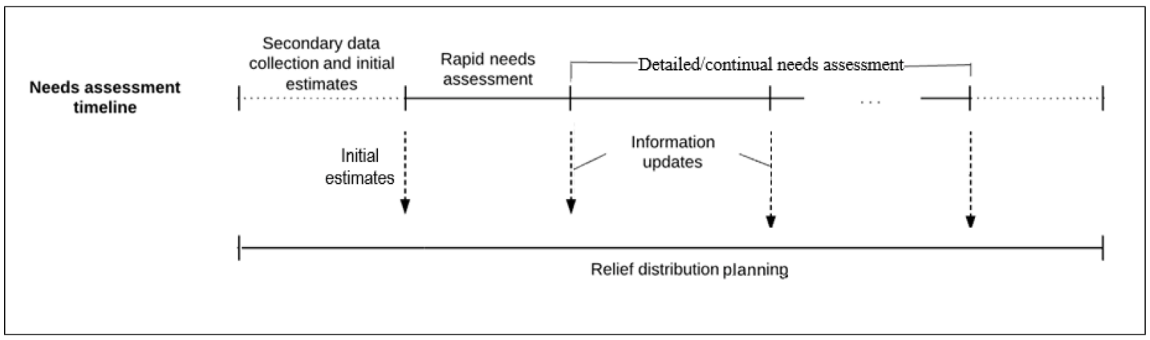

Fig. 1. Timeline of post-disaster needs assessment and relief distribution activities; the ARP-CA can address logistical planning and population sampling decisions during the rapid and detailed/continual needs assessment phases, and its outputs enhance relief distribution planning by means of information updates.

public agencies frequently employ probabilistic sampling methodologies adapted from programs like the World Health Organization's Expanded Programme on Immunization [e.g., 8] to estimate the prevalence of needs in various humanitarian settings. However, despite the widespread application of these sampling methods in practice, operations research (OR) approaches have not been adequately leveraged to integrate logistical planning with sampling decisions. In this study, we consider a probability sampling method, where the prevalence of needs is estimated based on the interviews conducted at selected sites.

Before initiating field visits for needs assessments that utilize probabilistic sampling methodologies, humanitarian agencies often develop initial estimates of needs using available information about the affected region, such as local news, governmental reports, and aerial imagery from satellites or drones. Once these estimates are established, agencies must address two critical decisions: (i) identifying the specific *sites to visit for additional data collection to refine the preliminary estimates*, and (ii) *determining the sample size, that is, the number of individuals to interview* within each selected site. These decisions directly impact two key measures of information quality: (i) *site coverage* (the extent to which information at different sites is captured) and (ii) *sampling accuracy* (the extent to which the estimated needs based on the collected sample reflect the actual needs of the population). The former measure depends on how many sites are visited, whereas the latter can be measured by *sampling (statistical) error*, representing the difference between the estimated needs from the available samples and those that could be obtained by a complete survey of the population. Both measures are critical for the effectiveness of humanitarian relief operations, as they directly affect demand size and uncertainty.

In practice, due to time constraints, humanitarian agencies often opt to visit a larger number of sites and conduct interviews with a smaller number of people at each site [e.g., 9,10]. However, as discussed by [11], this approach can lead to significant sampling errors due to insufficient sample sizes at each site. This results in wider confidence intervals around estimated needs, which may not meet acceptable levels of accuracy. Additionally, [7] highlights that humanitarian assessments often overlook sampling errors, indicating a need for practical methods and tools that prioritize assessment planning while accounting for site coverage and sampling accuracy.

In the OR literature, there is a growing body of work on needs assessment planning that addresses site selection and routing together [e.g., 3,12,13]. However, these studies often do not describe the sampling methodology in detail, or they primarily employ purposive sampling by selecting sites based on specific attributes, as in [3]. Moreover, they typically treat the time spent at each site as a fixed parameter rather than a decision variable. Yet, the selection of sites, the sequence of site visits, and the allocation of time at each site directly impact the amount of time available for interviews at each location, thereby affecting the trade-off between site coverage and sampling accuracy. In other words, logistical planning critically affects the quality of information obtained, both in terms of site coverage and sampling accuracy. Therefore, developing a systematic method to guide humanitarian agencies in choosing which sites to visit, determining the sequence of visits, and allocating the duration of interviews at each site, while balancing the trade-off between coverage and accuracy, would provide substantial practical value in improving the effectiveness of needs assessments during disaster response efforts.

To fill the existing gap, we introduce the *Assessment Routing Problem with Coverage and Accuracy* (ARP-CA), which supports site selection, vehicle routing and sampling decisions of an assessment team within a specified assessment duration. The primary objective of the ARP-CA is to maximize coverage (proportion of sites visited), while the secondary objective, which is measured by the half-width of confidence intervals around the estimated needs at each site, considers the precision of estimations for the visited sites. We model ARP-CA as a variant of the Selective Traveling Salesperson (Orienteering) Problem [14] and Prize Collecting Traveling Salesperson Problem [15], both of which are known to be $\mathcal{NP}$-hard. Because existing solvers may struggle with realistic ARP-CA instances, we develop efficient solution methods, including a constructive heuristic and a tabu search algorithm. Computational experiments using synthetic and real-world case study data demonstrate the effectiveness of our tabu search heuristic in quickly generating high-quality solutions. We apply our approach to a case study of the 2011 Van Earthquake in Turkey, implementing it with multiple teams and pre-stratified sites, and analyze resulting route structures to showcase practical applicability.

To the best of our knowledge, this study is the first to integrate logistical planning and sampling decisions within a probability sampling framework for post-disaster needs assessment, explicitly incorporating the accuracy of collected information. The proposed model and solution approaches can support planning different phases of humanitarian needs assessment operations (i.e., rapid, detailed, or continual) ([9] see also Fig. 1). Moreover, this approach is adaptable to other contexts requiring time-constrained sampling to estimate prevalence, such as household income surveys and health care assessments. In such settings, accurate estimation of prevalence is essential for the effectiveness of the subsequent actions by stakeholders like public health authorities and epidemiologists [2]. For instance, our approach could be tailored to support sampling strategies and logistical decisions for testing asymptomatic cases in schools during epidemics or pandemics [16].

The remainder of this paper is organized as follows. In Section 2, we provide a review of the related literature on humanitarian needs assessment problems. We formally define the ARP-CA and present our mathematical model in Section 3. Our solution approaches are discussed in Section 4, followed by a case study in Section 5. Finally, in Section 6, we conclude the paper and discuss potential future research directions.

2. Literature review

In this section, we review relevant studies that focus on humanitarian needs assessment operations and sampling methods used (Section 2.1) and related routing problems (Section 2.2).

2.1. Humanitarian needs assessment operations and sampling methods

Needs assessment is a fundamental component of humanitarian response operations, providing essential information about the needs of affected populations after disasters. Accurate and timely assessments enable humanitarian agencies to prioritize interventions and allocate resources effectively, thereby maximizing the impact of their efforts. Despite an increasing body of research focusing on needs assessment planning, studies that comprehensively integrate sampling methodologies with logistical decision-making remain scarce.

Practical resources, such as guidelines from humanitarian organizations [e.g., 1,6,9,10,17] outline the actions for planning needs assessment operations, emphasizing the importance of appropriate sampling techniques to ensure reliable data collection. These resources describe various sampling methods, including probability-based approaches like random sampling and cluster sampling, where settlement units are grouped into clusters and a certain number of interviews are conducted per cluster. The earliest known method is the two-stage cluster sampling with a 30×30 design [18], designed to provide statistically valid estimates efficiently. Due to resource limitations or logistical challenges, organizations may adopt reduced-intensity methods that aim to maintain statistical validity while being more feasible in disaster contexts [e.g., 9,10]. Examples include the Expanded Programme on Immunization (EPI) [18] or the Community Assessment for Public Health Emergency Response (CASPER) with a 30×7 design [19]. Other studies explore variations such as 33×6 or 67×3 designs [20], and a 60×7 design [21], all employing cluster sampling techniques tailored to specific contexts.

A substantial body of literature describes needs assessment planning with site selection and sampling decisions from previous disasters (see [22], for a review). For instance, after Hurricane Ivan in 2004, [23] apply the CASPER methodology, a probability-based cluster sampling technique, across two counties in three days to assess household damage and health. Similarly, after Hurricane Ike in 2008, Perry et al. [4] assess affected communities over two days, randomly selecting and assessing 69 clusters of 75 from 159 formed clusters. Another application was seen after the 2009 earthquake and tsunami in American Samoa, where Choudhary et al. [5] use CASPER methodology to randomly assess clusters from 244 census blocks within three days. In South Korea, Hong et al. [24] survey 120 districts post-Typhoons Bolaven and Tembin, assessing five households per district for various needs. Furthermore, several studies have underscored the critical role of Geographic Information Systems (GIS) in enhancing sampling strategies for rapid post-disaster needs assessments by enabling precise site selection and spatial data analysis [25,26]. Recent applications of random sampling methodologies have also been extended to public health contexts, such as COVID-19 prevalence estimation [27–29], with Asgary et al. [30] addressing the associated logistical challenges and resource requirements for effective testing.

While the OR literature on needs assessment planning is expanding, most existing studies overlook sampling design and do not consider the impact of sampling decisions on assessment. For instance, Huang et al. [12] propose heuristics based on continuous approximation for needs assessment routing to minimize arrival times at each site while ensuring all sites are visited, but they do not integrate sampling considerations into their model. In rapid needs assessment, some studies employ purposive sampling plans during the initial assessment stages. For example, Balcik [3] consider site selection decisions to maximize coverage of various site attributes (e.g., distance to epicenter, elevation) using purposive sampling. Building upon this study, Balcik and Yanikoglu [31] introduce a robust optimization scheme accounting for uncertainty in travel times, yet without incorporating sampling design into the decision-making process. Other studies, such as Pamukcu and Balcik [32], Li et al. [33], and Hakimifar et al. [34], extend these models in various ways but similarly do not involve probability-based sampling methods. Recent works on post-disaster damage assessment using unmanned aerial vehicles [e.g., 13,35,36] also focus on logistical aspects without incorporating sampling design.

In summary, while existing humanitarian needs assessment planning problems address site selection and vehicle routing decisions, they typically fix the time spent at each site as a parameter and do not consider population sampling decisions, particularly probability-based sampling methods, as part of the planning process. Moreover, practitioners who conduct field sampling using probability-based methods often do not employ analytical planning tools to support logistical decisions, which can lead to suboptimal resource utilization and data quality. Our study fills this critical gap by developing a new model and solution approaches that integrate probability-based sampling design with assessment routing decisions, considering both site coverage and sampling accuracy. We introduce the Assessment Routing Problem with Coverage and Accuracy (ARP-CA), where decisions on site visits and the time allocated for information gathering at each site are endogenous. This integration aims to support practical applications where logistical and information collection decisions are jointly optimized, enhancing the effectiveness of post-disaster needs assessments under time and resource constraints.

2.2. Related routing problems

Our study not only contributes to the humanitarian logistics literature by introducing a new needs assessment problem but also extends the vehicle routing literature by presenting a novel application and modeling approach. Specifically, the ARP-CA shares similarities with the selective traveling salesperson problems, including the Orienteering Problem (OP) and its variants, particularly those that incorporate time allocation decisions at nodes.

In the classical OP [37], a vehicle aims to maximize the total profit collected by visiting a subset of nodes within a given time limit, with each node offering a fixed profit upon visit. The OP is an $\mathcal{NP}$-hard problem, and numerous modeling and solution approaches have been developed to address it and its variants. Exact methods include integer programming formulations and branch-and-bound algorithms [38], but due to computational complexity, they are limited to small instances. Consequently, heuristic and metaheuristic methods have been extensively explored, such as greedy algorithms [39], local search heuristics [40], tabu search [41], and ant colony optimization [42]. Variants of the OP, such as the Team Orienteering Problem (TOP) [40] and the Profitable Tour Problem [43], have also been widely studied, each with specialized algorithms developed to solve them. In TOP, routes are optimized to maximize collected profit without exceeding time limits. Given its NP-hard nature, various exact and heuristic methods, such as Branch-and-Price variations [44,45], tabu search [46,47], and simulated annealing [48], have been developed.

The ARP-CA extends the OP by incorporating decisions on the time spent at each site for interviews to enhance sampling accuracy. In this respect, the ARP-CA is related to the *Orienteering Problem with Variable Profits* (OPVP) [49], where the profit collected at a node varies depending on the time spent at that node. In the OPVP, the time spent at each node is a continuous decision variable, and the profit collected is a concave, non-decreasing function of the time invested at the node. The vehicle must decide both the subset of nodes to visit and the amount of time to spend at each node to maximize the total profit within the given time constraint. This introduces a trade-off between spending more time at fewer nodes to collect higher profits or visiting more nodes with less time spent at each. Similarly, in our ARP-CA, the assessment team decides not only which sites to visit but also how much

time to allocate at each site for conducting interviews. The sampling accuracy at each site improves with increased time spent, analogous to the profit function in the OPVP.

However, there are major differences between the ARP-CA and the OPVP that necessitate different modeling and solution approaches. Firstly, in the ARP-CA, the time allocation at each site is *discretized into predefined levels*, corresponding to different precision levels of confidence intervals for the needs estimation. Each precision level requires a specific sample size, which translates into a discrete amount of time spent at the site (since each interview takes a fixed amount of time). Furthermore, the "profit" in the ARP-CA corresponds to the *statistical accuracy* of needs estimation, involving modeling the relationship between sample size and the half-width of confidence intervals. This introduces statistical considerations not present in the OPVP, where the profit functions are deterministic and typically modeled as concave functions of time without statistical interpretations. The discretization reflects practical considerations in field assessments, where survey teams plan for specific numbers of interviews based on statistical requirements.

Secondly, the ARP-CA has a *hierarchical objective function* that prioritizes site coverage (maximizing the number of sites visited) over sampling accuracy (minimizing the confidence interval half-widths at visited sites). This reflects the practical importance in humanitarian contexts of obtaining information from as many affected sites as possible, even if the sampling accuracy at each site is not maximized. In contrast, the OPVP aims to maximize the total profit collected, without an explicit priority on the number of nodes visited.

In summary, while existing humanitarian needs assessment planning studies address site selection and vehicle routing decisions, they typically do not incorporate population sampling decisions; the time allocated to each site is fixed and treated as a parameter. We introduce a new needs assessment planning problem that integrates population sampling decisions and formulate it as a variant of an OP. We develop practical heuristics to find high-quality solutions and support humanitarian response efforts on the ground.

3. Problem description and mathematical model

In this section, we define the Assessment Routing Problem with Coverage and Accuracy (Section 3.1), explain how we formulate its objective (Section 3.2), present an example to illustrate important features of the problem (Section 3.3), and present a mixed integer programming model (Section 3.4).

3.1. Problem definition

In post-disaster needs assessment operations, timely and accurate information on the impact of the event and immediate needs of the affected population (e.g., food, water, shelter, medical assistance) is critical. Typically, needs assessment begins within hours of the disaster and may continue for several weeks. Fig. 1 illustrates a timeline of post-disaster needs assessment activities, highlighting their relationship with ongoing relief distribution planning activities. As can be inferred from Fig. 1, an effective rapid and detailed/continual needs assessment planning, which is the main aim of the study in this paper, contributes to the consequent relief distribution planning by means of timely and accurate information updates.

Before commencing needs assessment operations on the ground, humanitarian organizations gather data from various sources including government agencies, national statistics organizations, local communities, and media to comprehend the scope of the situation and identify urgent needs. This data may include risk maps, aerial imagery from satellites and drones, population statistics, and socio-economic indicators, which contribute to developing initial estimates of needs. For instance, initial estimates might indicate that approximately half of the affected population may require shelter and food. These initial estimates are often broad and subject to a high degree of uncertainty. Resolving this uncertainty is crucial for effective relief distribution planning and efficient resource allocation.

During the *rapid needs assessment* phase, agencies conduct site visits, engaging in observations and interviews within affected regions to assess immediate needs. The duration and scope of these assessment activities depend on agency capacities and available resources. Emergency relief distribution begins concurrently with rapid needs assessments (Fig. 1), leveraging secondary data and existing information to provide timely assistance. This includes establishing local distribution networks and initiating emergency deliveries without waiting for assessment completion. As assessments progress, the ongoing relief efforts are adjusted based on the updated needs. In other words, life-saving interventions commence immediately while needs assessments continuously enhance the accuracy of needs estimates. As also shown in Fig. 1, throughout the relief horizon, agencies may conduct multiple *detailed and continual needs assessment* to evaluate the population's evolving needs, thereby guiding the planning of relief and recovery efforts. In practice, needs assessment and relief distribution activities are decoupled and performed in sequence, often by separate teams. Hence, planning an effective needs assessment plays an important role in providing accurate inputs for the relief distribution teams. To achieve this aim, this paper proposes an integrated approach to planning the routes of needs assessment teams and interviews to be made at each site. The models and solution approaches proposed in this study are applicable across the entire assessment horizon, supporting the planning of both rapid and detailed needs assessment activities.

In this paper, we focus on large-scale disaster events that impact multiple *sites* such as census blocks, villages, or counties. The assessment plan involves determining a sequence of sites to be visited by the assessment team and the number of individuals to interview at each site. Our objective is to refine initial needs estimates for each visited site to facilitate effective subsequent relief distribution. We introduce the Assessment Routing Problem with Coverage and Accuracy (ARP-CA) to address these decisions. While considering the planning of routes for multiple teams within a single formulation is possible, we focus on a single assessment team and model the ARP-CA as a variant of an Orienteering Problem. Accordingly, we assume that the affected region is pre-partitioned into subregions (e.g., using clustering methods as demonstrated in the case study in Section 5.2), and each team is assigned a specific subregion. These subregions are determined based on relevant contextual factors, such as geographical and demographic characteristics. Within each subregion, it is crucial to visit as many sites as possible and interview numerous individuals to gather high-quality information. The ARP-CA formulation is a difficult problem to solve, and efficient heuristics are necessary to support practical implementation (see Section 4).

A feasible ARP-CA route for an assessment team begins from a designated origin point, visits selected sites to conduct interviews with individuals, and returns to the origin within a specified assessment time limit. The goal of the ARP-CA is to maximize a composite metric combining weighted site coverage and sampling accuracy measures, while adhering to the total assessment time constraint. The specific formulation of these measures will be elaborated in the next section.

An overview of the key inputs, main decisions, and key performance indicators of the ARP-CA is provided in Fig. 2. The ARP-CA mainly takes as inputs locations and populations of the sites, network data, and initial needs estimates. By optimizing the decisions of which sites to visit, routing of the assessment team, and how many interviews to conduct at each site, the ARP-CA aims to achieve a high level of site coverage and needs estimation accuracy. In the remainder of this section, we elaborate on these factors and provide a mathematical model for the problem.

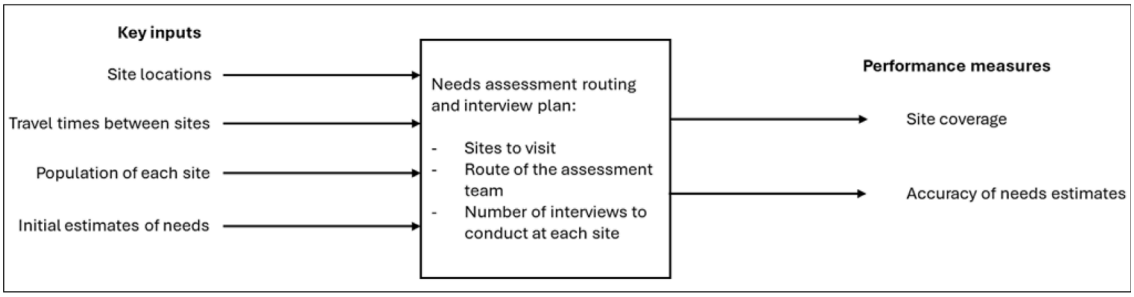

Fig. 2. Key inputs, main decisions, and important performance measures for the ARP-CA.

3.2. Defining and formulating coverage and accuracy objectives

In needs assessment, covering a large number of sites is essential to evaluate the needs across diverse communities, informing subsequent relief distribution decisions. The ARP-CA's *coverage* objective is defined as the proportion of sites visited by the assessment team within the assessment period. Prioritization of specific sites, such as those with higher urgency or more vulnerable populations, can be achieved by assigning weights.

The *accuracy* of a statistical sample pertains to how well the estimation from sampled data reflects the true characteristics of the target population. Discrepancies between these entities can arise from *sampling* and *non-sampling* errors. Sampling error stems from surveying only a portion of the population, potentially leading to a sample that does not fully represent the overall characteristics. Increasing sample size generally reduces sampling error [50]. Non-sampling errors, including non-response bias, respondent bias, and interview design issues, are beyond the scope of this study but can be mitigated through appropriate interview design and follow-up procedures. In the ARP-CA, our emphasis is on enhancing the accuracy of information collected during needs assessment by minimizing sampling errors.

In the ARP-CA, the assessment team aims to estimate the prevalence (proportion) of need for relief supplies at each site. Since the true prevalence values are unknown, the sampling error can only be estimated. One way to quantify this estimation uncertainty is by constructing a confidence interval (CI) around the estimated needs, which indicates the range within which the true needs are expected to fall with a certain probability (referred to as the *confidence level*). The half-width of the confidence interval, also known as the *standard error*, reflects the precision of the estimate. For a given confidence level, the precision of the estimate can be enhanced by reducing the half-width of the confidence interval. The half-width is influenced by the sample size (i.e., number of interviews), and its relationship can be expressed as follows.

Given a site i with a population N_i, let p_i denote the proportion being estimated. With an initial estimate $\hat{p}_i$ and n_i individuals to be interviewed, the half-width h_i of the CI for a confidence level $1-\alpha$ is given by:

$$h_i = z_{\alpha/2}\sqrt{\frac{\hat{p}_i(1-\hat{p}_i)}{n_i}}\sqrt{\frac{N_i-n_i}{N_i-1}}, \tag{1}$$

where $z_{\alpha/2}$ is the standard normal parameter. It is easy to verify that h_i is non-increasing with increasing n_i, which implies that the precision, and hence the sampling accuracy, can be improved by interviewing as many individuals in a site as possible. In the lack of any initial estimates, $\hat{p}_i$ can be set to 0.5, as this maximizes Eq. (1) and yields the most conservative (largest) sample size, with all other parameters held constant. In such a case, the initial CI half-width takes its maximum possible value of 0.5. In the existence of prior estimates, $\hat{p}_i$ and the initial CI half-width h_i may take any value within the intervals $[0,1]$ and $[0,0.5]$, respectively.

The h_i value in Eq. (1), which is a function of the number of interviews in a site, is used to formulate the accuracy objective of the ARP-CA; that is, for a preset confidence level $1-\alpha$, we minimize $\sum_{i\in\mathcal{N}} h_i$ to maximize sampling accuracy. However, this would lead to a nonlinear objective function for the mathematical model. To overcome this, we consider a discrete set of CI half-widths, which we call *precision levels*. More specifically, instead of setting h_i as decision variables, we use a set $\mathcal{L}$ to denote specific CI half-width levels for each site, and use a binary decision variable x_{il} to denote whether level l is chosen for site i. Accordingly, we define the CI half-width for a given precision level l at site i as h_{il} and maintain this notation throughout the remainder of the paper.

We formulate the ARP-CA with the primary objective of maximizing coverage (weighted proportion of sites visited), where weights may represent the vulnerability of sites due to factors including proximity to the disaster area/epicenter, higher proportion of vulnerable populations (such as areas of lower income or hither HIV prevalence), longer distance from locations providing relief, and so on. We also use a secondary objective of maximizing accuracy (average precision over all sites for a given confidence level), so that among the alternative solutions that maximize coverage, we choose the one that maximizes accuracy. To do so, we use a significantly larger coefficient for the coverage in the objective function than that of accuracy, and ensure that the minimum desired accuracy of each visited site exceeds a given threshold.

Furthermore, the population size N_i for each site i is determined using the most recent and reliable secondary data sources available to humanitarian organizations, such as national population censuses or pre-disaster demographic surveys. During the rapid needs assessment phase, organizations utilize this baseline data to select samples, ensuring that the sampling process is as accurate and representative as possible given the available information. However, post-disaster events can significantly alter population distributions through self-evacuation, displacement, or migration, which are often challenging to measure promptly. Our proposed approach facilitates the planning of needs assessments using available population estimates, aligning with current humanitarian practices. The efficiency of our methods enables decision-makers to perform sensitivity and additional analyses, address uncertainties in population estimates, and rapidly update sampling plans as more accurate information becomes available during the assessment process.

3.3. Example: Interaction of sampling and site selection decisions

We present two numerical examples that underscore the importance of integrating site coverage, accuracy, and routing in ARP-CA. First, we illustrate the required number of interviews based on population size, confidence level, and desired confidence interval half-width combinations. Increasing the number of interviews leads to less time available for routing and site coverage, demonstrating diminishing returns for accuracy improvement. Table 1 displays the necessary number of interviews for different population sizes, confidence levels, and precision

Table 1
Required number of interviews for varying target population, confidence level, and CI half-width values when $\hat{p} = 0.5$.

		CI half-width (h_i)						
Population (N_i)	Confidence level ($1-\alpha$)	5.0%	7.5%	10.0%	12.5%	15.0%	17.5%	20.0%
100	0.90	74	55	41	31	24	19	15
	0.95	80	64	50	39	31	25	20
250	0.90	131	82	54	38	27	21	16
	0.95	152	102	70	50	37	28	22
500	0.90	176	98	60	40	29	22	17
	0.95	218	128	81	55	40	30	23
1000	0.90	214	108	64	42	30	22	17
	0.95	278	146	88	58	41	31	24

Table 2
Data for the example in Fig. 3.

			Interview times (in hours) for CI half-width levels						
Nodes	Population	Weight	5.0%	7.5%	10.0%	12.5%	15.0%	17.5%	20.0%
1–4	300	1	7.15	4.35	2.80	1.90	1.40	1.05	0.85
5–8	100	2	3.70	2.75	2.05	1.55	1.20	0.95	0.75

requirements in the case of no prior estimates, i.e., assuming an estimated proportion $\hat{p} = 0.5$ and initial CI half-width $h_i = 0.5$. For instance, to ensure that the actual proportion lies within 5% of the estimate with a probability of 95%, 80 individuals out of a target population of size 100 should be interviewed. Conversely, only 20 individuals are required for a 20% CI half-width at the same confidence level and population size.

Additionally, the table reveals that the percentage of the population to be interviewed decreases as the total population size increases. For example, with a population of 1000, it is sufficient to interview 27.8% of the population to achieve a 5% CI half-width with a confidence level of 95%, whereas 80% of the population must be assessed when the site population is 100.

The second example underlines the importance of efficient routing to maximize both coverage and accuracy objectives. To observe the relationship between site coverage, accuracy, and routing decisions, Fig. 3 shows an ARP-CA instance with eight sites, each with varying populations and weights, the latter of which is determined by distance to the epicenter of the disaster. The data for this instance is provided in Table 2, assuming a 14 hour assessment time limit, 90% confidence level, interview time of 0.05 h per individual, and assuming no prior estimates for all nodes, i.e., $\hat{p}_i$ and h_i set at 0.5. Routes (a) and (b) in Fig. 3 present two feasible routes for this instance. Route (a) is generated using a greedy algorithm (detailed in Section 4.3), while route (b) optimizes the coverage and accuracy objectives hierarchically, representing an optimal solution for the instance. Although both routes nearly consume the full 14 hour limit, route (b) achieves greater efficiency (1.07 h less travel time than route (a)). Moreover, route (b) visits three sites with higher weights compared to the two sites visited by route (a). As a result, the coverage objective for route (b) is higher (0.82 compared to 0.73 for route (a)). The shorter travel time in route (b) also allows for more time dedicated to interviews, leading to improved accuracy with an average confidence interval half-width of 0.26 compared to 0.28 for route (a). Thus, an efficient site selection and routing approach not only facilitates information gathering from more sites but also enhances the accuracy of the collected information.

We next present our mathematical model for the ARP-CA.

3.4. Mathematical model

The notation used throughout the paper for the formulation of the ARP-CA and the solution process is provided in Table 3. To formulate the ARP-CA, we consider a set of nodes $\mathcal{N}$, where each node represents a site. The assessment route starts from a from an origin point, which could be the local assessment coordination center, represented by node {0}; $\mathcal{N}_0 = \mathcal{N} \cup \{0\}$. Network travel times are denoted by t_{ij} $\forall i, j \in \mathcal{N}_0$. For each site i, w_i denotes the importance weight of the site, which may be set based on various vulnerability factors. The total assessment time limit is $\mathcal{T}_{max}$, which includes the time spent for the interviews and travel between the nodes.

Each individual interview takes an average of β time units, which is assumed to be the same for all sites for simplicity. Moreover, $\mathcal{L}$ represents the set of precision levels, with level 1 denoting the most precise level (and hence the most narrow CI half-width) and level $|\mathcal{L}|$ representing the least precise one. The choice of $|\mathcal{L}|$ poses a trade-off between the granularity of the precision levels and required computational time. Based on preliminary experiments in our computational study, we use the seven CI half-width levels presented in Table 1. If a level $l \in \mathcal{L}$ is selected for node $i \in \mathcal{N}$, the CI half-width of the corresponding CI, calculated by (1), is given by h_{il}. For the same level l, n_{il} refers to the number of individuals interviewed (sample size) in site $i \in \mathcal{N}$, which equals:

$$n_{il} = \frac{z^2_{\alpha/2} \frac{\hat{p}_i(1-\hat{p}_i)}{h^2_{il}}}{\frac{N_i - 1}{N_i} + z^2_{\alpha/2} \frac{\hat{p}_i(1-\hat{p}_i)}{N_i h^2_{il}}}, \tag{2}$$

where $\hat{p}_i$ is the initial estimate for proportion of needs at site $i \in \mathcal{N}$. Note that for a given precision level $l \in \mathcal{L}$ and confidence level α, n_{il} is maximum when $\hat{p}_i = 0.5$, since $0 \le \hat{p}_i \le 1$. Hence, if there is no prior estimation, we set $\hat{p}_i = 0.5$ conservatively to aim for maximum possible interviews for that level.

The binary decision variable x_{il} denotes whether precision level $l \in \mathcal{L}$ is chosen for node $i \in \mathcal{N}$, whereas binary variable y_{ij} represents whether node $j \in \mathcal{N}_0$ follows node $i \in \mathcal{N}_0$ on the assessment route. The continuous decision variable v_i determines the CI half-width of site $i \in \mathcal{N}_0$, where $0 \le v_i \le 1$. We use a small parameter λ (such as 0.01) to weight the accuracy objective in the objective function. An auxiliary variable u_i is defined to formulate the subtour elimination constraints.

A multi-objective mixed integer programming model for the ARP-CA is presented below:

$$\text{maximize} \quad \frac{\sum_{i\in\mathcal{N}} \sum_{l\in\mathcal{L}} w_i x_{il}}{\sum_{i\in\mathcal{N}} w_i} - \lambda \frac{\sum_{i\in\mathcal{N}} v_i}{|\mathcal{N}|} \tag{3}$$

$$\text{s.t.} \quad v_i = h_i \left(1 - \sum_{l\in\mathcal{L}} x_{il}\right) + \sum_{l\in\mathcal{L}} h_{il} x_{il} \qquad \forall i \in \mathcal{N}, \tag{4}$$

$$\sum_{l\in\mathcal{L}} x_{il} \le 1 \qquad \forall i \in \mathcal{N}, \tag{5}$$

$$\sum_{l\in\mathcal{L}} x_{il} = \sum_{j\in\mathcal{N}_0} y_{ij} \qquad \forall i \in \mathcal{N}, \tag{6}$$

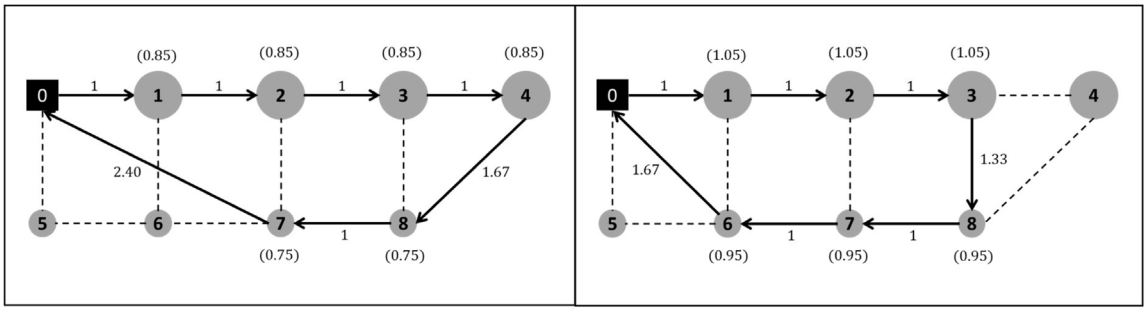

Fig. 3. (a) A feasible route obtained by a greedy algorithm and (b) the optimal route for an ARP-CA instance with 8 sites, where total available assessment time is 14 h; numbers next to the nodes and edges show their corresponding interview and travel times, respectively.

Table 3
Notation used throughout the mathematical model and solution approach.

Index sets	
$\mathcal{N}$	Nodes representing sites
$\mathcal{N}_0$	All nodes including the depot; $\mathcal{N}_0 = \mathcal{N} \cup \{0\}$
$\mathcal{L}$	CI half-width levels for each site
$\mathcal{F}$	Promising nodes in the constructive heuristic
$\mathcal{N}^{CI}$	Nodes in the feasible route resulting from the modified cheapest insertion heuristic
$\mathcal{N}^{NS}$	Candidate promising nodes in the constructive heuristic
$\tilde{\mathcal{N}}$	Nodes in the impact region of node $i \in \mathcal{N}$
$\mathcal{R}^F$	Nodes in the partial tour formed by insertion of all nodes in $\mathcal{F}$
$\mathcal{R}_{best}$	Nodes in the tour resulting from each iteration of the neighborhood search algorithm
Parameters	
$\hat{p}_i$	Initial estimate of the proportion of needs at site $i \in \mathcal{N}$
N_i	Population of site $i \in \mathcal{N}$
w_i	Importance weight of site $i \in \mathcal{N}$
n_{il}	Number of interviews required at site $i \in \mathcal{N}$ to achieve CI half-width level $l \in \mathcal{L}$
h_{il}	Half-width of the CI at site $i \in \mathcal{N}$ corresponding to level $l \in \mathcal{L}$
β	Time required for each interview
$\mathcal{T}_{max}$	Total overall assessment time limit
t_{ij}	Travel time between nodes $i \in \mathcal{N}_0$ and $j \in \mathcal{N}_0$
σ_i	Attractiveness score of node $i \in \mathcal{N}_0$ for the constructive heuristic
t_{route}	Total travel time of the resulting CI route $\mathcal{N}^{CI}$
$tra\mathcal{R}_{best}$	Total travel time of the best tour resulting from the neighborhood search algorithm
Decision variables	
x_{il}	$= \begin{cases} 1, & \text{if CI half-width level } l \in \mathcal{L} \text{ is chosen for site } i \in \mathcal{N} \\ 0, & \text{otherwise} \end{cases}$
y_{ij}	$= \begin{cases} 1, & \text{if item site } i \in \mathcal{N} \text{ and } j \in \mathcal{N} \text{ are visited consecutively in the assessment route} \\ 0, & \text{otherwise} \end{cases}$
v_i	The CI half-width of site $i \in \mathcal{N}_0$, where $0 \le v_i \le 1$
u_i	Auxiliary variable for subtour elimination

$$\sum_{l \in \mathcal{L}} x_{il} = \sum_{j \in \mathcal{N}_0} y_{ji} \qquad \forall i \in \mathcal{N}, \tag{7}$$

$$\sum_{i \in \mathcal{N}} y_{0i} = 1, \tag{8}$$

$$\sum_{i \in \mathcal{N}} y_{i0} = 1, \tag{9}$$

$$\sum_{i \in \mathcal{N}} \sum_{l \in \mathcal{L}} \beta n_{il} x_{il} + \sum_{i \in \mathcal{N}_0} \sum_{j \in \mathcal{N}_0} t_{ij} y_{ij} \le \mathcal{T}_{max}, \tag{10}$$

$$u_i - u_j + |\mathcal{N}+1| y_{ij} \le |\mathcal{N}| \qquad \forall i \in \mathcal{N}_0, j \in \mathcal{N}_0, \tag{11}$$

$$u_i \ge 0 \qquad \forall i \in \mathcal{N}, \tag{12}$$

$$x_{il} \in \{0,1\} \qquad \forall i \in \mathcal{N}_0, l \in \mathcal{L}, \tag{13}$$

$$y_{ij} \in \{0,1\} \qquad \forall i \in \mathcal{N}_0, j \in \mathcal{N}_0. \tag{14}$$

The first term of the objective function (3) maximizes the primary objective of site coverage by calculating the weighted fraction of sites visited, while the second term focuses on the accuracy objective and minimizes the average half-width of the CIs, multiplied by a small weight. The right-hand side of constraints (4) determines the CI half-width for each site. Accordingly, if a site is not visited, its CI half-width stays at h_i. Otherwise, it is assigned the CI half-width determined by (1) for that precision level. Constraints (5) ensure that at most one precision level is selected for each site, whereas (6) and (7) ensure that one arc enters and leaves a selected site. Constraints (8) and (9) enforce the construction of a single assessment route, and (10) imposes an overall assessment time limit. Constraints (11) prohibit subtours [51]. Constraints (12)–(14) define the domain of the decision variables.

For a single precision level in each node, the ARP-CA reduces to the Selective TSP, which is $\mathcal{NP}$-hard [37]. Although it is technically possible to convert an ARP-CA instance to an equivalent OP instance

by creating duplicates of sites for each precision level and enforcing to choose one accuracy level per site, such a transformation significantly increases the size of the network and the computational effort required to solve the resulting problem. This is compounded by the additional challenge of determining the precision level in ARP-CA. Furthermore, as needs assessment plans must be made immediately after a disaster, solutions must be obtained quickly in an environment with no advanced computational resources. The solution approach should also offer the flexibility to tailor the precision options according to the preferences of the decision-maker without significantly increasing the computational complexity. Hence, there is need to develop a novel heuristic approach that accounts for the unique structure of our problem without necessitating network expansion and finds high quality solutions for the ARP-CA in acceptable times. For this end, we propose efficient heuristic methods for the ARP-CA in the next section.

4. Solution methods

To solve the ARP-CA, we first develop an iterative constructive heuristic (CH), which is easy to implement and can produce promising solutions (Section 4.1). Subsequently, we introduce a Tabu Search (TS) metaheuristic (Section 4.2) to enhance solutions derived from the CH through local search. Lastly, we present a greedy algorithm (GA) as a benchmark algorithm to evaluate the CH and TS algorithms' performance in terms of solution quality and time (Section 4.3).

During the CH and TS algorithms, we solve two integer programming models defined by (16)–(20) and (21)–(24), respectively, to select nodes and/or set precision levels. These models can be solved very quickly without increasing the computational effort of the algorithms. Our TS metaheuristic also incorporates a diversification strategy that introduces major random changes in routes throughout the algorithm. Additionally, the TS algorithm is restarted multiple times with different parameters from the CH solution.

TS has proven effective for variants of routing problems [e.g., 3,47, 52], and our computational results confirm the effectiveness and efficiency of the proposed TS algorithm for ARP-CA, despite the inclusion of mathematical models and a multi-start strategy. Appendix A outlines the general flow of the TS algorithm and we proceed to elaborate on the solution approach.

4.1. A constructive heuristic

To obtain a good initial solution for the ARP-CA, we develop an iterative constructive heuristic, whose steps are provided in Algorithm 1. The main idea behind the proposed CH is to construct a feasible route, $\mathcal{N}^{CI}$, by identifying a set of promising nodes, $\mathcal{F}$, to be visited for certain, and adding a promising node to the partial route in each iteration. We define a promising node as the one that is likely to improve coverage and accuracy at the same time without considerable increase in the travel time.

The CH consists of four main steps. In Step 1.1, a promising node is found to initiate the route in the first iteration by computing a desirability score for each node. In Step 1.2, route $\mathcal{N}^{CI}$ is constructed by implementing a modified cheapest insertion (CI) heuristic, which first creates a partial route, $\mathcal{R}^F$, by inserting all nodes in $\mathcal{F}$ and, then, possibly adds a number of additional nodes into this partial route until the time limit $\mathcal{T}_{max}$. The constructed route $\mathcal{N}^{CI}$ in Step 1.2 is also used to identify candidate promising nodes in terms of route duration. In Step 1.3, we solve an integer programming model to identify a set of candidate promising nodes, $\mathcal{N}^{NS}$, in terms of the accuracy objective. In Step 1.4, candidate promising nodes from Step 1.2 and Step 1.3 are evaluated. One of the common nodes is selected and added to the set $\mathcal{F}$. The CH terminates when no further candidate promising nodes can be identified. The details of each step are provided below in detail.

Algorithm 1 The steps of the constructive heuristic

$\mathcal{F} = \emptyset$
$STOP = 0$
while $STOP = 0$ **do**
  $\mathcal{N}^{CI} = \{0\}$
  $t_{route} = t_{total} = 0$
  if $\mathcal{F} = \emptyset$ **then**
    Step 1.1: Route initialization
    Insert $j = \arg\max_{i \in \mathcal{N}} \{\sigma_i\}$ after the origin point into $\mathcal{N}^{CI}$
    $t_{route} \leftarrow t_{0j} + t_{j0}$, $t_{total} \leftarrow t_{route} + \beta\, n_{j,|\mathcal{L}|}$
  else
    Step 1.2: Node insertion by modified cheapest insertion
    $(\mathcal{R}^F, t_{route}, t_{total}) \leftarrow$ **cheapestInsertion**$(\{0\}, \mathcal{F}, t_{route}, t_{total})$
  end if
  $(\mathcal{N}^{CI}, t_{route}, t_{total}) \leftarrow$ **cheapestInsertion**$(\mathcal{R}^F, \mathcal{N} \setminus \mathcal{F}, t_{route}, t_{total})$
  Step 1.3: Node selection mathematical model
  $\mathcal{N}^{NS} = \emptyset$
  Solve the node selection model defined by (16)–(20)
  for all $i \in \mathcal{N}$ such that $\sum_{l \in \mathcal{L}} x_{il} = 1$ **do**
    $\mathcal{N}^{NS} \leftarrow \mathcal{N}^{NS} \cup \{i\}$
  end for
  Step 1.4: Addition of a promising node to set $\mathcal{F}$
  $k = \arg\min_{j \in \mathcal{N}^{NS} \setminus \mathcal{F}} \left\{ \min_{i \in \mathcal{N}^{CI} : t_{ij} < fixTime} \left\{ \sum_{k \in \mathcal{F}} t_{jk} \right\} \right\}$
  if $k = NULL$ **then**
    $STOP = 1$
  else
    $\mathcal{F} \leftarrow \mathcal{F} \cup \{k\}$
  end if
end while
return $\mathcal{N}^{CI}$, t_{route}, t_{total}

Algorithm 2 Function **cheapestInsertion**$(\mathcal{R}, C, t_r, t_{tot})$

$\mathcal{U} = C$
while $\mathcal{U} \neq \emptyset$ and $t_{tot} \leq \mathcal{T}_{max}$ **do**
  $j = \arg\min_{i \in \mathcal{U}} \left\{ \min_{k=\{1,2,\ldots,|\mathcal{R}|-1\}} \left\{ t_{\mathcal{R}_{[k]},i} + t_{i,\mathcal{R}_{[k+1]}} - t_{\mathcal{R}_{[k]},\mathcal{R}_{[k+1]}} + \beta\, n_{i,|\mathcal{L}|} \right\} \right\}$[1]
  $temp = \min_{k=\{1,2,\ldots,|\mathcal{R}|-1\}} \left\{ t_{\mathcal{R}_{[k]},j} + t_{j,\mathcal{R}_{[k+1]}} - t_{\mathcal{R}_{[k]},\mathcal{R}_{[k+1]}} \right\}$
  if $t_{tot} + temp + \beta\, n_{j,|\mathcal{L}|} \leq \mathcal{T}_{max}$ **then**
    Insert j into $\mathcal{R}$ as the $k+1^{st}$ element
    $\mathcal{U} \leftarrow \mathcal{U} \setminus \{j\}$
    $t_r \leftarrow t_r + temp$
    $t_{tot} \leftarrow t_{tot} + temp + \beta\, n_{j,|\mathcal{L}|}$
  else
    $\mathcal{U} = \emptyset$
  end if
end while
return $\mathcal{R}$, t_r, t_{tot}

Step 1.1 - Route initialization: To initialize a route, we identify a promising node by considering its potential effects in the route based on both travel time and interview time. For this, we compute a *desirability (attractiveness) score* σ_i for each node $i \in \mathcal{N}$. To quantify the attractiveness of a node, we define an *impact region* $\tilde{\mathcal{N}}(i)$ of node $i \in \mathcal{N}$, which includes node $i \in \mathcal{N}$ and its two closest neighbors, representing the potential predecessor and successor of node i on a potential route. The desirability score σ_i is calculated as:

$$\sigma_i = \left(t_{0i} + \sum_{j \in \tilde{N}(i), j \neq i} t_{ij} + \sum_{j \in \tilde{N}(i)} \beta\, n_{i,|\mathcal{L}|} \right)^{-1}, \tag{15}$$

[1] For $k = \{1, 2, \ldots, |\mathcal{R}| - 1\}$, $\mathcal{R}_{[k]}$ refers to the index of the kth visited node after the origin on route $\mathcal{R}$. By convention, $\mathcal{R}_{[0]} = \mathcal{R}_{[|\mathcal{R}|]} = \{0\}$.

where the three terms in the summation account for (i) the closeness of node i to the origin point, (ii) the closeness of node i to the nodes in $\tilde{N}(i)$, and (iii) interview times associated with the lowest precision level $|\mathcal{L}|$ for the nodes in $\tilde{\mathcal{N}}(i)$, respectively. Different from similar desirability scores used in various heuristics developed to solve the Orienteering Problem [e.g., 37,40], in our case, σ_i also includes the time spent for interviews at the sites.

Step 1.1 is executed only once, when $\mathcal{F} = \emptyset$ in the beginning of the algorithm. Node i with the largest σ_i score is selected and inserted after the origin node.

Step 1.2 - Node insertion by a modified cheapest insertion (CI) algorithm: We modify a typical CI heuristic [e.g., 53] by (i) incorporating interview times and (ii) ensuring that $\mathcal{T}_{max}$ is not exceeded after an insertion is made. Our CI heuristic takes an existing route, a set of candidate nodes, as well as the travel time and total assessment time of the route as inputs, and returns the updated route, travel time, and assessment time. In each iteration of Algorithm 1 (except the first iteration, where $\mathcal{F}$ only consists of a single from Step 1.1), the modified CI is executed twice. The first execution starts with the origin node and inserts the nodes in $\mathcal{F}$ into the route $\mathcal{R}^F$, while the second execution takes this route $\mathcal{R}^F$ as its input and inserts as many nodes as possible from $\mathcal{N} \setminus \mathcal{F}$ into this route. Consequently, $\mathcal{N}^{CI}$ includes all nodes in $\mathcal{F}$ and possibly a number of promising nodes from $\mathcal{N} \setminus \mathcal{F}$. A pseudocode of the modified CI heuristic is provided in Algorithm 2.

Step 1.3 - Node selection model: Given the resulting CI route $\mathcal{N}^{CI}$ and the travel time t_{route} from Step 1.2, the node selection model selects at most $|\mathcal{N}^{CI}| - 1$ nodes (as $\mathcal{N}^{CI}$ also includes the origin) and assigns precision levels to each, so that the total precision is maximized within an interview time limit of $\mathcal{T}_{max} - t_{route}$. The aim here is to select a subset of nodes that are promising in terms of improving accuracy. The integer programming formulation the node selection model is given by:

$$\text{minimize} \quad \sum_{i \in \mathcal{N}_0} \sum_{l \in \mathcal{L}} h_{il}\, x_{il} \tag{16}$$

$$\text{subject to} \quad \sum_{l \in \mathcal{L}} x_{il} \leq 1 \qquad \forall i \in \mathcal{N}_0 \tag{17}$$

$$\sum_{i \in \mathcal{N}_0} \sum_{l \in \mathcal{L}} x_{il} \leq |\mathcal{N}^{CI}| - 1 \tag{18}$$

$$\sum_{i \in \mathcal{N}_0} \sum_{l \in \mathcal{L}} \beta\, n_{il}\, x_{il} \leq \mathcal{T}_{max} - t_{route} \tag{19}$$

$$x_{il} \in \{0,1\} \qquad \forall i \in \mathcal{N}_0, l \in \mathcal{L} \tag{20}$$

The objective function (16) minimizes the total CI half-width associated with the selected nodes. Constraints (17) guarantee choosing one precision level for each selected node. The number of nodes to be selected is limited by constraint (18). The amount of time available for population sampling at the selected nodes is specified by constraint (19). Finally, constraints (20) define binary variables.

The resulting set of nodes by the solution of the node selection model is denoted by $\mathcal{N}^{NS}$.

Step 1.4 - Addition of a promising node to set $\mathcal{F}$: In this step, we search for a new promising node to be added into set $\mathcal{F}$, using the route $\mathcal{N}^{CI}$ from Step 1.2 and the node set $\mathcal{N}^{NS}$ in Step 1.3. Within the set of nodes $(\mathcal{N}^{CI} \cup \mathcal{N}^{NS}) \setminus \mathcal{F}$, we consider every node i such that (i) $i \in \mathcal{N}^{CI} \cap \mathcal{N}^{NS}$, or (ii) $i \in \mathcal{N}^{CI}$ or $i \in \mathcal{N}^{NS}$, and i is less than a pre-determined threshold units of distance from a node in the other set ($\mathcal{N}^{NS}$ if $i \in \mathcal{N}^{CI}$ and vice versa), as candidates to add to $\mathcal{F}$. Among these candidates, we select the node whose total travel time to the nodes in $\mathcal{F}$ is the smallest. If a candidate satisfying one of the two conditions cannot be found, then $\mathcal{F}$ cannot be updated further and Algorithm 1 terminates with $\mathcal{N}^{CI}$ as the initial feasible solution. Otherwise, the node is added to set $\mathcal{F}$ and we go back to Step 1.2, where the CI heuristic is applied twice as described above.

4.2. Improvement by a multi-start tabu search

The initial solution resulting from the CH is improved by a tabu search (TS) heuristic, comprising neighborhood search, precision level assignment, diversification, and multi-start components, detailed below. The TS algorithm iterates through a fixed number of neighborhood search and precision level assignment steps.

4.2.1. Neighborhood search

In neighborhood search, multiple local search moves are combined within the TS framework to reduce route duration and increase the number of visited sites. Once a feasible route is determined and the time that can be spent for sampling is known, we solve an integer program to set the precision levels at each visited site.

The neighborhood search phase involves four sequential operators. Three operators aim to reduce travel time without changing the number of visited nodes (thus maintaining coverage), while the fourth operator prioritizes coverage improvement. In the first three operators, we select the best move to minimize total route duration. The fourth operator allows an increase in route duration to enhance coverage, followed by reducing total route duration. Each move maintains a separate *tabu list* to prevent nodes involved in recent moves from being used again. Neighborhood search moves continue until reaching a predetermined number of iterations. Refer to Appendix B for detailed move descriptions.

The *2-opt* operator swaps two edges within the existing route, updating travel time and total time accordingly. A tabu list maintains pairs of tabu-active nodes to prevent certain edge exchanges. The *Swap* operator changes the positions of two nodes in the route, maintaining a tabu list of swapped node pairs. The *Replace 1-1* move substitutes a visited node with an unvisited one, with separate tabu lists for insertion and removal sub-moves. The *Replace 1-2* move extends Replace 1–1 by inserting two nodes not in the current tour, prioritizing nodes with higher weights to enhance tour coverage. Similar to Replace 1–1, this move maintains separate tabu lists for insertion and removal sub-moves.

Any attempt for each of the four moves yields a new route. For the move to be accepted, the new route must adhere to the total travel and assessment time limit and improve either the total travel time or the coverage (number of nodes visited) of the current iteration. A fixed tabu tenure is used for each of the four moves, indicating the number of iterations a node or a node pair remain in a tabu list. Once a new route is accepted, the tabu-active periods of each node or node pair in the tabu lists is updated.

Our TS algorithm employs a multi-start strategy. We clear all tabu lists at the start of each restart. Additionally, once in predetermined number of restarts, new tabu tenure values are assigned for four neighborhood search moves. In each restart, the neighborhood search moves are executed until a specified iteration count is reached.

4.2.2. Precision level assignment

At the end of each neighborhood search iteration, we allocate the remaining time from travel to each visited node to conduct interviews by assigning precision levels. For this end, we solve the following model:

$$\text{minimize} \quad \sum_{i \in \mathcal{R}_{best}} \sum_{l \in \mathcal{L}} h_{il}\, x_{il} \tag{21}$$

$$\text{subject to} \quad \sum_{l \in \mathcal{L}} x_{il} = 1 \qquad \forall i \in \mathcal{R}_{best} \tag{22}$$

$$\sum_{i \in \mathcal{R}_{best}} \sum_{l \in \mathcal{L}} \beta\, n_{il}\, x_{il} \leq (\mathcal{T}_{max} - tra\mathcal{R}_{best}) \tag{23}$$

$$x_{il} \in \{0,1\} \qquad \forall i \in \mathcal{R}_{best}, l \in \mathcal{L} \tag{24}$$

The objective function (21) of the model minimizes the total CI half-width over the nodes in the route. Selecting only one level per node

is imposed by Constraints (22). The interview time limit is given in Constraint (23), which is the remaining available time after the travel time of the best route ($traR_{best}$) is deducted. Constraints (24) enforce the assignment variables to be binary.

4.2.3. Diversification and multi-start approach

To diversify the local search, we employ a diversification scheme after a fixed number of improvement iterations. Three alternative strategies are considered. The first removes a predetermined proportion ($\theta_1\%$) of nodes from the beginning of the current route, while the second removes $\theta_2\%$ of nodes randomly. The third strategy relies on long-term memory, removing $\theta_3\%$ of nodes based on their frequency in the incumbent solution since the algorithm's start. In each iteration, one of these strategies is randomly implemented. Whenever one of these procedures is invoked, the TS algorithm jumps to a new solution and continues with local search after emptying the tabu list and resetting the tabu tenure. Our preliminary results in Section 5.1.2 demonstrate the robustness of the TS algorithm with these diversification strategies.

4.3. A greedy algorithm

To evaluate the performance of the CH and TS algorithms in terms of solution quality and computational time, we compare them against a Greedy Algorithm (GA) as a benchmark. The GA operates by leveraging a Cheapest Insertion algorithm, starting from the closest node to the depot and iteratively inserts nodes that lead to the least increase in travel time until the travel time limit is reached. It then assigns the worst precision level to visited nodes to establish a feasible solution. The remaining time is distributed for further interviews among the visited nodes. This is done iteratively by improving the precision level of each node, one level at a time, in increasing order of population size, until the time limit is reached.

5. Numerical analysis

We illustrate our model and test the performance of the proposed heuristics by conducting computational experiments on two sets of instances: randomly generated hypothetical instances (Section 5.1) and a case study based on the 2011 Van Earthquake in Turkey (Section 5.2). We also provide a comprehensive analysis of heuristic solutions (Section 5.3) aiming to explain the structure of ARP-CA routes, offer practical insights for ground needs assessment planning, and assess solution sensitivity to problem parameters.

Our heuristics are implemented in Java on a computer with an Intel Core i7-4790 3.60 GHz processor and 16 GB RAM. For the first set of instances, we solve the mixed integer programming model in Section 3.3 using ILOG CPLEX 12.9 through ILOG Concert Technology in Java, employed also for the integer programming models during Steps 2a and 2b of the TS heuristic, with a 1 hour runtime limit for each instance. Although we do not impose an explicit time constraint on the TS heuristic, we set limits on the number of restarts and iterations per restart. Throughout all experiments, the run time of the TS heuristic remains within an hour.

5.1. Results: Hypothetical instances

We first explain how we generate hypothetical instances and set algorithm parameters. Subsequently, we present and discuss the results of our study.

5.1.1. Test instances

Our experiments use four networks derived from instances in [54] (see Appendix C). We generate both *random* and *clustered* networks by selecting the first 30 and 50 nodes from the c101 and r101 instances, respectively. Each node's population is randomly assigned between 200 and 2500, and the node weights (w_i) are assumed to be identical.

To calculate network travel times, we compute the Euclidean distance and divide it by a speed factor of 30. We use seven CI half-widths from Table 1 for different precision levels and employ either a 90% or 95% confidence level. The required number of interviews for each precision level at each node is determined using Eq. (1), with each interview assumed to take 0.05 time units. For the 30-node instances, the maximum assessment time ($\mathcal{T}_{max}$) varies as 5, 15, and 25 time units, while for the 50-node instances, the limits are 10, 20, and 40 time units. To model settings without initial prevalence estimates, we set $\hat{p}_i = 0.5$ and the initial CI as $[0, 1]$ for all nodes. To consider cases with initial estimates, we also generate instances where the $\hat{p}_i$ values and the initial CI half-widths are generated randomly.

In total, we obtain 48 instances with four different network types (30NR and 50NR for random networks, 30NC and 50NC for clustered networks), three assessment time limits (5T, 15T and 25T for 30-node networks; 10T, 20T and 40T for 50-node networks), two confidence levels (90 A and 95 A for 90% and 95%, respectively) and two initial estimate structures (0.5P and RP for fixed and random initial estimates, respectively).

5.1.2. Setting algorithm parameters

The TS algorithm employs four types of neighborhood search moves (2-opt, swap, replace1-1, replace1-2), each with its own tabu tenure. Initially, each move starts with a tabu tenure of one, increasing by one every 10 iterations ($maxMainIter$). We limit restarts ($maxStart$) to 50, varying tabu tenure from 1 to 5 and reporting the best restart result. For each improvement iteration, we set the maximum number of consecutive executions for 2-opt and swap moves ($maxIter1$) to 500, and for replace1-1 and replace1-2 moves ($maxIter2$) to 1000. We use the best travel time as the aspiration criterion.

Preliminary tests compare diversification strategy performance outlined in Section 4.4. Appendix D presents these results, showing robustness in terms of diversity strategy, with the TS_0.33B strategy consistently delivering similar objective values across most instances. Notably, this strategy performs at least as well as others in all but one setting ($|\mathcal{N}| = 50R$ and $\mathcal{T}_{max} = 40$), demonstrating slightly better overall and worst-case performance. Consequently, we report the remaining results throughout this section under this strategy.

5.1.3. Assessment of heuristic performance

We assess the performance of the TS algorithm, GA, and CH benchmarks by comparing their solution quality and computational time against the best CPLEX results in Table 4. For each instance, we report the objective value of the best feasible CPLEX solution, along with CPU time, travel time, interview time, and number of sites visited for both CPLEX and TS. In cases where CPLEX cannot find the optimal solution within an hour, we also report the relative optimality gap ($\Delta\% = 100\% \times [z_{UB} - z_{LB}] / z_{UB}$), where z_{UB} and z_{LB} denote the objective value of the upper and lower bound found by CPLEX, respectively. We also calculate the heuristic gap from the best solver solution, where we set z_{LB} to the heuristic objective value. A negative % gap value then implies that the heuristic returns a better solution than CPLEX for that instance. The % gaps of GA, CH and TS from the best solver solution are denoted as δ_{GA}, δ_{CH} and δ_{TS}, respectively. A $|\delta\%| < 0.03\%$ is considered negligible and effectively treated as $\delta\% = 0.00\%$ as the deviation within the specified tolerance is permissible due to decimal rounding.

Table 4 reveals that CPLEX achieves optimality for only 16 out of 48 instances within the time limit, primarily solving smaller (30-node) instances. The average optimality gap across all instances is 7.58%, with $\Delta\%$ values of 5.51% and 9.65% for 30-node and 50-node instances,

Table 4
Objective and gap values, CPU times (in seconds), travel and interview times, and number of sites visited for the randomly generated instances using CPLEX, the GA, the CH, and the TS algorithm.

		CPLEX						GA		CH		TS				
Inst. #	Instance	Obj.	Δ%	CPU time	Travel time	Int. time	Sites vis.	δ_{GA}%	CPU time	δ_{CH}%	CPU time	δ_{TS}%	CPU time	Trav. time	Int. time	Sites vis.
1	30NR_5T_90A_0.5P	0.1287	0.00	1.34	1.31	3.65	4	0.00[a]	0.09	0.00[a]	2.51	0.00[a]	5.43	1.31	3.65	4
2	30NR_15T_90A_0.5P	0.3962	0.00	319.72	3.71	10.90	12	8.43	0.02	0.00	7.69	0.00	14.34	4.51	10.45	12
3	30NR_25T_90A_0.5P	0.6971	0.00	0.73	6.95	18.05	21	4.79	0.02	4.79	20.17	4.79	30.70	7.15	17.85	20
4	30NR_5T_95A_0.5P	0.0953	0.00	0.39	1.03	3.90	3	0.00[a]	0.02	0.00[a]	2.01	0.00[a]	3.59	1.03	3.90	3
5	30NR_15T_95A_0.5P	0.2959	3.12	3601.60	3.18	11.80	9	0.00[a]	0.09	0.00[a]	12.73	0.00[a]	19.98	3.45	11.45	9
6	30NR_25T_95A_0.5P	0.5300	0.00	120.90	5.51	19.40	16	0.00[a]	0.02	0.00[a]	6.37	0.00[a]	21.09	5.62	19.15	16
7	50NR_10T_90A_0.5P	0.1555	6.88	3600.61	1.62	6.85	8	0.00	0.02	0.00	27.48	0.00	38.02	2.41	7.55	8
8	50NR_20T_90A_0.5P	0.3360	3.36	3603.57	5.07	14.90	17	5.96	0.03	0.00[a]	35.84	0.00[a]	67.19	5.14	14.70	17
9	50NR_40T_90A_0.5P	0.6971	0.38	3600.84	10.15	29.85	35	2.87	0.02	2.87	77.19	2.87	157.34	10.71	29.25	34
10	50NR_10T_95A_0.5P	0.1154	5.88	3600.29	1.87	8.10	6	0.00[a]	0.02	0.00[a]	22.70	0.00[a]	31.18	1.88	7.90	6
11	50NR_20T_95A_0.5P	0.2558	3.39	3603.67	4.05	15.90	13	7.84	0.09	0.00	31.72	0.00	51.04	4.02	15.85	13
12	50NR_40T_95A_0.5P	0.5366	0.00	249.82	7.55	32.45	27	3.73	0.03	3.72	152.83	3.72	247.52	8.19	31.75	26
13	30NR_5T_90A_RP	0.1292	0.00	2.31	1.31	3.65	4	0.36	0.02	0.35	1.72	0.35	3.67	1.31	3.65	4
14	30NR_15T_90A_RP	0.3966	0.00	3015.54	4.76	10.20	12	0.10	0.02	0.10	11.16	0.10	17.92	4.51	10.30	12
15	30NR_25T_90A_RP	0.6974	0.00	56.44	7.30	17.65	21	4.83	0.03	0.00[a]	6.87	0.00[a]	17.71	7.52	17.45	21
16	30NR_5T_95A_RP	0.0958	0.00	2.01	1.03	3.90	3	0.51	0.02	0.51	1.97	0.51	3.58	1.03	3.90	3
17	30NR_15T_95A_RP	0.2964	5.06	3601.71	3.48	11.45	9	0.16	0.11	0.16	18.23	0.16	26.88	2.93	11.95	9
18	30NR_25T_95A_RP	0.5303	0.00	1994.60	5.93	19.05	16	6.36	0.02	0.07	11.98	0.07	23.70	5.62	19.25	16
19	50NR_10T_90A_RP	0.1560	9.09	3606.97	2.77	7.05	8	0.32	0.05	0.32	30.87	0.32	42.25	2.41	7.45	8
20	50NR_20T_90A_RP	0.3365	5.72	3604.84	5.65	14.35	17	0.00	0.03	0.14	58.00	0.14	86.79	5.26	14.70	17
21	50NR_40T_90A_RP	0.7172	0.00	78.15	9.50	29.60	36	5.59	0.11	2.79	30.95	2.79	127.88	10.75	29.20	35
22	50NR_10T_95A_RP	0.1159	9.62	3604.11	2.49	7.50	6	0.45	0.05	0.45	28.80	0.45	40.74	1.88	8.05	6
23	50NR_20T_95A_RP	0.2563	6.43	3602.37	4.58	14.20	13	0.20	0.02	0.20	71.15	0.20	95.76	15.95	4.02	13
24	50NR_40T_95A_RP	0.5370	3.21	3608.82	8.22	31.70	27	0.07	0.01	0.07	64.73	0.07	152.69	8.68	31.30	27
25	30NC_5T_90A_0.5P	0.1287	0.00	59.44	1.48	3.40	4	25.92	0.01	0.00[a]	1.64	0.00[a]	3.32	1.49	3.35	4
26	30NC_15T_90A_0.5P	0.3962	20.57	3600.38	4.34	10.65	12	0.00[a]	0.03	0.00[a]	10.37	0.00[a]	19.09	4.31	10.65	12
27	30NC_25T_90A_0.5P	0.7305	4.99	3608.96	6.34	18.60	22	9.14	0.02	4.57	14.70	4.57	24.72	6.32	18.60	21
28	30NC_5T_95A_0.5P	0.0953	0.00	0.68	1.32	3.60	3	0.00	0.02	0.00	1.28	0.00	2.65	1.20	3.55	3
29	30NC_15T_95A_0.5P	0.2959	19.94	3600.31	3.11	11.85	9	0.00[a]	0.02	0.00[a]	8.87	0.00[a]	17.71	3.28	11.65	9
30	30NC_25T_95A_0.5P	0.5300	11.70	3604.59	4.99	20.00	16	0.00[a]	0.02	0.00[a]	15.34	0.00[a]	26.69	5.04	19.95	16
31	50NC_10T_90A_0.5P	0.1555	21.44	3603.12	2.04	7.95	8	0.00	0.13	0.00	24.78	0.00	35.38	2.04	7.80	8
32	50NC_20T_90A_0.5P	0.3360	16.98	3601.77	5.10	14.90	17	5.96	0.03	0.00[a]	25.09	0.00[a]	55.43	5.54	14.45	17
33	50NC_40T_90A_0.5P	0.7172	7.61	3600.65	8.89	31.05	36	5.59	0.14	2.79	33.79	0.00[a]	103.81	9.30	30.55	36
34	50NC_10T_95A_0.5P	0.1154	16.66	3602.33	1.74	8.20	6	0.00[a]	0.03	0.00[a]	19.48	0.00[a]	29.27	1.73	8.20	6
35	50NC_20T_95A_0.5P	0.2558	13.89	3600.59	4.15	15.85	13	0.00	0.02	0.00[a]	78.31	0.00[a]	103.38	4.32	15.50	13
36	50NC_40T_95A_0.5P	0.5366	9.50	3601.08	7.75	32.20	27	3.72	0.02	3.72	69.69	0.00[a]	164.64	7.58	32.25	27
37	30NC_5T_90A_RP	0.1291	0.00	94.82	1.75	3.25	4	26.15	0.02	0.28	2.00	0.28	3.90	1.49	3.35	4
38	30NC_15T_90A_RP	0.3967	21.83	3600.56	4.59	10.40	12	0.11	0.08	−8.33	11.86	−8.33	21.67	4.24	10.60	13
39	30NC_25T_90A_RP	0.7308	6.56	3600.78	6.25	18.75	22	9.18	0.02	0.00[a]	8.17	0.00[a]	18.37	6.63	18.35	22
40	30NC_5T_95A_RP	0.0957	0.00	11.83	1.35	3.50	3	0.41	0.01	0.41	1.59	0.41	3.32	1.20	3.65	3
41	30NC_15T_95A_RP	0.2964	22.29	3600.93	3.40	11.60	9	0.16	0.01	0.16	15.17	−11.15	24.01	3.29	11.65	10
42	30NC_25T_95A_RP	0.5304	15.53	3600.76	5.56	19.20	16	−6.24	0.01	−6.24	7.33	−6.24	20.67	5.12	19.85	17
43	50NC_10T_90A_RP	0.1559	24.45	3600.80	3.15	6.75	8	0.32	0.02	0.32	18.73	−12.57	28.39	2.41	7.55	9
44	50NC_20T_90A_RP	0.3364	19.19	3600.98	5.54	14.45	17	0.14	0.02	0.12	20.86	0.12	59.23	5.03	14.95	17
45	50NC_40T_90A_RP	0.6775	15.75	3602.07	10.92	29.00	34	−2.90	0.02	−2.90	76.89	−5.86	146.04	10.01	29.90	36
46	50NC_10T_95A_RP	0.1359	2.82	3600.86	2.12	7.85	7	15.10	0.02	0.31	5.58	0.31	15.31	1.82	8.15	7
47	50NC_20T_95A_RP	0.2562	17.33	3601.15	4.75	15.15	13	0.16	0.03	0.16	62.25	0.16	89.45	4.56	15.15	13
48	50NC_40T_95A_RP	0.5370	11.96	3601.08	9.40	30.55	27	0.07	0.02	0.06	31.47	−3.68	111.74	7.51	32.45	28

[a] The value falls within the $|\delta\%| < 0.03\%$ tolerance.

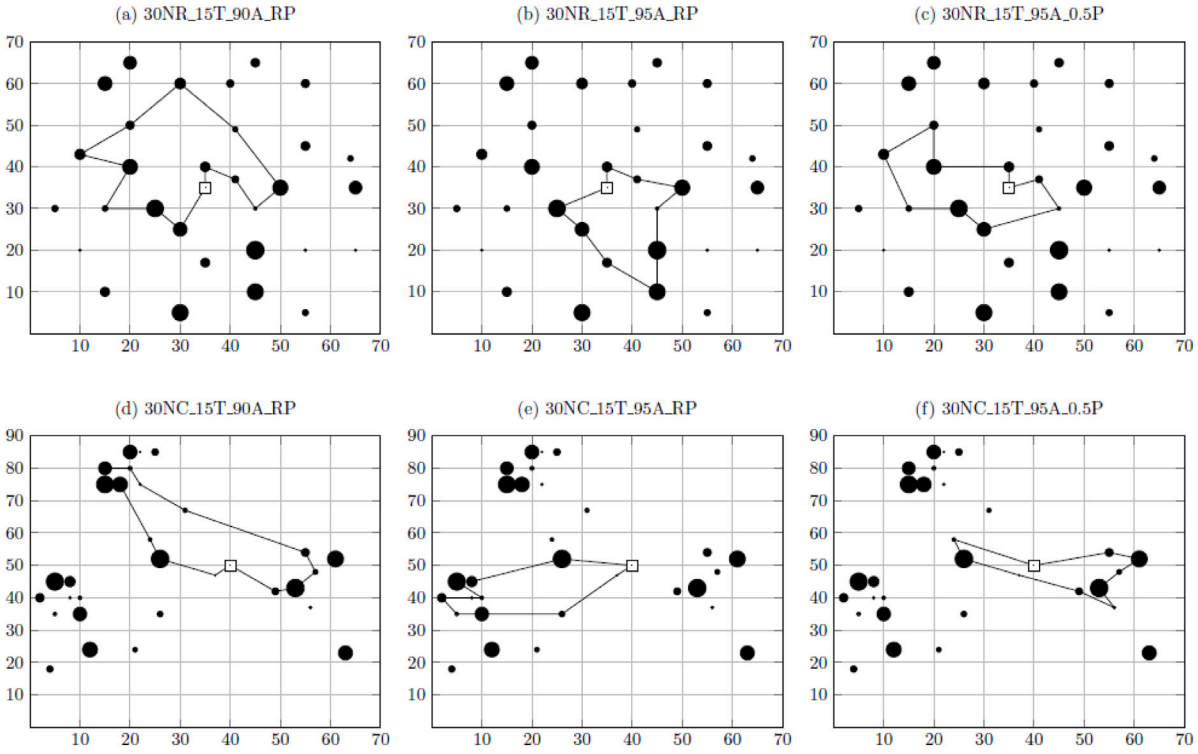

Fig. 4. Resulting routes for six hypothetical instances.

respectively. Clustered networks pose greater challenges for CPLEX, with only four out of 24 instances being optimally solved. Additionally, average $\Delta\%$ values for random and clustered networks are 2.61% and 12.55%, respectively. For 30-node instances, setting $\mathcal{T}max = 5$ or $\mathcal{T}max = 25$ tends to yield lower optimality gaps compared to $\mathcal{T}max = 15$. The benefit of $\mathcal{T}max = 5$ can be attributed to the smaller instance size, facilitating a more manageable node selection process. Conversely, with $\mathcal{T}_{max} = 25$, the node selection problem becomes easier due to the less stringent assessment time constraint. With 50-node instances, solution quality exhibits an inverse relationship with $\mathcal{T}_{max}$.

Comparing CH and CPLEX solutions highlights the CH's success in achieving strong results across many instances. According to Table 4, CH matches or exceeds CPLEX's objective value in 23 instances, with an average $\delta\%$ of 0.25% across all instances and a range of −8.33% to 4.79%. While CPLEX outperforms CH in 25 instances, the average $\delta\%$ for these is 1.18%. Moreover, CH completes each instance in just 27.10 s on average, compared to CPLEX's 42-minute average runtime, demonstrating its efficiency. However, there are instances where the CH is outperformed by the TS, which we explore further in Section 5.3. Despite its speed, the GA exhibits higher variability with an average $\delta\%$ of 3.03% across all instances, sometimes exceeding 25%. This variability suggests that the GA may lack robustness despite its efficiency.

A comparison of TS solutions with those of CPLEX underscores the effectiveness of TS in terms of solution quality and time. The average $\delta\%$ of TS solutions is −0.53% across all instances, with a range from −12.57% to 4.79%. Only five instances have a $\delta\%$ exceeding 1%. TS achieves equal or better objective values compared to CPLEX in 29 instances and outperforms CPLEX in 6 instances, with an average improvement of 7.97% over CPLEX in those cases. While CPLEX outperforms the TS heuristic in 19 instances, the average $\delta\%$ for these cases is only 1.17%. Notably, significant differences in objective values ($|\delta\%| > 1\%$) primarily arise from variations in the number of visited sites between the two solutions. Specifically, Table 4 illustrates that the CPLEX solution visits one more site than TS in five instances, while TS visits one more site in five instances and two more sites in one instance.

The TS algorithm performs better with 30 nodes (−0.61% average $\delta\%$) compared to 50 nodes (−0.46%) and on a clustered network (−1.75%) compared to a random one (0.69%). Additionally, CPLEX requires an average of 42 min per instance, whereas the TS heuristic finds a solution for each instance within 51.15 s on average. These results demonstrate that the TS approach not only achieves comparable or superior solutions compared to CPLEX but does so within 2% of its CPU time. This efficiency is particularly significant given the need for timely decisions in the context of the ARP-CA.

5.1.4. Effects of instance parameters on the objective values and route structures

To understand the effect of obtaining initial estimates on population needs, we compare instances with no initial prevalence estimates (denoted by 0.5P) to those with initial estimates (denoted by RP) in Table 4. For instances without initial estimates (rows 1–12 and 25–36), where $\hat{p} = 0.5$ for all nodes, the average objective value is 2.77% lower compared to instances with initial estimates (rows 13–24 and 37–48). This difference underscores the value of incorporating initial information into decision-making processes. When initial estimates are available, fewer interviews are needed to achieve the same confidence level compared to instances without initial estimates. Moreover, the routing can prioritize visiting nodes with lower initial accuracy when initial estimates are available, resulting in overall better accuracy levels.

We analyze the impact of network structure by comparing instances with random networks (denoted by NR) to those with clustered networks (denoted by NC) in Table 4. On clustered networks (rows 25–48), the average objective value is 3.89% higher than on random networks (rows 1–24). This improvement primarily stems from travel time savings due to shorter distances within a cluster, resulting in better coverage. The most significant improvement is observed under a 90%

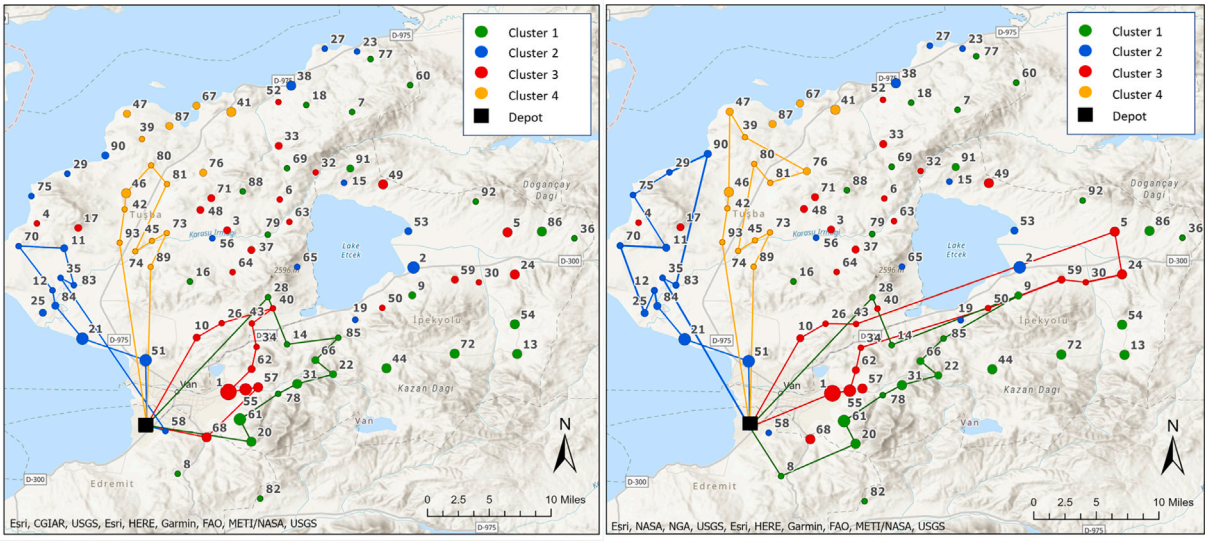

Fig. 5. Spatial representation of resulting routes for case study (a) instances #1,5,9 and 13 (b) instances #2,6,10 and 14.

confidence level (denoted by 90 A) compared to 95% (denoted by 95 A). Under the 90% confidence level, the reduced requirement for interview time increases the average number of sites visited from 12.46 to 16.42, resulting in a 32.75% increase in the objective value. This highlights the important trade-off between site coverage and sampling accuracy; achieving more accurate information requires sacrificing the number of sites visited, and vice versa.

To analyze the impact of varying initial estimates, network structures, and confidence levels on resulting routes, Fig. 4 presents TS results for six different instances with 30 nodes and $\mathcal{T}_{max} = 15$. Comparing (a) and (b), we observe a decrease in the number of visited nodes from 12 to 9, attributed to the increased confidence level. Additionally, as more interview time is required for each confidence level, the route in (b) traverses a denser part of the network to minimize travel times. Instances (b) and (c), differing in initial estimate structure, visit the same number of nodes, but four of the visited nodes differ. This is mainly due to route preference towards nodes with wider CI half-widths for initial estimates in (c). Similar comparisons can be drawn for clustered networks in (d) through (f).

5.2. Results: Case study

We illustrate our approach to a case study based on the 2011 Van Earthquake in Turkey, one of the most catastrophic disasters of that decade in the country, resulting in more than 600 casualties, 4000 injuries, and extensive damage to thousands of building. Numerous national and international humanitarian organizations participated in the relief efforts. We generate several instances using real data from the earthquake-affected area (Section 5.2.1), present results illustrating the performance of our algorithm (Section 5.2.2), and provide insightful observations based on the solutions obtained (Section 5.2.3).

5.2.1. Case dataset

The case study instances are generated using datasets from Noyan et al. [55] and Balcik [3], incorporating data on the coordinates of 93 affected sites, their distances to the earthquake's epicenter, travel times, and geographical features such as elevation and distance to lakes. Each site is assigned equal priority and weight levels for these experiments. More information about this dataset is in Appendix E.

To generate case instances for the ARP-CA, we employ a clustering approach based on geographic features of the sites such as elevation, distance to the lake, and distance to the earthquake's epicenter. Clustering sites in advance can be practical for capturing potential differences in needs between and within clusters. Additionally, if multiple assessment teams are involved, grouping similar sites can facilitate planning tasks and shifts for the teams. Various clustering methods, such as k-means clustering, density-based spatial clustering, mean shift clustering, and agglomerative hierarchical clustering, can be applied depending on the type of available data (e.g., continuous, binary, etc.). In this study, we implement a k-means clustering algorithm, which is a simple and well-known method [56]. The cluster size is determined based on the ratio of within-cluster variance to between-cluster variance. Using this approach on our case dataset, we obtain four clusters defined by the first break point with a smaller within-cluster variance-to-between-cluster variance ratio (see Appendix F). These four clusters, illustrated in Fig. 5, consist of 28, 22, 28, and 15 nodes, respectively.

To obtain further instances of different sizes, we consider instances, where (i) the first two clusters are merged (resulting in 50 nodes), (ii) the last two clusters are merged (resulting in 43 nodes), and (iii) where all clusters are merged (resulting in 93 nodes). In result tables, the first part of the instance names reflects the clusters, i.e., 1C, 2C, 3C and 4C to identify the cases with a single cluster, 12C and 34C to denote the merger of two clusters, and AC to include all four clusters. For the first four cases, we vary the maximum assessment time as 8 and 10 h, whereas with two merged clusters we use 12 and 18 h. When all clusters are merged, the time limit is varied as 24 and 36 h. In all instances, interview time per person is set as 0.05 h. We set the confidence level at 90% or 95%, resulting in a total of 28 instances (see Table 5).

5.2.2. Assessment of algorithm performance

We first evaluate the alternative algorithms' performances (i.e., CPLEX, CH, and TS) across various case instances, detailed in Table 5. Notably, CPLEX achieves optimal solutions in only 14 out of 28 instances. In the remaining cases, where an optimal solution is not attained, the optimality gaps can be as substantial as 9.23%. Importantly, none of the instances with 43 nodes or more can be solved optimally using CPLEX. Additionally, we observe that instances with lower total assessment time limits tend to exhibit larger optimality gaps. These observations confirm our earlier findings that the solution quality of CPLEX decreases as the size of the network increases and the $\mathcal{T}_{max}$ value decreases.

The comparison of CH and TS results with CPLEX (Table 5) highlights the TS heuristic's ability to achieve near-identical solutions to

Table 5
Objective and gap values, CPU times (in seconds), travel and interview times, and number of sites visited for the Van Earthquake Case Study instances using CPLEX, greedy approach (GA), constructive heuristic (CH), and the TS algorithm.

		CPLEX							GA		CH		TS					
Inst. #	Instance	Obj.	Δ%	CPU time	Trav. time	Int. time	Sites vis.	Total # of int.s	δ_{GA}%	CPU time	δ_{CH}%	CPU time	δ_{TS}%	CPU time	Trav. time	Int. time	Sites vis.	Total # of int.s
1	1C_28N_8T_90A_0.5P	0.3532	0.00	19.66	1.93	5.10	10	153	10.12	0.11	20.27	12.12	0.00	21.20	2.26	5.67	10	170
2	1C_28N_10T_90A_0.5P	0.4249	1.15	3600.10	2.34	5.93	12	178	8.43	0.02	8.43	12.58	0.00	18.93	3.11	6.80	12	204
3	1C_28N_8T_95A_0.5P	0.2816	0.00	2.58	1.65	6.37	8	188	12.72	0.28	25.44	7.31	0.00	12.03	1.63	6.27	8	188
4	1C_28N_10T_95A_0.5P	0.3532	0.00	18.08	2.10	7.87	10	236	10.13	0.09	10.11	12.25	0.00	17.48	2.12	7.87	10	236
5	2C_22N_8T_90A_0.5P	0.4054	0.00	2519.70	2.18	5.80	9	174	11.24	0.03	0.00[a]	4.98	0.00[a]	8.00	2.37	5.57	9	167
6	2C_22N_10T_90A_0.5P	0.5421	0.00	32.32	2.98	7.00	12	210	8.40	0.14	0.00	2.47	0.00	5.50	3.03	6.93	12	208
7	2C_22N_8T_95A_0.5P	0.3142	0.00	726.60	1.98	6.00	7	180	0.00[a]	0.02	0.00[a]	3.39	0.00[a]	6.06	2.02	5.97	7	179
8	2C_22N_10T_95A_0.5P	0.4054	0.00	1481.90	2.18	7.80	9	234	0.00[a]	0.02	0.00[a]	5.91	0.00[a]	9.05	2.22	7.77	9	233
9	3C_28N_8T_90A_0.5P	0.3174	0.00	39.79	2.25	5.73	9	172	0.00	0.02	0.00	7.56	0.00	11.54	2.28	5.70	9	171
10	3C_28N_10T_90A_0.5P	0.3891	3.42	3600.28	2.55	6.93	11	208	0.00[a]	0.02	0.00	11.48	0.00	17.13	3.06	6.93	11	208
11	3C_28N_8T_95A_0.5P	0.2458	0.00	154.01	1.74	6.23	7	187	0.00[a]	0.02	0.00[a]	8.16	0.00[a]	13.12	1.76	6.23	7	187
12	3C_28N_10T_95A_0.5P	0.3174	0.00	133.87	2.36	7.60	9	228	0.00	0.02	0.00	8.01	0.00	13.88	2.25	7.70	9	231
13	4C_15N_8T_90A_0.5P	0.5969	0.00	1636.97	2.26	5.73	9	172	0.00	0.11	0.00	4.55	0.00	6.52	2.26	5.73	9	172
14	4C_15N_10T_90A_0.5P	0.7975	0.00	133.82	2.82	7.17	12	215	0.00[a]	0.02	0.00[a]	2.64	0.00[a]	4.51	2.81	7.17	12	215
15	4C_15N_8T_95A_0.5P	0.4631	0.00	46.81	1.94	6.03	7	181	0.00	0.02	0.00	5.29	0.00	12.60	1.94	6.00	7	180
16	4C_15N_10T_95A_0.5P	0.5969	0.00	231.10	2.28	7.70	9	231	0.00	0.03	0.00	5.37	0.00	7.80	2.30	7.70	9	231
17	34C_43N_12T_90A_0.5P	0.3449	9.23	3600.55	2.84	8.73	15	175	20.28	0.06	0.00	12.08	0.00	38.03	3.66	8.33	15	167
18	34C_43N_18T_90A_0.5P	0.5315	4.86	3602.69	4.80	13.20	23	264	17.55	0.03	0.00[a]	20.44	0.00[a]	56.84	5.13	12.87	23	257
19	34C_43N_12T_95A_0.5P	0.2749	5.32	3600.76	2.75	9.17	12	183	16.96	0.11	0.00[a]	28.83	0.00[a]	52.96	2.82	9.17	12	183
20	34C_43N_18T_95A_0.5P	0.4149	5.98	3600.79	4.08	13.90	18	278	16.87	0.09	5.62	79.76	5.62	118.35	5.00	12.90	17	258
21	12C_50N_12T_90A_0.5P	0.2959	9.13	3601.63	3.22	8.77	15	175	6.77	0.03	0.00	19.65	0.00	48.18	3.62	8.37	15	167
22	12C_50N_18T_90A_0.5P	0.4564	7.02	3601.00	4.59	13.40	23	268	4.40	0.02	0.00[a]	25.15	0.00[a]	80.12	5.19	12.77	23	255
23	12C_50N_12T_95A_0.5P	0.2357	5.20	3600.95	2.63	9.17	12	183	8.49	0.03	8.49	47.06	0.00[a]	88.72	2.81	9.17	12	183
24	12C_50N_18T_95A_0.5P	0.3561	6.09	3601.03	3.93	13.97	18	279	5.64	0.03	0.00[a]	33.19	0.00[a]	113.21	4.08	13.87	18	277
25	AC_93N_24T_90A_0.5P	0.3509	5.00	3605.33	5.20	18.80	33	376	6.14	0.22	3.06	189.45	0.00[a]	988.28	5.42	18.47	33	369
26	AC_93N_36T_90A_0.5P	0.5343	3.14	3606.21	8.01	27.97	50	559	4.04	0.16	4.03	141.42	2.01	2186.82	8.66	27.30	49	546
27	AC_93N_24T_95A_0.5P	0.2646	7.80	3607.29	4.33	19.63	25	393	8.14	0.16	0.00[a]	308.70	0.00[a]	735.21	4.65	19.33	25	387
28	AC_93N_36T_95A_0.5P	0.4048	5.10	3603.38	6.52	29.47	38	589	2.65	0.09	2.65	498.42	0.00[a]	2049.00	6.91	29.07	38	581

[a] The value falls within the $|\delta\%| < 0.03\%$ tolerance.

CPLEX in nearly all instances, with small gaps of 2.01% and 5.62% in only two instances. This underscores the high-quality performance of the TS approach. Both CPLEX and TS visit a similar number of sites (15.43 versus 15.36 on average) and have comparable interview times (10.40 h versus 10.27 h on average), although TS spends slightly more time on travel (0.25 h more than CPLEX's average of 3.16 h). CH also produces high-quality solutions, with an average δ_{CH}% of 4.02% and 0.00% for instances 5–16. However, for instances corresponding to the first cluster, δ_{CH}% varies significantly between 8.43% and 25.44%. This variability suggests that the TS heuristic can complement CH effectively. In contrast, the GA exhibits lower robustness, with an average δ_{GA}% of 6.39% and exceeding 10% in eight instances.

The TS algorithm demonstrates superior efficiency compared to CPLEX across the case study instances. Among the 14 instances that are solved by both methods within an hour, CPLEX requires an average of 512 s, whereas TS solves them in just 10 s on average. Moreover, TS solves all 28 instances in an average of only six minutes, while CPLEX fails to solve half of them within the same timeframe. These results highlight the significant advantage of the TS heuristic in providing timely solutions for the ARP-CA.

While TS is superior compared to CH in terms of solution quality, CH is simple and intuitive. Therefore, we explore settings where CH can yield promising results as a standalone algorithm. Our observations from algorithm solutions highlight two key factors that significantly influence the quality of CH solutions: network topology and population distribution among affected sites.

First, the initial site selection for route initialization is an important factor affecting CH solutions. During this step, the algorithm assigns a desirability score to each node based on its proximity and the required interview time, considering other nodes in its impact region. This selection process is particularly crucial in clustered networks, where the CH tends to exhibit better performance with random networks compared to clustered ones. Additionally, if the selection of promising nodes and the cheapest insertion steps (Algorithm 2) favor a cluster not containing nodes optimal for the solution, the CH may generate a suboptimal outcome. Moreover, the number of nodes in each cluster impacts the CH's success. Specifically, if the total assessment time limit ($\mathcal{T}_{max}$) is large enough to visit more nodes, the assessment team completes visiting all nodes in the initially selected cluster and moves to another. Given the longer distance between nodes from different clusters compared to those within a cluster, the team travels further to reach another cluster within the time limit, significantly increasing travel time. From our case results, we have also observed that geographical features like lakes and elevation can restrict movement within the affected area. These constraints may lead the CH algorithm to produce sub-optimal solutions, as it may struggle to incorporate additional sites into the route under such conditions.

Secondly, population distribution also affects the CH solution. We observe that sites with smaller populations tend to be more appealing for visits due to sampling accuracy considerations. The specific locations of these smaller nodes affect the initial solutions identified by the CH. If these nodes are located close to the depot, they become very attractive candidates for the CH in terms of both the coverage and accuracy objectives. However, his can lead the CH algorithm to settle for a local optimum solution, with an initial route centered around the depot and including these promising nodes, potentially overlooking better overall solutions.

Although our results show that implementing CH and TS algorithms are highly beneficial in finding high-quality and near optimal solutions,

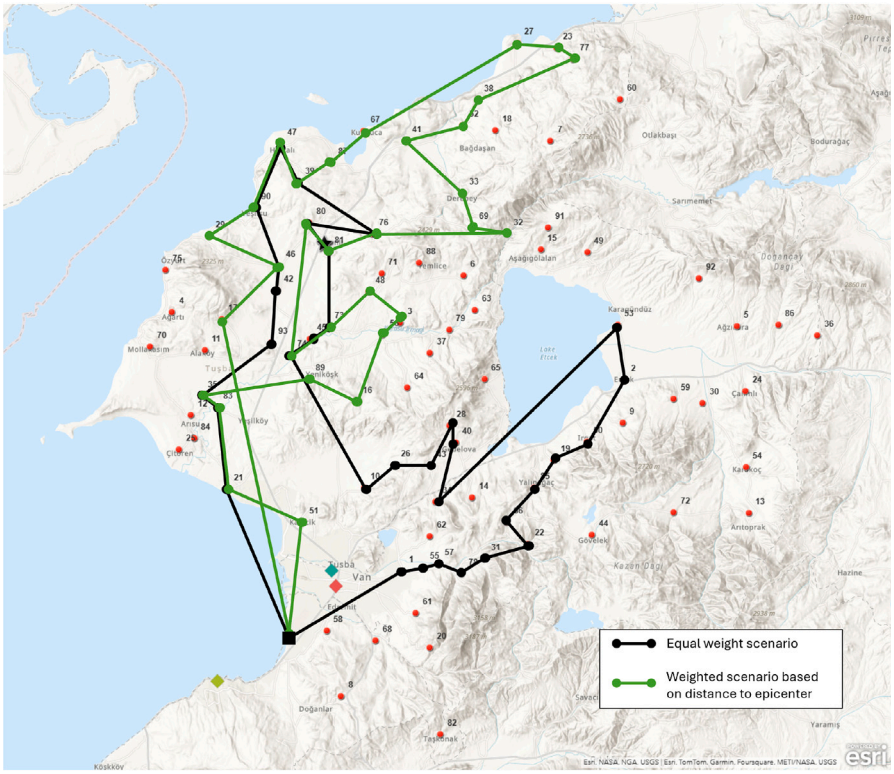

Fig. 6. Resulting routes for equal weight versus distance-based weighted models for Case Study instance #26.

if humanitarian agencies suffer from access to optimization software, a simple GA would be a practical alternative which seem to provide reasonable solutions for the ARP-CA. However, our results show that such a simple approach may lack evaluating the sites by their attractiveness for both coverage and sampling accuracy objectives. Therefore, agencies may benefit from the advantages of the proposed CH and TS algorithms, which provide reasonable solutions. In particular, the TS algorithm provides consistently high-quality solutions within minutes. Furthermore, the TS approach requires the solution of a very simple mixed integer program that solves in less than a second. In the absence of commercial optimization software, the embedded model in TS can be easily employed by free solvers.

5.2.3. Solution analysis

In this section, we present solutions of ARP-CA and discuss the effects of assessment time limit, network topology and confidence level choice on ARP-CA solutions obtained by the TS algorithm.

Example routes from case instances are shown in Fig. 4. To analyze the impact of assessment time limits, we compare instances with different $\mathcal{T}max$ values from each cluster. For example, comparing instances 1, 5, 9, and 13 with $\mathcal{T}max = 8$ to instances 2, 6, 10, and 14 with $\mathcal{T}_{max} = 10$, we observe that increasing the maximum assessment time from 8 to 10 h leads to a larger number of visited nodes from an average of 9.25 to 11.75 and from 170 to 208.75 interviews. Additionally, the routes with a higher assessment time can cover a wider area of the network and reach more distant sites from the starting point. In our experiments, the resulting routes tend to prioritize sites closer to the origin to optimize coverage and accuracy, assuming equal priority and weights for each site. However, decision-makers could prioritize differently to target more distant sites.

Comparing instances from different clusters (Fig. 5) highlights the impact of network structure on solutions. For clusters 2 and 4, where sites are near the lake or far from the origin, routes in (a) tend to form an arrow-like shape. In contrast, routes for clusters #1 and #3, where sites are closer together and to the origin, form a circular shape due to the network structure. Despite the origin's distant location in cluster 4, the dense concentration of sites in this cluster enables efficient routes without sacrificing sampling accuracy, visiting almost the same number of sites as other clusters. Additionally, network topology can limit travel options within sites, as seen in the route for cluster 3 in (b), where the region's topology forces the assessment team to choose a path either above or below Lake Ercek.

Finally, our case study solutions (Table 4) highlight the significant effect of confidence level choices on interview durations and, consequently, the number of sites visited. Specifically, increasing the confidence level from 90% to 95% requires, on average, an additional 18.43 interviews, resulting in a decrease of 3.86 sites visited within the same assessment time limit. This underscores the critical impact of parameter selection on assessment plans, highlighting the importance of carefully balancing sampling accuracy and coverage objectives during the planning phase.

5.2.4. Sensitivity analysis

In this section, we perform a sensitivity analysis on the case study instances to evaluate the impact of key parameters on ARP-CA solutions. Specifically, the analysis examines the effects of varying weighted coverage objectives and initial parameter values.

The first analysis focuses on how prioritizing sites near the epicenter within the coverage objective influences the solutions. Weighted coverage objectives are employed to emphasize specific regions during the site selection process. Following the methodology of Noyan et al. [55], the 93 sites are categorized into three distance-based groups using two threshold sets: (a) less than 7 km, (b) 7–16 km, and (c) more than 16 km, comprising 4, 13, and 76 sites, respectively, and (a) less than 10 km, (b) 10–20 km, and (c) more than 20 km, comprising 5, 26, and 62 sites, respectively.

For each grouping, two weight schemes are tested to prioritize sites closer to the epicenter: 0.6–0.3-0.1 and 0.5–0.3-0.2. These weight

Table 6
TS Heuristic Sensitivity to different weights based on distance to epicenter: Percent differences in travel time, interview time, and number of sites visited for the Van Earthquake Case Study instances.

		W1 (7 km–16 km|0.6-0.3-0.1)			W2 (7 km–16 km|0.5–0.3-0.2)			W3 (10 km–20 km|0.6–0.3-0.1)			W4 (10 km–20 km|0.5–0.3-0.2)		
Inst. #	Instance	Δ% Trav. time	Δ% Int. time	Δ% Sites vis.	Δ% Trav. time	Δ% Int. time	Δ% Sites vis.	Δ% Trav. time	Δ% Int. time	Δ% Sites vis.	Δ% Trav. time	Δ% Int. time	Δ% Sites vis.
1	1C_28N_8T_90A_0.5P	0.52	−0.06	0.00	−3.88	2.29	0.00	66.78	−27.10	−30.00	−3.56	2.29	0.00
2	1C_28N_10T_90A_0.5P	−0.01	0.00	0.00	11.11	−3.92	−8.33	12.80	−4.90	−8.33	2.75	0.00	0.00
3	1C_28N_8T_95A_0.5P	5.23	−0.05	0.00	−7.26	3.14	−12.50	85.33	−21.32	−25.00	1.96	−0.05	0.00
4	1C_28N_10T_95A_0.5P	29.45	−9.78	−10.00	8.49	−2.16	−10.00	32.52	−10.21	−10.00	12.09	−3.43	−10.00
5	2C_22N_8T_90A_0.5P	12.12	−4.85	0.00	9.02	−3.05	0.00	−6.46	3.53	0.00	−6.46	3.53	0.00
6	2C_22N_10T_90A_0.5P	5.29	−1.88	0.00	5.29	−1.88	0.00	20.32	−9.57	−8.33	0.75	0.05	0.00
7	2C_22N_8T_95A_0.5P	23.47	−7.87	0.00	23.47	−8.43	0.00	25.52	−8.99	0.00	26.83	−10.11	0.00
8	2C_22N_10T_95A_0.5P	17.91	−5.19	0.00	24.24	−6.91	0.00	39.34	−11.20	0.00	27.92	−8.19	0.00
9	3C_28N_8T_90A_0.5P	17.11	−8.19	0.00	22.77	−8.77	0.00	8.19	−2.92	0.00	19.58	−8.19	0.00
10	3C_28N_10T_90A_0.5P	2.48	−1.39	0.00	12.10	−5.72	0.00	13.11	−5.72	0.00	43.72	−19.19	−9.09
11	3C_28N_8T_95A_0.5P	30.59	−8.51	0.00	11.63	−4.76	0.00	89.03	−25.63	−14.29	25.71	−7.44	0.00
12	3C_28N_10T_95A_0.5P	9.63	−3.03	0.00	26.78	−8.23	0.00	64.53	−18.61	−11.11	28.28	−9.09	0.00
13	4C_15N_8T_90A_0.5P	−0.07	0.06	0.00	14.32	−6.34	0.00	26.16	−11.58	0.00	14.32	−6.34	0.00
14	4C_15N_10T_90A_0.5P	17.19	−6.56	0.00	10.52	−4.23	0.00	11.72	−4.70	0.00	10.52	−4.23	0.00
15	4C_15N_8T_95A_0.5P	24.80	−10.00	0.00	27.31	−10.56	0.00	24.48	−9.44	0.00	27.08	−7.78	0.00
16	4C_15N_10T_95A_0.5P	24.58	−7.36	0.00	28.96	−9.52	0.00	23.04	−6.93	0.00	24.58	−7.36	0.00
17	34C_43N_12T_90A_0.5P	12.04	−5.96	−6.67	13.72	−5.96	−6.67	10.86	−5.96	−6.67	12.90	−6.36	−6.67
18	34C_43N_18T_90A_0.5P	10.79	−4.43	−4.35	−1.62	0.23	0.00	21.20	−8.57	−8.70	9.71	−4.17	−4.35
19	34C_43N_12T_95A_0.5P	28.28	−9.12	−8.33	32.04	−11.30	−16.67	47.04	−14.58	−16.67	26.03	−8.40	−8.33
20	34C_43N_18T_95A_0.5P	−22.13	9.30	0.00	5.63	−1.55	−5.88	−5.49	2.58	0.00	−7.19	3.10	0.00
21	12C_50N_12T_90A_0.5P	0.68	−0.44	0.00	−1.96	−0.44	0.00	12.88	−6.01	−6.67	−8.59	3.15	0.00
22	12C_50N_18T_90A_0.5P	9.56	−3.68	−4.35	−3.82	1.54	0.00	−0.61	0.50	0.00	9.74	−3.68	−4.35
23	12C_50N_12T_95A_0.5P	10.75	−4.03	−8.33	16.46	−5.49	−8.33	2.44	−0.76	−8.33	25.76	−8.76	−8.33
24	12C_50N_18T_95A_0.5P	16.39	−4.59	−5.56	−2.06	0.94	0.00	14.30	−3.87	−5.56	14.45	−4.59	−5.56
25	AC_93N_24T_90A_0.5P	11.77	−3.09	−3.03	12.47	−3.09	−3.03	19.11	−5.43	−6.06	12.42	−3.09	−3.03
26	AC_93N_36T_90A_0.5P	−2.39	0.61	0.00	3.46	−0.98	−2.04	5.40	−1.83	−2.04	11.86	−3.79	−4.08
27	AC_93N_24T_95A_0.5P	−5.00	0.88	0.00	2.52	−1.19	0.00	9.78	−2.40	−4.00	16.09	−3.95	−4.00
28	AC_93N_36T_95A_0.5P	0.67	−0.13	0.00	5.30	−1.27	−2.63	8.26	−2.30	−2.63	9.82	−2.30	−2.63
Average		**10.42**	**−3.55**	**−1.81**	**10.97**	**−3.84**	**−2.72**	**24.34**	**−8.00**	**−6.23**	**13.90**	**4.58**	**−2.51**

schemes assign higher importance to sites in closer proximity while reducing emphasis on more distant locations, enabling a detailed assessment of how weighting strategies affect site selection outcomes.

The results from the TS heuristic across all four scenarios are summarized in Table 6, which reports percentage differences (Δ%) from the equal-weight scenario in terms of travel time, interview time, and the number of sites visited. The findings demonstrate that assigning higher weights to specific regions significantly affects site selection and the resulting routes. For instance, prioritizing closer sites increases their likelihood of selection but leads to an average 14.91% increase in travel time compared to the equal-weight scenario. This increase in travel time results in a 4.99% average reduction in time available for interviews and a 3.32% average decrease in the number of sites visited by assessment teams.

These results highlight the trade-offs in assigning weights to prioritize certain areas. Assigning higher weights to sites near the epicenter ensures enhanced coverage in regions likely to have sustained greater damage. However, if the objective is to gather information from a more diverse set of regions, an equal-weight approach may be more effective as it provides broader geographic coverage and diversity in site selection. Fig. 6 illustrates the changes in routes between a weighted scenario and an equal-weight scenario, further highlighting these trade-offs.

We further analyze the model's sensitivity to variations in initial prevalence values, considering the assumption that assessment teams may have more reliable prior information about needs in areas closer to district centers. Similar to the grouping approach used for weights, we divide the sites into three categories—high, medium, and low prior information—based on distance thresholds from the district center: (a) less than 7 km, (b) 7–16 km, and (c) more than 16 km. These groups contain 2, 13, and 78 sites, respectively.

To evaluate the effect of prior information, two prevalence combinations are tested: 0.9–0.7-0.5 and 0.7–0.6-0.5. In these scenarios, sites located more than 16 km from the district center are assumed to have no prior information, while higher levels of prior information are assigned to groups closer to the district center.

The results obtained from the TS heuristic for these prevalence combinations are summarized in Table 7, showing the percentage differences (Δ%) relative to the scenario with no prior information. Metrics include the objective function value, travel time, interview time, and the number of sites visited. The findings indicate that as the level of prior information increases, sites with higher prior prevalence values are more likely to be selected in the final solution. This is due to the smaller sample sizes required to achieve the same precision levels for these sites. Specifically, the results show an average increase in the objective value of 3.84% for the lower-information scenario (P1) and 5.09% for the higher-information scenario (P2), highlighting the positive impact of incorporating prior information. This improvement arises from either enhancing precision within the same coverage level or enabling visits to additional sites due to reduced interview times. Additionally, under scenario P2, the number of sites visited increases by an average of 4.25%.

These findings highlight the benefits of leveraging reliable prior information in the assessment process. By reducing the number of interviews required to meet precision goals, assessment teams can allocate their resources more effectively, achieving better coverage and higher precision across the selected sites.

Table 7
TS Heuristic Sensitivity to different initial prevalence values based on distance to district center: Percent differences in objective value, travel time, interview time, and number of sites visited for the Van Earthquake Case Study instances.

		P1 (0.5–0.6-0.7)				P2 (0.5–0.7-0.9)			
Inst. #	Instance	Δ% Obj	Δ% Trav. time	Δ% Int. time	Δ% Sites vis.	Δ% Obj	Δ% Trav. time	Δ% Int. time	Δ% Sites vis.
1	1C_28N_8T_90A_0.5P	−0.05	4.62	−0.65	0.00	−0.05	9.88	−3.00	0.00
2	1C_28N_10T_90A_0.5P	16.77	4.30	−0.98	−8.33	8.41	3.70	−1.47	8.33
3	1C_28N_8T_95A_0.5P	−0.06	8.28	−1.65	0.00	−0.06	51.64	−12.28	0.00
4	1C_28N_10T_95A_0.5P	−0.05	−0.07	−1.31	0.00	−0.04	25.64	−6.86	0.00
5	2C_22N_8T_90A_0.5P	12.13	6.83	−2.45	0.00	11.25	5.49	−1.26	11.11
6	2C_22N_10T_90A_0.5P	0.00	−1.55	1.01	0.00	0.00[a]	2.87	−0.91	0.00
7	2C_22N_8T_95A_0.5P	0.00	2.82	−1.17	0.00	14.51	18.50	−6.20	14.29
8	2C_22N_10T_95A_0.5P	33.64	6.89	−2.62	0.00	11.24	14.65	−4.33	11.11
9	3C_28N_8T_90A_0.5P	11.26	0.15	−1.17	0.00	11.29	15.45	−5.85	11.11
10	3C_28N_10T_90A_0.5P	9.19	7.39	−3.80	9.09	9.20	19.89	−9.57	9.09
11	3C_28N_8T_95A_0.5P	14.54	−2.07	0.59	0.00	14.57	21.10	−6.90	14.29
12	3C_28N_10T_95A_0.5P	0.00	0.10	0.43	0.00	11.28	16.99	−4.76	11.11
13	4C_15N_8T_90A_0.5P	0.00	−0.07	0.06	0.00	0.00[a]	14.32	−6.34	0.00
14	4C_15N_10T_90A_0.5P	0.00	−0.17	−0.05	0.00	0.00	0.00	0.00	0.00
15	4C_15N_8T_95A_0.5P	0.00	−0.14	0.00	0.00	0.00[a]	−0.14	0.00	0.00
16	4C_15N_10T_95A_0.5P	0.00	−0.01	0.00	0.00	0.00	−4.35	0.00	0.00
17	34C_43N_12T_90A_0.5P	0.00	−10.25	4.04	0.00	0.00	−9.84	4.44	0.00
18	34C_43N_18T_90A_0.5P	0.00[a]	−0.58	−0.03	0.00	0.00[a]	−0.78	0.00	0.00
19	34C_43N_12T_95A_0.5P	−0.04	−3.73	1.05	−8.33	0.00[a]	3.40	−1.49	0.00
20	34C_43N_18T_95A_0.5P	5.95	−14.27	6.25	5.88	5.95	−13.76	5.94	5.88
21	12C_50N_12T_90A_0.5P	0.00	−5.97	2.35	0.00	13.53	−0.95	0.36	0.00
22	12C_50N_18T_90A_0.5P	4.36	−6.37	2.85	0.00	4.38	3.73	−1.33	4.35
23	12C_50N_12T_95A_0.5P	0.00[a]	−6.44	1.78	0.00	8.47	4.27	−1.09	0.00
24	12C_50N_18T_95A_0.5P	0.00[a]	−6.08	1.66	0.00	0.00[a]	−3.68	1.44	0.00
Average		**3.84**	**−0.09**	**0.10**	**−0.06**	**5.09**	**7.83**	**−2.42**	**4.25**

[a] The value falls within the $|\delta\%| < 0.03\%$ tolerance.

5.3. Discussion on implementation and practical considerations

The successful adoption of the ARP-CA framework and the proposed heuristic solution approach in practice necessitates a well-structured integration into existing assessment workflows. Additionally, effective operational adaptation is crucial to align the approach with available resources and contextual constraints. This section outlines key practical considerations for incorporating the proposed approach into needs assessment practices.

Data availability and technology integration. The effective application of the ARP-CA framework depends on leveraging both pre-disaster secondary data and real-time field data. Pre-disaster data, such as demographic profiles, socio-economic vulnerabilities, infrastructure maps, and historical disaster impacts, provides a crucial baseline for identifying high-risk areas and establishing assessment priorities. However, challenges such as data fragmentation, outdated records, and restricted access can hinder the effective utilization of this information. Establishing centralized, regularly updated databases, GIS-based platforms, and automated data pipelines can significantly improve data availability and reliability. Additionally, fostering open data-sharing policies among government agencies and humanitarian organizations can enhance coordination efforts and streamline data preparation, further supporting the successful implementation of the proposed framework.

In addition to pre-disaster data management, timely updates to critical datasets are essential following a disaster to capture evolving conditions. Information such as infrastructure status, risk maps, service availability, and population displacement must be continuously updated to ensure accurate situational awareness. However, disrupted communication networks, data overload, and resource constraints often delay these updates, limiting effective decision-making. To address these challenges, integrating remote sensing technologies (e.g., satellite imagery, drones), mobile reporting systems, local government reports, and community feedback can improve data accuracy and ensure timely updates. Furthermore, a robust technological infrastructure is crucial to facilitate efficient data collection, integration, and analysis. GPS-enabled devices, mobile assessment applications, and cloud-based data management systems enhance data accuracy and streamline workflows. Additionally, offline data collection tools are essential for maintaining functionality in low-connectivity regions, ensuring critical information can still be gathered and utilized despite connectivity challenges.

Resource management. Effective post-disaster assessment requires strategic resource management to maximize efficiency and coverage. Resource availability (e.g., personnel, vehicles, equipment) directly affects team formation, assessment duration, and the number of sites that can be visited within given time constraints. In practice, several critical factors must be considered to enhance resource management. Vehicle suitability is a key concern, as transportation in disaster-affected areas may require off-road capabilities or specialized vehicles to navigate damaged infrastructure, debris, or challenging terrain. Assessing vehicle conditions and ensuring their readiness for field deployment is essential to avoid delays and access remote or isolated areas. Another important consideration is team familiarity with the field. Assigning teams with prior knowledge of the affected region can improve navigation efficiency, enhance communication with local stakeholders, and facilitate culturally sensitive interactions. Our two-phase approach supports taking into account these contextual constraints that appear on the ground. Dividing large regions into smaller subregions and establishing predefined clusters can simplify planning and team assignment.

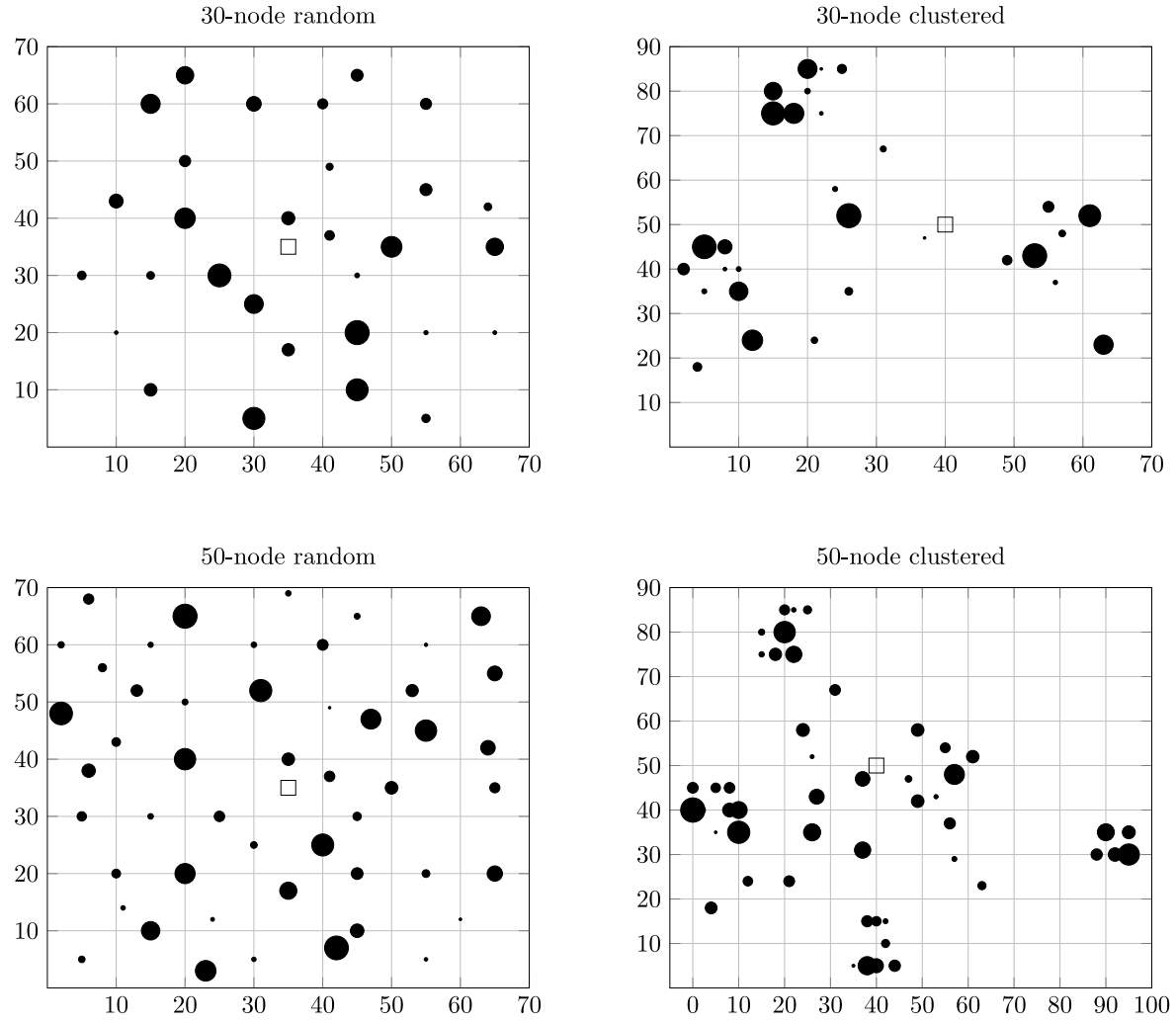

Fig. C.7. The four networks used for randomly generated instances, where the size of each node is directly proportional to its population and the depot node is shown in square.

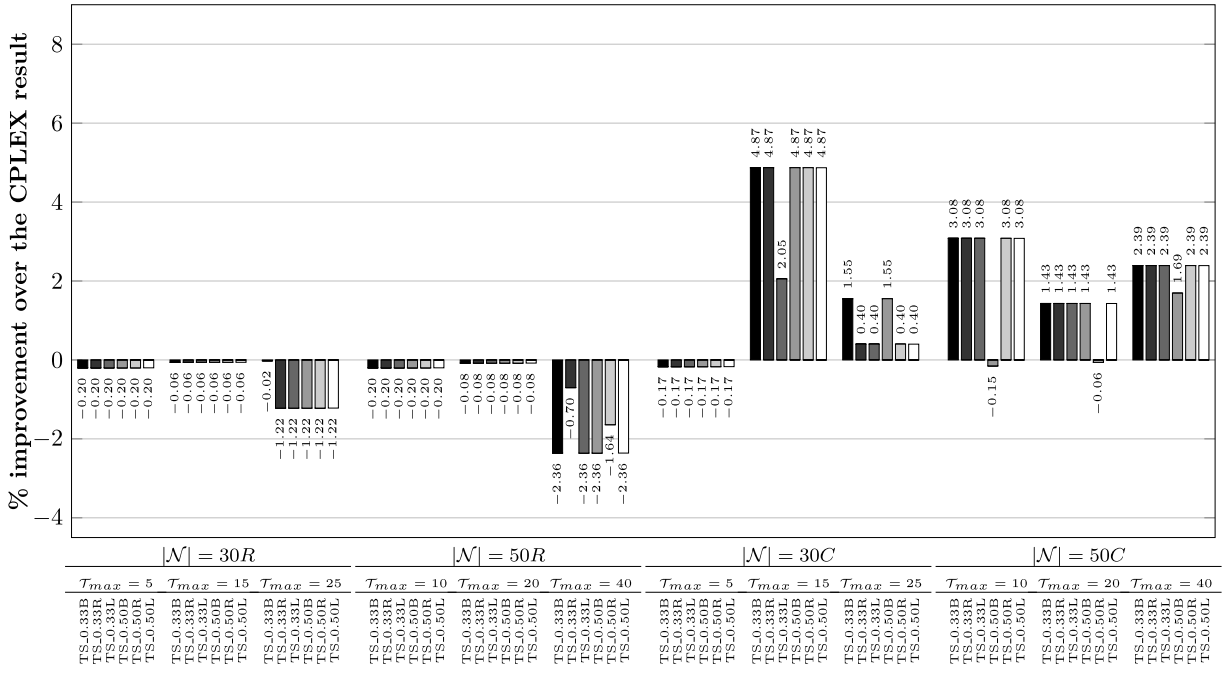

Fig. D.8. Average percent improvement of each algorithm setting over CPLEX for varying instance sizes and $\mathcal{T}_{max}$ values.

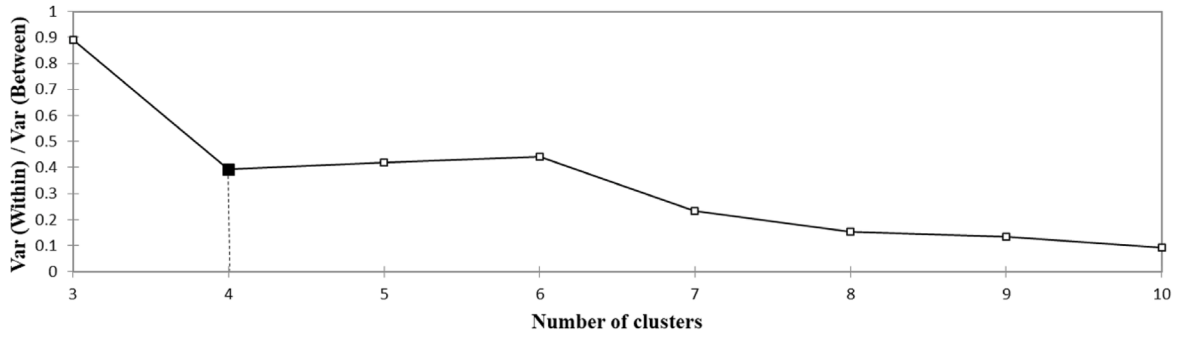

Fig. F.9. Within and between variance with respect to increasing number of clusters in the k-means algorithm.

Table C.8
Hypothetical instances with 30 sites: weights, known initial prevalence, unknown initial prevalence, coordinates and population of random and clustered networks.

				Random Network			Clustered Network		
N	w_i	RP	**0.5P**	X-coord	Y-coord	Pop'n	X-coord	Y-coord	Pop'n
0	-	-	–	35	35	–	40	50	–
1	1	0.45	0.5	41	49	717	25	85	929
2	1	0.5	0.5	35	17	1247	22	75	374
3	1	0.5	0.5	55	45	1217	22	85	266
4	1	0.49	0.5	55	20	428	20	80	564
5	1	0.48	0.5	15	30	785	20	85	1916
6	1	0.5	0.5	25	30	2315	18	75	2034
7	1	0.3	0.5	20	50	1136	15	75	2339
8	1	0.5	0.5	10	43	1406	15	80	1783
9	1	0.33	0.5	55	60	1124	10	35	1872
10	1	0.5	0.5	30	60	1487	10	40	490
11	1	0.5	0.5	20	65	1764	8	40	377
12	1	0.5	0.5	50	35	2095	8	45	1434
13	1	0.5	0.5	30	25	1899	5	35	500
14	1	0.5	0.5	15	10	1273	5	45	2392
15	1	0.4	0.5	30	5	2227	2	40	1175
16	1	0.5	0.5	10	20	361	63	23	1947
17	1	0.5	0.5	5	30	877	21	24	656
18	1	0.47	0.5	20	40	2067	12	24	2064
19	1	0.5	0.5	15	60	1932	24	58	512
20	1	0.48	0.5	45	65	1207	37	47	261
21	1	0.34	0.5	45	20	2428	49	42	966
22	1	0.36	0.5	45	10	2217	53	43	2427
23	1	0.44	0.5	55	5	848	61	52	2207
24	1	0.31	0.5	65	35	1755	57	48	677
25	1	0.43	0.5	65	20	385	56	37	441
26	1	0.5	0.5	45	30	486	55	54	1102
27	1	0.48	0.5	35	40	1330	4	18	886
28	1	0.5	0.5	41	37	972	26	52	2435
29	1	0.5	0.5	64	42	774	26	35	784
30	1	0.4	0.5	40	60	1021	31	67	608

Table C.9
Hypothetical instances with 50 sites: weights, known initial prevalence, unknown initial prevalence, coordinates and population of random and clustered networks.

				Random Network			Clustered Network		
N	w_i	RP	**0.5P**	X-coord	Y-coord	Pop'n	X-coord	Y-coord	Pop'n
0	-	-	–	35	35	–	40	50	–
1	1	0.5	0.5	41	49	252	25	85	825
2	1	0.5	0.5	35	17	1738	22	75	1642
3	1	0.5	0.5	55	45	2163	22	85	461
4	1	0.5	0.5	55	20	782	20	80	2161
5	1	0.5	0.5	15	30	569	20	85	1018
6	1	0.5	0.5	25	30	1062	18	75	1249
7	1	0.36	0.5	20	50	598	15	75	540
8	1	0.5	0.5	10	43	876	15	80	622
9	1	0.5	0.5	55	60	336	10	35	2253
10	1	0.33	0.5	30	60	562	10	40	1718
11	1	0.41	0.5	20	65	2432	8	40	1396
12	1	0.46	0.5	50	35	1281	8	45	1078
13	1	0.5	0.5	30	25	683	5	35	272
14	1	0.37	0.5	15	10	1868	5	45	960
15	1	0.5	0.5	30	5	452	2	40	704
16	1	0.42	0.5	10	20	890	0	40	2459
17	1	0.5	0.5	5	30	951	0	45	1092
18	1	0.5	0.5	20	40	2163	44	5	1154
19	1	0.5	0.5	15	60	531	42	10	823
20	1	0.5	0.5	45	65	595	42	15	498
21	1	0.5	0.5	45	20	1190	40	5	1426
22	1	0.34	0.5	45	10	1374	40	15	956
23	1	0.4	0.5	55	5	361	38	5	1843
24	1	0.5	0.5	65	35	1028	38	15	1130
25	1	0.32	0.5	65	20	1552	35	5	304
26	1	0.32	0.5	45	30	850	95	30	2146
27	1	0.5	0.5	35	40	1264	95	35	1310
28	1	0.46	0.5	41	37	1066	92	30	1365
29	1	0.5	0.5	64	42	1476	90	35	1719
30	1	0.37	0.5	40	60	1098	88	30	1146
31	1	0.5	0.5	31	52	2238	47	47	673
32	1	0.32	0.5	35	69	557	49	58	1274
33	1	0.34	0.5	53	52	1235	27	43	1512
34	1	0.5	0.5	65	55	1501	37	31	1647
35	1	0.5	0.5	63	65	1893	57	29	502
36	1	0.5	0.5	2	60	645	63	23	834
37	1	0.5	0.5	20	20	2038	21	24	1102
38	1	0.4	0.5	5	5	665	12	24	973
39	1	0.36	0.5	60	12	272	24	58	1306
40	1	0.4	0.5	40	25	2228	37	47	1484
41	1	0.4	0.5	42	7	2407	49	42	1277
42	1	0.5	0.5	24	12	387	53	43	427
43	1	0.5	0.5	23	3	2081	61	52	1267
44	1	0.5	0.5	11	14	461	57	48	2023
45	1	0.5	0.5	6	38	1354	56	37	1128
46	1	0.5	0.5	2	48	2298	55	54	1005
47	1	0.38	0.5	8	56	831	4	18	1198
48	1	0.32	0.5	13	52	1171	26	52	402
49	1	0.5	0.5	6	68	1039	26	35	1732
50	1	0.5	0.5	47	47	2012	31	67	1088

Table E.10
Case study instances: clusters, weights, coordinates and population.

N	Cluster	w_i	Latitude	Longitude	Pop'n	N	Cluster	w_i	Latitude	Longitude	Pop'n
0	–	–	38.495116	43.381376	–	47	4	1	38.833423	43.316832	737
1	1	1	38.508214	43.435471	21461	48	1	1	38.720826	43.402730	618
2	2	1	38.653578	43.651936	3769	49	1	1	38.750770	43.616188	1193
3	1	1	38.697100	43.433840	838	50	1	1	38.606341	43.615451	497
4	1	1	38.705138	43.211496	155	51	2	1	38.545312	43.338654	3101
5	1	1	38.694721	43.761717	1234	52	1	1	38.846861	43.494086	368
6	1	1	38.733197	43.495531	502	53	2	1	38.695860	43.645713	1009
7	3	1	38.835429	43.580160	156	54	3	1	38.587357	43.770504	1218
8	3	1	38.412542	43.376205	206	55	1	1	38.511102	43.455555	3049
9	3	1	38.620961	43.650470	779	56	2	1	38.687855	43.416485	517
10	1	1	38.571857	43.398385	900	57	1	1	38.513755	43.470275	1563
11	2	1	38.676196	43.243253	886	58	2	1	38.462595	43.361957	162
12	2	1	38.626929	43.229821	401	59	1	1	38.639467	43.700423	717
13	3	1	38.552495	43.773218	1119	60	3	1	38.866845	43.647902	112
14	3	1	38.563939	43.504043	341	61	3	1	38.476102	43.449025	2611
15	2	1	38.752745	43.570719	302	62	1	1	38.534873	43.462586	992
16	3	1	38.637255	43.390453	365	63	1	1	38.707080	43.506625	300
17	1	1	38.699980	43.259690	555	64	1	1	38.648115	43.440492	182
18	3	1	38.843585	43.526602	301	65	2	1	38.654382	43.515937	170
19	2	1	38.592523	43.583608	418	66	3	1	38.545413	43.537266	744
20	3	1	38.449858	43.462422	1243	67	4	1	38.842649	43.397956	553
21	2	1	38.570649	43.264832	5073	68	1	1	38.455034	43.409936	2029
22	3	1	38.528696	43.557866	527	69	3	1	38.769744	43.504114	144
23	2	1	38.906096	43.585889	482	70	2	1	38.678776	43.190081	147
24	1	1	38.645366	43.770289	1240	71	1	1	38.734670	43.415537	581
25	2	1	38.600740	43.218319	760	72	3	1	38.553032	43.700087	2112
26	1	1	38.588665	43.427575	312	73	4	1	38.693884	43.363366	193
27	2	1	38.909402	43.548074	250	74	4	1	38.672444	43.326459	397
28	3	1	38.618866	43.481866	222	75	2	1	38.737214	43.205101	139
29	2	1	38.763186	43.247137	376	76	4	1	38.764524	43.406096	754
30	1	1	38.636182	43.728318	430	77	3	1	38.897212	43.601854	241
31	3	1	38.517616	43.515716	1465	78	3	1	38.506065	43.493571	371
32	1	1	38.764825	43.537400	253	79	3	1	38.691975	43.481605	146
33	1	1	38.795972	43.494329	587	80	4	1	38.773023	43.344998	394
34	1	1	38.560918	43.468251	340	81	4	1	38.751239	43.363502	203
35	2	1	38.641361	43.239266	278	82	3	1	38.383517	43.472714	375
36	3	1	38.687687	43.840206	482	83	2	1	38.632964	43.254704	223
37	1	1	38.674186	43.462422	872	84	2	1	38.609024	43.232989	571
38	2	1	38.866210	43.508942	1778	85	3	1	38.571555	43.563917	414
39	4	1	38.803731	43.334513	347	86	3	1	38.695726	43.802140	1204
40	1	1	38.606005	43.487649	371	87	4	1	38.818913	43.366370	650
41	4	1	38.835496	43.438690	1201	88	3	1	38.742570	43.452272	476
42	4	1	38.721680	43.313928	328	89	4	1	38.654416	43.344483	413
43	1	1	38.588095	43.462930	386	90	2	1	38.784599	43.291426	672
44	3	1	38.535880	43.620393	1142	91	3	1	38.769343	43.577929	808
45	4	1	38.684773	43.346114	244	92	3	1	38.730786	43.724699	199
46	4	1	38.740561	43.315802	1408	93	4	1	38.682461	43.308098	432

Training and implementation strategy. Successful implementation of the proposed model requires targeted training for both field teams and assessment managers to ensure operational efficiency and consistency. Field teams should be trained in standardized data collection methods, including sampling techniques, interview protocols and data entry procedures. Additionally, training should cover strategies for managing safety concerns and responding to unexpected challenges in disaster-affected areas to ensure team preparedness and well-being. Assessment managers and coordinators should also receive higher-level training related to decision-making, resource allocation, and operational planning. This includes training on data integration and analysis, prioritization of assessment sites using historical and real-time data, and the implementation of the proposed approach within identified resource constraints. To improve operational feasibility, managers should also be trained to pilot the model in controlled environments before full-scale implementation.

6. Conclusion

This study addresses the critical problem of routing needs assessment teams and scheduling interviews in the aftermath of large-scale humanitarian events, such as disasters or conflicts. Practical experiences highlight the complexities of site selection and sampling strategies, where inadequate planning can lead to significant sampling errors and unreliable prevalence estimates. To address these challenges, we propose a novel optimization framework that integrates site coverage and sampling accuracy objectives to support time-sensitive needs assessment operations involving interviews with potential beneficiaries at selected sites.

The proposed optimization problem balances the trade-off between these objectives while leveraging efficient routing. Given the complexity of the problem, commercial solvers prove impractical for solving realistic-sized instances. To overcome this, we develop a heuristic solution approach combining an iterative constructive heuristic with a multi-start tabu search procedure, enabling the generation of high-quality solutions within reasonable computational times.

Our experiments, conducted on synthetic instances and a case study inspired by a historic earthquake in Van, Turkey, demonstrate the superiority of our approach over commercial solvers and comparable methods. The results also highlight the critical role of accurate initial prevalence estimates in improving assessment efficiency, as well as the influence of available assessment time and network topology on the resulting needs assessment routes.

While this study presents a novel optimization framework for humanitarian needs assessment problem, two primary limitations should be acknowledged. First, the model relies on the availability and reliability of secondary and real-time data, yet in many disaster contexts, data quality may be poor or incomplete. Addressing data uncertainty through robust optimization or probabilistic modeling could improve the model's reliability in data-scarce environments. Second, our approach employs an a priori clustering strategy, where large regions are divided into subregions before solving single-team routing problems. While this simplifies operations, following an integrated multi-team optimization approach can have efficiency benefits. It would be valuable to develop models and algorithms for multi-team planning, where clustering and routing decisions are optimized jointly considering practical needs and constraints.

These limitations open several promising avenues for future research. First, alternative exact and heuristic solution approaches could be explored to enhance the solution quality and computational efficiency for the proposed problem. Second, while this study adopts a practical approach by clustering sites a priori and solving a single-team problem, future work could extend the proposed model and solution approaches to a multi-team setting. In such a setting, clustering and routing decisions would be made in an integrated manner. This would involve developing new formulations and solution techniques that expand upon the team orienteering problem literature. Third, the current humanitarian logistics literature often treats needs assessment and disaster relief distribution as separate problems. However, demand uncertainty in relief distribution is strongly influenced by the accuracy of estimations made during the needs assessment phase. This study provides a foundation for linking these phases by characterizing demand uncertainty (e.g., through demand scenarios) based on quantified sampling accuracy, such as confidence levels and confidence interval half-widths. Another interesting future research direction is to convert a given ARP-CA instance to an equivalent Orienteering Problem or Prize Collecting TSP instance, which deems the solution approaches for these two problems to be applicable for the ARP-CA instances. Lastly, the proposed approach could be adapted to other critical settings where prevalence estimation is essential. For example, it could be applied to assess infection levels during pandemics or other public health crises. Extending the methodology to such domains would significantly broaden its applicability and societal impact.

CRediT authorship contribution statement

Duygu Pamukcu: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Melih Çelik:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Burcu Balçık:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Formal analysis, Conceptualization.

Appendix A. Pseudocode of the tabu search algorithm

Algorithm 3 The tabu search algorithm

Step 0: Initialization

$tIter_{move}(\mathcal{N}) = 0$ for all $move \in \{2-opt, swap, r1-1, r1-2\}$
$start = mainIter = iter1 = iter2 = 0$
$\mathcal{R}_{cur} = \mathcal{R}_{best} = \mathcal{N}^{CI} = traR_{cur} = traR_{best} = t_{route} = totR_{cur} = totR_{best} = totR_{next} = \emptyset$
$traR_{next} = totR_{next} = z_{best} = \infty$

Step 1: Construction of an initial solution

Execute **Constructive Heuristic**
$\mathcal{R}_{cur} = \mathcal{R}_{best} = \mathcal{N}^{CI}$, $traR_{cur} = traR_{best} = t_{route}$, $totR_{cur} = totR_{best} = t_{total}$

Step 2: Improvement Phase

while $start \leq maxStart$ **do**
  while $mainIter \leq maxMainIter$ **do**

Step 2a: Route improvement by neighborhood search

    $iter_1 = iter_2 = 0$
    while $iter_1 \leq maxIter_1$ **do**
      Execute **2-opt**$(\mathcal{R}_{cur}, traR_{cur}, totR_{cur}, tIter_{2-opt}(\mathcal{N}))$
      Execute **update**$(i, j, NULL, 2-opt, tTen_{2-opt}, tIter_{2-opt}(\mathcal{N}), \mathcal{R}_{cur}, \mathcal{R}_{next}, \mathcal{R}_{best}, traR_{cur}, traR_{next}, traR_{best}, totR_{cur}, totR_{next}, totR_{best}, inRoute(\mathcal{N}))$
      Execute **swap**$(\mathcal{R}_{cur}, traR_{cur}, totR_{cur}, tIter_{swap}(\mathcal{N}))$
      Execute **update**$(i, j, NULL, swap, tTen_{swap}, tIter_{swap}(\mathcal{N}), \mathcal{R}_{cur}, \mathcal{R}_{next}, \mathcal{R}_{best}, traR_{cur}, traR_{next}, traR_{best}, totR_{cur}, totR_{next}, totR_{best}, inRoute(\mathcal{N}))$
      $iter_1 \leftarrow iter_1 + 1$
    end while
    while $iter_2 \leq maxIter_2$ **do**
      Execute **replace1-1**$(\mathcal{R}_{next}, traR_{cur}, totR_{cur}, tIter_{r1-1}(\mathcal{N}))$
      Execute **update**$(i, j, k, r1-1, tTen_{r1-1}, tIter_{r1-1}(\mathcal{N}), \mathcal{R}_{cur}, \mathcal{R}_{next}, \mathcal{R}_{best}, traR_{cur}, traR_{next}, traR_{best}, totR_{cur}, totR_{next}, totR_{best}, inRoute(\mathcal{N}))$
      Execute **replace1-2**$(\mathcal{R}_{next}, traR_{next}, totR_{next}, tIter_{r1-2}(\mathcal{N}))$
      Execute **update**$(i, j, k, r1-2, tTen_{r1-2}, tIter_{r1-2}(\mathcal{N}), \mathcal{R}_{cur}, \mathcal{R}_{next}, \mathcal{R}_{best}, traR_{cur}, traR_{next}, traR_{best}, totR_{cur}, totR_{next}, totR_{best}, inRoute(\mathcal{N}))$
      $iter_2 \leftarrow iter_2 + 1$
    end while

Step 2b: Precision level assignment

    Solve the model defined by (21)–(24) to assign precision levels, with objective value z_{cur}
    if $z_{cur} < z_{best}$ **then**
      $z_{best} \leftarrow z_{cur}$
      $totR_{best} \leftarrow totR_{best} + z_{best}$
    end if
    $mainIter \leftarrow mainIter + 1$
  end while

Step 2c: Diversification

  Update $\mathcal{R}_{cur}$, $traR_{cur}$ and $totR_{cur}$ by removing the first $\theta_1\%$ of the nodes from the route
  or
  Update $\mathcal{R}_{cur}$, $traR_{cur}$ and $totR_{cur}$ by removing $\theta_2\%$ of the nodes randomly from the route
  or
  Update $\mathcal{R}_{cur}$, $traR_{cur}$ and $totR_{cur}$ by removing $\theta_3\%$ of the nodes i with largest $inRoute_i$
  $start \leftarrow start + 1$
end while

return $\mathcal{R}_{best}$, $traR_{best}$, $totR_{best}$

Appendix B. Pseudocodes of the algorithms in the improvement step

Algorithm 4 Function **2-opt**($\mathcal{R}_{cur}, traR_{cur}, totR_{cur}, tIter_{2-opt}(\mathcal{N})$)

```
for all i ∈ R_cur do
  for all j ∈ R_cur do
    if tIter_{2-opt}(i, suc_i) = tIter_{2-opt}(prec_j, j) = 0 then
      Exchange edges (i, suc_i) and (prec_j, j) on R_cur
      traR_next = traR_cur + t_{i,prec_j} + t_{j,suc_i} − t_{i,suc_i} − t_{prec_j,j}
      totR_next = totR_cur + t_{i,prec_j} + t_{j,suc_i} − t_{i,suc_i} − t_{prec_j,j}
    end if
  end for
end for
```

Algorithm 5 Function **swap**($\mathcal{R}_{cur}, traR_{cur}, totR_{cur}, tIter_{swap}(\mathcal{N})$)

```
for all i ∈ R_cur do
  for all j ∉ R_cur do
    if tIter_swap(i, j) = tIter_swap(j, i) = 0 then
      Remove node i from R_cur and replace node j in the same location
      traR_next = traR_cur + t_{prec_i,i} + t_{i,suc_i} − t_{prec_i,j} − t_{j,suc_i}
      totR_next = totR_cur + β(n_{j,|L|} − n_{i,|L|}) + t_{prec_i,i} + t_{i,suc_i} − t_{prec_i,j} − t_{j,suc_i}
    end if
  end for
end for
```

Algorithm 6 Function **replace1-1**($\mathcal{R}_{next}, traR_{cur}, totR_{cur}, tIter_{r1-1}(\mathcal{N})$)

```
for all i ∈ R_next do
  for all j ∉ R_next do
    if tIter_{r1-1,remove}(i) = tIter_{r1-1,insert}(j) = 0 then
      traR_next ← traR_cur − t_{prec_i,i} − t_{i,suc_i} + t_{prec_i,suc_i}
      totR_next ← traR_cur − β n_{i,|L|} − t_{prec_i,i} − t_{i,suc_i} + t_{prec_i,suc_i}
      R_next ← R_next \ {i}
      Execute cheapestInsertion(R_next, {j}, traR_next, totR_next)
    end if
  end for
end for
```

Algorithm 7 Function **replace1-2**($\mathcal{R}_{next}, traR_{next}, totR_{next}, tIter_{r1-2}(\mathcal{N})$)

```
for all i ∈ R_next do
  for all j ∉ R_next do
    for all k ∉ R_next, k ≠ j do
      if tIter_{r1-1,remove}(i) = tIter_{r1-1,insert}(j) = tIter_{r1-1,insert}(k) = 0 then
        traR_next ← traR_next − t_{prec_i,i} − t_{i,suc_i} + t_{prec_i,suc_i}
        totR_next ← traR_next − β n_{i,|L|} − t_{prec_i,i} − t_{i,suc_i} + t_{prec_i,suc_i}
        R_next ← R_next \ {i}
        Execute cheapestInsertion(R_next, {j, k}, traR_next, totR_next)
      end if
    end for
  end for
end for
```

Appendix C. Instances used for experiments on randomly generated networks

See Fig. C.7 and Tables C.8 and C.9.

Appendix D. Results on preliminary experiments comparing diversification strategies

In our tests, we run our instances under the aforementioned settings and by using each of the strategies. For each strategy, we try θ values of 33% and 50%. Fig. D.8 provides a summary of the average percent improvement over the best CPLEX solutions under each setting, where each column is the average of four instances and negative values indicate that CPLEX finds a better result than the TS approach with the given settings. The diversification strategies are denoted by the θ values (0.33 or 0.50), followed by the node removal strategy (B, R, and L for the cases where nodes are removed from the beginning of the route, randomly, and based on number of occurrences, respectively).

Algorithm 8 Function **update**($i, j, k, move, tTen_{move}, tIter_{move}(\mathcal{N})$, $\mathcal{R}_{cur}, \mathcal{R}_{next}, \mathcal{R}_{best}, traR_{cur}, traR_{next}, traR_{best}, totR_{cur}, totR_{next}, totR_{best}$, $inRoute(\mathcal{N})$)

```
if totR_next ≤ T_max then
  R_cur ← R_next, totR_cur ← totR_next, traR_cur ← traR_next
  if traR_cur ≤ traR_best then
    R_best ← R_cur, totR_best ← totR_cur, traR_best ← traR_cur
    for all i ∈ R_best do
      inRoute_i ← inRoute_i + 1
    end for
    if move ∈ {2 − opt, swap, r1 − 1} then
      tIter_move(i) = tIter_move(j) = tTen_move
      for all n ∈ N, n ≠ i, n ≠ j, tIter_move(n) > 0 do
        tIter_move(n) ← tIter_move(n) − 1
      end for
    else
      tIter_{r1-2}(i) = tIter_{r1-2}(j) = tIter_{r1-2}(k) = tTen_{r1-2}
      for all n ∈ N, n ≠ i, n ≠ j, n ≠ k, tIter_{r1-2}(n) > 0 do
        tIter_{r1-2}(n) ← tIter_{r1-2}(n) − 1
      end for
    end if
  end if
end if
```

Appendix E. Data used to generate case study instances

See Table E.10

Appendix F. Within/between cluster variance ratios for varying number of clusters for the case instances

See Fig. F.9.

Data availability

Data will be made available on request.

References

[1] ACAPS. The assessment capacities - project technical brief: Purposive sampling and site selection in phase 2. 2012, http://www.coordinationtoolkit.org/wp-content/uploads/ACAPS-Purposive-Sampling-and-Site-Selection.pdf. [Accessed July 2024].

[2] ACAPS. Qualitative and quantitative research techniques for humanitarian needs assessment. 2012, https://reliefweb.int/sites/reliefweb.int/files/resources/qualitative_and_quantitative_research_techniques.pdf. [Accessed July 2024].

[3] Balcik B. Site selection and vehicle routing for post-disaster rapid needs assessment. Transp Res Part E: Logist Transp Rev 2017;101:30–58.

[4] Perry M, Slentz M, Stephen R, Liu L, Tran D, Mukkavilli S, et al. Hurricane ike rapid needs assessment - houston, Texas, september 2008. CDC MMWR: Morb Mortal Wkly Rep 2009;58(38):1066–71.

[5] Choudhary E, Chen T-H, Martin C, Vagi S, Roth J, Keim M, et al. Public health needs assessments of tutuila Island, American samoa, after the 2009 tsunami. Disaster Med Public Heal Prep 2012;6(3):209–16.

[6] WHO. Needs assessments. 2000, https://apps.who.int/iris/bitstream/handle/10665/66584/WHO_MSD_MSB_00.2d.pdf. [Accessed July 2024].

[7] Benini A. A computer simulation of needs assessments in disasters. 2012, https://www.acaps.org/sites/acaps/files/resources/files/a_computer_simulation_of_needs_assessments_in_disasters-the_impact_of_sample_size_logistical_difficulty_and_measurement_error_november_2012.pdf. [Accessed July 2024].

[8] Johnson M, Wilfert R. Rapid needs assessments and gis. FOCUS Field Epidemiol 2008;5(3).

[9] IFRC. International federation of red cross and red crescent societies - guidelines for assessment in emergencies. 2008, https://www.icrc.org/en/doc/assets/files/publications/icrc-002-118009.pdf. [Accessed July 2024].

[10] UNICEF. Rapid assessment sampling in emergency situations. 2010, https://www.bond.org.uk/sites/default/files/resource-documents/rapid_assessment_sampling_booklet.pdf. [Accessed July 2024].

[11] Darcy J, Hofmann CA. According to need? Needs assessment and decision-making in the humanitarian sector. 2003, https://reliefweb.int/report/world/according-need-needs-assessment-and-decision-making-humanitarian-sector. [Accessed July 2024].
[12] Huang M, Smilowitz KR, Balcik B. A continuous approximation approach for assessment routing in disaster relief. Transp Res Part B: Methodol 2013;50:20–41.
[13] Oruc BE, Kara BY. Post-disaster assessment routing problem. Transp Res Part B: Methodol 2018;116:76–102.
[14] Archetti C, Speranza MG, Vigo D. Vehicle routing problems with profits. In: Vehicle routing: Problems, methods, and applications. second ed.. SIAM; 2014, p. 273–97.
[15] Balas E. The prize collecting traveling salesman problem. Networks 1989;19(6):621–36.
[16] Association NE. COVID-19 testing in schools: Why it matters and how to advocate for it. 2022, https://www.nea.org/covid-19-testing-in-schools. [Accessed July 2024].
[17] UNDP. Post-disaster needs assessments. 2013, https://www.undp.org/sites/g/files/zskgke326/files/publications/PDNA%20Volume%20A%20FINAL%2012th%20Review_March%202015.pdf.
[18] WHO. Measuring change in nutritional status: Guidelines for assessing the nutritional impact of supplementary feeding programmes for vulnerable groups. 1982, http://apps.who.int/iris/bitstream/handle/10665/38768/9241541660.pdf. [Accessed July 2024].
[19] CDC. Community assessment for public health emergency response (CASPER) toolkit. 2019, https://www.cdc.gov/nceh/casper/docs/CASPER-toolkit-3_508.pdf. [Accessed June 2022].
[20] Deitchler M, Deconinck H, Bergeron G. Precision, time, and cost: A comparison of three sampling designs in an emergency setting. Emerg Themes Epidemiol 2008;5(6).
[21] Waring S, desVignes Kendrick M, Arafat R, Reynolds K, Souza G, Bishop S, et al. Tropical storm allison rapid needs assessment - Houston, Texas, June 2001. CDC Morb Mortal Wkly Rep 2002;287(20):2646–7.
[22] Korteweg HA, van Bokhoven I, Yzermans C, Grievink L. Rapid health and needs assessments after disasters: A systematic review. BMC Public Health 2010;10(1):295–306.
[23] Bayleyegn T, Wolkin A, Oberst K, Young S, Sanchez C, Phelps A, et al. Rapid assessment of the needs and health status in santa rosa and escambia counties, florida, after hurricane ivan, september 2004. Disaster Manag Response 2006;4(1):12–8.
[24] Hong K, Song K, Shin S, Song S, Ro Y, Jeong J, et al. Rapid health needs assessment after typhoons bolaven and tembin using the public health assessment for emergency response toolkit in paju and jeju, Korea 2012. J Korean Med Sci 2017;32(4):1367–73.
[25] Waring S, Zakos-Feliberti A, Wood R, Stone M, Padgett P, Arafat R. The utility of geographic information systems (GIS) in rapid epidemiological assessments following weather-related disasters: Methodological issues based on the Tropical Storm Allison Experience. Int J Hyg Environ Heal 2005;208(1–2):109–16.
[26] Maya A. Rapid assessment in disasters. JMAJ 2013;56:19–24.
[27] Brynildsrud O. COVID-19 prevalence estimation by random sampling in population - optimal sample pooling under varying assumptions about true prevalence. BMC Med Res Methodol 2020;20:196.
[28] Sahlu I, Whittaker AB. Obtaining prevalence estimates of coronavirus disease 2019: A model to inform decision-making. Am J Epidemiol 2021;190(8):1681–8.
[29] Al Qotba HA, Al Nuaimi AS, Al Mujalli H, Zainel AA, Khudadad H, Marji T, et al. COVID-19 surveillance in the primary health care population of Qatar: Experience of prioritizing timeliness over representativeness when sampling the population. Front Public Heal 2021;9:1–7.
[30] Asgary A, Cojocaru MG, Najafabadi NM, Wu J. Simulating preventative testing of SARS-CoV-2 in schools: Policy implications. BMC Public Heal 2021;21:125.
[31] Balcik B, Yanikoglu I. A robust optimization approach for humanitarian needs assessment planning under travel time uncertainty. European J Oper Res 2020;282(1):40–57.
[32] Pamukcu D, Balcik B. A multi-cover routing problem for planning rapid needs assessment under different information-sharing settings. OR Spectr 2020;42:1–42.
[33] Li X, Liu X, Ma H, Hu S. Integrated routing optimization for post-disaster rapid-detailed need assessment. Int J Gen Syst 2020;49(5):521–45.
[34] Hakimifar M, Balcik B, Fikar C, Hemmelmayr V, Wakolbinger T. Evaluation of field visit planning heuristics during rapid needs assessment in an uncertain post-disaster environment. Ann Oper Res 2021;1–42.
[35] Coutinho WP, Fliege J, Battarra M. Glider routing and trajectory optimisation in disaster assessment. European J Oper Res 2019;274(3):1138–54.
[36] Zhang G, Zhu N, Ma S, Xia J. Humanitarian relief network assessment using collaborative truck-and-drone system. Transp Res Part E: Logist Transp Rev 2021;152:102417.
[37] Tsiligirides T. Heuristic methods applied to orienteering. J Oper Res Soc 1984;35(9):797–809.
[38] Laporte G, Martello S. The selective travelling salesman problem. Discrete Appl Math 1990;26(2–3):193–207.
[39] Golden BL, Levy L, Vohra R. The orienteering problem. Naval Res Logist 1987;34(3):307–18.
[40] Chao I-M, Golden BL, Wasil EA. A fast and effective heuristic for the orienteering problem. European J Oper Res 1996;88(3):475–89.
[41] Gendreau M, Laporte G, Semet F. A tabu search heuristic for the undirected selective travelling salesman problem. European J Oper Res 1998;106(2–3):539–45.
[42] Verbeeck C, Vansteenwegen P, Aghezzaf E. Solving the stochastic time-dependent orienteering problem with time windows. European J Oper Res 2016;255(3):699–718.
[43] Feillet D, Dejax P, Gendreau M. Traveling salesman problems with profits. Transp Sci 2005;39(2):188–205.
[44] Boussier S, Feillet D, Gendreau M. An exact algorithm for team orienteering problems. 4OR 2007;5(3):211–30.
[45] Keshtkaran M, Ziarati K, Bettinelli A, Vigo D. Enhanced exact solution methods for the team orienteering problem. Int J Prod Res 2016;54(2):591–601.
[46] Archetti C, Hertz A, Speranza MG. Metaheuristics for the team orienteering problem. J Heuristics 2007;13:49–76.
[47] Tang H, Miller-Hooks E. A tabu search heuristic for the team orienteering problem. Comput Oper Res 2005;32(6):1379–407.
[48] Lin S-W, Vincent FY. A simulated annealing heuristic for the team orienteering problem with time windows. European J Oper Res 2012;217(1):94–107.
[49] Erdoğan G, Laporte G. The orienteering problem with variable profits. Networks 2013;61(2):104–16.
[50] Thompson SK. Sampling. Hoboken, New Jersey: John Wiley & Sons; 2012.
[51] Miller CE, Tucker AW, Zemlin RA. Integer programming formulation of traveling salesman problems. J ACM 1960;7(4):326–9.
[52] Archetti C, Carrabs F, Cerulli R. The set orienteering problem. European J Oper Res 2018;267:264–72.
[53] Rosenkrantz DJ, Stearns RE, Lewis PM. An analysis of several heuristics for the traveling salesman problem. SIAM J Comput 1977;6(3):563–81.
[54] Solomon MM. Algorithms for the vehicle routing and scheduling problems with time window constraints. Oper Res 1987;35(2):254–65.
[55] Noyan N, Balcik B, Atakan S. A stochastic optimization model for designing last mile relief networks. Transp Sci 2016;50(3):1092–113.
[56] Jain AK. Data clustering: 50 years beyond K-means. Pattern Recognit Lett 2010;31(8):651–66.

Dr Duygu Pamukcu is An Assistant Professor at Providence College in Providence, United States. Prior to this position, she was a postdoctoral research fellow in the Department of Logistics and Operations Management at HEC Montréal in Canada. Dr Pamukcu holds a Ph.D. in Business Information Technology from Virginia Tech in the United States. Her research interests focuses on advancing the operations of organizations in humanitarian and crisis settings using exact and heuristic approaches, along with data-driven methods such as econometric analysis and statistics

Prof Melih Çelik is an Associate Professor in the School of Management at the University of Bath, United Kingdom. Dr Çelik holds a Ph.D. in Industrial Engineering and an MS in Operations Research from Georgia Institute of Technology in Atlanta, United States. His research focuses on applying operational research tools and methods to address decisions in healthcare and humanitarian operations, warehouse logistics, and facility management.

Prof Burcu Balçık is a Professor of Industrial Engineering at Ozyegin University in Istanbul, Türkiye. Prof Balçık received her Ph.D. in Industrial Engineering from University of Washington in United States, and her M.S. and B.S. degrees in Industrial Engineering from Middle East Technical University in Ankara, Türkiye. She also worked as a postdoctoral fellow at Northwestern University. Her main research interests are in the areas of humanitarian logistics, disaster management, supply chain management, mathematical modeling, and heuristic methods.