Diversity-aware Unmanned Vehicle Team Arrangement in Mobile Crowdsourcing

Yu Li
Hangzhou Dianzi University https://orcid.org/0000-0003-0092-2462

Haonan Feng
Hangzhou Dianzi University School of Computer Science and Technology

Zhankui Peng
Hangzhou Dianzi University School of Computer Science and Technology

Li Zhou (1489164382@qq.com)
Zhejiang University of Science and Technology

Jian Wan
Zhejiang University of Science and Technology

Research

Keywords: Unmanned vehicle, Spatial crowdsourcing, Group task assignment, Team Diversity

Posted Date: January 24th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1272310/v1

RESEARCH

Diversity-aware Unmanned Vehicle Team Arrangement in Mobile Crowdsourcing

Yu Li[1], Haonan Feng[1], Zhankui Peng[1], Li Zhou[1*] and Jian Wan[1,2]

*Correspondence: zhouli@hdu.edu.cn
[1]School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, China
Full list of author information is available at the end of the article

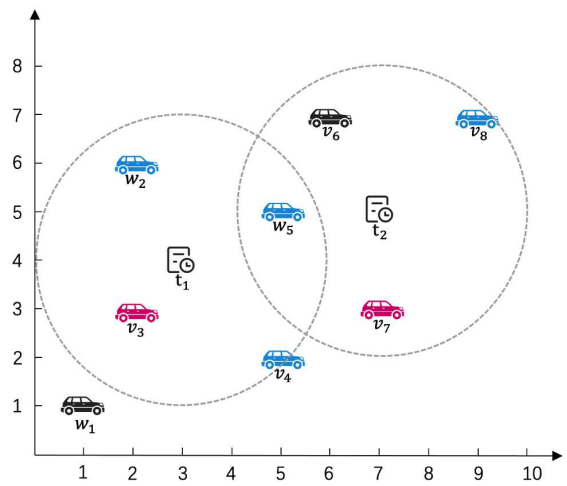
Abstract

With the continuous development of mobile edge computing and the improvement of unmanned vehicle technology, unmanned vehicle could handle ever-increasing demands. As a significant application of unmanned vehicle, spatial crowdsourcing will provide an important application scenario, which is about to organize a lot of unmanned vehicle to conduct the spatial tasks by physically moving to its locations, called task assignment. Previous works usually focus on assigning a spatial task to one single vehicle or a group of vehicles. Few of them consider that vehicle team diversity is essential to collaborative work. Collaborative work is benefits from organizing teams with various backgrounds vehicles. In this paper, we consider a spatial crowdsourcing scenario. Each vehicle has a set of skills and a property. The property denotes vehicle's special attribute, (e.g., size, speed or weight). We introduce a concept of entropy to measure vehicle team diversity. Each spatial task (e.g., delivering the take-out, and carrying freight) is under the time and budget constraint, and required a set of skills. We need to assure that the assigned vehicle team is diverse. To address this issue, we first propose a practical problem, called team diversity spatial crowdsourcing (TD-SC) problem which finds an optimal team-and-task assignment strategy. Moreover, We design a framework which includes a greedy with diversity (GD) algorithm and a divide-and-conquer (D&C) algorithm to get team-and-task assignments. Finally, we demonstrate efficiency and effectiveness of the proposed methods through extensive experiments.

Keywords: Unmanned vehicle; Spatial crowdsourcing; Group task assignment; Team Diversity

1 Introduction

With the rapid development of Long Term Evolution (LTE) network and the Fifth-Generation (5G) cellular network, unmanned vehicles have been widely used in real life. Such as unmanned vehicle could conduct spatial crowdsourcing task instead of people. Unmanned vehicle Spatial crowdsourcing is about to organize a number of unmanned vehicles to conduct the spatial tasks by physically moving to its locations, called task assignment. [1, 2]. Some studies on spatial crowdsourcing usually concentrates on the problems of task assignment [3, 4, 5, 6, 7], which are to allocate tasks to unmanned vehicles, and they presume tasks are all simple and easy. Some studies consider the complicated spatial tasks, which usually need to be accomplished by a team of unmanned vehicles with different skills. However, in real scenarios, a simple team may not complete the spatial task well. we also need to concern about team diversity.



Task	Skills
t_1	$\{s_1, s_2, s_3\}$
t_2	$\{s_2, s_4, s_5\}$

Vehicle	Skills	property
v_1	$\{s_2\}$	1
v_2	$\{s_1, s_3\}$	2
v_3	$\{s_3\}$	3
v_4	$\{s_4, s_5\}$	3
v_5	$\{s_1, s_2\}$	3
v_6	$\{s_3, s_4\}$	1
v_7	$\{s_5\}$	2
v_8	$\{s_2\}$	3

Figure 1 Information of unmanned vehicles/tasks

Collaborative work is benefits from organizing teams with various backgrounds unmanned vehicles [8, 9, 10, 11]. For instance, studies have suggested that diversity in a firm's knowledge background and its creativity are positively correlated [12]. Teams which have diverse unmanned vehicles are usually regarded to be competitive, because teams like that are easier get new thoughts [13].

Example 1. To intuitively understand the significance of the team diversity, we could consider the effect of non-diverse teams. A user requests a repairing house task which needs many different goods, such as cement, steels, pipes and woods. Obviously, it is a complex task which needs a team of unmanned vehicle to conduct it. When team members of delivering goods only are same size, it may couldn't done this task well. Therefore, we could form a diverse team which contains different size unmanned vehicles. It can be seen that diversity is often necessary to give the efficiency.

In this paper, we will consider an essential problem in the spatial crowdsourcing, namely Team Diversity Spatial Crowdsourcing (TD-SC), which aims to effectively assign diverse teams of unmanned vehicles to complicated spatial tasks, under the task constraints of timestamp, task's radius, budgets and team diversity index, so that the required skill sets of tasks are completely covered by those unmanned vehicles, and the total value of the assignment (defined as the total traveling cost of unmanned vehicles) is minimized.

Example 2. In the next, we will illustrate the TD-SC problem by a motivation example. Figure 1 shows the information of unmanned vehicles/tasks. There exist eight unmanned vehicles (v_1, ..., v_8) and two tasks (t_1, t_2). Each task is required to be assigned to one or more unmanned vehicles and labeled with its required skills, current location and valid radius. Each unmanned vehicle is associated with it's skills, current location and cluster which denotes it's special attribute (e.g., gender, race, country of residence or age range). The problem is to assign tasks to a team of unmanned vehicles so as to minimize the average task cost and optimal assigned

team diversity. For task t_1, its available unmanned vehicles are v_2, v_3, v_4, v_5 respectively. We can chose unmanned vehicle v_2 and unmanned vehicle v_5 to from a team to complete this task. As can be seen, the skills union of unmanned vehicle v_2 and unmanned vehicle v_5 is $< s_1, s_2, s_3 >$ which can exactly cover task t_1 required skills set $< s_1, s_2, s_3 >$. Moreover, unmanned vehicle v_2 and unmanned vehicle v_5 cluster is 2, 3 respectively. So We can make sure that the team is diverse.

The contributions made by our paper can be summarized as follows:

- We formally define the team diversity spatial crowdsourcing (TD-SC) problem in Section 2, under the constraints of skills covering, timestamp, task's radius, budget and team diversity index for spatial tasks in the spatial crowdsourcing.
- We propose two effective approaches, namely greedy with diversity (GD), divide-and-conquer with diversity (D&C) algorithms to tackle the TD-SC problem in Sections 4 and 5, respectively.
- As demonstrated by the experiments, our designed algorithms can effectively form diverse teams of unmanned vehicles for spatial tasks that can accomplish an optimal task assignment.

Section 3 introduces a general framework for our TD-SC problem in spatial crowdsourcing. Section 8 reviews previous works on spatial crowdsourcing. Finally, Section 9 concludes this paper. Fig1, 2, 3 are the comparison between our paper and other papers.

2 Related Works

2.1 Spatial Crowdsourcing

Table 1 Comparison of task between our paper and other papers

Paper ID	Task Features					
	Location	Start Time	Multi skills	Budget	Radius	Diversity
Our paper	✓	✓	✓	✓	✓	✓
Topk	✓	✗	✓	✗	✓	✗
Task Assign	✓	✓	✓	✓	✗	✗
Forming Diverse	✗	✗	✗	✓	✗	✓

Crowdsourcing has been widely studied in [14]. Previous works [14, 14, 15] often studied crowdsourcing problems. In these problems, workers do not need to move to the task's location. On the contrary, the spatial crowdsourcing problems [15] needs workers to move to some specific locations of tasks, and conduct the requested task. Spatial crowdsourcing problem can be classified into two kinds namely Server Assigned Tasks (SAT) and Worker Selected Tasks (WST) based on the task publishing modes. Specifically, the SAT mode is that spatial crowdsourcing sever directly assigning spatial tasks to available workers. SAT mode need sever to collect all the information of tasks and workers, such that it can maximize the number of assigned tasks [15]. The WST mode is that spatial tasks are published on server and all workers can get those information. so that workers can select spatial task according to their personal preferences and behavior habit [16]. In our work, we study our TD-SC problem based on the SAT model, where workers are paid for conducting tasks.

Our TD-SC problem targets at assigning workers to tasks by using our proposed algorithms, so that the needed skills of tasks can be contented.

Most of the former works in spatial crowdsourcing concentrate on assigning spatial tasks to the single worker. Nevertheless, some papers also focus on collaborative task assignment in spatial crowdsourcing [17, 18], which is assigning tasks to a team of workers. Cheng et al. [17] propose a algorithm called Cooperation-Aware Spatial Crowdsourcing to tackle team task assignment problem so that the spatial tasks can be completed with high cooperation quality scores. Our proposed TD-SC problem not only focus on the team collaboration but also team diversity. Collaborative work is benefits from organizing teams with various back-grounds workers.

Table 2 Comparison of worker between our paper and other papers

Paper ID	Worker Features			
	Location	Multi skills	Multi teams	Cluster
Our paper	✓	✓	✓	✓
Topk	✓	✗	✓	✗
Task Assign	✓	✓	✓	✓
Forming Diverse	✗	✗	✗	✓

2.2 Team formation problem

Team formation problem is also closely related. The problem of team formation is about finding the minimum cost team of unmanned vehicles according to their skills. Anagnostopoulos et al. [19, 20] studies the workload balance problem in the team formation problem. Majumder et al. [21] consider the the capacity constraint of experts issue in team formation problem.

Moreover, the problem of forming diverse team in crowdsourcing is also studied. Sara et al. [22] studies the diverse team formation problem. Faez et al. [23] present a method to form such diverse teams from people arriving sequentially over time in a firm. However, none of those papers consider the team diversity problem in spatial crowdsourcing. [7, 14] In our paper, we introduce a concept of entropy which can measure the team diversity in spatial crowdsourcing.

3 Methods

3.1 Problem Statement

In this section, we first introduce the essential concepts, and present the formal definition of the team diversity spatial crowdsourcing (TD-SC) problem, in which we assign diverse team to complicated spatial tasks.Table 1 summarizes the key notations used in the rest of our paper.

Unmanned vehicles in Different Clusters

We first define the multi-skilled unmanned vehicles in spatial crowdsourcing platform, in which we assign those unmanned vehicles to complicated spatial tasks. We assume that $S = \{s_1, s_2, ..., s_l\}$ is a set of L skills. Each unmanned vehicle has one or more skills in S, and can carry out some spatial tasks which require some skills in S. We also assume that $P = \{p_1, p_2, ..., p_k\}$ is a set of K properties. Each unmanned vehicle has one property. A property denotes unmanned vehicle's special attribute,

Table 3 SUMMARY OF NOTATIONS

Notations	Description
p	Timestamp
V_p	A set of unmanned vehicles at timestamp p
v_i	A unmanned vehicle $v_i \in V_p$
l_i	Location of the unmanned vehicle v_i
o_i	Online time of the unmanned vehicle v_i
S_i	A skill set of the unmanned vehicle v_i
$c_{i,j}$	Cost of the unmanned vehicle v_i to complete the task t_j
p_i	Property of the unmanned vehicle v_i
T_p	A set of tasks at timestamp p
t_j	A spatial the task t_j
l_j	Location of the task t_j
o_j	Online time of the task t_j
S_j	A skill set of the task t_j
b_j	Budget of the task t_j
r_j	Radius of the task t_j
d_j	Diversity of the task t_j

(e.g., size, speed or weight) which can be measure team diversity. **(Unmanned vehicles)** Let $V_p = \{v_1, v_2, ..., v_m\}$ be a set of n unmanned vehicles at timestamp p. Each unmanned vehicle is denoted by $v_i =< l_i, o_i, S_i, c_{i,j}, p_i > (1 \leq i \leq m)$, where l_i is the location of the unmanned vehicle in a 2D space, o_i is the online time of the unmanned vehicle, $S_i \subseteq S$ is the set of skills that the unmanned vehicle is good at, c_i is the cost for the unmanned vehicle to complete a task, p_i is the property of the unmanned vehicle.

3.1.1 Complex Spatial Tasks

Now, we define complicated spatial tasks in the spatial crowdsourcing platform, which are constrained by radius, budgets and diversity index. **(Tasks)** Let $T_p = \{t_1, t_2, ..., t_n\}$ be a set of complex spatial tasks at a timestamp p. Each task is denoted by $t_j =< l_j, o_j, S_j, b_j, r_j, d_j >$, where l_j is the location of the task in a 2D space, o_j is the online time of the task, $S_j \subseteq S$ is the set of skills that the task need, b_j is the budget of the task, r_j is the radius of the task, only those unmanned vehicles which are located in the circular range with the radius r_j around l_j can conduct the task t_j, d_j is the diversity of the team which contains those unmanned vehicles which can complete the task t_j.

3.1.2 Team Diversity Spatial Crowdsourcing Problem

In this subsection, we will formally define the team diversity spatial crowdsourcing (TD-SC) problem, whicß assigns a team of diverse unmanned vehicles to spatial tasks so that all unmanned vehicles can cover the skills required by tasks and the assignment can obtain high team diversity and low traveling cost.

Before we present the TD-SC problem, we first introduce the concept of task assignment instance. **(Task Assignment Instance)** At timestamp p, given a unmanned vehicle set V_p and a task T_p, a *task assignment instance set* is a form $< v_1, v_2, ..., v_n, t_j >$, where each unmanned vehicle $v_i \subseteq V_p$ is assigned to one spatial task $t_j \subseteq T_p$.

Intuitively, the task assignment instance,$< v_1, v_2, ..., v_n, t_j >$, is one valid assignment. Each assignment instance must satisfies all constraints of task t_j, with regard to location radius (i.e., r_j), online time (i.e., o_j), budget (i.e., b_j), skills (i.e., S_j) and diversity (i.e., d_j). The *task assignment instance* means that all unmanned vehicles' skills can cover the spatial task' skills S_j and no redundant unmanned vehicles.

(TD-SC Problem) Given a time interval P, a spatial crowdsourced task t_j, a set of unmanned vehicle V_p, the problem of team diversity spatial crowdsourcing problem is to assign the available team with diverse unmanned vehicles to spatial task $t_j \subseteq T_p$, at each timestamp $p \subseteq P$, such that the following constrains are satisfied:

1) Radius constraint: each unmanned vehicle $v_i \subseteq V_p$ must within the radius of the task t_j.
2) Timestamp constraint: each unmanned vehicle $v_i \subseteq V_p$ which is assigned to t_j must share the same timestamp p with spatial task t_j.
3) Budget constraint: the sum of all unmanned vehicles, $< v_1, v_2, ..., v_n >$ in task assignment instance, cost must lower than spatial task t_j's budget.
4) Skill constraint: all unmanned vehicles, $< v_1, v_2, ..., v_n >$ in task assignment instance, must cover the spatial task's skill S_j.
4) Diversity constraint: all unmanned vehicles, $< v_1, v_2, ..., v_n >$ in task assignment instance, diversity must more than specific value, such that the team, $< v_1, v_2, ..., v_n >$ in task assignment instance, is diversity.

3.2 Framework of Our Approach

In this section, we present a general framework, namely **TD-SC_Framework**, in Algorithm 1 for solving the TD-SC problem, which assigns a team of diverse unmanned vehicle to spatial tasks for many rounds. S denotes the strategy of unmanned vehicles and task assignment in all time interval P. For each round, at timestamp p, we first retrieve a set, V_p, of all the available unmanned vehicles, and a set, T_p, of all the available spatial tasks (lines 3-4). The set V_p includes those unmanned vehicles which newly arrive at the system, and have completed the previously assigned tasks. Thus they are available to conduct new tasks in the current round. Moreover, the available task set T_p contains existing spatial tasks that have not been assigned to unmanned vehicles, and the ones that newly arrive at the system.

After we obtain the available spatial tasks T_p and unmanned vehicles V_p, we can apply our approach obtain a good unmanned vehicles and task assignment strategy S_p, which is a subset of S (line 6).

Finally, for each unmanned vehicles and task assignment strategy S_p, we will notify unmanned vehicle v_i to conduct task t_j (lines 7-9).

Algorithm 1: TD-SC Framework

Input: a time interval P, $V = \{v_1, v_2, ..., v_{|V|}\}$ and our approach(,...,)
Output: unmanned vehicle and task assignment strategy S with in the interval P

1 $S \leftarrow \phi$
2 **for** *each time interval P in all Time* **do**
3 retrieve all the available spatial tasks to T_p ;
4 retrieve all the available unmanned vehicles to V_p ;
5 use our approach to obtain a good assignment instance set S_p ;
6 $S \leftarrow$S$\cup$S$_p$
7 **for** *each S_p in S* **do**
8 **for** *each unmanned vehicle v_i in S_p* **do**
9 Inform unmanned vehicle v_i to conduct task t_j ;

3.3 TD-SC Greedy with Diversity Approach

In this section, we will propose a greedy algorithm. Algorithm 2 shows the pseudo code of our task assignment with team diversity algorithm. Initially, we set I_p to be empty, since no tasks are assigned to any unmanned vehicles (line 1). Next, for each task t_j in T_p, we find out all valid unmanned vehicles $t_jVehicles$ in the crowdsourcing system at timestamp p (line 3). Here, the valid unmanned vehicles $t_jVehicles$ satisfy 5 conditions: (1)each unmanned vehicle $v_i \subseteq V_p$ must locate in restricted radius of the task t_j; (2)each unmanned vehicle $v_i \subseteq V_p$ which is assigned to t_j must share the same timestamp p with spatial task t_j; and (3)unmanned vehicle v_i have skills that task t_j requires.

Then, for each unmanned vehicle $v_i \in$t$_jVehicles$, we would select one best unmanned vehicle with the highest score increase, and add it to $Team_j$ (lines 5-6). If $Team_j$ could complete the task t_j, we continue to compute the diversity of $Team_j$, denoted by D_j (line 8). If the diversity of $Team_j$ more than lower bound of diversity, we remove all unmanned vehicle in $t_jVehicles$ from the available unmanned vehicles V_p, and the valid pairs $< t_j, Team_j >$ into unmanned vehicles and task assignment strategy S_p (lines 9-11). If the diversity of $Team_j$ lower than lower bound of diversity, we replace each unmanned vehicle in $Team_j$ to form a new team, until the new team's diversity could satisfies the lower bound of diversity (lines 13-14).If $Team_j$ couldn't complete the task t_j, we forgo this task (lines 15-16).

Specifically, we use monotone function obtain $team_j$ diversity (line 8). In subsection 2.1, We assume that $P = \{p_1, p_2, ..., p_k\}$ is a set of K properties. Each unmanned vehicle has one property. Now, we assume people belong to K clusters $p_k \subseteq P$ is a partition of all people P. We also define $r_{i,j}$ as the quality of unmanned vehicle v_i to do task t_j. In our context, for a specific task t_j, we define an objective function $f_j : E \rightarrow \mathbb{R}$ which rewards diversity as follows:

$$f\left(Teamj\right) = \sum_{k=1}^{K} \sqrt{\sum_{v_i \in Team_j} r_{i,j}} \tag{1}$$

Table 4 Running process of greedy with diversity algorithm

Round	v_2	v_3	v_4	v_5
1	$\sqrt{2}*$	$\frac{1}{\sqrt{2}}$	0	$\frac{2}{\sqrt{5}}$
2	-	0	0	$\frac{1}{\sqrt{5}}*$

Example 3. Back to our running example in Example 2. The running process of the greedy algorithm as follows. For task t_1, we retrieve its valid unmanned vehicles $\{v_2, v_3, v_4, v_5\}$. Every unmanned vehicle's value is shown in Table 5. So that we chose v_2 with the biggest value in the first round. Since v_2 cannot complete this task by himself, we continue to choose v_5 with the biggest value. According to function 1, task t_1 diversity value is 0.69. So that it is a good team which meet the team diversity requirements. For task t_2, we retrieve its valid unmanned vehicles $\{v_6, v_7, v_8\}$. Similarly, we can form a team with three unmanned vehicles $\{v_6, v_7, v_8\}$ to conduct this task. The team diversity value is 1.09 which also meet the team diversity requirements.

Algorithm 2: The Greedy Algorithm

Input: The available spatial tasks T_p, The available unmanned vehicles V_p

Output: Task assignment instance set S_p

1 $S_p \leftarrow \phi$
2 **for** *each task* $t_j \in T_p$ **do**
3 $\quad t_jVehicles \leftarrow$ get valid unmannedvehicles of task t_j ;
4 $\quad$ **for** *each unmanned vehicles* $v_i \in t_jVehicles$ **do**
5 $\quad\quad v_{best} \leftarrow \text{argmax}_{v_i}\left(\frac{MAXITEM(Team_j \cup v) - MAXITEM(v)}{c}\right)$;
6 $\quad\quad Team_j \leftarrow \text{Team}_j \cup \text{v}_{best}$;
7 $\quad\quad$ **if** $Team_j$ *could complete task* t_j **then**
8 $\quad\quad\quad D_j \leftarrow$ *use monotone function obtain* $Team_j$ *diversity* ;
9 $\quad\quad\quad$ **if** $D_j > lowerBound\ of\ Diversity$ **then**
10 $\quad\quad\quad\quad V_p \leftarrow V_p - Team_j$;
11 $\quad\quad\quad\quad S_p \leftarrow S_p \cup < t_j, Team_j >$;
12 $\quad\quad\quad$ **else**
13 $\quad\quad\quad\quad$ retrieve new unmanned vehicle from $t_jVehicles$ to form new $Team_j$;
14 $\quad\quad\quad\quad$ go to line7;
15 $\quad\quad$ **else**
16 $\quad\quad\quad$ forgo this task;

3.4 TD-SC Divide and Conquer Approach

The Greedy algorithm could solve this problem. But it may incur that we can only accomplish local optimality. Consequently, in this section, we present a divide-and-conquer algorithm (D&C), which first divides the TD-SC problem into smaller sub-problems, such that each sub-problems includes a subset of spatial task, and then conquers the sub-problems recursively until the final set size is 1.

Specially, for each sub-problem, we will process this problem by recursion. we should note that we can solve the problem by the greedy algorithm 2 when the sub-problem includes only one task. During the recursive process, we will merge assignment results from sub-problems, and obtain the assignment results by reconciling the conflictive assignment instance. Finally, we can return the task assignment instance set I_p.

3.4.1 TD-SC Problem Decompositions

In this subsection, we decompose a TD-SC problem into sub-problems. Given n unmanned vehicle and m spatial tasks, we part those spatial tasks into k sub-groups, each sub-group contains $\lceil m/g \rceil$ tasks. Algorithm 3 presents the pseudo code of our decomposition algorithm, namely **TD-SC_Decomposition**, which return TD-SC sub-problems, P_s, after decomposing the original TD-SC problem.

Specially, we could set g with different number according to the spatial task's number. in our case, we let g be 5 (line 1). Then we initialize empty sub-problems, P_s (lines 2-3). Next, we obtain one sub-problem P_s at a time (lines 4-8). In particular, for each round, we retrieve a task t_j and its top-$\lceil m/g \rceil$ nearest tasks (line 5). Then, for each task t_j, we obtain its valid unmanned vehicles which can meet the task t_j requirement (such as: timestamp constraint, radius constraint, skill constraint.) (lines 6-7). Finally, we return all the decomposed sub-problems P_s.

Algorithm 3: TD-SC_decomposition

Input: m available spatial tasks in T_p
Output: decomposed TD-SC sub-problems, $P_s(1 \leq s \leq g)$

1 estimate the best number of groups, g, for V_p
2 **for** $s = 1$ *to* g **do**
3 $\quad P_s \leftarrow \phi$
4 **for** $s = 1$ *to* g **do**
5 $\quad$ let set $T_p^{(j)}$ contain the next task t_j and its top-($\lceil m/g \rceil$ - 1) nearest tasks
6 $\quad$ **for** *each task* $t_j \in T_p^j$ **do**
7 $\quad\quad$ obtain all valid unmanned vehicle for task t_j
8 $\quad$ add $T_p^{(j)}$ to P_s
9 return $P_s(1 \leq s \leq g)$

3.4.2 TD-SC Problem Divide-and-Conquer

In this subsection, we propose a divide-and-conquer algorithm, namely TD-SC_D&C, which recursively parts the initial TD-SC problem into sub-problems, recursively solves every sub-problem, and merges assignment results of sub-problems by resolving the conflicts.

Specifically, in Algorithm 4 (TD-SC_D&C), we first initialize empty task assignment instance set S_p (line 1). Then we call the TD-SC_Decomposition approach (as mentioned in Algorithm 3) to get sub-problems P_s (line 2). For each sub-problem P_s, if P_s includes more than 1 task, we will recursively call Algorithm 4 (TD-SC_D&C) itself to divide the sub-problem P_s (lines 4-5). Or, when sub-problem P_s includes only one task, we can apply algorithm 2 (The Greedy Algorithm) to solve this sub-problem T_p^s, and get assignment results $I_p^{(s)}$ (lines 6-7). Afterwards, we can get an assignment instance set I_p for each sub-problem P_s, and merge them into one

task assignment instance set I_p, by reconciling the conflict unmanned vehicles (lines 8-10). Specifically, I_p is merged with an assignment set I_p^s from sub-problem P_s at each time (lines 9-10). we call algorithm 5 (TD-SC_Conflict_Reconcile) to solve the conflict. Eventually, we return the final result of merged task assignment instance set I_p (line 11).

Algorithm 4: TD-SC_D&C

Input: The available spatial tasks T_p, The available unmanned vehicle V_p

Output: Task assignment instance set I_p

1 $I_p \leftarrow \phi$
2 invoke TD-SC_Decomposition(V_p, T_p), and obtain subproblems T_p^s
3 **for** *s = 1 to g* **do**
4 **if** *the number of tasks in subproblem T_p^s is more than 1* **then**
5 $I_p^{(s)}$ = TD-SC_DC(V_p, T_p)
6 **else**
7 invoke algorithm 2 (The Greedy Algorithm) to solve sub-problem T_p^s, and obtain assignment results $I_p^{(s)}$
8 **for** *i = 1 to g* **do**
9 get the next subproblem T_p^s;
10 I_p = TD-SC_Conflict_Reconcile(I_p, $I_p^{(s)}$)
11 return I_p

3.4.3 TD-SC Problem Conflict Reconciliation

In this subsection, we propose the conflict reconciliation algorithm, which solves the conflicts while merging assignment results of sub-problems. We presume that I_p is the merged assignment instance set. There is a new sub-problem P_s with assignment set $I_p^{(s)}$. Algorithm 5 (TD-SC_Conflict_Reconcile) presents the merging algorithm, namely **TD-SC_Conflict_Reconcile**, which incorporates two assignment sets I_p and $I_p^{(s)}$ by resolving conflicts.

Specifically, the same unmanned vehicle v_i could be assigned to two different tasks from two sub-problems. But a unmanned vehicle can only be assigned to one task at the same time. So we must to avoid that. Our algorithm need to find a set, V_c, of all conflicting unmanned vehicles between I_p and $I_p^{(s)}$ (line 1). After that, we get conflicting unmanned vehicle v_i and its k corresponding spatial tasks. For each task, we calculate the value that the unmanned vehicle v_i can bring to this task. we assign the unmanned vehicle v_i to the task with highest value (lines 4-9). Specifically, the value one unmanned vehicle can bring to a task is that the number of unmanned vehicle v_i can be used to task t_j divide the distance cost between unmanned vehicle v_i and task t_j (line 6). Then we let conflicting unmanned vehicle v_i assign to the task which has higher value, and substitute the conflicting unmanned vehicle v_i with another available unmanned vehicle for task with lower value. It is worth noting that if no other unmanned vehicles are available for replacing v_i , we may need to sacrifice the task with lower value. After resolving all conflicts, we merge assignment instance set I_p with $I_p^{(s)}$ (line 12), and return the merged result I_p.

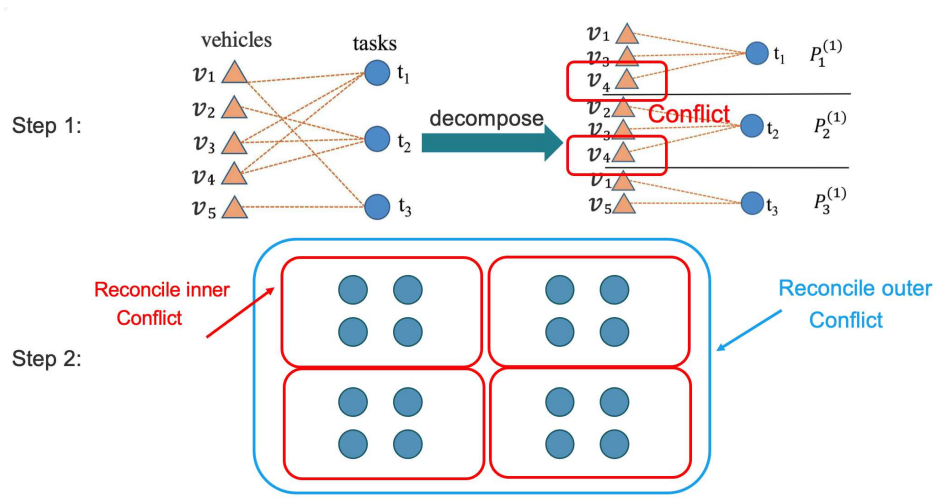


Figure 2 Conflict reconcile

Algorithm 5: TD-SC_Conflict_Reconcile

Input: the current assignment instance set, I_p, of subproblem P have been merged, and the assignment instance set, $I_p^{(s)}$, of subproblem P_s

Output: a merged task assignment instance set, I_p

$V_c \leftarrow find\ all\ conflicting\ unmannedvehicles\ between$ I_p and $I_p^{(s)}$
while $V_c \neq \phi$ **do**
  get a unmanned vehicle $v_i \in V_c$ *with the highest traveling cost in Ip(s), and its k corresponding spatial tasks*
  $bestValue_{t_j} = 0$
  for $i = 0$ *to* k **do**
    $value_{t_j} \leftarrow argmax_{v_i} \left(\frac{MAXITEM(Team_j \cup w)}{c} \right)$
    if $value_{t_j} < bestValue_{t_j}$ **then**
      $bestValue_{t_j} = value_{t_j}$
  let the conflicting unmanned vehicle v_i *stay in the best task's team;*
  find another unmanned vehicle v_j *for the task which lost the the conflicting unmanned vehicle* v_i*;*
  $V_c = V_c - \{v_i\}$
$I_p = I_p \cup I_p^{(s)}$
return I_p

Example 4. Back to our running example in Example 2. The running process of the divide and conquer algorithm as follows. First we invoke TD-SC_Decomposition(V_p, T_p), and obtain sub-problems T_p^s. In this case, we get two sub-problems T_p^s. T_p^1 contains task t_1. T_p^2 contains task t_2. For each sub-problem, we invoke algorithm 2 (The Greedy Algorithm) to solve sub-problem T_p^s, and obtain assignment results $I_p^{(s)}$. For sub-problems T_p^1, we retrieve task t_1 valid unmanned vehicles $\{v_2, v_3, v_4, v_5\}$. Similarly, we choose unmanned vehicle v_2 and unmanned vehicle v_5 to conduct this task, and its team diversity is 0.69. For sub-problems

Table 5 Running process of divide and counquer algorithm

Round	v_5	v_6	v_7	v_8
1	$\frac{1}{2}$ *	$\frac{1}{\sqrt{5}}$	$\frac{1}{2}$	$\frac{1}{\sqrt{8}}$
2	-	$\frac{1}{\sqrt{5}}$ *	$\frac{1}{3}$	
3	-	-	$\frac{1}{3}$*	0

T_p^2, we retrieve task t_2 valid unmanned vehicles $\{v_5, v_6, v_7, v_8\}$. Every unmanned vehicle's value is shown in Table 6. Similarly, we can form a team with three unmanned vehicles $\{v_5, v_6, v_7\}$ to conduct this task. However, unmanned vehicle v_5 is assigned to two different tasks which is not acceptable. So we reconcile this conflict by algorithm 5. We calculate the value of unmanned vehicle v_5 which can bring to those tasks respectively. Task t_1 value is $\frac{2}{\sqrt{5}}$, and t_2 value is $\frac{1}{2}$. So that we let v_5 stay in task t_1 team. Then we will find another unmanned vehicle for task t_2. As can be seen, we find that v_8 can perfectly substitute v_5. Finally, task t_1 team is $\{v_2, v_5\}$ which team diversity is 0.69. task t_2 team is $\{v_6, v_7, v_8\}$ which team diversity is 1.09.

4 Results and discussion

4.1 Performance Metrics

We measure the performance of team diversity with two elements——how much team diversity it increases to the team and how much efficiency it loses. In order to measure advancement on team diversity, we measure the Shannon entropy of a team of diverse unmanned vehicles, with and without using our approach. It quantifies the uncertainty in predicting the cluster label of an individual that is taken at random from the dataset. Shannon entropy of a team is given by: $-\sum_{i=1}^{K}(p_k \log p_k)$, where p_k is the proportion of people on that team from cluster k. Therefore, the impact of team diversity can be measured as advancement in average entropy for all teams. We define the *entropy gain* (EG) as:

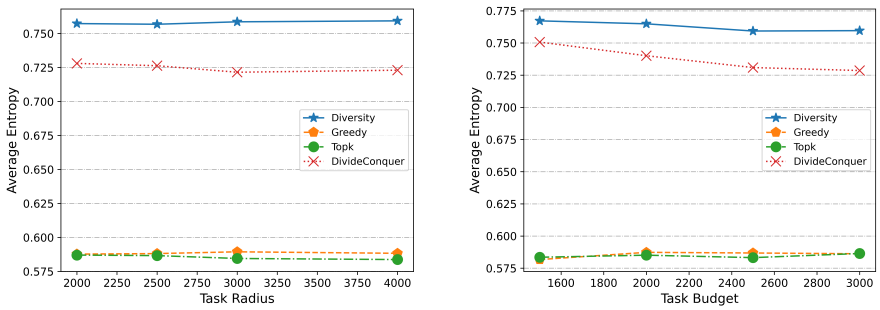
$$\text{EG} = \frac{\text{Average entropy based on team diversity}}{\text{Average entropy without team diversity}} \tag{2}$$

If a team of unmanned vehicles are evenly come from different clusters, entropy for a team is maximized. In the same way, if a team of unmanned vehicle are all come from same cluster, entropy for a team is minimized.

To measure the loss of efficiency owing to diverse matching, we use the *cost of diversity* (CoD) metric which measures the trade-off in completed task's number under a team diverse matching objective. In particularly, we define the metric to measure the completed task's number loss.

$$\text{CoD} = 1 - \frac{\text{Number of completed tasks based on diversity algorithm}}{\text{Number of completed tasks based on greedy algorithm}} \tag{3}$$

For instance, at timestamp p, we first retrieve a set, V_p, of all the available unmanned vehicles, and a set, T_p, of all the available spatial tasks. If we use diversity



(a) Average Entropy With Different Radius (b) Average Entropy With Different Budget

Figure 3 Average Entropy

approach to match 80 tasks, we use greedy approach to match 100 tasks. Then CoD will be 0.2, suggesting that encouraging diversity requires lost twenty percent of task.

4.2 Experimental Study

In this section, we conduct experiments to demonstrate the effectiveness and efficiency of our proposed algorithms. All the experiments were run on a MacOS with Intel Core i5 @ 3.1 GHz and 16 GB memory, and all the algorithms were implemented in Java with JDK 11.

4.2.1 Experimental Set up

Data Sets. We use a real data set collected from DiDi, which is a Chinese vehicle for hire company. We collected the taxi-calling data sampled from October 2016 in XiAn by a large-scale online taxicalling platform in China. In the DiDi dataset, every order has a ID, a matched driver ID, a start timestamp, a end timestamp, a start location, a end location.

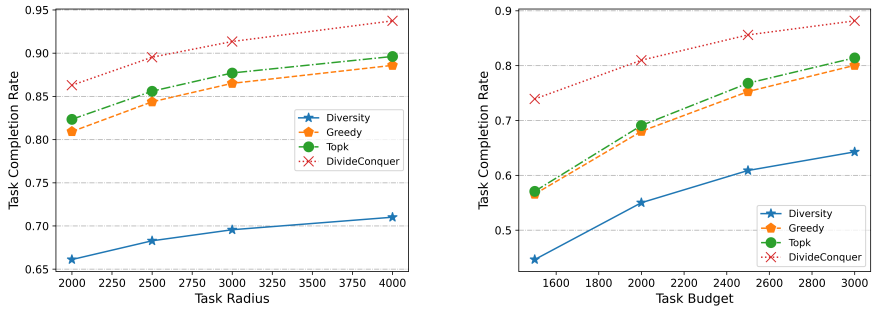
We use synthetic data based on the DiDi's real data set to test our approaches. Specifically, each unmanned vehicle has a ID, a start location, a online timestamp, one or two skills and property. Each task has a ID, a start location, a online timestamp, one or more needed skills and budget.In this paper, our synthetic dataset includes the information of 10,536 unmanned vehicles and 3,512 spatial tasks.

Evaluation. We compare and evaluate the performance of following methods:

1. Baseline: a baseline approach without considering diversity approach.
2. Top-k: Top-k Team Recommendation and Its Variants in Spatial Crowdsourcing
3. GD: Greedy Diversity algorithm
4. DC: Divide and Conquer algorithm

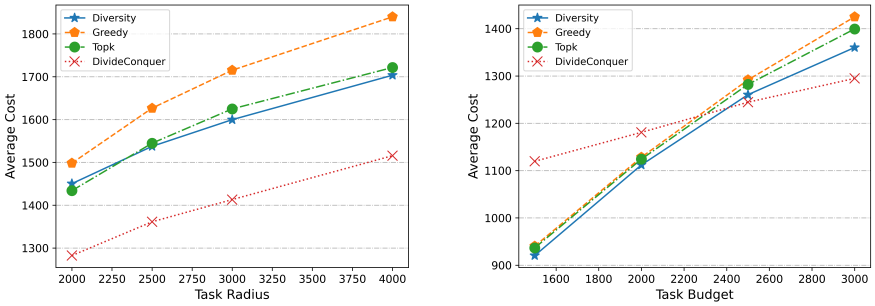
4.2.2 Experimental Results

In this subsection, we show the effects of the range of task budgets, the range of task radius.



(a) Task Completion Rate With Different Radius (b) Task Completion Rate With Different Budget

Figure 4 Average Entropy



(a) Average Cost With Different Radius (b) Average Cost With Different Budget

Figure 5 Average Entropy

Effect of the Range of Task Radius. Figure illustrates the experimental results on different task's radius of task B_j from 2000 to 4000. In figure (a), when the value of task's radius increase, the task's team average entropy of baseline and Top-k increase slightly, and the task's team average entropy of GD and DC decrease. However, our proposed two algorithm can obtain a high value. It suggests that our algorithms can make the team diverse in task assignment. In figure (b), when the value of task's radius increase, the task completion rate of four approaches all increase. The reason is that, the more task radius the more available unmanned vehicles for tasks. In figure (c), when the value of task's radius increase, the task completion rate of four approaches all increase. The reason is that, task cost is positively correlated with the task radius. Although, our GD algorithm average cost is highest, our DC algorithm can decrease the task cost largely.

Effect of the Task Budgets. Figure illustrates the experimental results on different values of task B_j from 1500 to 3000. In figure (a), the task's team average entropy of baseline and Top-k increase, when the value of task budgets gets larger. In contrast, the task's team average entropy of GD and DC decrease, when the value of task budgets gets larger. GD and DC can achieve higher value than baseline and Top-k. It shows that our proposed two approaches are more better. As shown in

Figure (b), the task completion rate of all the four algorithms increase, when the value of task budgets gets larger. GD and DC can achieve task completion rate than baseline and Top-k. It shows that our proposed two approaches are more effective. As shown in Figure (c), the task's average cost of all the four algorithms increase, when the value of task budgets gets larger. Task's average cost of GD is highest in those four approaches. Task's average cost of DC is generally lowest. It shows that our proposed DC approaches are effective.

5 Conclusions

In this paper, we study a novel spatial crowdsourcing problem, called the team diversity in spatial crowdsourcing, which assigns a team of diverse unmanned vehicles to the multi-skill-required spatial tasks, so that the needed skills of spatial tasks can be contented by a team of unmanned vehicles. To address the TD-SC problem, we design a general framework, which not only includes a greedy algorithm but also divide-and-conquer algorithm, which can effectively retrieve MS-SC answers. Finally, we conduct a lot of experiments which verify the efficiency and effectiveness of the proposed algorithms.

Acknowledgements
Not applicable.

Funding
This work was supported in part by the National Natural Science Foundation of China under Grant 61972358, and in part by the Natural Science Foundation of Zhejiang Province under Grant LY21F020018.

Abbreviations
TD-SC: team diversity spatial crowdsourcing; GD:greedy algorithm with diversity; D&C: divide-and-conquer algorithm; LTE: Long Term Evolution network; 5G: the 5th Generation cellular network; SAT: Server Assigned Tasks; WST: Worker Selected Tasks; EG: entropy gain for all teams; CoD: cost of diversity for a team; Top-k: Top-k Team Recommendation and Its Variants in Spatial Crowdsourcing.

Availability of data and materials
The datasets used and analyzed during the current study are available from the corresponding author on reasonable request.

Ethics approval and consent to participate
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
YL and HF designed the problem and proposed the experimental algorithms. HF and ZP simulated the method and tested it, besides, they drafted the manuscript. YL and LZ revised and improved the manuscript. JW provided technical support. All authors read and approved the final manuscript.

Author details
[1]School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, China. [2]School of Information and Electronic Engineering, Zhejiang University of Science and Technology, Hangzhou, China.

References

1. Kazemi, L., Shahabi, C.: Geocrowd: enabling query answering with spatial crowdsourcing. In: Proceedings of the 20th International Conference on Advances in Geographic Information Systems, pp. 189–198 (2012)
2. Tong, Y., She, J., Ding, B., Wang, L., Chen, L.: Online mobile micro-task allocation in spatial crowdsourcing. In: 2016 IEEE 32Nd International Conference on Data Engineering (ICDE), pp. 49–60 (2016)
3. Kazemi, L., Shahabi, C., Chen, L.: Geotrucrowd: trustworthy query answering with spatial crowdsourcing. In: Proceedings of the 21st Acm Sigspatial International Conference on Advances in Geographic Information Systems, pp. 314–323 (2013)
4. To, H., Ghinita, G., Shahabi, C.: A framework for protecting worker location privacy in spatial crowdsourcing. Proceedings of the VLDB Endowment **7**(10), 919–930 (2014)
5. To, H., Shahabi, C., Kazemi, L.: A server-assigned spatial crowdsourcing framework. ACM Transactions on Spatial Algorithms and Systems (TSAS) **1**(1), 1–28 (2015)
6. She, J., Tong, Y., Chen, L., Cao, C.C.: Conflict-aware event-participant arrangement and its variant for online setting. IEEE Transactions on Knowledge and Data Engineering **28**(9), 2281–2295 (2016)

7. Tong, Y., She, J., Ding, B., Chen, L., Wo, T., Xu, K.: Online minimum matching in real-time spatial data: experiments and analysis. Proceedings of the VLDB Endowment **9**(12), 1053–1064 (2016)
8. Stirling, A.: A general framework for analysing diversity in science, technology and society. Journal of the Royal Society Interface **4**(15), 707–719 (2007)
9. Clutterbuck, D.: Coaching the Team at Work, (2011)
10. Lenhardt, V.: Coaching for Meaning: The Culture and Practice of Coaching and Team Building, (2004)
11. Levi, D., Askay, D.A.: Group Dynamics for Teams, (2020)
12. Hewlett, S.A., Marshall, M., Sherbin, L.: How diversity can drive innovation. Harvard business review **91**(12), 30–30 (2013)
13. Paulus, P.B., van der Zee, K.I., Kenworthy, J.: Cultural diversity and team creativity. In: The Palgrave Handbook of Creativity and Culture Research, pp. 57–76 (2016)
14. Alt, F., Shirazi, A.S., Schmidt, A., Kramer, U., Nawaz, Z.: Location-based crowdsourcing: extending crowdsourcing to the real world. In: Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries, pp. 13–22 (2010)
15. Tong, Y., Chen, L., Zhou, Z., Jagadish, H.V., Shou, L., Lv, W.: Slade: A smart large-scale task decomposer in crowdsourcing. IEEE Transactions on Knowledge and Data Engineering **30**(8), 1588–1601 (2018)
16. Deng, D., Shahabi, C., Demiryurek, U.: Maximizing the number of worker's self-selected tasks in spatial crowdsourcing. In: Proceedings of the 21st Acm Sigspatial International Conference on Advances in Geographic Information Systems, pp. 324–333 (2013)
17. Cheng, P., Chen, L., Ye, J.: Cooperation-aware task assignment in spatial crowdsourcing. In: 2019 IEEE 35th International Conference on Data Engineering (ICDE), pp. 1442–1453 (2019)
18. Gao, D., Tong, Y., Ji, Y., Xu, K.: Team-oriented task planning in spatial crowdsourcing. In: Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint Conference on Web and Big Data, pp. 41–56 (2017)
19. Anagnostopoulos, A., Becchetti, L., Castillo, C., Gionis, A., Leonardi, S.: Power in unity: forming teams in large-scale community systems. In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, pp. 599–608 (2010)
20. Anagnostopoulos, A., Becchetti, L., Castillo, C., Gionis, A., Leonardi, S.: Online team formation in social networks. In: Proceedings of the 21st International Conference on World Wide Web, pp. 839–848 (2012)
21. Majumder, A., Datta, S., Naidu, K.: Capacitated team formation problem on social networks. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1005–1013 (2012)
22. Cohen, S., Yashinski, M.: Crowdsourcing with diverse groups of users. In: Proceedings of the 20th International Workshop on the Web and Databases, pp. 7–12 (2017)
23. Ahmed, F., Dickerson, J., Fuge, M.: Forming diverse teams from sequentially arriving people. Journal of Mechanical Design **142**(11), 111401 (2020)