Wireless Personal Communications (2022) 123:2863–2894
https://doi.org/10.1007/s11277-021-09267-5

SigSense: Mobile Crowdsensing Based Incentive Aware Geospatial Signal Monitoring for Base Station Installation Recommendation Using Mixed Reality Game

Aakashjit Bhattacharya[1] · Debashis De[1]

Accepted: 19 October 2021 / Published online: 30 October 2021

Abstract
SigSense, a mobile crowdsensing-based geospatial video game, has been proposed to survey live signal strength using smartphones.It provides attractive incentives to the contributers. Live data collected as a survey through this game is used to recommend locations for installing the base stations to improve the signal quality using the Greedy Algorithm. A large plot of land is considered a large matrix. We have recursively divided the land into smaller submatrices. Then signal strength survey of each submatrix is performed through Mobile Crowd Sensing using SigSense. The recommendation system advices the locations for installing the network base stations for improving signal strength. An incentive is provided to a player based on the game's rules, making it a win–win situation for both the player and the network service provider. The unique feature of this game is that it can be played even in an area where is low mobile network coverage. A player's details are hidden from other players through unique masked ids and mixed reality.

Keywords Mobile crowdsensing · Incentive mechanism · Greedy Algorithm · Signal strength · Recommendation system

1 Introduction

In the twenty-first century, smartphones and other smart devices have become affordable and an integral part of everyone's life. As per Gartner's prediction, by 2022, 80% of smartphones will have on-device AI capabilities [1]. This has created many opportunities for mobile crowdsensing with low infrastructural investment. These smartphones use advanced technologies and come fitted with different sensors like GPS, proximity sensor, motion sensor, orientation sensor, grip sensor, gyroscope sensor, fingerprint sensor, accelerometer, and several other sensors along with powerful processors.

✉ Debashis De
dr.debashis.de@gmail.com

Aakashjit Bhattacharya
aakashjit4mtech@gmail.com

[1] Centre of Mobile Cloud Computing, Department of Computer Science and Engineering, Maulana Abul Kalam Azad University of Technology, Haringhata, Nadia, West Bengal 741249, India

Combining and coordinating these sensors and microprocessors to work together makes these smartphones an essential part of Mobile Crowd Sensing (MCS) to collect live data since MCS is generally location-dependent [2]. Over time, many new applications of MCS are evolving since MCS takes advantage of the ubiquity and sensing power of these devices [3]. With time cheaper, powerful, and multi-sensor-enabled handheld intelligent devices are replacing the traditional computers and have become an integral part of our day-to-day life [4]. Crowd Sourcing of sensor data from mobile devices and later analyzing those collected data for extraction of required information to measure, analyze or estimate any process of similar interest is called Crowd Sensing. In mobile crowdsourcing, the entire crowdsourcing framework consists of mainly three types of roles: the first type of role is played by a group of people who require real-world crowdsourced data. The second type of role is played by crowdsources who contribute towards collecting the crowdsourced data, and the third role is played by the platform that is used for crowdsourcing.

Signal strength data presents opportunities for researchers to enable Smart City planning and applications for improving cities' functionality, efficiency, competitiveness, and sustainability to encourage the involvement of all its citizens in the city's governance and innovation processes. Geospatial Signal strength data helps to measure, understand, plan the infrastructure of the Mobile network efficiently. It is used for a more accurate prediction of future demands, better planning, and decision-making. Data quality issues are challenges with high volume real-time data streaming, scalability of data analysis, and learning techniques to introduce exciting challenges.

The evolution of 5G technology is the biggest revolution in mobile internet connectivity as it gives ten times lower latency than 4G and high reliability and scalability. 5G NB-IoT provides ultra-low latency, low energy consumption, and reliable wireless communication support for IoT devices. MCS requires many participants with smartphones to sense their surroundings via the built-in sensor of mobile devices [5]. There already exist many other popular sensing systems like Signal Guru [6], SmartTrace [7], LiFS [8], VTrack [9], Sensorly (www.sensorly.com), and many more [11]. The more the number of participants in MCS, the better the success rate of the crowdsensing. Many participants voluntarily participate in games, but many want incentives in return for their contributions, including battery life, mobile bandwidth, and the two most important things: privacy and time. Participants must have smartphones or tablets. Crowdsensing enables end-users with a smartphone to visit real-world physical locations to collect live information with the help of these intelligent handheld devices in exchange for rewards or incentives [10]. Whenever a participant uses their smartphones for these systems, multiple resources of their smartphones are used [11].

Most importantly, the location of those mobile devices gets tracked, which can compromise the privacy of the participating user. So, in the present world, although many people get allured to participate in the survey due to incentives, barely a few take part in the MCS survey [12] if privacy is not taken care of properly. So, in this game, we have used the concept of mixed reality to hide the original location of the players, as shown in Fig. 1. We have also used masked IDs to conceal the actual identity and details of the players. Mixed reality aims at blending the real and the virtual environment [13, 14].

The types of incentives a player gets while playing the game are entertainment as an incentive, taking service as an incentive, and monetary incentive. A user receives the incentive based on the area coverage, the location's remoteness, and the quality of data collected following the rules and regulations that the authority has laid down.

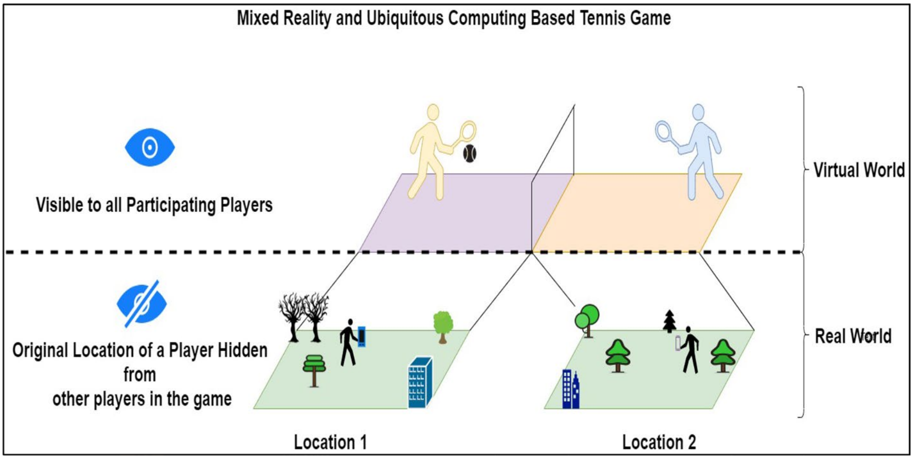

Fig. 1 Tennis game based on ubiquitous computing using mixed reality using mobile crowdsensing

And based on the collected data, Greedy Algorithm has been applied to recommend the locations where the mobile towers need to be installed. Greedy Algorithm aims at obtaining the globally best solution through achieving the locally best solution in each step.

2 Motivation and Contributions

A player cannot play a game without network coverage in the current and existing systems. But it is possible in our proposed game because the list of locations to be traversed next is fed into the player's device just before starting the game and or in the middle of the game every time the player is in the network coverage area. Only at the end of the game, the recorded sets of observations are uploaded to the server reducing the number of hits on the server significantly and saving network bandwidth. Contributions are as follows

1. SigSense is a geospatial, mixed-reality, incentive-aware game-based signal strength surveying system.
2. The game can be played even in areas without network coverage, and the survey can be performed.
3. Players have a choice for customizing their gaming area at the beginning of the game.
4. The player contribution-aware incentive mechanism has been introduced as per the surveyed area.
5. Illegal access to the App and Server APIs is restricted by preventing rooted Android devices and jailbroken Apple devices from taking part in the game, as this can affect the quality and authenticity of survey data acquired from those devices.
6. Based on the Greedy Algorithm, our system recommends locations to install new network base stations depending upon the area's signal strength and the base station choice selected by the network service provider or as recommended by our system by default.

3 Related Works

In [15], Broll and Benford have designed a game named Tycoon, considering the coverage of GSM cells. Each player competes with other players in this game to gain the most significant number of credits achieved by purchasing virtual objects like buildings in a predefined gaming area. Each GSM cell in the game area is virtually mapped to either a consumer or a producer. Each player collects resources from producers and exchanges for the virtual objects from consumers to earn credits: cell ID and GPS trace generated through these interactions are used to evaluate the spatial coverage of a GSM cell.

In [16], the Treasure game is designed to create maps of Wi-Fi coverage in the urban area as a by-product of the game-play. The game takes advantage of human movement but never involves human computation because the strength of the Wi-Fi signal is measured with computers' help.

In [17], we find that Drozd et al. have developed the game Hitchers, in which a player will expose virtual hitchers in their current GSM cell, which has a specific target destination. This virtual hitcher can be picked up and carried away by other players in the game, and in this way, these hitchers travel across the city and record a large number of trajectories used for building GSM cell-id maps.

There is a game known as EyeSpy [18], which is an application designed to generate navigation-ready photos and labels from mobile multiplayer games. In this game, a player takes a picture of a geographic location, and the other players have to find those places and validate them.

The proposed game SigSense has been designed to help us measure the signal's strength at various places based on mobile crowd sensing and helps the participating players earn incentives based on the game's rules.

4 Proposed Network-Monitoring Tennis Game Algorithm

4.1 Network-Monitoring Tennis Game Proposed Design

Our proposed model SigSense is a ubiquitous computing and mixed reality-based mobile signal strength surveying system across multiple regions by keeping the player's identity hidden. Moreover, in exchange for this survey, incentives are given to the players in e-wallet recharge, e-cash, mobile recharge. SigSense falls under the Hybrid Distribution Spatial Sensing Task category as the task is carried out with the help of GPS sensors fitted in the device [19], and the players playing in a game may or may not be close to each other.

In Fig. 2. the bi-directional arrow represents the communication between mobile devices and mobile towers. The green color bi-directional arrows represent that the mobile device is under the coverage of that mobile tower, and the dashed red color bi-directional lines mean that the mobile device is not under the range of the mobile base station. The recorded data will be transmitted to the cloud in JSON format only at the end of the survey to save the network bandwidth and reduce the number of hits on the server. A player will have to register them to the game server with the help of the POST request, only for the first time to Sign Up, and after that, they will have to sign in to their account using their user id. The game server will then allocate a Unique ID (UID) to the player for a lifetime and a Unique Masked ID (MUID) at the start of each game, which the operator will later use for the

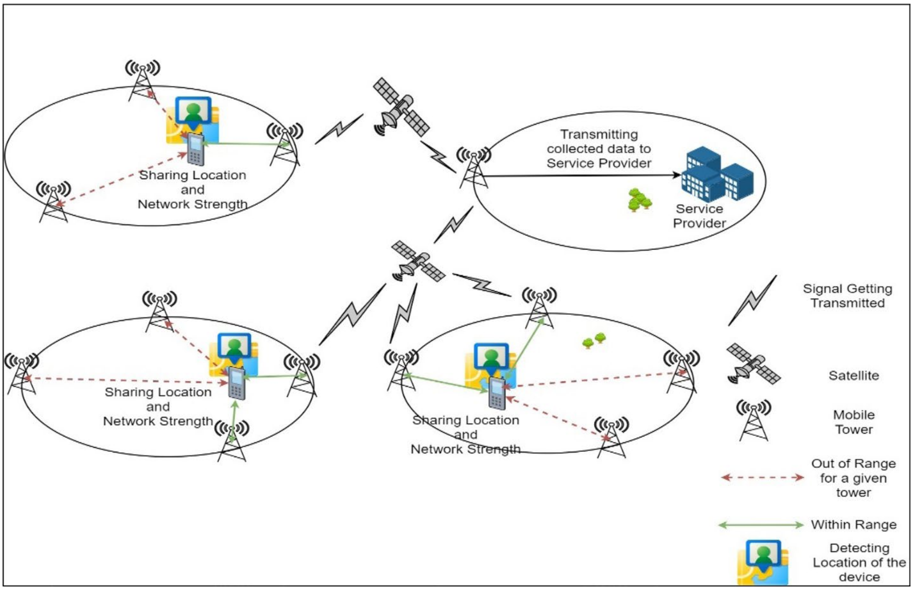

Fig. 2 The workflow of the network-monitoring tennis game using mobile crowdsensing

incentive mechanism. MUID is like an authentication token that is assigned to a player at the beginning of a game. It expires after the incentive calculation is over at the server for a particular game. We have proposed mixed reality in SigSense to hide the original location of the players, as shown in Fig. 1, to maintain the player's privacy. In the virtual world, opponent players will serve the ball at a given position in the court of a player in the game, which is mapped to any real-world physical location. The players will have to reach that physical location and hit the ball back. It will be an illusion to the players in the game that the ball they are serving or hitting is going to the court of the opponent player. A sample of the collected data is shown in Fig. 3. Rooted Android devices and jailbroken Apple devices are not allowed to participate because, in these types of devices, the end-user gets complete access to the device's system files and can easily modify those system files as per requirement. Malicious advance level users can change the application software files and create an illusion to the server that the player is playing, but in reality, they might not be playing the game and instead, through illegal API calls, can earn the incentive. So, to prevent this sort of unlawful activity, we have intentionally designed the Algorithm in such a way so that any rooted or jailbroken device can't earn any incentive, and their readings will not be considered.

As shown in Fig. 3, we will consider only the reading shown in the left side digital meter, giving the serving cell reading. We have considered Airtel Network Provider signal strength as a reference dataset. In the reading where the coordinates are displayed, the left reading signifies latitude, and the right indicates longitude. Signal strength is −108 dB at latitude 22.9565482 and longitude 88.543066, as shown in Fig. 3.

In [5], MCS has been decomposed into a four-layered architecture comprising the application, data, communication, and sensing layers. This paper has re-introduced these layers as per our video game into four corresponding layers: virtual World layer, Data layer, communication layer, and sensing layer, as shown in Fig. 4. In the virtual world, the players

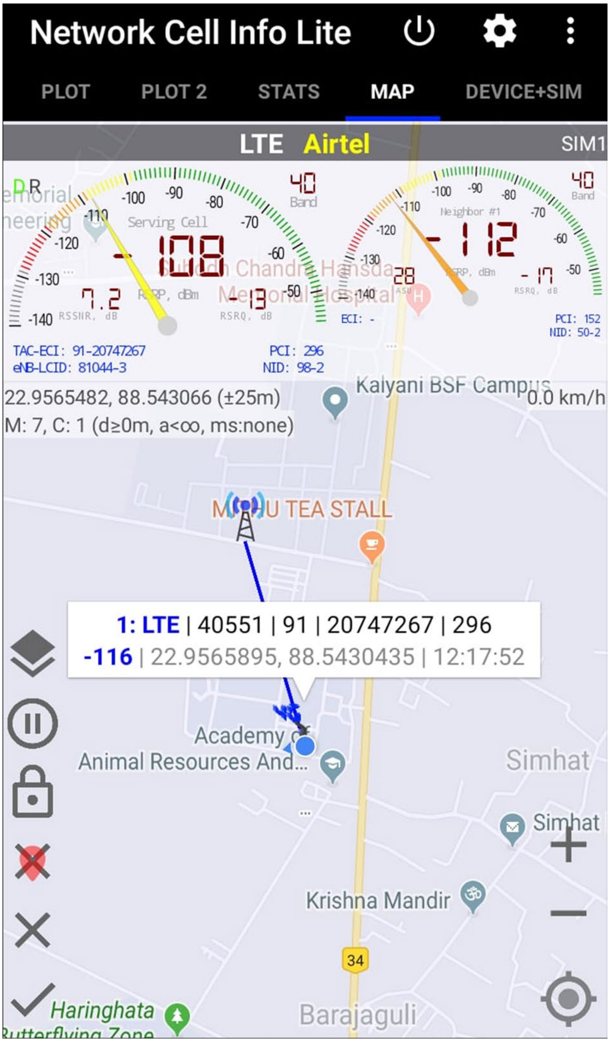

Fig. 3 Network cell lite info sample reading of the network strength

play the game on a tennis court on the device screen. In the data layer, we are storing the sensed data in JSON format. In the communication layer, we use the mobile network to set a communication link between the handheld device and the server; the last layer is the sensing layer used for sensing real-time data for our signal strength survey in various locations. The implementation of this model in the real-world scenario happens in the way, as shown in Fig. 1, where the player's original location is hidden from the other players. The players are playing in a Virtual World Game, the tennis court.

4.2 The Symbol Table For Variables in the Algorithms

See Table 1.

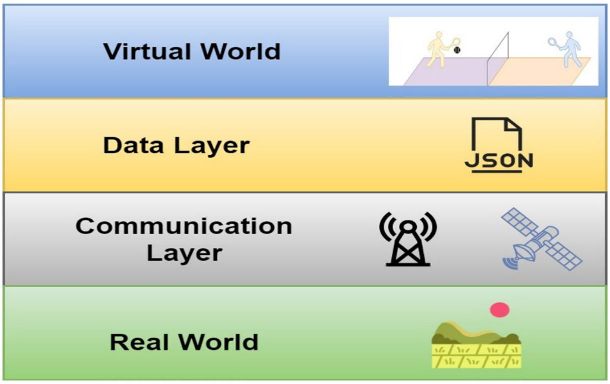

Fig. 4 Four-layer architecture of the Mobile Crowdsensing model based on ubiquitous computing and mixed reality

4.3 Network-Monitoring Tennis Game Algorithm for Registering User

Algorithm I: Network-Monitoring Tennis Game Registering Algorithm

Input: *IMEI, usr, mob, L, OID, OS, M, Area, Time*
Output: *UID, MUID* will be assigned to the player
Procedure:

```
Start
    Check BP, BS
    If BP<50 and JBRA==False
        If BP<30
            Display "Battery is too Low for Participating in the Game."
            Return false
        End If
        If BS== "Charging" and BP > 30
            Display "At least charge 50%, and then you can play the game."
            Return false
        End If
    Else
        If JBRA==True
            Not Allowed to participate in the game
            return false and exit
        End If
        PIR=true
        If the device is in Roaming
            Warn the User about that and ask for the willingness of the user to participate
            If the user is willing to participate
                PIRL =true
            Else
                PIRL =false
                return false and exit
            End If
        End If
          Area= selectArea()// stores the area that the player wants to play within
          UID=checkUID()// returns UID if already registered player and if the new player then
      returns a newly generated UID and returns NULL if the IMEI number of the device has been
      previously reported to be jailbroken or rooted
          Time=getCurrentSystemTimestamp()
          MUID=checkMUID()// returns MUID id for a particular game
        If PIRL == true and a_w_s == Yes and UID != NULL
            Send the IMEI, usr, mob, L, OID, OS, M, Area, Time, UID, MUID
         Else
           return false
        End If
    End If
End
```

Table 1 Symbol table for variables used in the different algorithms

Variable	Purpose	Algorithm number
IMEI	Stores the IMEI number	4.3
Usr	Stores the Name of the user of the device	4.3
Mob	Stores the mobile number of the SIM taking part in the game	4.3
L	Stores the location of the participating smart device	4.3
OID	Stores the Operator ID	4.3,4.5
OS	Stores Device Operating System and its version	4.3
M	Stores Model of the device	4.3
Area	Stores range of Area selected by the player to do the survey	4.3
Time	Timestamp when the player started to register for a game	4.3
UID	Unique ID of a player assigned to the player first time when he registers for the game	4.3,4.6
MUID	Masked Unique ID to hide the identity of a player	4.3,4.4
BP	Battery Percentage	4.3
BS	Battery Status, whether charging or discharging	4.3
JBRA	Flag value returns True if the device is Jail Broken or Root Access else returns False	4.3,4.4,4.5
PIR	Flag to represent whether willing to Participate In Roaming	4.3
PIRL	Flag to represent Participate In Roaming and Local	4.3
a_w_s	The flag representing Allowed for automatic switching from Normal to roaming in-between game, this flag stores the permission from the user	4.3
SS	Signal Strength at specific positions	4.5
RS	Current Roaming Status of the player returns True if playing in roaming else false	4.5
NOTMOSA	Number Of Times player Moved Out of the Selected Area	4.5,4.6
GOOS	Flag storing the value of Game On Off Status from the server. This option is provided so that the network service provider can explicitly stop a game from the server	4.5
PIRA	The flag representing a player is Playing In a Roaming Area	4.5
CLLat	Current Location Latitude	4.5
CLLon	Current Location Longitude	4.5
TID	Tower ID	4.5

Table 1 (continued)

Variable	Purpose	Algorithm number
LPSP	List of Places Surveyed by a Player	4.6
NOGP	Number Of Games Played	4.6
NUPS	Number of Unique Places Surveyed that is having no signal coverage	4.6
UP	Uniqueness Point earned for surveying a place	4.6
AOIETD	Amount Of Incentive Earned Till Date	4.6
AOIECG	Amount Of Incentive Earned in the Current Game	4.6
NOSD	Number Of Services Done	4.6
NOSurvDone	Number Of Surveys Done	4.6
SAOSFOTP	If the player is Selecting the Area Of the Survey From Own Teams Playing area denoting flag	4.4,4.6

Algorithm 1 is used to check whether all the essential requirements to participate in a game are satisfied by a particular smartphone. If all the basic requirements of the device are met, then the device can register and join a game.

4.4 Network-Monitoring Tennis Game Algorithm for Player Sending Request

Algorithm II: Serving Player's Algorithm

Input: *MUID* of the current Player and other participating players of that particular game,
Output: JSON containing the locations to visit for the survey by the other players, deviceRootedFlag
Procedure:
1. **Start**
2. Fetch the current *MUID* of the current Player
3. *GOOS*=on, deviceRootedFlag=False
4. Store the *MUIDs* of the opponent players in that game
5. Select a *MUID* from the list of *MUIDs* stored in the array of the current user
6. Store the path the target player will have to take to reach the location
7. n<- Store number of locations to traverse by each of the opponent players
8. m<- Fetch the number of opponent players in the game
9. **for** i = 1 to m
10. **Loop Begins**
11. *GOOS* = Check game status from server // *returns false if the game is over*
12. **If (**gameDurationOver()==False&&<50% player willing to end game and GOOS = = on)//gameDurationOver() if time limit of the game is over, it returns True else returns False
13. Create an array of location objects for storing the locations to be visited by player **i** and save it in the JSON format
14. **If the** area of survey is from the player's team member's area
15. **SAOSFOTP++** and store the number of times the player has done this in JSON
16. **End**
17. **Else**
18. Disable option for sending a request to other players
19. **Loop Ends**
20. POST the JSON data to the server
21. **If** *JBRA*==True
22. Send a message to the player "Your device is rooted/ jailbroken; hence you are not allowed to participate in the game."
23. *deviceRootedFlag* =true
24. return false
25. **End If**
26. **End**

Throughout the game, a list of opponent players will be shown to a player. As described in Algorithm II, he can allocate single or multiple coordinates to each opponent team player and send the data to the server both at the start and middle of the game as per game rules.

4.5 Network-Monitoring Tennis Game Algorithm for Player Receiving Request

Algorithm III describes how the signal strength survey of various places is taken at the receiver end and how the incentive is calculated when the game is being played.

Algorithm III: Receiving Player's Algorithm

```
Input: JSON containing the list of places to traverse by the current player
Output: JSON containing the following values S.S., JBRA, NOTMOSA
Procedure:
Start
    Initialize an array otherPlayerRequest []=getRequestsFromServer()// getting sets of requests for
    survey from the server
    GOOS=on, deviceRootedFlag=False
    counter = 0, PIRA =0
    NOTMOSA =0
    numberOfRequests=0
    While GOOS = = on
    Loop Begins
        If (gameDurationOver()==False&&<50% player  willing  to end game)
            Receive Requests of other players from the server and insert that in otherPlayerRequest []
            request_i = otherPlayerRequest [counter++]
            Store R.S., S.S. _i. CLLat, CLLon, Time, request _i, TID and its OID, numberoftimes that
            location has selected for survey by incrementing its value by one respectively and push these
            values in a JSON
            If the player moves out of the area allocated to them
                NOTMOSA++
            End If
            If the player is in a roaming area and surveying
                PIRA ++
            End If
            GOOS = Check game status from server // returns false if the game is over
        Else
            disableAcceptingRequestOfSurveyFromOtherPlayers()
            If allSurveysOver()
                break
    Loop Ends
  If JBRA==True
    deviceRootedFlag=true
    Transmit NOTMOSA, PIRA, deviceRootedFlag in a JSON format to the server
  Else
    Transmit SS, NOTMOSA, PIRA, deviceRootedFlag in a JSON format to the server
  End If
End
```

4.6 Network-Monitoring Tennis Game Algorithm for Incentive Calculation

Algorithm IV: Incentive Issuing Algorithm

Input: JSON containing the LPSP, UP, NOGP, NUPS, AOIETD, NOSurvDone, NOSD, deviceRootedFlag, NOTMOSA, UID
Output: Signal strength at those specified locations and incentive calculations and other game parameters like the number of games played and the number of unique spots surveyed will get stored in the server database and updated each time the player plays the game.
Procedure:

1. **Start**
2. NUPS, NOGP, AOIETD, *numberOfSureveysDone,* AOIECG=0, NOTMOSA gets stored in this variable after fetching data from the server
3. NOGP ++
4. **If** *deviceRootedFlag*==True
5. *AOIETD =0 // if device is jailbroken or rooted, all the incentives will be forfeited from player*
6. return false
7. **End If**
8. **for** i = 1 to numberOfObservationsStoredInJson
9. **Loop Begins**
10. **if** location [i]! = NULL
11. *UP* = *checkUniqueness*(location [i])// returns 10points if surveyed for 1st time, 5, 2 points if surveyed second or 3rd time else returns 1 point; returns 20 points, 15 points, 12 points and 11 points respectively if surveyed while in roaming
12. **if** *location*[i]. surveyed () = = true // returns true if the player surveyed else returns false indicated the player serviced the location to other players for survey
13. *AOIETD += UP*
14. *NOSurvDone ++*
15. **else**
16. *AOIECG += (UP)/* locationSurveyedByNumberOfPlayers(location[i])
17. *NOSD ++*
18. **End if**
19. **if** *UP* = = 10 or *UP* ==15
20. *NUPS ++*
21. **if** *NUPS* MOD 5 = = 0
22. issuePremiumTokenToPlayer(playerid) // issuing premium token to the player
23. **End if**
24. **End if**
25. **End if**
26. **Loop Ends**
27. *AOIECG += AOIECG *(NOTMOSA+ SAOSFOTP) *(-0.5)// deducting points for moving out of selected area and selecting area of survey from own team member's area*
28. *AOIETD += AOIECG*
29. **if** *AOIETD* MOD 1000 = = 0
30. notifyServiceProviderToGiveIncentive(playerUniqueID)
31. **End If**
32. **End**

Algorithm IV shows how incentives are calculated for each of the players after the game is over. It also highlights the various penalties levied on the players if they break the game's rules.

4.7 Network Tower Installation Location Recommendation System

Algorithm V: Optimal Mobile Tower Installation Algorithm

Input: JSON containing the network strength of the surveyed area
Output: Locations where the mobile towers are required to be installed
Procedure:
1. Start
2. blockScores[] = calculateScoreOfEachBlockOfMatrixBasedOnSignalStrength()
3. defaultMobileTower = selectDefaultMobileTowerAtBeginningByUser()// this will give user option to select default mobile tower to be considered
4. **If** defaultMobileTower == NULL
5. defaultMobileTower = MicroCell()// stores the default details of a MicroCell from the API
6. defaultMobileTower.getDetailsAboutMobileTower()
7. **Else**
8. defaultMobileTower.getDetailsAboutMobileTower()//gets the details of the type of the tower selected
9. **End If**
10. sortArrayInAscendingOrderBasedOnBlockScore()// for setting highest priority to the block having the least score
11. blocks_with_bad_signal_strength[]=checkTheCoordinatesThatAreAssociatedWithTheLeastScores()// returns list of blocks with top 3 lowest scores
12. **If** anyOneOfTheBlockDonotHaveScore(40,35,30,25,20)// returns true if any one of the block in the matrix don't have scores 40 or 35 or 30 or 25 or 20
13. Recommend user the coordinate where to install the base station based on the guidelines that have been laid down in Table 5.
14. towerChoice=Take the details of the base station from the user about the range of the base station and fetch its details from server
15. **If** towerChoice == NULL
16. towerChoice=defaultMobileTower
17. **End If**
18. update the score of all the blocks that are under the network coverage area of the Base Station by calculating distance between the tower and the blocks using Haversine formula
19. updateScoreOfAllBlocksInTheMatrix()
20. sortArrayInAscendingOrderBasedOnBlockScore()
21. blocks_with_bad_signal_strength[] = checkTheCoordinatesThatAreAssociatedWithTheLeastScores()
22. **End If**
23. Continue steps 12 to 22 till all the blocks in that matrix or area are either having one of the following scores 40 or 35 or 30 or 25 or 20
24. **End**

In Algorithm V, we try to get the locally best solution by recommending installing a base station to the block with the lowest score. After a tower is planted in a given iteration, in the next iteration, new block scores are generated based on the rules laid down in Table 5, and the blocks are sorted in ascending order, and again we are finding the block with the lowest score and planting the tower in that block, and the process continues till all the blocks have an optimal score. In this way, we provide the globally best solution using the Greedy Algorithm by solving the network coverage problem locally. In this system, we are Greedy for better signal strength. The main advantage of this system is that we allow the network service provider to select the type of base station they want to plant, based on their budget and requirements, and then accordingly, we recommend where to place the next mobile tower in the next iteration. Suppose the user does not provide any input, then by default, the system will consider the MicroCell base station and recommend the locations where the mobile signal towers are to be planted. This provides an optimal solution for the network service provider while planning to install the next network tower.

Network-monitoring tennis game rules

1. The network service provider will decide the maximum amount of area a player can survey in a single game. This rule has been laid down so that every player gets a fair amount of opportunity to participate in the game and earn incentives.
2. No user can have an overlapping area or boundary.
3. Different games can have overlapping areas.
4. Before the start of the game, the players will have to select their area, which is either square or rectangle in shape.
5. If the player moves out of their selected area after the game starts, their incentives will get deducted. The App installed in the mobile phone will itself check it based on GPS location; there is no need to do continuous Server-Client communication to track that.
6. The server can check a player's current location if required.
7. If a device is jailbroken or rooted before the start of the game, it cannot take part in the game. Nevertheless, suppose the user roots the device even after taking part in the game. In that case, the incentives earned by the player in his entire career will be forfeited by the network service provider, and the player will be informed only after the end of the game.
8. Pn- > Pm represents that Pn is hitting the ball in Pm's court. When Pn- > Pm, Pn will send the place where the virtual ball will go, and Pm will have to reach the position using GPS. After reaching the place, the signal strength at that position will be tested and recorded automatically.
9. Pn can determine the path for P_m, and if P_m follows the way, that will help P_m earn an extra incentive.
10. If a player is at a location where there is no signal, two scenarios can occur. A player is neither able to send or receive requests for the survey of coordinates. To deal with this scenario, we have proposed an idea. Any player can select multiple places for the opponent player and send the requests collectively to the Server in JSON format. At the same time, on the receiving end, the device can store multiple locations that are allotted to that player to be surveyed on the request from the opponent player(s). And this transmission of data in JSON format will occur when a player is at a location with network coverage with sufficient signal strength.
11. In a particular game having more than two players, the exact same location can be assigned to a player by the other two or more opponent players. If the player earns any incentive by traversing that place, then that incentive allocated to survey that location will be equally distributed among all the ball servers.
12. Incentives will be in the form of E-wallet cash or mobile recharge or recharge coupons.
13. Whenever the total number of spots surveyed by a player in the entire has no signal coverage is divisible by 5—a premium token issued to the player from the server.
14. Minimum 2 players are required to play the game, one from each side at least, else the game will not start or get terminated in the middle of the game.
15. A player cannot select a location for the survey by the opposite team from within its team area; instead, they have to choose a location from the opponent player's area compulsorily.
16. If a player selects a place from their own team's area, they will be fined.
17. The winner and loser of a game are determined based on the amount of incentive points earned by the respective teams; It will make the players explore new places to maximize the chance of winning more games and gaining more and more incentives.
18. The game ends either when 50% or more than 50% of players in the game are willing to end the game or when the time duration of the game is over, but before exiting the game, each player will have to finish their list of surveys allocated to them, and exit the game through game GUI to earn incentives.

19. The network operator has the discretionary power to allow/ disallow an area surveyed.
20. Points earned to incentive amount in the currency will be dependent on the network service provider
21. Rules of giving points to players based on the area traversed are shown in Table 2:-

5 Case Studies of Network-Monitoring Tennis Game Using MCS

Four different scenarios when a user is taking part in a game have been shown in Fig. 5.

Each of these four scenarios has been explained below:

Scenario 1: Adjacent to a mobile signal tower.

In this scenario, we can see that the player is in a location where there the signal strength is relatively high. So, the communication of the mobile device with the server will get an ample amount of bandwidth.

Scenario 2: Zone with zero network coverage.

Here we find that the player is in a zone where there is no signal coverage at all. In this type of situation, the player will be assigned the next place to visit for a survey using the stored list of locations allotted to him by the opponent players. It is possible because when the player was there in an area of signal coverage, the next set of locations to be traversed is fed into its mobile device from the server. And also, when the player is in this zone, the player can't send any target area to the opponent players. But once the player comes in an area with network coverage, the player can allocate multiple locations to the opponent player(s).

Scenario 3: Intersection point.

In this case, the user is under the coverage of two network towers so that the network strength will be optimum for the communication of the device between the mobile device and the server.

Scenario 4: Weak signal coverage.

In this case, the player is in an area where the signal strength is weak; here in this region, the device can also communicate with a server, but the data transmission rate will be less.

6 Results and Discussions

Figure 6 shows the area that we have taken into consideration for the survey. In Fig. 6, we have divided the region into multiple parts using rhombus. The two rhombuses that are further subdivided into yellow color smaller blocks are considered for playing the game. The area shown in Fig. 6 is about 45 acres, which is not feasible for a single player to cover efficiently. Each player will have their distinct area of survey, and as mentioned above, and no two players will be allowed to cover the same piece of land in a single game. The area is divided into parts to give all players a fair amount of chance to participate in the game and earn an incentive. This initiative is taken to prevent the greedy players from taking the considerable land area for surveys under them.

We have deployed Player 1 and Player 2 in the subdivided smaller areas that are highlighted in yellow, as shown in Fig. 6. In this survey, we have related each area allotted to a player with

Table 2 Points provided to a player based on location surveyed

The total number of times a place is explored in all the games that are played	Serving player(s) points SP- > Number of players in a game who have selected a particular spot for survey	Receiving player points
1st time	10/S.P., i.e., if SP = 4, then each player serving player will get 10/4 = 2.5 points as the score	10 points
2nd time	5/SP	5 points
3rd time	2/SP	2 points
> 3 times	1/SP	1 point

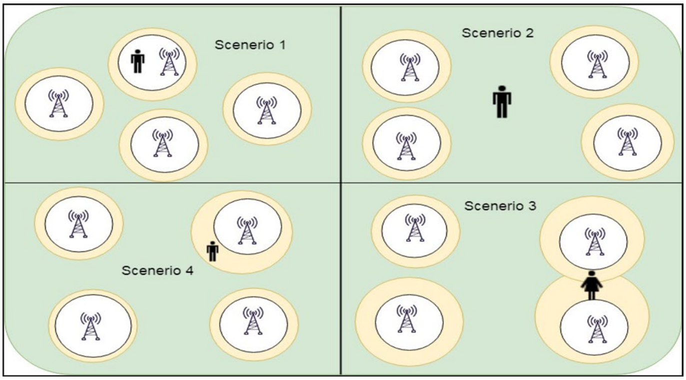

Fig. 5 Various scenario that can occur while playing the game

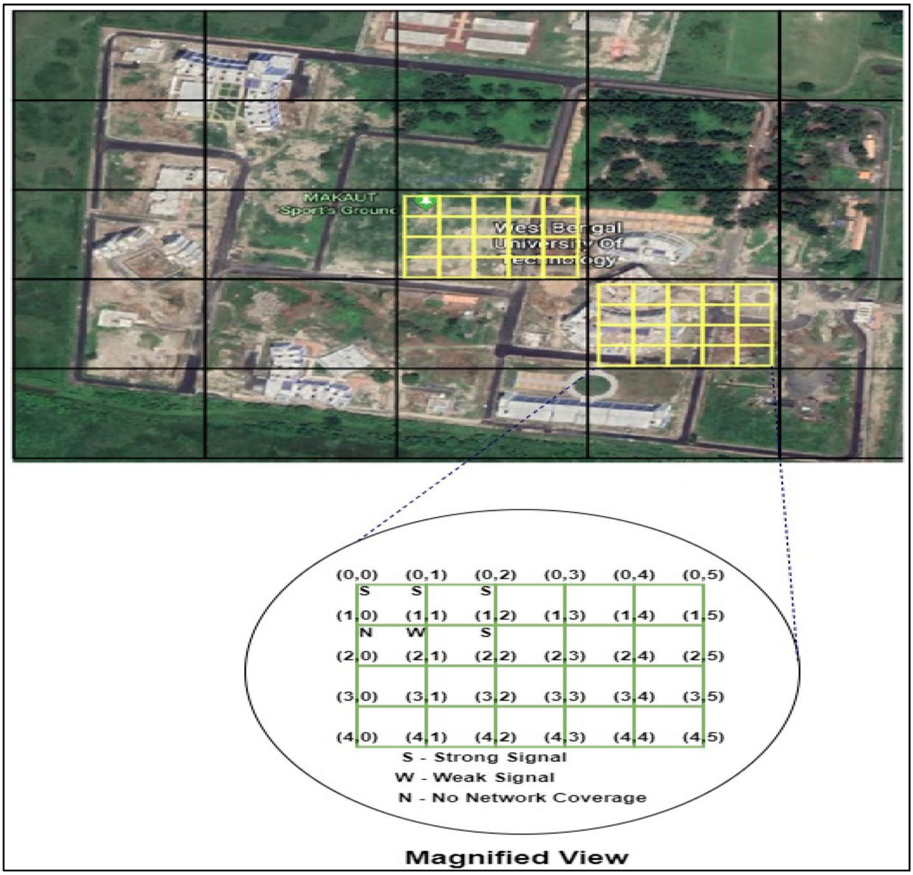

Fig. 6 Satellite image of the Maulana Abul Kalam Azad University of Technology West Bengal, main campus that is considered under survey

Table 3 Live data taken from player 1 (PA)

Latitude	Longitude	Coordinate in matrix	Representation PA < Coordinate no. >	Signal strength (in dB)
22.9575447	88.5428937 (± 24 m)	(0, 0)	PA 00	– 105
22.9577588	88.5431079 (± 63 m)	(0, 1)	PA 01	– 107
22.9577762	88.5433507 (± 16 m)	(0, 2)	PA 02	– 97
22.9577034	88.5434878 (± 7 m)	(0, 3)	PA 03	– 90
22.9576981	88.5435345 (± 10 m)	(0, 4)	PA 04	– 96
22.9576825	88.5435839 (± 16 m)	(0, 5)	PA 05	– 92
22.9583905	88.5417651 (± 98 m)	(1, 0)	PA 10	– 100
22.9575611	88.5425759 (± 91 m)	(1, 1)	PA 11	– 102
22.9574682	88.543055 (± 18 m)	(1, 2)	PA 12	– 89
22.9574588	88.5432415 (± 20 m)	(1, 3)	PA 13	– 91
22.957509	88.5434939 (± 10 m)	(1, 4)	PA 14	– 93
22.9574588	88.5436441 (± 21 m)	(1, 5)	PA 15	– 99
22.9566574	88.5425532 (± 546 m)	(2, 0)	PA 20	– 101
22.9570635	88.5429437 (± 14 m)	(2, 1)	PA 21	– 109
22.9573394	88.5431042 (± 20 m)	(2, 2)	PA 22	– 97
22.9574361	88.5435191 (± 13 m)	(2, 3)	PA 23	– 91
22.9566569	88.5430451 (± 45 m)	(2, 4)	PA 24	– 101
22.9573126	88.5437944 (± 24 m)	(2, 5)	PA 25	– 92
22.9568988	88.5427784 (± 14 m)	(3, 0)	PA 30	– 93
22.9570709	88.5428594 (± 20 m)	(3, 1)	PA 30	– 86
22.956788	88.5429408 (± 22 m)	(3, 2)	PA 32	– 83
22.9566092	88.5430065 (± 93 m)	(3, 3)	PA 33	– 90
22.9570197	88.5435025 (± 15 m)	(3, 4)	PA 34	– 94
22.9564092	88.5430067(± 88 m)	(3, 5)	PA 35	– 92
22.9568471	88.5428762 (± 33 m)	(4, 0)	PA 40	– 86
22.9565901	88.542862 (± 24 m)	(4, 1)	PA 41	– 92
22.9567627	88.542999 (± 14 m)	(4, 2)	PA 42	– 90
22.9566997	88.5429675 (± 15 m)	(4, 3)	PA 43	– 95
22.9567088	88.5428895 (± 24 m)	(4, 4)	PA 44	– 85
22.9568515	88.5436317 (± 13 m)	(4, 5)	PA 45	– 102

a matrix. In this matrix, the intersecting points denote the coordinates that will be surveyed. The coordinates of the matrix taken into consideration for the survey are shown in the Magnified View of Fig. 6. We have used Network Cell Info Lite for collecting the observations. Considering (1), We have observed the signal strength of the Airtel 4G sim as our reference. (2) In the observation tables and graphs given below, coordinate (m, n) is represented as 'mn' for the sake of understanding. So, in this way, if we consider the entire University campus as a large Matrix divided into smaller submatrixes, it will be pretty helpful for the players to survey the coordinates in those smaller matrixes in detail.

Given below is the survey report generated from a game. Tables 3 and 4 contain the survey report of the signal strength recorded by Player 1 and Player 2, respectively.

Table 4 Live data taken from player 2 (P2)

Latitude	Longitude	Coordinate in matrix	Representation P.B. <Coordinate no.>	Signal strength (in dB)
22.9568234	88.5425682 (± 29 m)	(0, 0)	PB 00	– 89
22.9566516	88.542879 (± 78 m)	(0, 1)	PB 01	– 94
22.9584093	88.5425532 (± 10 m)	(0, 2)	PB 02	– 95
22.9584817	88.5424886 (± 17 m)	(0, 3)	PB 03	– 92
22.9586469	88.5421668 (± 80 m)	(0, 4)	PB 04	– 103
22.9584574	88.5424937 (± 50 m)	(0, 5)	PB 05	– 90
22.9584487	88.5421269 (± 16 m)	(1, 0	PB 10	– 91
22.9573476	88.5426238 (± 37 m)	(1, 1)	PB 11	– 97
22.9612236	88.54339 (± 2200 m)	(1, 2)	PB 12	– 98
22.958344	88.5424681 (± 26 m)	(1, 3)	PB 13	– 95
22.9583312	88.5425988 (± 14 m)	(1, 4)	PB 14	– 101
22.956616	88.5428011 (± 360 m)	(1, 5)	PB 15	– 107
22.9851355	88.542072 (± 23 m)	(2, 0)	PB 20	– 99
22.9568649	88.5427809 (± 44 m)	(2, 1)	PB 21	– 93
22.9579673	88.542367 (± 23 m)	(2, 2)	PB 22	– 93
22.9568359	88.542774 (± 40 m)	(2, 3)	PB 23	– 97
22.9580356	88.5425204 (± 10 m)	(2, 4)	PB 24	– 99
22.958107	88.542536(± 18 m)	(2, 5)	PB 25	– 101
22.9580095	88.5421614 (± 32 m)	(3, 0)	PB 30	– 97
22.9575099	88.5424127 (± 14 m)	(3, 1)	PB 31	– 92
22.9580013	88.5423464 (± 10 m)	(3, 2)	PB 32	– 99
22.9575882	88.5425882 (± 33 m)	(3, 3)	PB 33	– 104
22.9573778	88.5426465 (± 57 m)	(3, 4)	PB 34	– 101
22.9579605	88.5426992 (± 15 m)	(3, 5)	PB 35	– 102
22.9579284	88.5421294 (± 10 m)	(4, 0)	PB 40	– 92
22.957954	88.5423033 (± 22 m)	(4, 1)	PB 41	– 93
22.9577613	88.5423683 (± 35 m)	(4, 2)	PB 42	– 99
22.9579405	88.5424889 (± 88 m)	(4, 3)	PB 43	– 95
22.9576516	88.542879 (± 83 m)	(4, 4)	PB 44	– 101
22.9578617	88.5425928 (± 118 m)	(4, 5)	PB 45	– 103

Figure. 7 and Fig. 8 show signal strength across different coordinates in the areas where Player 1 and Player 2 have played the game. The dotted line shows the average trend of the signal strength across the coordinates for each player.

This formula will give us the sample number over an outdoor walking test path:

$$Ns = L\frac{Nm}{V} \tag{1}$$

Here, $\mathbf{N_s}$ means the total number of collected samples, Nm is the number of collected samples per second, L is the path's length, and V is the walker's speed [20].

In [21], we observed different algorithms used to detect the user location and the route with the help of the serving cell-ID or NodeB to which the user is connected. The second Algorithm takes into consideration all NodeBs and then calculates the

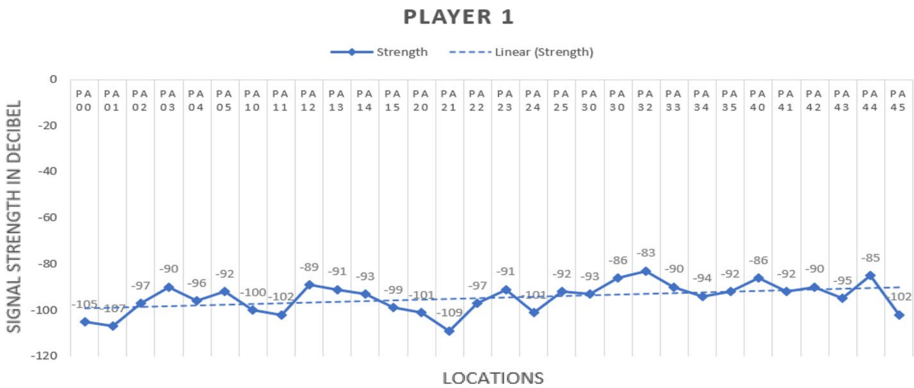

Fig. 7 Graphical representation of signal strength across multiple locations for Player 1

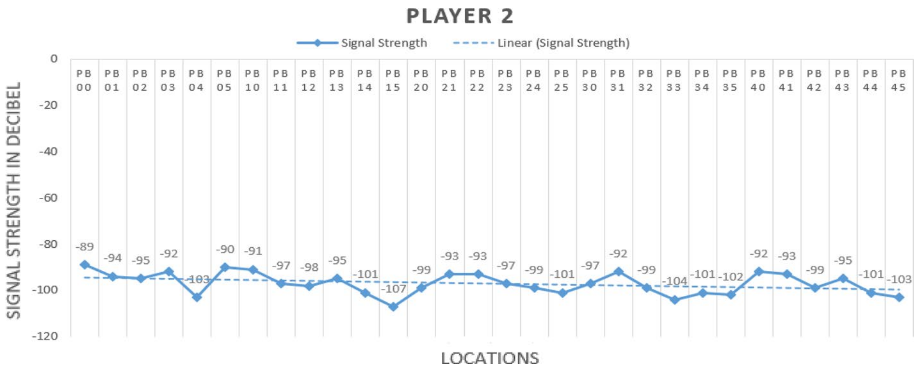

Fig. 8 Graphical representation of signal strength across multiple locations for Player 2

centroid. If there is only a single NodeB, this Algorithm's result is the same as the cell-ID technique.

Depending upon the signal strengths across all the coordinates as shown in the Magnified View of Fig. 6, we have classified them into three categories, namely:- Strong Signal Strength Area(S), Weak Signal Strength Area(W), No Signal Coverage Area(N). For the sake of understanding Table 5, we have shown all the scenarios that we have taken into consideration before deciding whether to install a mobile tower or not and where to install a Mobile Signal Tower. Each row in the table represents possible scenarios of each and individual block of the matrix, as shown in the Magnified View of Fig. 6. For instance, the underlined values, or in row 5 in Table 5, represent a scenario where a particular block has two coordinates with Strong Signal Strength, one coordinate with weak signal strength, and one coordinate has no signal coverage.

We have assigned block scores to each type of block depending upon the signal strength at all its four coordinates. For instance, let us take into consideration row 5 from Table 5. Well will give each coordinate of a block 10 points for every **S**, 5 points for **W,**

Table 5 Various Scenarios we have taken into consideration to decide whether to install new mobile tower(s) and if yes, then how many new mobile towers at what locations for minimal costing and optimal result

Strong signal strength (**S**)	Weak signal strength (**W**)	No signal (**N**)	Possible scenarios with these conditions	Whether the installation of the tower required	Where to install the tower	The score is given to each block of the matrix as per **S**, **W**, **N**
4	–	–	All coordinates having good network coverage	No	N.A	40
3	**1**	–	Only one coordinate has **W**	No	N.A	35
3	–	**1**	Only one coordinate has **N**	Yes	At **N**	31
2	**2**	–	1. Two coordinates with **W** are adjacent	No	N.A	30
			2. Two coordinates with **W** diagonally opposite to each other	No	N.A.	30
2	**1**	**1**	1. One **W** and one **N** adjacent to each other	Yes	At **N**	26
			2. One **W** and one **N** diagonally opposite to each other	Yes	At **N**	26
1	**3**	–	All coordinates having optimal signal coverage	No	N.A	25
2	–	**2**	1. Two **N**s adjacent to each other	Yes	At anyone **N**	22
			2. Two **N**s diagonally opposite to each other	Yes	At anyone **N**	22
1	**2**	**1**	One coordinate has **S**, another one coordinate has **N,** and two coordinates have **W**	Yes	At **N**	21
–	**4**	–	All four coordinates are **W**	No	N.A	20

Table 5 (continued)

Strong signal strength (**S**)	Weak signal strength (**W**)	No signal (**N**)	Possible scenarios with these conditions	Whether the installation of the tower required	Where to install the tower	The score is given to each block of the matrix as per **S**, **W**, **N**
1	**1**	**2**	1. Two **N**s adjacent to each other	1. Yes/No	1. Check adjacent blocks, and if they have block scores 31 or 26 or 21 or 22 or 16 or 12, don't plant a base station. Else plant a base station at any one **N**	17
			2. Two **N**s diagonally opposite to each other	2. Yes/No	2. Check adjacent blocks; if they have block scores of 31 or 26 or 21 or 16, don't plant a base station. Else plant a base station at anyone **N**	17
–	**3**	**1**	Only one coordinate has **N**	Yes	At **N**	16
1	**–**	**3**	Three coordinates have **N**	Yes	Check the adjacent blocks that are having the least score, and install a base station at any one of the **N** that is common in both these blocks	13
–	**2**	**2**	Two coordinates have **N**	Yes	Check the adjacent blocks that are having the least score, and install a base station at any one of the **N** that is common in both these blocks	12

Table 5 (continued)

Strong signal strength (**S**)	Weak signal strength (**W**)	No signal (**N**)	Possible scenarios with these conditions	Whether the installation of the tower required	Where to install the tower	The score is given to each block of the matrix as per **S**, **W**, **N**
–	**1**	**3**	Three coordinates have **N**	Yes	Check the adjacent blocks that are having the least score, and install a base station at any one of the **N** that is common in both these blocks	8
–	–	**4**	All the coordinates have **N**	Yes	Check the adjacent blocks that are having the least score, and install a base station at any one of the **N** that is common in both these blocks	4

Note: In Table5 , ‘**S**’, ‘**W**’, ‘**N**’ in bold is used to highlight: Strong Signal Strength Area(**S**), Weak Signal Strength Area(**W**), No Signal Coverage Area(**N**). And symbol ‘-’ signifies that a particular type of zone is not inside a square of the sub-divided matrix ’S’, ’W’, ’N’ are different types of zones based on signal strength. For example if we look at block J in **Fig.** 10, we will find that ‘**S**’ is absent and in it’s three corner there is ‘**W**’ and in one corner it is having ‘**N**’. So the score of that block is 16 and under ‘**S**’ column of this particular category of box we have marked it ‘-’ as shown in Table 5 and 1 for **N.** So, in this case, the total score of that block is—10*2 + 1*5 + 1*1 = 26 points.

By looking at these points allotted to each type of block, our Algorithm can quickly give a review to the network service provider the probability of installing a new mobile tower, as installing a mobile tower will incur an item of expenditure to the mobile network service provider. The formula for calculating the score is:-

$$\text{Score} = (10 * \text{No}: \ \text{of S}) + (5 * \text{No}: \ \text{of W}) + (1 * \ \text{No}: \ \text{of N})$$

From [22, 23], we know that the Haversine formula is required to calculate the huge circle distance using latitude and longitude as a parameter between any two coordinates on earth or any sphere, as shown in Fig. 9. Then from that, we are calculating the amount of area covered under a block.

From Fig. 9, we can calculate the area of the block in this way: -

The trigonometric equation for the haversine function is

$$harvesine(\theta) = \sin^2\left(\frac{\theta}{2}\right)$$

The Haversine of the central angle is calculated using the formula:-

$$\left(\frac{Distance}{Radius}\right) = harvesine(\alpha 2 - \alpha 1) + \cos(\alpha 1)\cos(\alpha 2) harvesine(\beta 2 - \beta 1)$$

Here *radius* signifies the radius of the earth, which is equal to 6371 km, and *distance* represents the distance between the two coordinates, $\alpha 1$, $\alpha 2$ is the latitude of both the points and $\beta 1$, $\beta 2$ are the longitudes of the two coordinates respectively.

Therefore, the distance between the two points, we calculate using the inverse sine function, that is

$$Distance = 2 * radius * \sin^{-1}\left(\sqrt{\sin^2\left(\frac{\alpha 2 - \alpha 1}{2}\right) + \cos(\alpha 1)\cos(\alpha 2)\sin^2\left(\frac{\beta 2 - \beta 1}{2}\right)}\right)$$

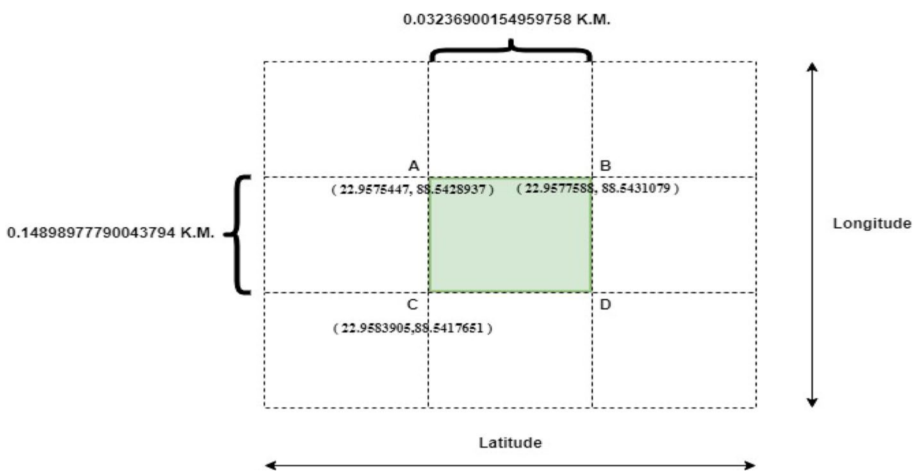

Fig. 9 Calculating distance between two coordinates in a block of a matrix

Let us assume we have three coordinates A (22.9575447, 88.5428937), B (22.9577588, 88.5431079), C (22.9583905, 88.5417651), and we have to calculate the distance between these coordinates and get the diagonal distance of the block formed using these three coordinates.

Distance between coordinate A and coordinate B(AB) = 32.36900154959758 m ≈ 32.37 m.

Distance between coordinate A and coordinate C (AC) = 148.98977790043794 m ≈ 149 m.

Distance between coordinate B and coordinate C (BC) = 154.39 m.

Therefore, the approximate length of the diagonal AD = $\sqrt{AB^2 + AC^2}$ = 152.47 m.

Network coverage of various Base stations is given below in Table 6. So, depending upon the area of coverage, network base stations will be selected, get an optimized solution that will provide sufficient bandwidths to the users.

Now depending upon the location and the purpose of the base station, the base stations will be placed, and accordingly, the signal strength of the nearby coordinates in the matrix will be updated, and the block score of each of the blocks will be recalculated based on Table 5. Every time a new mobile tower is placed in any coordinate, depending upon the range of coverage of the mobile base stations, the block scores will be updated as shown in Figs. 11 and 12. Updating the block scores is essential as it will help to find the locations where there is weak network coverage and give recommendations on where to plant the mobile tower. Figure 10 shows an area before the base stations are installed based on our recommendation system.

In Fig. 10, we have calculated the block score of all the blocks in the given matrix, and the blocks (**L, P**) that are having the least score as per the rules laid down in Table 5. have been highlighted with Red Color, and the blocks (**B, C**) having excellent signal strength in all its four coordinates are highlighted with Green Color. In Figs. 10, 11 and 12 **S**, **W**, **N** represent strong, weak, and no signal coordinates, respectively, and Green, Orange, Red colors represent blocks with the average signal strength strong, weak, and no signal respectively. After applying the recommendation system based on the block scores, we got a better result, which is shown in Fig. 11. After the first iteration.

After the first iteration, as shown in Fig. 11 of the recommendation system block L, P has strong signal strength, and their adjacent blocks and coordinates have a positive impact on the mobile signal strength due to the installation of the tower. But in blocks E, F, I, J, there is still a coordinate that has to go no signal coverage; hence we will have

Table 6 A comparison study of various base stations

Base station type	Coverage radius	Use
Femtocell	10 m	For home or office purposes
Domestic repeater	100 m	For the home, office, or factory use
Picocell	200 m	Multi-storied buildings, hotel, car parking
Microcell	1–2 km	Shopping centers, transport hubs, mine sites, city blocks, temporary events, or natural disasters
Macro-cell	5–32 km	Suburban, city, and rural areas
Macro-cell extended reach	50–150 km by using extender cell technology	Suburban and rural areas

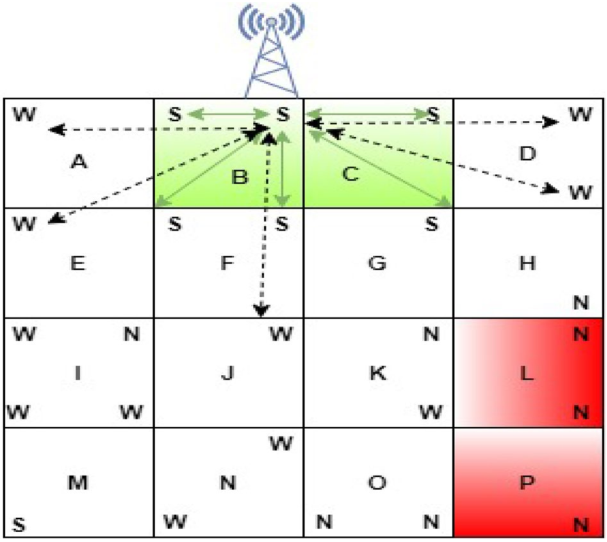

Fig. 10 Signal strength across various blocks in a matrix before a recommendation

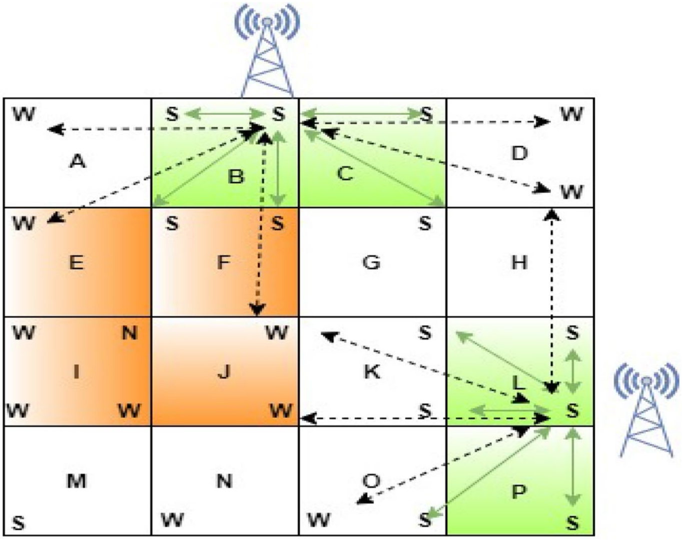

Fig. 11 Scenario of the blocks of the matrix after 1st iteration of recommendation

to install a base station and update the score of the blocks. Currently, the blocks E, F, I, J are having block scores of 21, 26, 16, 16 respectively, and then as per our recommendation system and rules laid down in Table 5, we will have to plant a base station at **N** common in blocks E, F, I, J. Figure 12 shows the updated blocks of the matrix after the installation of a base station in the next iteration.

The advantages of the proposed game are as follows**:**

1. Players across multiple cities/countries can play in a single game.

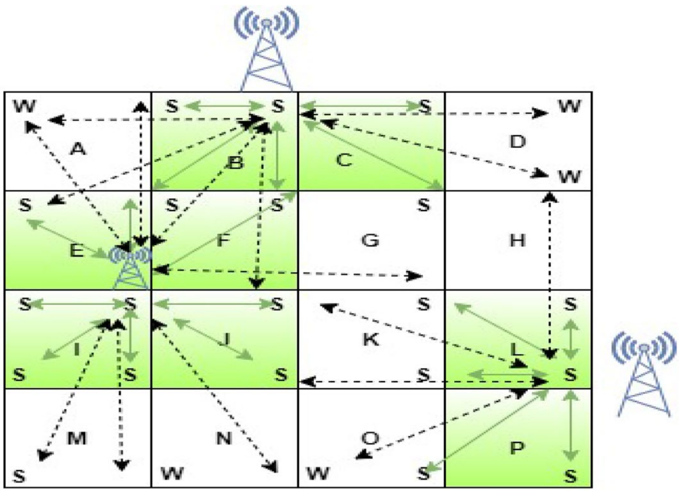

Fig. 12 Scenario of the blocks of the matrix after 2nd iteration of recommendation

2. The coverage area is flexible, and players can choose their area of the survey in their free will by abiding by the set of rules.
3. Only two players are sufficient to play this game.
4. New users who register for the first time are assigned a unique user id.
5. Each observation is first recorded in JSON format and stored in the device memory. The survey data is transmitted to the server only after the game is over, thus reducing the network bandwidth consumption.
6. As MUID is generated for each player at the start of a game, the original details of a player will be hidden from other players who are taking part in this game. This MUID will be obsolete after a survey report is collected and incentives are calculated after a particular game. MUID is a sort of authentication token from the server assigned to a user for identification in a match.
7. No rooted Android device or jailbroken Apple devices will be allowed to participate in this game, and their surveyed data will not be considered to avoid fake survey data.
8. As the areas are divided, so there is less chance for greedy players to take more amount of the area under their game area; this, in turn, will give a fair amount of opportunity for different users to take part in the game.
9. In free time or pass-time, players can play this game to earn incentives.
10. With the help of Mixed Reality, the original location of a player has been masked from other participating players [24].
11. We have given an optimal solution about installing a mobile network tower or not in a given area block.
12. We have also given an optimal solution about where to install the mobile tower, to reduce the cost of the Network Service provider behind the installation of the mobile towers.

Table 7 gives an in-detailed comparison study about the features and flaws of existing games in terms of privacy, security, and many other issues, and how proper measures to deal with those scenarios.

Table 7 Comparison table with various parameters with different games

	Proposed work SigSense	Tycoon [19]	Treasure game [5]	Hitcher [20]	Eyespy [21]
The minimum number of players required	Minimum 2	Multiplayer	Minimum 4 (2 teams of size 2)	Multiplayer	Multiplayer
The game can be played in the absence of a signal in the middle of the game	Yes	Yes	No	No	No
The incentive for all the contributing players in the form of cash/e-cash/mobile recharge	Yes	No	Yes	No	Yes
Each player will have their records of points and achievements across multiple games	Yes	No	No	No	No
Mixed reality used in the game for location privacy preservation	Yes	No	No	No	No
Technology Used	4G/5G Network, GPS	GSM Network, GPS	Wi-Fi Network, GPS	GPRS, position using the Cell ID	Wi-Fi Network, GPS
Continuous communication between Client and Server required	No	Yes	Yes	Yes	Yes
Places to Survey	Decided by Opponent players	Decided by Game	Decided by Server	Determined by and dependent on a group of players	Decided by Player
Observation and readings automatically are done by handheld devices	Yes, the player has to only go to the location. The device will itself collect real-time data	Yes, the player has to only go to the place, and the device collects data automatically	Yes, the player has to only go to the location, and the device collects information automatically	No	No, photos, tags, and labels manually were done by players
Area of survey	Both Rural and Urban	Cities and countries of California	Urban	Both Rural and Urban	City

Table 7 (continued)

	Proposed work SigSense	Tycoon [19]	Treasure game [5]	Hitcher [20]	Eyespy [21]
The game played across multiple countries/states at a time in a single game	Yes	Only within California	No	No	No
Masked Unique ID provided for the privacy of the game player	Yes	No	Not mentioned	No	No
The current location of the player is getting shared with the server	No	Yes, except when goes offline for gambling	Yes	Yes	Yes
Chance of SQL Injection in Database	No	N.A	N.A	Yes	N.A
Taking Permission from the player whether to continue to play in roaming	Yes	No	Not Required	No	Not Required
Possible hacking of mobile devices through compromised Wi-Fi access points is possible	No	No	Yes	No	Yes
Survey of common places by two or more players in a single game	No	Yes	Yes	Yes	Yes

7 Conclusion

Using SigSense is multi-purpose objective gets served at the same time, which includes privacy preservation, entertainment, incentives, survey, and recommendation to install new base stations. While proposing this system, we have kept the privacy preservation of a player as the highest priority through mixed reality and MUID, followed by the survey and the recommendation system. It's a win–win situation for both the network service provider and a player because the network service provider is getting the study of the signal strength through the game, and at the same time, the player is getting entertained and earning incentives for playing the game. SigSense provides a fair amount of opportunity for all the players to participate in the game and earn incentives. Overall, our proposed game is a complete package of everything, which makes a survey quite entertaining and attractive and secure at the same time for the players.

References

1. Pandey, S. R., Tran, N. H., Bennis, M., Tun, Y. K., Manzoor, A., & Hong, C. S. (2020). A crowdsourcing framework for on-device federated learning. *IEEE Transactions on Wireless Communications, 19*(5), 3241–3256.
2. Li, Z., Liu, H., & Wang, R. (2019). Service benefit aware multi-task assignment strategy for mobile crowd sensing. *Sensors, 19*(21), 4666.
3. Saadatmand, S., & Kanhere, S. S. (2019). MRA: A modified reverse auction based framework for incentive mechanisms in mobile crowdsensing systems. *Computer Communications, 145*, 137–145.
4. De, D. (2019). *Mobile cloud computing: Architectures, algorithms and applications*. Chapman and Hall/CRC.
5. Capponi, A., Fiandrino, C., Kantarci, B., Foschini, L., Kliazovich, D., & Bouvry, P. (2019). A survey on mobile crowdsensing systems: Challenges, solutions, and opportunities. *IEEE Communications Surveys & Tutorials, 21*(3), 2419–2465.
6. Koukoumidis, E., Peh, L. S., & Martonosi, M. R. (2011). Signalguru: Leveraging mobile phones for collaborative traffic signal schedule advisory. In *Proceedings of the 9th international conference on mobile systems, applications, and services* (pp. 127–140).
7. Costa, C., Laoudias, C., Zeinalipour-Yazti, D., & Gunopulos, D. (2011). SmartTrace: Finding similar trajectories in smartphone networks without disclosing the traces. In *2011 IEEE 27th international conference on data engineering* (pp. 1288–1291). IEEE.
8. Wu, C., Yang, Z., & Liu, Y. (2014). Smartphones based crowdsourcing for indoor localization. *IEEE Transactions on Mobile Computing, 14*(2), 444–457.
9. Thiagarajan, A., Ravindranath, L., LaCurts, K., Madden, S., Balakrishnan, H., Toledo, S., & Eriksson, J. (2009). Vtrack: Accurate, energy-aware road traffic delay estimation using mobile phones. In *Proceedings of the 7th ACM conference on embedded networked sensor systems* (pp. 85–98).
10. Yang, M., Zhu, T., Liang, K., Zhou, W., & Deng, R. H. (2019). A blockchain-based location privacy-preserving crowdsensing system. *Future Generation Computer Systems, 94*, 408–418.
11. Zhang, X., Yang, Z., Sun, W., Liu, Y., Tang, S., Xing, K., & Mao, X. (2015). Incentives for mobile crowd sensing: A survey. *IEEE Communications Surveys & Tutorials, 18*(1), 54–67.
12. Tao, D., Wu, T. Y., Zhu, S., & Guizani, M. (2020). Privacy protection-based incentive mechanism for Mobile Crowdsensing. *Computer Communications, 156*, 201–210.
13. Bekele, M. K., Pierdicca, R., Frontoni, E., Malinverni, E. S., & Gain, J. (2018). A survey of augmented, virtual, and mixed reality for cultural heritage. *Journal on Computing and Cultural Heritage (JOCCH), 11*(2), 1–36.
14. Flavián, C., Ibáñez-Sánchez, S., & Orús, C. (2019). The impact of virtual, augmented and mixed reality technologies on the customer experience. *Journal of Business Research, 100*, 547–560.
15. Broll, G., & Benford, S. (2005). Seamful design for location-based mobile games. In *International conference on entertainment computing* (pp. 155–166). Springer, Berlin, Heidelberg.
16. Barkhuus, L., Chalmers, M., Tennent, P., Hall, M., Bell, M., Sherwood, S., & Brown, B. (2005). Picking pockets on the lawn: the development of tactics and strategies in a mobile game. In *International conference on ubiquitous computing* (pp. 358–374). Springer, Berlin, Heidelberg.

17. Drozd, A., Benford, S., Tandavanitj, N., Wright, M., & Chamberlain, A. (2006). Hitchers: Designing for cellular positioning. In *International conference on ubiquitous computing* (pp. 279–296). Springer, Berlin, Heidelberg.
18. Bell, M., Reeves, S., Brown, B., Sherwood, S., MacMillan, D., Ferguson, J., & Chalmers, M. (2009). Eyespy: Supporting navigation through play. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 123–132).
19. Khan, F., Rehman, A. U., Zheng, J., Jan, M. A., & Alam, M. (2019). Mobile crowdsensing: A survey on privacy-preservation, task management, assignment models, and incentives mechanisms. *Future Generation Computer Systems, 100*, 456–472.
20. Ongbwa, B. A. E., Kora, A. D., Mbemba, H., & Amelete, S. (2018). Objective approach of 3G/4G networks indoor RF coverage assessment. In *2018 4th International conference on frontiers of signal processing (ICFSP)* (pp. 17–23). IEEE.
21. Trogh, J., Plets, D., Surewaard, E., Spiessens, M., Versichele, M., Martens, L., & Joseph, W. (2019). Outdoor location tracking of mobile devices in cellular networks. *EURASIP Journal on Wireless Communications and Networking, 2019*(1), 1–18.
22. Prasetya, D. A., Nguyen, P. T., Faizullin, R., Iswanto, I., & Armay, E. F. (2020). Resolving the shortest path problem using the haversine algorithm. *Journal of Critical Reviews, 7*(1), 62–64.
23. Geetha, K., & Kannan, A. (2019). An efficient information system for providing location based services in network environments. *Wireless Personal Communications, 109*(4), 2377–2398.
24. Wei, X., Hong, H., Daniel, D. Z., & Wei, W. (2017). Emergency event web information acquisition using crowd web sensors. *Wireless Personal Communications, 95*(3), 2393–2411.

Aakashjit Bhattacharya earned M.Tech in Computer Science Engineering from the Maulana Abul Kalam Azad University of Technology(Formerly known as West Bengal University of Technology), West Bengal, India. He has worked as a full-time Development Engineer at Calsoft Inc. , Pune, India. . He has International Rank 10th in the final round of International Informatics Olympiad 2010 conducted by Silver Zone. He was awarded certificate of Credit in Computer Skills from the University of New South Wales, Sydney, Australia. His area of interest includes the Internet of Things, Edge Computing, Cloud Computing, Mobile Crowd Sensing.

Debashis De earned his M.Tech from the University of Calcutta in 2002 and his Ph.D (Engineering) from Jadavpur University in 2005. He is the Professor and Director in the Department of Computer Science and Engineering of the West Bengal University of Technology, India, and Adjunct research fellow at the University of Western Australia, Australia. He is a senior member of the IEEE. Life Member of CSI and member of the International Union of Radio science. He worked as R&D engineer for Telektronics and programmer at Cognizant Technology Solutions. He was awarded the prestigious Boyscast Fellowship by the Department of Science and Technology, Government of India, to work at the Herriot-Watt University, Scotland, UK. He received the Endeavour Fellowship Award during 2008–2009 by DEST Australia to work at the University of Western Australia. He received the Young Scientist award both in 2005 at New Delhi and in 2011 at Istanbul, Turkey, from the International Union of Radio

Science, Head Quarter, Belgium. His research interests include mobile cloud computing, Green mobile networks. He has published in more than 300 peer-reviewed international journals and 100 International conference papers, six researches monographs and ten text books. His h index is 32. Total citation is 4964.