Study of fuzzy logic and particle swarm methods in map matching algorithm

Ajay Kr. Gupta¹ · Udai Shanker¹

Received: 22 October 2019 / Accepted: 5 March 2020 / Published online: 11 March 2020
© Springer Nature Switzerland AG 2020

Abstract

Navigation system is used in deploying the real-time position of vehicles on the map of route with the help of Global Positioning System (GPS). However, the imprecision is included in the map of route due to erroneous GPS signal (i.e., inclusion of errors related to receiver and/or propagation). Also, the current navigation system involves dense road network areas having higher probability of imprecise inputs. The map matching (MM) method requires capability of inherent tolerant to these imprecise inputs and is capable of suggesting certain vehicles more likely to run on one specific link than other links in navigation system. The fuzzy inference system (FIS)-based technique converting the qualitative terms into quantitative values is very effective to deal with human intervention and linguistic vagueness in MM algorithms. In case of highly unreliable positioning errors included in MM, the particle swarm optimization (PSO) is used as an option for continuously updating sensor information to find vehicle motion and respective route link on map. This has directed us to do the comparative study of both the above two methods with the other methods. Simulation results show that FIS- and PSO-based MM algorithm significantly outperforms the previous algorithms in terms of running time, accuracy and recall.

Keywords Context-aware systems · Location-based services · Internet of things · Pervasive computing · Ubiquitous computing · Algorithms · Map matching · Road network · Fuzzy logic

1 Introduction

Mobile devices are continuously upgrading features allowing usage of context data such as location-dependent data [1] in order to efficiently serve the client queries with valid answer from anywhere at any time. Route is an imperative travel property in most of the location-aware application. One can achieve information about route by regularly capturing Global Positioning System (GPS) [2] data via the cell phone connected to a server. It is necessary to manage the battery power resources efficiently if tracking for an entire day is required. Frequent visit position recalculations will yield higher exactness at the cost of significant energy. If one has to entertain unnecessary frequent information

transmission, then it will increase the phone bill cost and utilize additional network resources such as batteries energy as well. On the other hand, infrequent or rare positions recalculations will build battery life, but are unable to receive accurate result from traveling user's device real-time location [3] or travel route. A well practical dynamic location management algorithm with continuously tracking and GPS request rate-fixing ability of user location and user's route construction needs to be implemented. A sample of GPS Log and Trajectory for Client Movement is given in Fig. 1. The problem of low sampling rate GPS Trajectories-based Map Matching (MM) [4] can be stated as follows.

✉ Ajay Kr. Gupta, ajay25g@gmail.com; Udai Shanker, udaigkp@gmail.com | ¹Department of Computer Science and Engineering, M. M. University of Technology, Gorakhpur, India.

Fig. 1 Log and trajectory for client movement

	Latitude,	Longitude,	Time
p1:	59.9848	102.3319,	11:10
p2:	59.9841	102.3472,	11:13
.....			
p _n :	59.9858	102.3358,	11:29

Fig. 2 Illustration of road segments

Definition 1 GPS log: It is a point arrangement of $P_t = p_{t_1}, p_{t_2}, \dots, p_{t_m}$. Here, every point $p_{t_i} \in P_t$ contains $p_{t_i}.t$, $p_{t_i}.lat$, $p_{t_i}.lng$ as timestamp, latitude and longitude, respectively.

Definition 2 GPS trajectories: It is represented by T . It is a sequenced set of GPS focuses in which time interim between any two back-to-back GPS focuses not surpassing a settled limit known as threshold T , i.e., $T: p_{t_1} \rightarrow p_{t_2} \rightarrow \dots \rightarrow p_{t_m}$, where $T > p_{t_{i+k}}.k > 0$ for $(1 \leq i < m)$ and $p_{t_i} \in L$. $m = |T|$ is the count of sample points, and k is known as the interval of these sample points.

Definition 3 Segment of Road: It is represented by symbol r and is a directed line associated with id information $r.id$, initial point $r.start$, an ending point $r.end$, length value $r.l$, travel speed $r.v$ and intermediary point sets representing the road using a polygon. Figures 1 and 2 suggest actual segments of road in OpenStreet Map Search.

Definition 4 Road network: It can be represented as $G(V, R)$, where V is an arrangement of set of points where one road link is connected to others road link in the street portions and R can be viewed as an arrangement of set of directed lines or edges representing street sections, where each edge is associating with two vertices.

Definition 5 Path: Path P is an arrangement of associated street fragments that begins at vertex V_i and ends at V_j , i.e., $P: r_1 \rightarrow r_2 \rightarrow \dots \rightarrow r_n$, where $V_i = e_1.start$, $V_j = e_n.end$, $e_{m+1}.start = e_m.end$, $1 \leq m \leq n$.

The structure of paper is as follows. Review of the problem definition, motivation and key contributions linked to this research is presented in Sect. 2. Followed by this, related works are listed in Sect. 3. The ArcGIS model used in the suggested map matching is described in Sect. 4. The PSO and FIS-based MM algorithms are described in Sect. 5. In addition, an explanation of the comparative study by simulation result is given in Sect. 6. Finally, the paper's conclusion and future scopes of the research domain are given in Sect. 7.

2 Problem definition, motivation and key contribution

Assume a vehicle is moving along a finite system of streets, N ; here, one does not know. Instead one has a road network representation $G(V, E)$ of the system in ArcGIS map. For simplicity, it is assumed that there is a one-to-one correspondence between the streets in N and the arcs in $G(V, E)$. The $G(V, E)$ consists of piecewise linear arcs representing roads. Each arc, $A \in E$, can be described by finite sequence of points $(A_0, A_1, \dots, A_n)$. The points A_0 and A_n are called nodes. These are the endpoints of an arc and often points on which it is not possible to move to another arc. They correspond to dead ends or junctions in the street system. Additionally, the GPS device provides us an estimate of the vehicle's location within a certain intervals known as GPS trajectory T . The actual location of the vehicle at time t is denoted by P_t and the estimated location by P_e . The goal is to discover the road A from G which corresponds to T with its real path by matching actual location P_t to the estimated location P_e . The chain of discovered consumer positions is aligned on virtual map by means of producing a directed and annotated street map from GPS lines (or traces). The MM problem is illustrated in Fig. 3. It is a preprocessing step in any location-dependent information system (LDIS).

The need of map matching algorithm is important due to two reasons. First reason is due to the mistakes in the GPS device leading to wrong location of a vehicle. Second, due to transfer, storage and other costs, sampling rates are not all the time at high frequency, making it difficult to tell the route. Therefore, it is a vital operation for application

Fig. 3 Map matching problem

spending to track the data, data management for traffic analysis, frequent path finder and taxi pickup recommending system.

Frequent map matching steps yield more accurate result at the cost of unnecessary information transmission and additional network resources such as batteries energy, network bandwidth. On the other hand infrequent position recalculations will result in inaccurate result from traveling user's device real-time location. The sensor information is changing continuously in time and space according to the mobile client's motion. When map matching error is high due to low bandwidth or some other system errors, then the system should have ability to locate the mobile user on map without compromising accuracy of the map matching. Therefore, a well practical dynamic location management algorithm with continuously tracking and GPS request rate-fixing ability of user location and user's route construction needs to be implemented. To reduce error of map matching, sensor's information is processed into the various map matching algorithms, e.g., particle swarm optimization (PSO), fuzzy logic, Bayesian inference, Kalman filter, Dempster method or the Shafer's principle for improvement in the precision available at any time. The previous map matching policies did not considered the available inputs, e.g., heading, speed and vehicle's historical trajectory, etc., missing the relation between the road ties. In fact, they ignore the sources of error correlated with the road network and the orientation system. Based on the above shortcomings, the paper suggests advance map matching algorithm based on fuzzy logic which considers various input variables such as velocity direction, horizontal dilution of precision, position output perpendicular distance from candidate edge, heading change, heading error, GPS fix interval time, number of epochs. PSO has a limited number of parameters. The effect of the parameters on the solutions is small compared to other methods of optimization. It is less reliant on a number of

initial points than on other techniques of optimization. It can also tackle problems of the nonlinear, nonconvex, constant, discrete and integer form vector problems. Among various optimization techniques, we have first chosen PSO as a map matching optimization algorithm here due to its fastness and being more efficient in comparison with the other optimization algorithms.

The paper key contributions are summarized with the following points.

1. Integration of fuzzy inference and recommendation in MM algorithms to filter the errors of navigation system and get an accurate FIS-based navigation systems—various input variables used by fuzzy inference and recommendation in MM algorithms include velocity direction, horizontal dilution of precision, position output perpendicular distance from candidate edge, heading change, heading error, GPS fix interval time, number of epochs, etc.
2. Integrating PSO in MM algorithm of unreliable navigation systems—it has very high rate of positioning errors related to MM. For two datasets with extreme variation in direction and initial position PSO is used here to solve the local optimal problem to get better transformation outcomes. The system then uses part of the global map as the dataset of reference with overlapping points for corresponding map matching.
3. Involvement of false matching relationships pin the traditional MM method—here, PSO-based MM of road for running the vehicle is adopted which leads to reduction in false matching relationships. The corresponding precision of the PSO-based MM algorithm has arisen considerably. Moreover, fuzzy logic-based algorithm shows best map matching accuracy and most insensitive to the variation in sampling interval. This paper includes a comparative study of two proposed methods, i.e., FIS and PSO incorporation in MM

algorithms in terms of recall and accuracy to that of conventional MM algorithms.

4. Providing a scheme to analyze whether implementation of the system will meet the specification or not—to achieve this goal, a shortest path discovery implementation, i.e., A^* algorithm in the FIS, R-tree-based spatial indexing to save map information, included in the implementation of LBS. For implementation of the algorithm, the navigation system consists of 16-bit microcontroller 80C196KB, user interface system, route guidance system and a positioning system.
5. In the area where available bandwidth is very low or sometimes, completely gone. In this case, GPS request sampling rate fixing is very tough and possibilities of map matching error or more formally LBS map matching failure are very high. This paper is equipped with practical dynamic location management algorithm to support fault-tolerant approach with continuously tracking and GPS request rate-fixing ability of user location and user's route construction.
6. Analysis of effects of different parameters related to FIS and PSO on system performance. By using a simulation for the probabilistic path forecast model, we have generated a big quantity of historical information for parameter training. In FIS-based MM, a field test for 20 min validates the performance of the algorithm. We have conducted an assessment of performance (precision and effectiveness) of MM method from actual driving information, whose path was already known.

3 Related work

One can achieve information about route by regularly capturing GPS data from the cell phone through a server. Navigation system involves deploying real-time position on road network which is obtained from given sensor, e.g., GPS. Various parameters contribute some positioning errors to these navigation systems, e.g., masking of GPS signal, errors related to receiver, propagation and/or also satellite. An observed GPS position has measurement errors due to GPS devices limitation and sampling errors due to low sampling rate [5]. Therefore, it is needed to be preprocessed to align it on a given virtual map. The challenges of MM differ depending upon the sampling rate and GPS accuracy. In practice, low sampling rate GPS trajectories are easily available. These GPS trajectories have up to one point every 3–5 min. But, current MM schemes deal with high sampling rate GPS trajectories only. These GPS trajectories typically have up to one point every 5–30-s GPS data. If one employs current MM schemes to low sampling rate points, it will become less effective as increment of uncertainty in data. Numerous strategies for

sending cell phone location updates to a server have been examined in [6–9]. In polling survey, server pulls the coordinates (or any other location information) from the cell phone with certain interim. It is valuable for infrequent position refreshes, for instance, application showing the present location of gadgets on a map guide. This polling scheme is inefficient for real-time frameworks or where detailed route recording is necessary. This scheme has disadvantage that when interim (period) is set to be small; then, endless measure of pointless information can be sent to the uplink of the server. Again, if time interval set to be long, then framework will be more proficient; however, they do not address the issues of real-time applications. Zone-based map matching is another scheme in which the mobile node sends observed GPS point information when it goes into or leaves specific area. In distance-based strategies, mobile node sends GPS fixes only when mobile node has surpassed a separation limit or threshold. These schemes have disadvantage as well that it sends unnecessary location updates when mobile nodes follow straight line route. Another scheme to MM is dead reckoning that finds whether it needs to send the GPS fix to server solely based on the most recently sent location information or not. But, this scheme requires that the estimation function must be concurrently executed on the server and devices. If mobile device finds some deviation in its position from what is expected to be by executing estimation function based on most recent server information, then it will send the new location information to the server. This method is advantageous as it rarely transmits the location data, but it has one disadvantage as well because it requires the potentially costly estimation functions to be continuously executed on both the server and the mobile device which consumes more resources and unsuitable for real-time framework. In navigation, dead reckoning is the procedure of using so as to figure one's present position a formerly decided position, or settle, and propelling that position based upon known or evaluated speeds over slipped by time and course. The tall buildings block the GPS signals. It is the urban canyon problem. Dead reckoning is still necessary for uninterrupted service. With no view of the GPS satellites, the simplest thing the software can do to use last known position and last known velocity and extrapolate forward in time. This approximation starts to fail quickly as one's car speeds up or makes turns. Next best is to use the accelerometer, gyro and compass. The phone has these sensors. This gives the software information about car orientation and change of velocity. Fused with last known GPS position, this provides better dead reckoning, i.e., the extrapolated position remains accurate for longer time. What the car's navigation unit has that one's phone does not is the actual distance-traveled information and the wheel revolution count. This is a more

accurate distance measurement than can be provided by the phone's accelerometer. This is one reason why the car's navigation computer is better than phone's nave software in spotty GPS coverage situations.

To obtain extremely economical location-based services (LBS), an associative algorithm rule that mixes aspects of many update techniques and dynamically adjusts at run time is needed. In [10], critical point algorithm can be viewed as a hybrid algorithmic rule fulfilling this goal. An associative algorithmic rule is needed to manage the trade-off between GPS fix request rate and energy-supporting real-time application needs. The various approaches [5] for MM of GPS observation can be broadly categorized into the following classes.

1. *Local/incremental method* It is used to find local matches of geometries. To do so, it evaluates the candidate edges by summing the scores of corresponding individual similarity measures. It accounts for two types of score to evaluate candidate edge, namely distance similarity score and orientation similarity score. If one finds adjacent edges for each sample, then he can use adaptive clipping method as in [11] which it employs Dijkstra algorithm to find shortest path on local free space graph. The algorithm takes $O(mn \log m)$ time, where n and m are number of GPS points and number of edges in the road network, respectively. In [12], author proposed a segment-based matching approach which gives confidence values to different sampling points. The matching is done in order of high confidence sampling points to the low confidence sampling points. While matching a new position, this method considers a small portion of the trajectory only which are near to the position. The performance of this method is very well if sampling frequency is high, e.g., 1–10 s. But if sampling rate reduced that is if the interval for sampling is increased, then it will show the problem of arc-skipping resulting in inaccuracy.
2. *Global method* This method is used to match the entire trajectories with the given road network. To achieve this goal, Frechet distance is used which records the continuity of curves. In [13], algorithm searches all the critical values through different parameters. The solution of decision problem is given by exploring a path from the lower left corner to the upper right corner in free space. The algorithm time complexity is $(mn \log^2 mn)$, where n and m are the number of nodes in the road network and number of edges, respectively. The effect of outliers is reduced using average Frechet distance described in the later work [13]. The aim of global methods is to minimize Frechet distance between the trajectory and the matched road segments.
3. *Statistical method* Statistical-based method for MM is gaining attention over decades. In [14], hidden Markov model was incorporated with Bayesian classifier for map matching of GPS fixes. In [15], cubic spline interpolation and the extended Kalman filter are combined with spatiotemporal matching methods to improve the system performance. Shortest path computations are evolved in the spatiotemporal matching policies. Dijkstra's, search decomposition, A* algorithm [16], bidirectional search [17] and hierarchical search [18] are well-known shortest path computation algorithms. For faster computation, the shortest path computation performs preprocessing with policies such as ALT [19]. Euclidean distance method and ALT policy have the tighter lower bound than that of A* algorithm. Moreover, various other approaches are also came into existence such as in [20]. The author proposed method for intersection geometries extraction. In [20], centerlines lanes count is considered, and in [21], road type and speed limit are considered to MM. The trajectories may involve irregularities related to the acceleration, the speed traveled, direction changes and abrupt expected distance between points. So, a filtering step is necessary before the map generation process, in which traces are tested for any irregularities. One can replace points from the GPS trace by an interpolated point if given point does not satisfy the given criteria. The k-means algorithm, trace merging and kernel density estimation (KDE) are the classes of algorithms used for map generation. K-means algorithm is performed to deduce the set of GPS points by cluster centroids [22, 23], which are linked together to formulate road geometry. KDE approach estimates kernel density based on the GPS points and edges. This estimation of kernel density helps in geometry extraction for roads. Table 1 shows the summary of MM-associated technical papers.

To filter the navigation system errors and get an accurate FIS-based navigation systems, it needs integration of fuzzy inference and recommendation in MM algorithms. Various input variables such as velocity direction, horizontal dilution of precision, position output perpendicular distance from candidate edge, heading change, heading error, GPS fix interval time, number of epochs could be used in fuzzy inference and recommendation in MM algorithms. Fuzzy logic-based algorithm used in various LBS problems shows better accuracy and is the most insensitive to the variation in sampling interval.

In the past, Olivas and Valdez [30] illustrated a summary of the various methods used for type-2 fuzzy particle swarm, bee colony and bat algorithms in controller intervals. Castillo et al. [31] also proposed a new type-2 fuzzy model

Table 1 Summary of map matching-associated technical papers

Authors	Feature	Performance measurement method	Real data	Dataset	Technique
Schroedl et al. [20]	Intersection geometry	Lane error versus amount of data	Generated from GPS traces	250 traces synthetically perturbed	k-means
Niehoefer et al. [21]	Edge classification	Amount of data versus relative error, ground truth versus eyeball	Google maps	7 traces	Trace merge
Edelkamp and Schroedl [24]	Lane finding	Lane error versus amount of data	Generated from GPS traces	250 traces synthetically perturbed	k-means
Davies et al. [25]	Not applicable	Ground truth versus eyeball	UK ordnance survey	Campus GPS points	KDE
Worrall and Nebot [26]	Compact representation	Raw versus compact	None	Vehicle traces by mining	k-means
Guo et al. [27]	Not applicable	Amount of data versus relative error	None	Synthetic GPS traces	k-means
Chen and Cheng [28]	Not applicable	Ground truth versus eyeball	Google Earth	Automobiles traces	KDE
Cao and Krumm [29]	GPS trace classification	Bing maps versus route and ground truth versus eyeball	Bing maps	Campus shuttles GPS points	Trace merge

architecture methodology. Valdez et al. [32] provided a comprehensive review of some optimization techniques, in which fuzzy logic takes care of parameter adaptation. In the field of intelligent control, Castillo et al. [33] identified the use of hierarchical genetic algorithms to automate fuzzy logic.

Moreover, integration of PSO in unreliable navigation systems can be done where there is very high rate positioning errors related to MM exist. Traditional MM method involves many false matching relationships. PSO-based MM of road for running the vehicle can be adopted which leads to reduction in false matching relationships. Melin et al. [34] developed a new method for changing the social and cognitive parameters of PSO (c_1 and c_2) with the aid of fuzzy logic. Valdez et al. [35] proposed a new optimization strategy that blends GA and PSO, utilizing fuzzy logic to integrate GA and PSO effects. In [36], Olivas and Valdez integrated fuzzy with PSO by suggesting interval type-2 fuzzy logic-based dynamic parameter adaptation in particle swarm optimization. For two datasets with extreme variation in direction and initial position PSO can be used to solve the local optimal problem to get better transformation outcomes. The system then can use part of the global map as dataset of reference with overlapping points for corresponding map matching.

4 ArcGIS

ArcGIS, as the name goes, is a geographic information system tool developed by ESRI back in 1999 and written in computer language C++. Current releases of ArcGIS are versions 10.4.1 and 10.4.2. It is compatible with Android, iOS and Windows. It has varied uses starting with making/digitizing maps, compiling data, analyzing mapped data and most importantly managing all these information on a single database. Basically, it allows one to capture, store, analyze, manipulate and manage huge amount of data at one platform. The best part is that one can export the data into various other modeling of software to further explore possibilities and to play around one's datasets. ArcGIS has millions of geoprocessing tools. ArcGIS is first starting platform in GIS Industry. All GIS and online application depend on this software. In the coming era, one can think that everyone will be dependent on GIS systems. For example in India, one can say his cities are being converted to Smart City, but converting a simple city to smart city, one needs a big GIS System. Apart from this, one needs to learn SQL PostGIS, R, QGIS, etc. These are the free application with great GIS features. An ArcGIS map facilitates us to

create customized road networks, base maps, choose travel modes, asset locator, etc., with mobile system and navigator.

5 MM algorithms

MM is the operation of matching recorded geographic coordinates to a logical model of the real road networks making task easier of urban computing such as traffic systems. More than 10 algorithms have been tested in decades for this problem. But some of them are tested only on specific database, and that makes it hard to participate for the decision which algorithm to apply for different situations. To address this problem, a wide range of existing MM algorithms survey is given in [37, 38]. Based on the main techniques, one will put them in different categories and compare them through experiments of real-world and artificial dataset with different characteristics. One can also report vast finding obtained from the experiment and provides insights of the strength and weakness of existing MM algorithm which will help participant to apply the appropriate algorithm. In MM integration of digital map data with its GPS and dead reckoning (DR), i.e., odometer reading (Δd), errors associated with it (ϵ_d), gyro-rate reading ($\Delta\theta$), errors associated with it (ϵ_θ) are responsible for required navigation performance. EKF algorithm is used by Quddus and colleagues [39] for this integration of GPS and DR. Finding the real location of any moving vehicle on a road is key requirement in any MM algorithm. GPS/DR outputs are vehicle speed (v), heading (θ), northing (N), easting (E) and the error variances, respectively (i.e., σ_v , σ_h ,

σ_e , σ_n). MM gives the output as correct link, the directional position solution (easting and northing of respective links) and uncertainty associated with it (σ_{mm}).

In map matching process, road hyperlinks having its confidence value more than the defined minimum confidence threshold (min_thresh) are considered as candidate link. If the candidate area does not incorporate any link, system takes the assumption that car is off-route of recognized avenue link. The general MM framework is depicted in Fig. 4. In a scenario, where the candidate area includes only one segment, the selection technique would be simple. If multiple links are presented around position fixes, then FIS, neural network or PSO can be used to perceive the right hyperlink among the candidate links. Any MM algorithm [40–42] consists of two sections. First section has the steps for correct link identification, and the second one has location determination steps on the selected link. The initial MM process steps involve position fix to the link perpendicular distance, link bearing, vehicle direction, GPS receiver first fix time, search space derived from the error variances, etc. Identification of the first link is made by few first link position fixes.

5.1 Using fuzzy inference system

Computing can be categorized as hard computing as well as soft computing. Hard computing is related to numerical analysis, crisp systems and binary logic, while soft computing is related to probabilistic reasoning, fuzzy logic and neural networks. The current trend follows the use of fuzzy logic in mixture with genetic algorithms [43] and neurocomputing to instruct fuzzy

Fig. 4 Map matching framework

membership functions [44]. Adaptive Neuro-Fuzzy Inference System (ANFIS) in MATLAB 6.5 can be used to optimize fuzzy membership functions. Membership functions determination is done by least squares or backpropagation method. Antecedent (input) to the consequent (output) connection in FIS is made by the If-then rules. The weights in FIS are defined based on their criticality. The basic fuzzy logic consists of some important steps which are shown in Fig. 5.

The output of Mamdani-type FIS is fuzzy set, while the output of Sugeno-type FIS is either weighted expression or a constant.

e.g., Mamdani: If P is $Y1$ and Q is $Y2$, then R is $Y3$. (P , Q , R are fuzzy sets.)

Sugeno: If P is $Y1$ and Q is $Y2$, then $R = aY1 + bY2 + c$ (weighted expression).

In order to implement fuzzy logic, one can use Mamdani fuzzy inference system (FIS). A FIS initially fuzzifies the value of the parameter which results the value to be in range of $[0, 1]$. For this, a membership function is defined. Various kinds of membership function can be used. But for the simplicity of the model, one can use the simplest type of membership function known as the triangular membership function (TMF). It fuzzifies $v_i \in V$ for each $p_i \in P$. Further, the output generated by the inference system should also be defined by a variable in the form of fuzzy value means its range should fall between 0 and 1 (inclusive). In our case, a triangular membership function for this output variable is used and the TMF can be defined mathematically as given in the following equation.

$$\text{Membership function } (\mu_f) = \begin{cases} 0, & x \leq a \\ \frac{b-x}{c-b}, & a \leq x \leq c \\ \frac{b-x}{c-b}, & c \leq x \leq b \\ 0, & b \leq x \end{cases}$$

Fig. 6 Graphical representation of triangular membership function

The representation of TMF is given in Fig. 6.

Another form of this membership function can be used according to the nature and behavior of the introduced parameters in which the equation simply maps the value between 0 and 1, $\mu(x) \rightarrow (0, 1)$, or one can say $\mu(x)$ maps the value $v_i \in V$ in the range of $[0, 1]$. This is called as fuzzification of parameters. Now, two boundary parameters a and b are also needed to be optimized and normalized for every boundary condition defined by the membership function of each parameter. So, if a parameter has n boundary condition, then one has to optimize $2n$ boundary parameters. In a more general term, if a parameter $p_i \in P$ has a b_i number of boundaries, then the output generated by FIS is in fuzzy form which defines a limit on minimum and maximum values of it. So, if the range of T -value is $T_{\max}$ to $T_{\min}$, then,

$$\text{Number of parameters to optimize } \left(\sum_{i=1}^n b_i \right) + 2$$

The likelihood associated with a link is the crisp output which is obtained by weighted average method. The min() function is used to obtain degree of applicability (ω_i) for

Fig. 5 Fuzzy inference system (FIS)

each fuzzy rule. In order to calculate the likelihood value, the following expression is derived by considering certain range of membership function from a to b .

$$P_{f1} = \frac{v_1 - a}{b - a} \quad P_{f2} = \frac{v_2 - a}{b - a} \dots \dots P_{fn} = \frac{v_n - a}{b - a}$$

$$\text{Range } (P_{f1}, P_{f2}, P_{f3}) \rightarrow [0, 1]$$

$$P_{\min} = \text{Min}[P_{f1}, P_{f2}, P_{f3}]$$

Likelihood - value

$$= 10 \times \left(1 - \frac{\int_a^b [P_{\min} \times (P_{\min} \times (b - a) + a)] dx}{\int_a^b [P_{\min}] dx} \right)$$

P_{fn} represents the fuzzified value of any parameter P_n , and $P_{\min}$ is the minimum value among all the parameter after fuzzification.

The various input variables which are used in fuzzy logic map matching algorithm are listed below.

- Moving object velocity v (m/s),
- Direction of velocity with respect to links
- Direction of movement with respect to links,
- HDOP (horizontal dilution of precision)
- Position output perpendicular distance from candidate edge PD (m),
- Heading change (obtained from the gyro) (H)
- Heading error, HE (degree)
- GPS fix interval time (obtained from GPS)
- Number of epochs

The following types of fuzzy rule sets are involved in the fuzzy logic map matching algorithm.

- Fuzzy rule set for initial link and vehicle position on the link.
- Fuzzy rule set for the next map matching link followed by current link.
- Fuzzy rule set for junction of the links.

Table 2 Set of rules used in the rule base

Perpendicular distance	HDOP	Heading change (obtained from the Gyro) (H)	L1
L	L	L	L
L	L	M	L
L	L	H	M
L	M	L	L
L	M	M	M
L	M	H	M
L	H	L	L
L	H	M	L
L	H	H	H
M	L	L	L
M	L	M	M
M	L	H	M
M	M	H	M
M	M	M	M
M	M	H	H
M	H	L	L
M	H	M	H
M	H	H	H
H	L	L	L
H	L	M	L
H	L	H	M
H	M	L	M
H	M	M	H
H	M	H	H
H	H	L	H
H	H	M	H
H	H	H	H

While designing the rule base, dependency of likelihood value on V_i is considered which is predicted by simulating it individually. For each parameter, limited numbers of boundaries like HIGH, MEDIUM and LOW are defined according to fuzzified value of that parameter as depicted

Fig. 7 Representation of boundary parameter of P

in Fig. 7. The selection of the boundaries is the matter of choice, and one can choose as many as it is required for implementing the system. Suppose $S = \{H, M, L\}$, where $H \leftarrow \text{High}$, $M \leftarrow \text{Medium}$, $L \leftarrow \text{Low}$, here $S_1[i]$ means i th element of S for the first parameter. In the same way, $S_n[i]$ denotes i th element of S for n th (last) parameter and $i \leftarrow 1, 2, 3$. Table 2 depicts all set of rules that is used in the rule base. The general form of rules in the rule base is given below, in which S_1 , S_2 and S_3 denote set of boundary for the first, second and third parameters, respectively. The FIS MM single output is the likelihood of position fix to the link. In Sugeno, fuzzy map matching constants are taken for low as 10, average as 50 and high as 100.

The initial map position (IMP) fix is denoted by P as given in Fig. 8. FIS evaluates link 1 to be the real link for position fix P . Then in subsequent step the FIS, finding of the link 1 will be the real link for position fix Q and R . Therefore, it can be observed that the position fix S lies on the link 1.

5.2 MM based on PSO

PSO-based approach and basic MM steps are combined to propose a hybrid model in this paper. The PSO-based approach is used to optimize the dataset to find best link in link space. The idea of PSO was taken from social psychology and computer graphics. When a group of birds go for the search of food, they continuously communicate to each other. Due to this communication process, they share the information to each other which one is the best path to follow and which one is the best location to be safe and to get food. Due to this social behavioral nature, if one finds out a desirable path to go, all the rest birds follow the path quickly. It is used to achieve the optimally (globally) best solution among the given number of solutions. PSO is a branch of an evolutionary computation and well known in optimization techniques. PSO is much similar to genetic algorithms. But unlike GA, it does not possess any evolution operators like mutation and crossover. It contains a lesser

number of parameters so that the adjustment is easy here. It is used to solve both continuous and discrete optimization problems. In PSO, it maintains a number of potential solutions which are termed as the particles and the whole population of solutions called swarm optimization [45–47]. It was originated by British Military in 1940s. It is mathematical technique and used in finding the maxima or minima of certain function for a feasible region. It may be linear or nonlinear. As birds fly in 3D space, they search for food or location to survive; algorithm uses the particle's (bird's) information moving in n -dimensional space. Solutions are extracted from particle as it is treated as a solution in an N -dimensional space. The variables used in PSO are number of iterations, number of particles or swarm size, velocity and acceleration coefficients. PSO algorithm is easy to implement, and its performance measurement is better in comparison with other optimization algorithms. The parameters of PSO [48] are discussed below.

1. **Swarm size** The count of solutions (particle) within the swarm is termed as the swarm size. For ' n ' particles, the swarm size will be n . With the increase in swarm size, the area of search space also increases, and further, the running time and computational complexity also overburden due to this.
2. **Iteration numbers** The count of iteration to achieve the goal depends upon the problem. Lesser iterations count may cause the end of the searching prematurely. Iterations count in higher amount makes process time-consuming.
3. **Velocity components** This is made up from the 3 of its components as inertia (previous movement direction data), cognitive component (i th particle past performances) and social component (i th particle performances related to a group of neighbor particle).
4. **Coefficients of acceleration** The c_1 and c_2 are the two acceleration coefficients, where c_1 and c_2 measure the degree to which a particle confidence is in itself and its neighbors, respectively.

Fig. 8 Case of IMP

5. **Fitness function** Fitness function is used to conclude how much fit the particle/solution is, i.e., it evaluates the quality of the candidate solution at its best.

5.2.1 Proposed fitness function for PSO algorithm

First, the collection of pairs of candidate sequences $\{P_1, P_2, \dots, P_i\}$ from neighborhood matching pairs (NMP) is abstracted as particles for traffic MM. These i number of particles for each links are abstracted from the neighborhood matching links of current location fix of moving vehicle. $P_i \in \text{NMP}$, a particle can be defined as below.

$$\{(A_1, B_1), (A_2, B_2) \dots \dots (A_{n-1}, B_{n-1}), (A_n, B_n)\}$$

$A_i \in \text{road source network } (R_A)$ and is a point of the map trajectory, and B_i is the corresponding link in the road network. The trajectory point and road link pair are decided in the start of the evaluation of the road source network. In the highway network, single-node matching pairs correspond to 1-D of M -D particle space. M matching pairs represent a particular similarity comparison solution that corresponds to the particular particle place in M -dimensional space. The measurement of resemblance between nodes demonstrates that it is possible to express the similarity measurement model of road which is used to calculate the fitness of each neighborhood link from given trajectory. Here, higher is the fitness function value of a record, higher will be the fitness of that record and the best will be the instance among the records. Now, the proposed fitness function f for i th instance, i.e., particle pair similarity (PPS_{*i*}), is expressed as given below.

$$f = \text{PPS}_i(A_j, B_k) = \varepsilon_1 \times \text{TOP}_{\text{Sim}}(A_j, B_k) + \varepsilon_2 \times \text{GEO}_{\text{Sim}}(A_j, B_k) + \varepsilon_3 \times \text{SEM}_{\text{Sim}}(A_j, B_k)$$

where TOP_{Sim} is the degree of resemblance between road entities of the topological features. GEO_{Sim} is the resemblance between street entities of geometric characteristics. SEM_{Sim} is the similarity between street entities of semantic characteristics. ε_1 , ε_2 and ε_3 are assigned weight-ages for respective contributions.

The PSO-based MM algorithm flowchart is given in Fig. 9. PSO-based MM algorithm uses (subset of) road network base dataset (numeric) retrieved from UCI repository. Here, a subset of whole dataset including feature selection in preprocessing (to reduce number of attributes) is used. The fitness function and PSO parameters are being updated in each iteration; also, the classification accuracy is checked over WEKA. The default values of parameters used in PSO are given below.

Fig. 9 Flowchart of PSO-based map matching algorithm

$$\begin{array}{ll} \text{Position } x_i^t = 0 & w = 0.729 \\ \text{Velocity } v_i^t = 0 & \text{rand}_1 = 0.4 \\ C_1 = C_2 = 0 & \text{rand}_2 = 0.6 \end{array}$$

For each instance (record) set, Pbest is current fitness value and Gbest is best (highest) fitness value among all the particles (instance). Update position and velocity of each particle (instance) are given by the equation below.

Here $c_1 = c_2 = 2$, $w_i = 0.729$, $\text{rand}_1 = 0.4$, $\text{rand}_2 = 0.6$

Position $x_{id}^{\text{old}} = 0$ Velocity $v_{id}^{\text{old}} = 0$

Particle (instance) velocity = $v_{id}^{\text{new}} = w_i \cdot v_{id}^{\text{old}} + c_1$
 $\cdot \text{rand}_1 \cdot (p_{id} - x_{id}) + c_2 \cdot \text{rand}_2 \cdot (p_{gd} - x_{id})$

Particle (instance) position = $x_{id}^{\text{new}} = x_{id}^{\text{old}} + v_{id}^{\text{new}}$

pbest = fitness value till current = x_{id}^{new}

The value after calculation is further checked for the gbest (i.e., the record among all, which have highest fitness value of particles). At this step, one can get matching pair similarity in one dimension. Although in real-time M dimension particle space, many particle matching pair's changes happen. The PSO evolves several iterations to update particle fitness value synchronously according to each dimension position and velocity changes. The finalization in the entire particle position will be acknowledged at long last after final cycle, and the likeness of the final street can be portrayed toward the end when stop criterion is met.

6 Simulation result of the FIS- and PSO-based map matching algorithm with example dataset

All algorithms have been implemented in Microsoft Visual Studio 2005 (C++) with ArcGIS on a computer with a 3.50 GHz Intel Core i7-4770 K CPU and 32 GB RAM. Map information is saved with R-tree-based spatial indexing [49]. To discover the shortest path in the FIS and PSO-based MM method, A* algorithm [16] was used. We have conducted two evaluations in this work. The first one was an assessment of performance (precision and effectiveness) of MM method from actual driving information, whose path was already known. By using simulation information, the second one assesses the enhancement in precision obtained through parameter training. By using a simulation for the probabilistic path forecast model, we have generated a large quantity of historical information for parameter training. In FIS-based MM, a field test for 20 min validates the performance of the algorithm. Speaking details of the hardware used in the implementation of the algorithm, the navigation system

consists of 16-bit microcontroller 80C196KB, user interface system, route guidance system and a positioning system. The range of moving vehicle speed is from 10 to 50 km/h. The navigation system is L1-only coarse acquisition code capability, five parallel-channels, single-board, micro-tracer low power. The navigation system consists of a serial port RS232 for communication of mapped position from navigation system to computer and another sensor handling serial port for the GPS receiver. Table 3 shows the amount of nodes and the amount of road network data.

Table 4 represents confusion matrix for given PSO- and FIS-based map matching method.

In this depiction of matrix, P and S represent the amount that the given method properly results the accurate match and the number that the method properly denies, respectively. The mismatches Q and R are the amount that the given method wrongly results as lawful match into accurate mismatch and lawful mismatch into accurate match. The scientists give some of the most popular metric schemes to measure the efficiency of algorithms such as precision, recall and accuracy. These are used in this article to assess the matching outcomes. The formula is as follows.

$$\text{Positive precision} = \frac{P}{Q + R}$$

$$\text{Negative precision} = \frac{S}{R + S}$$

$$\text{Positive recall} = \frac{P}{P + R}$$

$$\text{Negative recall} = \frac{S}{Q + S}$$

$$\text{Accuracy} = \frac{P + Q}{P + Q + R + S}$$

Performance in terms of both precision and effectiveness was assessed. Recall is the proportion of correctly identified relevant pages, while precision is the proportion of the correct predicted relevant pages. The comparison in accuracy and recall rate between the traditional method and PSO-based MM algorithm is depicted in Fig. 10. Traditional MM method involves many false matching relationships. Here, PSO-based MM of road for running the vehicle is adopted which leads to reduction in false matching relationships. The corresponding precision of the PSO-based

Table 3 Nodes and road network data

Area	Use	Node count	Road count
Delhi	Source data	60	72
	Target data	54	64

Table 4 Confusion matrix in map matching

		Actual	
		Match	Mismatch
Map matching processed	Accurate matches	P	Q
	Accurate mismatch	R	S

Fig. 10 Comparison of traditional method and PSO-based MM algorithm **Table 5** Comparison of traditional MM and PSO-based MM algorithm

Method	Accuracy	Precision	Recall	F-measure
Traditional map matching	56	51	60	55
PSO based map matching	83	82	84	83

MM algorithm is increased from 51 to 82%. Further, the recall rate of the suggested algorithm has risen considerably from 60 to 84%. Experiments show that PSO significantly outperformed in accuracy and recall to those of previous algorithms due to the parallelism of its own algorithm as given in Table 5. *F*-measure (*F1*) is a function of precision, and recall can be given by the following equation.

Fig. 11 Evaluating running time

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Running time in this work was the point assignment, which is for online MM techniques and is the most significant precision metric. Many scientists suggested efficiency-enhancing methods. Figure 11 displays the running of the different sampling interval techniques. The horizontal axis reflects the number of trajectories used in sampling, and running time is represented by the vertical axis. The runtime performance comparison of all four algorithms is given in Table 6.

We have compared the 3 algorithms, namely GridST [50], statistic [51] and ST matching [5], with proposed FIS-based MM algorithm. We have used Road network of Delhi, India, having lakes of vertices and edges on it. We implemented synthetic trajectory dataset for the simulator by randomly selecting starting and endpoint with

consideration of certain fixed range of vehicle speed. The runtime performance of all four algorithms shows similar trends, i.e., it is proportional to number of trajectories in navigation system. ST matching and GridST algorithms involve matrix calculation which is time-consuming. Therefore, they have higher runtime and less efficiency than fuzzy logic algorithms and the statistic.

Figure 12 displays the effectiveness of different sampling intervals of the techniques. The horizontal axis reflects the sampling interval, and accuracy is represented by the vertical axis. The accuracy performance comparison of all four algorithms is given in Table 7.

The four tested algorithms exhibit similar matching accuracy trends, i.e., it is proportional to length of sampling interval in navigation system. From graph, we conclude that the technique of matching the FIS map is more accurate than earlier techniques. Due to similar basic logic,

Table 6 Effects of no. of trajectories on running time of traditional and fuzzy logic-based MM algorithm

No. of trajectories (in thousands)	MM methods/running time (in seconds)			
	ST matching	GridST	Fuzzy logic based	Statistic
1	50	40	25	22
2	120	110	55	50
3	150	130	75	70
4	180	155	90	80

Table 7 Effects of sampling interval on accuracy of traditional and fuzzy logic-based MM algorithm

Sampling interval (in seconds)	MM methods/accuracy (in %)			
	GridST	Statistics	Fuzzy logic based	ST matching
20	91	92	94	90
40	91.5	92.5	92.5	92
60	90	89	91	91
80	88	87	90	89
100	86	85	89	87

Fig. 12 Evaluating impact by sampling rate on accuracy

Table 8 Effect of number of A* MM queries, speed and trajectory length (in km) on accuracy and running time on one-day dataset containing 13,671 trajectories and 155,068 points

A* queries	Speed (points/s.)	Trajectory distance (in km)	Match percentage (in percentage)	Running time (in seconds)
821,444	1358	2	98.86	112.675
346,150.5	2639.8	3	98.86	57.975
159,972.5	4428.8	4	98.86	34.575
291,237	3687.1	2	98.83	41.575
25,174.5	16,919.8	3	98.83	9.055
6728.5	22,031.1	4	98.83	6.955

Table 9 Effect of speed and trajectory length (in km) on accuracy, running time and memory tested on the 15-day dataset containing 322,347 trajectories and 3,402,360 points

Speed (points/s)	Trajectory distance (in km)	Match percentage (in percentage)	Running time (in seconds)	Memory (in megabyte)
1636.5	2	99.26	1037.53	73.2
7138	3	99.26	237.86	100.6
10,777	4	99.26	157.55	133.1
10,777	2	87.38	67.77	70.9
23,465	3	96.87	70.24	97.8
22,736	4	98.34	73.6	130.8

GridST and ST matching algorithms present similar matching accuracy. Statistics MM algorithm has the worst matching accuracy, while fuzzy logic-based algorithm shows best map matching accuracy [52] and most insensitive to the variation in sampling interval [53].

Effect of number of A* map matching queries, speed and trajectory length (in km) on accuracy, running time and memory tested on one-day dataset containing 13,671 trajectories and 155,068 points is given in Tables 8 and 9.

7 Conclusions

The MM algorithm involves correlating sampling point to best suitable location on real map to find actual route traversed by the vehicle. Many MM algorithms have been introduced in previous decades. This article provides a comparison of online MM techniques using FIS and PSO with likelihood or similarity measurement. Experiments show that PSO significantly outperformed in accuracy and recall to those of previous algorithm due to the parallelism of its own algorithm. Further, the fuzzy logic-based MM algorithm achieves a significant performance improvement in terms of running time and accuracy as compared to previous cache replacement policy such as GridST, statistics and ST matching for LBS.

The avenue of the further research is to automate heterogeneous data sources cleaning processes such as emergency vehicles, public transportation vehicles, taxis and cell phones device in MM navigation system that will overcome the challenges faced by MM application in LBS

and further focus on user privacy issues. As future work, one can use large-scale, real-world datasets in fuzzy logic and PSO-based MM that needs enormous data cleaning effort. Moreover, multi-layer perceptron, Markov predictor, decision tree classifier and sequential pattern mining could be integrated to design an assistive technology (recommender system) in healthcare for helping users with intellectual, physical or sensory-diminished capabilities persons by reducing moving object navigation errors in LBS. Implementation of recommender system should integrate a predictive computing feature to the system, so that it would be capable of predicting a user error and alerting him in advance.

Compliance with ethical standards

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Gupta AK, Shanker U (2018) Location dependent information system's queries for mobile environment. In: 23rd international conference on database systems for advanced applications (DASFAA). International Workshops, Gold Coast, pp 1–9
2. Kouame K, Mcheick H (2019) Architectural QoS pattern to guarantee the expected quality of services at runtime for context-aware adaptation application. SN Appl Sci 1:405
3. Kumar A, Misra M, Sarje AK (2005) Strategies for cache invalidation of location dependent data in mobile environment. In:

- Proceedings of the 2005 international conference on parallel distributed processing technology applications (Pdpta'05), LDD, vol 1–3. Las Vegas, Nevada, USA, pp 38–44
4. Rohani M, Gingras D, Gruyer D (2016) A novel approach for improved vehicular positioning using cooperative map matching and dynamic base station DGPS concept. *IEEE Trans Intell Transp Syst* 17(1):230–239
 5. Lou Y, Zhang C, Zheng Y, Xie X, Wang W, Huang Y (2009) Map-matching for low-sampling-rate GPS trajectories. In: *Proceedings of 17th ACM SIGSPATIAL international symposium on advances in geographic information systems (ACMGIS 2009)*, 4–6 November 2009, Seattle, Washington, USA, pp 352–361
 6. Gupta AK, Shanker U (2017) SPMC-PRRP: a predicted region based cache replacement policy. In: *International conference on data and information sciences (ICDIS-2017)*, MP, India, pp 313–326
 7. Gupta AK, Shanker U (2017) Modified predicted region based cache replacement policy for location dependent data in mobile environment. In: *Sixth international conference on smart computing & communications. NIT Kurukshetra, India*, pp 917–924
 8. Gupta AK, Shanker U (2018) CELPB: a cache invalidation policy for location dependent data in mobile environment. In: *Proceeding of 22nd international database engineering & applications symposium (IDEAS 2018)*, Villa San Giovanni, Italy, pp 302–306
 9. Gupta AK, Shanker U (2017) SPMC-CRP: a cache replacement policy for location dependent data in mobile environment. In: *Sixth international conference on smart computing & communications. NIT Kurukshetra, India*, pp 632–639
 10. Barbeau S, Labrador M, Perez A, Winters P, Georggi N, Aguilar D, Pérez R. (2008) Dynamic management of real-time location data on GPS-enabled mobile phones. In: *2008 2nd international conference on Ubiquitous computing, systems, services and technologies (UBICOMM'08)*, pp 343–348
 11. Wenk C, Salas R, Pfoser D (2006) Addressing the need for map-matching speed: Localizing global curve-matching algorithms. In: *18th International conference on scientific and statistical database management (SSDBM'06)*, pp 379–388
 12. Chawathe SS (2007) Segment-based map matching. In: *2007 IEEE intelligent vehicles symposium*, pp 1190–1197
 13. Brakatsoulas S, Pfoser D, Salas R, Wenk C (2005) On map-matching vehicle tracking data. In: *Proceedings of the 31st international conference on very large data bases (VLDB'05)*. VLDB Endowment, pp 853–864
 14. Pink O, Hummel B (2008) A statistical approach to map matching using road network geometry, topology and vehicular motion constraints. In: *2008 11th International IEEE conference on intelligent transportation systems*, pp 862–867
 15. Hummel K, Tischler B (2005) Robust, GPS only map matching: exploiting vehicle position history, driving restriction information and road network topology in a statistical framework. In: Billen R (ed) *Proceedings of the GIS research UK 13th annual conference—GISRUK 2005*, 6–8 April 2005, no. 3, pp 68–77
 16. Hart PE, Nilsson NJ, Raphael B (1968) A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans Syst Sci Cybern* 4(2):100–107
 17. Nicholson TAJ (1966) Finding the shortest route between the two points in a network. *Comput J* 9:275–280
 18. Jing N, Huang Y-W, Rundensteiner EA (1996) Hierarchical optimization of optimal path finding for transportation applications. In: *Proceedings of the ACM conference on information and knowledge management*, pp 261–268
 19. Goldberg AV, Harrelson C (2005) Computing the shortest path: a search meets graph theory. In: *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms (SODA '05)*. Society for industrial and applied mathematics, USA, pp 156–165
 20. Schroedl S, Wagstaff K, Rogers S, Langley P, Wilson C (2004) Mining GPS traces for map refinement. *Data Min Knowl Disc* 9(1):59–87
 21. Niehofer B, Burda R, Wietfeld C, Bauer F, Lueert O (2009) Gps community map generation for enhanced routing methods based on trace-collection by mobile phones. In: *Proceedings of the 2009 first international conference on advances in satellite and space communications (SPACOMM'09)*. IEEE Computer Society, Washington, DC, pp 156–161
 22. Gupta AK, Shanker U (2020) CELP-IR: caching efficiency with location prediction based invalidation and replacement policy. *Int J Sens Wirel Commun Control*. <https://doi.org/10.2174/2210327910666200206122007>
 23. Gupta AK, Prakash S (2018) Secure communication in cluster-based ad hoc networks: a review. In: Lobiyal DK, Mansotra V, Singh U (eds) *Next-generation networks*, vol 638. AISC. Springer, Singapore, pp 537–545
 24. Edelkamp S, Schrödl S (2003) *Computer science in perspective*. Springer, New York, Inc., New York, Ch. route planning and map inference with global positioning traces, pp 128–151
 25. Davies JJ, Beresford AR, Hopper A (2006) Scalable, distributed, real-time map generation. *IEEE Pervasive Comput* 5(4):47–54
 26. Worrall S, Nebot E (2007) Automated process for generating digitised maps through GPS data compression. In: *Proceedings of the 2007 Australasian conference on robotics & automation*, Brisbane, Australia
 27. Guo T, Iwamura K, Koga M (2007) Towards high accuracy road maps generation from massive gps traces data. In: *2007 IEEE international geoscience and remote sensing symposium*, pp 667–670
 28. Chen C, Cheng Y (2008) Roads digital map generation with multi-track GPS data. In: *2008 International workshop on education technology and training 2008 international workshop on geoscience and remote sensing*, vol 1, pp 508–511
 29. Cao L, Krumm J (2009) From GPS traces to a routable road map. In: *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems (GIS'09)*. ACM, New York, pp 3–12
 30. Olivas F, Amador-Angulo L, Perez J, Caraveo C, Valdez F, Castillo O (2017) Comparative study of type-2 fuzzy particle swarm, bee colony and bat algorithms in optimization of fuzzy controllers. *Algorithms* 10(3):101
 31. Castillo O, Melin P, Pedrycz W (2011) Design of intervaltype-2 fuzzy models through optimal granularity allocation. *Appl Soft Comput* 11(8):5590–5601
 32. Valdez F, Vazquez JC, Melin P, Castillo O (2017) Comparative study of the use of fuzzy logic in improving particle swarm optimization variants for mathematical functions using co-evolution. *Appl Soft Comput* 52:1070–1083
 33. Castillo O, Huesca G, Valdez F (2008) Evolutionary computing for topology optimization of type-2 fuzzy controllers. In: Castillo O, Melin P, Kacprzyk J, Pedrycz W (eds) *Hybrid intelligent systems. Studies in fuzziness and soft computing*. Springer, Berlin, vol 208, pp 163–178
 34. Melin P, Olivas F, Castillo O, Valdez F, Soria J, Valdez M (2013) Optimal design of fuzzy classification systems using PSO with dynamic parameter adaptation through fuzzy logic. *Expert Syst Appl* 40(8):3196–3206
 35. Valdez F, Vazquez JC, Gaxiola F (2018) Fuzzy dynamic parameter adaptation in ACO and PSO for designing fuzzy controllers: The cases of water level and temperature control. *Adv Fuzzy Syst* 2018:1–19

36. Olivas F, Valdez F, Castillo O, Melin P (2016) Dynamic parameter adaptation in particle swarm optimization using interval type-2 fuzzy logic. *Soft Comput* 20(3):1057–1070
37. Ibrahim RW (2019) A new approach of utility function based on fractional Gini aggregation operator for intelligent access web selection. *SN Appl Sci* 1:1126
38. Selvaraj S, Sundaravaradhan S (2020) Challenges and opportunities in IoT healthcare systems: a systematic review. *SN Appl Sci* 2:139
39. Quddus M, Ochieng W, Zhao L, Noland R (2003) A general map matching algorithm for transport telematics applications. *GPS Solut* 7:157–167
40. Luo W, Tan H, Chen L, Ni LM (2013) Finding time period-based most frequent path in big trajectory data. In: *Proceedings of the ACM SIGMOD international conference on management of data (SIGMOD 2013)*, New York, NY, USA, 22–27 June 2013, pp 713–724
41. Yuan NJ, Zheng Y, Xie X, Wang Y, Zheng K, Xiong H (2015) Discovering urban functional zones using latent activity trajectories. *IEEE Trans Knowl Data Eng* 27(3):712–725
42. Zheng K, Zheng Y, Xie X, Zhou X (2012) Reducing uncertainty of low-sampling rate trajectories. In: *IEEE 28th international conference on data engineering (ICDE 2012)*, Washington, DC, USA (Arlington, Virginia), 1–5 April 2012, pp 1144–1155
43. Agamennoni G, Nieto JJ, Nebot EM (2011) Robust inference of principal road paths for intelligent transportation systems. *IEEE Trans Intell Transp Syst* 12(1):298–308
44. Barzehkar M, Dinan NM, Mazaheri S et al (2019) Landfill site selection using GIS-based multi-criteria evaluation (case study: SaharKhiz Region located in Gilan Province in Iran). *SN Appl Sci* 1:1082
45. de Jager D, Zweiri Y, Makris D (2019) A particle swarm optimization approach using adaptive entropy-based fitness quantification of expert knowledge for high-level, real-time cognitive robotic control. *SN Appl Sci* 1:1684
46. Shuaib M, Abdulhamid SM, Adebayo OS et al (2019) Whale optimization algorithm-based email spam feature selection method using rotation forest algorithm for classification. *SN Appl Sci* 1:390
47. Niehoefer B, Burda R, Wietfeld C, Bauer F, Lueert O (2009) GPS community map generation for enhanced routing methods based on trace-collection by mobile phones. In: *Proceedings of the 2009 1st international conference advanced satellite Space communications (SPACOMM 2009)*, pp 156–161
48. Qu M, Zhu H, Liu J, Liu G, Xiong H (2014) A cost-effective recommender system for taxi drivers. In: *The 20th ACM SIGKDD international conference on knowledge discovery and data mining (KDD 2014)*, New York, NY, USA, 24–27 August 2014, pp 45–54
49. Wolfson O, Sistla AP, Chamberlain S, Yesha Y (1999) Updating and querying databases that track mobile units. *Distrib Parallel Databases* 7(3):257–387
50. Chandio AA, Tziritas N, Zhang F, Xu C-Z (2015) An approach for map-matching strategy of GPS-trajectories based on the locality of road networks. In: Hsu C-H, Xia F, Liu X, Wang S (eds) *IOV 2015*, vol 9502. LNCS. Springer, Cham, pp 234–246
51. Schuessler N, Axhausen K, Zurich E (2009) Map-matching of GPS traces on high-resolution navigation networks using the multiple hypothesis technique (MHT), Working paper. In: *Institute for transport planning and system (IVT)*, ETH Zurich, Zurich, vol 568, pp 1–22
52. Swaroop V, Gupta GK, Shanker U (2011) Issues in mobile distributed real-time databases: performance & review. *Int J Eng Sci Technol* 3:3504–3517
53. Gupta AK (2020) Spam mail filtering using data mining approach: A comparative performance analysis. In: *Handling priority inversion in time-constrained distributed databases*, pp 253–282

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.