Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 579031, 10 pages
http://dx.doi.org/10.1155/2014/579031

Research Article

A Mobile, Dynamic, and Privacy-Preserving Matching System for Car and Taxi Pools

Shin-Yan Chiou and Yi-Cheng Chen

Department of Electrical Engineering, Chang Gung University, 259 Wen-Hwa 1st Road, Kwei-Shan, Tao-Yuan, Taiwan

Correspondence should be addressed to Shin-Yan Chiou; ansel@mail.cgu.edu.tw

Received 29 October 2013; Accepted 31 January 2014; Published 23 March 2014

Academic Editor: Moran Wang

Recently, rates of vehicle ownership have risen globally, exacerbating problems including air pollution, lack of parking, and traffic congestion. While many solutions to these problems have been proposed, carpooling remains one of the most effective approaches. Recently, several carpooling platforms have been built on cloud computing systems, with originators posting online list of departure/arrival points and schedules from which participants can search for rides that match their needs. However, it can be difficult to make matches quickly and the systems are subject to privacy concerns in that they may disclose private information such as names, registration data, and departure/arrival schedules. This paper proposes a dynamic matching method for car/taxi pools for use in mobile devices via ad hoc Wi-Fi networks. The proposed method also preserves user privacy including names and departure/arrival schedules. Moreover, the system does not require the user to register any personal data, so such data cannot be leaked. The system was implemented on the Android mobile platform, allowing users to immediately and securely access the system via their smart phones.

1. Introduction

In recent years, car ownership has exploded globally, exacerbating the negative side effects of driving, including depletion of fossil fuels and the production of carbon dioxide and other emissions; thus driving accelerates the development of air pollution, acid rain, and climate change. In addition, increased car ownership has increased traffic congestion [1–4], strained the availability of existing parking facilities [5–12], and reduced economic performance [13]. Scholars have proposed many solutions to these issues, including carpooling [14–20], which remains one of the most effective solutions.

In 1983 Fagin and Williams [14] first proposed a sharing mechanism, while other scholars [15–18] have also proposed various carpooling methods. In these methods, group members take turns serving as drivers. However, this approach is difficult to promote or apply widely because it only works if group members already know each other and agree ahead of time to organize a carpool.

In recent years, increases to network speeds have prompted scholars [19, 20] to propose the establishment of networked platforms [21–29] as a mechanism for organizing carpools. Di Martino et al. [19] proposed the Lift4U platform, a carpool matching algorithm based on cloud and positioning systems. The platform uses Bing Maps and GPS modules to provide access to car sharing, bike sharing, and public transport on mobile platforms. Lalos et al. [20] also describe a dynamic framework for car and taxi pool services through the use of positioning systems on mobile devices to determine the location of carpool members. These positions are then shown directly on the mobile device interface, helping carpool drivers to arrange pickups.

These systems use Internet and cloud computing technologies [30] to provide access to information required to participate in carpooling schemes; thus group members need not know each other or organize the carpool ahead of time. However, initiating a carpool or finding carpool originators entails a process of online registration, login, search, and comparison, thus producing a match between rider demand and driver availability. Therefore, in addition to being ineffective in helping carpoolers to coordinate in real time, these methods require originators to expose their personal information including name, location, destination, driving time, and expose participants' personal information including name and telephone number.

This paper proposes a carpooling matching method deployed on mobile devices, wherein users can instantly initiate or participate in carpools via smartphones or tablet PCs via ad hoc Wi-Fi networks. The proposed method also protects user privacy, using a private matching method [31, 32] to protect the privacy of participants on both sides of the transaction, including name, origin, destination, and driving time.

The proposed system was implemented on Android smartphones, allowing users to directly, immediately, and securely find carpoolers. The implementation can be used directly to find rides from private carpools or taxi pools. For example, imagine a group of strangers emerging from a subway station and arriving at the adjacent bus stop immediately after their intended bus had pulled away, leaving them to wait another 20 minutes for the next bus. These individuals can access the proposed system via their smartphones to arrange for a taxi pool, thus saving time and money, while protecting their personal information.

The remainder of this paper is organized as follows. Section 2 discusses security issues and solutions for the proposed method and describes the research methods. Section 3 describes the proposed method, while Section 4 presents the analysis methods used, including accuracy analysis, locational error, and security analysis, and provides a comparison with other related methods. Section 5 introduces the implementation of the proposed system on Android phones and the final section presents concluding remarks.

2. Related Work

Most research into carpooling focuses on scheduling algorithms and matching algorithms, which are briefly introduced here.

2.1. Carpool Scheduling Algorithms. Fagin and Williams [14] and others [15–18] have proposed fair carpool scheduling algorithms wherein the participants take turns to drive the rest of the group. Here, we introduce four approaches including simple rotation, simple tokens, subsets, and fair carpool. Among these, simple rotation has a fixed set of N members, where person i is responsible for driving on the ith day and every N driving days. This approach is simple to describe and it is easy to determine who drives next. However, inconsistencies (e.g., if a driver cannot drive and has to swap days) can disrupt the schedule and make it difficult to determine who drives next.

In the simple tokens approach, a participant P pays a driver D one "ride token" each time P rides with D. The person with the fewest tokens would be the next driver. However, external conditions may render the designated driver unable to drive on a given day without warning.

In the subset approach, each of the $2^N-(N+1)$ subsets of carpool members for N persons can drive particular groups. Drivers get "subset ride tokens" from that subset to determine who drives next. However, this approach is very complex given the large number of subsets.

In fair carpool [14], each participant pays U/k tokens to the driver in each carpool, where U is a constant value and k is the number of participants in each carpool time. However, this approach is unbalanced.

2.2. Carpooling Matching Algorithm. Many studies have proposed carpooling matching algorithms. Here, we review Di Martino et al.'s Lift4U platform [19], which uses Bing Maps and GPS modules to provide a range of carpooling applications (see Figure 1(a)). This method includes registration and data matching phases as follows.

(1) Registration Phase. In this stage, the user registers online. The user's identity is confirmed, using information including the user's name, gender, date of birth, and telephone number.

(2) Data, Matching Phase. This stage includes four steps through which the user is matched with a ride.

Step 1. The user accesses Bing Maps to establish points of departure and destination.

Step 2. The user sets a search radius around the departure point to search for other people seeking the same destination (see Figure 1(b)).

Step 3. The user finds other potential carpoolers within the search area (see Figure 1(c)), plots routes, and calculates the distances.

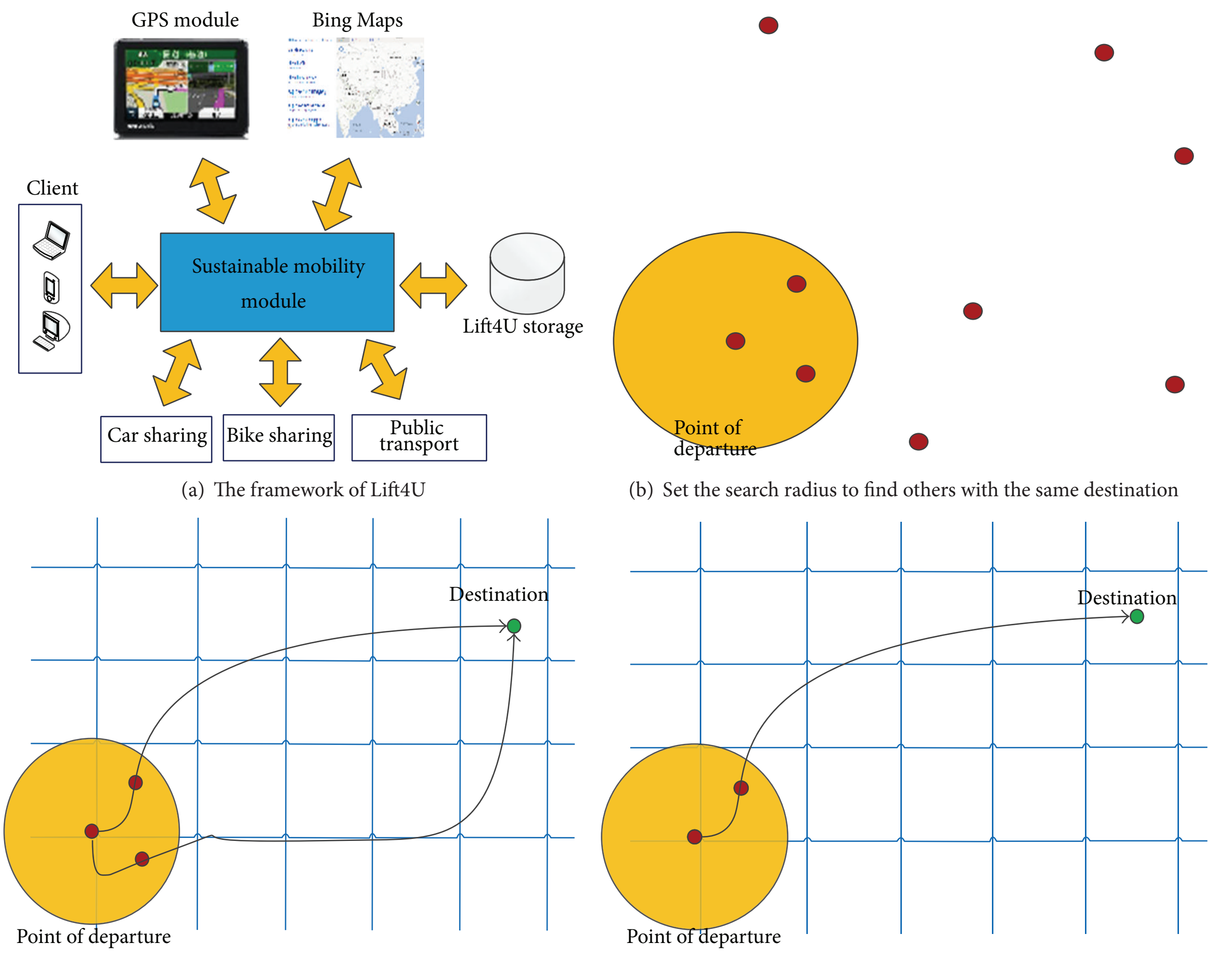
Step 4. The user determines the shortest route and invites potential carpoolers to join based on which pickups result in the shortest route (see Figure 1(d)).

3. Proposed Scheme

We use the error-tolerable private matching to achieve private carpool matching. This method not only allows the user to directly use mobile devices to find potential carpoolers with similar points of origin and destinations in real time but also allows the user to expand the search range using a quantifier, for which the quantification space can be autonomously selected (see Figure 2), thus finding more potential carpoolers.

This paper uses the symbols listed in Notations Section, where $Q(x,y)$ represents x quantized as $x^{(q)} \in \{ip \mid i = 0,1,2,\ldots,p = 2y\}$ s.t. $-y < x^{(q)} - x \leq y$. $\text{Map}(\cdot)$ is the transformed longitude and latitude function. For example, $\text{Map}(x)$ represents the transformed longitude and latitude for map message x. Using Google Maps, the map message Dst_A is thus transformed into the longitude and latitude message LL_A, and $\text{LL}_A = \text{Map}(\text{Dst}_A)$.

Figures 3 and 4 illustrate the operation process (using a taxi pool as an example). Figure 3 provides a detailed flow



(a) The framework of Lift4U

(b) Set the search radius to find others with the same destination

(c) Plot potential routes

(d) Select carpoolers

Figure 1: Lift4U system.

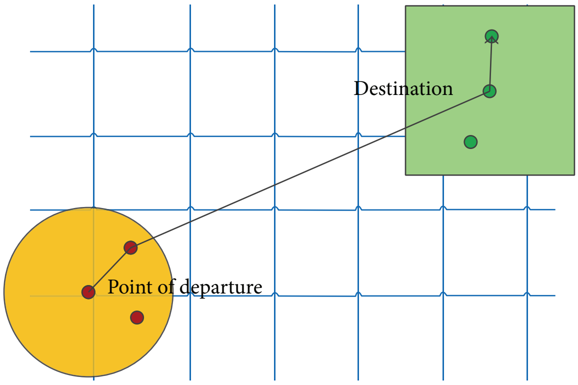


Figure 2: Carpooling schematic.

chart of the system operations, providing a detailed understanding or point of reference for programmers. Figure 4 provides a simplified schematic flow diagram for the system. In this system, "originator" is the car owner inviting others to join a carpool or is an initial taxi passenger inviting others to share the ride, and "participant" is a person who wants to join a car or taxi pool. The process goes through the following steps.

Step 1. A user can either select "Listen" to determine whether someone has already initiated the carpool matching service (go to Step 2) or directly acts as the originator of a car or taxi pool (go to Step 3).

Step 2. After Listening, if the user receives a "Taxipool" or "Carpool" message, he/she can then elect to act as participant or originator. Otherwise if a "Taxipool" or "Carpool" message is not received and the participant option is not available, the user can still elect to act as originator, or return to Step 1.

Step 3. If the user elects to act as originator, an appropriate (i.e., "Carpool" or "Taxipool") message is broadcast.

Step 4. If the user elects to act as participant, once he/she receives a "Taxipool" or "Carpool" message, he/she agrees to the match by sending a "GoMatch" message to the originator. Otherwise, go to Step 1.

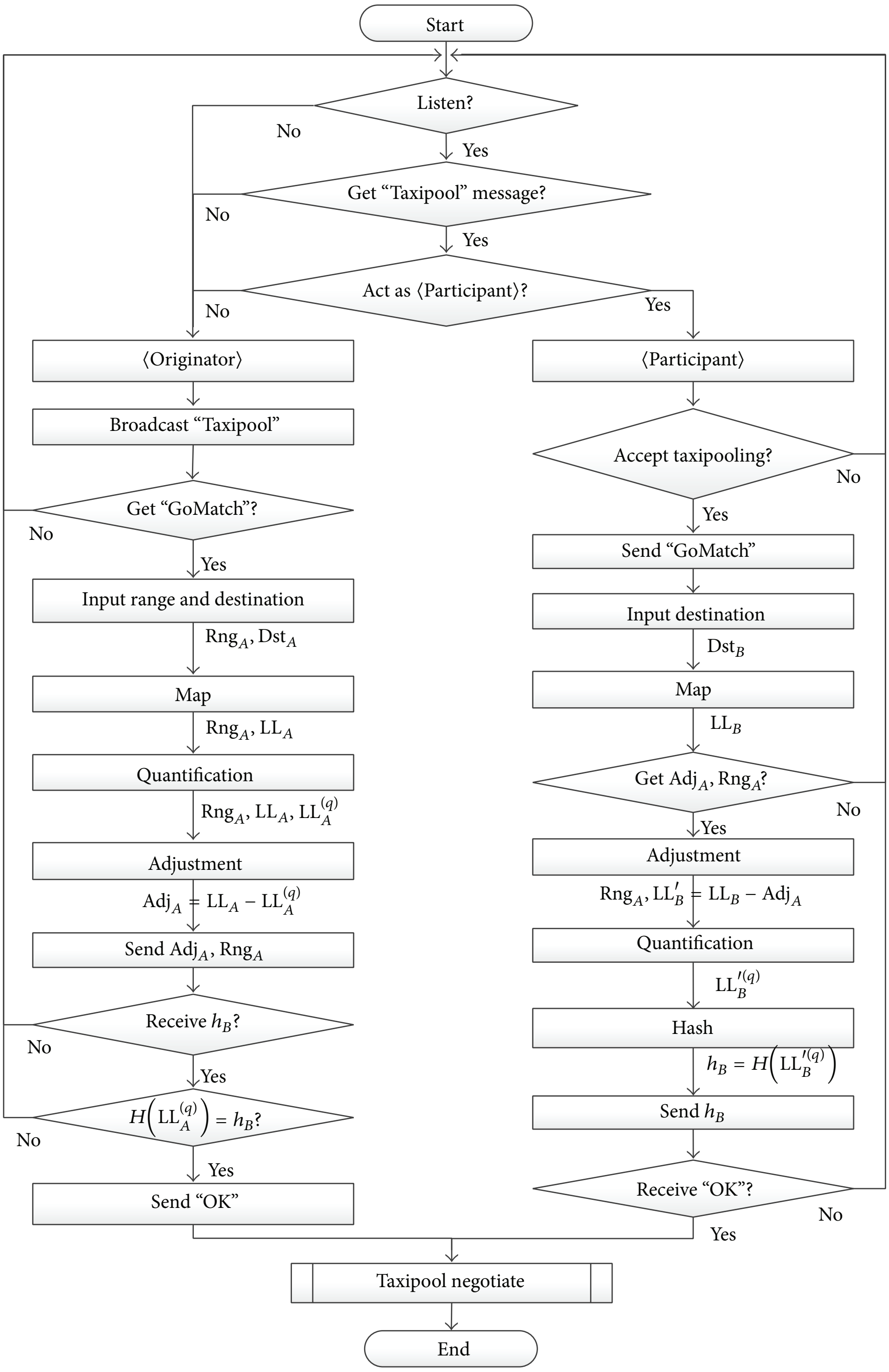

Start
Listen?
No
Yes
Get "Taxipool" message?
No
Yes
Act as ⟨Participant⟩?
No
Yes
⟨Originator⟩
⟨Participant⟩
Broadcast "Taxipool"
Accept taxipooling?
No
Get "GoMatch"?
No
Yes
Yes
Send "GoMatch"
Input range and destination
Input destination
Rng_A, Dst_A
Dst_B
Map
Map
Rng_A, LL_A
LL_B
Quantification
Get Adj_A, Rng_A?
No
Rng_A, LL_A, $\mathrm{LL}_A^{(q)}$
Yes
Adjustment
Adjustment
$\mathrm{Adj}_A = \mathrm{LL}_A - \mathrm{LL}_A^{(q)}$
Rng_A, $\mathrm{LL}'_B = \mathrm{LL}_B - \mathrm{Adj}_A$
Send Adj_A, Rng_A
Quantification
$\mathrm{LL}_B'^{(q)}$
Receive h_B?
No
Hash
Yes
$h_B = H\left(\mathrm{LL}_B'^{(q)}\right)$
$H\left(\mathrm{LL}_A^{(q)}\right) = h_B$?
Send h_B
No
Yes
Send "OK"
Receive "OK"?
No
Yes
Taxipool negotiate
End

Figure 3: System programming flow chart.

Step 5. The originator receives the "GoMatch" message and inputs his/her own destination Dst_A and search range Rng_A. He/she computes $\mathrm{LL}_A = \mathrm{Map}(\mathrm{Dst}_A)$, which uses $\mathrm{Map}(\cdot)$ to transform Dst_A into the latitude and longitude message LL_A and uses the quantizer to calculate $\mathrm{LL}_A^{(q)} = Q(\mathrm{LL}_A, \mathrm{Rng}_A)$, thus using LL_A to obtain the quantized $\mathrm{LL}_A^{(q)}$. He/she then uses the adjustment function to calculate $\mathrm{Adj}_A = \mathrm{LL}_A - \mathrm{LL}_A^{(q)}$. Finally he/she transmits Adj_A and Rng_A to the participant.

Step 6. The participant inputs his/her own destination Dst_B and calculates $\mathrm{LL}_B = \mathrm{Map}(\mathrm{Dst}_B)$. After receiving Adj_A and Rng_A from the originator, the participant uses the adjustment

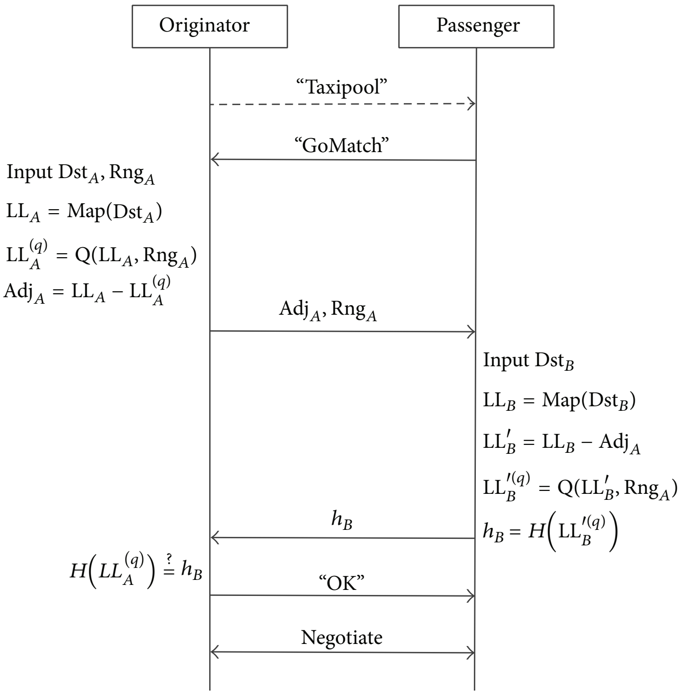

Originator
Passenger
"Taxipool"
"GoMatch"
Input Dst_A, Rng_A
$\mathrm{LL}_A = \mathrm{Map}(\mathrm{Dst}_A)$
$\mathrm{LL}_A^{(q)} = \mathrm{Q}(\mathrm{LL}_A, \mathrm{Rng}_A)$
$\mathrm{Adj}_A = \mathrm{LL}_A - \mathrm{LL}_A^{(q)}$
$\mathrm{Adj}_A, \mathrm{Rng}_A$
Input Dst_B
$\mathrm{LL}_B = \mathrm{Map}(\mathrm{Dst}_B)$
$\mathrm{LL}'_B = \mathrm{LL}_B - \mathrm{Adj}_A$
$\mathrm{LL}_B^{\prime(q)} = \mathrm{Q}(\mathrm{LL}'_B, \mathrm{Rng}_A)$
h_B
$h_B = H\left(\mathrm{LL}_B^{\prime(q)}\right)$
$H\left(LL_A^{(q)}\right) \overset{?}{=} h_B$
"OK"
Negotiate

FIGURE 4: System flow diagram.

function to calculate $\mathrm{LL}'_B = \mathrm{LL}_B - \mathrm{Adj}_A$ and then computes $\mathrm{LL}_B^{\prime(q)} = Q(\mathrm{LL}'_B, \mathrm{Rng}_A)$. Finally, the participant calculates $h_B = H(\mathrm{LL}_B^{\prime(q)})$ and transmits h_B back to the originator.

Step 7. After the originator receives h_B, he/she calculates $H(\mathrm{LL}_A^{(q)})$ and determines whether it matches h_B. If it matches, he/she transmits an "OK" message to the participant.

Step 8. Finally, when the participant receives the "OK" message, he/she begins to negotiate the taxi pool or carpool arrangement with the originator.

4. System Analysis and Comparison

4.1. Match Correctness. In this system, matching will only succeed if the distance between Dst_A and Dst_B falls within Rng_A. That is to say matching is only successful when

$$-\mathrm{Rng}_A < \mathrm{LL}_A - \mathrm{LL}_B \le \mathrm{Rng}_A. \tag{1}$$

Theorem 1. *In this system, if $-Rng_A < LL_A - LL_B \le Rng_A$, matching will be successful.*

Proof. Since $\mathrm{Adj}_A = \mathrm{LL}_A - \mathrm{LL}_A^{(q)}$ and $\mathrm{LL}'_B = \mathrm{LL}_B - \mathrm{Adj}_A$, we have

$$\mathrm{LL}_A - \mathrm{LL}_B = \mathrm{LL}_A^{(q)} - \mathrm{LL}'_B. \tag{2}$$

From the assumption $-\mathrm{Rng}_A < \mathrm{LL}_A - \mathrm{LL}_B \le \mathrm{Rng}_A$, we get

$$-\mathrm{Rng}_A < \mathrm{LL}_A^{(q)} - \mathrm{LL}'_B \le \mathrm{Rng}_A. \tag{3}$$

Also from the definition of $Q(x, y)$, we know that $-\mathrm{Rng}_A < \mathrm{LL}_B^{\prime(q)} - \mathrm{LL}'_B \le \mathrm{Rng}_A$ since $\mathrm{LL}_B^{\prime(q)} = Q(\mathrm{LL}'_B, \mathrm{Rng}_A)$. We can get

$$-\mathrm{Rng}_A \le \mathrm{LL}'_B - \mathrm{LL}_B^{\prime(q)} < \mathrm{Rng}_A. \tag{4}$$

From (3) and (4), we get $-2\mathrm{Rng}_A < \mathrm{LL}_A^{(q)} - \mathrm{LL}_B^{\prime(q)} < 2\mathrm{Rng}_A$. Because both $\mathrm{LL}_A^{(q)}$ and $\mathrm{LL}_B^{\prime(q)}$ belong to $\{ip \mid i = 0, 1, 2, \ldots, p = 2\mathrm{Rng}_A\}$, we can conclude that $\mathrm{LL}_A^{(q)} = \mathrm{LL}_B^{\prime(q)}$. □

Conversely, if (1) does not hold, matching fails. That is, if $\mathrm{LL}_A - \mathrm{LL}_B > \mathrm{Rng}_A$ or $\mathrm{LL}_A - \mathrm{LL}_B \le -\mathrm{Rng}_A$, matching fails.

Theorem 2. *In this system, if $-Rng_A < LL_A - LL_B \le Rng_A$ does not hold, matching fails.*

Proof. (1) If $\mathrm{LL}_A - \mathrm{LL}_B > \mathrm{Rng}_A$, from (2), we get

$$\mathrm{LL}_A^{(q)} - \mathrm{LL}'_B > \mathrm{Rng}_A. \tag{5}$$

From (4), we know that

$$\mathrm{LL}'_B - \mathrm{LL}_B^{\prime(q)} \ge -\mathrm{Rng}_A. \tag{6}$$

From (5) and (6), we conclude that $\mathrm{LL}_A^{(q)} - \mathrm{LL}_B^{\prime(q)} > 0$. Therefore, $\mathrm{LL}_A^{(q)} \ne \mathrm{LL}_B^{\prime(q)}$.

(2) If $\mathrm{LL}_A - \mathrm{LL}_B \le -\mathrm{Rng}_A$, from (2), we get

$$\mathrm{LL}_A^{(q)} - \mathrm{LL}'_B \le -\mathrm{Rng}_A. \tag{7}$$

From (4), we know that

$$\mathrm{LL}'_B - \mathrm{LL}_B^{\prime(q)} < \mathrm{Rng}_A. \tag{8}$$

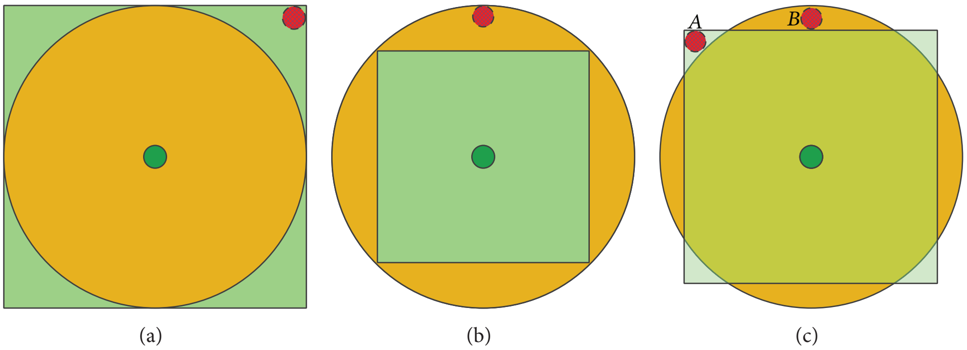


Figure 5: Pairing error.

From (7) and (8), we conclude that $\mathrm{LL}_A^{(q)} - \mathrm{LL}_B^{\prime(q)} < 0$. Therefore, $\mathrm{LL}_A^{(q)} \neq \mathrm{LL}_B^{\prime(q)}$. □

4.2. Obtaining Latitude/Longitude Values and Quantified Scopes and Regulating Pairing Errors. Google Maps features different coordinate formats and GPS modes, requiring conversion for interoperability. The conversion formula is as follows:

$$X \text{ degrees } Y \text{ minutes } Z \text{ seconds (GPS mode)} = X + \frac{Y}{60} + \frac{Z}{3600} \text{ (Google Maps decimal degrees)}. \quad (9)$$

Longitude and latitude have different relationships to distance. In latitude, one degree is equivalent to 111.19 km, while one degree of longitude is equivalent to 101.35 km. Therefore, an input range with an error value of 1 km can be used to obtain a quantized interval of 2 km, which is converted to approximately 2/111.19 degrees latitude and 2/101.35 degrees longitude. This produces a set of quantized values.

If the quantized values for latitude and longitude are obtained, a restriction exists that the matching range can produce a square-shaped "square pairing" area, two km on each side (see Figure 2), rather than a circular range with a diameter of two km for "circular pairing." Thus, the distance from the point of departure to the four corners of the square pairing area exceeds Rng_A, but matching is successful (see red stripped ball in Figure 5(a)), while the maximal error distance is $+(\sqrt{2}-1)\mathrm{Rng}_A$ (or the maximal distance error rate is +41.4%), and the average false acceptance rate (FAR) is 21.46% and false rejection rate (FRR) is zero. However, one can also adjust the determination method for control of maximal error distance in $-(1-(\sqrt{2}/2))\mathrm{Rng}_A$ (approximately −29.29% with FAR being zero and FRR being 36.34%) (see Figure 5(b)) or $\pm(3-2\sqrt{2})\mathrm{Rng}_A$ (approximately ±17.16% with FAR being 4.57% and FRR being 16.62%) (see Figure 5(c)). Herein, Figure 5(c) is a relatively acceptable option. That is, for the user's input error range of 1 km (1000 m), the worst-case scenario would be (1) successful matching at a distance of 1171.5 meters (see point A in Figure 5(c)) or (2) unsuccessful matching at a distance of 828.5 meters (see point B in Figure 5(c)).

4.3. Security Analysis. When using the program, users may be concerned that others may be able to access their departure or destination point information. The proposed protocol ensures user privacy, allowing users to engage with the system without fear that their personal information may be leaked to unauthorized parties. In the proposed scheme, attackers are unable to obtain a plain text message through analyzing the transmitted information. Moreover, after finishing this protocol, neither the originator nor participant is aware of each other's information unless they share a common destination. We analyze possible means of attack as follows.

(1) Participant Attempts to Eavesdrop on the Privacy of Originator. In the proposed protocol, the participant is only able to receive Adj_A and Rng_A messages and is unable to access private information related to the originator such as intended destination.

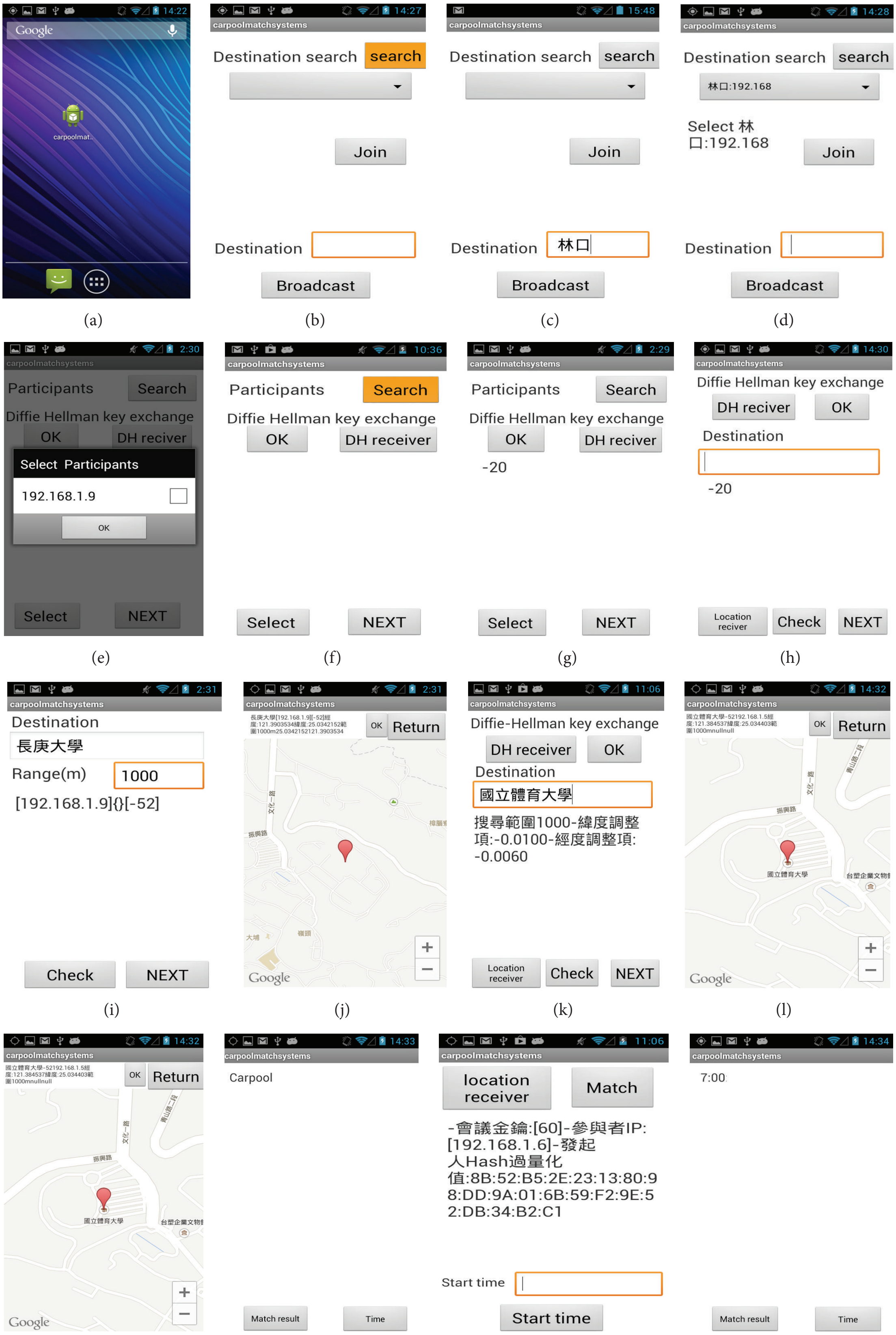
(2) Originator Attempts to Eavesdrop on the Privacy of Participant. In the proposed protocol, the originator is only able to receive the participant's h_B message. If matching fails, the originator is unable to access private information related to the participant such as intended destination.

(3) Third Party Attackers Attempt to Eavesdrop on the Privacy of Originator or Participant. In the proposed protocol, although an attacker can eavesdrop on the Adj_A, Rng_A, and h_B messages, he/she is unable to access private information, such as the intended destination of either the originator or participant.

Thus, if the originator and participant have different intended destinations, neither is aware of the other. In addition, if an attacker eavesdrops on these messages, the messages cannot be compromised thus preserving the privacy of both parties.

In addition, if the negotiated content is subject to eavesdropping concerns, one of the following two advanced steps can be taken.

(1) The originator (in Step 7) calculates $\mathrm{SK}_{AB} = H(\mathrm{LL}_A^{(q)} + 1)$ and the participant (in Step 8) calculates



(a)

(b)

(c) Destination 林口

(d) 林口:192.168

Select 林口:192.168

(e) 192.168.1.9 □

OK

(f)

(g) -20

(h) -20

(i) Destination 長庚大學

Range(m) 1000

[192.168.1.9]{}[-52]

(j) 長庚大學[192.168.1.9][-52]經度:121.3903534緯度:25.0342152範圍1000m25.0342152121.3903534

(k) Destination 國立體育大學

搜尋範圍1000-緯度調整項:-0.0100-經度調整項:-0.0060

(l) 國立體育大學-52192.168.1.5經度:121.384537緯度:25.034403範圍1000mnullnull

(m) 國立體育大學-52192.168.1.5經度:121.384537緯度:25.034403範圍1000mnullnull

(n)

(o) -會議金鑰:[60]-參與者IP:[192.168.1.6]-發起人Hash過量化值:8B:52:B5:2E:23:13:80:98:DD:9A:01:6B:59:F2:9E:52:DB:34:B2:C1

(p) 7:00

Figure 6: Implementation.

Table 1: Comparison.

Function	Fagin and Williams [14]	Lift4U [19]	Lalos et al. [20]	Proposed system
Carpool	v	v	v	v
Does not require GPS for matching	v			v
Users do not need to know each other		v	v	v
Spur-of-the-moment matching			v	v
Does not require registration				v
Does not require Internet access	v			v
Privacy protection				v

$\mathrm{SK}_{BA} = H(\mathrm{LL}_B^{\prime(q)} + 1)$, thus obtaining a secret protocol key $\mathrm{SK}_{AB} = \mathrm{SK}_{BA}$. This key can be used for encryption and decryption, thus allowing participants to negotiate carpool details.

(2) After the originator receives the "GoMatch" message (in Step 5), he/she can conduct a Diffie-Hellman key exchange of the key SK_{AB} with the participant, and this key can then be used to encrypt the following message.

4.4. Comparison. Table 1 presents a comparison of our proposed system with scheduling algorithms (e.g., Fagin and Williams [14]) and carpooling matching algorithms (including Lift4U [19] and Lalos et al.'s algorithm [20]). All the algorithms can provide carpool functions, while [19, 20] require GPS for the comparison. In terms of scheduling algorithms, because the system requires prior communication to coordinate matching, the users must know each other prior to using the system. In addition, because scheduling algorithms are planned ahead of time and Lift4U relies on a networked platform for matching, therefore it is unable to provide real-time, spur-of-the-moment carpooling. Most methods require prior registration or login and use networked platforms to run their matching algorithms, thus requiring Internet access. Also, most methods fail to consider privacy concerns. However, the proposed method provides all of these functions, while providing privacy protection.

5. Implementation

To validate the proposed method, we used the Android development platform to implement the matching program on a NEXUS S smartphone and used the SHA-1 hash algorithm as the hash function for the matching process. Using Wi-Fi as the transmission medium, we conducted actual tests of the privacy-enabled carpool matching system on a smartphone running Android.

As the flow chart shows in Figure 3, after the user initiates the program (see Figure 6(a)), the program first engages the "Listen" (i.e., "search") mode (see Figure 6(b)). If it fails to find a ride, it organizes a new carpool (see Figure 6(c)). We assume that Alice initiates a new carpool. She then inputs her approximate intended destination (e.g., city) and uses Wi-Fi to broadcast a carpool or taxi pool message to find potential carpoolers. When multiple users (i.e., participants) receive the carpool or taxi pool message through the "Listen" mode, (see Figure 6(d)), willing parties which match the originator's query specifications (in this case "Bob") transmit back a "GoMatch" (i.e., "Join") message. Alice receives the "GoMatch" message and then decides whether to approve the match (see Figure 6(e)).

Next Alice can opt to use a Diffie-Hellman key exchange (i.e., the "DH receiver") (see Figure 6(f)). If so, both parties will use the Diffie-Hellman key exchange to establish a session key (see Figures 6(g) and 6(h)). Next, Alice inputs her own destination Dst_A and error range Rng_A (see Figure 6(i)), using map to transmit the converted coordinates, longitude, and latitude (see Figure 6(j)) and generate the resulting adjustments Adj_A with Rng_A for transmission to Bob.

Once Bob receives Adj_A and Rng_A, he inputs his own destination Dst_B (see Figure 6(k)), using map to transmit the converted coordinates (see Figure 6(l)). The calculated h_B is then transmitted to Alice who then compares whether $H(\mathrm{LL}_A^{(q)})$ and h_B are identical (see Figure 6(m)). If the match is successful, Alice sends the "OK" message to Bob. When Bob receives the successful match message (see Figure 6(n)), he and Alice begin to proceed to the carpool negotiation (see Figures 6(o) and 6(p)).

6. Conclusion

In this paper, we propose a protocol by which mobile devices can be used to provide instant matching for carpools and taxi pools in a way that protects the information privacy of all parties (e.g., names, telephone numbers, travel times, origins, and destinations). We review previous similar schemes and compare their various features and properties, for example, whether they require the use of GPS, whether the users need to know each other in advance, whether they are required to register, whether they can protect the user's personal information, and whether they can be used to arrange spontaneous carpools instantly. We also perform a security analysis to illustrate the privacy controls of the proposed scheme. We discuss how to obtain the quantification space and quantized values from the longitudes and latitudes, explain methods for arranging matches within a given error range, and make recommendations for actual system implementation. We also show that matching will succeed as long as the distance between the two parties falls within a set error distance range and will fail if this distance is exceeded. Finally, we describe the implementation of the proposed scheme on an Android smartphone to demonstrate feasibility. Future work will focus on developing a more accurate method for producing "circular pairing" rather than "square pairing" to reduce error values or error ratios arising from the determination of coordinates.

Notations

LL_A:	Latitude and longitude of A
$\text{LL}_A^{(q)}$:	Quantified latitude and longitude of LL_A
Adj_A:	Adjustment value of A
Dst_A :	Destination input for A
Rng_A:	Destination range for A
SK_{AB}:	Session key for A and B
Carpooling:	Carpool match initiating message
GoMatch:	Match agreement message
$H(\cdot)$:	One-way hash function
$Q(x, y)$:	$x^{(q)} \in \{ip \mid i = 0, 1, 2, \ldots, p = 2y\}$ s.t. $-y < x^{(q)} - x \le y$
$\text{Map}(\cdot)$:	Latitude and longitude conversion function.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was partially supported by the NSC under Grant NSC 102-2221-E-182-038. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] H.-S. Qi, D.-H. Wang, and P. Chen, "Formation and propagation of local traffic jam," *Discrete Dynamics in Nature and Society*, vol. 2013, Article ID 748529, 12 pages, 2013.

[2] X. Zhang, H. Gong, Z. Xu, J. Tang, and B. Liu, "Jam eyes: a traffic jam awareness and observation system using mobile phones," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 921208, 9 pages, 2012.

[3] C. Fozza and M. Longinotti, "T-cell traffic jam in Hodgkin's lymphoma: pathogenetic and therapeutic implications," *Advances in Hematology*, vol. 2011, Article ID 501659, 9 pages, 2011.

[4] J. Fukuyama, "A probabilistic protocol for multihop routing in VANETs," *Journal of Electrical and Computer Engineering*, vol. 2010, Article ID 185791, 11 pages, 2010.

[5] J. Zhu, Y. Feng, and B. Liu, "PASS: parking-lot-assisted carpool over vehicular ad hoc networks," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 491756, 9 pages, 2013.

[6] Z. Chen, J. (Cecilia) Xia, and B. Irawan, "Development of fuzzy logic forecast models for location-based parking finding services," *Mathematical Problems in Engineering*, vol. 2013, Article ID 473471, 6 pages, 2013.

[7] Z. Zhang, X. Li, H. Yuan, and F. Yu, "A sreet parking system using wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 107975, 10 pages, 2013.

[8] Y. Ji, D. Tang, W. Guo, P. T. Blythe, and G. Ren, "Detection of outliers in a time series of available parking spaces," *Mathematical Problems in Engineering*, vol. 2013, Article ID 416267, 12 pages, 2013.

[9] Y. Liu, W. Wang, C. Ding et al., "Metropolis parking problems and management planning solutions for traffic operation effectiveness," *Mathematical Problems in Engineering*, vol. 2012, Article ID 678952, 6 pages, 2012.

[10] H. Zhao, L. Lu, C. Song, and Y. Wu, "IPARK: location-aware-based intelligent parking guidance over infrastructure-less VANETs," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 280515, 12 pages, 2012.

[11] Z. Mei, Y. Tian, and D. Li, "Analysis of parking reliability guidance of urban parking variable message sign system," *Mathematical Problems in Engineering*, vol. 2012, Article ID 128379, 10 pages, 2012.

[12] Z. Mei and J. Chen, "Modified motor vehicles travel speed models on the basis of curb parking setting under mixed traffic flow," *Mathematical Problems in Engineering*, vol. 2012, Article ID 351901, 14 pages, 2012.

[13] G. Ning, Z. Zhen, P. Wang, Y. Li, and H. Yin, "Economic analysis on value chain of taxi fleet with battery-swapping mode using multiobjective genetic algorithm," *Mathematical Problems in Engineering*, vol. 2012, Article ID 175912, 15 pages, 2012.

[14] R. Fagin and J. H. Williams, "A fair carpool scheduling algorithm," *IBM Journal of Research and Development*, vol. 27, no. 2, pp. 133–139, 1983.

[15] D. Coppersmith, T. Nowicki, G. Paleologo, C. Tresser, and C. Wu, "The Optimality of the On-line greedy algorithm in carpool and chairman assignment problems," IBM Research Report RC23721 (W0509-015), 2005, http://domino.watson.ibm.com/library/cyberdig.nsf/papers/057648E1F532E47C8525708-30050B106/$File/rc23721.pdf.

[16] T. Selker and P. H. Saphir, "TravelRole: a carpooling / physical social network creator," in *Proceedings of the International Symposium on Collaborative Technologies and Systems (CTS '10)*, pp. 629–634, 2010.

[17] M. A. Vargas, J. Sefair, J. L. Walteros, A. L. Medaglia, and L. Rivera, "Car pooling optimization: a case study in Strasbourg (France)," in *Proceedings of the IEEE Systems and Information Engineering Design Symposium (SIEDS '08)*, pp. 89–94, 2008.

[18] S. Yan and C.-Y. Chen, "A model and a solution algorithm for the car pooling problem with pre-matching information," *Computers and Industrial Engineering*, vol. 61, no. 3, pp. 512–524, 2011.

[19] S. di Martino, R. Galiero, C. Giorio, F. Ferrucci, and F. Sarro, "A Matching-algorithm based on the cloud and positioning systems to improve carpooling," in *Proceedings of the 17th International Conference on Distributed Multimedia Systems (DMS '11)*, pp. 90–95, 2011.

[20] P. Lalos, A. Korres, C. K. Datsikas, G. S. Tombras, and K. Peppas, "A framework for dynamic car and taxi pools with the use of positioning systems," in *Proceedings of the Computation World: Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns (COMPUTATIONWORLD '09)*, pp. 385–391, November 2009.

[21] CAR2GETHER, http://www.car2gether.com.

[22] Autostrade Carpooling, http://www.autostradecarpooling.it/.

[23] NuRide, http://www.nuride.com.

[24] OpenID Foundation, http://www.openid.net.

[25] Carpooling King, http://www.carpoolking.com/tw/zh-tw/.

[26] CarpoolingWorld, http://www.qcar.org.tw/.

[27] carpooling.com, http://www.carpooling.com/us/.

[28] Taxi pool, http://www.ucl.be/en-283417.html.

[29] eRideShare.com, http://www.erideshare.com/.

[30] P. Mell and T. Grance, *Draft NIST Working Definition of Cloud Computing*, vol. 15, 2009.

[31] S. Y. Chiou and Y. H. Huang, “Mobile common friends discovery with friendship ownership and replay-attack resistance,” *Wireless Networks*, vol. 19, no. 8, pp. 1839–1850, 2013.

[32] S.-Y. Chiou, “Secure method for biometric-based recognition with integrated cryptographic functions,” *BioMed Research International*, vol. 2013, Article ID 623815, 12 pages, 2013.

Advances in Operations Research
Advances in Decision Sciences
Journal of Applied Mathematics
Algebra
Journal of Probability and Statistics
The Scientific World Journal
International Journal of Differential Equations
International Journal of Combinatorics
Advances in Mathematical Physics
Journal of Complex Analysis
Journal of Mathematics
Mathematical Problems in Engineering
Abstract and Applied Analysis
Discrete Dynamics in Nature and Society
International Journal of Mathematics and Mathematical Sciences
Journal of Discrete Mathematics
Journal of Function Spaces
International Journal of Stochastic Analysis
Journal of Optimization