Fractional Geo-Encryption Protocol for Mobile Networks

Khaled H. Moustafa
Military Technical College, Cairo, Egypt

Abstract

The wide spread of WLAN and the popularity of mobile devices increase the frequency of data transmission among mobile users. However, most of the data encryption technology is location-independent. Indeed, an encrypted data can be decrypted anywhere. The encryption technology cannot restrict the location of data decryption. Therefore, our objective is to add a layer of security to the network without breaking the network rules and to decrease network traffic. These aims can be achieved by two ways. First, almost all the fractionally encrypted messages sent by senders are decrypted by the receivers by applying the inverse fractional Fourier transform in the same order of rotations. Second, a decrease in the message queuing delay occurs due to the no need for encryption technique as it was achieved by the fractional Fourier transform. In order to meet the demand of mobile users in the future, a location-dependent approach is proposed in this paper. Firstly a target latitude/longitude coordinate must be determined. The coordinate is incorporated with a random key for data encryption by fractional Fourier transform. The receiver can only decrypt the ciphertext when the coordinate acquired from GPS receiver is matched with the target coordinate, and the exact key for fractional transformation is applied.

1. Introduction

In the present time, it is possible to use hand-held GPS to navigate within a few meters because of the removal of signal-degrading Selective Availability (SA) from GPS signals on the 1st May 2000. The differential GPS (DPGS) provides the accuracy to less than one meter. In the daily life, GPS receiver is popularly used in car navigation, fleet management and even more fields. Previously, a GPS receiver was connected to the mobile devices, via cable or Bluetooth but it was somehow inconvenient for users. Now, mobile devices such as mobile phones, PDAs and laptops are equipped with an integral GPS receiver with most of the wireless communication capabilities, including GSM/GPRS/EDGE, quad-band GSM phone capabilities, IEEE 802.11 g, etc. and announced worldwide.

2. Geo-Encryption to ensure data security

The location-based encryption or geo-encryption used refers to a method of encryption in which the encrypted information, cipher text, can be decrypted merely at a specific location. It enables data to be encrypted for a specific place or broad geographic area and backs up constraints in time in addition to space and fully protects against any attempts to bypass the location feature. Any attempt to decrypt the data at any other location, the original plaintext information is not revealed due to the failure of the decryption process. A location sensor such as a GPS receiver is used to determine the location in order to encrypt and decrypt the data.

Scott and Denning designed a data encryption algorithm by using the GPS, named Geo-Encryption which was based on the traditional encryption system and communication protocol. For the sender, the data was encrypted according to the expected PVT (Position, Velocity and Time) of the receiver [2, 3]. A PVT-to-GeoLock mapping function was used to get the GeoLock key which performed bitwise exclusive-OR with a generated random key to achieve a GeoLock session key.

This session key was then transmitted to the receiver by using asymmetric encryption. For the receiver, an anti-proof GPS receiver was used to obtain the PVT data. Then, the GeoLock key was acquired using the same PVT-to-GeoLock mapping function. The key was performing exclusive-OR operation with the received GeoLock session key to obtain the final session key. Finally, the ciphertext was decrypted using the final session key. However, the PVT-to-GeoLock mapping function is the primary mechanism to ensure that the data can be decrypted successfully. If they occasionally communicate, it is troublesome for both sender and receiver to own the same mapping function before the data transmission.

3. Fractional Fourier Transform

The Fourier transform is one of the most important mathematical tools used in physical optics, linear system theory, signal processing, communications, quantum mechanics and else [7].

The word “fraction” is nowadays very popular in different fields of science. By recall fractional derivatives in mathematics, fractal dimension in geometry, fractal noise, fractional transformations in signal processing, and so forth. In general, “fractional” means that some parameter has no longer an integer value.

The classical (ordinary) Fourier transform can be regarded as a $\pi/2$ rotation in the time-frequency plane, and the FRT performs a rotation of signal to any angle. Moreover, fractional Fourier transform serves as an orthonormal signal representation for chirp signals. The fraction Fourier transform is also called rotational Fourier transform or angular Fourier transform in some documents [8].

Benefiting from the inherent structure of the FRT for non-stationary digital signal processing and analysis, a new data encryption-decryption scheme for GPS-based mobile networks is presented.

4. Encryption Algorithm

A method for data encryption using fractional Fourier transform and chaos theory is shown. At the Sender's side, random phase masks are generated using iterative chaos functions. The input data is combined with the first random phase mask and is then transformed using the fractional Fourier transform. After the first fractional Fourier transform, the second random phase mask, again generated by using the chaos functions. The second fractional Fourier transform operation is then carried out to obtain the encrypted data as shown in figure 1.

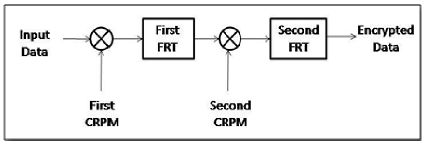

Fig. 1 Encryption process using FrFT_chaos function

At the receiver's side, Second random phase masks are generated using conjugate chaos functions. The encrypted data is combined with the second random phase mask and is then transformed using the Inverse fractional Fourier transform. After the first inverse fractional Fourier transform, the first random phase mask is again generated by using the conjugate chaos functions. The second Inverse fractional Fourier transform operation is then carried out to obtain the input data sent by sender as shown in figure 2.

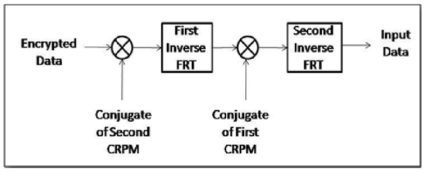

Fig. 2 Decryption process using IFrFT_chaos function

The advantage of the CRPM based encryption technique is that only the initial element called the seed value of the chaos function can be used to generate the CRPM for decryption. Thus, there is no need to send the whole phase mask to the receiver side for decryption. Two-dimensional chaos functions have advantage over one-dimensional chaos functions. In two-dimensional chaos function, two seed values are needed to encrypt and decrypt the data and hence security of the encrypted data is increased. Also using fractional Fourier transform provides superior signal recovery in case of signal distortion while transmitting through wireless network.

5. The Improved Geo-Encryption Protocol

There are many location-independent methods proposed for the security of data transmission in which the location of the receiver for data decryption could not be restricted by the sender. If the location is added to a data encryption algorithm, then this algorithm will be useful for increasing the security of mobile data transmission in the future. Therefore, an Improved Geo-Encryption Protocol (IGEP) was proposed in this paper. The latitude/longitude coordinate was used as the key for data encryption in IGEP. When a target coordinate is determined for data encryption, the ciphertext can only be decrypted at the expected location. Since the GPS receiver is inaccurate and inconsistent depending on how many satellite signals received. It is difficult for receiver to decrypt the ciphertext at the same location exactly matched with the target coordinate. It is impractical by using the inaccurate GPS coordinate as key for data encryption. Consequently, a Toleration Distance is designed in IGEP. The sender can also determine the Toleration Distance and the receiver can decrypt the ciphertext within the range of Toleration Distance.

6. Protocol Overview

The Improvement builds on top of existing wireless multihop routing protocols, thus it will not address the routing issues of mobile multi-hop networks. A simplified version of the geo-encryption protocol was evaluated by simulating a modified DSR protocol using GlomoSim. The Improvement will handle the communication of movement information between mobile nodes and the updating of this information whenever nodes move unexpectedly. One of its main goals is to enable mobile nodes to exchange their movement information accurately and to reduce the overhead on the network.

When a mobile node, say N1, wishes to communicate with another node N2, it broadcasts a message to discover a route to N2. When such a route exists, node N1 will receive a route reply specifying the sequence of hops to reach node N2. Therefore, Node N1 will need to keep a table of the positions of the nodes it intends to send data to. Position update messages from these nodes containing their coordinates update the table frequently.

The destination node N2, upon receipt, will map its true position to the intended location of the message using the shape parameters. If the result of the mapping matches the location sent by N1, the decryption will be considered successful else the message will be dropped.

7. The Algorithm

Sender's part:

1. Identify a first spatial location for a current location of the receiver, which is achieved by a GPS, and estimate the second spatial location according to the magnitude and the direction of the receiver.
2. Determine a vector between the first spatial location and the second spatial location, then extract a direction, velocity and time components with respect to the first spatial location and initiate the Toleration Distance.
3. The extracted parameters resulting from step 2 is utilized to generate a random phase mask using iterative chaos functions.
4. Combine the input data with the first random phase mask and transform using the fractional Fourier transform.
5. Again generate a second random phase mask by using the chaos functions and the extracted parameters resulting from step 2 and combine with results from step 4.
6. Transform the result from step 5 using second fractional Fourier transform operation to obtain the encrypted data.
7. The encrypted data is transmitted through network in a form similar to a white noise to the receiver.

Receiver's part:

1. Receive encrypted data throw the network.
2. Determine the Receiver's Current Location, Direction and Velocity.
3. Second random phase mask is generated using conjugate chaos functions and the determined parameters resulting from step 2.
4. Combine the encrypted data with the second random phase mask and transform using the Inverse fractional Fourier transform.
5. Again generate the first random phase mask by using the conjugate chaos functions and the extracted parameters resulting from step 2.
6. The second Inverse fractional Fourier transform operation is then carried out.
7. If the acquired coordinate used to encrypt the data is matched with the target coordinate within the range of Toleration Distance, the encrypted data can be decrypted back to obtain the original plaintext. Otherwise, the result is indiscriminate and meaningless.

8. The IGEP proposed model

A movement model based on the geo-encryption technique was proposed in which both sender and receiver are mobile [1], and can securely deliver their current locations to one another whenever necessary. Accordingly, the intended movement of each mobile node that will be receiving geo-encrypted messages needs to be delivered to the potential sender nodes in order to estimate the mobile node's expected location at any point in time. This comes by sending information regarding the mobile node's movement, called movement parameters, to the sender through a sequence of message exchanges.

8.1 Movement parameters

In the proposed model, the geo-locking function takes shape, time, velocity, direction, and two maneuverability parameters. The ellipse shape suits the decryption zone because it has a length and breadth, and when both are equal, the ellipse becomes a circle that provides uniform coverage in all directions. The period during which decryption is possible is defined by the time parameter. When N1 is in motion, N2 will need to calculate a time parameter that represents a future time when N1 will actually be in the decryption zone when a geo-encrypted message arrives for decipherment at N1.

Figure 3 shows the four movement parameters that a mobile node uses to advertise its movement information.

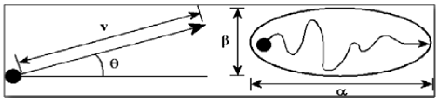

Fig. 3 Diagram to illustrate the four movement parameters

The velocity parameter, υ, describes the recipient's speed. Velocity (υ) is determined from observing the distance travelled during a specified time unit. The direction parameter, θ, measured as the positive angle between the positive x-axis and the velocity vector on a Cartesian coordinate system describes the recipient travelling direction.

The speed manoeuvrability parameter, α, indicates the frequency at which the moving recipient might need to change speeds while travelling to the new destination and influences the length of the ellipse-shaped decryption zone. Secondly, the breadth-manoeuvrability parameter, β, defines how much the moving recipient might deviate from a straight line while traveling to a new destination and affects the breadth of the ellipse-shaped decryption zone.

Based on the recent movement of a mobile station, it determines its own velocity and maneuverability parameters and exchanges them with other stations for use in geo-locking messages. The decryption zone only needs to be large enough for N1 to extract the geo-secured decryption key within the specified time period, not for N1 to decrypt the accompanying message.

8.2 Movement updates

One of the most important control messages in our movement model is the movement update message. The mobile receiver node must keep information about its movement and advertise it to the sender if needed in order to keep track of its location.

N1 sends its current movement parameters (υ, θ, α, and β), N1's current location (LN1), and the current time t to N2 in a movement update geo-encryption message:

$$E(\{\upsilon, \theta, \alpha, \beta, LN1, t\}, LN2)$$

Based on this information, node N2 can predict node N1's future location until N1 sends another movement update message to N2. Because N2 will be estimating N1's location based on this movement information, there will be errors between the estimated and actual location of N1. Therefore, LN1 and t are sent in the movement update message from N1 to N2 so that N2 can determine N1's future location knowing that N1 was at location LN1 at time t. Similarly, N2 has to know N1's initial location at the start of the mobile geoencryption session.

9. The model equations

Suppose the mobile node N1 starts at time t_0 at a location whose longitude and latitude values are $LN10(X_0,Y_0)$, which are assumed to be initially known to node N2 using the GPS. Periodically, node N1 collects GPS location satellite readings $LN1_t\ (X_t,Y_t)$ at time t with $t = t_1, t_2, t_3$, such that $t_i = t_0 + i_d$ where d is a fixed time unit interval whose value is arbitrary but known.

To define the decryption region for the mobile node N1, it is assumed that some initial values are available for the movement parameters α_0, β_0, υ_0, and θ_0 at time t_0. Given these initial values for the movement parameters and LN10(X_0,Y_0) as the initial values for the center of the ellipse, the decryption region for node N1 is defined initially by substituting these values in Eq. (2).

The line of movement makes an angle θ_0 with the positive direction of the latitude. As time progresses, the decryption region for node N1 moves along that line at a constant velocity V_0 (Figure 5). The movement of node N1 itself is arbitrary in any direction and any velocity but otherwise restricted to the decryption region at all times. The parameters of the center of the decryption region constantly change with time but not the shape [4]. The parameters of the shape of the region are fixed and could be changed only when a predetermined fixed number n of time units has passed. The center (CX_t,CY_t) of the decryption region at time t is given by

$$CX_t = X_0\ (t - t_0)\ V \cos\theta$$

$$CY_t = Y_0\ (t - t_0)\ V \sin\theta \qquad (1)$$

Thus, at any time node N2 needs only the initial parameters and the time value t to locate the center of the decryption region. In addition, the maneuverability parameters α and β together with the movement direction θ determine the shape of the region. If the region was assumed to have the bivariate normal distribution with center (CX_t,CY_t) and if the 3-sigma rule was adopted then the equations relating the shape parameters of the region with the manoeuvrability parameters are given by

$$\sigma_x = CX_t + {}^{1}/{}_{6}(\alpha - CX_t)\cos\theta + {}^{1}/{}_{6}(\beta - CY_t)\sin\theta$$

$$\sigma y = CY_t - {}^{1}/{}_{6}\ (\alpha - CX_t)\sin\theta + {}^{1}/{}_{6}(\beta - CY_t)\cos\theta \qquad (2)$$

Hence, at time t, the decryption region is defined by:

$$R(Xt,Yt) = \frac{(Xt - CXt)^2}{\sigma^2 X} + \frac{(Yt - CYt)^2}{\sigma^2 Y} - 2\,\rho\,\frac{(Xt - CXt)}{\sigma X} + \frac{(Yt - CYt)}{\sigma Y} \leq c \qquad (3)$$

where $\rho = \cos\theta$ and c is a constant determined from values of α and β.

10. Parameter estimation and update

10.1 Estimating the movement parameters

Although the GPS readings were not needed to locate and determine the decryption region in the above equations, nevertheless they must be found to estimate and update the movement parameters. The GPS readings LN1_t (X_t,Y_t) at time $t = t_1, t_2, t_3$, . . . are found and always used to calculate and update the velocity V, and the angle θ, of the decryption region. These values will be used to update the initial values V_0 and θ_0 each time the region changes its direction or velocity. In such an event, the initial values α_0, β_0 were updated using the last n GPS readings (X_t,Y_t) using the Gauss–Markov model given by

$$Zt = \gamma Z_{t-1} + (1 - \gamma)\mu + \sqrt{1-\gamma^2}\,\varepsilon_{t-1}$$

$t = 1, 2, 3, \ldots$

where $0 \leq \gamma \leq 1$ is a tuning parameter representing different levels of randomness, μ is the asymptotic mean of Zt, and ε_t are uncorrelated stationary random Gaussian process with zero mean and unknown standard deviation σ. In case γ equals to 1, the model is similar to random walk model, while in case γ equals to zero, the model is the constant velocity fluid flow model. The value of γ could either be random or estimated from the data. Assuming that both the velocity and the angle of the mobile node follow the model, and taking the asymptotic means of velocity and angle equal, respectively, to their initial values V_0 and θ_0, the estimates of velocity and the angle from the kth period are obtained as

$$\hat{v}_k = \gamma\ \hat{v}_{k-1} + (1 - \gamma)\ V_0$$

$$\hat{\theta}_k = \gamma\ \hat{\theta}_{k-1} + (1 - \gamma)\ \theta_0, \text{ where } k = 1, 2, 3, \ldots \qquad (4)$$

$$\hat{V}_0 = \frac{1}{n-1}\sum_{t=1}^{n-1}\sqrt{\left(\frac{Xt - Xt\text{-}1}{d}\right)^2 + \left(\frac{Yt - Yt\text{-}1}{d}\right)^2}$$

$$\hat{\theta}_0 = \arctan\left\{\frac{1}{n-1}\sum_{t=1}^{n-1}\left(\frac{Yt - Yt\text{-}1}{Xt - Xt\text{-}1}\right)\right\} \qquad (5)$$

Notice that the k^{th} period starts at time t_n(k_1) and ends at time t_nk_1, the formulas for estimating α and β were obtained by inverting the formulas in (Eq. (2)) to get

$$\hat{\alpha} = CX_t + 6(\hat{\sigma}x - CX_t) \cos\hat{\theta} - 6\ (\hat{\sigma}y - CY_t) \sin\hat{\theta}$$

$$\hat{\beta} = CY_t + 6(\hat{\sigma}x - CX_t) \sin\hat{\theta} - 6(\hat{\sigma}y - CY_t) \cos\hat{\theta} \qquad (6)$$

where

$$\hat{\sigma}_x = \frac{1}{n-1}\sum_{t=1}^{n}(Xt - \overline{X})^2, \quad \overline{X} = \sum_{t=1}^{n} Xt\ /n$$

$$\hat{\sigma}_y = \frac{1}{n-1}\sum_{t=1}^{n}(Yt - \overline{Y})^2, \quad \overline{Y} = \sum_{t=1}^{n} Yt\ /n \qquad (7)$$

10.2 Updating the movement parameters

Each time the movement parameters are estimated, the mobile node must decide whether or not to replace the old values of the parameters with the new values and whether or not to advertise them. Only when the difference between the old and new parameter values is more than the threshold determined by the mobile node, the old values are automatically replaced by the new ones.

In addition to the movement parameters, the initial location parameters (X_0, Y_0) of the mobile node must also be updated once either or both of *V* or *θ* are found significant. This is because the geo-encryption process depends on determining the center (CX_t, CY_t) and, as noted from Eq. (1), the recipient needs (X0,Y0) to estimate the center. If at time t^* a significant *V* or *θ* is detected then not only the four movement parameters are advertised but also the new value for t_0 which is estimated by $\hat{t}0 = t^*$. Given the values of $\hat{V}$, $\hat{\theta}$, and $\hat{t}$ the recipient will use Eq. (1) to estimate the updated initial location ($\hat{X}0$; $\hat{Y}0$).

Selecting optimal threshold values for the parameters (the smaller they are the more often updated and advertised) is the main criteria for optimizing the decryption zone in order to achieve balance between the probability of the mobile node falling within the decryption zone and the frequency of movement updates and advertisements.

11. Fingerprint generating

Obtain latitude/longitude coordinate from the GPS receiver is the GLL format (Lat/Lon data) N 30.0314 means 30° and 314 min north latitude. E 31.2106 means 31° and 2106 min east longitude. The coordinates are multiplied 1000 to be converted to an integer value. Then, the integer is divided by a the values are 5.4 and 6 for latitude and longitude corresponding to 1 m of the Toleration Distance – which is initialized by 5 m in the proposed model - according to the estimation of CoordTrans tool of Franson Company, respectively. In advance, one bit is put in front of the integral part of the above result. The bit is zero for east and south and one for west and north. Finally, the transformation results of the above step are combined with the Velocity, Direction and Toleration Distance to generate a fingerprint used as a seed value in encrypting the messages.

12. Fingerprint generating

A new method for data encryption using FRT and chaos is proposed. The data to be encrypted is FRT two times and the random phase masks generated using chaos functions are placed in the intermediate planes. Three chaos functions have been used to generate the chaotic random phase masks (CRPM). These functions are logistic map, tent map and Kaplan–Yorke map.

12.1 The fractional Fourier transform

The FRT is a generalization of the ordinary Fourier transform. For a transformable function f(x), the FRT of order p is defined as:

$$f_p(x_p) = F_p\{f(x)\}(x_p) + \int_{-\infty}^{+\infty} f(x)K_p(x, x_p)dx \qquad (8)$$

The kernel is given by

$$K_p(x, x_p) = \frac{e^{-i(\pi\hat{\phi}/4-\phi/2}}{\sqrt{|sin\ \varphi|}} e^{[i\pi(x^2\cot\phi - 2xx_p\ \csc\phi + x_p^2\cot\phi)]}, 0 < |p| < 2.$$

$$= \delta(x - x_p), p = 0,$$

$$= \delta(x - x_p), p = \pm 2,$$

(9)

where $\varphi = p(\pi/2), \hat{\varphi} = sgn(\sin\varphi)$. *p* is the order of the FRT. F_p expresses the FRT of order ‘p’, x and x_p are the coordinates in the input domain and output p^{th} fractional domain, respectively. For p=1, the FRT is equivalent to the ordinary Fourier transform. The fourth order of the FRT is equivalent to the original function. The FRT is a linear transform. The FRT is additive in indices, i.e.

$$F_{p1}\left\{F_{p2}\{f(x)\}\right\} = F_{p1+p2}\{f(x)\}.$$

12.2 Chaos functions

Chaos functions have been used mainly to develop the mathematical models of non-linear systems. There are several interesting properties of the chaos functions. These functions generate iterative values which are completely random in nature but limited between bounds.

Convergence of the iterative values after any value of iterations can never be seen. Chaos functions have extreme sensitivity to the initial conditions. Three chaos functions have been used. The first chaos function used in model is the logistic map and is defined as:

$$f(x) = p.x.(1-x) \qquad (10)$$

This function is bounded for 0<p<4 and can be written in the iterative form as:

$$x_{n+1} = p.x.(1-x_n) \qquad (11)$$

with ‘x_0’ as the initial value. This is also known as the seed value for the chaos function. The second chaos function used is the tent map [160, 161] and is defined as:

$$f(x) = a.x \qquad \text{for} \quad 0 \le x \le 0.5, \ (12)$$

$$f(x) = a(1-x) \qquad \text{for } 0.5 \le x \le 1 \ \ (13)$$

This function is bounded for 0<a≤ 2 and can be written in the iterative form as:

$$x_{n+1} = a.x_n \qquad \text{for} \quad 0 \le x \le 0.5, \ (14)$$

$$x_{n+1} = a(1-x_n) \qquad \text{for } 0.5 \le x \le 1. \quad (15)$$

with ‘x_0’ as the initial value. The third chaos function used in the model is the Kaplan–Yorke map and is defined as:

$$f(x) = a.x.mod\ 1, \qquad (16)$$

$$f(x) = by + \cos(4\pi x). \qquad (17)$$

This is bounded for 0≤a≤2 and 0≤b≤1 and can be written in the iterative form as:

$$x_{n+1} = a.x_n.mod\ 1, \qquad (18)$$

$$y_{n+1} = b\,y_n + \cos(4\pi x_n), \ (19)$$

with ‘x_0’ as the initial value. These chaos functions are used to generate random phase masks. Logistic map and tent map are one-dimensional chaos functions and the Kaplan–Yorke map is a two-dimensional chaos function. For the two-dimensional chaos function, two seed values are required to generate the CRPM.

12.3 Proposed encryption technique

The proposed encryption and decryption technique is based on the FRT and chaos. The FRT is performed two times and the random phase masks generated by using the chaos functions are introduced at the intermediate planes. Let f(x) denotes the original data to be encrypted. The input parameters is multiplied by the first CRPM represented by the phase function exp[iπC_1(x)], where C_1(x) is the random number sequence generated by the chaos function. The first FRT operation of order p_1 is performed over this to give us:

$$F_{p1}\{ f(x) \exp[i\pi C_1(x)]\}, \qquad (20)$$

where ‘p_1’ is the fractional order of the first FRT. This distribution is encoded by the second CRPM which is mathematically expressed as the phase function of the second CRPM exp[iπC_2(x_2)] together, where C_2 (x_2) is the second chaotic function obtained by using a different seed value. Then it goes into the second FRT operation. The distribution at the output plane is given by:

$g(x)= F_{p2}$ { F_{p1}{f(x) exp[$i\pi C_1(x)$]}

exp[$i\pi C_2(x)$]}, (21)

where ‘p_2’ is the fractional order of the second FRT operation. The decryption process is the inverse of the encryption process. The first inverse FRT (FRT of order – p_2) operation is performed over g(x) and then multiplied by the conjugate of the second CRPM. On the output obtained, the second inverse FRT (FRT of order - p_1) is performed over it and then multiplied by the conjugate of the first CRPM. The decrypted data is then obtained.

The decryption process is expressed as:

f(x)= F-p1 { F-p2 { g(x)} exp[-$i\pi C_2(x)$]} exp[-$i\pi C_1(x)$] (22)

13. Model Assumptions

In the design of the model, several security and reliability assumptions: that routing is secure; that authentication is assured by protocols other than geo-encryption; that the GPS hardware works flawlessly, is tamper-proof and unspoofable; that transmissions use some sort of spread spectrum method in order to counter triangulation attempts by rivals searching for our stations; and that rival electronic countermeasures do not jam our mobile stations’ transmitters (presumably in laptop computers).

The position update aspect of the protocol be reactive similarly to DSR routing [4]. To simulate geo-encryption using DSR, the following assumption was made:

Authentication is present and free: the establishment of communication between two nodes is assumed authenticated and anytime encryption fails the authenticated relationship is assumed reestablished automatically by the position update message.

Message decryption is considered part of another layer. Our simulations are only concerned with position updates; if the message contains the right position parameters *x* and *y* to a certain tolerance ± Tolerance, the message is considered decrypted.

When a receiver receives a message with no set protocol flags (RouteRequest, RouteReply, and PositionUpdate), the node creates its own real X and Y values (GPS) and compares then with those of the packet header. If $x \in [X_{real} \pm \text{Tolerance}]$ and $y \in [Y_{real} \pm \text{Tolerance}]$ the message is considered decrypted and an acknowledgment message is sent back to the sender otherwise the message is considered not decrypted.

Since the message is not decrypted. Therefore, a message constructs with the real X and Y values in the *x* and *y* fields, sets the Position Update Flag and sends the message on the reverse route that the received message arrived on. When the message is received and it has its Position Update Flag set, it then updates the table entries corresponding to the source node that sent the message.

To evaluate the protocol lines were added to the trace file indicating the following events:

- A message was successfully decrypted.
- A message failed decryption.
- A Position Update Message was sent.
- A position Update Message was received.

14. Simulation Setup

First, the data was plotted for one of the files at our disposal and, after proper unit conversion, determined a 150 · 150 m area. The movements of nodes was selected within that area during a 15 min period.

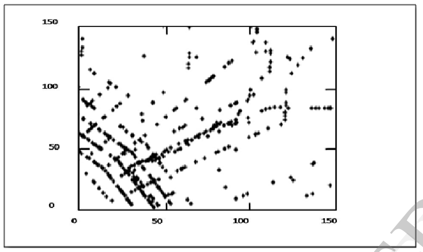

Fig. 4 Final simulation data

And finally, from the remaining data, the 50 nodes with the most number of updates in the given period were selected as shown in Figure 4. From that an initial position file and movements file were created to include in our simulation file. In order to create several movement files with decreased mobility; the pause times 10, 25, 50, 75, 100, 200, 400, 650 and 900 s were included.

For each mobility file, three runs with 10, 20 and 30 senders with 10, 20 and 30 receivers respectively were simulated to record the decryption ratio, ratio of the successfully decrypted messages amongst those that were received.

15. Simulation results

In this section, the performance evaluation of the previous model GEP (TD 10), where GEP refers to Geo-encryption Protocol and TD refers to tolerance distance , will be evaluated and compared to performance of IGEP (TD 10) algorithm with tolerance of 10, where IGEP refers to Improved Geo-encryption Protocol. The performance of GEP (TD 10) and IGEP (TD 10) will be evaluated considering different Network Size with fixed tolerance distance 10 m.

15.1 Using 10 Senders and 10 Receivers

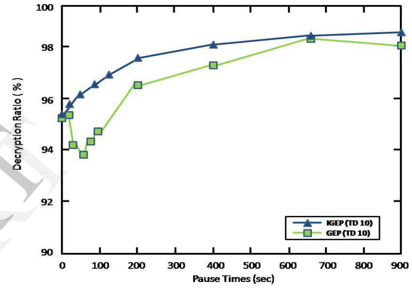

Fig. 5 Decryption Ratio vs. Mobility for 10 sources

The decryption ratio at lower mobility is better than that at higher mobility as shown in figure 5. This is due to the fact that higher mobility means that nodes move more often away from their estimated positions at the sending nodes which results in more messages not being decrypted.

15.2 Using 20 Senders and 20 Receivers

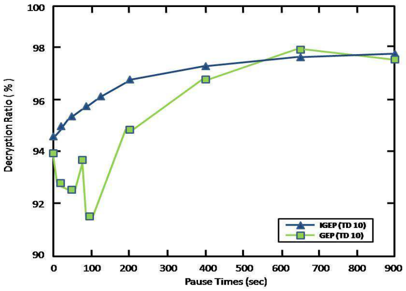

Fig. 6 Decryption Ratio vs. Mobility for 20 sources

As shown in figure 6.3, it was clear that an increase in users, from 10 to 20 senders and receivers, results in a decrease in decryption. This could be elaborated by the increased delay in message delivery because of the increased network congestion. If a message is buffered often along the way, it permits time to the destination node to move further away from its current position and thus increase the change of a decryption failure.

15.3 Using 30 Senders and 30 Receivers

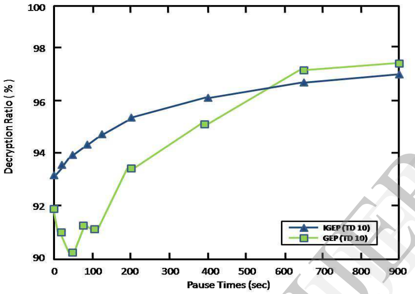

Fig. 7 Decryption Ratio vs. Mobility for 30 sources

As the number of senders and receivers increases, as shown in figures 5, 6 and 7, network congestion increases. This increases message buffering which enables the destination node to change current position and ensures the fact that at low mobility, the gap between the ratios of the three cases is lessened.

In conclusion, the decryption ratio of messages using IGEP (TD 10) algorithm of tolerance distance value of 10 is better than that GEP (TD 10) of tolerance distance value of 10 algorithm within the increase of Network Size.

16. Conclusion and Future work

Traditional encryption technology cannot restrict the location of mobile users for data decryption. Geo-encryption is an approach to location-based encryption that builds on established cryptographic algorithms and protocols. It allows data to be encrypted for a specific place [6]. In order to meet the demand of mobile users, IGEP algorithm is proposed in this paper. The algorithm can support both fixed and mobile applications and a variety of data-sharing and distribution policies. It provides full protection against location bypass. IGEP provide a new function by using the latitude/longitude coordinate as the key of data encryption. A Toleration Distance (TD) is also designed to overcome the inaccuracy and inconsistent of GPS receiver. As a result, IGEP is effective and practical for the data transmission in the mobile environment.

Current design of IGEP algorithm is mainly based on the FrFT algorithm. Other algorithms, such as Cyclic transforms, Wigner distributions, Cosine and Sine transforms, Hilbert transform, Hankel transform, Radon transform, wavelet transform, DFrFt (Discreet Fractional Fourier Transform), etc., can used to replace the FrFT algorithm when necessary.

The alternative IGEP algorithms incorporating with the mature algorithms can be developed to demonstrate its flexibility

Some factors can be incorporated into IGEP, such as time, to increase the security strength and usability of IGEP.

17. References

[1] D.E. Denning, Geo-encryption, GeoCodex LLC and Naval Postgraduate School, June 2004.

[2] L. Scott, D. Denning, Geo-encryption: Using GPS to Enhance Data Security, GPS World, 2003.

[3] Trimble Information Services, Powering the Transformation of Location Data into Location Information, 2002.

[4] Ala Al Fuqaha and Omar Al-Ibrahim, "Geo-encryption Protocol for mobile networks", Computer Comunication Jurnal, Vol. 30, No. 11, 2007.

[5] Scott, L. and D.E. Denning, Using GPS to enhance data security: Geo-Encryption. GPS World, 14: 40-49, 2003.

[6] Geo-Encryption|GPS World, [URL: www.gpsworld.com], last updated: 20/9/2013.

[7] M. Ertosun, H. Ath, H. Ozaktas, and B. Barshan, "Complex Signal Recovery from Two Fractional Fourier Transform Intensities: Order and Noise Dependence", Optics Communications, Vol. 244, No. 1, pp. 61-70, January 2005.

[8] Amein et al, "Fractional Chirp Scaling Algorithm: Mathematical Model," IEEE Trans on SP, vol.6, No.2, 2008.