

# SIP registration optimization in mobile environments using extended Kalman filter

Vesselin Tzvetkov  
vesselin.tzvetkov@arcor.net

Arcor AG&Co KG, Alfred-Herrhausen-Allee 1, 65760 Eschborn, Germany

**Abstract-** The Internet is becoming the dominating medium for modern communications. The SIP protocol delivers the majority of VoIP services. The SIP usage in mobile environments becomes reality due to the widespread of portable devices together with dynamic lifestyle. Unfortunately, SIP is not designed to work in mobile environments where access network and IP addresses change frequently. Intermediate NAT devices change additionally the network parameters without notifying the end hosts. To eliminate disconnection due to change of the IP or port parameter, the SIP client must proactively register and update its current parameters in this way. The frequency of SIP registration is decisive for the potential disconnection. The current registrations are executed in constant intervals set regardless of the mobility and network properties. The result in mobile environments is wasted resources and overloaded links. A new method for optimization of the registration intervals is presented in this paper. The key idea is to set the registration intervals proportional to the probability for disconnection. The probability density function is built from the past disconnections. The new approach is based on extended Kalman filter. A simulation shows the qualities of the new method, where clear outperformance is demonstrated. The algorithm can be implemented in protocols like IKE, routing protocols, Mobile IP etc.

**Index Terms-** Session Initialization Protocol, SIP, registration, extended Kalman Filter, mobile updates

## I. INTRODUCTION

The telecommunication providers are exchanging their legacy phone networks by VoIP service, which is part of the Next Generation Network (NGN). The SIP [6] is the dominating protocol for delivering VoIP services. This trend leads to a widespread of smart phones with IP interfaces, like 3G, WiFi etc. The mobile lifestyle causes mobile usage of Internet, thus accessing Internet while moving.

The mobile environment is characterized by frequent changes of the access network and consequently, changes of the host's IP. This causes disconnection until reestablishment by the application. Additional cause for disconnection can be an intermediate NAT device. A NAT device exchange the IP or port without notifying of the participants. The hosts behind NAT device are not aware of their public IP and port and their changes. The same effect has statefull firewall, when active session is deleted. A multi-homed hosts change also their source IP depending of the outgoing interface, which can be frequent in mobile environments. A multi-homed host has multiple active IP interfaces, like 3G and WiFi. Every modern smart phone becomes multi-homed, when two IP connections are used simultaneously.

To handle these difficulties, the application must proactively update their IP and port at their communication participants. An end-to-end connection check is made in this way. In SIP, this procedure is called registration [6] and shortly described in section III.

The SIP registration is decisive for the disconnection. Short registration intervals assure short disconnection on the one hand. On the other hand, each registration requires network and host resources. There is a trade off between the disconnection and the resources.

The SIP protocol is not designed to operate in mobile environments where the IP and port change frequently. The SIP registration procedure is executed at constant intervals. The intervals are not dynamically adapted neither to the network properties nor to the mobile behavior. For example: if the host is changing the access network only during the day, there is no sense of frequent registrations at the night. The registration interval can be several minutes up to an hour for typical VoIP deployments. The result of the constant registration is frequent connection drops or overloaded network links in mobile environments. There are wasted resources and poor performance.

## II. TARGETS AND OBJECTIVES

The key idea in this work is to set the registration intervals proportional to the probability of disconnection. The registrations must be frequent when there is a high probability of IP or port change. In contrast, the registrations must be rare when there is low probability of disconnection. The principle is shown at Figure 1, where the registrations are shown at the x-axis and the Probability Density Function (PDF) at the y-axis.

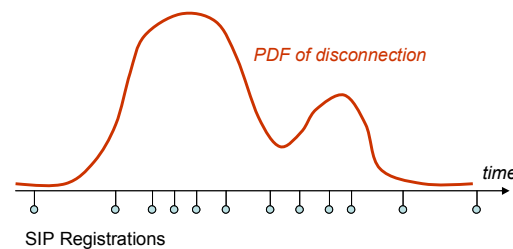


Figure 1 Registrations dependent on the probability for disconnection.

The PDF is constructed using the history of disconnections. The PDF is dynamic depending on the network and mobile host properties. The registration intervals become dynamic. In this way, resources are saved and minimal disconnection is achieved.

## III. SIP REGISTRATION

The SIP registration procedure consists of request-response messages as defined in [6]. There are two

messages for unauthenticated registration. The client sends a registration message and the server accepts or rejects it. The registration consists of four messages when client authenticates. The client sends a registration message. The server replies with nonce value and authentication needed. The client hashes the nonce and its password and sends the result in a new registration request (third message). The server replies with accepted or rejected depending on the authentication result.

The registration has lifetime coded in the expire field or in contact header as parameter. The client must renew the registration before the lifetime expires. The client may suggest a value in the registration request. The registrar replies with definite expiration value, which could be less than the suggested one. The registrar (server) sets the live time of the registration. This is defined in its local policy.

In the new method, the server compares the IP and UDP header of the registrations to find if the parameters between the registrations have changed. If the IP or port has changed between two following registrations, then there is disconnection in this interval. There is Boolean result “true” or “false” if parameters have changed. This Boolean value and the registration interval are the input of the new algorithm, which sets the registration interval.

The new method do not require any specification changes on the SIP protocol. The registrar must get the feed back of the headers and vary the expiration parameter by the registration.

#### IV. CHALLENGES AND DIFFICULTIES

The well-developed classical estimation methods, like Kalman filter [2], cannot be deployed in a straight forward way. They require numerical values typically delivered by some measurements. The measurement consists commonly of signal with added noise. The target is to separate the signal from the noise to obtain a better estimation.

The register procedure delivers Boolean value and not a numerical one as required. The Boolean result gives if the IP address or port has changed or not. The time point of the change is unknown to the mobile host. There are not numerical measurements and therefore, the classical estimation method can not be used directly. The time point of disconnection can be narrowed down to the registration interval. For example: if the registration interval is 300 seconds, than the disconnection occurs somewhere within this 300 sec.

On the one hand, decreasing the registration interval improves the estimation’s precision. On the other hand, it increases the required resources expressed in updates packets and CPU utilization. There is a clear trade off between the intervals size and the resources.

#### V. CONTRIBUTIONS

At first step, a new model for the registration must be created in order to solve the issues mentioned in the previous chapter. The new model must allow the use of extended Kalman filter in the registration procedure. The author suggests new transformation function in order to calculate the time point of registration. Practical contribution is the simulation as proof of concept showing the qualities of the new developed method and comparing the results to constant update intervals.

#### VI. ABSTRACTION AND TERMINOLOGY

An abstraction model must be created before going deeper in method description. The generalization allows deploying the suggested algorithm to other protocols.

A change of the IP addresses and UDP port is called network *event*. The *movement* of the host means also change the IP or port underling the process of change. The registration procedure returns *true* when the host has not moved between the previous and current registration, thus IP or port has not changed. The result *false* means the host has moved. The time point of the event is denoted as *Event Time Point* (ETP), see Figure 2. The intervals between the Event Time Points are called *Event Intervals* (EI). The points at which the update procedure is executed are called *Update Time Points* (UTP). The interval between two following UTPs is *Update Interval* (UI). The UTP is the time point when the Boolean result is received.

The *Disconnection Interval* (DI) is the interval between the Event Time Point (ETP) and the following Update Time Point (UTP), see Figure 2. All send packets in DI get lost. The host has moved but the participants are not notified. Consequently, they use malicious IP or port.

The *Maximal Disconnection Interval* (MDI) is the maximal disconnection tolerable by the application. The value depends of the application type and can be some milliseconds for real time media. For data services can reach several minutes, like e-mail. The DI must not exceed the MDI.

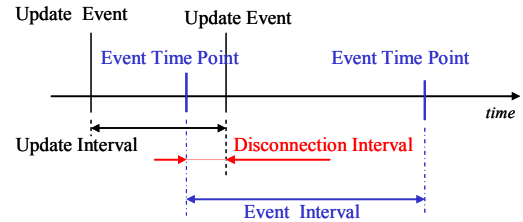


Figure 2 Abstraction

##### A. Time points relative to filter cycle

A filter has a prediction and an update phases building filter’s cycles. The prediction (prior estimation) for the ETP is made at the prediction phase. In the update phase, the uüdates are executed until obtaining one event. Then the PDF is posterior estimated. After it, the filter cycle start again with prediction phase. In order not to work with absolute times, all Time Points are set relative to the beginning of the filter cycle. In every filter cycle, there is one ETP and multiple UTPs. The cycle begins with UTP.

#### VII. REGISTRATION OPTIMIZATION

The Disconnection Interval (DI) must be minimized. Unfortunately, the host is not aware of the ETP as already mentioned. The idea is to minimize the Update Interval (UI) containing the ETP, which is known to the mobile application. The DI is smaller than UI with ETP, thus minimizing UI with ETP minimizes the DI at the same time.

To save recourses the UI without ETP must be maximized, thus the MDI. Event the probability for ETP is extremely low, the updates must be executed at lest MDI. These properties of UI are achieved by setting the UTP anti-proportional to the PDF of ETP (section II).

The extended Kalman filter [3] delivers single numerical value of the predicted ETP and the semi standard deviation. These are not sufficient to construct the PDF of ETP. For this reason, it is assumed that the PDF of ETP has Normal distribution with mean at the predicted ETP. The standard deviation of the Normal distribution is set proportional to the result deviation of the extended Kalman filter.

### VIII. MODEL FOR EXTENDED KALMAN FILTER.

At least two models are required to deploy Kalman filter [3, 1]. The first one describes the time evaluation of the system in the context of the variable of interest. It expresses the natural properties of the process, for example the velocity of a free-falling body. The second describes the measurement. The equation gives the relation between the variable of interest and the result of the measurement.

#### A. Natural properties of the Event Intervals

The natural properties of the Event Intervals (EIs) distribution in time are unknown, thus movements of the Mobile Node cannot be defined at this stage. Unfortunately, the Kalman filter cannot be implemented without the evaluation description as already noticed. To overcome, we try to define a general equation which does not restricts the movements of a mobile host.

A random walk is assumed for the EI time evaluation. Let  $x_k$  be the Event Interval at step  $k$ . Then  $x_k = Ax_{k-1} + w$ , where  $w = N(0, \sigma)$  is white noise and vector  $A$  called transformation matrix. The vector  $A$  is set to 1 (one), thus there is no change transformation. The standard deviation is calculated recursively from the posterior Event Intervals (already observed measurements) at every filter cycle.

#### B. Measurement model

The second model represents the measurement of the Event Time Point (ETP), thus the Measurement Time Point (MTP). The exact measurement of the ETP is not possible. The ETP can only be narrowed down to the Update Interval (UI) with the ETP but not to exact numerical value.

The assumption is made that the MTP is in the middle of the UI with ETP. The absolute error  $I$  is minimized in this point. Let  $y_k$  be the Measured Interval of the  $k$  prediction/update cycle. The measurement function is  $f()$ , thus  $y_k = f(x_k)$ .

The MTP depends on the UI with ETP. The UI depends on the predicted (prior estimated) EI. The UTPs are calculated with transformation function. The function transforms the constant update intervals equal MDI to intervals anti-proportional to the PDF of ETP according chapter II. The transformation function is denoted with  $T()$  and defined in chapter X. The MDI is  $x_{\max}$ . The measurement function is then:

$$y_k = f(x_k) = \frac{1}{2} T(x_{\max} \cdot T^{-1}(x_k - \hat{x}_k) \bmod x_{\max}) + \frac{1}{2} T((x_{\max} + 1) \cdot T^{-1}(x_k - \hat{x}_k) \bmod x_{\max})$$

The extended Kalman filter [3, 5] requires the standard deviation of the measurement (Gaussian distributed

system). Without this definition it is impossible to derivate the filter. Unfortunately, the standard deviation of measurement function cannot be found analytically since the function is interrupted. It can not be calculated in mathematical sense.

Decisive property is that input value of the Kalman filter must be proportional to the standard deviation. A standard deviation is small for values with high probability and large for values with low probability. A variable with the same characteristic is UT with ETP used instead of the standard deviation. In this way, the extended Kalman filter will deliver correct prediction values, but numerically wrong standard deviation. The result standard deviation will be proportional to the real one to some degree. It is denoted as semi standard deviation. It is for the  $k$  the cycle:

$$v_k = T((x_{\max} + 1) \cdot T^{-1}(x_k - \hat{x}_k) \bmod x_{\max}) - T(x_{\max} \cdot T^{-1}(x_k - \hat{x}_k) \bmod x_{\max})$$

### IX. EXTENDED KALMAN FILTER

The Bayesian inference expresses the relation between belief in a hypothesis prior and posterior an evidence [1]. The equations are briefly repeated here. Let the posterior PDF function at filter cycle  $k-1$  be  $p(x_{k-1}|y_{1:k-1})$ . The priory PDF is calculated in the prediction phase:

$$p(x_k|y_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|y_{1:k-1})dx_{k-1}$$

The posterior PDF is calculated after the measurement becomes available:

$$p(x_k|y_{1:k}) = \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{p(y_k|y_{1:k-1})}, \text{ where}$$

$$p(y_k|y_{1:k-1}) = \int p(y_k|x_k)p(x_k|y_{1:k-1})dx_k$$

This is the optimal solution and has only a theoretical and general character since the posterior PDF cannot be calculated analytically in all cases.

When the models are linear and the posterior PDF function is normally distributed, the Kalman filter [2] solves the equations analytically. The model developed in chapter VIII is non-linear, so the Kalman filter cannot be used. The extended Kalman Filter [3] deals with non-linear models and it will be implemented here. The extended Kalman filter (EKF) approximates the system to linear using truncated Taylor series and then uses the analytical Kalman filter.

The predicted value of Event Interval (EI) at filter cycle  $k$  is  $\hat{x}_k^-$ , called prior estimation. The minus sign at superscript expresses that the measurement at step  $k$  is not available, thus it is a prediction is done at state  $k-1$ . The estimation of EI after the measurement becomes available is  $\hat{x}_k^+$ . The plus superscript denotes that the measurement at  $k$  filter cycle is available. The filter derivation can be found in [3] and can be denoted as:

#### Prediction:

$$\hat{x}_k^- = A.x_{k-1}$$

$$P_k^- = A.P_{k-1}^{-1}A^T + \sigma$$

### Measurement update:

$$\hat{x}_k^+ = \hat{x}_k^- + K(f(x_k, \hat{x}_k^-) - f'(x_k, \hat{x}_k^-))$$

$$K = \frac{P_k^- f'(\hat{x}_k^-)}{f'(\hat{x}_k^-, \hat{x}_k^-) P_k^- f'(\hat{x}_k^-, \hat{x}_k^-)^T + v}$$

$$P_k^+ = (I - K f') P_k^-$$

### X. TRANSFORMATION FUNCTION

The length of the Update Intervals must be inversely proportional to the PDF of ETP, see Figure 1. Furthermore, the Update Intervals must not exceed the Maximal Disconnection Interval (MDI). The PDF of ETP (Normal distribution) and the length of the UI are shown at Figure 3.

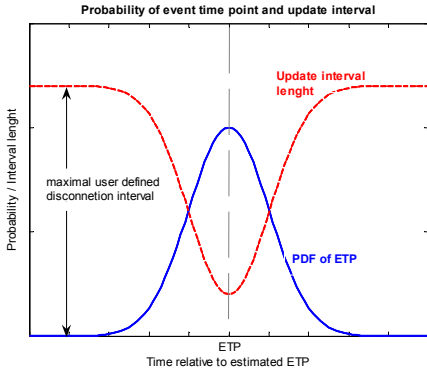


Figure 3 PDF of EI and the UI

The idea is to develop a transformation function which calculates the UTP from constant update points. The function transforms the constant intervals of update to non-linear ones satisfying the condition of section II. The function has parameters the predicted ETP and the MDI. The transformation function is very suitable for the implementation.

The Update Intervals according to Figure 3 can be defined by:

$$L(t) = a - b \cdot N(t, \mu, \sigma),$$

where  $L(t)$  is the length of the Update Interval at time  $t$ . The  $N(t, \mu, \sigma)$  denotes the Normal distribution (PDF of ETP) at point  $t$  with mean  $\mu$  and standard deviation  $\sigma$ . The constant  $a$  is the maximal disconnection interval defined by the application. The  $b$  defines the pitch of the bell curve.

Let  $T()$  be transformation function. Then  $T(x_{i+1}) - T(x_i) = L(x_i)$ , where  $x_i, x_{i+1}, \dots$  are the time points of constant intervals. This key equation can be transformed:

$$T(x_{i+1}) - T(x_i) = L(x_i) \quad x_{i+1} - x_i \rightarrow 0$$

$$\frac{\partial T(x)}{\partial x} = L(x) = a - b \cdot N(x, \mu, \sigma)_i$$

$$T(x) = ax - u \cdot \text{erf}\left(\frac{x}{k}\right) + C,$$

where

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$

is the error function. The  $a, u, k$  and  $C$  are constants.

### XI. APPROXIMATED TRANSFORMATION FUNCTION

The transformation function has only a theoretical character. For practical implementation, the straight and inverse transformation are required. The derived function is a complementary and its inverse  $T^{-1}(x)$  cannot be found analytically. The idea is to find approximation of this ideal function, which has inverse function.

The function is approximated to the sum of functions in different intervals which have inverse transformation:

$$T'(x) = \begin{cases} u \cdot \text{erf}(x/k) + C_1 & x \in [-l, +l] \\ ax + C_2 & x \in (-\infty, -l) \cup (l, \infty) \end{cases}$$

Then the inverse function is:

$$T'^{-1}(x) = \begin{cases} k \cdot \text{erf}^{-1}((x - C_1)/u) & x \in [T'(-l), T'(l)] \\ (x - C_2)/a & x \in (-\infty, T'(-l)) \cup (T'(l), \infty) \end{cases}$$

Splitting the function in this way leads to a sub optimal solution, but which delivers very good results.

#### A. Coefficients of the transformation function

The coefficients of the approximated function must be derived in order to use the function. The coefficients are calculated from conditions, which must be. There are not sufficient objective conditions and therefore, subjective ones are used additionally. They are listed here:

1) The first condition is that the transformation function and its inverse must be continuous (not interrupted). This gives a relation between the two functions at the crossover point  $l$ :

$$u \cdot \text{erf}(l/k) + C_1 = al + C_2$$

2) The predicted Event Interval is reasonable to be equal to  $l$ , thus  $l = P_k^+$ , where  $P_k^+$  is the predicted Event Interval. The  $k$  index denotes the prediction cycle.

3) The linear update interval is the Maximum Disconnection Interval defined by the application. This value is chosen to simplify the calculation.

4) The update interval at the estimation ETP is zero, thus  $C_1 = 0$ .

5) The output of the transformation function (the non-linear update interval) must also not exceed Maximum Disconnection Interval, thus:

$$\frac{\partial}{\partial x} u \cdot \text{erf}(x_k/k) \geq 1 \quad x_k \in (-P_k^+, P_k^+)$$

and equal to 1 outside this interval:

$$\frac{\partial(ax + C_2)}{\partial x} = 1 \Rightarrow a = 1$$

6) A subjective condition says that the  $\text{erf}$  function reaches 95% of its maximum at the cross point between the components functions, thus:

$$\text{erf}(P_k^+/k) = 0.95 \Rightarrow P_k^+ + C_2 = 0.95u$$

7) The transformation function has its minimum at the predicted ETP, where the probability is highest. The

predicted ETP is set as point zero and all points in relation to it.

These seven conditions deliver the solution for the coefficients of the transformation function. The following system equations are obtained by setting all of the conditions together:

$$\begin{cases} \frac{\partial}{\partial x} u \cdot \text{erf}(x_k / k) \geq 1, & x_k \in [-P_k^+, P_k^+] \\ u \cdot \text{erf}(P_k^+ / k) = P_k^+ + C_2 \\ P_k^+ + C_2 = 0.95u \end{cases}$$

A partial solution of the system is:

$$k = \frac{P_k^+}{\text{erf}^{-1}(0.95)}, \quad u = \frac{k\sqrt{\pi}}{2} e^{2\text{erf}^{-1}(0.95)^2},$$

$$C_2 = 0.95 \cdot \frac{\sqrt{\pi}}{2} e^{\text{erf}^{-1}(0.95)^2} - P_k^+$$

The transformation function is:

$$T'(x) = \begin{cases} u \cdot \text{erf}(x / k) & x \in [-P_k^+, P_k^+] \\ x + C_2 & x \in (-\infty, -P_k^+) \cup (P_k^+, \infty) \end{cases}$$

The inverse function is:

$$T'^{-1}(x) = \begin{cases} k \cdot \text{erf}^{-1}(x / u) & x \in [T'(-P_k^+), T'(P_k^+)] \\ (x - C_2) & x \in (-\infty, T'(-P_k^+)) \cup (T'(P_k^+), \infty) \end{cases}$$

An example of the approximated function is shown at Figure 4. The points are set relative to the ETP. The function obeys the required conditions: it becomes linear at infinite and is not linear (decreasing) for values near to the ETP.

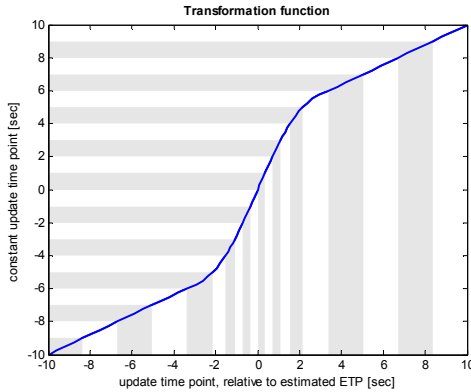


Figure 4 Transformation function

## XII. SIMULATION

The main target of the simulation is to achieve proof of concept and to show the qualities of the new method. The results are compared with the updates at constant interval in a fair way. The fair way means that the same number of updates is used for the simulation time. The two methods use the same total number of updates, input values and simulation time. The better method achieves less disconnection, thus better estimation. The constant updates is commonly the only used method, so it is compared the

current protocols to the new one. Furthermore, the method can be compared to [8, 7]

The constant parameters in the simulation are chosen to represent real cases, where small disconnection is required and the mobile host moves fast. The following parameters are used by the experiments: The application requires 5 sec of Maximum Disconnection Interval (MDI). The simulation is made with 5000 samples, thus there were 5000 prediction cycles.

The simulation shows the clearly the advantages of the new method. Here some representative cases are shown in the following section.

### A. Non linear EI with white noise

The Event Intervals are generated with recursion, where the next value depends of the previous one:

$$aEI_k = \frac{aEI_{k-1}}{2} + \frac{25r_{k-1}}{1 + r_{k-1}^2} + 8\cos(1.2k) + N(200,20)$$

The  $k$  denotes the EI (filter cycle). The EI and the estimated EI are shown at Figure 5. The results at Figure 6 are presented in histogram form. There is a very good performance of the algorithm.

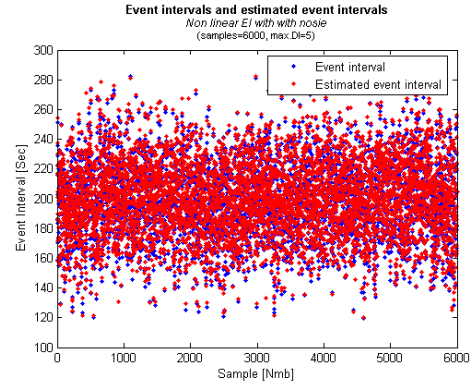


Figure 5 Non-linear data, EI and estimated EI

The abbreviations at the diagram are:

- A dotted black line shows the mean Update Interval (UI), thus mean of maximal disconnection.
- A dashed black line shows the maximal Disconnection Intervals (DIs) by constant update interval see Variable const UI.
- A red dashed line marks the maximum Disconnection Interval (error) in this simulation
- A grey dotted line shows the user defined maximum disconnection interval, thus MDI.

Certain very important qualitative values are shown in the bottom-right corner of the histogram. Their definitions are:

- {mean DI} is mean of all Disconnection Intervals.
- {mean UI} is the mean of the Update Intervals containing the Event Time Point.
- {max UI} is the maximum of all Update Intervals during the simulation.
- {max user def DI} is the Maximum Disconnection Interval.
- {const UI}. This is the maximum disconnection achieved by constant update method using the same resources as the new method.



- {const UI mean DI } is the mean of all DIs by constant UI with the same resources.
- {const UI / mean UI} gives the ratio between the const UI and mean UI. This is a direct comparison of the performance of the algorithm in a simple way.
- {LU < const UI} is the number in percentage of UI with ETP smaller than the maximum disconnection by constant interval (const UI).

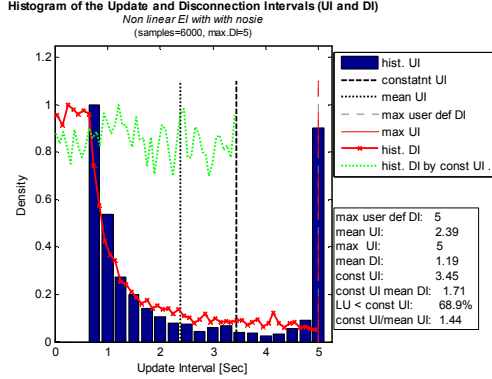


Figure 6 Non-linear data, histogram of the results

The algorithm has significant outperformance over the constant intervals. The mean UI with ETP by the new methods is 1.44 times smaller than the constant updates. Over 69 % of UI with ETP are smaller than by the constant updates. The MDI is reached in small number of cases.

#### B. Sinus based EI with white noise

For this simulation is a sinus based signal used with white noise:

$$aEI_k = \sigma \sin\left(4\pi \frac{k}{S}\right) + \frac{7}{10} \sigma \sin\left(\frac{3}{2} \pi \frac{k}{S}\right) + \sin c\left(\frac{k}{S}\right) + N\left(500, \frac{\sigma}{2}\right)$$

The variable S indicates the total number of samples, S=5000. The k is the index of the calculated sample. The EI and estimated EI are presented at Figure 8. The results are presented at Figure 9.

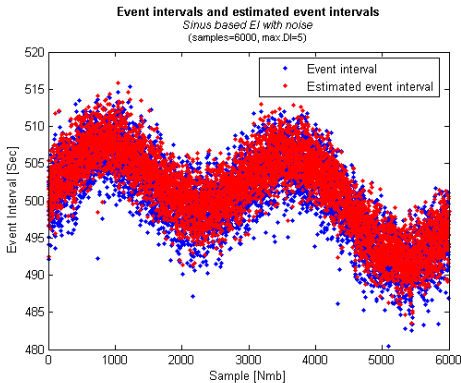


Figure 8 Sinus based data, EI and estimated EI

The algorithm shows impressive performance. The mean UI is 2.68 times better than by the constant interval. More than 92 % of the EI were better predicted and handled by the new method. The UIs with ETP by sample are shown at the Figure 7. Interesting fact is that the UI with ETP follows the data cure.

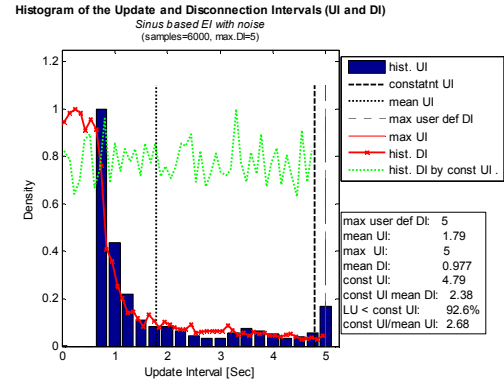


Figure 9 Sinus based data, histogram of the results

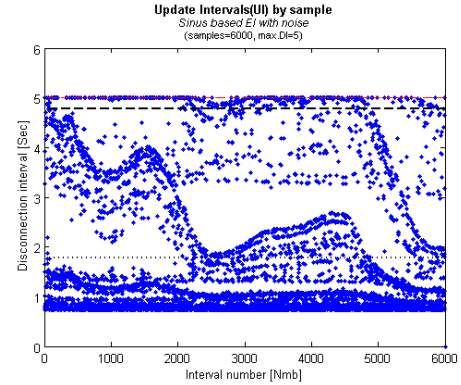


Figure 7 Sinus based data, UIs with ETP

### XIII. CONCLUSION

The new method has a significant outperformance comparatively to the current constant update intervals method. It makes use of the history of the past event to adjust the update frequency. It can be deployed in multiple protocols having an update procedure, which can be abstracted to this one. The algorithm has the advantage against similar methods [7, 8], that it does not require optimization of multiple parameters. Practically, only the MDI values must be defined by the application.

### REFERENCES

- [1] Sanjeev Arulampalam, Simon Maskell, Neil Gordon, Tim Clapp, "A Tutorial on Particle Filters for On-line Non-linear/Non-Gaussian Bayesian Tracking (2001), IEEE Transactions on Signal Processing
- [2] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems", Copyright © 1960 by ASME
- [3] Arthur Gelb, "Applied Optimal Estimation", MIT Press, 1974, ISBN: 0262570483
- [4] N. Wiener, "The Extrapolation, Interpolation and Smoothing of Stationary Time Series," John Wiley & Sons, Inc., New York, N.Y., 1949.
- [5] Peter S Maybeck, "Stochastic Models, Estimation and Control.", Mathematics in Science and Engineering, June 1979
- [6] RFC 3261 "SIP: Session Initiation Protocol", June 2002
- [7] Vesselin Tzvetkov, "Optimization of update intervals in Dead-Peer-Detection using adaptive Fuzzy Logic", IEEE AINA, 2007
- [8] Vesselin Tzvetkov, "Optimization of mobile updates using Particle filter", IEEE ChinaCom 2008