Using Sub-optimal Kalman Filtering for Anomaly Detection in Networks

Joseph Ndong,
Department of Mathematics and Computer Science,
University Cheikh Anta Diop of Dakar, Sénégal
E-mail: joseph.ndong@ucad.edu.sn

***Abstract*—Possibility theory can be used as a suitable framework to build a normal behavioral model for an anomaly detector. Based on linear and/or nonlinear systems, sub-optimal filtering approaches based on the Extended Kalman Filter and the Unscented Kalman Filter are calibrated for entropy reduction and could be a good basis to find a suitable model to build a *decision variable* where, a decision process can be applied to identify anomalous events. Sophisticated fuzzy clustering algorithms can be used to find a set of clusters built on the decision variable, where anomalies might happen inside a few of them. To achieve an efficient detection step, a robust decision scheme is built, by means of possibility distributions, to separate the clusters into normal and abnormal spaces. We had studied the false alarm rate *vs.* detection rate trade-off by means of ROC (Receiver Operating Characteristic) curves to show the results. We validate the approach over different realistic network traffic.**

***Index Terms*—Extended Kalman Filter, Unscented Kalman Filter, Fuzzy Clustering, Anomaly Detection, Possibility theory.**

I. Introduction

Recently, some works, related to anomaly detection in communication networks, have been concentrated on Linear Kalman filtering [16], [15], [14]. However, despite its strength, the linear Kalman filter runs well with hard difficulties. Generally, the innovation process is expected to be a Gaussian white noise. However, in practice, this is hardly the case as frequently the observed signals are non gaussian/nonlinear themselves. In this work we show that a decision variable can be made from the innovation processes and organize in clusters where anomalies might be detected. Another difficulty is related to the calibration of the input matrices of the linear Kalman filter. Another problem is related to the choice of the model type: linear or nonlinear. This is generally a challenging task to build a good system for anomaly detection.

Our hope in this paper is to show that the sub-optimal algorithms based on EKF and UKF can be view as valuable and alternative tool for anomaly detection, in case when the state and measurement processes are linear. We believe that one should build a bank of different filters and perform a comparative study which could have as a final hope to find out the best model.

A. Normal behavior modeling

The framework of EKF and UKF is based on the following difference equations:

$$\begin{cases} x_{t+1} = f(x_t) + w_t \\ y_t = h(x_t) + v_t \end{cases} \tag{1}$$

where $x_t \in \mathbb{R}^n$ and $y_t \in \mathbb{R}^m$ are multi-dimensional vectors representing respectively the system state and the measurement. The system is assumed to be excited by an unknown process noise $w_t \sim N(O, Q_t)$ and the measurement are disturbed by unknown measurement noise $v_t \sim N(O, R_t)$.

B. How to build the Decision Variable ?

The *decision variable* is built using the multi-dimensional innovation process obtained as output of the filters. The ***one-dimensional*** decision variable (DV) process is obtained by applying the formulas:

$$decisionvariable = e(t)^T V e(t) \tag{2}$$

where the matrix V (obtained as output of each Kalman filter) is the inverse of the **variance** of the multi-dimensional innovation process $e(t)$, T denotes the transpose.

1) Extended Kalman Filter: We use the first order EKF which is based on linear quadratic approximations with a gaussian: $p(x_t|y_{1:t}) = N(x_t|m_t, P_t)$. Due to lack of space, we give the necessary equations needed to run the EKF [3], [2]. The filter runs into two steps as a predictor-corrector algorithm:

- *prediction*:

$$\begin{cases} m_t^- = f(m_{t-1}) \\ P_t^- = F_x(m_{t-1}) P_{t-1} F_x^T(m_{t-1}) + Q_{t-1} \end{cases} \tag{3}$$

- *correction (update)*:

$$\begin{cases} v_t = y_t - h(m_t^-) \\ S_t = H_x(m_t^-) P_t^- H_x^T(m_t^-) + R_t \\ K_t = P_t^- H_x^T(m_t^-) S_t^{-1} \\ m_t = m_t^- + K_t v_t \\ P_t = P_t^- - K_t S_t K_t^T \end{cases} \tag{4}$$

where the matrices $F_x(m)$ and $H_x(m)$ are the Jacobians of the functions f and h, with elements:

$$[F_x(m)]_{jj'}^{t-1} = \frac{\partial f_j(x,t-1)}{\partial x'_j}|_{x=m} \tag{5}$$

$$[H_x(m)]^t_{jj'}=\frac{\partial h_j(x,t)}{\partial x'_j}|_{x=m} \tag{6}$$

2) Unscented Kalman Filter: In the following, we derive the prediction and update equations for the UKF based on UT (Unscented Transformation) [11].

- *prediction*: computed the predicted state mean m_t^- and the predicted covariance P_t^- as:

$$\begin{cases} X_{t-1}=[m_{t-1}\ldots m_{t-1}]+\sqrt{c}[0\sqrt{P_{t-1}}-\sqrt{P_{t-1}}] \\ \hat{X}_t = f(X_{t-1}) \\ m_t^- = \hat{X}_t(w_m) \\ P_t^- = \hat{X}_t W[\hat{X}_t]^T+Q_{t-1} \end{cases} \tag{7}$$

- *correction (update)*: Compute the predicted mean μ_t and covariance of the measurement S_t, and the cross-covariance of the state and measurement C_t:

$$\begin{cases} X_t^-=[m_t^-\ldots m_t^-]+\sqrt{c}[0\sqrt{P_t^-}-\sqrt{P_t^-}] \\ Y_t^-= h(X_t^-) \\ \mu_t = Y_t^- w_m \\ S_t = Y_t^- W[Y_t^-]^T+R_t \\ C_t = X_t^- W[Y_t^-]^T \end{cases} \tag{8}$$

One can then compute the filter gain K_t and the updated state mean m_t and covariance P_t as:

$$\begin{cases} K_t= C_t S_t^{-1} \\ m_t=m_t^-+K_t[y_t-\mu_t] \\ P_t= P_t^- -K_t S_t K_t^T \end{cases} \tag{9}$$

II. How to build the normal subspace ?

Once the number of clusters found, we run a two-step approach to build the normal space formed by some clusters, the remaining labeled as abnormal. First, since we do not have any a priori knowledge of the clusters distribution, we affect to each cluster a degree of possibility by means of possibility distribution. The second step try to extract the "normal" clusters. We believe that the degree of normalcy of a cluster depends only of the degree of normalcy of the data inside the cluster itself. We use the memberships from the clustering operation to calculate the degrees of possibility of the data themselves. A thorough analysis of the second king of possibility distributions makes us find a threshold to apply to a cluster's degree of possibility to decide if it is normal.

A. Clustering operation

Here, for the purpose of efficiency and comparison , we perform the clustering operation with five algorithms, namely: k-means, k-medoid, fuzzy c-means (FCM) and Gustafson-Kessel (GK)algorithms.

1) K-means and K-medoid clustering algorithms: With an $N\times n$ dimensional data set, K-means allocates each data point to one of c clusters to minimize the within-cluster sum of squares defined as:

$$\sum_{i=1}^{c}\sum_{k\in A_i}||X_k-v_i||_2, \tag{10}$$

where A_i is a set of data points in the $i-th$ cluster and v_i is the mean for that points over cluster i. In K-means clustering v_i is called the cluster prototypes, i.e the cluster centers; it is defined by:

$$v_i=\frac{\sum_{k=1}^{N_i}x_k}{N_i}, x_k\in A_i, \tag{11}$$

where N_i is the number of data points in A_i.

In K-medoid algorithm, the cluster centers are the nearest data points to the mean in one cluster $V=\{v_i\in X|1\leq i\leq c\}$.

2) Fuzzy C-means clustering algorithm: The Fuzzy C-means clustering algorithm is based on the minimization of an objective function called C-means functional. It is defined by Dunn as:

$$J(X;U,V)=\sum_{i=1}^{c}\sum_{k=1}^{N}(\mu_{ik})^m||x_k-v_i||_A^2, \tag{12}$$

where $V=[v_1,v_2,\ldots,v_c]$, $v_i\in\mathbb{R}^n$ is a vector of cluster prototypes, which have to be determined, and the quantity:

$$D_{ikA}^2=||x_k-v_i||_A^2=(x_k-v_i)^TA(x_k-v_i), \tag{13}$$

is a squared inner-product distance norm.
The equation Eq. 12 is a measure of the total variance of x_k from v_i. The minimization of this quantity can be done with the popular Picard iteration trough the first-order conditions for stationary process of the objective function. These stationary points can be found by means of Lagrange multipliers as:

$$\overline{J}(X;U,V,\lambda)=\sum_{i=1}^{c}\sum_{k=1}^{N}(\mu_{ik})^mD_{ikA}^2+\sum_{k=1}^{N}\lambda_k(\sum_{i=1}^{c}\mu_{ik}-1), \tag{14}$$

and by setting the gradient of $\overline{J}$ with respect to U, V and λ to zero. If $D_{ikA}^2>0$, $\forall i,k$ and $m>1$, then $(U,V)\in M_{fc}\times\mathbb{R}^{n\times c}$ may minimize Eq. 12 only if

$$\mu_{ik}=\frac{1}{\sum_{j=1}^{c}(D_{ikA}/D_{jkA}^{2/(m-1)}}, 1\leq i\leq c, 1\leq k\leq N, \tag{15}$$

and

$$v_i=\frac{\sum_{k=1}^{N}\mu_{ik}^m x_k}{\sum_{k=1}^{N}\mu_{ik}^m}, 1\leq i\leq c, \tag{16}$$

3) Gustafson-Kessel clustering algorithm: Gustafson and Kessel extended the standard fuzzy c-means algorithm by employing an adaptive distance norm, in order to detect clusters of different geometrical shapes in one data set. Each cluster has its own norm-inducing matrix A_i, which yields the following inner-product norm:

$$D^2_{ikA} = (x_k - v_i)^T A_i (x_k - v_i), 1 \leq i \leq c, 1 \leq k \leq N. \quad (17)$$

The matrices A_i are used as optimization variables in the c-means functional, allowing each cluster to adapt the distance norm to the local topological structure of the data. If A denotes a c-tuple of the norm-inducing matrices $A = (A_1, A_2, \ldots, A_c)$, then the objective functional of the GK algorithm is defined by:

$$J(X; U, V, A) = \sum_{i=1}^{c} \sum_{k=1}^{N} (\mu_{ik})^m D^2_{ikA_i}. \quad (18)$$

We implemented in Matlab the numerically robust algorithm described in [12].

To find the appropriate number of clusters, one can cluster data for different values of $c \in \{2, 3, \ldots, c_{max}\}$, and using *validity measures* to assess the goodness of the obtained partitions. Different scalar validity measures have been proposed in the literature, none of them is perfect by itself, therefor we used several indexes in our work for the hope of comparison.

B. Optimum number of clusters

We use the following validity measures [7] as a tool to determine the optimum number of classes in our clustering operation.

- *Partition Coefficient* (PC): measures the amount of "overlapping" between clusters. It is defined by Bezdek [6] as follows:

$$PC(c) = \frac{1}{N} \sum_{i=1}^{c} \sum_{k=1}^{N} (\mu_{ij})^2, \quad (19)$$

  where μ_{ij} is the membership of data point j in cluster i. The optimal number of cluster is at the maximum value.
- *Classification Entropy* (CE): it measures the fuzzyness of the cluster partition only, which is similar to the Partition Coefficient. It is defined as:

$$CE(c) = -\frac{1}{N} \sum_{i=1}^{c} \sum_{k=1}^{N} (\mu_{ij}) log(\mu_{ij}), \quad (20)$$

- *Partition Index* (SC): It is defined as [7]:

$$SC(c) = \sum_{i=1}^{c} \frac{\sum_{j=1}^{N} (\mu_{ij})^m ||x_j - v_i||^2}{N_i \sum_{k=1}^{c} ||v_k - v_i||^2}, \quad (21)$$

  SC is useful when comparing different partitions having equal number of clusters. A lower value of SC indicates a better partition.
- *Separation Index* (S): on the contrary of partition index (SC), the separation index uses a minimum-distance separation for partition validity [7]. It is defined as:

$$S(c) = \frac{\sum_{i=1}^{c} \sum_{j=1}^{N} (\mu_{ij})^m ||x_j - v_i||^2}{N_i min_{i,k} ||v_k - v_i||^2}, \quad (22)$$

- *Xie and Beni's Index* (XB): It is defined as [10]:

$$XB(c) = \frac{\sum_{i=1}^{c} \sum_{j=1}^{N} (\mu_{ij})^m ||x_j - v_i||^2}{N_i min_{i,j} ||x_j - v_i||^2}, \quad (23)$$

  The optimal number of clusters should minimize the value of the index.
- *Dunn's Index* (DI): It is defined as:

$$DI(c) = min_{i \in c} min_{j \in c, j \neq i} \frac{min_{x \in C_i, y \in C_j d(x,y)}}{max_{k \in c} \{max_{x,y \in C} d(x,y)\}} \quad (24)$$

  The maximum of DI gives the optimum number of clusters.

III. Building normal space with possibility theory

The normal space is built into a two step-wise approach is necessary. Dubois and Prade' s procedure, [1], produces the most specific possibility distribution among the ones dominating a given probability distribution. In this paper, this method is generalized to the case where the probabilities (of generating the clusters) are **unknown**. It is proposed to characterize the probabilities of generating the different clusters by *simultaneous confidence intervals* with a given confidence level $1 - \alpha$. So a procedure for constructing a possibility distribution is described, insuring that the resulting possibility distribution will dominate the true probability distribution in at least $100(1 - \alpha)$ of the cases.

In a second phase, we will also use a procedure to computing possibilities for data points inside a cluster in order to know if this cluster is normal or abnormal. This can be achieved by means of memberships of the data points, i.e the probability of generating the data sample.

In the following, we suppose that there's K well-formed clusters. We consider the parameter vector $p = (p_1, p_2, \ldots, p_K)$ of probabilities characterizing the unknown probability distributions of a random variable X on $\Omega = \{\omega_1, \ldots, \omega_K\}$. Let n_k denotes the number of observations of cluster k in a sample of size N. Then, the random vector $n = (n_1, \ldots, n_K)$ can be considered as a *multinomial* distribution with parameter p. A confidence region for p at level $1 - \alpha$ can be computed using *simultaneous confidence intervals* as described in [4]. Such a confidence region can be considered as a set of probability distributions.

A consistency principle between probability and possibility was first stated by Zadeh, [5] in an unformal way: "*what is probable should be possible*". This requirement is translated via the inequality:

$$P(A) \leq \Pi(A) \qquad \forall A \subseteq \Omega \quad (25)$$

where P and Π are, respectively, a probability and a possibility measure on a domain $\Omega = \{\omega_1, \ldots, \omega_K\}$. In this case, Π

is said to dominate P. Transforming a probability measure into a possibilistic one then amounts to choosing a possibility measure in the set $\Im(P)$ of possibility measures dominating P. This should be done, by adding a strong order preservation constraint, which ensures the preservation of the shape of the distribution:

$$p_i < p_j \Leftrightarrow \pi_i < \pi_j \qquad \forall i, j \in \{1, \ldots, K\}, \tag{26}$$

where $p_i = P(\{\omega_i\})$ and $\pi_i = \Pi(\{\omega_i\})$, $\forall i \in \{1, \ldots, K\}$. It is possible to search for the most specific possibility distribution verifying (25) and (26) (a possibility distribution π is more specific than $\pi^{'}$ if $\pi \leq \pi^{'}, \forall i$). The solution of this problem exists, is unique and can be described as follows. One can define a strict partial order P on Ω represented by a set of compatible linear extensions $\Lambda(\mathsf{P}) = \{l_u, u = 1, L\}$. To each possible linear order l_u , one can associate a permutation σ_u of the set $\{1, \ldots, K\}$ such that:

$$\sigma_u(i) < \sigma_u(j) \Leftrightarrow (\omega_{\sigma_u(i)}, \omega_{\sigma_u(j)}) \in l_u, \tag{27}$$

The most specific possibility distribution, compatible with $p = (p_1, p_2, \ldots, p_K)$ can then be obtained by taking the maximum over all possible permutations:

$$\pi_i = \max_{u=1,L} \sum_{\{j|\sigma_u^{-1}(j) \leq \sigma_u^{-1}(i)\}} p_j \tag{28}$$

A problem arises for calculating the possibilities for the clusters themselves, since we do not know the probabilities p. A solution can consist to build *confidence intervals* for each cluster ω_i. In interval estimation, a scalar population parameter is typically estimated as a range of possible values, namely a confidence interval, with a given confidence level $1 - \alpha$.

To construct confidence intervals for multinomial proportions, it is possible to find simultaneous confidence intervals with a joint confidence level $1 - \alpha$. The method attempts to find a confidence region $\mathcal{C}_n$ in the parameter space $p = (p_1, \ldots, p_K) \in [0; 1]^K | \sum_{i=1}^{K} p_i = 1$ as the Cartesian product of K intervals$[p_1^-, p_1^+]$...$[p_K^-, p_K^+]$ such that we can estimate the coverage probability with:

$$\mathbb{P}(p \in \mathcal{C}_n) \geq 1 - \alpha \tag{29}$$

At this moment, we can use the Goodman, [13] formulation in a series of derivations to solve the problem of constructing the simultaneous confidence intervals. Let

$$A = \chi^2(1 - \alpha/K, 1) + N \tag{30}$$

$$B_i = \chi^2(1 - \alpha/K, 1) + 2n_i, \tag{31}$$

$$C_i = \frac{n_i^2}{N}, \tag{32}$$

$$\Delta_i = B_i^2 - 4AC_i, \tag{33}$$

Finally, the bounds of the confidence intervals are defined as follows:

$$[p_i^-, p_i^+] = \left[\frac{B_i - \Delta_i^{\frac{1}{2}}}{2A}, \frac{B_i + \Delta_i^{\frac{1}{2}}}{2A} \right] \tag{34}$$

It is now possible, based on these above interval-valued probabilities, to compute the most possibility distributions (degrees of the different clusters) dominating any particular probability measure. Let P denotes the partial order induced by the intervals $[p_i] = [p_i^-, p_i^+]$:

$$(\omega_i, \omega_j) \in \mathsf{P} \Leftrightarrow p_i^+ < p_j^- \tag{35}$$

As explained above, this partial order may be represented by the set of its compatible linear extensions $\Lambda(\mathsf{P}) = \{l_u, u = 1, L\}$, or equivalently, by the set of the corresponding permutations$\{\sigma_u, u = 1, L\}$. Then for each possible permutation σ_u associated to each linear order in $\Lambda(\mathsf{P})$, and each cluster ω_i, we can solve the following linear program:

$$\pi_i^{\sigma_u} = \max_{p_1, \ldots, p_K} \sum_{\{j|\sigma_u^{-1}(j) \leq \sigma_u^{-1}(i)\}} p_j \tag{36}$$

Finally, we can take the distribution of the cluster ω_i dominating all the distributions π^{σ_u}:

$$\pi_i = \max_{u=1,L} \pi_i^{\sigma_u} \qquad \forall i \in \{1, \ldots, K\} \tag{37}$$

At this point, we propose to build a measure of possibility distribution π_{normal} as a threshold, and then a cluster will be considered as normal if its possibility distribution satisfies :

$$\pi_i \geq \pi_{normal}, \tag{38}$$

Otherwise it is ranged in subspace potentially suspicious. And our attention will be placed in this subspace for anomaly detection.

To find the possibility distribution π_{normal}, we take into account the memberships of the data points inside a cluster. The memberships can be seen as the probability that data point belongs to the different clusters. These memberships are calculated with the Gustafson-Kessel clustering algorithm which gives us, for each data point x_t the probability distribution $p = (p_1, p_2, \ldots, p_K)$ (for each data point the constraints $\sum_{i=1}^{K} p_i = 1$ is always true.

We can use Eq. (28) to calculate the possibility distribution of each data point x_t of the sample x. We obtain a matrix π_K^N of dimension $K \times N$ (remember K is the number of components (clusters) and N is the length of the data sample x). We take the **mean** for each column (each column containing the possibility distribution for data point x_t) lying in all clusters. Then we obtain a second matrix π_1^N and finally we use Eq. (39) to derive the threshold π_{normal} :

$$\pi_{normal} = max(\pi_1^N) \tag{39}$$

A. Model Validation

1) Experimental data: Abilene and SWITCH networks: In this work, we used a collection of data coming from the Abilene network. The Abilene backbone has 11 Points of Presence(PoP) and spans the continental US. The data from this network was collected from every PoP at the granularity of IP level flows. The Abilene backbone is composed of Juniper routers whose traffic sampling feature was enabled. Of all the packets entering a router, 1% are sampled at random. Sampled packets are aggregated at the 5-tuple IP-flow level and aggregated into intervals of 10 minute bins. The raw IP flow level data is converted into a PoP-to-PoP level matrix using the procedure described in [8]. Since the Abilene backbone has 11 PoPs, this yields a traffic matrix with 121 OD flows. Each traffic matrix element corresponds to a single OD flow, however, for each OD flow we have a seven week long time series depicting the evolution (in 10 minute bin increments) of that flow over the measurement period. All the OD flows have traversed 41 links. Synthetic anomalies are injected into the OD flows by the methods described in [8], and this resulted in 97 detected anomalies in the OD flows. The anomalies injected in the Abilene data are small and high *synthetic volume anomalies*. We used exactly the same Abilene data as in [9]. So for a full understanding on how the **ground-truth** is obtained (based on EWMA and Fourier algorithms), we refer the reader to [9].

2) Results and comparison: The first result of our study is devoted to entropy reduction. The approach shows the ability of the EKF and UKF to estimate the state of the system under noisy measurements. We implemented these filters in Matlab to the linear dynamical system described in our previous work [14], [15] which is our reference to compare the Linear Kalman Filter to the EKF and the UKF. This means that the functions f and h are respectively set to $C_t x_t$ and $A_t x_t$. We suppose that system state and measurement are time invariant. To calibrate the EKF, we first need to find the unknown parameters C, A, Q and R and also the different Jacobian matrices. Since we consider a linear system, the matrices C, Q and R can be obtained with the same method we deal with in our paper referenced in [15] based on the expectation-maximization algorithm. We run the filters for each column timeseries and for the Abilene and Switch networks, we use the same constant quantities $Q = 10.92$ and $R = Q \times 15$ and the estimation is quite perfect. These same values are used to run the EKF and UKF for the sake of comparison. Additional matrices (i.e Jacobians) are needed for EKF, that's why it is often difficult to build a suitable model based on this framework. But, for our study we just specify these quantities as the values of C and A since the data observations themselves are very simple timeseries. They are set to $F = 9.1$, $H = 5$. By inspection of the graphs in figure Fig. 1, it seems that EKF and UKF performs with the same level of performance when they are calibrated with the same parameters. The goodness of an algorithm can be evaluated with the root mean square RMS) error defined as: $\sqrt{\frac{1}{N}\sum_{k=1}^{N} | x_k - E(x_k - y_{1:k}) |^2}$. Table I shows the RMS error for the EKF and UKF algorithms and it makes clear that the UKF performs better than the EKF. This filtering results give quite the same performance as when we used the linear Kalman filter [14].

After filtering for the purpose of entropy reduction, our aim is to analyze residuals for the scope of anomaly detection. We suppose that anomalies might be rare and might happen on a few number of clusters. We deal with the partition problem where we want to find the appropriate number of clusters built from the innovation process. During this optimization task, parameters were fixed to the following values: $m = 3$, $\epsilon = 0.001$, $\rho = 1$ for each cluster, $c \in [2, 9]$ (interval in which we would find the suitable number of clusters). The values of the validity measures, depending on c, are plotted in figures Fig. 2 and 3 for the K-means, K-medoid, FCM and GK algorithms. The results are shown for the Switch network. Globally, in figure Fig. 3, the validity measures PC and CE from the FCM algorithm does not give us reliable information to obtain the best number of clusters. They are typically increasing (CE) and decreasing (PC) without break (local minimum or maximum). With the K-means, the different graphs show clearly that the number of clusters can be set to $c = 3$ (maximum of the Dunn Index). The K-medoid and more precisely the robust GK algorithm, via the values of XB and DI, obviously confirm that $c = 3$. The same analysis show that, when using the Abilene trafic, the best number of cluster is $c = 4$. The analysis shows that, one must use different clustering algorithms and validity measures to ensure that the selection of the best number of cluster is rigorous.

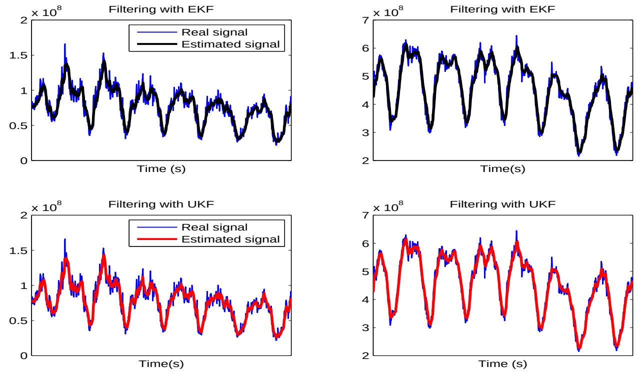

Fig. 1: Real and estimated links obtained using EKF and UKF. Abilene network.

TABLE I: Root Mean Square Error after running the EKF and UKF filter.

Switch TCP

Link i	1	2	3	4	5
EKF	0.4021	0.4023	0.3317	0.3479	0.3107
UKF	0.3349	0.3194	0.2498	0.2655	0.2292

After having the optimum number of clusters, we progress in a next step where we search for which clusters are normal and which ones are abnormal. To this end, we affect to each cluster a degree a possibility, as explained in section III. The results are depicted in table Table II. And finally, when applying our decision scheme for cluster normalcy identification, we decide to put the label No if a cluster in abnormal and Yes otherwise. To decide if a cluster is normal or not, we have just to find the degree of possibility which acts as a threshold. This threshold in calculated by using the results in table Table III and proceed equations Eq. (38) and Eq. (39). These results show that, in all cases, the clusters labeled as abnormal have always a few number of data, that

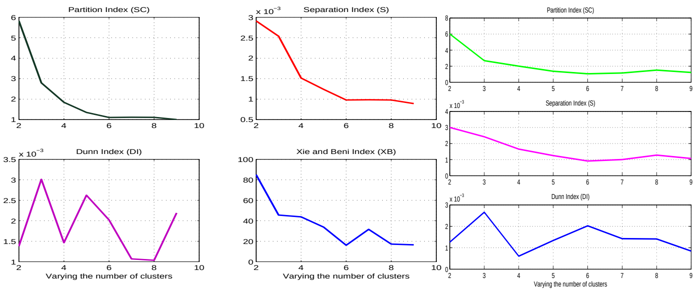

Fig. 2: Validity measures in order to find the best number of clusters. The left 4 graphs using K-means and the right 3 graphs using K-medoid. Switch network.

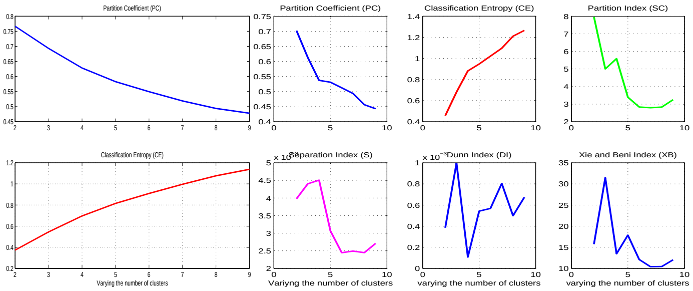

Fig. 3: Validity measures in order to find the best number of clusters. The left 2 graphs using FCM and the right 6 graphs with GK. Switch network.

TABLE II: Interval-valued probabilities, possibility distributions, and length of each cluster. We apply Eq. (38) and show if a cluster is normal or not.

	Abilene			
cluster i	1	2	3	4
p_i^-	0.1741	0.2799	0.2518	0.2114
p_i^+	0.2122	0.3242	0.2949	0.2521
π_i^S	**0.4244**	1.0000	1.0000	0.7165
π_{normal} (Eq. 39)	**0.4393**			
Eq. (38)	**false**	true	true	true
cluster Normalcy	**No**	Yes	Yes	Yes
Length cluster i	**194**	304	275	233

	UDP trafic		
cluster i	1	2	3
p_i^-	0.1795	0.4101	0.3674
p_i^+	0.2039	0.4406	0.3974
π_i^S	**0.2039**	1.0000	0.5899
π_{normal} (Eq. 39)	**0.5429**		
Eq. (38)	**false**	true	true
cluster Normalcy	**No**	Yes	Yes
Length cluster i	**383**	851	765

TABLE III: Memberships of the data points and corresponding possibility distributions, ($\alpha = 0.05$).

	TCP trafic					
time t	1	2	3	. . .	2000	2001
	memberships					
cluster 1	0.9985	0.9985	0.2597	. . .	0.9318	0.2612
cluster 2	0.0010	0.0010	0.2303	. . .	0.0403	0.6769
cluster 3	0.0005	0.0005	0.5100	. . .	0.0279	0.0619
	corresponding possibility distributions					
cluster 1	**1.0000**	0.0015	0.0005	. . .	1.0000	0.3231
cluster 2	**1.0000**	0.0015	0.0005	. . .	0.0682	1.0000
cluster 3	0.4900	0.2303	0.0279	. . .	0.0.0279	0.0619

is what we expected. After finding the abnormal clusters, we just use a basic test of variance to detect anomalies. The results are shown in the ROC curves depicted in figure Fig. 4. The ROC curve is a convenient tool to learn about the tradeoff between the percentage of anomalies detected (detection rate-DR) and the false positive rate (false alarms-FPR). The results demonstrate in our study that the UKF performs better than the EKF, perhaps due to the fact that it is ore simple and easy to calibrate the UKF filter than the EKF. For example, for the UDP trafic we gain about 80% of DR with 0% of FPR for the UKF while the EKF produces 0.05% of FPR for the same DR. We obtain the same interpretation for the other trafic.

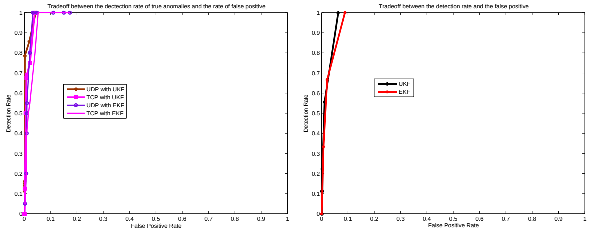

Fig. 4: ROC curve illustrating the tradeoff between the detection rate and the false positive rate. The results are drown using EKF and UKF. Left graph Switch TCP and UDP trafic and right graph for Abilene.

IV. Conclusion

In this work, we have shown that the EKF and UKF can be used to build convenient models for the purpose of anomaly detection in communication networks. The calibration of the UKF is more easy and the difficulty to build the Jacobian matrices is the possible reason that EKF produces a mean square error more important than for UKF. Based on possibility distributions, we have developed a new scheme that allows us to build the normal and abnormal spaces. We then analyze, by means of ROC curve, the tradeoff between the detection rate and the false positive rate. A difficult task in the final procedure of tracking the true anomalies is related to the choice of the test (here test of variance) in order to reduce considerably the false positive. All our experiences to training the possibility theory framework use a confidence level set to 95% (corresponding to $alpha = 0.05$). We have runs multiple other scenarii with confidences lying between 90% and 99% and the results are the same.

References

[1] Dubois, D., Prade, H. and Sandri, S.: On possibility/probability transformations. In Proceedings of the Fourth Int. Fuzzy Systems Association World Congress (IFSA91), Brussels, Belgium, pages 50-53, (1991).

[2] Maybeck, P. Stochastic Models, Estimation and Control, Volume 2. Academic Press. 1982. Using MATLAB. Wiley Interscience. 2001.

[3] Bar-Shalom, Y., Li, X.-R., and Kirubarajan, T. Estimation with Applications to Tracking and Navigation. Wiley Interscience. 2001.

[4] Masson, M., H. and Denoeux, T.: Inferring a possibility distribution from empirical data. Fuzzy Sets and Systems 157(3): pp. 319-340, 2006.

[5] Zadeh, L.,A. Fuzzy sets as a basis for a theory of possibility. Fuzzy Sets and Systems, 1: pp. 3-28, 1978.

[6] Bezdek, J. C. Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, 1981.

[7] Bensaid, A. M., Hall, L.O, Bezdek, J.C., Clarke, L.P., Silbiger, M.L., Arrington, J. A and Murtagh, R.F Validity-guided (Re)Clustering with applications to image segmentation. IEEE Transactions on Fuzzy Systems, 4:112- 123, 1996.

[8] Lakhina, A., Crovella, M. and Diot, C.: Characterization of network-wide traffic anomalies. In Proceedings of the ACM/SIGCOMM Internet Measurement Conference. pp. 201-206.(2004)

[9] Lakhina, A., Crovella, M.,Diot, C.: Diagnosing Network-Wide Traffic Anomalies. In ACM SIGCOMM (2004).

[10] Xie X.,L. and Beni, G. A. Validity measure for fuzzy clustering. IEEE Trans. PAMI, 3(8):841-846, 1991.

[11] Wan, E. and Van Der Merwe, R. The Unscented Kalman Filter. Wiley Publishing, 2001.

[12] Babuska, R., Van der Veen, P. J. and Kaymak, U: Improved covariance estimation for Gustafson-Kessel clustering. IEEE International Conference on Fuzzy Systems, pages 1081-1085, 2002.

[13] Goodman, L. A.: On simultaneous confidence intervals for multinomial proportions. Technometrics, 7(2): pp. 247-254, 1965.

[14] Ndong, J., Salamatian, K., :A Robust Anomaly Detection Technique Using Combined Statistical Methods. CNSR 2011, *IEEE Xplore* 978-1-4577-0040-8, pp: 101-108. (May 2011).

[15] Ndong, J., Salamatian, K.,: Signal Processing-based Anomaly Detection Techniques: A Comparative Analysis. INTERNET 2011, The Third International Conference on Evolving Internet. ISBN: 978-1-61208-141-0.

[16] Ndong, J.: Anomaly Detection: A Technique Using Kalman Filtering and Principal Component Analysis. ATAI NTC 2012 GSTF 2012.