15

MANET Mining: Mining Association Rules

Ahmad Jabas
Department of Computer Science and Engineering, Osmania University
India

1. Introduction

The growing advances in mobile devices, processing power, display and storage capabilities, together with competitive market has enabled *information technology* to be more affordable and available to almost everybody around the world. Moreover, with the advent of wireless communications and mobile computing, another type of wireless communications, called Mobile Ad hoc NETworks (*MANETs*), came into existence.

The operation of *MANET* does not depend on pre-existence infrastructure or base stations, since there is no central node in the network and nodes collaboratively share all the network activities. The simplicity of *MANET* deployment comes with a cost of complexity of the algorithms in different layers. In addition, the absence of the infrastructure induces new challenges to wireless networks in the fields of routing, security, power conservation, quality of service, and so on.

For better perception of the new concepts in this chapter, a summary of the necessary background in Data Mining (*DM*) is given, and particularly more emphasis and in depth explanation is given on *association rule mining technique*, an area upon which the new concepts of this chapter revolves.

DM or Knowledge Discovery in Databases (*KDD*) is defined as "The nontrivial extraction of implicit, previously unknown, and potentially useful information from data" (Frawley et al., 1992). *DM* is the process of finding hidden relationships in data sets and summarizing these patterns in models. These patterns can be utilized to understand the whole data sets. In simplified terms, *DM* is a technology that allows an applicant to discover knowledge, which is hidden in large data sets, by applying various algorithms (Hofmann, 2003).

This chapter shows how *DM* approaches are applied to *MANET*, in that the traffic of *MANET* is mined in a simple way called "*MANET* Mining using Association Rule Techniques". *MANET* Mining enables the establishment of the fact that there are still some hidden relationships (patterns) amongst routing nodes, even though nodes are independent of each other. These relationships may be used to provide useful information to different *MANET* protocols in different layers. Precisely, *MANET* Mining, discovers hidden patterns (meta-data) in the third layer to be used as common tokens (keys) in the application layer in a bid to address one challenging security problem in *MANET*, namely, key distribution. This is the first time this approach has been used to solve key distribution problem in *MANET*.

Interestingly, security in *MANET* has been paid a lot of attention over the past few years. One of the most challenging security issue in *MANET* is key management where there is no on-line access to trusted authorities. Key management is the central part of any secure communication, and is the weak point of system security and protocol design. Most

cryptographic systems rely on the underlining secure, robust, and efficient key management system.
Key management scheme is the prerequisite for all security primitives and thus, it is the basis for secure $MANETs$. However, the performance of existing key distribution schemes developed so far is undesirable in the terms of efficiency and scalability. Besides, these schemes revolve around Third Trusted Party (TTP) and therefore, compromising this TTP means disclosing all the issued keys. Surprisingly, the fully distributed and self-organized key distribution schemes without TTP are still not robust to changing topology or intermittent links commonly encountered in $MANETs$ (Chan, 2004).
Section 2 gives an overview of *association rule* and its application to social networks. Section 3 explains how data mining approaches are applied in $MANET$ and introduces a new distributed algorithm, $MANET$ Mining. Section 4 provides a detailed explanation of applying Association Rule Techniques to $MANET$ traffic. Section 5 shows how Association Rule Mining Techniques are used on $MANET$ traffic with a Step Threshold. Section 6 shows an important application of $MANET$ Mining to key distribution. Section 7 concludes the chapter and draws some future research directions.

2. Data Mining: An overview of association rule mining technique

2.1 Association rule

This section presents a methodology known as association rule mining, useful for discovering interesting relationships hidden in huge data sets. Association rules have received lots of attention in DM due to their many applications in marketing, advertising, inventory control, and many other areas (Simovici & Djeraba, 2008). Association Rules can be derived using supervised and unsupervised processes (Joe, 2009).
Let $A = \{l_1, l_2, l_3, l_4, ..., l_m\}$ be a set of items. Let T be a set of transactions on a database. A transaction t is said to support an item l_i, if l_i is present in t. Moreover, t is said to support a subset of items $X \subseteq A$, if t supports each item l in X (Pujari, 2001).
$X \subseteq A$ is said to have a Support s in T, denoted by s(X), if s percent of transactions in T support X.
A subset X is said to be a Frequent Set (FS) in T with respect to σ (where σ is a user-specified minimum Support), if

$$s(X) \geq \sigma$$

FS is called Maximal Frequent Set (MFS) if no supper set of this set is FS. The following are important properties of MFS:

– Downward Closure: Any subset of FS is FS.

– Upward Closure: Any supper set of an infrequent set is an infrequent set.

Moreover, the set of all Maximal Frequent Sets ($MFSs$) is called maximum frequent set.
For a given database, an association rule is an expression of the form:

$$X \Longrightarrow Y$$

where X and Y are subsets of A. The intuitive meaning of such a rule is that a transaction of the database which contains X tends to contain Y.
Some used measures of rule interestingness are:

1. Confidence (τ): The association rule $X \Longrightarrow Y$ holds with confidence τ if τ% of transactions in T that supports X also supports Y.

2. Support (σ): The association rule $X \Longrightarrow Y$ has Support σ in the transaction set T if σ% of transactions in T support $X \cup Y$.

Association rules have another synonym, *market basket*. *Market basket* analysis (Association Rule Mining) is a research technique for retailers that is used to discover customer purchasing patterns (Post, 2005). In direct marketing, DM has been used extensively to identify potential customers for a new product (target selection) (Javaheri, 2007). Accordingly, accumulated data is analyzed to know the behavior of the customers.
The supermarket may be interested in identifying associations between item sets; for example, it may be interested to know how many of the customers who bought bread and cheese also bought butter (Simovici & Djeraba, 2008). Furthermore, nowadays a `market basket` is applied to e-commerce rather than supermarkets. For example, whenever customers shop an item online, they might read a recommendation after that "Customers who bought this item also bought ..." or "Buy these two items together and save ...".
Binary format can be used to represent *market basket*, each row is a transaction and each column is an attribute (item). An item is represented as a binary variable, if the item is present the value of the variable is one, otherwise its value is zero.
The problem of mining association rules can be decomposed into two subproblems (Agrawal & Shafer, 1996):

1. Find all set of items (itemsets) whose support is greater than the user-specified minimum Support (σ). Itemsets with minimum Support are called frequent sets (itemsets).
2. Use the frequent itemsets to generate the desired rules. The general idea is that if, say for example, $ABCD$ and AB are frequent itemsets, then we can determine if the rule $AB \Rightarrow CD$ holds by computing the ratio:

$$confidence = \frac{Support(\{ABCD\})}{Support(\{AB\})} \geq \tau$$

   Note that this rule has minimum support because ABCD is frequent.

Because of the multiplicity and variety of Association Rules Mining (ARM) techniques, Apriori algorithm is chosen and applied in this section as a de facto algorithm for mining association rules.

2.2 Apriori algorithm

The problem of deriving association rules from data was first formulated by Agrawal, Imielinski and Swami in 1993 and is called the *market-basket* problem (Agrawal et al., 1993). They introduced in their work the Apriori algorithm, which is the most commonly used association rule discovery algorithm that utilizes the frequent sets. This algorithm make use of the downward closure property. Algorithm 1 shows the pseudo-code of Apriori algorithm (Agrawal & Shafer, 1996; Yao et al., 2003).
One of the advantages of the method is that before reading the database at every level, it graciously prunes many of the sets which are unlikely to be frequent sets. Apriori algorithm has become a reference algorithm, and has been improved in several ways in terms of time complexity, the number of scans of the database, size of transaction, threshold and so forth.
Since association rules are derived from $MFSs$, the terms MFS and association rules are used interchangeably.

Algorithm 1 Apriori

```
Initialize: k := 1, C_1 = all the 1-itemsets;
read the traffic bit-matrix to count the Support of C_1 to determine L_1
while L_{k-1} ≠ φ do
    C_k = gen-candidate-itemsets with the given L_{k-1}
    prune(C_k)
end while
L_1 := {frequent 1-itmesets};
k := 2; //k represents the pass number
for all rows ∈ bit-matrix do
    increment the count of all candidates in C_k that are contained in r;
    L_k := All candidates in C_k with minimum Support;
    k := k + 1
end for
Answer L := ⋃_k L_k;
```

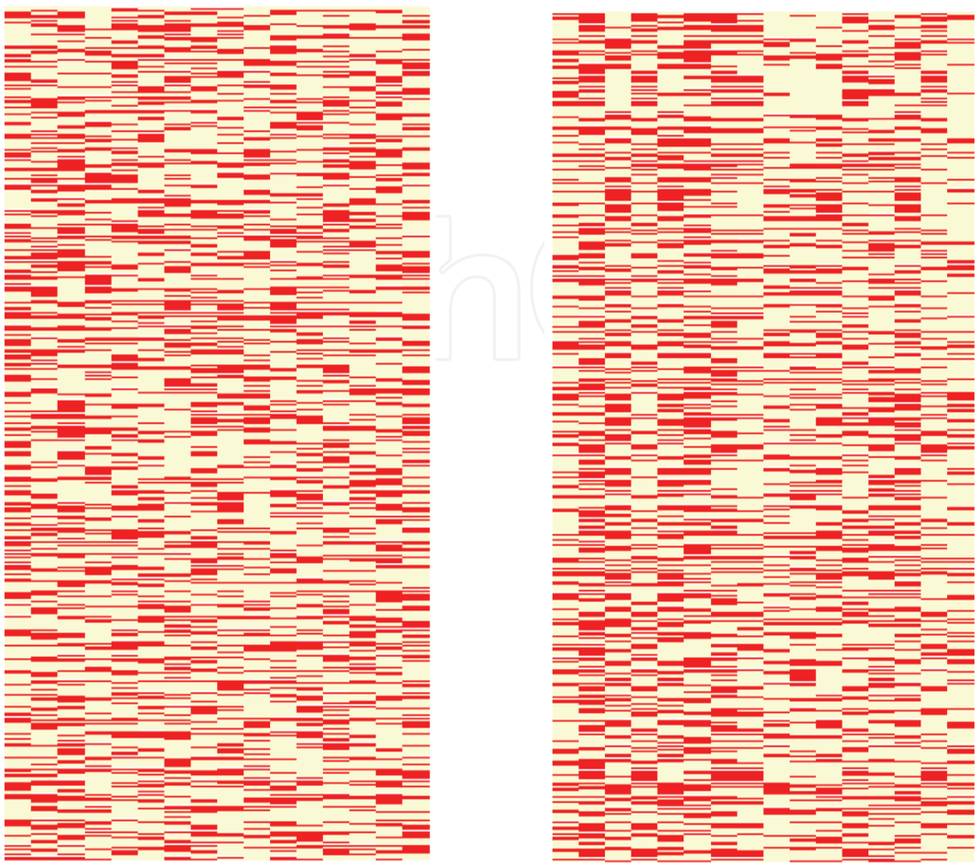
Association Rule	Confidence
{ budget resolution = no, MX-missile = no, aid to El Salvador = yes} ⟶ {Republican}	91.0%
{ budget resolution = yes, MX-missile = yes, aid to El Salvador = no} ⟶ {Democrat}	97.5%
{ crime = yes, right-to-sue = yes, physician fee freeze = yes} ⟶ {Republican}	93.5%
{ crime = no, right-to-sue = no, physician fee freeze = no} ⟶ {Democrat}	100%

Table 1. Association rules extracted from the 1984 US Congressional Voting Records.

2.3 Application of association rule mining technique to social networks

Market basket is used not only in supermarkets but also in social networks. For example, one of the hidden knowledge in social networks is mining criminal relationship (Fard & Ester, 2009).

Tan (Tan et al., 2006), gave a simple and clear example of a social network in small community and applied association analysis to United States congressional voting records. The data-set is maintained in University of California Irvine (*UCI*) machine learning repository and includes votes for each of the U.S. house of representatives congressmen on the 16 key votes, 1984 (Asuncion & Newman, 2007). Figures 1(a) and 1(b) show random and voting data respectively. Even though both figures look random, there are still underlying relationships/patterns in the data and these relationships can be revealed through *DM* techniques, for example, Apriori algorithm. As a result, table 1 shows some of the relationships/outcome obtained by applying Apriori algorithm on the voting data set. Notably, at confidence of 91%, the first association rule is derived, which says that most of the members who voted yes for "aid" to "El Salvador" and no for "budget resolution" and "*MX* missile" are Republicans; while at 97.5% another association rule is derived which says that those who voted no for "aid" to "El Salvador" and yes for "budget resolution" and "*MX* missile" are Democrats. Of course, by changing the confidence level new rules can be found.



(a) Random data. (b) Voting data.

Fig. 1. Congress voting records

3. MANET mining

3.1 Introduction

The powerful methods for discovering knowledge from data go beyond the boundaries of traditional statistics, machine learning and database querying, to be applied in the field of *MANET*.

Section 3.2 compares *DM* and *MANET* Mining, while the rest of the subsections study how to mine traffic in a network called *MANET*, present a general distributed algorithm named *MANET* Mining with major concentration on *ARMs*, and explain the mathematical analysis of *MANET* itemsets.

3.2 Comparison of MANET and data mining: Visions of convergence

Data mining, in reference to transactions is similar, to a large extent, to mining packets in *MANET* in the following aspects (Jabas et al., 2008c):

1. Each transaction in data mining is a set of items (attributes). In case of *MANET*, the nodes are the attributes and the transaction is the transmission of one packet.
2. Data mining is applicable to a database with a large number of transactions. *MANET* mining is applicable to a traffic with a large number of packets.

3. The purpose of mining association rules in a database is to discover all rules that have Support and Confidence (predictability) greater than or equal to the user-specified minimum Support and minimum Confidence. In case of $MANET$, the rules represent the most likely patterns among the cooperating/routing nodes.
4. Each Frequent Set FS in data mining is equivalent to the common nodes of different paths in $MANET$.

3.3 MANET mining algorithm

$MANET$ consists of a finite collection of computational entities (nodes) communicating by means of messages (packets). Accordingly, $MANET$ communication algorithms are distributed by nature.

A distributed algorithm or protocol for given entities is a set of rules that specify the functionality of each entity. The collective but autonomous execution of those rules, possibly without any supervision or synchronization, must enable the entities to perform the desired task to solve the problem (Santoro, 2007). Algorithm 2 and its subprograms (algorithms 3 and 4) represent the pseudo codes of the new general distributed $MANET$ Mining protocol (algorithm).

The algorithm shows that mining techniques are applied with a Threshold to the mining procedure. Depending on the address information in a given packet, a node can be a source node, rely node or destination node. The bit-matrix is constructed from the bit-vectors of routed packets, which can be carrying data or acknowledgement.

For a $MANET$ with n nodes, the header of each packet should have a bit-vector of length n bit. Each bit represents inrormation about the participation of a node in the routing process of a packet. The default values of these vector's bits are zeros. Only the traversed node assigns a value of one to its entry in the packet's bit-vector, that means, the vector's values corresponding to untraversed nodes remain unchanged/unaltered at the default value of zero.

The algorithm does not overload the network by introducing new packets. Nodes do not add new traffic to $MANET$, but just capture the passing packets and assign a value of one to their corresponding entry in the packet's bit-vector (header). Since one bit is enough to represent one node, few bytes in the packet's header are required to represent the network. Each node is capable of extracting a sample of the traffic in $MANET$ to construct its own bit-matrix without any request from any other node.

The final status ϕ in the algorithm indicates that the algorithm works continuously to build the bit-matrix preparing it for mining at anytime later, i.e., the algorithm is proactive and the nodes do not reach a final status.

Spontaneously, whenever the threshold value is attained, the mining procedure (algorithm) is invoked. Despite the many types of thresholds, namely, time stamp, size of bit-matrix, number of differences between successive entries, columns, and so forth, two thresholds are studied in this chapter. The first is timestamp threshold, which is utilized in section 4 while the second, called the Step Threshold, and defined as the number of differences between two successive entries in the bit-matrix, is utilized in section 5.

3.4 Mathematical modelling and analysis of MANET itemsets

Hegland introduced a formal mathematical model to describe itemsets and associations in DM domain (Hegland, 2005). This section explains how Hegland's mathematical model can be applied in $MANET$.

Algorithm 2 MANET Mining

```
• Status Values: S = {SOURCE, RELAY, DESTINATION};
  S_INIT = {SOURCE, RELAY, DESTINATION};
  S_TERM = φ.
• Restrictions:
  R = {Connectivity}.
SOURCE
                Spontaneously
                        BEGIN
                                contribute to the bit-matrix;
                                ROUTE
                        END
                Spontaneously
                        BEGIN
                                MINE
                        END
                receiving (Acknowledgement)
                        BEGIN
                                contribute to the bit-mtrix;
                        END
RELAY
                Spontaneously
                        BEGIN
                                MINE
                        END
                Receiving (Data-Packet)
                        BEGIN
                                contribute to the bit-matrix;
                                ROUTE;
                        END
                Receiving (Acknowledgement)
                        BEGIN
                                contribute to the bit-matrix;
                                ROUTE;
                        END
DESTINATION
                Spontaneously
                        BEGIN
                                MINE
                        END
                Receiving (Data-Packet)
                        BEGIN
                                contribute to the bit-matrix;
                                ROUTE
                        END
```

Algorithm 3 Procedure MINE

1: BEGIN
2: **if** the user-specified threshold is met **then**
3: apply mining algorithms (Apriori) to the bit-matrix;
4: **end if**
5: END

	N_0	N_1	N_2	...	N_j	...	N_{n-1}
$itemset_0$	$a_{0\,0}$	$a_{0\,1}$	$a_{0\,2}$	...	$a_{0\,j}$	...	$a_{0\,n-1}$
$itemset_1$	$a_{1\,0}$	$a_{1\,1}$	$a_{1\,2}$	...	$a_{1\,j}$	...	$a_{1\,n-1}$
$itemset_2$	$a_{2\,0}$	$a_{2\,1}$	$a_{2\,2}$	...	$a_{2\,j}$	...	$a_{2\,n-1}$
...	...	...	...	...	...	...	...
$itemset_i$	$a_{i\,0}$	$a_{i\,1}$	$a_{i\,2}$	...	$a_{i\,j}$	...	$a_{i\,n-1}$
...	...	...	...	...	...	...	...
$itemset_{m-1}$	$a_{m-1\,0}$	$a_{m-1\,1}$	$a_{m-1\,2}$	...	$a_{m-1\,j}$	...	$a_{m-1\,n-1}$

Table 2. The bit-matrix.

Consider a $MANET$ with n nodes, whose source node is N_s and destination node is N_d. Nodes are enumerated from N_0 to N_{n-1}. $MANET$ is applying some routing protocol, where each delivered packet carries an itemset. Itemsets (bit-vectors) are sets of strings of n binary numbers, where $a \in A := \{0,1\}^n$. In table 2, the value of the item j is set to one in the corresponding itemset iff the j^{th} node contributed to the process of routing the corresponding packet, otherwise, the item's value remains at the default value of zero. The set of itemsets (bit vectors) forms a bit-matrix, where the j^{th} column represents the node N_j and the i^{th} row represents the i^{th} itemset.
The nodes involved in routing are chosen randomly. Thus, the corresponding itemsets and bit-matrix $A \in \{0,1\}^{m,n}$are random, where m is the number of itemsets. The elements $a_{i\,j}$ are binary random variables.
Assume the probability distribution function $p :\rightarrow [0,1]$, where:

$$\sum_{a \in A} p(a) = 1$$

and $A = \{0,1\}^n$. The probability with distribution p is denoted by P and has:

$$P(A) = \sum_{a \in A} p(a)$$

Algorithm 4 Procedure ROUTE

1: BEGIN
2: set the node's bit-vector value to one
3: use the given routing algorithm
4: END

The data can be represented as an empirical distribution (Cumulative Distribution Function (CDF)) with:

$$P_{emp}(a) = \frac{1}{n}\sum_{i=1}^{n}\delta(a - a^{(i)})$$

where $\delta(a)$ is the indicator function; $\delta(0) = 1$ and $\delta(a) = 0$ if $a \neq 0$. For simplicity, the empty *market basket* is denoted by 0 instead of (0, ..., 0).
Mining of frequent itemsets means to find itemsets (bit-vectors) that occur frequently in the traffic. Accordingly, the itemsets are partially ordered with respect to the inclusion. In other words, $a \leq b$ if the set with representation a is a subset of the set with representation b or $a = b$. Now, the Support of an itemset a can be defined with the partial order as follows:

$$s(a) = P(c \mid a \leq c)$$

$s(a)$ is also called anticummulative distribution function of the probability P. The Support is function s : $A \rightarrow [0,1]$ and $s(0) = 1$. The Support is antimonotone, i.e., if $a \leq c$ then $p(a) \geq p(b)$. Previous equation can be reformulated in terms of $p(a)$ as:

$$s(a) = \sum_{c \geq a} p(c)$$

This is a linear system of equations which can be solved recursively using $s(e) = p(e)$, where $e = (1,...,1)$ is the maximum item set and:

$$p(a) = s(a) - \sum_{c \geq a} p(c)$$

That means the Support function $s(a)$ provides an alternative description of the probability measure P, which is equivalent to p.

3.5 The random itemsets of MANET nodes

Assume that the nodes that contribute to routing are chosen randomly, i.e., the items (bits) in a are chosen independently with probability p_0. This corresponds to routing protocol that randomly chooses nodes as packets move along the way from the source to the destination.
Then, the distribution is:

$$p(a) = p_0^{|a|}(1 - p_0)^{n-|a|}$$

where $|a|$ is the number of bits (contributing nodes) in the itemset a and n is the total number of nodes in $MANET$. As any $c \geq a$ has at least all the bit sets which are sets in 'a' extracted for the Support

$$s(a) = p_0^{|a|}$$

and the frequent itemsets are those with at most the following number of items:

$$|a| \leq log(\sigma_0)/log(p_0)$$

This relation finds itemsets that have a specific Support and and the probability of choosing a node.
Assume that the items are chosen independently with different probabilities p_j, then the probability of choosing an itemset is:

$$p(a) = \prod_{j=1}^{n} p_j^{a_j}(1 - p_j)^{1-a_j}$$

and

$$s(a) = \prod_{j=1}^{n} p_j^{a_j}$$

Using Zif's law:

$$p_j = \frac{\alpha}{j}$$

where α is a constant. This means that itemsets with few popular itemsets are most likely.

4. MANET mining: Mining temporal association rules (TARs)

4.1 Introduction

$MANET$ is an autonomous system of mobile routers (and associated hosts) connected by wireless links, the union of which forms an arbitrary graph. The ability to specify the topology of a $MANET$ could also prove to be crucial if limitations in the scalability and total capacity of $MANETs$ are found to exist (Rashmi, 2009; Robinson, 2007).

This section proves that even though the topology is changing rapidly, i.e., the graph is not constant, there are still hidden relationships among the nodes. The association rule techniques are responsible for revealing these relationships, and seek to identify what items go together (Olson & Delen, 2008). The Correlation Ratio (CR), defined in the next subsection, measures the strength of the relationships.

Section 4.2 demonstrates how to apply the association rule technique periodically, every Δ second as a Threshold, to the bit-matrix constructed from $MANET$ traffic. Accordingly, these rules are denoted in this chapter by Temporal Association Rules ($TARs$). Section 4.3 is concerned with simulation with different parameters and the results show that there are still some relationships among the nodes even though the topology is changing.

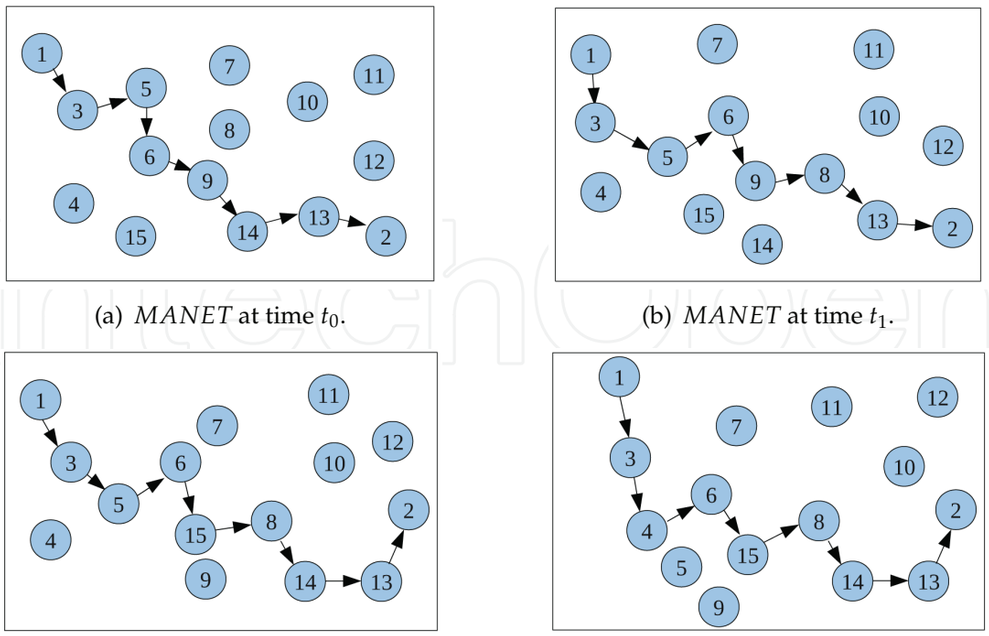
4.2 Demonstration on a simple MANET

This section shows the application of association rule techniques to $MANET$ and explains the use of Apriori algorithm to mine $MANET$ traffic (Jabas et al., 2008b). Consider $MANET$ scenarios in Fig. 2 each representing a 15 node $MANET$, in which node N_1 is the source and node N_2 is the destination. For simplicity, a small number of packets is considered, i.e., not more than five packets in each of the scenarios. The initial path at time t_0, depicted in Fig. 2(a), is $<N_1, N_3, N_5, N_6, N_9, N_{14}, N_{13}, N_2>$ and the source sends 3 packets, after which the topology changes, in the sense that node N_{14} leaves and node N_8 enters the route leading to the second scenario in Fig. 2(b), in which the source sends 5 packets through the path $<N_1, N_3, N_5, N_6, N_9, N_8, N_{13}, N_2>$ at time t_1. The third scenario, in Fig 2(c), is derived from the second scenario as a result of change in topology, i.e., node N_9 leaves while node N_{14} and node N_{15} enter. 4 packets are sent through the new path $<N_1, N_3, N_5, N_6, N_{15}, N_8, N_{14}, N_{13}, N_2>$. Finally, the fourth scenario is derived from the third scenario by some change in topology as shown in Fig. 2(d) such that node N_5 leaves and node N_4 joins. 4 packets are sent through the path $<N_1, N_3, N_4, N_6, N_{15}, N_8, N_{14}, N_{13}, N_2>$.

The bit-matrix in both the source and destination corresponding to this transmission of packets is shown in table 3. Apriori algorithm is applied to the bit-matrix with support $\sigma = 70$ %. Shown below are the stepwise application of Apriori algorithm on the bit-matrix:

i:=1

$C_1 = \{\{N_1\}, \{N_2\}, \{N_3\}, \{N_4\}, \{N_5\}, \{N_6\}, \{N_7\}, \{N_8\}, \{N_9\}, \{N_{10}\}, \{N_{11}\}, \{N_{12}\}, \{N_{13}\}, \{N_{14}\}, \{N_{15}\}\}$



(a) $MANET$ at time t_0. (b) $MANET$ at time t_1.

(c) $MANET$ at time t_2. (d) $MANET$ at time t_3.

Fig. 2. Simple $MANET$ demonstration of TARs.

After pruning C_1 the set of frequent sets with one element is:
$L_1 = \{\{N_1\},\{N_2\},\{N_3\},\{N_5\},\{N_6\},\{N_8\},\{N_{13}\}\}$
i:=2
$C_2 = \{\{N_1,N_2\},\{N_1,N_3\},\{N_1,N_5\},\{N_1,N_6\},\{N_1,N_8\},\{N_1,N_{13}\},\{N_2,N_3\}\ ,\{N_2,N_5\},$
$\{N_2,N_6\},\{N_2,N_8\}\ ,\{N_2,N_{13}\},\{N_3,N_5\},\{N_3,N_6\},\{N_3,N_8\},\{N_3,N_{13}\},\{N_5,N_6\},\{N_5,N_8\},$
$\{N_5,N_{13}\},\{N_6,N_8\},\{N_6,N_{13}\},\{N_8,N_{13}\}\}$
After pruning C_2, the set of frequent sets with two elements is:
$L_2 = \{\{N_1,N_2\},\{N_1,N_3\},\{N_1,N_5\},\{N_1,N_6\},\{N_1,N_8\},\{N_1,N_{13}\},\{N_2,N_3\}\ ,\{N_2,N_5\},$
$\{N_2,N_6\},\{N_2,N_8\}\ ,\{N_2,N_{13}\},\{N_3,N_5\},\{N_3,N_6\},\{N_3,N_8\},\{N_3,N_{13}\},\{N_5,N_6\},\{N_5,N_{13}\},$
$\{N_6,N_8\},\{N_6,N_{13}\},\{N_8,N_{13}\}\}$
i:=3
$C_3 = \{\{N_1,N_2,N_3\},\ \{N_1,N_2,N_5\},\ \{...\},\ ...\}$
After pruning C_3, the set of frequent sets with three elements is:
$L_3 = \{\{N_1,N_2,N_3\},\ \{N_1,N_2,N_5\},\ \{...\},\ ...\}$
i:=4
$C_4 = \{\{N_1,N_2,N_3,N_5\},\ \{...\}...\}$
After pruning C_4, the set of frequent sets with four elements is:
$L_4 = \{\{N_1,N_2,N_3,N_5\},\ \{...\},\ ...\}$
i:=5
$C_5 = \{...\}$
After pruning C_5, the set of frequent sets with five elements is:
$L_5 = \{...\}$
i:=6
$C_6 = \{\ ...\}$
After pruning C_6, the set of frequent sets with six elements is:

No.	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_{10}	N_{11}	N_{12}	N_{13}	N_{14}	N_{15}
01	1	1	1	0	1	1	0	0	1	0	0	0	1	1	0
02	1	1	1	0	1	1	0	0	1	0	0	0	1	1	0
03	1	1	1	0	1	1	0	0	1	0	0	0	1	1	0
04	1	1	1	0	1	1	0	1	1	0	0	0	1	0	0
05	1	1	1	0	1	1	0	1	1	0	0	0	1	0	0
06	1	1	1	0	1	1	0	1	1	0	0	0	1	0	0
07	1	1	1	0	1	1	0	1	1	0	0	0	1	0	0
08	1	1	1	0	1	1	0	1	1	0	0	0	1	0	0
09	1	1	1	0	1	1	0	1	0	0	0	0	1	1	1
10	1	1	1	0	1	1	0	1	0	0	0	0	1	1	1
11	1	1	1	0	1	1	0	1	0	0	0	0	1	1	1
12	1	1	1	0	1	1	0	1	0	0	0	0	1	1	1
13	1	1	1	1	0	1	0	1	0	0	0	0	1	1	1
14	1	1	1	1	0	1	0	1	0	0	0	0	1	1	1
15	1	1	1	1	0	1	0	1	0	0	0	0	1	1	1
16	1	1	1	1	0	1	0	1	0	0	0	0	1	1	1

Table 3. The bit-matrix both in the source and the destination nodes

$L_6 = \{\{N_1, N_2, N_3, N_5, N_6, N_{13}\}, \{N_1, N_2, N_3, N_6, N_8, N_{13}\}\}$
i:=7
The maximum frequent set is:
$MFSs = L_6 = \{\{N_1, N_2, N_3, N_5, N_6, N_{13}\}, \{N_1, N_2, N_3, N_6, N_8, N_{13}\}\}$
The set of all frequent sets is:
$L = L_1 \cup L_2 \cup L_3 \cup L_4 \cup L_5 \cup L_6$
The maximum frequent set (MFS) gives the relationships (patterns) among the nodes for a set of entries regardless of the total number of nodes in these entries.
A new metric which defines the strength of relationship among nodes in the pattern is shown below:

$$Correlation\ Ratio\ (CR) = \frac{Average\ Size(MFSs)}{Length(route)}$$

Where:
Average Size ($MFSs$): refers to the average number of the nodes in the pattern extracted from the bit-matrix for a specific period of time as a Threshold.
Length ($route$): refers to the total number of nodes in the bit-matrix that are routing for the same period of time.
Higher CR indicates better correlation among nodes. CR may approach 1 for wireless static topology, which means that the entries in the bit-matrix are participating in the routing process.
By mining the whole bit-matrix shown in Table 3 with Support $\sigma = 70$ %, $MFS = \{\{N_1, N_2, N_3, N_5, N_6, N_{13}\}, \{N_1, N_2, N_3, N_6, N_8, N_{13}\}\}$ is obtained. Notably, the average number of nodes in $MFSs$ is 6 and the number of nodes involved in routing (active nodes) in this bit-matrix is 11, i.e., columns that contain at least a one value (unshaded columns in Table 3), therefore,

$$CR\ (with\ \sigma = 70\ \%) = \frac{6}{11}$$

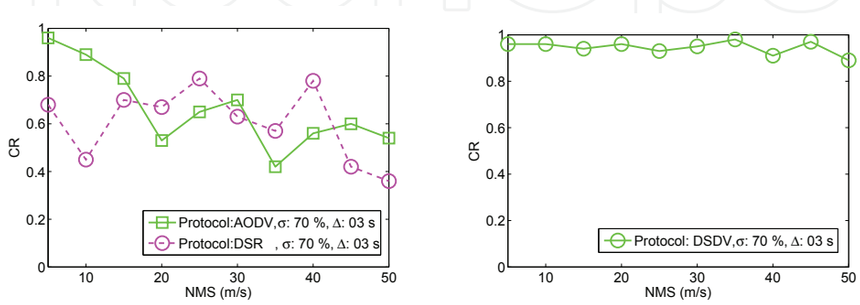
Parameter	Value
Number of the nodes	100
Routing protocol	$AODV, DSDV, DSR$
Mobility model	Random way point
Pause time	1 s
Radio transmission range	250 m
Channel capacity	1 $mbps$
Data flow	CBR, FTP
Data packet size	512 bytes
Node placement	random
Terrain area	1500×1500 m^2
Simulation time	120 s for CBR with $AODV$, $DSDV$ and DSR, 200 s for TCP with $AODV$ and DSR 400 s for TCP with $DSDV$
Propagation model	Two Ray Ground
Node Mobility Speed (NMS)	5, 10, ..., 50 m/s with $AODV$ and DSR 1, 2, ..., 10 m/s with $DSDV$

Table 4. Simulation parameters.

In this example, for simplicity's sake, few nodes and less network traffic is considered. Practically, more nodes may communicate for longer time resulting into a huge number of heterogeneous packets (traffic). The mining of such traffic for a long period of time results into a low CR. This is because, after some time, the topology completely changes with reference to the initial topology. In other words, the number of common nodes decreases with time and consequently, the CR decreases. As a prerequisite, the rate of mining Δ should be in consonance with the rate of changing of topology.

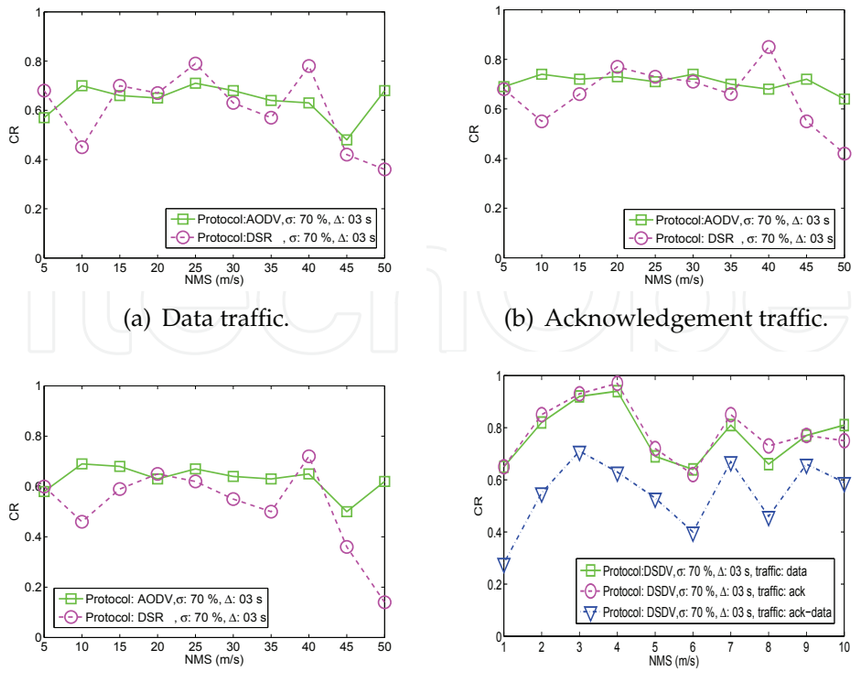
4.3 Simulation and results

This section shows how Apriori algorithm is applied to extract $MFSs$ from different types of $MANET$ traffic. The simulation is performed by $NS2$ (ns2, 2009; Greis, 2007; Fall, 2007). Parameters used in the simulator are summarized in Table 4. Hundred nodes are distributed randomly in the simulation area of 1500×1500 m^2 and with a 250 m transmission range for each node. The Propagation model of the signal is "Two Ray Ground". The channel capacity is 1 $mbps$. The random mobility mode of the nodes is generated by the $CMU's$ node-movement utility "setdest" with different Node Mobility Speeds (NMS) within the range of 5-50 m/s. The nodes do not move through out the simulation time, i.e., they stop according to a constant pause time parameter which lasts for one second. The packet size is 512 bytes.



(a) Data traffic. (b) Data traffic.

Fig. 3. The variation of CR with NMS for connection-less traffic.



(a) Data traffic.

(b) Acknowledgement traffic.

(c) Both acknowledgement and data traffic.

(d) All types of traffic.

Fig. 4. The effect of increasing NMS on CR for connection oriented traffic.

Three standard routing protocols are used in the simulation to deliver the packets from the source to the destination. Two of them are reactive: Ad hoc On-demand Distance Vector ($AODV$) and Dynamic Source Routing (DSR); and one is proactive, Destination Sequenced Distance Vector ($DSDV$).

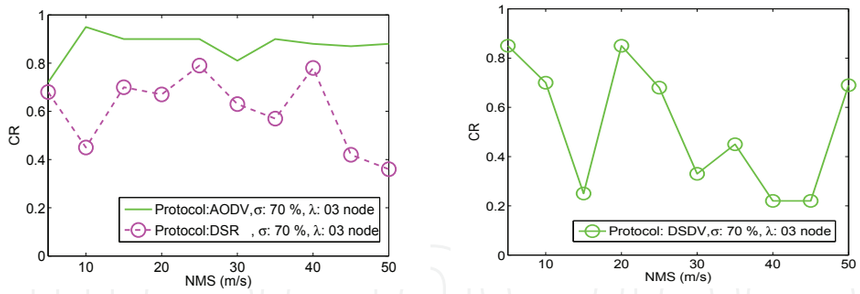
With reactive routing protocol the simulation time of the second application lasts for 200 s, while with proactive protocol, namely $DSDV$, the simulation lasts for 400 s. The NMS varies discretely by one unit between 1 m/s and 10 m/s with $DSDV$, while for the other two protocols the NMS varies discretely in five unit step between 5 m/s and 50 m/s.

The CR metric defined in previous subsection is applied to different scenarios generated in this subsection to evaluate the strength of the hidden relationships (MFS patterns) among the nodes.

Two applications are chosen. The first, the Constant Bit Rate (CBR) application is simulated along with the connection-less transmission protocol/User Datagram Protocol (UDP) for 120 s. Figures 3(a) and 3(b) show how CR varies with NMS for $AODV$, DSR and $DSDV$ protocols with connection-less traffic. The second, the File Transfer Protocol (FTP) is simulated along with connection-oriented Transport Control Protocol (TCP).

Figures 4(a), 4(b) and 4(c) show the behavior of CR as NMS varies for $AODV$ and DSR for data, acknowledgment and both types of packets. Similarly, Fig. 4(d) analyzes the CR for the connection-oriented traffic of $DSDV$ routing protocol for the three types of traffic packets.

It is evident from the above CR/NMS graphs that there are relatively good relationships ($TARs$) among the nodes, and therefore, $MANET$ traffic is a raw material for mining and for revealing these relationships. Despite the high NMS of nodes, up to 50 m/s for $AODV$ and DSR and 10 m/s for $DSDV$, there are still good relationships among the routing nodes.



(a) Data traffic. (b) Data traffic.

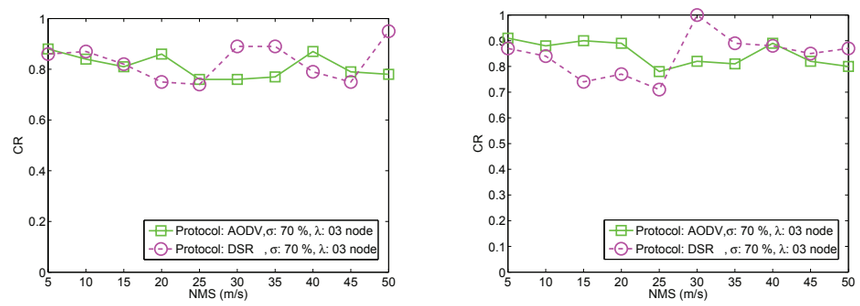
Fig. 5. The variation of CR with NMS for connection-less traffic.

These relationships can be interpreted as topologies representing the routing nodes for a small period of time.

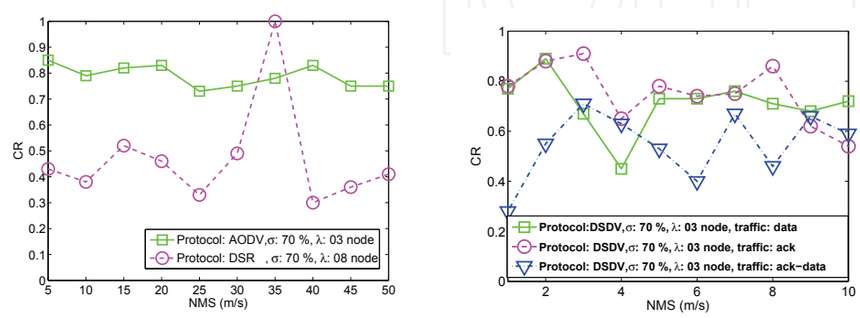
5. MANET mining: Mining step association rules (SARs)

5.1 Introduction

Section 4 explains how Temporal Association Rules ($TARs$) are mined from the $MANET$ traffic, extracted and mined at a rate of Δ. In this section, a new threshold is imposed on the mining process to monitor the difference (change) between successive entries in the bit-matrix. This difference is denoted by Step and the mined rules are denoted, in this chapter, by Step Association Rules ($SARs$). The length of the Maximal Frequent Set (MFS) depends on the



(a) Data traffic. (b) Acknowledgement traffic.



(c) Both acknowledgement and data traffic. (d) All types of traffic.

Fig. 6. The variation of CR with NMS for connection oriented traffic.

similarity among the entries (bit-vectors) in the bit-matrix. Similar bit-vectors leads to longer $MFSs$ and higher CR. In fact, longer $MFSs$ mean more packets are carried in the same routes. $TARs$ do not support sudden changes in the $MANET$ topology (routes), while $SARs$ support and detect the modification in the topology.
An explanation of how to mine $SARs$ is shown in section 5.2, while the application of SAR techniques to $MANET$ traffic and illustration of the results of imposing the Step Threshold λ on the mining procedures is demonstrated in section 5.3.

5.2 Demonstration using a simple MANET

This section explains how SAR mining is executed in $MANET$ (Jabas et al., 2008a). Whenever there is a change in route followed by the packets a corresponding change is reflected in the bit-matrix; for example, in reference to table 3, there are two bit changes between the third entry and the fourth because of change in route/topology .
This demonstration involves three steps. First, the bit-matrix is mined without imposing a Step Threshold. Second, it is splitted into two bit-matrices according to the Step Threshold $\lambda > 2$ nodes. Finally, the new bit-matrices are mined separately.
Following these steps, the bit-matrix in table 3 is used to obtain the results. First, the bit-matrix is mined with Support $\sigma = 70$ %, and the resultant MFS ($SARs$) obtained are:
$L_6 = MFSs = \{\{N_1, N_2, N_3, N_5, N_6, N_{13}\}, \{N_1, N_2, N_3, N_6, N_8, N_{13}\}\}$.
The average number of nodes in $MFSs$ is 6 and the number of nodes involved in routing in this bit-matrix is 11 (i.e. columns that contain at least one "1"value). $CorrelationRatio\ (with\ \sigma = 70\ \%) = \frac{6}{11}$.
Second, Step Threshold $\lambda > 2$ is applied. Accordingly, the first 8 entries saved in the first sub-bit-matrix and the next remaining 8 entries are saved in the second sub-bit-matrix. By mining these sub-bit-matrices with the same Support as in the first step, the following $MFSs$ and their corresponding CRs are obtained as shown below:
$MFSs_1 = L_7 = \{\{N_1, N_2, N_3, N_5, N_6, N_9, N_{13}\}\}$
$MFSs_2 = L_8 = \{\{N_1, N_2, N_3, N_6, N_8, N_{13}, N_{14}, N_{15}\}\}$
$CR_1(\sigma = 70\ \%) = \frac{7}{9}$
$CR_2(\sigma = 70\ \%) = \frac{8}{10}$
Note that CR_1 and CR_2 are higher than CR obtained in the first step. This means that application of Step Threshold leads to stronger CR.

5.3 Simulation and results

This section applies the same simulation scenarios and parameters used in section 4 to show "Step Threshold" effects. Simulation parameters are summarized in Table 4.
Again, the two transport protocols, connection-less and connection-oriented, along with proactive and reactive routing protocols are studied. The same metric, CR, defined in 4.2 is utilized to evaluate the strength of the relationship. The two parameters, the Step Threshold λ and the Support s, with their effect on $MANET$ with different routing protocol are studied in this section.
Figures 5(a) and 5(b) show the effect of increasing NMS on CR with Step Threshold. We conclude that there are relatively strong relationship among the nodes in $AODV$ and DSR protocols as depicted in figure 5(a). However, in 5(b), CR is fluctuating indicating that relationship is not reliable for $DSDV$. Figures 6(a), 6(b), 6(c) and 6(d) show the effect of increasing NMS on CR with Step Threshold, and therefore, it is possible to mine the $MANET$ traffic.

Generally, step threshold improves CR or the strength of the relationships among the nodes.

6. Case Study: Key management in MANET

6.1 Introduction

Like in many distributed systems, security in $MANET$ widely depends on the use of a secure key management mechanisms. Most of the cryptographic mechanism in $MANET$ depends on a key management infrastructure (Ghoreishi & Analoui, 2009).

A key is a piece of input information for cryptographic algorithm. Key disclosure is tantamount to revealing the encrypted information itself, therefore, there is need for Key Encryption Key (KEK) algorithm applied at local host level (Fumy & Landrock, 1993). In view of this, specific key management systems have been developed to fit the characteristics of $MANET$.

Two principles for ad hoc key management are identified (Yi & Kravets, 2004):

1. The first principle is the node participation principle which states that "A key management framework for ad hoc network should rely on a large number of nodes for availability purposes, but on small group of nodes for security purposes ".
2. The second principle, requires the services of TTP principle and states that "A key management framework should use a TTP to improve the quality of authentication of the framework ".

Luo presented Ubiquitous and Robust access control Solution ($URSA$) for $MANETs$ (Luo et al., 2003). $URSA$ implements ticket certification services through multiple-node consensus and fully localized instantiation, and uses tickets to identify and grant network access to cooperative nodes. The merits of this protocol are high efficiency, secure local communication and system availability. The demerits include extremely large threshold compare to the network degree and requirement for off-line configuration before accessing the network.

Seung, (Yi & Kravets, 2003), proposed a new key management framework, MObile Certificate Authority ($MOCA$) for $MANETs$, where the certificate service is distributed to n $MOCA$ nodes that are most secure. $MOCA$ scheme uses threshold cryptography, which is an application of secret sharing. The concept of secret sharing is that it is mathematically possible to divide up a secret to n pieces in such way that anybody who requires the full secret can collect any k pieces out of those n $MOCA$ to reconstruct the full secret. k becomes the threshold needed to reconstruct the secret.

Later, Seung presented a composite key management scheme that uses virtual CA and certificate chaining simultaneously in a single ad hoc network (Yi & Kravets, 2004), thereby, combining the central trust with the fully distributed trust models.

Sections 4 and 5 explain how ARM techniques can be applied to $MANET$ traffic to extract hidden patterns. This section shows an important application of these patterns in the security field. A new method for key distribution in $MANET$ has been developed to mine the traffic moving in $MANET$ in a distributed manner to obtain $MFSs$ that are used as tokens (keys) (Jabas et al., 2010).

Section 6.2 explains new security framework that relies on $MFSs$. Section 6.3 explains the key revocation and renewal issues for the new framework. Section 6.4 gives the mathematical model of the new framework and analyzes its effectiveness in $MANET$. Section 6.5 discusses the experimental analysis of the the new framework. Section 6.6 lists the features that make the new scheme outstanding.

6.2 Key distribution through traffic mining (KDTM)

A group of nodes, the routing nodes, contribute to the bit-vector of the routed packet, which in turn is used to build the bit-matrices in both the source and the destination nodes. Both end nodes apply Apriori algorithm to these bit-matrices, and interestingly, the same $MFSs$ (patterns) are obtained at only these two nodes.

For a $MANET$ with n nodes, a bit-vector of size n is required for each packet routed. Before the source sends a packet to the destination, the entries of the bit-vector are initialized to zero and the bit-vector is attached to the header of the packet. As the packet passes through the routing nodes, the corresponding entries in the bit-vector are set to one. When the packet reaches the destination node the bit-vector is extracted from the packet and attached to the acknowledgement packet that is sent back to the source node, and also this extracted bit-vector is stored in the node's bit-matrix. The size of the bit-matrix, i.e., the number of entries, is a function of packets received and extracted.

Now, the source and destination nodes mine the bit-matrices to extract their $MFSs$, and both nodes must obtain the same result, i.e., the same $MFSs$, used as common tokens between the source and the destination.

The advantage of MFS mining algorithms is that they are tolerant, in the sense that if the bit-matrix is changed slightly, intentionally or by accident, either in the source or in the destination or in both, the same $MFSs$ are obtained in the source and destination nodes. At the same time, if any node other than the source or destination tries to build and mine its own bit-matrix, different $MFSs$ are obtained. This difference in $MFSs$ is brought about by the fact that mining is applied to different bit-matrices.

In addition to the end-nodes, any routing node may also extract the bit-vectors of passing packets for an interval of time and may cache them in its bit-matrix. This means that the routing node does not need any request or permission from any other node to perform these activities. $KDTM$ does not increase the network traffic by inducing control packets as is the case with other key distribution schemes.

A summary of $KDTM$ steps are shown below:

- Nodes synchronization: Nodes should be at least loosely synchronized. There are several distributed algorithms for synchronization (Lamport, 1987; Simons et al., 2006)
- The sender node attaches a bit-vector to each sent packet.
- The receiver deattaches the bit-vector of each received packet and firstly, caches it to its bit-matrix; secondly, attaches it to the corresponding acknowledgement to be sent to the source/sender.
- The sender deattaches the bit-vector of each received acknowledgement and add it to the corresponding bit-matrix of the sending node.
- Later, the two parties, the sender/the receiver, trigger the mining algorithm to mine bit-matrices using either Mining Rate or Step as a threshold. The resultant MFS obtained, is used as the secret key. Full details of the steps can be obtained from algorithm 2.

Blackhole attacks do not affect the new $KDTM$. Blackholes either dump the data packets on their way to the destination or dumps the acknowledgement on their way to their source. In the first scenario, the bit-matrices at the end nodes are not affected and so is the MFS (key) obtained from them. In the second scenario, because the source does not receive the ACK the source retransmit the packet. At the destination, since the retransmitted packet has the same id as the previous packet a swap is performed in the corresponding entries in the bit-matrix.

This swap is equivalent to dropping one of the two similar bit-vectors in the bit-matrix. Since there is utterly no difference between the source and the destination matrices the same $MFSs$ (key) are obtained.
Wormhole attacks do not affect $KDTM$. Wormhole leaks routed packets at a node to the outside world. Still the $MFSs$ built from the leaked packets is not the same as that of the end nodes because not all traffic from the source to the destination pass through the same route.
$KDTM$ is immune to Man-in-Middle (MIM) attacks. In passive MIM attack, the malicious node just builds and mines its bit-matrix, however, the resultant MFS obtained is different from that of the end nodes. Still in this situation, the MFS obtained at the end nodes is not affected because MIM does not alter the bit-vector of both passing data packets and passing ACK. Two scenarios are observed in active MIM attacks. The first scenario, the malicious node forges/modifies the bit-vector of the passing data packets. This means the same alteration is reflected in both the bit-matrices of the end nodes. In the second scenario, the MIM alters the bit-vector of the passing ACK. This means the same change is induced in the source bit-matrix but not in the destination bit-matrix. The difference induced between source and destination bit-matrices is insufficient, because a small number of ACK pass through the same route; and therefore, the same MFS obtained at the end nodes.
Notably, active MIM can be identified through checking of bit-vector by routing nodes before sending it to the next node; and if any node discover that its bit (or the bits of her neighbors who have not received the packet) is changed, then this node should send a warning message to the other nodes in the $MANET$ that there is an active MIM in the network.
Simulation results show that $KDTM$ is tolerant, in that adding/deleting bit-vectors randomly to/from bit matrix up to 30 % does not change the resultant MFS. Further more, $KDTM$ allows concatenating several $MFSs$ or keys in a bid to develop a stronger key.
$KDTM$ may applies Nitin's watch dog and Pathrater concepts to eliminate malicious nodes in the transmission range of the end nodes so that the extracted key is not compromised (Kyasanur & Vaidya, 2003).
$KDTM$ is a new cross layer key distribution scheme, which extracts MFS from network layer to be used in other layers, for instance, the application layer.

6.3 Key revocation

Key disclosure is very frequent in $MANET$. There is no guarantee that the route between the communicating nodes is free of malicious nodes.
In contrast to using static long-term keys, dynamic short-term cryptographic keys can be used to minimize the availability of ciphertext, encrypted with the same key, and therefore, making it difficult to compromise the key (Menezes et al., 1996). Accordingly, key renewal is compulsory to reduce the amount of disclosed packets in case the key is compromised. In the new method, key renewal, not affected by any other factor and is very simple because the key is mined as long as there is traffic, may be done at any time.
Key can be changed periodically between the two communicating nodes. The parameters such as Support σ, Mining Rate Δ and step threshold λ may be changed to mislead the MIM. This is somehow similar to frequency hopping in wireless communication used for security purpose.
The next two sections analyze mathematically and experimentally the new framework.

$n = 0$ $C(0,0)$

$n = 1$ $C(1,0)$ $C(1,1)$

$n = 2$ $C(2,0)$ $C(2,1)$ $C(2,2)$

$n = 3$ $C(3,0)$ $C(3,1)$ $C(3,2)$ $C(3,2)$

... ...

$n = n$ $C(n,0)$ $C(n,1)$ ... ... $C(n,i-1)$ $C(n,i)$

Fig. 7. Pascal triangle

6.4 Mathematical analysis of the new framework

One of the main features of Apriori algorithm is tolerance, in the sense that arbitrarily adding some rows (bit-vectors) with random values to the data set (bit-matrix) does not affect the end result (outcome), and therefore, the same MFS is obtained. Further more, deleting some rows (bit-vectors) randomly from a data set (bit-matrix), does not change the output of the algorithm. At the same time, it is very difficult to guess the output of the algorithm without acquiring the whole bit-matrix.

The algorithm can be applied on three different types of traffic. The first type is the data traffic. The algorithm extracts the $MFSs$ from the bit-matrix of bit-vectors of data packets. The second type is the acknowledgement traffic and the third type is a mixture of data and acknowledgement packets.

Consider a $MANET$ with a set of n nodes. The output of Apriori algorithm is $MFSs$ in an increasing order and without repetition. The number of ways to form MFS of length i is:

$$C(n,i) \tag{1}$$

i \ n	100	150	200	250	300	350	400	450	500	550	600	...
03	2^{18}	2^{20}	2^{21}	2^{22}	2^{23}	2^{23}	2^{24}	2^{24}	2^{25}	2^{25}	2^{25}	...
04	2^{23}	2^{25}	2^{27}	2^{28}	2^{29}	2^{30}	2^{31}	2^{31}	2^{32}	2^{32}	2^{32}	...
05	2^{27}	2^{30}	2^{32}	2^{33}	2^{35}	2^{36}	2^{37}	2^{38}	2^{38}	2^{39}	2^{39}	...
06	2^{31}	2^{35}	2^{37}	2^{39}	2^{40}	2^{42}	2^{43}	2^{44}	2^{45}	2^{46}	2^{46}	...
07	2^{35}	2^{39}	2^{42}	2^{44}	2^{46}	2^{47}	2^{49}	2^{50}	2^{51}	2^{52}	2^{52}	...
08	2^{39}	2^{43}	2^{47}	2^{49}	2^{51}	2^{53}	2^{54}	2^{56}	2^{57}	2^{58}	2^{58}	...
09	2^{42}	2^{47}	2^{51}	2^{54}	2^{56}	2^{58}	2^{60}	2^{61}	2^{63}	2^{64}	2^{64}	...
10	2^{46}	2^{51}	2^{55}	2^{59}	2^{61}	2^{63}	2^{65}	2^{67}	2^{68}	2^{70}	2^{70}	...
11	2^{49}	2^{55}	2^{60}	2^{63}	2^{66}	2^{68}	2^{71}	2^{72}	2^{74}	2^{76}	2^{76}	...
12	2^{52}	2^{59}	2^{64}	2^{68}	2^{71}	2^{73}	2^{76}	2^{78}	2^{79}	2^{81}	2^{82}	...
13	2^{55}	2^{63}	2^{68}	2^{72}	2^{75}	2^{78}	2^{81}	2^{83}	2^{85}	2^{87}	2^{87}	...
14	2^{58}	2^{73}	2^{72}	2^{76}	2^{80}	2^{83}	2^{85}	2^{88}	2^{90}	2^{92}	2^{93}	...
15	2^{61}	2^{76}	2^{76}	2^{80}	2^{84}	2^{87}	2^{90}	2^{93}	2^{95}	2^{97}	2^{98}	...
16	...	...	...	...	...	...	...	...	...	...	...	...

Higher Security

Higher Security

Table 5. A combinatoric relationship (C(n, i)) between n and i, where $n \equiv$ number of nodes and $i \equiv$ length of MFS.

Accordingly, all the possible ways to form an MFS of variable length i is:

$$C(n,2) + C(n,3) + ... + C(n,i) + ... + C(n,n-1) + C(n,n) \quad (where\ 2 \leq i \leq n) \tag{2}$$

See figure 7, the sum of the nth row of Pascal triangle is given by (Mott et al., 1992):

$$C(n,0) + C(n,1) + C(n,2) + \ldots + C(n,i) + ... + C(n,n-1) + C(n,n) = \quad 2^n \tag{3}$$

From 2 and 3, the total number of ways is:

$$C(n,2) + \ldots + C(n,i) + ... + C(n,n-1) + C(n,n) = 2^n - (n+1) \tag{4}$$

If $i = 2$, then the source and the destination are neighbors, that means no intermediate nodes.
If $i = n$ then the topology is chained.
Equation 4 assumes that the MFS may contain any number of nodes not exceeding n. In fact, this may be correct in one case only, a chain network topology. For example, queue of soldiers following their commander.
The number of routing nodes related to several factors, namely the routing protocol, sending/receiving range, and so on.

6.5 Experimental analysis of the new framework

In this section, the length of $MFSs$ that are used as tokens (keys), is measured experimentally. The $NS2$ simulator is utilized to generate different scenarios. Same parameters that are used in sections 4 and 5, and listed in table 4, are used in this section except for the density of nodes. In reference to the density of nodes in $MANET$, Royer (Royer et al., 2001) shows that the optimum number of neighbors, for 0 m/s mobility or stationary nodes, is around seven or eight per node. This number differs only slightly from what Kleinrock proved for a stationary network (Kleinrock & Silvester, 1978). The density of nodes in wireless network is given by:

$$Density\ (8\ for\ optimal) = n * (\pi * R^2)/(X * Y)$$

where R is the radio transmission range of the node; X and Y are the dimensions of the terrain area, whose area is defined by product $X * Y$.
Tables 5 and 6 show that the bigger the size of MFS, the safer or more secure is the key obtained. In reference to table 6, the evaluation of average size of MFS eliminates short distances, i.e., distances less than five nodes for $AODV$ and DSR protocols.
For example, the average length of the key (MFS) is $i = 15$, which corresponds to the strength of the key of $C(300,15) = 2^{84}$, using the following parameters for simulation: NMS = 10 m/s; mining rate Δ=5 s; number of nodes = 300; terrain area = 2700×2700 m^2; Support σ=40 %; routing protocol is DSR; and data traffic.

					The Average Size of MFS (token)											
					Support = 40%			Support = 50%			Support = 60%			Support = 70 %		
Speed (m/s)	Time (s)	Nodes (#)	Area (m × m)	Protocol	Data	Ack	Ack-Data	Data	Ack	Ack-Data	Data	Ack	Ack-Data	Data	Ack	Ack-Data
10	02	200	2200 X 2200	AODV	8	8	8	7	8	7	7	7	7	6	6	6
				DSR	11	11	11	10	10	10	10	7	7	10	6	6
		250	2400 X 2400	AODV	7	8	8	7	7	6	6	6	5	5	5	5
				DSR	12	12	12	12	12	10	10	11	8	10	10	6
	05	300	2700 X 2700	AODV	7	7	7	6	7	7	6	6	6	5	5	5
				DSR	15	15	13	14	13	13	14	13	12	12	11	12
		350	3000 X 3000	AODV	7	7	8	7	6	7	6	6	6	5	5	5
				DSR	11	12	11	11	10	9	10	9	8	9	6	8
		400	3200 X 3200	AODV	13	12	13	13	13	12	13	12	11	12	12	11
				DSR	15	15	15	15	15	15	15	15	15	15	15	14
		450	3500 X 3500	AODV	14	14	13	13	13	13	13	13	13	12	12	11
				DSR	17	15	14	17	14	13	16	14	13	14	13	11
01	02	100	1500 X 1500	DSDV	5	5	5	5	5	5	5	5	4	5	5	3
02					6	5	6	6	5	6	6	5	3	6	5	3
03					4	4	4	4	4	4	4	4	3	4	4	2
04					5	5	5	5	5	4	5	5	3	5	4	2
05					5	4	5	4	4	4	4	4	3	4	4	2

Table 6. The average length of MFS.

6.6 Outstanding features of the new Scheme

Several features make the new scheme more effective, more flexible, more tolerant and more secure than the present key distribution schemes in $MANET$. These features include:

- Robustness: The protocol is flexible and works in all circumstances, In other words, the absence of any number of nodes in the network topology at any time does not affect the the new protocol. All nodes in other schemes, such as schemes proposed by (Becker et al., 1998; Burmester & Desmedt, 1994; Kim et al., 2001), should be online before the key establishment process is completed (Chan, 2004).
- Transparency: The new scheme is transparent and works in all scalable routing protocols.
- Packet Size Independence: The new security protocol is independent of the packet size and type. In other words, it operates on all types of traffics, such as data, acknowledgement and control.
- Key Revocation and Renewal: The key can be renewed or removed any time even before its expiry time. These activities reinforce the security of the key.
- Overhead at Intermediate Nodes: The new scheme has low overhead on intermediate nodes, achieved through eliminating cryptographical checking of packets at intermediate nodes. The present schemes which use public key cryptography have high overhead on intermediate nodes.
- Scalability: The new scheme allows the number of nodes to be adjusted. Notably, the bigger the number of nodes in the network the bigger the number of ways to choose $MFSs$ and the higher the security.
- Time and Space Complexities: Experimental results of the new protocol show that the time-complexity of the protocol for $MANETs$ is of second order. These complexities depend

directly on the number of node ($MANET$ size), the distance (in terms of number of nodes) between the communicating nodes, and the speed of ARM algorithms used. The space complexity is Sizeof(bit-vector) * Numberof (bit-vectors), where bit-vectors is equivalent to the number of contributing packets.

– Message Complexity: The new scheme has a message complexity of zero for all routing protocols. For source routing protocols such as DRS , which need not attach the bit-vector at all because each data packet has its route; still the message complexity is zero. Even for other protocols the complexity is zero because the bit-vector is attached to packets, and therefore, no security-dedicated packets are sent.

– Fault Tolerance: The failure of a number of nodes does not affect the new protocol because the same bit-entries are dropped from all bit-vectors.

– Adjustability: The new scheme is adjustable. For instance, Apriori is tunable through the Support parameter of MFS, size of bit-matrix and bit-vector extraction time. It is not necessary to attache bit-vector to each packet.

7. Conclusion and future research directions

$KDTM$, a cross layer scheme, shows that $MANET$ traffic in the third layer is raw material that can be mined and utilized in other layers. In addition, the scheme shows how to collect dynamic data from complex and chaotic $MANET$ with large population of mobile nodes and convert it into knowledge. The algorithm mines the MFS patterns through ARM technique employing two methods TAR and SAR mining.
The new concepts generated by $KDTM$ and this chapter as a whole can be extended in several ways. Described below are some of the possible enhancements and extensions:

– Security Enhancement: $MANET$ mining techniques can be used in identifying malfunctioning or blackholes or compromised nodes in $MANETs$ through analyzing the $MFSs$. Such nodes, if identified by a number of other nodes in $MANET$, are discarded/excluded from the list of trusted nodes.

– Maximizing the Network Life Span: Energy conservation is of paramount importance in $MANET$, therefore, uniform energy consumption of nodes increases considerably the lifetime of the network. MFS can be used to identify active and dormant nodes. Dormant nodes in $MANET$ increase the workload on active nodes and thereby decreasing their lifespan. It is therefore evident that decreasing the number of dormant nodes translates into increasing the life span of the $MANET$. Accordingly, $MFSs$ may be considered as a life span metric.

– Load Balancing: Heavily-loaded nodes may become a bottleneck that lowers the network performances through congestion and longer time delays. $MFSs$ can be used as an indicator to avoid over utilized nodes and select energy rich nodes for routing.

– Activity Based Clustering: Similar to other clustering metrics, like power, distance and mobility, among others, node activity levels can be considered as a metric for cluster formation. Nodes belonging to one MFS (pattern) are most likely connected and can be used as a cluster. Another metric for clustering is the Support parameter, i.e., the higher the Support level the higher the relationship among the routing nodes.

– Routing and Multicasting: Nodes belonging to one MFS are most likely connected. Accordingly, delivery or sending of packets is guaranteed amongst nodes in the same MFS.

– Applying Different Association Rules Mining Types: This chapter applies positive association rules mining techniques that mine binary attributes and considers that the utilities of the itemsets are equal. The frequency of an itemset may not be a sufficient indicator of interest. Non-boolean fuzzy association rule mining such as weighted/utility association rules, may find and measure all the itemsets whose utility values are beyond a user specified threshold that suggest different decisions. For example, in battlefield a commander can give higher weight/utility to his higher rank commanders and less weight to soldiers in order to find the hidden relationships (rules) amongst them. These rules may give an idea about soldiers who are in touch with each other, with commanders, and so on.
– Wireless Sensor Networks (WSN) has the inherent characteristics of *MANETs*, and therefore, the aforementioned benefits of using *MFS* in *MANETs* may also be applicable in *WSN*.

8. References

Agrawal, R., Imielinski, T. & Swami, A. (1993). Mining association rules between sets of items in large databases, *Proceeding of the 1993 ACM SIGMOD International Conference on Management of Data*, ACM, New York, NY, USA, Washington, D.C., United States, pp. 207–216.

Agrawal, R. & Shafer, J. C. (1996). Parallel mining of association rules, *IEEE Transactions on Knowledge and Data Engineering* 8(6): 962–969.

Asuncion, A. & Newman, D. J. (2007). UCI machine learning repository. URL: *http://www.ics.uci.edu/∼mlearn/MLRepository.html*

Becker, K., Wille, U. & Wille, U. (1998). Communication complexity of group key distribution, *Proceedings of the 5th ACM conference on Computer and communications security*, ACM New York, NY, USA, San Francisco, California, United States, pp. 1–6.

Burmester, M. & Desmedt, Y. (1994). Vol. 950/1995 of *Lecture Notes in Computer Science*, Springer Berlin, Heidelberg, chapter A Secure and Efficient Conference Key Distribution System, p. 275.

Chan, A. C. F. (2004). Distributed symmetric key management for mobile ad hoc networks, *Proceeding of IEEE INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, Vol. 4, IEEE Press Piscataway, NJ, USA, Hong Kong, pp. 2414–2424.

Fall, K. (2007). *The NS Manual*, The VINT Project, University of California.

Fard, A. M. & Ester, M. (2009). Collaborative mining in multiple social networks data for criminal group discovery, *International Conference on Computational Science and Engineering*, IEEE CS Digital Library, Vancouver, Canada, pp. 582–587.

Frawley, W. J., Piatetsky, G. & Matheus, C. J. (1992). Knowledge discovery in databases: An overview, *AI Magazine* 13(3): 57–70.

Fumy, W. & Landrock, P. (1993). Principles of key management, *IEEE Journal on Selected Areas in Communications* 11(5): 785–793.

Ghoreishi, S. M. & Analoui, M. (2009). Design a secure composite key-management scheme in ad-hoc networks using localization, *International Journal of Computer Science and Network Security* 9(9): 35–49.

Greis, M. (2007). *Tutorial for the Network Simulator NS2*, http://www.isi.edu/nsnam/ns/tutorial/.

Hegland, M. (2005). Wspc/lecture notes series: The apriori algorithm - tutorial, *Technical report*, Australian National University, CMA, John Dedman Building, Canberra ACT

0200, Australia.

Hofmann, M. (2003). *The development of a generic data mining life cycle (dmlc)*, Master's thesis, MSc. in Computing Science , Dublin Institute of Technology, Duplin, USA.

Jabas, A., Abdulal, W. & Ramachandram, S. (2010). An efficient and high scalable key distribution scheme for mobile ad hoc network through mining traffic meta-data patterns, *Fifth IEEE International Conference on Network and System Security (IEEE NSS'10)*, IEEE CS Digital Library, Melbourne, Australia.

Jabas, A., Garimella, R. M. & Ramachandram, S. (2008a). Manet mining: Mining step association rules, *Fifth IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE MASS'08)*, IEEE CS Digital Library, Atlanta, Goergia, USA, pp. 589–594.

Jabas, A., Garimella, R. M. & Ramachandram, S. (2008b). Manet mining: Mining temporal association rules, *Third International Workshop on Intelligent Systems Techniques for Ad hoc and Wireless Sensor Networks (IEEE IST-AWSN 2008), Sydney, Australia*, IEEE CS Digital Library, Sydney, Australia, pp. 765–770.

Jabas, A., Garimella, R. M. & Ramachandram, S. (2008c). Proposing an enhanced mobile ad hoc network framework to the open source simulator ns2, *Mosharaka International Conferences on Communications, Computers and Applications (IEEE MIC-CCA'08)*, IEEE CS Digital Library, Amman, Jordan, pp. 14–19.

Javaheri, S. H. (2007). *Response modeling in direct marketing, a data mining based approach for target selection*, Master's thesis, Continuation Courses, Marketing and e-commerce, Department of Business Administration and Social Sciences, Division of Industrial marketing and e-commerce.

Joe, B. (2009). Do association rules represent supervised or unsupervised learning, *Technical report*. http://wardselitelimo.com/2009/07/02/.

Kim, Y., Perrig, A., & Tsudik, G. (2001). Communication-efficient group key agreement, *In 17th International Information Security Conference (IFIP SEC01)*, Kluwer Academic Publishers Norwell, MA, USA, Paris, France, pp. 229–244.

Kleinrock, L. & Silvester, J. (1978). Optimum transmission radii for packet radio networks or why six is a magic number, *Proceedings of the IEEE National Telecommunications Conference*, IEEE CS Digital Library, Birmingham, Alabama, p. 4.3.14.3.5.

Kyasanur, P. & Vaidya, N. H. (2003). Detection and handling of mac layer misbehavior in wireless networks, *International Conference on Dependable Systems and Networks (DSN'03)*, IEEE CS Digital Library, San Francisco, California, pp. 173–182.

Lamport, L. (1987). Synchronizing time servers, *Technical report*, Digital Equipment Corporation. Systems Research Center.

Luo, H., Kong, J., Zerfos, P., Lu, S. & Zhang, L. (2003). Ursa: Ubiquitous and robust access control for mobile ad-hoc networks, *58th IEEE Vehiclular Technology Conference VTC'03*, Vol. 3, IEEE Press Piscataway, NJ, USA, Orlando, Florida, USA, pp. 2137–2141.

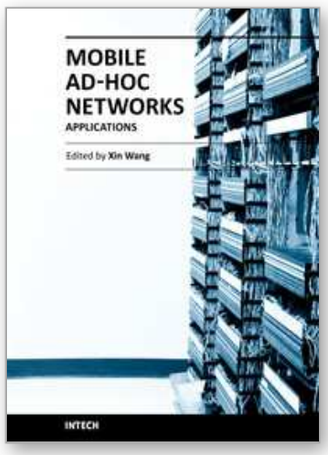
Menezes, A., Oorschoot, P. V. & Vanstone, S. (1996). *Handbook of Applied Cryptography*, CRC Press, San Antonio, Texas.

Mott, J. L., Kandel, A. & Baker, T. P. (1992). *Discrete Mathematics for Computer Scientists and Mathematicians*, Reston Publishing Company, Inc.

ns2 (2009). The network simulator (ns2), Information Sciences Institute.
URL: *http://nsnam.isi.edu/nsnam/index.php/Main-Page*

Olson, D. L. & Delen, D. (2008). *Advanced Data Mining Techniques*, Springer, Verlag Berlin

Heidelberg.
Post, G. V. (2005). *Database Management Systems: Designing And Building Business Applications*, McGraw-Hill, Irwin.
Pujari, A. K. (2001). *Data Mining Techniques*, Universities Press, 3-6-747/1/A and 3-6-754/1, Himayatnagar, Hyderabad 500 029, Andhra Pradesh, India.
Rashmi (2009). Manet (mobile adhoc network), http://www.saching.com/Article/MANET--Mobile-Adhoc-NETwork–/334 [Access time: 20 Oct., 2009].
Robinson, J. A. (2007). Connecting the edge: Mobile ad-hoc networks (manets) for network centric warfare, *Technical report*, AIR UNIV MAXWELL AFB, Maxwell-Gunter Air Force Base Montgomery, Alabama, USA.
Royer, E. M., Melliar-Smith, P. M. & Mosery, L. E. (2001). An analysis of the optimum node density for ad hoc mobile networks, *IEEE International Conference on Communications, ICC*, Vol. 3, IEEE CS Digital Library, Helsinki, Finland, pp. 857–861.
Santoro, N. (2007). *Design and Analysis of Distributed Algorithms*, John and Wiley and Sons, Inc. Hoboken, New Jersey, Hoboken, New Jersey.
Simons, B., Welch, J. L. & Lynch, N. (2006). *Fault-tolerant distributed computing*, Vol. 448/1990 of *Lecture Notes in Computer Science*, Springer, Berlin / Heidelberg, chapter An overview of clock synchronization, pp. 84–96.
Simovici, D. A. & Djeraba, C. (2008). *Mathematical Tools for Data Mining, Set Theory, Partial Orders, Combinatorics*, Springer-Verlag Limited, Uk, London.
Tan, P.-N., Steinbach, M. & Kumar, V. (2006). *Introduction to Data Mining*, Addison-Wesley.
Yao, J., Li, X. & Jia, L. (2003). A new method based on ltb algorithm to mine frequent itemsets, *International Conference on Machine Learning and Cybernetics*, IEEE CS Digital Library, Xian, China, pp. 71–75.
Yi, S. & Kravets, R. (2003). Moca: Mobile certificate authority for wireless ad hoc netwroks, *Proc. of the 2nd Annual PKI Research Workshop (PKI)*, National Institute of Standards and Technology, Gaithersburg, USA.
Yi, S. & Kravets, R. (2004). Composite key management for ad hoc networks, *The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services. MobiQuitous'04*, IEEE CS Digital, Boston, USA, pp. 52–61.



Mobile Ad-Hoc Networks: Applications

Edited by Prof. Xin Wang

ISBN 978-953-307-416-0
Hard cover, 514 pages
Publisher InTech
Published online 30, January, 2011
Published in print edition January, 2011

Being infrastructure-less and without central administration control, wireless ad-hoc networking is playing a more and more important role in extending the coverage of traditional wireless infrastructure (cellular networks, wireless LAN, etc). This book includes state-of the-art techniques and solutions for wireless ad-hoc networks. It focuses on the following topics in ad-hoc networks: vehicular ad-hoc networks, security and caching, TCP in ad-hoc networks and emerging applications. It is targeted to provide network engineers and researchers with design guidelines for large scale wireless ad hoc networks.

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen