Foreword

szabo@hit.bme.hu

Our „Infocommunications Journal" is published by the Scientific Association for Infocommunications (HTE), a Sister Society of IEEE.

Until the end of 2008, we published English issues twice a year, which were compiled mostly from the best research papers published in Hungarian during the preceding half a year period. As of January 2009, we are going to increase the number of English issues to four, with the objective to become a quarterly international journal. We publish original research papers in the aforementioned areas after rigorous peer reviewing process. In this first issue of the year, I would like to announce our International Advisory Committee that will support the Hungarian editorial team in maintaining the quality of published papers.

The scope of our journal spans a wide range of technical areas, covering the large variety of topics of interest of our Society, including the „classical" telecommunication topics, information technology related to telecommunications, media technologies and media communications, thus representing the process of convergence of telecommunications, digital broadcasting and information technology. Our scope also includes some inter-disciplinary areas such as economics, marketing, regulation and management aspects of infocommunications, as well as the society-related issues.

„Infocommunications Journal" is intended to become a recognized international publication forum for researchers not only from Hungary but also from neighboring countries, and in principle, from all over the world.

Our International Advisory Committee

In this issue

This is the last issue in which we publish English versions of research papers, carefully selected from the preceding five Hungarian issues. Being a selection, the papers' topics span a wide range of issues of current interest as the reader can see from the short summaries below. We present the papers in the order of their original publication times in the respective Hungarian issues from August through December 2008. The last paper was accepted from open call.

The paper by *Zoltán Czirkos and Gábor Hosszú* titled "P2P based intrusion detection" presents a novel security method. The software entities utilizing this method create a peer-to-peer application level network to share information about intrusion attempts detected. Data collected this way is then used to enhance the protection of all participants. The system is completely decentralized, thus it remains functional over an unstable network or when many peers are attacked at once. The stability of the overlay and the broadcast algorithms are both analyzed in this article.

Nowadays, nearly all car manufacturers can build a cruise control system (tempomat) in their cars. In some top-end cars also the distance of objects in front of the car is measured and the tempomat tries to maintain the following distance. In these adaptive cruise control systems, however, the detection range and field of view of the sensors are limited. *Balázs Mezny, Péter Laborczi and Géza Gordos* present in their paper "Ad-hoc adaptive cruise control algorithm" an adaptive cruise control system, which sets the speed of the vehicle according to messages distributed over an ad-hoc wireless network. Wireless communication eliminates the problems caused by bad visibility or being out of line of sight. The distributed messages contain the exact position, speed and direction of the sender vehicle.

The paper "Reliable Gossiping in Inter-Vehicle Communication" by *Miklós Máté and Rolland Vida* also demonstrates the importance of intelligent transport control systems. Due to the increasing traffic density in urban areas, a computer-aided robust collision avoidance and traffic control system should be established, based on decentralized inter-vehicle communication. Vehicles group themselves into a special ad hoc network with high mobility and low link reliability and novel ad hoc routing solutions are needed for these special conditions. The scheme proposed in the paper is a location aided gossiping protocol, which concentrates the information spreading to areas where it is most likely to be useful.

Andrea Farkasvölgyi, Ákos Németh and Lajos Nagy deal with MIMO antenna systems that are essential for good performance of indoor wireless networks. The authors present simulation and measurement results for a 3x3 MIMO antenna system, with the aim of maximizing the MIMO channel capacity for indoor environment. The dependence of the channel capacity on the antenna position is analyzed by simulations. The effect of the frequency dependence of the antenna system for the channel capacity also examined in case of conjugate-matching and non-conjugate-matching.

In the paper "A Client-driven Mobility Frame System – Mobility Management from a New Point of View", the authors, *Benedek Kovács and Péter Fülöp* introduce a new mobility management approach. The main idea is that not the network but the mobile node should manage the mobility for itself, the network nodes provide just basic services for mobile entities: connectivity and administration. A protocol called Client-based Mobility Frame System (CMFS) was constructed for this mobility environment. Examples of mobility management approaches such as the centralized and hierarchical or cellular-like ones are also defined and hints are given what kind of algorithms might be implemented upon the Client-based Mobility Frame System. After the theoretical analysis simulations show the applicability of the new protocol framework.

Kristóf Aczél and István Vajk in their paper "Note-based sound source separation of polyphonic recordings" address an important problem of the decomposition of a polyphonic musical piece to separate instrument tracks which has always been a challenge. Isolating the tracks is out of reach of today's technology. The paper proposes a novel method for the separation of monophonic musical recordings. The architecture of the proposed separation system is given. It uses samples of real instruments for regaining the missing data, thereby allowing for the separation and correction of recordings that cannot be retaken.

The paper titled "Home access network model specifications" by *Izabela Krbilová, Vladimír Hottmar and Bohumil Adamec* investigates a home network configuration consisting of residential gateway and a number of intelligent peripheral devices capable of autonomous activity. A queuing model is built by means of bulk service in a closed circuit which circulates constant number of requests. Performance and time characteristics of peripherals communicating with residential gateway are determined. The presented results illustrate mutual dependence of the number of network peripherals and time characteristics determining operation of the network.

Csaba A. Szabó
Editor-in-Chief

P2P based intrusion detection

ZOLTÁN CZIRKOS, GÁBOR HOSSZÚ

Budapest University of Technology and Economics, Department of Electron Devices
hosszu@nimrud.eet.bme.hu

Keywords: Peer-to-Peer, P2P, intrusion detection, NIDS, overlay

This paper presents a novel security method. The software entities utilizing this method create a peer-to-peer application level network, which is then used to share information about intrusion attempts detected. Data collected this way is then used to enhance the protection of all participants. The system is completely decentralized, thus it remains functional over an unstable network or when many peers are attacked at once. The Kademlia P2P overlay is found to be the most suitable to create such a network. The stability of the overlay and the broadcast algorithms are both analyzed in this article.

1. Introduction

A number of security systems are presented in the literature, which run instances of applications and different hosts which communicate among each other [3], [4]. The novelty of the solution developed by us is that its nodes create a *P2P* (Peer-to-Peer) overlay on the Internet. The organization is automatic and does not require any user intervention. This network model provides great stability, which is needed by the nodes to quickly and reliably exchange information. The system can remain operative even on unreliable networks due to attacks and link failures. The software that implements this solution is named *Komondor*, after a famous Hungarian shepherd's dog.

Section 2 of our paper presents related work and systems, which are similar to ours. P2P overlay networks are explained, and also two distributed intrusion detection systems are mentioned. Section 3 shows the design goals and internal operation of the Komondor system. Section 4 explains the Kademlia overlay network in detail, this is necessary to understand why it is the most suitable overlay to use as a substrate of Komondor. Section 5 summarizes our results and experience collected until now.

2. Related work

2.1. P2P overlays

Application level networks (or sometimes called *overlays*) with *peer-to-peer* (P2P) topology can be structured or unstructured.

The *nodes (peers)* of unstructured networks can easily be dispensed. The network handles joining and leaving nodes in a very flexible way. The usual queries (data lookups) are also processed by the nodes, forwarding the lookup query to each other. Examples for these unstructured networks are *Gnutella, Freenet* and *FastTrack*.

Structured peer-to-peer networks usually implement a *distributed hash table* (DHT). These networks store key-value pairs and enable the users to quickly lookup a value associated with a precisely given key. In contrast to unstructured networks, logical links between nodes are determined by a set of rules; the topology of the network is exactly defined. Every stored piece of information (or file) is sent to a precisely selected node. Nodes are assigned a *node identifier (NodeID)* chosen from a fairly large interval of numbers (for example, 160 bits).

Similarly, each stored piece of data is assigned a key, which can be a hashed value of a file name, for example. The output of the hash function is in the same domain as the node identifiers. Every node stores key-value pairs, which have their hashed keys closest to its own NodeId. So if one knows the key, it is easy to find the node storing the value associated with that key. This is called *consistent hashing* [8,9].

Structured networks are different from each other in terms of topology, routing algorithms and the distance function (which calculates the distance between two identifiers, or a hashed key and an identifier.)

2.2. Distributed intrusion detection

The usual distributed intrusion detection systems deployed on networks are centralized, and are generally used for collection of data only [4]. Applications which are decentralized and also capable of intervention (intrusion prevention) are presented only lately.

The prevention system named PROMIS (and its ancestor, Netbiotic) uses the JXTA framework to enable nodes to exchange information of detected intrusion attempts [12]. It builds an overlay network which is partially centralized. The nodes entering the PROMIS network receive information about the number and rate of suspicious events detected, and they tune the security level of the Web browser built into the operating system accordingly. This method gives a general protection against malicious applications, but also reduces usabi-

lity of the system. The approach is somewhat similar to the epidemic prevention measures used in daily life.

The spam (e-mail junk) filtering system named Spamwatch is built on the Tapestry network [13]. The application is a plugin for the e-mail client. The hashed content of the e-mail messages marked by users as junk mail are stored in a distributed hash table; the same message on another user's computer can automatically be discarded. By using the DHT, the lookup of a message is fast, and generates little network traffic.

3. The proposed system

In the Komondor system, detection of intrusion attempts is distributed, by means of a DHT's based on the Kademlia network [1]. The following goals were important during the development of the system:

- Building a stable overlay network to exchange information.
- The data should be exchanged as quickly as possible.
- Decentralization of the system, enabling nodes to be missing.
- Masking the security holes of nodes based on intrusion attempts detected.

The several hosts running the Komondor software create a virtual, application level network, which is sometimes called an overlay. The speed of exchanging data about intrusion attempts largely depends on the network model employed. The system is built on a peer-to-peer (P2P) based overlay to ensure decentralization and stability [11], in contrast to the client-server model with a much higher risk of failure.

In the Komondor system, a structured network, a DHT is employed to store data of intrusion attempts. Keys are IP addresses of intruders, values are the information of intrusion attempts. Report of intrusion attempts from a specific attacker will be sent to a single node, as all nodes use the same hash functions. If that node analyzes the reports and sees that the IP address in question belongs to an attacker, it initiates a broadcast message, to alert other Komondor nodes of the possible danger. Every node is interested in receiving information which enables it to strengthen its protection. Compared to PROMIS, the protection built up by Komondor is not general; rather it is against the recognized attackers only.

Detection at multiple points and collection of data can be very efficient. Consider the following example. Let us assume that an attacker is trying to find an open, badly configured SMTP server to send junk e-mail. It tries to connect to many nodes protected by the Komondor system to find out which nodes have an SMTP service at all. One node sees an incoming connection which is immediately lost. From the viewpoint of a single host, an event like this alone does not necessarily indicate an attacker. It could possibly be a really lost connection, caused by some link failure, or an e-mail sending canceled upon the user's request. But if this event is detected on many of the nodes, then it immediately becomes suspicious. In the Komondor system, the IP address of the attacker determines which node will be the collector of information. That node will be responsible for processing data about a specific attacker; so sharing information about every suspicious event is necessary.

The main goal of our research is the investigation of the stability and reliability of the *Kademlia* P2P overlay system, and also the exploration of possibilities of peer-to-peer based intrusion detection. Our Komondor software has Linux and Microsoft Windows versions currently implemented. It uses Snort and the system log files to collect information about events; to create protection, it tunes the firewall of the operating system. Other detection and prevention modules are also planned to be developed in the later versions.

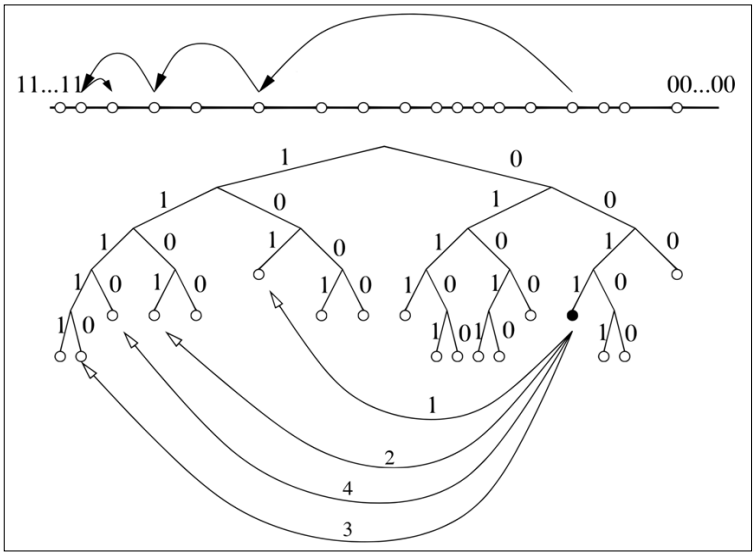
4. Using the Kademlia overlay in Komondor

4.1. The Kademlia overlay system

To investigate the stability and reliability of the Kademlia overlay, and to understand the broadcast messaging system built over the overlay, we outline the topology and internal operation of Kademlia in this section.

Kademlia uses distributed hash tables (DHT's). The nodes participating in a Kademlia network can be represented with a binary tree [5]. Kademlia nodes store the same amount of connection info (IP address, port number) for every subtree. These lists are called *k-buckets* in the original paper. The size of these lists is usually denoted by *k*, which is a system-wide configuration parameter. In a well populated network, taller subtrees usually contain a lot more nodes than *k*, so a node has rela-

Fig. 1. Routing in Kademlia



tively less information about the distant subtrees, as compared to the subtrees in close proximity.

Routing is shown on *Fig. 1.* (In papers about Kademlia, the subtrees which have only one node are usually not shown as a tree, only as a leaf. In the example below, all nodes have 5 bit identifiers.) If the node with the identifier 00110 would want to send a message to the node with ID 11100, it has nothing else to do that send a message to *anyone* in the 1* subtree, who will have greater knowledge of nodes in the 11* subtree and so on.

The order of lookups is shown by the arrows with numbers. The sending of message is completed in *O(log n)* steps this way. The distance of two identifiers is calculated using the *exclusive* or function. The magnitude of the distance between two peers is proportional to the height of the subtree containing both of them. This is why this network can be represented as a binary tree, and why it is called the *XOR topology*. Due to the symmetry of the XOR function, the distribution of incoming and outgoing messages is the same. The routing table of nodes is automatically refreshed by network traffic; so the network strengthens itself with all messages.

Comparing Kademlia to other DHT systems, its unusual property is the freedom of nodes (also requiring Kademlia to use UDP instead of TCP.) To lookup a value associated with the given key, the message is not forwarded from node to node, but rather a node itself finds out who the destination of the message is. This makes handling of replication very easy. A node intending to store a key–value pair does not send data to the closest node to the key; rather it sends it to the *k* closest nodes. By selecting a value for *k*, the stability of the overlay can be tuned. But as we will later see, by selecting $k>1$, the availability of data stored can also be enhanced.

The Kademlia protocol requires nodes to maintain lists of other nodes with at least *k* entries for every subtree. Connection information is refreshed in every hour, if necessary. The value of *k* must be chosen so that it would be very unlikely for all *k* nodes to quit the network in an hour. The nodes quitting the network are *not* required to send their stored key–value pairs to other nodes in the network. So if a node disappears, data stored by it would also be gone, if replication was not implemented. Note that in a DHT, being able to communicate with a node implies being able to retrieve key–value pairs stored by that node. So the level of replication must be the same as the number of nodes in a *k-bucket*. Thus this is the only system-wide configuration parameter needed by Kademlia.

4.2. Reliability of Kademlia

The overlay in the Komondor system is created by a version of Kademlia modified only slightly compared to the original. The conclusions presented in the following sections are also applicable to the original overlay. Our test runs of the Komondor system proved that replication in Kademlia is much more important than in other types of overlay networks, as it is very common in a real environment that nodes cannot connect to each other, due to packet losses, network address translation or other reasons. Therefore it is possible that a key–value pair stored by a single node cannot be reached by others, as some may not be able to connect to it. Replication partially solves this problem. If the pair is stored not only by a single node, but by a range of nodes (ie. *k* nodes), replication increases the probability that at least one node will be able to answer the lookup request. It is also possible that some k-buckets of nodes are not correctly populated with the addresses of other nodes at a time; replication in this case will also solve the problem of unavailability. (The routing tables of structured networks may be incorrect for small intervals, when a lot of nodes join or quit in a short time. This is called *high churn* [10].)

To prove the above statement, we developed a simulator application for Kademlia. It is called *Kadsim*. Although much of the simulations were focused on the Komondor network and its behavior, the results are general and can be applied to other Kademlia-based network, too. Kadsim works the following way. Given a number of nodes, it creates a connectivity matrix, which is essentially the adjacency matrix of the possible communication between nodes.

Also given a message, which is virtually a randomly chosen identifier; in the Komondor system, a hashed value of the attacker's IP address. The most important demand of Komondor against the overlay is that there should be always at least one node, where reports about a specific attacker can be collected. So Kadsim models the case when *all* nodes in the overlay detect some attack from the IP address in question. Every node hashes the IP address, and looks up the resulting identifier in the overlay. Nodes that cannot be reached by some other nodes do not count. (However, they may be reached by others.) Usual DHT networks, storing files for example, work exactly the same way; a given key is looked up in a tight range of nodes near the identifier.

Finishing the simulation, Kadsim sorts the number of messages received by each node, using the distance of NodeID's for the comparison. The results are plotted in *Fig. 2.* Ideally, when there are no network errors, and all links are operational, the resulting function is a single step: the *k* closest nodes to the key get all messages, and others get no messages at all. If there are link failures, the function will be lower and wider (see Fig. 2). For example, if the level of replication is $k=16$, and one of the nodes cannot access other nodes who are the 12th and 15th closest to the key, it will store key at the 16th and 17th closest ones.

If the distribution of link failures is flat, there will be no node in the overlay, where reports of attacks can be fully collected, no matter how high the level of replication is. In such a case, Kademlia would be a very bad choice for overlay topology. Real networks like the Internet are fortunately not like this: link failures are unevenly scattered throughout the network. There are hosts,

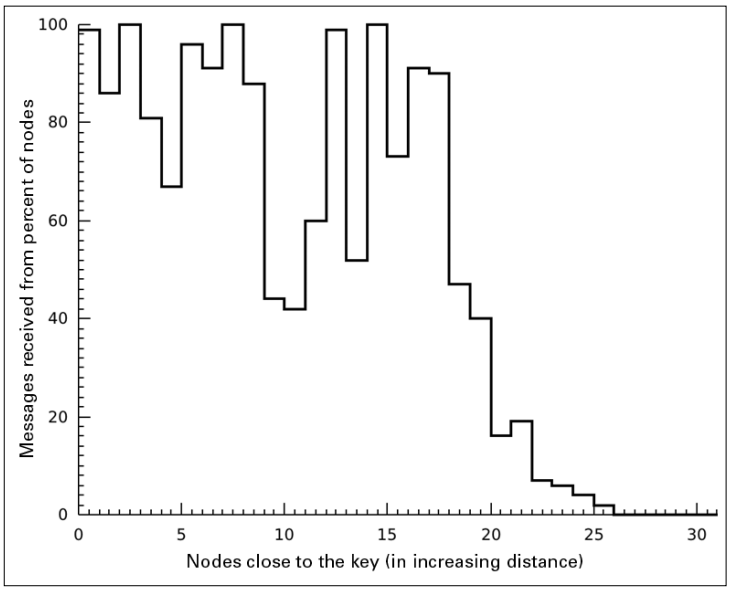


Fig. 2.
Storing keys in a Kademlia overlay; replication is 16-fold, ratio of failing links is 20%

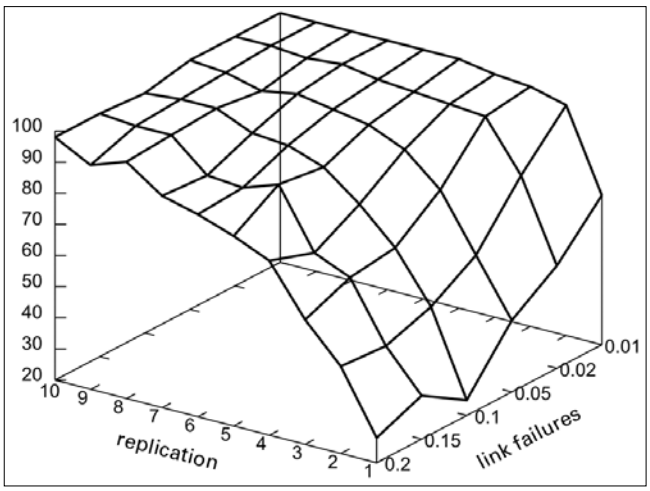
which have public IP address, those are easy to connect to. Others are behind network address translation, and cannot always be reached. The exact distribution can be very different for various networks; Kadsim models the distribution as a polynomial function. When this distribution is not flat, there are nodes who can receive the attack report with very high probability.

Simulation showed that a moderately high level of replication, *k=8* sufficiently ensures that a suitable node will exist in the network with high probability *(Fig. 3)*. For a hundred of nodes this seems to be too much, compared to other P2P networks, but increasing the number of nodes *there is no need to increase replication.* There will be at least one of the selected nodes which are able to communicate with others.

4.2.1. Mathematical Modelling of Link Failures

Nodes joining a DHT overlay usually randomly choose their own node identifier from a very large range of numbers. Also the output of hash functions can also be treated as a random number. The network seems to choose the node responsible for a specific key randomly. This property makes modelling the overlay relatively simple.

Fig. 3.
Ratio of successful lookups in the Kademlia overlay



The error ratio for the node with identifier *m* is given by (1):

$$h(m) = c \cdot \left(\frac{m}{n}\right)^{\alpha}, \qquad (1)$$

where *n* is the total number of nodes ($0 \le m < n$). α sets the distribution of errors $\alpha=2$ for quadratic distribution. *c* is a constant setting the maximum number of errors. These parameters can be selected experimentally, and they depend on the actual size and properties of the underlying physical network.

Function (1) should output an integer number, as the error ratio multiplied by the number of all nodes, $n \cdot h(m)$, is an integer. For a high number of errors, the difference is negligible. Approximations using equation (1) will not be applicable to networks with a very low error rate, where $n^*h(m)$ is almost zero for the whole range of nodes. 0.3 errors have no real-world meaning, only 0 or 1 error.

As the underlying physical network, the Internet is not perfect; we also cannot expect the overlay to be so. Rather we can set a numeric expectation, for example we would like our network to be able to retrieve data in 99% of all cases. If the ratio of allowable errors is β= 1%, the probability of a successful lookup is 1-β, if the inequation $h(m) \le \beta$ holds for a given node. Those are the nodes, which can be accessed from most other ones.

As node identifiers are usually as large as 128 or 160 bits, the range of addresses can be treated as continuous. As all nodes bear randomly selected identifiers, and also the output of hash functions applied to keys seem to be random and evenly distributed over the range of possible identifiers, *m/n* is virtually a random number chosen from the interval [0,1). If we solve the inequality to express *m/n*, we get the number of nodes which match the specified criterion:

$$\frac{m}{n} \le \sqrt[\alpha]{\frac{\beta}{c}} \qquad (2)$$

Let *P'* be the probability of a successful lookup. As $0 \le m/n < 1$ holds, and *m/n* is randomly chosen over the interval, inequality (3) will hold for *P'*:

$$P' \le \sqrt[\alpha]{\frac{\beta}{c}} \qquad (3)$$

If the overlay employs replication, data is stored at *k* different points of the network. So we have *k* chance to choose different random numbers from the interval [0,1). If we manage to choose a suitable number at least once, the lookup will be successful. Calculating the probability of all lookups failing, and subtracting that value from 1, we get (4).

$$P = 1 - (1 - P')^{k} \qquad (4)$$

Formula (4) gives the probability of successful lookups with a given ratio of networks failures. The necessary level of replication can be calculated with the formula. *Fig. 4* shows the ratio of cases when the probability of successful lookups is at least 99% (*1%* failure allowed), as a function of network failures and level of replication.

As one can see, with a relatively high ratio of failing links (*10%*), replication *k=5* is enough to ensure successful lookups. If the overlay consists only of a small number, for example tens of nodes, *k=5* may seem too much. But this *k=5* can be used to any number of nodes. The formula gives results which closely match our simulation; the difference can only be seen for small error rates, as it was expected due to the approximation in (1).

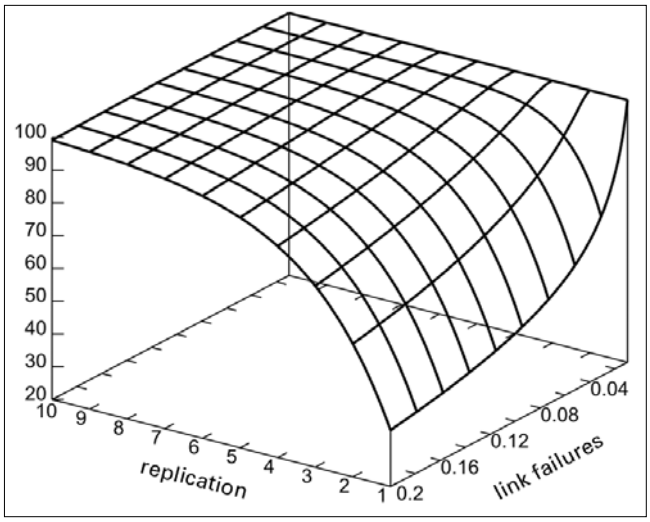


Fig. 4.
Approximated ratio of successful lookups in Kademlia

4.3. Broadcast messages in P2P overlays

Broadcast (one to all) messages in P2P overlays are not very common, due to the very big number of nodes. Usually no algorithm is designed to send these broadcast messages, as this contradicts with the main design goal of scalability. However there are applications which need this type of messaging, Komondor is an example. When a node has collected enough information making sure that an IP address belongs to an attacker, it initiates a broadcast message over the network. Another common application of broadcast messages is implementing lookups for partially given keywords, as this is not an elementary service in DHT networks (for example, one cannot lookup a partially given file name.)

The inherent topology of structured networks can be used to quickly and efficiently deliver broadcast messages. Using the built in topology will always give the best results. One reason for this, that the topology is built such a way that any node can be reached in logarithmically many steps, so the broadcast message will reach all nodes in logarithmic time.

The second is, that during sending the message, there will be no need to create new connections or initiate lookups. The topology can essentially be seen as an *implicit multicast tree*.

The Komondor system is an application where the fast sending of broadcast messages is essential. It is usually very easy to create reliable messaging over an unreliable channel, however detecting a packet loss needs quite long time. According to our tests, the broadcast algorithms presented in our article send the message in a few seconds to all nodes; to detect loss of a packet alone needs more time than this. If we do not try to resend the packets, the simulation will give us the shortest time in which the broadcast can be finished. Using replication, the time can be shorter than it is needed to detect packet loss. Simulating broadcast without resends will also give us the ratio of cases when the broadcast is successful, and is able to keep this time.

We developed three algorithms to send broadcast messages over Kademlia.

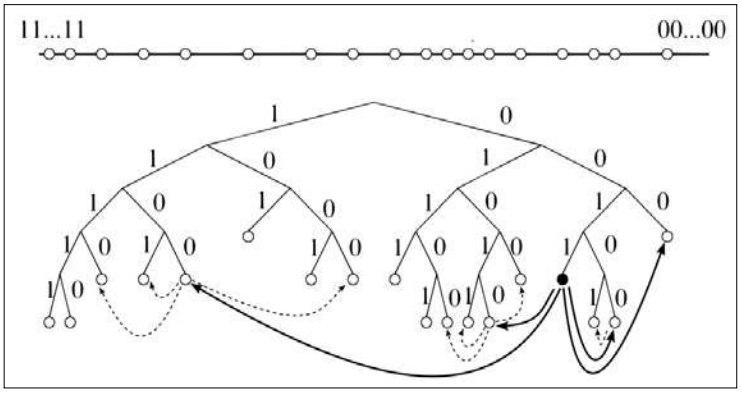
4.3.1. Broadcast Using Flooding

All nodes send received messages to any other nodes they know. As a specific message can be received in duplicates, every broadcast should be tagged with a unique identifier. Known messages are dropped by nodes. This solution is simple, but it generates a lot of network traffic, especially when k-buckets are large. It has no practical use, but is rather a theoretical reference; by simulating this method on an overlay, one can see the time the broadcast requires.

4.3.2. Broadcast Using the Topology

In the second algorithm, every subtree in the Kademlia overlay is assigned a node, which is responsible for broadcasting the message in its own tree *(Fig. 5)*. The node with the identifier (00110, black dot) initiates the broadcast by sending it to one freely chosen node from each of its k-buckets (normal arrows). These nodes are 11000, 01010, 00100 and 00000. The nodes receiving the message are responsible for sending them on in their own subtrees, which are 1****, 01***, 000** and 0010*. This shown using dashed lines. Broadcast using this method will be finished in logarithmic time.

Fig. 5. Broadcast messaging in Kademlia



Nodes forwarding messages must know which subtree they are responsible for. Every message is tagged with a small integer, which denotes the height of the subtree; this shows how many prefix bits the address of the subtree should share with the NodeID. The Kademlia protocol makes sure that at least one node is always known for every subtree; there is no need to maintain an auxiliary routing table for the broadcast.

Messages are forwarded to the subtree and all smaller trees:

```
broadcast(text, height)
  for i=height to number of bits
    if bucket i is not empty, then
    select a random node from bucket i
    send the message to the node: text, i+1
  endif
endfor
```

This method is very cost-efficient as there are no duplicate messages. The number of messages sent grows exponentially, so the complete process takes logarithmic time. Problems can arise when there are packet losses on the network, as not only single nodes, but complete subtrees will miss the broadcast. Messages are actually directed to subtrees in this method: the original sender sends the message to the other half tree, and is itself responsible for his own half tree. Then it sends to the other quarter of the overlay, and is responsible for its own quarter and so on. Every subtree has a single responsible node.

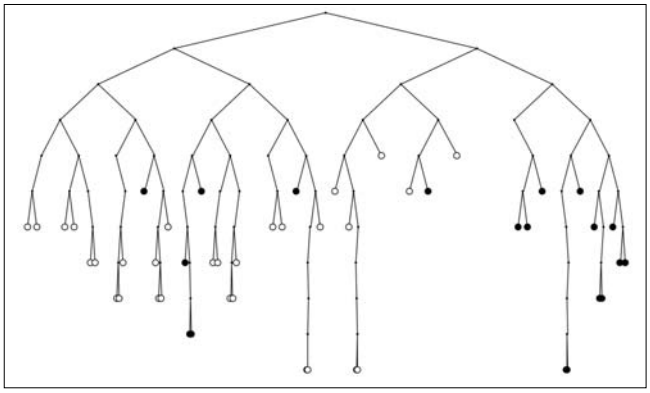


Fig. 6.
Implicit tree broadcast messaging in Kademlia

Fig. 6 shows a simulation of this method. Nodes shown as white dots received the message, while black ones did not. As one can see, there are complete subtrees drawn in black. It is possible for such a message to be lost, which was sent to a high subtree. In a worst case scenario, the number of nodes not getting the message can be more than 50%, *independent* from the packet loss ratio. Although the network is decentralized, this algorithm is not in its essence; as the importance of messages is vastly different, depending on which subtree they are addressed to.

4.3.3. Broadcast Using the Topology with Replication

Addressing the problem mentioned above, the two algorithms can be combined. This algorithm is similar to the second, but from every subtree, not a single, rather multiple nodes are selected to be responsible for forwarding the message. This way, the probability of skipping a subtree is falling rapidly. Duplicate messages are possible in this case, so a unique identifier is required for all broadcasts initiated. Replication level can vary from two to *k*, the size of k-buckets.

4.4. Comparison of broadcast algorithms

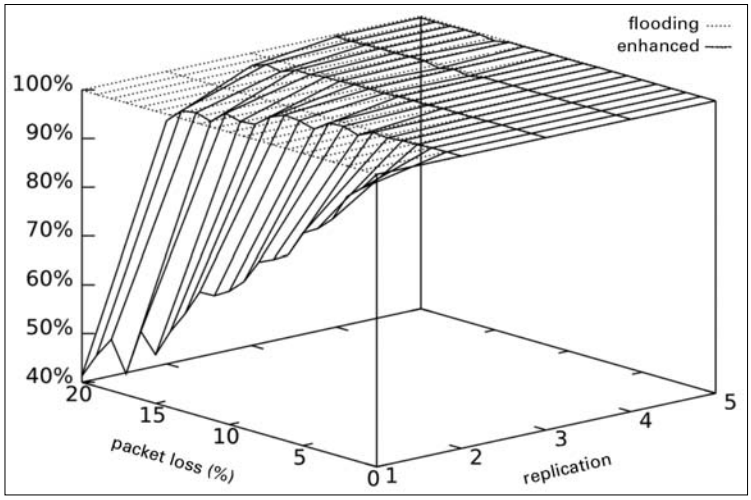
To evaluate the algorithms presented above, we developed a simple, application specific simulator. The program records the following data:

– number of all messages sent,
– number of messages per node,
– the number and ratio of nodes receiving the broadcast,
– the time required for sending the message to as many nodes as possible.

In terms of traffic costs, flooding gives the worst results. The number of messages grows rapidly with increasing the node count or sizes of k-buckets. The second algorithm using the implicit multicast tree evidently results one message for each node. For the third method, the number of messages grows rapidly for large k-buckets, but only slowly for increasing the number of nodes. For *k=5*, there were 7 messages/node for an overlay of 100 nodes, and only 9 for 1000 nodes.

To evaluate the reliability of the algorithms, we simulated an overlay of 200 nodes. Packet loss ratio varied from 0% to 20%, replication from onefold to fivefold. Flooding almost always yields perfect results; the error is smaller than line width in *Fig. 7*. This is due to the enormous number of messages. The reliability of the enhanced algorithm is of course the same as the second for *k=1*, so it is not denoted individually. In turn, using *k=2*, this algorithm produces 90% reliability even if one fifth of the packets lost; *k=3* gives 97%.

Fig. 7.
Reliability of different broadcast algorithms in Kademlia



The time required to complete the broadcast is mainly determined by the latencies of the contacts stored in the k-buckets. If we go against the recommendation of the original Kademlia paper, and instead of the oldest contacts, we select a contact with low latency for the k-buckets, the time of lookups and broadcast both decreases significantly. The latency (or the round trip time, RTT) can easily be measured using PING messages, but it can also be approximated [6]. In the case simulated by our Kadsim application, the broadcast was two and a half times faster. Of course this ratio depends on the distribution of latencies, too.

The quickest method is of course the flooding *(Fig. 8)*, as messages sent in every possible direction will obviously travel the quickest way, too. Replication will speed up the algorithm for randomly selected nodes, and for RTT selected nodes, it will not. The implicit tree broadcast algorithm is the slowest, due to its rigidity. The third algorithm is between the previous two; using replication, it can be faster than the implicit tree algorithm using RTT selected nodes. This is also caused by messages travelling more than one way at a time.

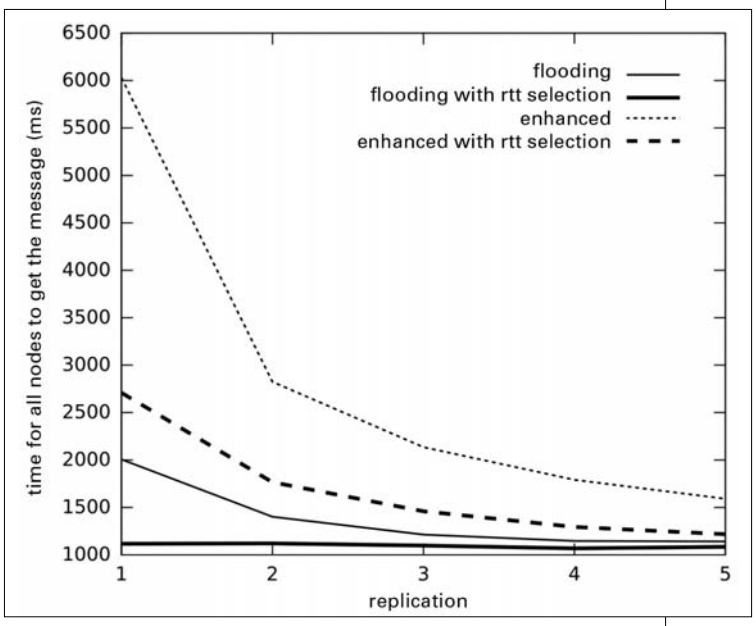


Fig. 8. The time needed for the broadcast

In the above figure, results from one hundred test runs were averaged. The lowest latency was around 15 ms, the average 0.5 s, and the highest was around 1.3 s.

5. Conclusions

The DHT-based intrusion prevention system presented in our paper is capable of creating a robust overlay of the participant software entities. Using a structured overlay network, the detection is distributed, but still it creates little processing and network traffic overhead. The reliability of the two elementary services, namely sending attack reports and broadcasting alerts can both be increased using replication. The only system-wide configuration parameter affecting the substrate, the level of this replication can be determined in advance, using the methods presented.

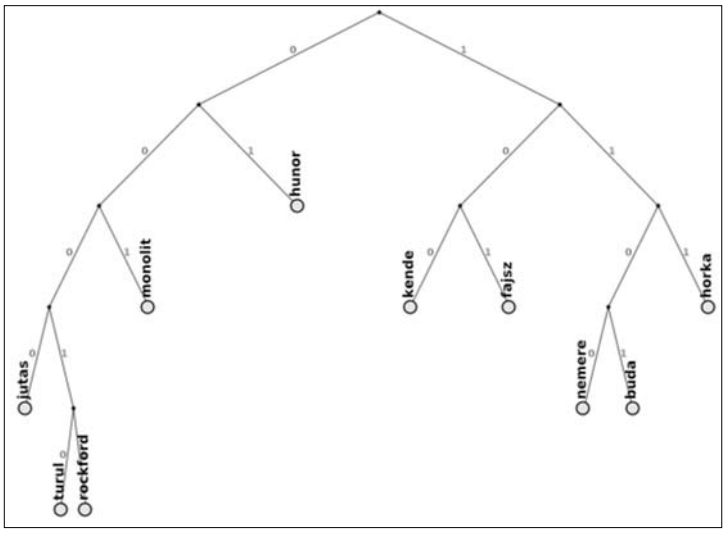


Fig. 9. A Komondor overlay in operation

Fig. 9 presents a screenshot of a smaller Komondor test overlay, with its binary tree topology. During its test runs lasting several months, it detected and prevented many intrusion attempts, and the substrate was proven to be stable. Attack reports which could be used by multiple nodes were alerts of SSH and HTTP intrusion attempts. The Snort application we used for intrusion detection logged many events which were not usable at all. For example computer viruses do not attack a single selected host for a long time. Against that kind of malware, the PROMIS system is more useful than ours [12].

The topic of our further research is the type of alerts to send on the overlay. One has to select the types of attacks, which are worth collecting and analyzing distributedly. Special attention must be paid to the case of Komondor nodes which run on different operating systems and applications: heterogeneity can increase security, as it is easier to see an attack manifesting if the system is immune. Using data collected this way may however be more difficult, as the protection must always be tailored to the host and environment in question.

Later research topics will include protection against malicious nodes building into the Komondor overlay itself. It is very easy to imagine that a compromised node sends alerts, attack reports about otherwise well-behaving clients; this way causing denial of service to the authenticated users of a system. This problem must be dealt with whatever distributed intrusion detection system one uses.

Authors

ZOLTÁN CZIRKOS is a PhD student at the Technical University of Budapest. His main fields of interest are operating system security and peer to peer communications. In 2005, he won the second award at the Conference of Scientific Circle of Students, with his paper "Development of P2P Based Security Software". He published several technical papers and wrote chapters as co-author in the field of the collaborative security.

GÁBOR HOSSZÚ received the M.Sc. degree from Technical University of Budapest in electrical engineering and the Academic degree of Technical Sciences (Ph.D.) in 1992. After graduation he received a three-year grant of the Hungarian Academy of Sciences. Currently he is a full-time associate professor at the Budapest University of Technology and Economics. He published several technical papers, chapters and books. In 2001 he received the three-year Bolyai János Research Grant of the Hungarian Academy of Sciences. He lead a number of research projects. His main interests are internet-based media communications, multicasting, P2P communications, network intrusion detection systems, character encoding and VHDL-based system design.

References

[1] Czirkos Z.,
Developing a P2P Based Intrusion Detection System,
In Proc. of the Conf. of Scientific Circle of Students,
Budapest, 11-11-2005,
2nd Award (in Hungarian).

[2] Gnutella homepage,
http://www.gnutella.org/

[3] Snort – the de facto standard
for intrusion detection/prevention,
http://www.snort.org/
(Retrieved 26-11-2008)

[4] OSSEC – Open Source Host-based Intrusion
Detection System,
http://www.ossec.net/
(Retrieved 26-11-2008)

[5] P. Maymounkov and D. Mazieres,
Kademlia: A Peer-to-peer Information System Based
on the XOR Metric.
In Proc. of IPTPS02, Cambridge, USA, March 2002.
http://www.cs.rice.edu/Conferences/IPTPS02/

[6] F. Dabek, R. Cox, F. Kaashoek and R. Morris,
Vivaldi: A Decentralized Network Coordinate System.
In Proc. of the ACM SIGCOMM'04 Conference,
Portland, OR, August 2004.

[7] Z. Czirkos, G. Hosszú,
"On the Stability of Peer-to-Peer Networks
in Real-World Environments" – chapter in book,
Encycl. of Information Communication Technology,
2nd ed., Editors: Antonio Cartelli and Marco Palma,
Information Science Reference,
Hershey, USA, 2008. ISBN: 978-1-59904-651-8,
pp.622–630.

[8] D. Karger, E. Lehman, F. T. Leighton, M. Levine,
D. Lewin and R. Panigrahy,
Consistent hashing and random trees:
Distributed Caching Protocols for Relieving Hot Spots
on the World Wide Web.
In Proc. of the 29th Annual ACM Symposium on
Theory of Computing, May 1997.
pp.654–663.

[9] I. Stoica, R. Morris, D. Karger,
M. F. Kaashoek and H. Balakrishnan,
Chord: A Scalable Peer-to-peer Lookup Service for
Internet Applications.
Technical Report TR-819, MIT, March 2001.

[10] S. Rhea, D. Geels, T. Roscoe and J. Kubiatowicz,
Handling Churn in a DHT.
In Proc. of USENIX Technical Conf., June 2004.

[11] G. Hosszú, Z. Czirkos,
'Network-Based Intrusion Detection' chapter in book,
Encycl. of Internet Technologies and Applications,
Editors: Mário Freire and Manuela Pereira,
Information Science Reference,
Hershey, USA, 2007. ISBN: 978-1-59140-993-9,
pp.353–359.

[12] Vasileios Vlachos, Diomidis Spinellis,
A Proactive Malware Identification System based on
the Computer Hygiene Principles.
Information Management and Computer Security,
Vol. 15, No. 4, 2007.
pp.295–312.

[13] Feng Zhou, Li Zhuang, Ben Y. Zhao, Ling Huang,
Anthony D. Joseph and John Kubiatowicz,
Approximate Object Location and Spam Filtering
on Peer-to-peer Systems.
In ACM Middleware 2003.

Ad-hoc adaptive cruise control algorithm

BALÁZS MEZNY, PÉTER LABORCZI, GÉZA GORDOS

Institute for Applied Telecommunication Technologies (IKTI),
Bay Zoltán Foundation for Applied Research

{mezny, laborczi, gordos}@ikti.hu

Keywords: wireless ad hoc networks, adaptive cruise control, tempomat, algorithm, simulation

Nowadays, nearly all car manufacturers can build a cruise control system (tempomat) in their cars if the customer demands. Usually, these systems can maintain a certain speed, set by the driver of the vehicle. The tempomat system can be extended with a distance measurement sensor in some top-end luxury cars, to measure the distance of objects in front of the car (this can be another car or some kind of obstacle). By using these systems, a certain following distance can be set, and the tempomat tries to maintain it by using small amounts of acceleration or breaking according to the data provided by the sensor. These adaptive cruise control systems are expensive, and the detection range and field of view of the sensors are limited. In this paper, we present an adaptive cruise control system, which sets the speed of the vehicle according to messages distributed over an ad-hoc wireless network. The wireless communication eliminates the problems caused by bad visibility or being out of line of sight. The distributed messages contain the exact position, speed and direction of the sender vehicle. The described algorithm determines also the vehicle to be followed, when the driver turns on the tempomat.

1. Introduction

Nowadays, car manufacturers strive to get more customers with more and more built-in comfort services. These services help the driver, so the driver needs to exert less effort to drive. Occasionally, it could be hard to choose or maintain the appropriate speed. For example, a long trip on the motorway, when one has to maintain constant speed over several hours, can be exhausting. A cruise control system can assist the driver in this task. These kinds of ever improving cruise control systems are on the market for several decades from nearly every car manufacturer.

However, these cruise control systems are not able to adapt to the fluctuations of the traffic, they are only able to maintain a set speed, until the driver turns off the system. If the state of the traffic changes, for example an accident causes congestion on a motorway and cars have to decelerate then the regular cruise control systems cannot offer assistance to the driver.

Adaptive cruise controls (ACC) systems started to spread recently. The ACC systems use some kind of sensors to monitor the road segment in front of the car, and warn the driver in case of an obstacle on the road. Due to the integration of the built-in sensors and actuators, it is possible to intervene into the brake system of the car, and slow it, so the driver has time to react to the situation. The drawback of the ACC systems is that they can function properly only under nearly optimal conditions. ACC systems can not provide sufficient information for the driver in case of bad weather, or in a road curve where the front sensor is not able to monitor a long enough segment of the road.

In this paper we provide a solution to the described problems, by exploiting the possibility that wireless communication can be established among the vehicles [1], and accurate information can be provided for the speed regulation. The advantage of the wireless communication is that direct line of sight is not needed, so it remains operable in curves, furthermore, the range of the wireless communication is around 300 meters, nearly the double of the range of the radars used in ACC systems [2].

Another possible application could be in urban traffic. On an urban road it is not common that a certain velocity can be maintained for an extended period of time. Under these conditions the traditional cruise control systems can not be used, because it can not adapt to a changing environment. With the system presented in this paper, it is possible to set the vehicle speed according to the preceeding vehicle. When the preceeding vehicle decelerates then the following vehicle will slow down, and if it accelerates then the following car will adapt its speed to that as well. The driver using our system does not have to deal with the actual accelerations or decelerations; the embedded computer handles the calculations.

2. The algorithm

The cruise control algorithm chooses a target to follow from the vehicles in front of the car. The aim of the algorithm is to regulate the velocity of the vehicle, so the following distance between the target and the follower vehicle is in accordance with the current speed of the vehicles, maintaining a safe following distance. A minimal speed threshold of 20 kph is built into the algorithm. If the velocity of the vehicle is lower than this threshold then the speed regulation is turned off. This

threshold was set due to the fact that at so low speeds the safe following distance is commensurable to the error of the GPS position.

The algorithm uses the data received through radio communication and the GPS information gathered by its own GPS receiver to calculate the current speed to be set.

Stored values for position information are as follows:

Lon: longitudinal coordinate in radians up to 8 decimal places (accuracy to approximately 6 centimeters).

Lat: latitudinal coordinate in radians up to 8 decimal places (accuracy to approximately 6 centimeters).

Vel: length of the velocity vector in kilometer per hours.

Hdg: direction of the velocity vector up to 2 decimal places.

Svs: nr. of tracked space vehicles (GPS satellites) by the GPS receiver.

Tof: timestamp of the GPS information.

The following information is also included in a message used by the algorithm:

Original sender ID:
The identification number of the originating unit of the message.

Sender ID:
The identification number of the unit, which retransmitted the message most recently.

TTL (Time To Live):
This sets how many times a message can be retransmitted. Our cruise control algorithm currently sets this value to 1, so the messages are propagated to a distance of one hop.

The algorithm calculates the distance between the vehicles by using the position information contained in the messages. A Kalman filter [3] was used to refine this distance calculation.

The *sender ID* and the *original sender ID* has to be checked in the received packet, if either one is equal to the ID of the *own ID*, the message is dropped.

The next step is to compare the heading contained in the packet to the vehicles own heading. If the difference is lower than a set threshold, the algorithm figures the two cars are heading the same way. This threshold is variable, currently set to 20 degrees.

If the headings match, it has to be decided whether the sender of the message is ahead of the own vehicle. This is a similar calculation like where the headings were compared. The own position vector is subtracted from the position vector in the message, and the difference vector is compared to the heading vector. If the difference between these two vectors is less than 90 degrees, the sender of the message is in front of us.

If these conditions are not met, then the sender of the packet is either behind us or is headed to another direction, and it cannot be followed. In this case the algorithm drops the packet.

If the sender has the same heading as our vehicle and is in front of it, it can be followed. In this case the state machine described in the following section processes the message further.

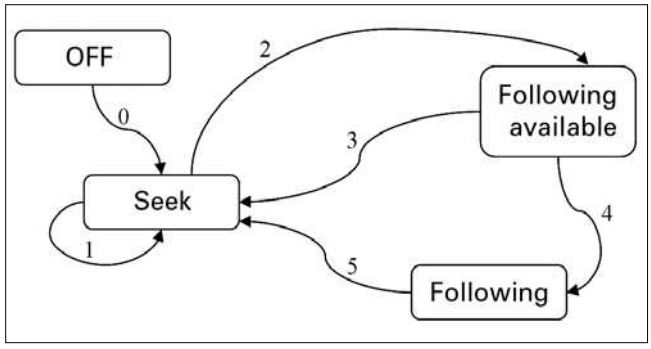
2.1. The state machine

The adaptive cruise control algorithm makes decisions according to the following stored variables:

- ID of the target (followed) vehicle.
- Distance to the target vehicle.
- Distance to the target vehicle in the moment when speed synchronization begins.
- Velocity of the target vehicle.
- Velocity of the target vehicle the moment when speed synchronization begins.
- Timestamp of the most recent message received from the target.

The algorithm works according to a state machine as shown in *Fig. 1*.

Fig. 1. State machine



2.1.1. Seek phase

When turned on, the algorithm is in this phase and waits for messages from other vehicles. It is decided here whether the actual following is possible, if the constraints described in the previous section are satisfied.

First the distance to the sender of the message is calculated by the Haversine formula [4], which returns the shortest distance between two points on a sphere. This calculation is done according to the GPS positions contained in the message and in the own receiver.

If the algorithm already has a target, it has to be decided whether the target sent the message or another car which is closer than the target.

If the message was sent by the target then the timestamp of the message and the distance is stored, and a counter is increased by one. This counter shows how many messages have been received from the same target.

If the message was not from the current target, but the calculated distance is shorter than the target's distance, then the originator of the message becomes the new target. The distance and the timestamp is stored in this case as well, and the counter is set to zero (transition 1 on Fig. 1).

If the algorithm didn't have a target, the calculated distance, the ID of the sender of the message and the

timestamp of the message are stored. A counter is started from zero. This shows the consecutive messages received from the same target.

There are some thresholds in the algorithm, and exceeding these thresholds resets the state machine to the seek phase. This happens if the target vehicle's speed is lower than 20 km per hour or if no message is received from the target for five seconds. The counter is also set to zero. The stored target is unchanged, because it is still the closest vehicle in front of the own vehicle.

The following available phase is reached, if three consecutive messages have been received from the same target, while it remains the closest vehicle in front (transition 2 on Fig. 1). This threshold of three messages is a needed delay to ensure, that the chosen target is in a stable enough position to be followed. For example on a motorway when the following is engaged and someone overtakes both our vehicle and our target. When the overtaking vehicle is between us, the algorithm would decide, that it is the new target, because it is closer than the current target. At this time the algorithm would start to increase our speed to match the new target's speed. This would cause our vehicle to crash into the one in front of it, which was the real target vehicle.

2.1.2. Following available phase

In this phase there is a vehicle in front of our vehicle in a stable position and it's possible to follow it. The distance between the two vehicles is refreshed by every message received by the algorithm.

Transition 3 on Fig. 1 happens, if the target is changed. This could happen for example when someone overtakes us, or the current target overtakes another vehicle. The same thresholds are applied as in the seek phase. If the speed of the target is lower than 20 km per hour or 5 seconds passed without receiving a message from the target resets the state machine. This time the driver is able to reset the algorithm by pressing the corresponding button on the control panel.

2.1.3. Following phase

The algorithm enters this phase if the driver presses the engage button on the control panel (transition 4 on Fig. 1).

The algorithm resets in this phase as well, if the previously mentioned conditions are met, or when the driver turns off the following (transition 5 on Fig. 1).

Upon entering this phase the speed and distance of the target is stored. These values will be used in the calculation of the actual desired distance:

$$d_d = \frac{v}{v_0} \cdot (d_0 - l) + l, \qquad (1)$$

with the following notations:

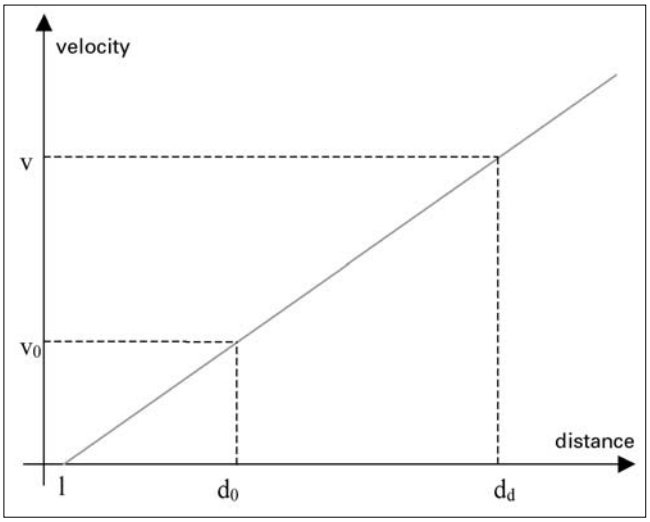
d_d: desired distance
v: current velocity of the target
v_0: velocity of the target when the following was engaged
d_0: distance of the target when the following was engaged
l: changeable parameter, the minimal distance to be kept between the GPS receivers

The distance values are the result of the Kalman filter, because the unfiltered distance values vary greatly since the position measurements do not happen in the same time. The GPS receiver used in our hardware has an update frequency of 1 Hertz, so under certain circumstances, for example at a speed of 50 km per hour, the difference between the GPS position and the actual position can be 13.8 meters.

The extension of the distance by the parameter l is needed, because the distance calculation gets the distance between the GPS receivers, not between the front and rear bumpers of the vehicles. This parameter can be set for various types of vehicles. It is currently set to 4 meters for passenger cars, but it could be set to for example 10 meters for trucks.

When the following starts, a line is defined according to the velocity and distance values at that time. When the speed of the target vehicle changes, the corresponding following distance can be calculated along this line *(Fig. 2)*.

Fig. 2. Following distance calculations



The following speed has to be set to reach the desired distance:

$$v_d = \frac{d}{d_d} \cdot v, \qquad (2)$$

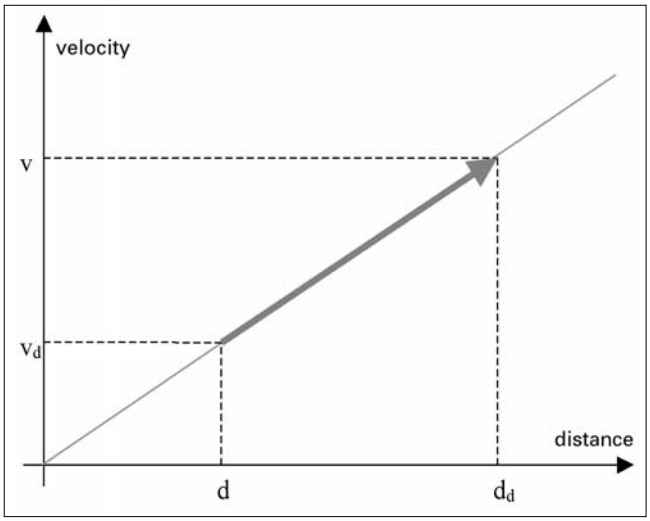
with following notations:

v_d: desired velocity to reach the desired distance,
d: current distance to the target,
d_d: desired distance,
v: current velocity of the target.

A hysteresis was built into the algorithm in order to avoid continuous intervention to the engine or brake systems. If the difference of the current distance and the desired distance is below 5 percent of the desired distance (d_d), the desired velocity (v_d) is set to the speed of the target.

The transition on the velocity-distance graph can be seen on *Fig. 3.* If the speed of the target changes, the corresponding following distance can be acquired according to (1). Using these values knowing the current speed, the velocity can be calculated to converge to the desired distance.

Fig. 3. Speed regulation



For example, as shown in Fig. 3, the target maintains a stable speed, and the following distance is set to a certain value (d) calculated by the algorithm. Both of the cars are traveling at the same velocity (v_d). If the speed of the target vehicle changes to v, a new desired distance (d_d) has to be reached. The distance between the vehicles will increase, because the target is moving faster than the other car. The algorithm will increase the velocity of the following car, as the distance converges to the desired distance, according to (2).

The acceleration has to be computed to adjust the engine control or the brake system properly. The calculation of the desired acceleration is as follows:

$$a_d = \frac{v_d - v_s}{T}, \tag{3}$$

with the following notations:
a_d: desired acceleration,
v_d: desired velocity,
v_s: current velocity of the own car,
T: free parameter to control the reaction time of the algorithm.

The maximal acceleration is 5 m/s^2 and the maximal deceleration is 9 m/s^2. These are the acceleration values of a typical passenger car.

To limit the number of repetitions, the acceleration is only computed if either of the following conditions is met:

- The distance of the target differs from the desired distance by more than 1 percent.
- The velocity of the target differs from the desired velocity by more than 5 percent.
- The difference between the speed of the target and the desired speed is greater than the difference between the desired speed and the own speed.

As a conclusion, the aim of the speed regulation is to reach the appropriate following distance. The engine is controlled according to the difference between the current speed and the desired speed (3).

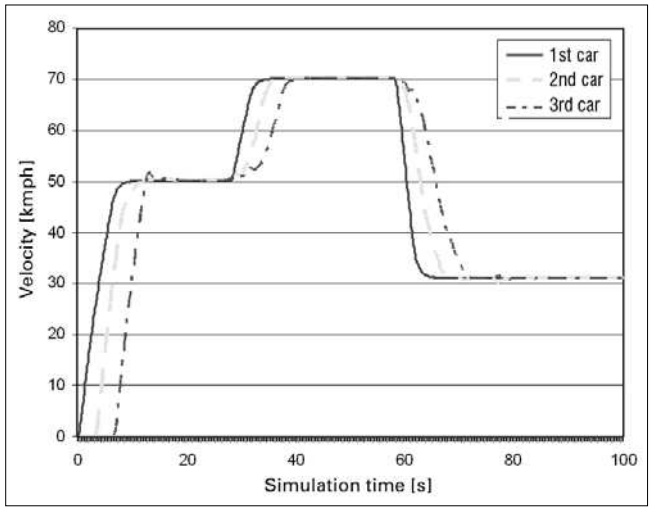
3. Simulation results

The testing of the algorithm was done on a simulator, described in [5] and [7]. At a later stage it was tested in real environment, embedded in vehicles.

Three cars were examined following each other in the simulator. The velocity of the first car was set to certain values. It started at 50 km per hour, at the third of the simulation time it was increased to 70 km per hour, and at two thirds of the simulation time it was decreased to 30 km per hour. The velocities of the following two cars were regulated by the algorithm. The messages containing the position information were broadcasted every tenth of a second to allow precise adjustments.

In the simulations a following distance of two seconds was set between the vehicles, because this is the minimal safe following distance. The results of this simulation can be seen in *Fig. 4*.

Fig. 4. Velocities of the vehicles



The vehicles accelerate to 50 km per hour at the start of the simulation, where the ACC algorithm is engaged.

It can be seen that the second and third cars reach their target's velocity with a little oscillation. It is also noticeable, that the acceleration of the following cars differs from their target's acceleration. This is due to the mechanism of the algorithm as it adapts the following distance to the changed velocity. In case of acceleration the following car drops behind a bit, and in case of deceleration it catches up on its target.

The distance between the cars was registered in the course of the simulation. The desired velocity calculated by the algorithm is based on this value. The various distance values between the first and second car can be seen on *Fig. 5*.

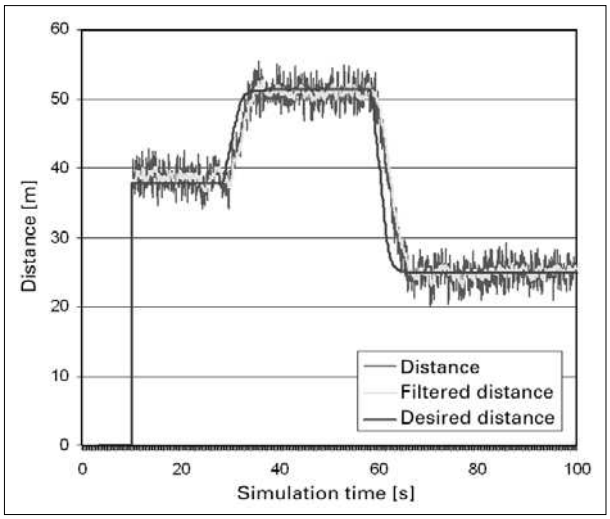


Fig. 5.
Distance values between the 1st and 2nd vehicles

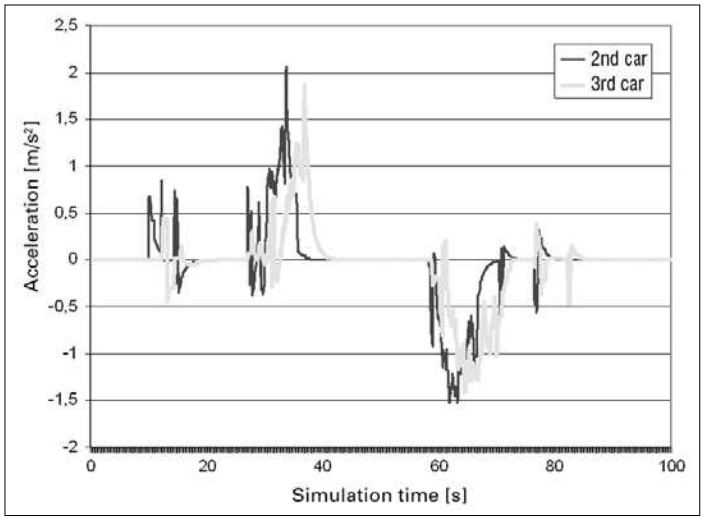


Fig. 6. Acceleration values

The algorithm was turned on at the 10th second of the simulation; the registration of the distance values was started at that point. The unprocessed distance value is shown by the narrow dark line. Though the position information is updated at a frequency of 10 Hz, this distance value is noisy due to the time difference of the position measurements and GPS error. Kalman filtering was used to eliminate most of the noise. The filtered distance values are shown by the bright line. The dark line shows the desired distance, which is calculated by the algorithm according to the actual speed of the target and the values stored at the start of the following phase. The algorithm tries to minimize the difference between the bright and the dark line by adjusting the velocity of the following vehicle. It can be seen, that after the transient caused by the acceleration or deceleration of the target, the distance between the cars settles at the desired distance.

The second and the third vehicle's acceleration values can be seen on *Fig. 6*.

The acceleration and deceleration periods can be examined on the graph, and it can be seen, that the threshold was never exceeded, where the algorithm would limit the acceleration value. The speed regulation algorithm would not cause uncomfortable acceleration or deceleration, as the maximal acceleration was around 2 m/s^2 and the maximal deceleration was 1.5 m/s^2. The noticeable acceleration spikes would be limited by the inertia of a real vehicle.

4. Field tests

The algorithm was tested under realistic circumstances. It was downloaded into two control units described in [8]. The control units were integrated into a passenger car and a truck, and the control unit in the truck was connected to the CAN bus, so the algorithm could operate the brakes and intervene to the engine control.

It can be noticed, that the speed regulation is not as precise as in the simulations for two reasons. The first is that the Kalman filtering was not implemented at the time of the test, the second is that the used GPS receivers had an update frequency of 1 Hertz, which is the tenth of the assumed frequency in the simulations.

The data of one of the test runs can be seen in *Fig. 7.*

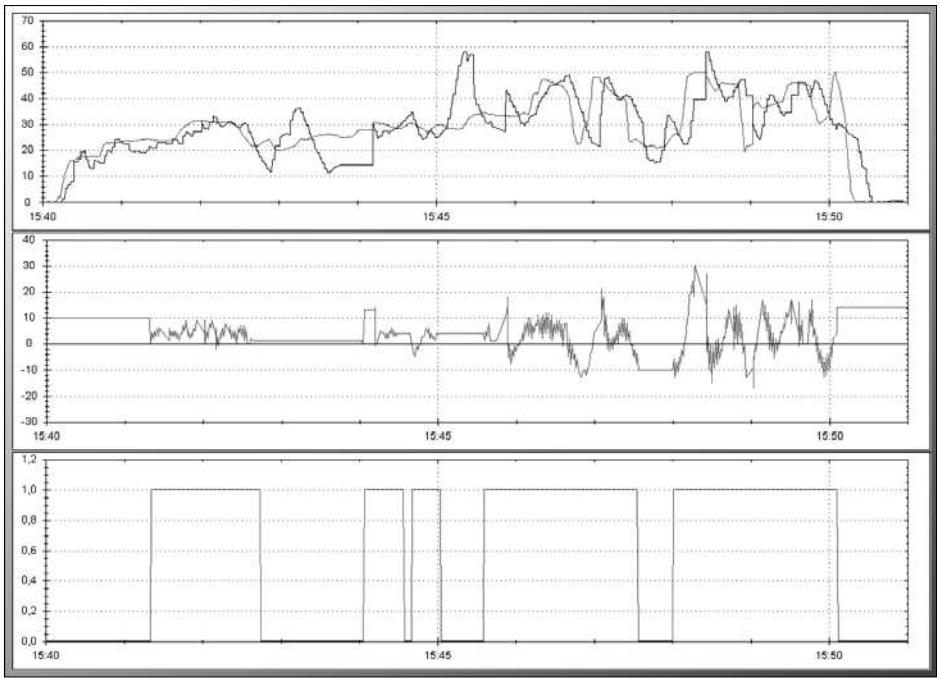


Fig. 7.
Field test results

The time on the horizontal axis is in minute units. On the top graph the velocities of the two cars can be examined. The darker line shows the following truck, the brighter line shows its target. The center graph shows the control value handed on to the CAN bus controller by the algorithm. This value is proportional to the acceleration, the exact coefficients for acceleration and deceleration had to be adjusted to the vehicle.

The bottom graph shows the status of the state machine. This value was 1 when the algorithm was in following phase, and the value was 0 otherwise. When the algorithm was engaged the truck adjusted its speed to the speed of its target with a little delay. This delay was caused by the infrequent position update messages.

The noise in the control value is due to the error of the distance calculation, as described in the previous section. This noise could be eliminated by using the same Kalman filtering as in the simulations.

This measurement noise and the one second delay between position update messages caused the speed regulation to not be as precise as in the simulations.

5. Conclusions

We presented an algorithm that is able (1) to match the velocity of the vehicle to a followed vehicle by utilizing wireless communication, (2) to adapt to the changes in the velocity of the followed vehicle while (3) maintaining a safe following distance.

The proposed cruise control system can be used if the vehicles have the necessary devices for communication, position determination and calculations. Therefore, it could be introduced in truck convoys of transportation companies where only the driver of the first vehicle needs to pay increased attention; the following drivers could let the algorithm to follow the preceding vehicle.

Authors

BALÁZS MEZNY received his M. Sc. Degree in 2008 from the Budapest University of Technology and Economics. He started his Ph. D. studies in 2008 with a scholarship from the Bay Zoltán Foundation for Applied Research, Institute for Applied Telecommunication Technologies (IKTI). He is working at IKTI with a scholarship since 2007. His research interests include intelligent transportation systems and optimization.

DR. PÉTER LABORCZI is currently employed as a senior research fellow at IKTI, the Institute for Applied Telecommunication Technologies. He received his M.Sc. degree in 1999 and his Ph.D. degree in 2002 both in Computer Science from the Budapest University of Technology and Economics. In 2002 he was invited to Arsenal Research, Vienna, Business Unit Transport Telematics, where he was a Marie Curie Fellow until 2004 in the framework of an EU research project. Dr. Laborczi is the co-author of more than 30 papers in refereed journals and conference proceedings. His research interests include intelligent transportation and infocommunication systems, graph theory, algorithms; network design, optimization and analysis.

DR. GÉZA GORDOS received his M.Sc, Ph.D and Dr. Habil. in Telecommunications from the Budapest University of Technology and Economics (BME) in 1960, 1966 and 1994, resp. He holds D.Sc. from the Hungarian Academy of Sciences. He is full professor with the BME serving as head of two sections in the Department of Telecommunications and Media Informatics since 1976. His previous employments include 3 years at various universities abroad (UK, USA) and 7 years in industry, among others as Chairman of the Board of the Hungarian Telecommunications Co. (1992-93). His research activities have been focusing on the technology and management of telecommunication systems and services as well as on speech processing. In 2004 he accepted the invitation from the Hungarian equivalent of NSF to establish and lead the Institute for Applied Telecommunication Technologies (IKTI).

References

[1] Raymond Freymann,
"Connectivity and Safety",
5th European ITS Congress, Hannover, Germany,
June 2005.

[2] A. Török, P. Laborczi, G. Gerháth,
"Constrained Dissemination of
Traffic Information in Vehicular Ad Hoc Networks"
accepted for Presentation at the IEEE 68th
Vehicular Technology Conference (VTC2008-Fall),
Calgary, Canada, 21-24 September 2008.

[3] G. Welch, G. Bishop,
"An Introduction to the Kalman Filter",
Technical Report, UMI Order Number: TR95-041.
University of North Carolina at Chapel Hill, 1995.

[4] W. Gellert, S. Gottwald, M. Hellwich,
H. Kästner and H. Küstner,
"The VNR Concise Encyclopedia of Mathematics",
2nd edition, Chapter 12, Van Nostrand Reinhold,
New York, NY, 1989.

[5] Gordos Géza, Gerháth Gábor, Kardos Sándor,
Laborczi Péter, Mezny Balázs, Vajda Lóránt,
"Improving Intelligent Transportation System's
performance with the help of MANETs",
Híradástechnika, Vol. LXI., No.12, 2006,
pp.29–34. (in Hungarian).

[6] P. Laborczi, A. Török, L. Vajda,
S. Kardos, G. Gordos,
"Vehicle-to-Vehicle Traffic Information System
with Cooperative Route Guidance",
in Proc. of the 13th World Congress on Intelligent
Transport Systems, London, UK, 8-12 October 2006.
CD-ROM, Paper no. 2237.

[7] Csák Bence,
"Ambient intelligence on public roads",
Híradástechnika, Vol. LXI., No.12, 2006,
pp.35–39. (in Hungarian).

Reliable gossiping in inter-vehicle communication

MIKLÓS MÁTÉ, ROLLAND VIDA

Budapest University of Technology and Economics,
Department of Telecommunications and Media Informatics, High-Speed Networks Laboratory
{mate, vida}@tmit.bme.hu

Keywords: inter-vehicle communication, gossiping, urban environment, reliability

Due to the increasing traffic density in urban areas, a computer-aided robust collision avoidance and traffic control system should be established, based on decentralized inter-vehicle communication. Vehicles group themselves into a special ad hoc network with high mobility and low link reliability. Traditional ad hoc routing solutions cannot cope with these conditions, while flooding based approaches consume too many resources. Our proposed scheme, Localized Urban Dissemination (LUD), is a location aided gossiping protocol, which concentrates the information spreading to areas where it is most likely to be useful. The reliability of simple gossiping, however, is not enough for emergency message dissemination, but our simulations prove that a simple modification in the packet forwarding scheme can overcome this limitation.

1. Introduction

The primary goal of Intelligent Transportation Systems (ITS) is to increase road safety by detecting emergency situations in advance and notifying the drivers about the traffic events. Such systems can be efficient only if the vehicles communicate with each other and share the measurements of their sensors to take coordinated actions. Inter-vehicle communication might be realized either in an infrastructure-based manner, in a pure ad hoc fashion, or as a mixture of these methods. In this paper we examine how emergency messages might be disseminated in a Vehicular Ad Hoc Network (VANET).

A VANET is a special kind of ad hoc network, as the vehicles have much higher average speed than the nodes of a sensor network, but their mobility pattern is restrained by the road network. In a VANET messages related to road safety and cooperative traffic jam avoidance should be distributed by a flooding-based protocol, as they are not addressed to a single destination, but to all cars that are interested in receiving them. These are typically the ones that need to change their speed or path in order to decrease the jam or to avoid the danger. It is an important design goal to determine where such vehicles can be found, because the broadcast in the ad hoc network is expensive, so the flood should be localized to the area of interest.

The vehicles that must be informed are in a certain vicinity of the source of the message; all the vehicles that receive the warning will take counteractions, and after a certain distance the message becomes irrelevant. We can safely assume that all vehicles are equipped with GPS receivers; therefore, a spatial flood limitation is a viable solution.

In the followings we present our Localized Urban Dissemination (LUD) protocol, which limits the message flood into areas where vehicles are interested in the message with high probability [1]. Unlike most of the similar protocols, the target area in LUD is not determined by the source, but certain forwarding nodes decide if the message is worth forwarding in a certain direction or not. This distributed decision scheme makes our solution radically different from the restricted flooding protocols based on a predefined hop count.

The rest of the article is organized as follows. Section 2 gives an introduction of the gossiping scheme, and Section 3 explains how it can be used to disseminate emergency messages in urban environments. Section 4 gives a detailed description of the operation of the LUD protocol and its properties, and the way its reliability can be increased. Finally, Section 5 summarizes the results, draws the conclusions and shows our future plans.

2. Gossiping

Historically, gossiping was first introduced in distributed databases to reduce the cost of synchronization [2]. It works as follows. All nodes know the list of the nodes in the system, and in each timeframe they randomly choose a subset to synchronize with. This reduces the number of messages exchanged in a timeframe, and as a consequence the convergence time may also decrease due to the significantly shorter periods and the marginally slower information propagation if the peers are selected carefully [3].

Gossiping in a wireless multi-hop ad hoc network (MANET) requires special peer selection strategies, as these networks exhibit properties that are different from the ones based on a fixed infrastructure. In the latter it is usually safe to assume that the cost to reach every peer is the same, and it is easy to set up a point-to-point connection between any pair of nodes (for example a TCP connection). In a MANET, however, the cost of reaching a peer dramatically increases with distance, and the wireless medium is inherently broadcast based. More-

over, handling link unreliability and energy constraints are important only in a MANET. One of the possible ways of gossiping in MANETs is to flood the messages and drop them randomly with a predefined $p_{drop}>0$ probability. This random peer selection scheme favors the nodes that are close to the source and also utilizes the broadcast nature of the wireless medium.

In inter-vehicle communication the nodes are usually placed along a line (the road), and we observed that in this topology the gossiping scheme effectively limits the distance a message can reach. Namely, if the rebroadcast probability is p, then the expected value of the hop count is $1/(1-p)$, which is not infinite if $p<1$. This is exactly what is expected to be needed in emergency message propagation. The question is: how to set p to get the optimal target area?

3. Characteristics of an Urban Environment

In an urban environment the road topology is not just a lonely road, but a dense network of streets and junctions. Yet, this environment is similar to a lonely highway in the sense that the message flood follows the roads, as the buildings block the propagation of radio signals. The big difference is that from point A to point B there can be several different paths; thus, emergency messages do not need to reach a certain point, like the highway exit, because vehicles a few blocks away can already change their route in case of an accident.

It is true that there are multiple paths between the two points, but of course not all of them are taken with the same probability by the vehicles. This is due to the fact that vehicles arriving at a junction prefer certain outgoing directions over others. Some roads are one-way, some lead to important places, and turning left is usually forbidden in large intersections. When disseminating emergency messages these conditions must be taken into account, because the radio resources are scarce, and broadcasting has huge overhead [4]. To make a message dissemination protocol efficient, the traffic conditions must be considered when defining the coverage area.

4. Localized Urban Dissemination

Our proposed emergency message dissemination protocol, called Localized Urban Dissemination (LUD), is a gossiping-based emergency message dissemination protocol [1].

Gossiping makes it very easy to make a certain road segment included in, or excluded from the coverage area. The p rebroadcast probability should be changed in the junctions depending on the traffic conditions. Thus, the vehicles need to be equipped with a digital map, and set their role to *Decider* or *Forwarder*, according to their current position. The ones arriving at a junction become Deciders, and must recalculate the rebroadcast probability of the packets they forward. The nodes that are not in a junction are Forwarders; they simply rebroadcast the messages with the probability written in the packet header.

4.1. The Decision Scheme

The heart of any gossiping-based protocol would be the decision scheme that sets the rebroadcast probabilities of the packets. The decision scheme of LUD sets the p rebroadcast probability so that the probability of a message forwarded on a given path reaching a certain point P becomes equal to the probability of vehicles from that point going to the source of the message on the same path. If we call the first event A and the other one B, then the formula will look as follows:

$$\mathbf{P}(A) = C\mathbf{P}(B), \tag{1}$$

where a C scaling factor is inserted to let the source scale the size of the coverage area. We will see that the resulting scheme is memoryless, and this scaling factor disappears after the first junction.

Forwarding the message along a path is a geometric process, because it is a series of independent Bernoulli trials. Its parameter is the probability of the success on the elementary trials, which is the p rebroadcast probability. The probability of the before mentioned event A is

$$\mathbf{P}(A) = \sum_i p_i^{h_i}, \tag{2}$$

because the message reaches a certain point only if all nodes on the path have chosen to forward it. The rebroadcast probability may be different for each road segment, and the different road segments are h_i hops long. A hop can be longer than the inter-car distance if there are multiple cars in the radio range of the transmitter node. The LUD protocol requires a Medium Access Control (MAC) protocol to be used that can select the farthest receiver in the direction of the flooding to rebroadcast the message. Such protocols are CBF [5] and CFB [6] for example.

The probability of vehicles going to the source of the message can be described with two parameter sets. The first is a Q_i matrix of steering probabilities for each junction; an element $q^i_{j,k}$ for junction i represents the probability that cars coming from the neighboring junction j go to neighboring junction k. The second dataset consists of s_i stop probabilities for each road segment to model finite journeys. A car reaches the source of the message along the path of the message only if it chooses the appropriate roads and it does not stop in between.

Turning this into an equation we get:

$$\mathbf{P}(B) = \sum_i q^i_{j,k}(1-s_i), \tag{3}$$

if we assign the identifiers to the junctions on the path as shown in *Fig 1*.

The Decider being in junction D decides how likely it is that vehicles coming from junction D+1 are interested

in the message, because the Decider itself came from junction D+1. The equation it must solve is

$$\sum_{i=0}^{D} p_i^{h_i} = C \sum_{i=D}^{1} q_{j,k}^{i} (1 - s_i), \quad (4)$$

where the quantity in question is q_D .The reversed indexing on the right side refers to the order the vehicles going to the source encounter the junctions.

Equation (4) has a very interesting property, because it describes a geometric process. It is known, that

$\mathbf{P}(X > x+y | X > x) = \mathbf{P}(X > y)$

if X follows a geometric distribution. In our case the Decider calculates the probability of the message reaching the next junction *if* it reached the current one. This conditional probability causes the dissemination process to forget the past, and all that remains of equation (4) is

$$p_D^{h_D} = q_{D+1,D-1}^{D} (1 - s_D). \quad (5)$$

To calculate p_D, the Decider must know the q_D turning probability for the junction and the s_D stop probability on that road segment. These might be derived from the ranks of the roads and maybe some other data the digital map can provide (e.g., stopping on a main road is highly improbable, and so is turning into a side street). Turning lanes and one-way roads are also indicated by the map. This method, however, is far from being perfect, because there are lots of things that influence the paths of the vehicles, and most of them are not present on the maps or their effect is not easy to determine.

The precision is very important when setting p, as the error caused by the insufficient knowledge of the traffic conditions can be severe. The expected hop count is $1/1–p$, which means small changes in p might trigger great leaps in the size of the coverage area.

If a traffic monitoring system collects the necessary data, a Traffic Conditions DataBase (TCDB) can be built that contains the q and s values that describe the usual traffic conditions. We expect that the navigation system that uses our dissemination protocol can eliminate most of the traffic jams, hence the usual conditions will change slowly, and the causes of the sudden deviations from the usual conditions are handled efficiently. The TCDB should be downloaded and regularly updated by the navigation device of all vehicles. The updates must not be sent on the same channel as the emergency messages, but some other means of wireless Internet access, like Wi-Fi hotspots in the parking lots or near the traffic lights. A vehicle with an outdated TCDB should not take on the role of a Decider.

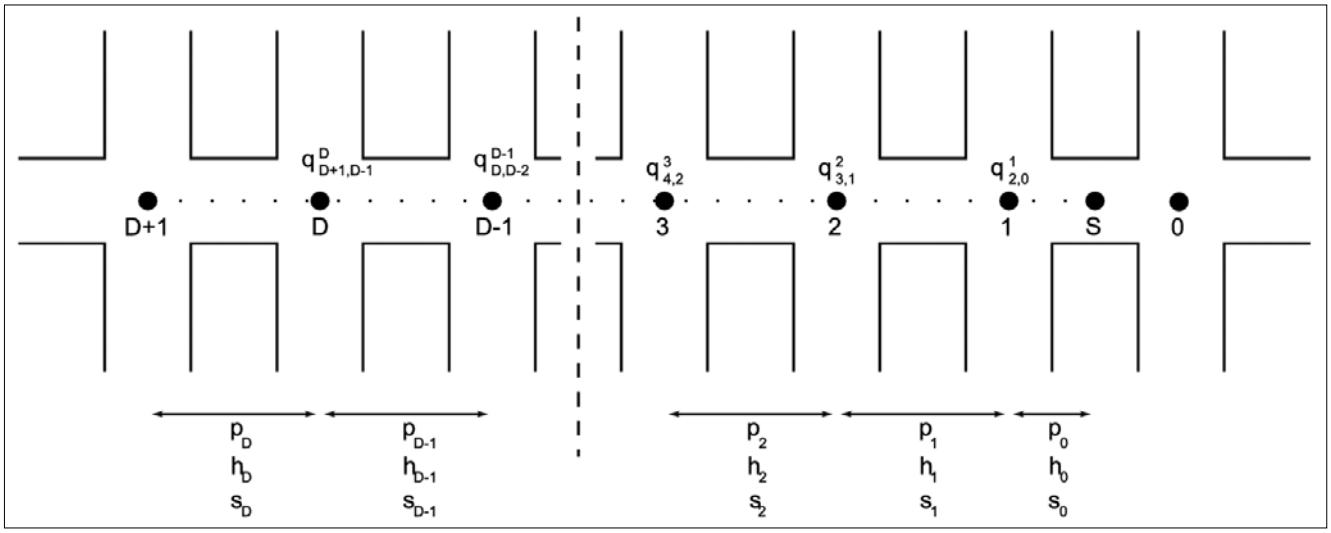
Equation (5) also contains h_D, the length of the next road segment the message might be forwarded on. The length in meters can be read from the digital map, and the Decider came to the junction from that road, so it should have some signal strength and next hop distance measurements. The LUD protocol, as mentioned earlier, needs a MAC protocol that selects the farthest node in the given direction, which implies that the MAC layer is capable of providing the necessary data.

4.2. Forwarding

The decision scheme and the gathering of its input data are the most important parts of the LUD protocol, but there are also some less obvious, but very interesting details of the protocol. The characteristics of the radio channel and the properties of the urban environment provide some challenges, but they also offer opportunities to improve the efficiency of the emergency message dissemination. The five fields the routing header of the packets should contain are: a unique identifier of the notification, the rebroadcast probability, the location of the source, the target junction (the one the message is heading to), and the previous junction (where its p was last set).

Every multi-hop routing algorithm needs to avoid routing loops. Even if the lifetime of the messages is limited, the bandwidth waste of delivering a message to nodes that already received it is not acceptable. In the case of LUD the problem is severe, as it is expected that the emergency messages are generated in large quantities. By exploiting the information sources of the protocol, the

Fig. 1. Parameters for the decisions along a path



routing loops can be avoided with a simple, stateless test: the p rebroadcast probability must be set to zero if the Decider sees that the next junction of the message (where the Decider came from) is closer to the source than the current one. The position of the source must be included in the packet header for this to work, but the message has to contain it anyway, because it carries a notification about a local change in the traffic conditions with the location of the event.

The messages are broadcasted along all possible paths. Multiple instances of them are spawned in a junction, because the vehicles arriving from all directions become Deciders and they all set the rebroadcast probabilities for the road segment they came from. Locking the dissemination of a message to the road segment the two junctions define (the position of the last decision and the supposed next junction) is beneficial for the multi-path dissemination. The coverage area becomes entirely defined by the decisions, and there is less overhead if the messages cannot "wander" around.

The source can choose in which direction the message should start to spread if the Deciders let messages into new road segments only if their target junction is the current one. The most common scenario in IVC is the backward propagation: the vehicle that detects something notifies the other vehicles that follow it. In LUD the source sets the target junction of the message to the one behind it, and the previous junction to the one ahead of it. Using the notations of Fig. 1, the message reaches junction 0, but the Deciders drop it due to the target mismatch.

The packets are distinguished by the unique identifier of the notification they carry, and the identifier of the two junctions that define their actual road segment. When two instances of the same notification go along a road segment they are indistinguishable, and they fuse into a single instance. The coverage area becomes smaller due to this, and Equation (4) is also not entirely true anymore, because some of the possible propagation paths are cancelled. However, the gain in the number of duplicated messages being eliminated makes it worth it.

4.3. Reliability of the dissemination

Emergency message propagation, like any other system, has its own reliability criteria. These criteria can be organized into two categories: the ones the MAC protocol is responsible for, and the ones the dissemination protocol must meet.

The calculation of the size and shape of the coverage area, as we presented in the previous section, considered the only reason of packet drops being the coin flip trial of the Forwarders. The wireless medium is, however, highly unreliable and spontaneous packet drops caused by interference, noise and fading are inevitable. There are numerous MAC protocol proposals for VANETs that ensure a particular level of reliability (e.g., [7]), and it should also be possible to combine them with one of the directed broadcast MAC protocols mentioned earlier. The tradeoff between reliability and propagation speed is the most important design aspect of the MAC protocol, as emergency messages need to be propagated fast, and no vehicles should remain uninformed, but this is out of the scope of this paper.

A major advantage of the memoryless nature of the decision scheme lies in its simplicity, but unfortunately it also has a serious drawback. The standard deviation (the square root of the variance) of the geometric distribution is $\sqrt{p}/(1-p)$, which almost equals its mean $(1/(1-p))$. This means that the dissemination is highly unreliable, because the message can be dropped at any time, and reaching zero or very few hops has a too high probabili-

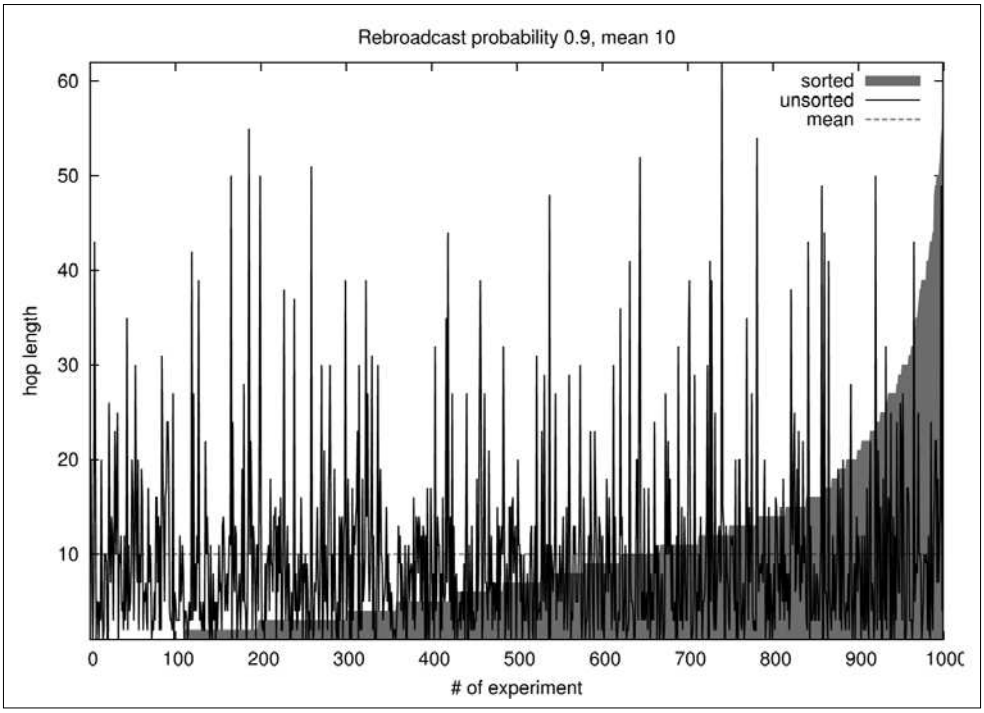


Fig. 2. Hop lengths in realizations of the geometric process

ty. The high variance of the geometric process is caused by the terminal verdict of the elementary trials.

According to simulations for a single road, shown in *Fig. 2*, there are high peaks in the hop lengths, and after sorting the results numerically the dominance of the ones that are smaller than the mean becomes clearly visible: the stairs are broad for small values, and the peak at the end is much higher than the mean. The difference between the resulting coverage area and the theoretical one should be minimized in order to improve the reliability of the dissemination.

Reducing the variance of a distribution is best done with averaging. There are two possible ways to produce an averaged dissemination area. One is to send more notifications of the same event – this is easy, as events are usually detected by more than one vehicle, and the only thing to do is not to suppress additional notifications. This solution, however, increases the number of messages flowing in the network, wasting the precious resources.

A better way to decrease the variance is to modify the elementary coin flipping trial of the Forwarders to make the resulting distribution the average of *k* independent runs. The modified trial is a voting game: the message only gets dropped if *k* nodes voted for dropping. This scheme results in a smoother distribution, as the verdict of an elementary trial is not a terminal one. However, it is not memoryless anymore, because the counter of the dropping votes must be included in the packet header. To restore the expected value of the hop count, the rebroadcast probability must be decreased to:

$$p' = 1 + k(p-1)\,. \tag{6}$$

Fig. 3 shows the simulation results for the band of standard deviation around the mean for the geometric process and two averaged processes. The effect of the averaging, i.e., the decreased standard deviation means an increase in the reliability of the dissemination. It is also visible that using this modified scheme messages go at least *k* hops, but it has no harmful consequences.

5. Summary and conclusions

The Localized Urban Dissemination protocol provides a limited flooding by using a gossiping scheme that randomly drops packets with a given probability. In urban environments the buildings block the propagation of radio signals; hence the area that must be covered by the dissemination can be determined by using a digital map. Using information about the usual traffic conditions, the vehicles being in the junctions can decide if it is worth forwarding the message in the next road segment or not. The shape of the coverage area evolves dynamically as a result of this chain of decisions.

In theory, the LUD protocol is well suited to emergency message dissemination, as the distributed computing eliminates the long delay the source would need to spend on calculating the shape of the coverage area. It also uses locally available knowledge, like the actual traffic density, which improves the quality of the coverage area. Here, quality means that nodes that actually need the information should receive it, but the precious bandwidth of the wireless channel should not be wasted with superfluous packets.

Simulations have shown that gossiping limits indeed the message flood into the designated area. The disadvantage of the random packet drop is the high variance in the hop number the messages reach; however, a simple change in the behavior of the Forwarders, and a stored state in the packet headers can increase the reliability of the protocol to an acceptable level.

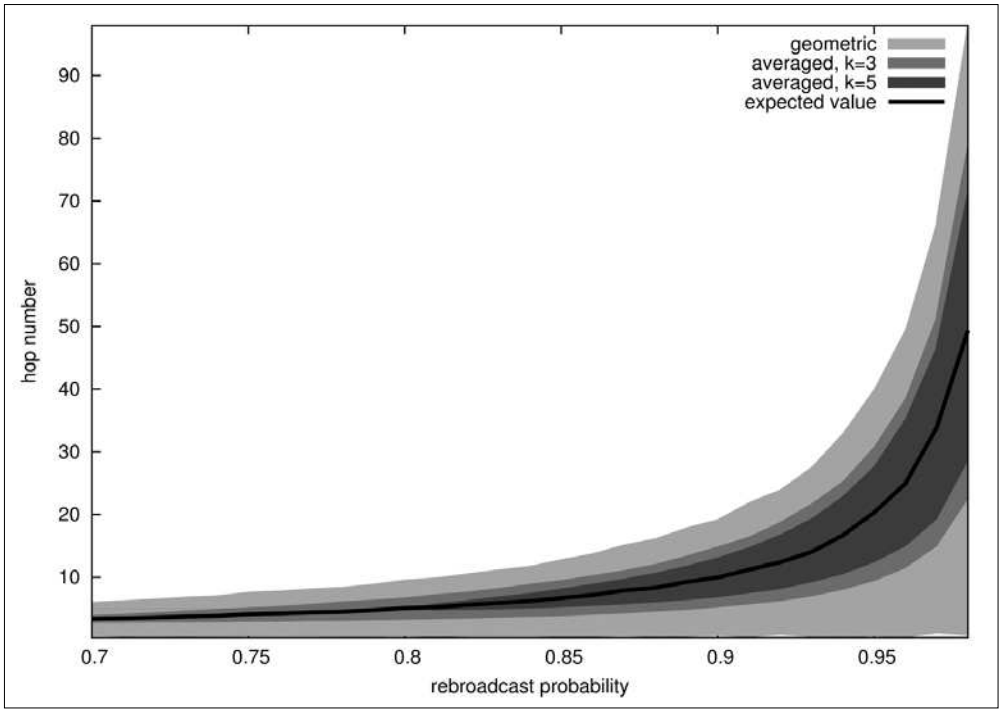


Fig. 3.
Comparison of the standard deviation of the geometric and the averaged disseminations

In the future the continued theoretical inspection might reveal other interesting properties of the LUD protocol, and we expect that its efficiency can be further increased with additional small modifications to one of its algorithms. Packet level simulations will also be needed to analyze the effects of the packet collisions and node mobility on the multi-hop dissemination.

Authors

MIKLÓS MÁTÉ is a PhD student at the Department of Telecommunication and Media Informatics of the Budapest University of Technology and Economics, where he obtained his MSc degree in 2007. His research interests include scalable routing protocols and efficient information propagation in ad hoc networks.

ROLLAND VIDA is Associate Professor at the Budapest University of Technology and Economics. He obtained his BSc and MSc degrees in Computer Science from the Babes Bolyai University, Cluj-Napoca, Romania, in 1996 and 1997 respectively, and his PhD degree from the Université Pierre et Marie Curie, Paris, France, in 2002. Between 2003 and 2005 he obtained the György Békésy Postdoctoral Fellowship, and in 2007 the János Bolyai Research Fellowship. In the last 5 years Dr. Vida has acted as organizer, TPC member or reviewer for more than 30 international conferences, participated in several national and European research project, and taught different networking courses at universities in Romania, Slovakia and Hungary. In 2008 he was elected as Chair of International Affairs of the Scientific Association for Infocommunications, Hungary.

References

[1] M. Máté, R. Vida,
"Probability-based information dissemination in urban environments",
In Proc. of Eunice 2008, Brest, France,
September 2008.

[2] A-M. Kermarrec, M. van Steen,
"Gossiping in distributed systems",
ACM SIGOPS Operating Systems Review,
Vol. 41, Issue 5, 2007.

[3] S. Boyd, A. Ghosh, B. Prabhakar, D. Shah,
"Randomized gossip algorithms",
IEEE Transactions on Information Theory,
Vol. 52, No. 6, pp.2508–2530.
June 2006.

[4] S-Y. Ni, Y-C. Tseng, Y-S. Chen, J-P. Sheu,
"The broadcast storm problem in a mobile ad hoc network",
In Proc. of the 5th annual ACM/IEEE International Conference on Mobile Computing and Networking,
Seattle, Washington, United States,
August 1999.

[5] D. Chen, J. Deng, P.K. Varshney,
"On the forwarding area of contention-based geographic forwarding for ad hoc and sensor networks"
In Proc. of SECON'05,
September 2005.

[6] H. Füßler, H. Hartenstein, J. Widmer,
M. Mauve, W. Effelsberg,
"Contention-Based Forwarding for Street Scenarios",
In Proc. of WIT 2004, Hamburg, Germany,
March 2004.

[7] G. Korkmaz, E. Ekici, F. Özgüner,
"An Efficient Fully Ad-Hoc Multi-Hop Broadcast Protocol for Inter-Vehicular Communication Systems",
In Proc. of IEEE ICC'06, Istanbul, Turkey,
June 2006.

Simulation and measurement of a MIMO antenna system

ANDREA FARKASVÖLGYI, ÁKOS NÉMETH, LAJOS NAGY

Budapest University of Technology and Economics,
Department of Broadband Infocommunications and Electromagnetic Theory

{andrea.farkasvolgyi, akos.nemeth, lajos.nagy}@hvt.bme.hu

Keywords: MIMO channel capacity, measurement of MIMO system, MIMO antennas, DB model

In this paper we present simulation and measurement results of the channel capacity of a 3?3 MIMO antenna system. The aim of this research is maximization of a MIMO channel capacity for indoor environment. The dependence of the channel capacity on the antenna position was analyzed by simulations. We have also examined the effect of the frequency dependence of the antenna system (in case of conjugate-matching and non-conjugate-matching) for the channel capacity. Based on the simulation results in the created and measured antenna system the antennas were at a right angle to each other. At the two chosen different structures we measured the antenna parameters and the channel capacity. In this paper we present the results of the measurements which clearly confirm our simulations. We will point out to the differences between the two antenna structures.

1. Introduction

Wideband indoor wireless systems are gaining increasing importance nowadays. This is why the analysis of MIMO systems which eliminate the problems of indoor propagation is of primary significance. In case of indoor propagation a frequent problem is that there are disturbing objects between the transmitter and the receiver antennas, consequently there are no direct line of sight in the wireless channel. The objects in the channel adversely affect the transmission because they scatter and reflect the signals, resulting in attenuation and phase errors. MIMO systems can be a solution to these problems.

MIMO system can eliminate the phase, distance and polarization diversity. Thus in an indoor environment the theoretically highest channel capacity can be nearly achieved. It is known that the channel capacity scales linearly with the number of antennas at both the receiver and transmitter for complex Gaussian fading channels. When designing a complete multiple antenna system we have to try to approach a maximal mean capacity with a minimal number of antennas in the system.

For multiple antenna systems an important problem is the reduction of the number of antennas for practicability and usability reasons. We will assume that the multiple antenna system with three elements on both the receiver and transmitter issues is the simplest structure for the highest mean capacity.

In this paper we present simulation and measurement results for the channel capacity of a 3x3 MIMO antenna system. The aim of this research is the maximization of a MIMO channel capacity for indoor environment. Three-dimensional (3-D) double-bouncing (DB) stochastic scattering model was used for the channel simulations. The dependence of the channel capacity on the antenna position was analyzed by simulations [1,2].

We have also examined the effect of the frequency dependence of the antenna system (in case of conjugate-matching and non-conjugate-matching) for the channel capacity. Based on the results of the simulation we have created the antenna system and measured the antenna parameters and the channel capacity. In this paper we present the results of the measurements which clearly confirm our simulations. We will point out to the difference between the two antenna structures.

2. Simulation model and calculation methods

The investigated MIMO system contains tree wire dipole antennas both on the transmitter and the receiver device.

2.1. Wire antenna analysis

Let us consider an antenna consisting of many arbitrary oriented wire elements. Starting with Maxwell equations and by enforcing the boundary condition for the total tangential electrical field on the antenna wire, it is possible to obtain the simplified general integral equation for arbitrary oriented wires. The tangential component of the electrical field generated by the current which flows on the conductor's surface is E^s_{tan} while the incident field generated by the excitation is E^i_{tan}. Then the boundary condition is

$$E^i_{\tan} + E^s_{\tan} = 0 \qquad (1)$$

on the conductor's surface. The E^s electrical field can be derived from the A magnetic vector potential due to the current I:

$$\mathbf{E}^s = \frac{1}{j\omega\varepsilon}\left(\operatorname{grad}\operatorname{div} + k^2\right)\mathbf{A} \qquad (2)$$

where

$$\mathbf{A} = \frac{\mu}{4\pi}\int_L \mathbf{I}\frac{e^{-jkR}}{R}ds \qquad R = |\mathbf{r} - \mathbf{r}'|$$

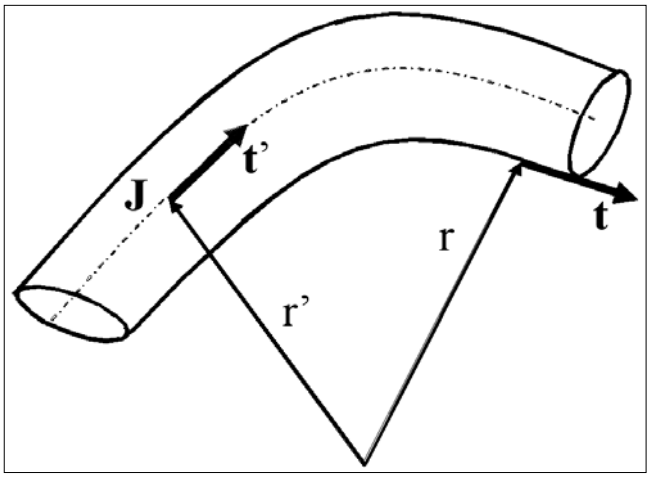


Fig. 1. Thin wire geometry

The Pocklington's procedure, applied to wire antennas, supposes the current to be located over a thin filament over the conductor *(Fig. 1)*.

The Pocklington's integral equation can be obtained finally by substituting (2) into (1).

$$E^i_{\tan} = -\frac{1}{4\pi j\omega\varepsilon}\int_L \left(\mathbf{t}\cdot\mathbf{t}' k^2 I \frac{e^{-jkR}}{R} + \frac{\partial}{\partial l'} I \frac{\partial}{\partial l}\frac{e^{-jkR}}{R}\right) dl' \quad (3)$$

The Method of Moments (MM)

Operator equation (3) has the form

$$Lf = g \quad (4)$$

where

L represents the linear integro-differential operator,

f is the unknown current and

g is the known excitation.

The *f* should be determined and MM expands by a set of *N* known expansion functions (f_1, f_2, f_3, ...), as a linear combination

$$f = \sum_n \alpha_n f_n \quad (5)$$

The α_n expansion coefficients are to be determined for the selected set of expansion functions. Substituting (5) into (4) and considering the linearity of *L*, we have the equation for *N* unknowns:

$$\sum_n \alpha_n L f_n = g \quad (6)$$

Taking the inner product of (6) with other set of functions, the weighting functions, the system of linear equations can be derived:

$$\sum_n \alpha_n \langle w_m, Lf_n \rangle = \langle w_m, g \rangle \quad (7)$$

Comparing (7) and (3), the system of equations to be solved can be written in the form

$$[Z_{mn}][I_n] = [V_m] \quad (8)$$

where

$[Z_{mn}]$ is the impedance matrix,

$[V_m]$ is the voltage vector.

The inner product is defined as a line integral over *L* and using the Galerkin method to simplify the equation (8), the impedance matrix and voltage vector elements are as follows:

$$Z_{mn} = \frac{j\omega\mu}{4\pi}\int_{s_n}\int_{s_m} \mathbf{t}_n \mathbf{t}_m f_n f_m \frac{e^{-jkR_{mn}}}{R_{mn}} ds_m ds_n + \\ + \frac{1}{4\pi j\omega\varepsilon}\int_{s_n}\int_{s_m} \frac{df_n}{ds_n}\frac{df_m}{ds_m}\frac{e^{-jkR_{mn}}}{R_{mn}} ds_m ds_n$$

$$V_m = \int_{s_m} f_m E^i_{\tan} ds_m$$

To solve (8) we used piecewise sinusoidal expansion and weighting functions.

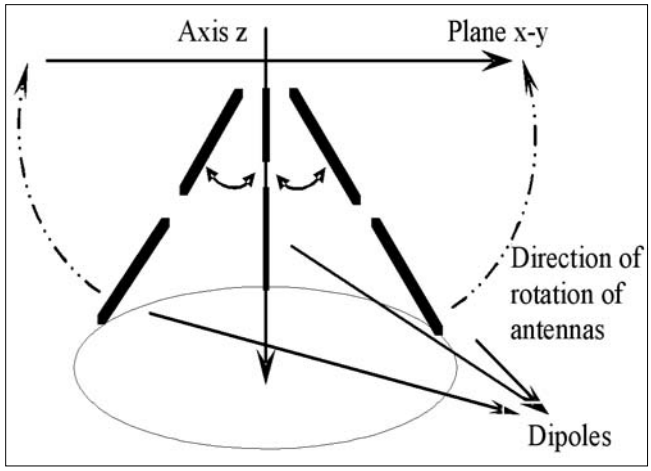
The resulting mutual impedance between MIMO antenna elements can be obtained using the *N* port analysis to the whole system of antennas.

2.2. 3D – environment simulation model

The antenna system is situated in a 3-D scattering environment indoor channel. Waves of arbitrary polarizations are incident on the antenna structure from all possible directions. The waves launch from the transmitter antennas and first they reach the elements of the primary reflection surface and from here they re-scatter to the second group of scatterers and finally they are reflected to the receiving antennas. The transmission matrix (**H**) which connects the receiver and transmitter antennas is filled by assuming DB scattering [1,2].

Our multiple antenna system is composed of M_t=3 and N_r=3 electric dipoles at both the transmitter and the receiver units. In this way, the transmission channel matrix *H* consists of nine transmission links (3x3). At the start of the simulation the antennas were oriented in the Z axis and later they were rotated toward the X-Y axes (the structure was opened like an umbrella). The radiated electric field of each dipole is applied for the calculation of the transmitter matrix. The current distribution for each electric dipole is sinusoidal, which is often assumed for finite length dipoles. *Fig. 2* shows the method of rotating of antennas in the simulation structure [3].

Fig. 2.
3x3 MIMO antenna structure for maximizing the mean capacity by rotation of the antennas, parallel at the transmitter and the receiver units, from the axis Z toward X-Y plane (it's opened like an umbrella)



This simulation model statistically describes the material, surface and motion of these objects which results in phase and amplitude error in the course of propagation. By this method we could describe the continuously varying indoor environment.

For a MIMO radio channel with channel matrix H, the SVD is given as $H=SVD^T$, where S and D^H (complex conjugate transpose of D) are complex unitary matrices, $V=diag(\sqrt{\lambda_1},....,\sqrt{\lambda_r})$ diagonal square-matrix with λ_1, λ_2,...., λ_r are the positive eigenvalues of HH^H and $r \leq \min\{Mt,Nr\}$ denotes the rank of HH^H. With the assumption of known channel at the transmitter, the theoretical capacity from water filling [4] is given as

$$C = \sum_{i=1}^{r'} \log_2(1+\lambda_i SNR_i) \qquad (9)$$

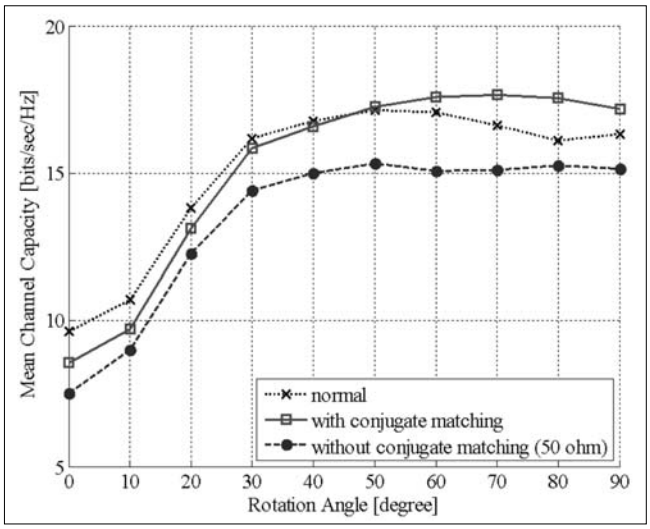
where $SNR_i=P_t/\sigma^2$ is the individual SNR of the eigenmodes after water filling and r' denotes the number of useful eigenmodes with positive power allocation [4,5].

3. Results of simulations

3.1. Effect of mutual coupling for the channel capacity

We simulated the motion of the antennas by the rotation method described above. At the beginning of this simulation the antennas are parallel with the axis Z. In the midst of the simulation the antennas opened in the space like an umbrella. At the end of the simulation the antennas reached the X-Y plane. In this case the antennas are on the farthest position, where the phase between antenna and axis Z was changing from 0° to 90°. The result of the simulation shows perfect symmetry for the X-Y plane. We look for the perfect position for the maximal mean channel capacity in consideration of the effect of mutual coupling in case of conjugate matching and non-conjugate matching. *Fig. 3* shows the mean capacity versus the angle of rotation.

Fig. 3.
Channel capacity of a 3x3 MIMO antenna system, in the normal case, in the case of mutual coupling with and without conjugate matching



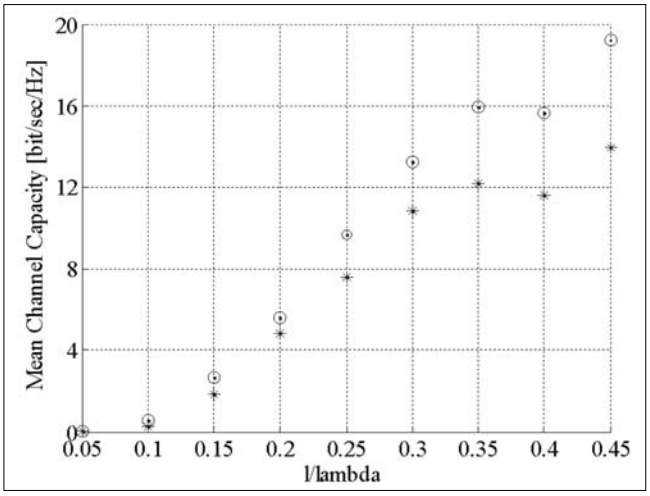
In Fig.3 when the rotation angle is zero the antenna elements of structure are close to the Z axis, and when the phase is ninety degrees the antennas are on the X-Y plane.

In normal case the maximal channel capacity is about 45°. In case of conjugate matching the maximal channel capacity is at 70° and without conjugate matching the capacity is maximal and approximately constant from 50°. We chose the 45° – structure, because the realization was the easiest in this case, since the antennas were on three different edges from one corner of a cube in this adjustment.

3.2. Alternation of frequency

In the next phase of simulation we investigated the mean channel capacity for various frequencies. It was easy to realize by changing the length of antenna at a constant wavelength. *Fig. 4* shows the result of the simulation.

Fig. 4.
Mean capacity of a 3x3 MIMO dipole antenna system versus of l/λ, with (o) and without (∗) effect of mutual coupling



We investigated the effect of the mutual coupling for 0.05 to 0.45 antenna length to wavelength ratios. The optimal is when the dipole length is $l/\lambda > 0.35$). Evidently we found that in case of conjugate matching the mean channel capacity is higher than for non conjugate matching case. According to our expectation the effect of the mutual coupling is stronger if the antenna-length is reduced.

4. Measurements

Based on the results of the simulation, the realized antenna system both on the transmitter and receiver side were built on three different edges from one corner of a virtual cube. We made two different structures: the first was like the simulated unit, the other was a modified variation of the preceeding constellation *(Fig. 8)*. The antennas are shifted from the edges to the center of faces which are touched at edges *Fig. 5*.

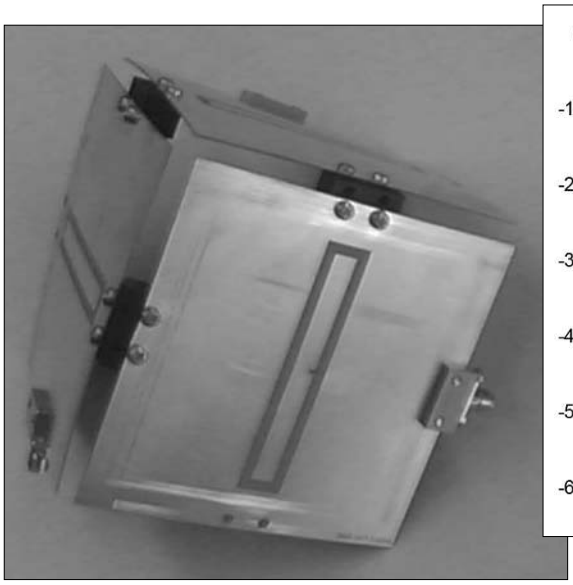


Fig. 5. Three antennas in the middle of the faces

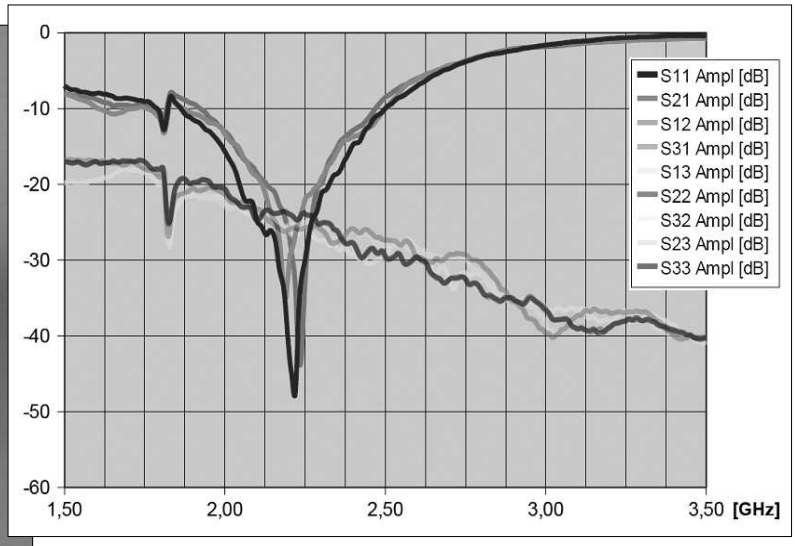


Fig. 6. Measured S-parameters versus frequency

Fig. 7.
Calculated channel capacity versus frequency

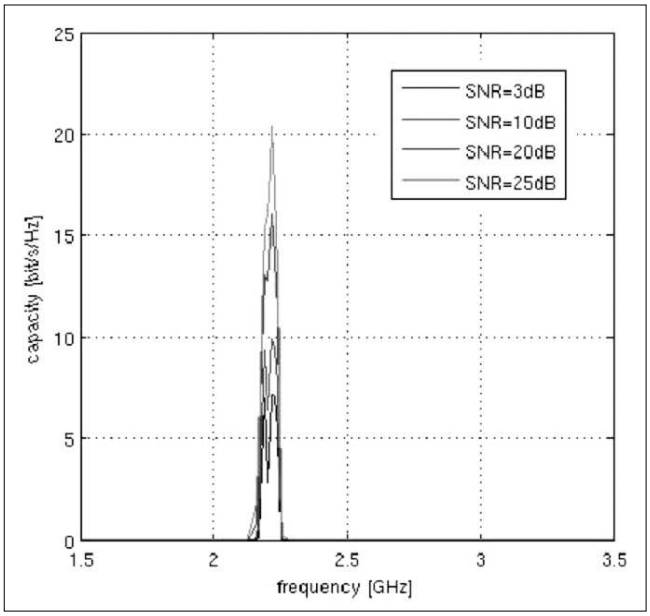


Fig. 5 and 6 show the realized antenna structures. The result of the measurement are the Sij (mutual coupling) and Sii (reflection) parameters of the antennas. Fig. 6 (antennas on faces) *and 9* (antennas on edges) show these results. From the S parameters we calculated the mutual coupling values and the mean channel capacity *(Fig. 7 and 9)*.

The results show that the second structure (antennas on the faces) is better than the first one, because it realizes the highest channel capacity. Note that the antennas in the second case (antennas on the faces) do not disturb each other, because they are placed far apart.

The next challenge will be the direct measurement of the channel capacity. So far we have carried out a test measurement.

5. Conclusions

In this paper we investigated a 3x3 MIMO antennas system. Simulations were made for analysing the effects of antenna positions on the mean channel capacity. We found that the maximum mean channel capacity is achieved by the structure in which the antennas are perpendicular to each other. We examined the frequency dependence of the antenna structure also by simulation in case of conjugate and non-conjugate matching. The simulation gave the expected results, thus the maximal channel capacity is at $l/\lambda = 0.35$ in case of conjugate matching.

Based on the simulation results in the realized and measured structure the antennas were perpendicular to each other. The measurements confirmed our results of simulations.

.

Acknowledgment

This work was carried out in the framework of a project supported by the Mobile Innovation Center, Budapest, Hungary.

Authors

LAJOS NAGY has finished his studies at the Budapest University of Technology and Economics, specialization telecommunications in 1986 and his postgraduate engineering studies in 1988, obtaining a diploma with distinction. He has a Dr. Univ. degree (1990) and a Ph.D. degree (1995). At present he is Assoc. Professor and Head of the Department of Broadband Infocommunications and Electromagnetic Theory at the Budapest University of Technology and Economics. His research interests include applied electrodynamics, mainly antenna design, optimization and radio frequency propagation models. Dr. Nagy is Secretary of the Hungarian National Committee of URSI and the Hungarian delegate to the Section C of URSI. He is leading the Hungarian research teams in COST 248 and ACE2 EU projects. He has published over 100 papers.

Fig. 8. Three antennas on the edges

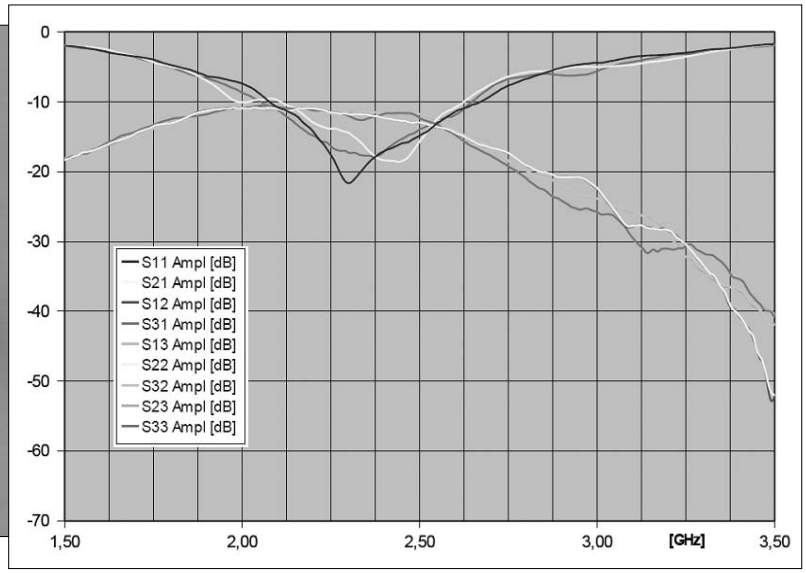
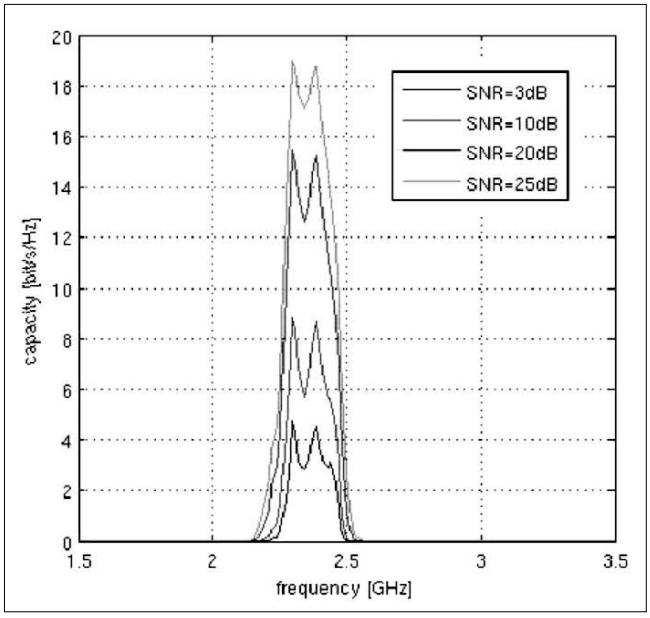


Fig. 9. Measured S parameters versus frequency

Fig. 10.
Calculated channel capacity versus frequency



ÁKOS F. NÉMETH was born in Budapest, Hungary in 1981. He received the MSc degree in Electrical Engineering from the Budapest University of Technology and Economics (BME) in 2005. Since the graduation he is working on the PhD degree at the university's Department of Broadband Infocommunications and Electromagnetic Theory. Currently he is also working for the Pannon GSM Telecommunications Plc as RF engineer. He has been doing research in the field MIMO systems since 2006, focusing on mutual coupling based on the small spatial distance between the antenna elements of the antenna systems.

ANDREA FARKASVÖLGYI received the MSc degree in Electrical Engineering from the Budapest University of Technology and Economics (BME) in 2002. She finished the PhD school at the Department of Broadband Infocommunications and Electromagnetic Theory in 2006. Presently she is an assistant lecturer at the same department. Her main topics are MIMO antenna systems and channel analysis.

References

[1] B.N. Getu, J.B. Andersen,
"The MIMO cube-A compact MIMO antenna,"
IEEE Trans. Wireless Commun., Vol. 4, No. 3,
pp.1136–1141., May 2005.

[2] B.N. Getu, R. Janaswamy,
"The Effect of Mutual Coupling on the Capacity of the MIMO Cube,"
IEEE Wireless Propagation Letters, Vol. 4,
pp.240–244., 2005.

[3] A. Farkasvolgyi, L. Nagy,
"Optimization for MIMO antennas system,"
IST Conference, 2007.

[4] http://www.mathworks.com/
matlabcentral/fileexchange/
loadFile.do?objectId=12491&objectType=file

[5] B. Vucetic, Jinghong Yuan,
Space-Time Coding. Wiley, 2004.

[6] L. Zombory, M. Koltai,
Computer analysis of electromagnetic fields.
Műszaki Könyvkiadó, (in Hungarian), 1979.

[7] R. E. Collin,
Antennas and Radiowave Propagation.
New York, McGRaw-Hill Book Company, 1985.

[8] Mats Gustafsson, Sven Nordebo,
"Characterization of MIMO Antennas Using Spherical Vector Waves,"
IEEE Transactions on Antennas and Propagation,
Vol. 54, No. 9, Sep. 2006.

[9] Chi Yuk Chiu*, Ross D. Murch,
"Experimental Results for a MIMO Cube"
IEEE Transactions on Antennas and Propagation.

[10] R. Janaswamy,
"Effect of element mutual coupling on the capacity of fixed length linear arrays,"
IEEE Antennas Wireless Propagation Letters, Vol. 1,
pp.157–160., 2002.

A client-driven mobility frame system – Mobility management from a new point of view

BENEDEK KOVÁCS, PÉTER FÜLÖP

Budapest University of Technology and Economics, Department of Telecommunications
{bence, fepti}@mcl.hu

Keywords: mobility management, CMFS, modelling, client-based, handover

In this paper a new mobility management approach is introduced. The main idea in this approach, that not the network but the mobile node should manage the mobility for itself (similarly to the IP concept). The network nodes provide just basic services for mobile entities: connectivity and administration. We construct a protocol called the Client-based Mobility Frame System (CMFS) for this mobility environment. We propose some basic mobility management solutions that should be implemented in the mobile clients and provide details about a working simulation of a complete Mobility Management System. Examples of mobility management approaches such as the centralized- and hierarchical or cellular-like ones are also defined and hints are given what kind of algorithms might be implemented upon the Client-based Mobility Frame System. After the theoretical analysis a simulation shows the applicability of the newly introduced protocol framework.

1. Introduction

Seamless information mobility is a requirement in the today's world. Although there are many operating solutions there is still need for IP mobility since IP is the most widespreadly used protocol. The communicating equipments are identified with their permanent IP address and the communication is done on IP networks. Many works have discussed the problem of managing the movement of the clients since the Internet was designed to be static and does not support mobility by itself. There are different solution proposals for the problem and all of them have their drawbacks and good features.

If one takes a close look at these systems they always deal with the tradeoff between complexity (simplicity) and optimality. Naturally, this can not be resolved but we will transform it into another dimension: from network level to individual level.

In this paper we introduce an agent based mobility management strategy. This is an alternative point of view and it could be easier to implement our solution, than the classical ones. We do not say that we have found the optimal system to provide IP or other kind of mobility but we will come up with a new idea and framework which is very different from the classical approaches and can be the most cost-efficient in many cases.

The basic idea is that, unlike in the GSM or Mobile IP (both IPv4 and IPv6), the network will no longer have to provide any logic for the management algorithm. The whole network can remain simple and the nodes will only have to handle simple commands by recognizing, executing and forwarding simple messages generated by the mobil entity itself. The management system is implemented in the mobile client, consequently each node is able to choose the most suitable mobility for itself on the same network.

We show how to apply the classical strategies like cellular or hierarchical approaches to our system. First we will present a protocol description and define the Client-based Mobility Frame System then we give simple mobility applications like the Mobility Management Systems itself.

2. Client-driven Mobility Frame System

We have introduced the new idea and explained the basics of its operation. In this Section we will define a Client-driven Mobility Frame System (CMFS) specifying the basic roles in the network and what capabilities the fixed network nodes are required to have to be able to communicate with the moving entity. A simple method will be given for the mobile node to discover the service network and build up its own logical network. We will handle the cleansing of the service network database.

2.1. Some basic notations

Since there are many kind of notations in this field, to avoid misunderstandings, some of the basic ones we use are defined here. The mobility model will be the same abstract one as in one of our previous works [1]:

- The *Mobile Nodes* (MN, alias mobiles, moving entities) are the mobile devices who want to communicate to any other mobile or fixed partner.
- There are *Mobility Access Points* (MAP) as the only entities who are capable to communicate with the Mobile Equipments. (Note: mobility does not necessary imply radio communication. It means only that the Mobile Node changes its Mobility Access Points and when it is attached to one, communication between them can be established).
- The *Mobility Agents* (MA) are network entities running the mobility management application.

- There is a *Core Network* that provides communication between the Mobility Access Points and has a structure that can be described with a graph. Vertices are either Mobility Access Points or Mobility Agents other serving nodes who is not part of the mobility management application and the edges can be any kind of links (even radio links) for the data communication between the vertices.
- *Home Agent*, a special MA that is a kind of basis to the Mobile just like in the Mobile IPv4 [6] or the Mobile IPv6 [7] case or the HLR in the GSM case. Whenever the MN is paged its exact or approximate location can always be found in the database of this node.

2.2. The idea

The client driven mobility management we introduce is inspired and based (but does not depend!) upon the fact that a mobil user typically moves within a range of access points and rarely leaves to far away agents. In order that the mobile could manage its own mobility it has to maintain a database of the nodes it communicates with. This is called the Logical Network (LN). The MN always should be able to have an up-to-date information of the nodes of this network. The size of this depends on the algorithm the mobil uses.

To give an example we will show later that to implement a basic Mobile IPv4-like solution on our framework system the MN only has to maintain information about 3 (or even only 2) nodes in the network so for a node with very limited capacity this can be a good enough solution.

We want to point out that the most important advantage of our solution is that the service providers do not have to choose an exact mobility approach, which could be very inefficient. The mobile nodes in CMFS can choose the optimal algorithm for themselves, thus the mobility solution can be the most cost-efficient and adaptable for various circumstances.

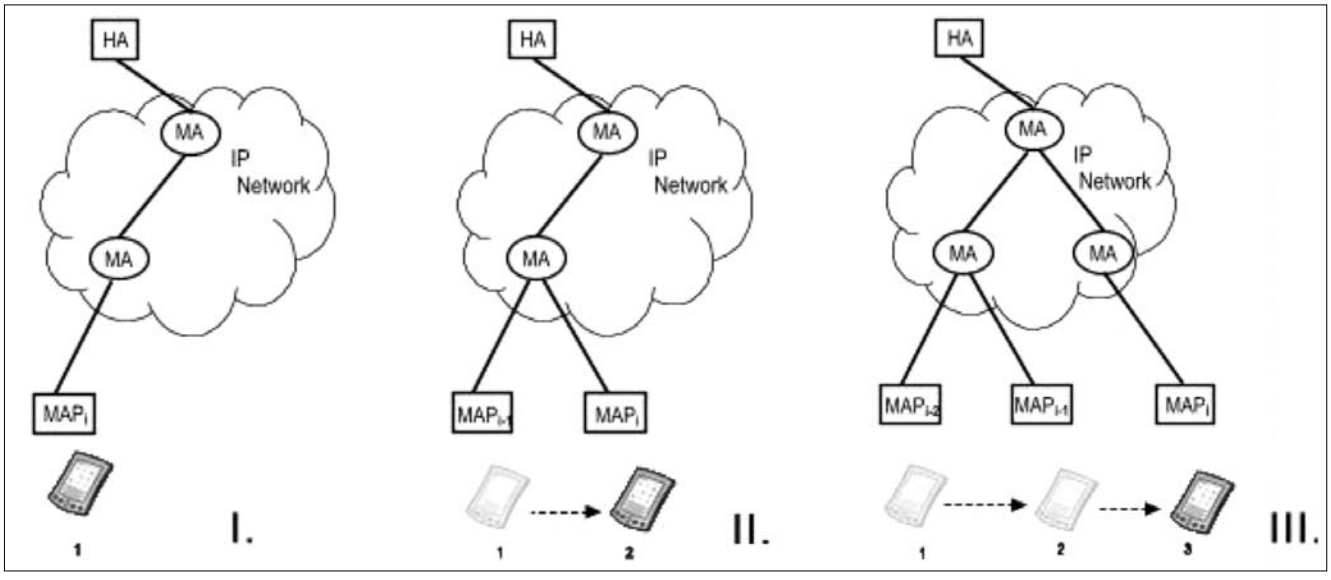
2.3. Network discovery

In order to implement more complex mobility managements, the MN should construct and maintain a larger logical network, it have to get to know the network entities. There are several algorithms to discover heterogeneous IP networks. In this paper, we do not focus on the selection of the most efficient or optimal one. We introduce a simplest method and aim to prove that the system we propose actually works.

Although the simplest would be to use the special IP packet options value for network layer packet tracing, it is not feasible since most network entities are not able to interpret these packets due to lack of implementation and poor specification. The other possibility is to use the *traceroute* application or anything like that which does not depend on any facilities and use the TTL (Time To Live) function of the IP. By using small TTL values which quickly expire, traceroute causes that the routers along a packet's normal delivery path generate automatically an ICMP Time Exceeded message. We use a similar method in our own protocol specification. We implement it in the update procedure. Let us examine this update and network discovery function without a precise protocol specification. (As for the protocol implementation, see the next section).

For a basic algorithm like MIP [6] at the first step, the Mobile Node (MN) registers with its Home Agent (HA) that is the first Mobility Access Point (MAP) in the Logical Network (LN). Then the MN moves to a Foreign Network (FN). It gets a Care-of-Address (CoA) as in the MIP solution but also tells the MAP what it should do: whether it should register with the Home Agent (HA) or with any other Mobility Agent (MA) etc. If the MN wants to know the path to the HA, then it sets a bit in the *register* message, which triggers *reply* messages with a timestamp from all, or only from specific MAs on the way. The timstamp could be used to determine the weight of the links towards the MAs. The MN records the discovered MAs to its database and to the Logical Network it maintains.

Fig. 1. The logical network build-up



Based on this information it can implement more complex management algorithms. When it moves to another network and communicates with another MAP it can delete the former one from its database and proceed. This is how the MN can maintain the whole network it has to know. More complex Network Discovery procedures are going to be discussed when needed.

2.4. The requirements for the network MAs

To be able to serve the MNs and their algorithms we define requirements for the network. Once, the MAs are aware of all of them, an MN can use any kind of Mobility Management Strategy (MMS) for itself. This also means that different terminals are allowed to use the most suitable MMS for themselves.

Since all the management is MN initiated, the MA has to provide a kind of routing function. The HA should always know where to route a packet towards the MN, or drop the call. There should be a database registry for this, for example an association between the MNs permanent IP (Care of Address, CoA) and routes: where to forward the packet towards the CoA. All the MAs should work in a similar way. Once the MN with its CoA is paged at an MA it should route the packets to the MAP of the MN. If there is no route to the MN it should simply drop the packets. How can an MN register to an MA? When it attaches to a new MAP after a successful handover it naturally registers there. It also adds the information whether this MAP should continue the registration process to an MA in an upper level or not.

Let us construct such a message:

```
[Dst: MAPi, Src: MN, Actions: Register MN to MAPi via MN;
  [Dst: MAj , Src: MAPi, Actions: Register MN to MAj via MAPi,MAPii,MAPiii;
        [Dst,Src,Actions: ; ;
              [ ...
              [Dst: HA, Src: MAn, Actions: Register MN to HA via MAn]
              ]
        ]
  ]
].
```

What the MA should do is to understand this message and maintain the following entry in its database: if the paged node is MN then it should be searched via HA, MA_n, ..., when MN is searched at MA_j then MA_j knows that it can be reached MAP_i, MAP_{ii}, MAP_{iii} meaning that the packet is routed to all the 3 nodes representing a CIP-like algorithm [2]. If there is no such multiple route and the messages do not always contain the HA, then a HMIP-like protocol [3] is implemented. If there is an update that goes from MN directly to HA, then a MIP-like approach [6] is implemented, if these last two kind of messages are mixed then a DHMIP-like approach [5] is presented. If the node sends messages like

```
[Dst: MAPi, Src: MN, Actions: Register MN to MAPi via MN;
  [Dst: MAPi¡1 //The former node//,
        Src: MN, Actions: Register MN to MAPi¡1 via MAPi
  ]
]
```

then a HAWAII-like protocol [8] is implemented.

In case of wireless tracing (for example LTRACK [4]), two different messages would be sent:

```
[Dst: MAPi, Src: MN, Actions: Register MN to MAPi via MN;]

[Dst: MAPi¡1 //The former node//,
Src: MN, Actions: Register MN to MAPi¡1 via MAPi]
```

We have provided an implementation example that can be modified after reasonable discussions but just like IP or any kind of protocol it should be standard in any network the MN wants to communicate in.

2.5. CMFS Protocol

For specification of the aforementioned command structure we developed an application layer protocol called CMFS Protocol (CMFSP). A CMFSP message is carried in UDP packets. We have chosen it instead of the TCP because the TCP does not operate well with the radio interfaces. The TCP conceives the high bit error rate of the radio channel as congestion, and decreases the window size that ends in significant speed fall-off. For this reason the mobility applications generally use UDP to the communication.

The CMFSP message structure follows strict rules as it can be seen in *Fig. 2*.

The header contains 4 fields of 1 byte elements, a *type, length, flags* and *number of actions* and a 4 byte element. Presently two different types of CMFSP messages are differentiated, a *request* and *reply* message. The *length* shows the full length of the CMFSP packet included the header. The *destination* tells the node that it has to process the message. The first bit of the flags field is the trace bit (F). If it is set to 1, it means the first MA on the way to the HA must send a CMFSP reply message, in order that the MN be able to build its Logical Network. The second bit of the flags is the specified trace bit (S). In this case only the MAs must send reply, which are labelled in the CMFSP message. The third bit (L) means all the MAs on the way to the HA must send a CMFSP reply message. The last bit of the flags (C) is set if MN wants to get capacity information from the MAs.

The *payload* of the CMFSP message contains the actions, which have to be accomplished at the specific nodes. One can see that there are three different kinds of actions defined. The first is the *Register* action that indicates a route registration in the given MA via the given destination to a specified target. The second type is the *Delete* action that erases the specified registered data from the MA database. The *Send* action type instructs the MA, that the payload of the action field has to be send as a CMFSP message.

3. Examples of mobility management strategies implemented in CMFS

The good thing in the Client-based Mobility Frame System is that not just all the MNs can use different handover management strategies but a single MN can switch between them easily upon request or in a seamless way

if implemented so. What the MN does is collecting the network parameters and makes decisions upon them and commands the network nodes accordingly with the messages defined above. Here we will show how to implement the most common mobility approaches like solutions into our Client-based Mobility Frame System.

3.1. Personal Mobile IP – PMIP

The operation of Personal Mobil IP is simple and easy. Once the MN attaches to MAP it registers itself to the HA. The operation is very similar to MIP and has a great advantage. The MN has to make no extra computation and has to maintain no extra database while there are always a few routes in the MAP.

[Dst: MAPi, Src: MN, Actions: Register MN to MAPi via MN;
[Dst: HA, Src: MAPi, Actions: Register MN to HA via MAPi];
[Dst: MAP$_{i}$1, Src: MAPi, Actions: Delete MN in MAP$_{i}$1 via MN]
].

Where the second message is needed only if clearing the network is up to the MN unlike in MIPv4. This solution is referred as pure PMIP (P-PMIP).

The simple PMIP protocol operates alike MIP and has approximately the same capacity consumption as well as we will see later. We would like to point out that the MN has to maintain a Logical Network of three nodes only. However, a great benefit of our proposal is that any MN can implement different version (e.g. soft handover) of the protocol without any modification in the network entities.

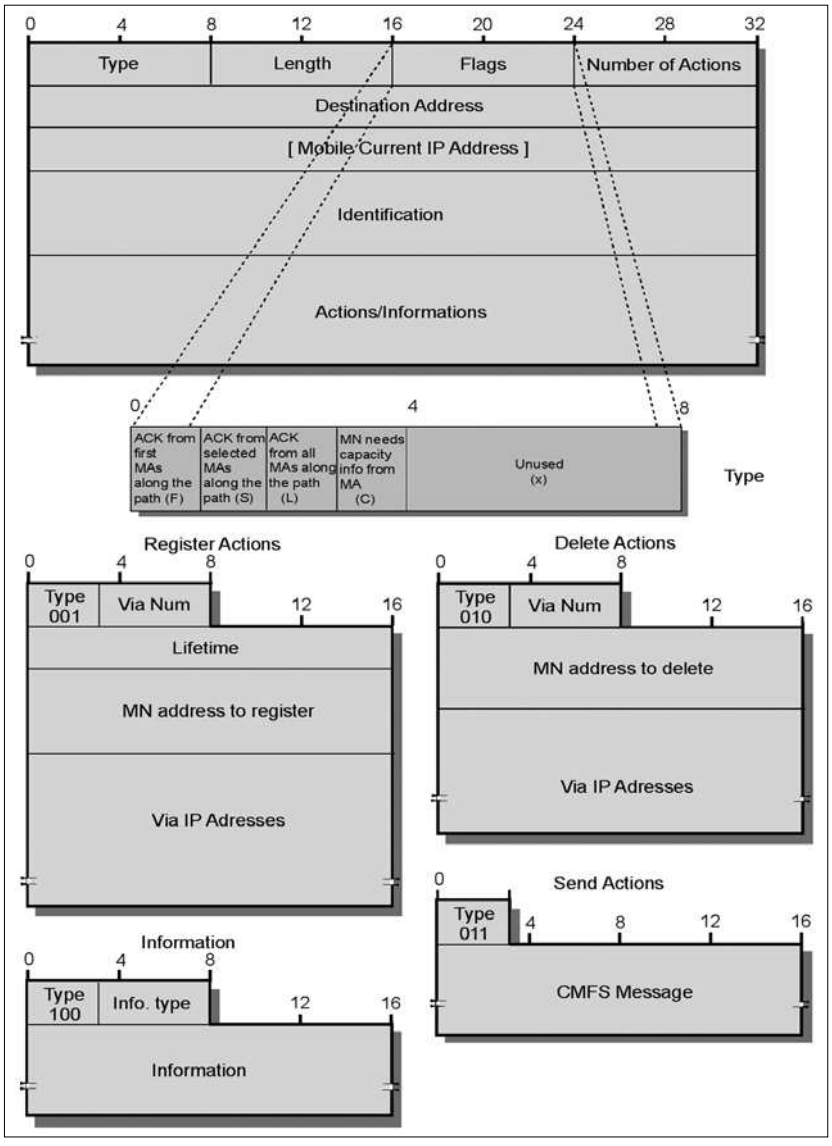
Then the Extended PMIP (E-PMIP) is an example of extension of PMIP when there is no packet loss and no obsolete routes in the databases of the MAs but of course the messages are more complex.

One can see what happens in case of a handover on *Fig. 3*.

[Dst: MAPi, Src: MN, Actions: Register MN to MAPi via MN;
[Dst: MAP$_{i}$1, Src: MAPi, Actions: Register MN in MAP$_{i}$1 via MAPi;
Delete MN in MAP$_{i}$1 via MN;
[Dst: HA, Src: MAP$_{i}$1, Actions: Register MN to HA via MAPi;
Delete MN in HA via MAP$_{i}$1;
[Dst: MAP$_{i}$1, Src: HA, Actions: Delete MN in MAP$_{i}$1 via MAPi
]]]]

The performance analysis can be found in Section 6.

Fig. 2. CMFSP message structure



3.2. Personal Hierarchical Mobile IP – PHMIP

The operation of a HMIP micro-mobility (talking about an only two-layered hierarchy) would pose the question: which node should be the MA in the hierarchical mobility approach. We suppose that seeing the traceroute messages, the MN can decide it. The messages are again simple and easy to construct.

More problems arise when talking about multiple layered hierarchical solutions. The MN has to make complex calculations for setting up the network tree but still the only problem will be to locate the logical junctions in the node (those MAs which are not MAPs). However, once this is solved the implementation again easy since there is no need to configure the network itself and implement the protocol in a static way.

Now let us give a simple method to choose the MAs that will be used to construct the hierarchy tree of the network. At the beginning the MN is attached to its HA then it moves to another MAP. The MN records all the MAs along the way (from the MAP to HA). Then when it makes a handover it records the way again. The first common element of the route (from the MN) is then dedicated to be a Hierarchy Point.

This method is very easy to implement and rather simple. We show in our simulation work that it still outperforms the basic protocols.

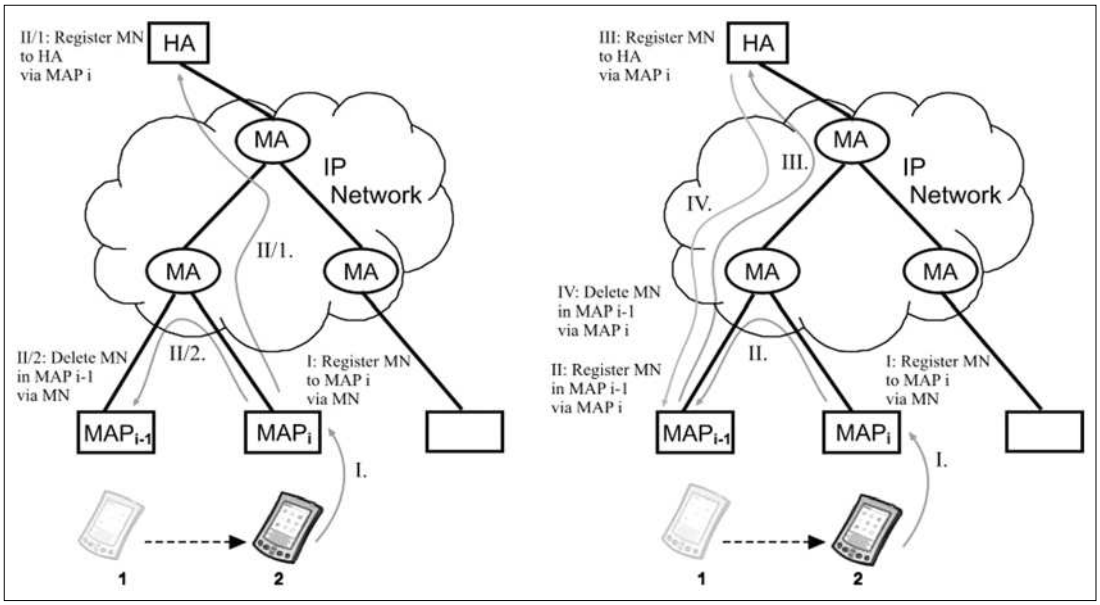


Fig. 3.
In the left figure one can see the basic P-PMIP protocol, while the figure on the right depicts the operation of the action-linearized Personal Mobile IP Mobility Management System (E-PMIP) with soft handover mechanism.

Fig. 4.
The operation of PHMIP

3.3. Personal Tracking Mobile IP – PTMIP

A tracking-like (see *Fig. 5*) solution would be again easy to implement. In this case the tracking handover is introduced when the MN orders the new MAP to report always only to the previous MAP it was attached to like in the DHMIP [5] or LTRACK [4] protocols.

When the MN is paged the message is sent through all the nodes along the way. For this reason, after a number of tracking handovers the MN performs a normal handover i.e. registers back with the HA (or to some hierarchy point in a more complex solution).There are many proposed methods to decide between the two types of handovers. In our simulation we implemented a simple suboptimal solution when the MN registers back at every *i*th step.

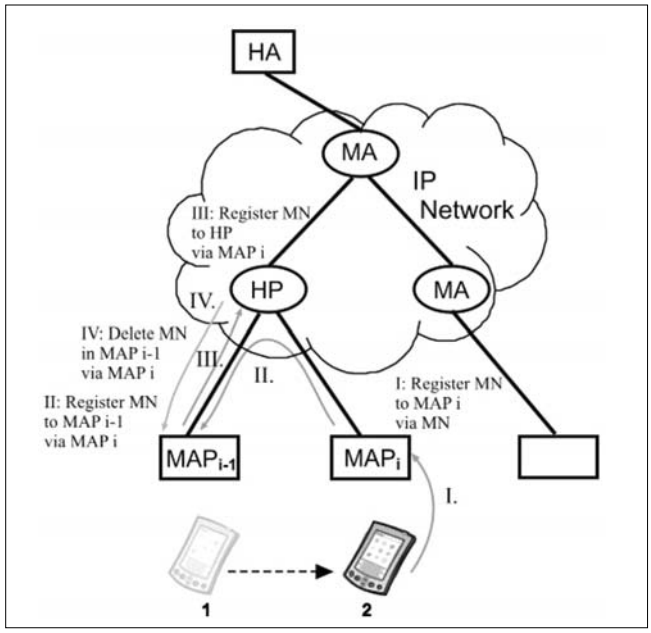
3.4. Personal Cellular Mobile IP – PCMIP

Since the widespread use in GSM the cellular solutions became popular in most mobility applications. The idea is to avoid registrations when the MN moves within a given set of MAPs but then it has to search for it at each MAP when it is paged. There is a extensive literature of cell forming algorithms. We give an alternative one.

We want to point out that in this case the paging areas are different for each MN and are formed in an almost optimal way by each MN individually. We expect better performance in large networks. The MN should send registration messages only when it moves to a new Paging Range (PR).

In this case it orders the leader of the new Paging Range to register at an upper level that the MN is in the PR. The MN also tells the IDs of the MAPs in the Paging Range (PR) to the leader of the PR so that the latter be aware who to broadcast the messages when the mobile is paged.

The following message tells to the specific MAP (the leader) the MAPs (MAP$_i$, MAP$_{ii}$) belonging to that given PR:



[Dst: MAPleader //The leader of the paging area//,
Src: MN, Actions: Register MN to MAPleader via MAPi, MAPii, ... ,
[Dst: HA, Src: MAPleader, Actions: Register MN to HA via MAPleader
]
]

The problem to solve for cellular algorithms is the problem of forming he Paging Ranges. Forming the cells at an optimal cost using the total frequency of handovers on aggregate level (not individually for each MN) is NP hard. Consequently, the problem is NP hard for only one MN too. However, there are alternative solutions giving a solution that is good enough.

4. Simulation and numerical results

We have made a simulation to show at first that or proposed method actually works and secondly to compare it with existing technologies. The simulation was written in the open source OMNet++ [13] using C++ language.

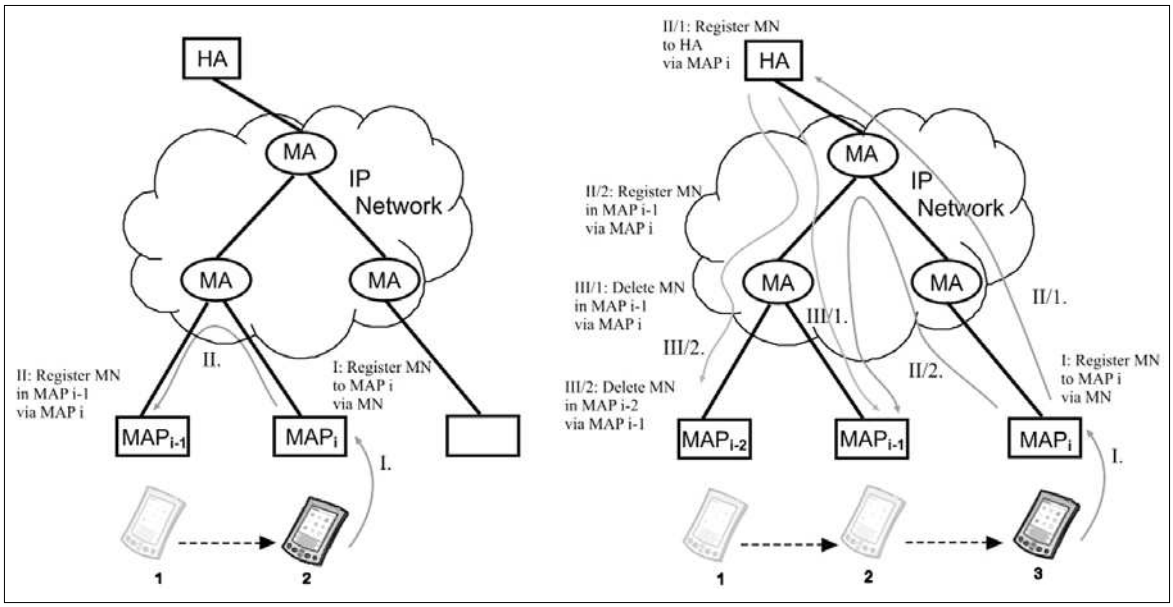


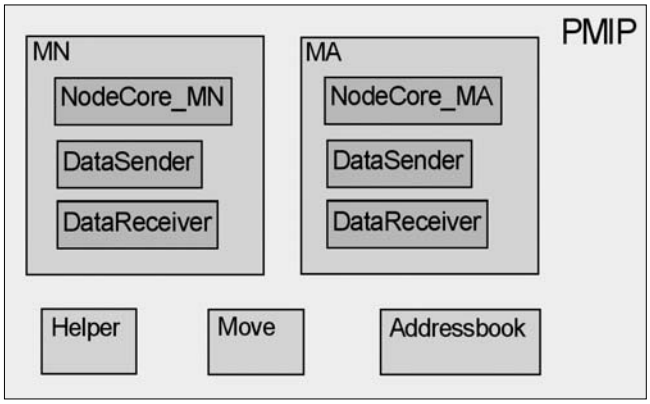
Fig. 5. The operation of the Personal Tracking Mobile IP protocol. The tracking handover is depicted in the figure on the left while the figure on the right is about the normal handover.

The simulation consists of two main modules, namely MN and MA, and some other simple components that are needed to model the operation environment *(Fig. 6)*. The two main modules have similar internal structure. Both has a *DataSender* and a *Data Receiver* to be able to send and receive messages while their logic is hidden in *NodeCore MN* and *NodeCore MA*, respectively.

The whole CMFS protocol is implemented in the Node Core components. The NodeCore MN constructs CMFS messages, maintains a database and builds up the Logical Network. The NodeCore MA understands the CMFS messages and executes the actions, maintains the database and routes the messages and packets using it.

The DataSender module creates traffic in the network to a random target and at random times while the DataReceiver is responsible for receiving and analyzing it. The number and size of packets, the frequency of sending data and the possible targets for a node can be set as a parameter of the simulation. The receiving side measures the average number of handovers, number of arrived/sent/lost packets and their averages in 1 min but can be extended to record other QoS parameters like delay or jitter too.

Fig. 6.
The component structure of the simulation of CMFS written in OMNet++



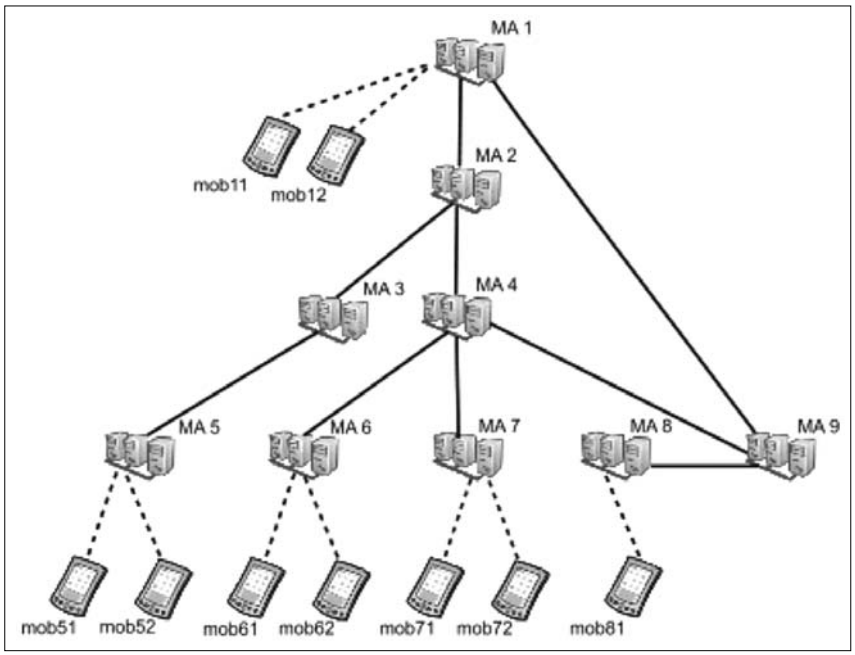


Fig. 7. The test-network used in the simulation

The *Addressbook* module is the template for the databases in the MAs. The module Move is responsible for the directions and frequency of movement of the MNs. The Helper component implements some functions and objects that are not logically part of any of the above ones.

We have constructed a virtual test environment consisting of 9 MAs and 9 MNs with the initial MN distribution depicted in *Fig. 7*.

We have run the simulation on various mobility parameters for all the algorithms separately. All the nodes made calls according to a Poisson process to random targets with a biased uniform distribution so about 80% of the calls were terminated at mobile clients. The mobility ratio (number of handovers per received call) was varied to show how it affects the performance.

The performance of the protocols is depicted in *Fig. 8*. However at low mobility level (when there are only a few handovers between two calls) E-PMIP is better than the classical MIPv4 but as the mobility ratio increases the protocol performs worse in terms of signalling load on the network. It is because it requires more operations and messages in the network to provide better QoS parameters. We can see that the P-PMIP is always better than the MIPv4. This is because if we look at the two protocols both have the same signalling strategy but MIPv4 needs Agent Advertisement messages to maintain connectivity while in the client based system it can rely on lower layers. We can conclude that the basic solutions work at approximately the same costs. However, E-PMIP shows that it is possible to improve the performance while not changing the protocol at all (only on the MN side).

In the simulation we implemented the PTMIP also, and we examined it with different tracking handover numbers. *Fig. 9* shows the results. More interesting simulations can be applied in the OMNET++ framework developed by us, but the most important conclusion can be seen: the CMFS is correctly works, and all the well-known mobility protocols can be implemented in it.

5. Conclusion

We have introduced a mobility management system that solves IP mobility from a very different point of view than any other mechanism known so far.

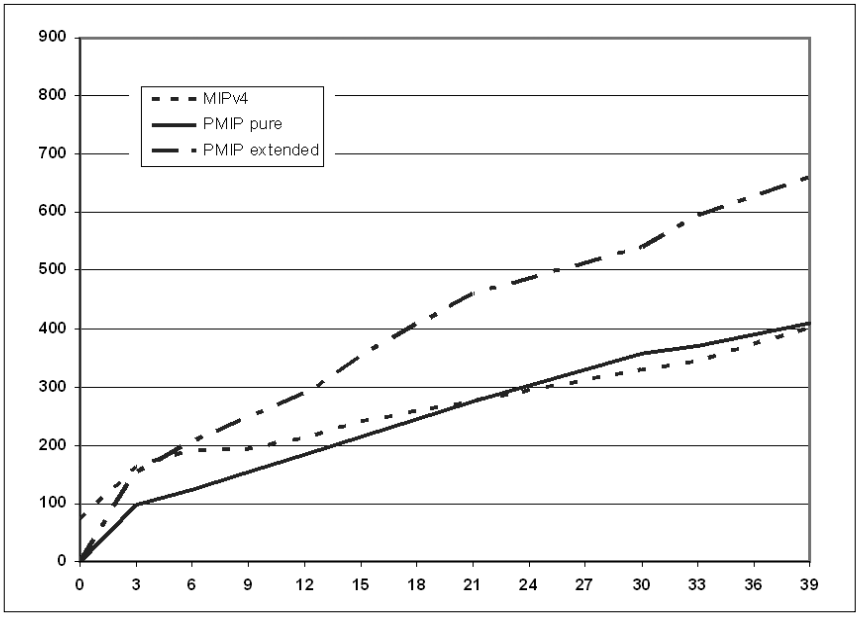


Fig. 8.
Comparison of the three mobility management systems MIPv4, PMIP pure and PMIP extended.
The horizontal axis shows the number of handovers between two arrived calls while the number of bytes transmitted on the network by each protocol is presented on the vertical axis.

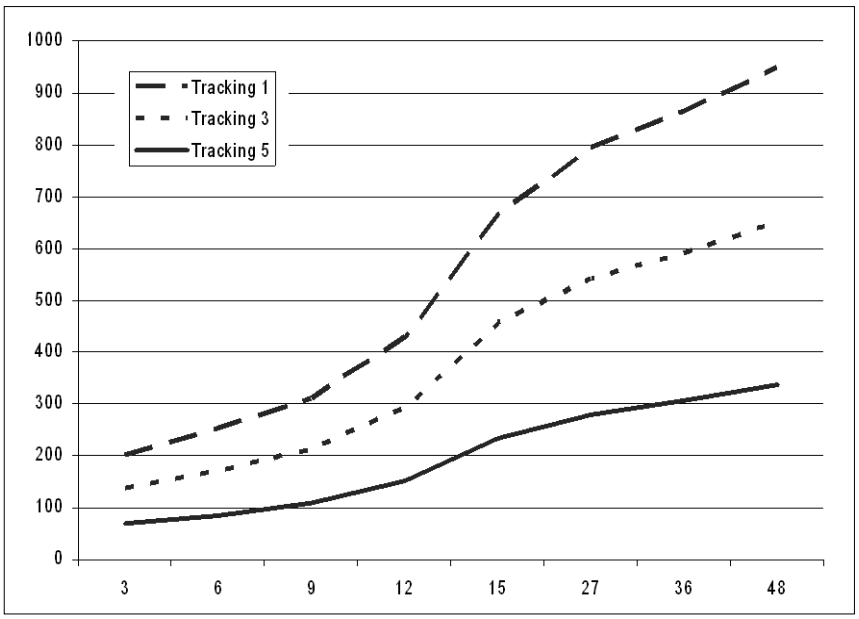


Fig. 9.
This figure depicts the performance of three tracking-like approaches namely PTMIP with 1, 3, 5 tracking handovers. Note that for this simulation a couple of additional links were inserted into the network.

We have shown example algorithms taking ideas from classical solutions. We prepared a simulation and tested our protocol in operation. Using it we compared the performance of some basic solutions and we have shown that extensions may be beneficial for both the MN and the network. Further extensions are possible: since the MN records the details of a MAP it can also perform quality measurement or reliability measurement, thus classify the MAPs and networks and use this information in the future (for example when multiple MAPs are available).

We have shown how CMFS would work over IP. However, it is rather simple to extend the whole to IMS too.

Authors

BENEDEK KOVÁCS received his M.Sc. degree in Technical Informatics from the Budapest University of Technology and Economics in 2006. Currently he is a PhD student on the faculty of Mathematics at the same university on the Department of Mathematical Analysis and member if the High Speed Network Laboratory, Hungary. Currently he is doing research work at Ericsson Telecommunications Hungary, Traffic Lab. His research interest is IP Mobility, Overload and Load Regulation methods and intensity estimation of Stochastic processes in telecommunications and parameter estimation of dynamic processes. The topic of the upcoming PhD theses is the optimal control of dynamical stochastic systems.

PÉTER FÜLÖP received his M.Sc. degree in Technical Informatics from the Budapest University of Technology and Economics in 2005. Currently he is a PhD student on the faculty of Informatics at the same university on the Department of Telecommunications. He is working as a system test engineer at Ericsson Telecommunications Hungary at Research and Development Department. His research interest is IP mobility, interworking of heterogeneous mobile networks and movement modeling in cellular, mobile networks. The PhD thesis is going to be written on Complex Mobility Management Systems and Applications.

References

[1] Kovács B., Fülöp P., Imre S.
"Extended Mobility Management Framework",
International Conference: 6th Computer Information Systems and Industrial Management Applications Conference Proceedings, CISIM 2007,
pp.191–196., 2007.

[2] A. T. Campbell, J. Gomez, A. G. Valkó,
"An Overview of Cellural IP",
Wireless Communications and Networking Conference IEEE, Vol. 2, pp.606–610., 1999.

[3] C. Castelluccia,
"A Hierarchical Mobile Ipv6 Proposal",
INRIA Technical Report, pp.48–59, 2000.

[4] Kovács B., Szalay M., Imre S.,
"Modelling and Quantitative Analysis of LTRACK – A Novel Mobility Management Algorithm",
Mobile Information Systems, Vol. 2, No.1,
pp.21–50., 2006.

[5] W. Ma, Y. Fang,
"Dynamic Hierarchical Mobility Management Strategy for Mobile IP Networks",
IEEE Journal of Selected Areas In Communications, Vol. 22, No. 4,. pp.664–676., 2004.

[6] C.E. Perkins,
"Mobile IP",
IEEE Communications Magazine, Vol. 40, No. 5,
pp.66–82., 2002.

[7] W. Fritsche, F. Heissenhuber,
"Mobile IPv6 –
Mobility Support for the Next Generation Internet",
IAB GmbH, 2000.

[8] R. Ramjee, T.La Porta, S. Thuel,
K. Varadhan, L. Salgarelli,
"A Hierarchical Mobile IP Proposal",
INRIA Technical Report, 1998.

[9] S. Das, A. Misra, P. Agraval, S. K. Das.
"TeleMIP: Telecommunications-Enhanced Mobile IP Architecture for Fast Intradomain Mobility",
IEEE Personal Communications, Vol. 7, No. 4.,
pp.50–58., 2000.

[10] Szalay M., Imre S.,
"Hierarchical Paging –
A novel location management algorithm",
ICLAN'2006 – International Conference on Late Advances in Networks, 6-8 Dec. 2006, Paris, France.

[11] K. D. Wong, W. W. Lee,
"Likelihood of Lost Binding Updates when Mobile Nodes Move Simultaneously",
MOMM'2005 – Third International Conference on Advances in Mobile Multimedia, 2005, pp.9–19.

[12] Kovács B., Fülöp P., Imre S.,
"Mobility Management Algorithms for the Client-driven Mobility Frame System – Mobility from a Brand New Point of View",
MOMM'2008.

[13] http://www.omnetpp.org/

Note-based sound source separation of polyphonic recordings

KRISTÓF ACZÉL, ISTVÁN VAJK

Budapest University of Technology and Economics, Department of Automation and Applied Informatics
{aczelkri, vajk}@aut.bme.hu

Keywords: polyphonic music, separation, instrument print, energy split

Decomposing a polyphonic musical piece to separate instrument tracks has always been a challenge. Isolating the tracks is out of reach of today's technology. This article proposes a novel method for the separation of monophonic musical recordings. The architecture of the proposed separation system is given. It uses samples of real instruments for regaining the missing data, thereby allowing for the separation and correction of recordings that cannot be retaken.

1. Introduction

Modifying the musical structure of existing polyphonic pieces would create new dimensions in sound processing. By splitting a recording into its source instrument signals we could fix arbitrary notes in any recordings or simply modify the melody of an instrument in a polyphonic piece.

The problem lies in the fact that although it is possible to record a musical event using many microphones, this is not common for various reasons, apart from some exceptions in pop music. Moreover, multitrack recordings are also mixed down to two channels (stereo) in most of the cases, which practically renders any attempts to modify the original tracks useless. After this step the individual notes in the recording cannot be modified, only the whole signal can be altered using different kinds of filters.

In our research we have developed a sound source separation system that allows for the separation of arbitrary note signals from the remaining part of the mixture. The musical notes of interest can be selected by the user, while the other notes remain in the mixture unaltered. This approach makes our separation system particularly applicable for fixing bad notes in existing recordings.

As reliable automatic musical transcription and instrument recognition is out of reach of today's technology, in our work we allow a reasonable amount of user input and processing time to achieve better separation quality. User input involves entering the musical score (note onset/offset, frequencies, used instruments). Due to the nature of real-life music, this input will never be 100% accurate, even if the user is presented with some kind of hint about the concrete recording (e.g. a spectrogram is plotted and shown to the user). However, it can be precise enough for getting a first rough estimate on the note parameters in the recording

There is a great need for complementary information in sound separation in addition to the raw sound signal that is being processed. The complexity of separation lies in the fact that the information we would like to retrieve from the original signal is actually not present. Significant amount of research work has been devoted to regaining the lost information in different ways.

Model based systems represent a very promising approach. In this category a parametric model of the input sources is established that serves as a set of constraints on the output signals. The model parameters are obtained from the mixture itself. The two main branches of this area are rule-based algorithms [1] that use prior information to build the model, and Bayes estimation [2] where prior information is explicitly given using probability density functions. In music applications, the most commonly used approach is sinusoidal modeling which suits the separation of pitched instruments and voiced speech very well [3].

Unsupervised learning methods usually operate on the basis of simple non-parametric models, and require less information on the original sources. They try to gather information on the source signal structures from the mixed data itself using information-theoretical principles, such as statistical independence between the sources. The most common approaches used to estimate the sources are based on independent component analysis (ICA), non-negative matrix factorization (NMF), and sparse coding. These algorithms usually factorize the spectrogram (or other short-time representation of the signal) into elementary components. This is followed by clusterization that builds the separated output channels from the elementary components.

This paper proposes a separation system that is based on the model-driven approach. A global architecture is given for the proposed separation system, while its parts are also discussed in detail. The established model, the *instrument print* is elaborated along with the Simplified Energy Split algorithm that distributes the energy of the mixture between the output channels. The separation system is capable of separating note signals sharing the same fundamental frequency which is unsolved by most of the other separation approaches.

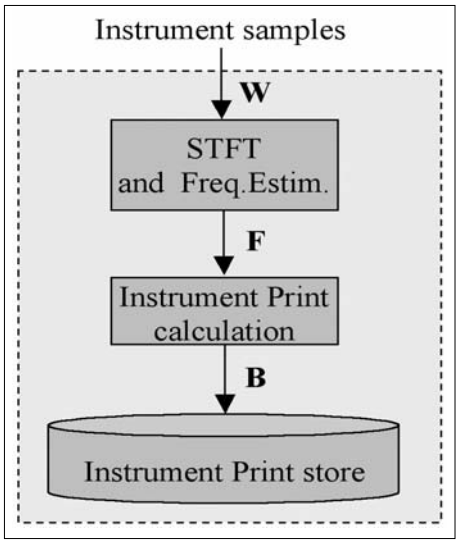


Fig. 1. Block diagram of instrument print creation

W	*Simple waveform*
F	*Frequency Estimated spectrogram:* In addition to the STFT amplitudes ($c_{k,t}$) and phases ($\varphi_{k,t}$) it contains the true frequency ($f_{k,t}^{true}$) of the respective bins.
B	*Bandogram:* A spectrogram split to subbands, in which the energy is summed. Only these sums are stored, no information amplitudes or phase information.

Table 1. Notation of the sound separation block diagram

2. Overview of the separation process

The separation system operates in frequency domain. For that reason all signals have to be transformed between time- and frequency domains at the inputs and outputs of the system.

In addition to the usual STFT we also employ the frequency estimation method proposed by Brown [7] which provides a much more precise spectral image than the standard STFT spectrogram. This method is elaborated in [8] in more detail.

The system can operate in two modes. In the first mode, the *instrument print creation mode,* the system takes sample waveforms from real-life instruments and transforms them to a representation that will later be useful for separation purposes. For this purpose we propose the instrument print model that is based on the bandogram representation of instrument sounds [8]. A bandogram is similar to a spectrogram in many respects, it can be obtained by summing the latter in certain frequency ranges. For separation purposes we need sets of instrument bandograms from each instrument playing in the musical piece to be separated.

Fig. 1 depicts the signal flow and blocks of the process, while the notations used are explained in *Table 1.* Section 3 elaborates the details of bandogram and instrument print creation.

The second operation mode of the system, the *separation mode,* is depicted in *Fig. 2.* It extracts individual note signals from the source recording using three inputs: the original music, the musical score that is entered by the user, and the instrument prints that were created in the first operation mode. The most important blocks are the Simplified Energy Splitter (SES) and the beating-correction. The task of the SES is the redistribution of the energy in the recording between the output channels. The signals created by the SES often suffer from beating. This phenomenon is already present in the original recording; however, it gets noticeable only in the separated signals. The beating correction step targets to eliminate this artifact, which is covered in Section 5.

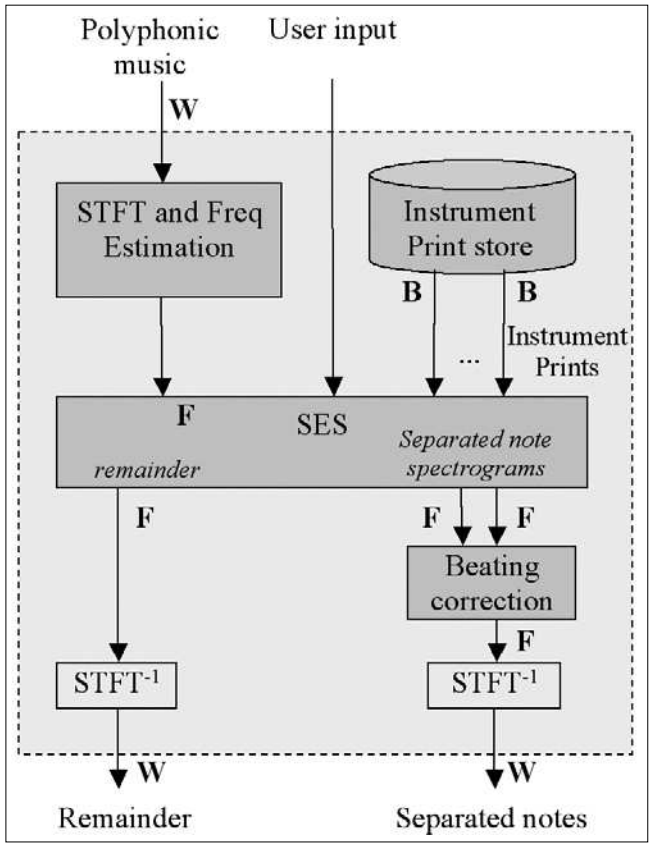
3. Instrument Prints

Even today we do not have complete knowledge about the process of human hearing. Most of the research concludes that our brain stores some kind of memory of instruments [9]. This extra information helps us recognize the melody in a complex mixture. In the process of sound separation as well we need extra information, therefore we try to copy the operation of the human brain. Our instrument model reflects this idea.

The proposed instrument model, the *instrument print,* is a set of instrument samples originating from the same instrument using different frequencies and intonations (blowing strength of the flute, hardness of the piano key hit etc.) Each print may contain one or more orthogonal intonation dimension, depending on how 'freely' a specific instrument can be played.

There may be e.g. a 'warmth' and a 'loudness' dimension for saxophone notes, the values of which range from 1 to 10. These dimensions cannot always be defined in mathematical terms; very often they can only be labeled

Fig. 2. Block diagram of the separation phase



by subjective ones, like the two above. In short, an instrument print is much like a collection of samples of different f_0 fundamental frequencies and different values in the intonation space. It can be illustrated by the following function:

$$\underline{\mathbf{A}}(\underline{\mathbf{M}}, f, f_0, t) \tag{1}$$

where $t, m_x, f_0 \in \mathrm{R}^+$,
$0 < m_x < m_{x,max}$,
$0 \le t < \infty$,
$0 < f, f_0 \le 20000 Hz$.

This function shows how amplitudes change over time (t) over the frequency range (f) for a specific note played on a certain f_0 fundamental frequency and played with intonation $\underline{\mathbf{M}}$.

In reality it is sufficient to store the sum of the amplitudes in certain frequency bands. This representation is called a *bandogram*. The subbands are aligned on a logarithmical frequency scale. A bandogram can be defined as:

$$A_{\mathbf{M},f_0,b,t} = \sum_{\rho(f_{k,t}^{true},f_0,b)} c_{k,t} \,, \tag{2}$$

where $c_{t,k}$ and $f_{t,k}^{true}$ are the amplitude and estimated true frequency of the k[th] component, $\rho(f, f0, b)$ is true if the distance of f and f_0 is exactly b subbands, where b is calculated as

$$b = \left\lfloor \log_{\sqrt[R]{2}} \frac{f_0}{f_{k,t}^{true}} \right\rfloor . \tag{3}$$

The width of the frequency bands is specified by R, that is, the number of bands per octave. In our experiments we concluded that R=12 provides good enough resolution in frequency, and it is also easy to understand as an octave consists of 12 semitones. In reality we cannot store all the possible samples of an instrument. Missing samples can be calculated by interpolation.

4. The separation problem

As the solution for the separation problem is extremely complex, if at all possible, here we propose a simplified solution that makes the separation possible at the expense of slightly lower quality. Let $\underline{\mathbf{c}}_{r\tau} = \left\{ c_{r\tau,k} \cdot e^{\gamma_{r\tau,k}} \right\}$ denote the spectrum of the recording at time $r\tau$ ($r \in \mathbf{N}$), $\underline{\mathbf{s}}_{i,r\tau}^{orig} = \left\{ s_{i,r\tau,k}^{orig} \cdot e^{\sigma_{i,r\tau,k}} \right\}$ and $\underline{\mathbf{d}}_{r\tau} = \left\{ d_{r\tau,k} \cdot e^{\delta_{r\tau,k}} \right\}$ denote the spectrum of the i[th] note and the noise component, respectively. The original separation can be formed as:

$$\underline{\mathbf{c}}_{r\tau} = \sum_{\forall i} \underline{\mathbf{s}}_{i,r\tau}^{orig} + \underline{\mathbf{d}}_{r\tau} \,, \tag{4}$$

where

$$c_{r\tau,k}, s_{i,r\tau,k}^{orig}, \sigma_{i,r\tau,k}, \gamma_{r\tau,k} \in \mathrm{R}^+ .$$

As (4) cannot be solved without any further constraints, it will be simplified in a way that the resulting quality loss is barely noticeable. Previous research [10-12] concluded that the human ear is insensitive to the phase information of sound signals as long as phase continuity is guaranteed between subsequent frames.

Based on these findings, (4) can be modified by eliminating the unknown $\sigma_{i,r\tau,k}$ and $\delta_{r\tau,k}$ phases:

$$\gamma_{r\tau,k} = \sigma_{i,r\tau,k} = \delta_{r\tau,k} \,. \tag{5}$$

thereby modifying (4) to the following form:

$$\left| c_{r\tau,k} \right| = \sum_{\forall i} \left| s_{i,r\tau,k} \right| + \left| d_{r\tau,k} \right| , \tag{6}$$

where we are looking for the values of $|s_{i,r\tau,k}|$ and $|d_{r\tau,k}|$ for each $i,r\tau$ and k, if $|c_{r\tau,k}|$ and $\gamma_{r\tau,k}$ are known.

Using the proposed simplification results in a slight quality loss: beating caused by notes that are located on close frequencies is not resolved directly. This artifact is handled by a post-processing step.

5. The Simplified Energy Splitter

This section describes the heart of the separation process, the Simplified Energy Splitter. The SES has the task of redistributing the energy in the source recording between the output channels, the separated note signals, using the instrument prints. The right prints are selected using the score, intonation and instrument information given by the user.

We propose the following iterative algorithm for the separation. We start with the spectrogram ($\underline{\mathbf{c}}$) of the original recording. Each output track will be denoted by its $\underline{\mathbf{s}}_i$ spectrogram (zero in the beginning). In each step a fraction of the amplitude content in the selected bandograms is transferred from the amplitude spectrum of the recording to the separated note signals into the right frequency band. The amount of transferred amplitude is a δ fraction of the energy in the used instrument prints. May the recording no longer contain enough amplitude, then the full remaining energy is transferred. δ can be calculated as:

$$\delta = \frac{A_{i,\mathbf{M}_i,f_{0,i},b}(r\tau - T_{onset,i})}{\sum_{\rho(f_{k,r\tau}^{true},f_0,b)} c_{[j,i],r\tau,k}} \cdot \frac{1}{J} . \tag{7}$$

Each iteration comprises I substeps, where I denotes the number of instruments in the time frame. In one substep we transfer amplitudes to one output channel only. The amplitude content of the recording in the d[th] iteration and i[th] substep is as follows:

$$c_{[j,i+1],r\tau,k} = \begin{cases} \rho(f_{k,r\tau}^{true}, f_0, b) : \max\left(0, (1-\delta) \cdot c_{[j,i],r\tau,k}\right) \\ otherwise: \quad c_{[j,i],r\tau,k} \end{cases} \tag{8}$$

The amplitude content of the i[th] note is:

$$\underline{\mathbf{s}}_{i,[j+1],r\tau} = \underline{\mathbf{s}}_{i,[j],r\tau} + (\underline{\mathbf{c}}_{[j,i-1],r\tau} - \underline{\mathbf{c}}_{[j,i],r\tau}) \,. \tag{9}$$

The reason for using an iterative algorithm is as follows. In the case of one-step amplitude transfer we may encounter cases where the right amount of amplitude is transferred to the track of a loud output note while the amplitude content of the recording decreases to zero, thereby leaving no amplitude for other notes. This case can be avoided by transferring only a fraction of

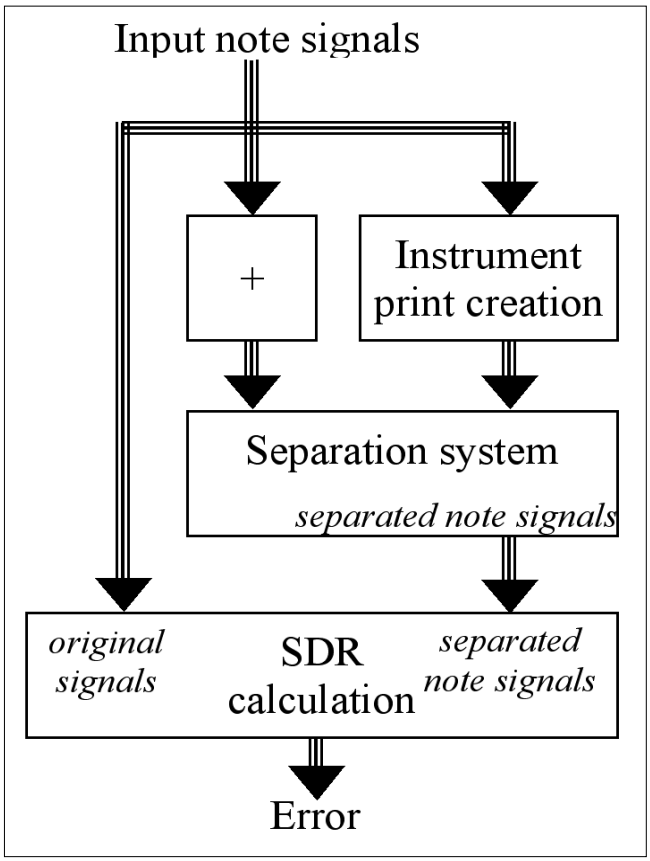


Fig. 3. The test system

the required amplitude in each step, ensuring a fair division of the energy between the outputs. The algorithm is elaborated in earlier publications [8].

Cancellations in the recording are much more audible after separation. However, in most cases this artifact can be eliminated by post processing. By comparing the separated signals to the instrument prints the cancellations can be located and amplified back to the right level.

6. Test results

The performance of the separation system was inspected using synthetic tests. Our test system is depicted in *Fig. 3*. We used the instrument sample collection of the University of Iowa [13] that contains 3841 different samples of string, woodwind, brass and keyboard notes.

In each of our tests a random set of instrument note waveforms were selected. The waveforms were converted to instrument prints. The selected samples were then mixed together and fed to the separation system as the input recording. The separated outputs were then compared to the original samples using two different measures.

The first measure is the conventional Signal-to-Distortion Ratio. The original time-domain signal is subtracted from the output, and this residual is compared to the original signal:

$$SDR_i = 10\log_{10}\frac{\sum_n \tilde{s}_i^{orig}(n)^2}{\sum_n\left[\tilde{s}_i(n)-\tilde{s}_i^{orig}(n)\right]^2}, \qquad (10)$$

where $\tilde{s}_i^{orig}$ and $\tilde{s}_i$ denote the waveform of the original and the i^{th} separated signal, respectively. The second measure operates in frequency domain using the same principle:

$$SDR_i^F = 10\log_{10}\frac{\sum_{\forall r\tau}\sum_{k=0}^{K} s_{i,k}^{orig}(r\tau)^2}{\sum_{\forall r\tau}\sum_{k=0}^{K}\left[s_{i,k}(r\tau)-s_{i,k}^{orig}(r\tau)\right]^2} \qquad (11)$$

The results are shown in *Fig. 4*. In the case of two concurrent notes the performance of the system is 15 dB which slowly degrades as the polyphony increases.

Beside the level of polyphony the other important factor influencing the separation quality is the quality of the instrument prints. Our experiments were repeated using incorrect prints sampled from the same instrument type but not of the very same instrument (e.g. using a different brand of piano). In this case the measured quality was 2 dB lower.

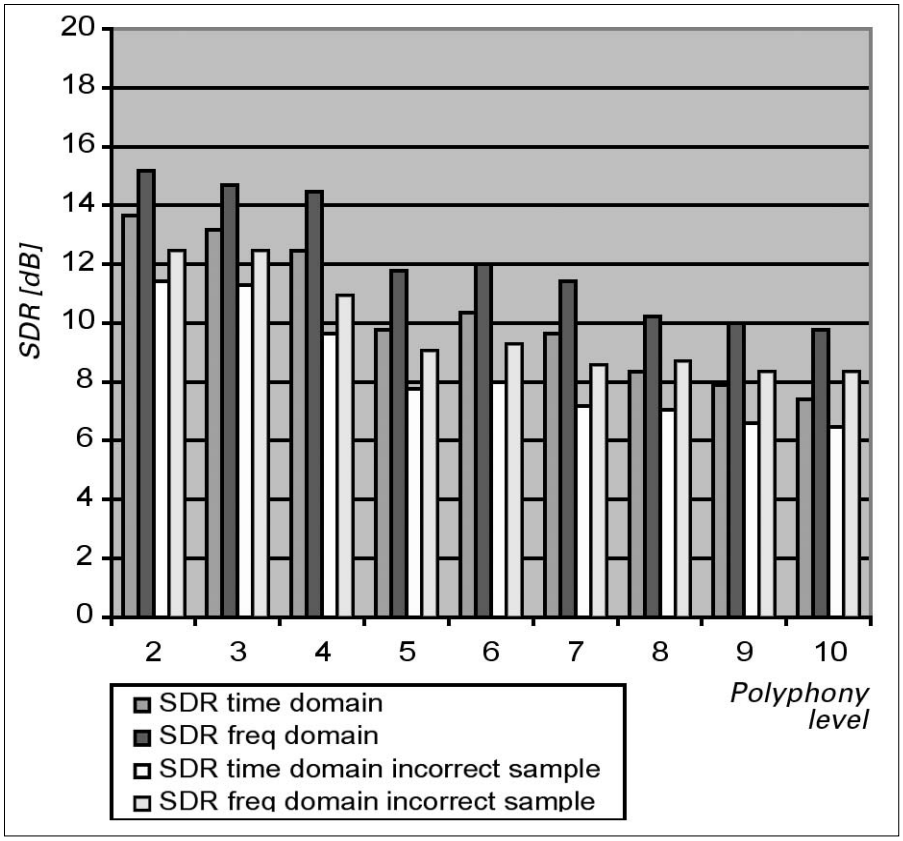


Fig. 4. Test results

7. Summary

We have developed a method for separating the signal of single instrument notes from a recording using pre-recorded instrument prints. The global system architecture of the separation process was given, along with the description of its building blocks. We have established a simple, yet powerful model for storing instrument prints, and the Simplified Energy Splitter was proposed as an algorithm for solving the energy redistribution problem.

We have demonstrated the potential of the system on synthetic and real-life test cases. Simulation experiments on generated mixtures of pitched real-life musical instruments were carried out. In these experiments we obtained an average SDR above 18 dB for two simultaneous sources, and the quality decreased gradually as the level of polyphony increased.

Example waveforms of the synthetic tests as well as real-life separation results can be downloaded from *http://avalon.aut.bme.hu/~aczelkri/separation*.

Authors

KRISTÓF ACZÉL received his degree in Technical Informatics in 2004 from the Budapest University of Technology and Economics. Currently he is a PhD student at the Department of Automation and Applied Informatics doing research in the field of analysis and manipulation of polyphonic music recordings. He is also working as a software research engineer at Nokia Siemens Networks, where he is involved in the design and development of screen, image and document sharing solutions.

ISTVÁN VAJK received the degree in Electrical Engineering in 1975 from the Budapest University of Technology and Economics, Hungary. He was a postgraduate student at the same university, where he obtained his Ph.D. degree in 1977. Since then, he has been with the Faculty of Electrical Engineering and Informatics, working at the Department of Automation and Applied Informatics. He was given the Candidate of Sciences Degree for the implementation problems of adaptive controllers in 1989 and the Doctor of Sciences Degree for the identification from noise measurements using SVD/EVD algorithms in 2007 from the Hungarian Academy of Sciences. Since 1994 he has been the head of Department of Automation and Applied Informatics. His main interest covers the theory and application of control systems, especially adaptive systems and system identification, as well as real-time software engineering.

References

[1] Every, M.R., Szymanski, J.E., "Separation of synchronous pitched notes by spectral filtering of harmonics". IEEE Transactions on Audio, Speech, and Language Processing, Vol. 14, No. 5, pp.1845–1856., 2006.

[2] Cemgil, A.T., "Bayesian Music Transcription", PhD thesis, Radboud University Nijmegen, 2004.

[3] Virtanen, T., "Sound Source Separation in Monoaural Music Signals", PhD thesis, University of Kuopio, 2006.

[4] Mitianoudis, N., Davies, M.E., "Using Beamforming in the audio source separation problem", 7th International Symposium on Signal Processing and its Applications, pp.89–92., 2003.

[5] Smaragdis, P., Brown, J.C., "Non-Negative Matrix Factorization for polyphonic music transcription", IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp.177–180., 2003.

[6] Plumbley, M., Abdallah, S., Blumensath, T., Davies, M., "Sparse representations of polyphonic music", EURASIP Signal Processing Journal, Vol. 86, No. 3, pp.417–431., 2006.

[7] Brown, J.C., Puckett, M.S., "A high resolution fundamental frequency determination based on phase changes of the Fourier Transform", J. of the Acoustical Society of Am., Vol. 94, No. 2, pp.662–667., 1993.

[8] Aczél, K., Vajk, I., "Note separation of polyphonic music by energy split", WSEAS International Conference on Signal Processing, Robotics and Automation, pp.208–214., 2008.

[9] McAdams, S., "Recognition of Auditory Sound Sources and Events. Thinking in Sound: The Cognitive Psychology of Human Audition", Oxford University Press, 1993.

[10] Zwicker, E., Flottorp, G., Stevens, S.S., "Critical band width in loudness summation", J. of the Acoustical Society of Am., Vol. 29, pp.548–557, 1957.

[11] Smith, S.W., The Scientist and Engineer's Guide to Digital Signal Processing, California Technical Publishing, 1997.

[12] Edler, B., Purnhagen, H., "Parametric Audio Coding", IEEE International Conference on Communication Technology, Vol. 1, pp.614–617., 2000.

[13] The University of Iowa, Musical Instrument Samples Database (2008.07.07), http://theremin.music.uiowa.edu

Home access network model specifications

IZABELA KRBILOVÁ, VLADIMÍR HOTTMAR, BOHUMIL ADAMEC

Department of Control and Information Systems, Department of Telecommunications and Multimedia, University of Žilina, Slovakia

{krbilova, hottmar, adamec}@uniza.sk

Keywords: residential gateway, stated equations, time-dependent probabilities, particular request, service time, peripherals

The paper investigates a home network configuration consisting of residential gateway RG and a number of intelligent peripheral devices capable of autonomous activity. A queuing model is built by means of bulk service in a closed circuit which circulates constant number of requests. Performance and time characteristics of peripherals communicating with residential gateway are determined. The presented results illustrate mutual dependence of the number of network peripherals and time characteristics determining operation of the network.

1. Introduction

Modern broadband *home networks* are expected to provide all new integrated multimedia services with added value (video on demand, latest news on demand, teleshopping, distance learning) along with securing, controlling and automatization of the household. The development of the digital households in the past was hindered by the lack of modern broadband technologies for access and home networks.

Nowadays, however, the innovations of broadband access technologies and the considerable investments in access networks infrastructure have eliminated the restrictions. In spite of the fact that advanced wireless and wireline technologies designed for home networks show clearly that all major technical problems concerning implementation of the networks have been overcome, distribution of integrated multimedia services among large numbers of users is still limited mostly due to separation of home networks. Residential gateway is an essential element of a modern home network. It is the access and concentration point which switches the functions for telecommunication and general data traffic, distribution of entertainment services for homes and controlling and management of various electric and electronic devices.

For the purposes of this article we will assume a home network configuration consisting of residential gateway RG and *n* intelligent peripheral devices PD capable of *autonomous activity*. *Fig. 1* shows one of the possible variants of home access network utilizing current technologies [8,9,19,11].

Fig. 2 shows communications among some peripheral devices within the home network. It is apparent that the busiest and thus crucial node is residential gateway.

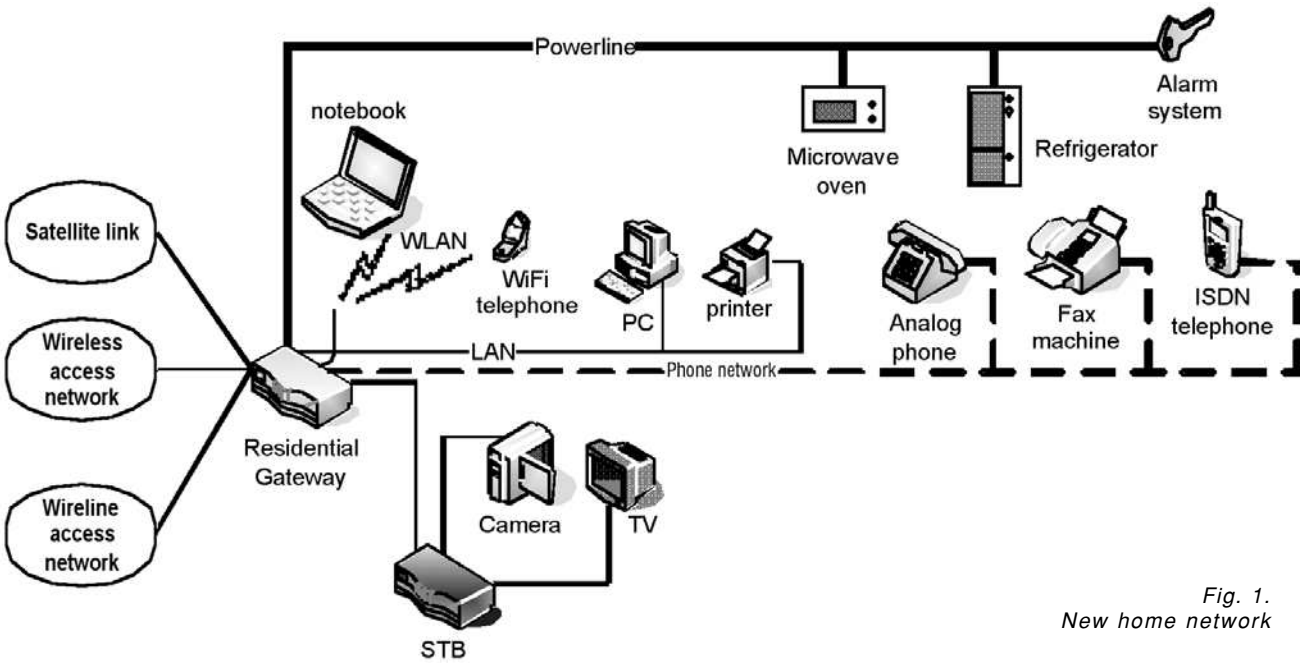


Fig. 1. New home network

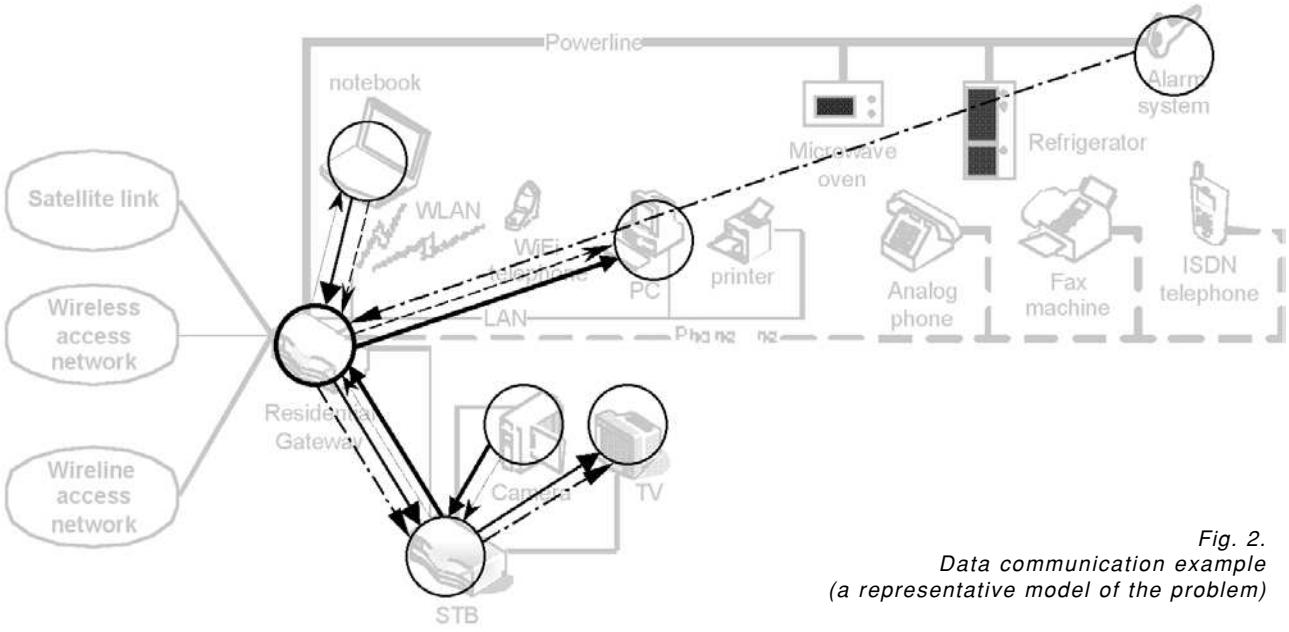


Fig. 2.
Data communication example
(a representative model of the problem)

Let peripheral devices P_1 *to* P_n communicate with residential gateway based on the principle of requests and responses. Unless the gateway is busy communicating with one of the peripherals or with its own activity, it is able to receive a demand and respond to it. If one of the peripherals requests communication with residential gateway when it is busy, it will transmit the demand and it will be ordered in queue. If one of the peripherals P_1 *to* P_n finished its communication with residential gateway, it will either perform its autonomous activity (being out of the system) and thus it becomes a potential source of demands or it repeatedly requests communication with the residential gateway and enters the system and queues up in case the residential gateway is busy.

These peripherals will be recognized as out of system peripherals. The sequence of peripherals requesting communication will result from the sequence they entered the system. There is no priority scheme, the first-in first-out rule can be used. As soon as the residential gateway is free, it will respond to the first peripheral in the queue. The described model clearly shows the interdependence of residential gateway load and the number of requested operations, where the number of the requests is equal to the number of peripherals. The number of the peripherals in and out of the system is *n*, so the system is finite and it is a closed unit. In the next section we will analyze the suggested problem and calculate the quantities and time characteristics of the system based on the model of bulk service [1,2,6,7].

2. Quantification and modelling of the system

According to the above discussion we will analyse characteristics and calculate parameters of a closed system [2,4,5]. For this purpose we will introduce the following assumptions:

- One of the first things we have to deal with is to define the flow of requests entering the system. We assume that the returns of the requests into the system correspond to Poisson elementary flow of requests with exponential distribution of their arrival intervals. General distribution is assumed for a given service interval of the residential gateway.
- Let P_1 *to* P_n be the requests demanding communication from the residential gateway. *N* requests hence circulate in the system. Let λ be the parameter of a random variable with exponential division. $1/\lambda$ is hence average interval of residential gateway response to the request (e.g. the interval of the response transferred to the peripheral P_i) until the request (of peripheral P_i) returns back to the system.
- Let the service time of the residential gateway *t* be a continuous random variable with mean value τ and general distribution. If the density of service time probability is *g(t)*, the mean value of the service time can be calculated by the relation:

$$\tau = \frac{1}{\mu} = \int_0^\infty t \cdot g(t)dt\,, \qquad (1)$$

where μ is the mean value of service time.

Communication regime between residential gateway and peripherals defined in this way will be modelled through the system of a bulk service and in compliance with Kendall's designation we will define it as *closed QS with constant number of requests M/G/1/FIFO*. As already stated, according to the theory of bulk service, individual peripherals present requests communicating with the system.

Operational time P_i of each peripheral in the network consists of three phases that change [2,3]:

a) time interval of the request outside the system;

b) waiting time of the request – peripheral P_i in the service system requesting a response from the residential gateway;
c) service time of the residential gateway.

In the following we will examine circulation of the requests in the system. Let N be the average number of returns of a request into the system in a time interval. Let W be the average waiting time of a request in the queue. Then the equation

$$N\left(1/\lambda + W + \tau\right) = 1 \qquad (2)$$

denotes the mean value of service duration supposing a request by mean values [1,2,3]. In order to determine variables N and W in the relation 2) it is necessary to define times of relieving and occupying the gateway. Work time of the residential gateway involves two alternating intervals; occupying the residential gateway and relieving the gateway. As the system contains n requests, mean value of occupying the gateway is product $nN\tau$. We will now focus on the average time of relieving the gateway. Let p_k be the conditional probability of the fact that only one request will enter the system assuming that the system contains k requests. Average number of intervals when the gateway is unoccupied equals the product nNp_0, where p_0 is the probability of empty system.

If all the requests (peripherals P_1 *to* P_n) are out of the system at the moment, the probability of the assumption that after a time interval t all the requests will remain out of the system is $e^{-\lambda nt}$ and the probability of the assumption that a request will enter the system in a time interval $(t+\Delta t)$ je $\lambda n\Delta t + o(\Delta t)$ where (Δt) is a function converging to zero faster than linear [1-3].

The probability that the interval of not occupying the residential gateway will finish between $o(\Delta t)$ is $e^{-\lambda nt}\lambda n\Delta t + o(\Delta t)$ and average interval time of unoccupied gateway will be:

$$\int_0^\infty te^{-\lambda nt}\lambda n dt = \frac{1}{\lambda n}. \qquad (3)$$

The sum of all intervals of unoccupied gateway is given by the relation

$$nNp_0\frac{1}{\lambda . n} = N\frac{p_0}{\lambda}. \qquad (4)$$

Considering the relation (4) and the given average time of occupying the residential gateway ($nN\tau$) we get

$$N\left[n\tau + \frac{p_0}{\lambda}\right] = 1. \qquad (5)$$

Applying the relations (1) and (5) for average waiting time of a request in the queue W we obtain

$$W = (n-1)\tau - \frac{1-p_0}{\lambda}. \qquad (6)$$

In the following section we will examine a condition of the system where service time of the gateway will be interrupted by other request(s) entering the system. Let us consider two time instants t_1 and t_2. t_1 is the time when the request left the system. We assume that at the time t_1 the system still contains r requests and t_2 represents departure time of another request from the system and hence the system contains $r-1$ requests.

Time interval $t_2 - t_1$ is the service time of a request if $r > 0$. If $r = 0$, the system is empty after the first request left the system. Then the time interval $t_2 - t_1$ equals the sum of two time intervals, namely the time interval starting in t_1 until another request arrives and the time interval equal to service time of the second request. Other requests may enter the system only during service time of the second request due to the fact that the system remained empty between t_1 and the arrival of another request.

Let the service time of a request be t. If we consider Poisson flow of request occurrence in the system, the probability of the occurrence of j requests in the system during the service time t of the particular request:

$$v_j(t) = \frac{(\lambda t)^j}{j!}e^{-\lambda t} \quad \text{for } j = 1, 2, 3, ... \qquad (7)$$

Then we assume that service time distribution is determined by its probability density $g(t)$. The probability that j requests will enter the system during the service time of a particular request will be:

$$\beta_j = \int_0^\infty v_j(t)g(t)dt. \qquad (8)$$

Equation (8) quantifies the probability of j requests entering the system while an other request is being processed/serviced but does not reflect the change of the probability if the number of requests is finite where requests number limits outside the system are $j \in \langle 1; n-1 \rangle$.

We will now analyze processes that may occur in the system during the interval $t_2 - t_1$. Let p_k be the probability of the transition of the system from condition k to condition $k+1$. The probability also holds true in the reversed order.

This assumption results from the fact that the number of transitions from condition k to $k+1$ must equal the number of transitions from condition $k+1$ to k. In order to determine the probability, we must define the probability $\beta_{k,j}$ which represents the number of requests j occurring in the system during the service time when it contained k requests.

Following equations will provide the desired data:

$$p_0(t_2) = p_0(t_1)\beta_{1,0} + p_1(t_1)\beta_{1,0}$$

$$p_1(t_2) = p_0(t_1)\beta_{1,1} + p_1(t_1)\beta_{1,1} + p_2(t_1)\beta_{2,0} \qquad (9),(10),(11)$$

$$p_2(t_2) = p_0(t_1)\beta_{1,2} + p_1(t_1)\beta_{1,2} + p_2(t_1)\beta_{2,1} + p_3(t_1)\beta_{3,0}$$

Analogically, equation for k requests remaining in the system after time interval t_2 can be constructed. Then

$$p_k(t_2) = p_0(t_1)\beta_{1,k} + p_1(t_1)\beta_{1,k} + p_2(t_1)\beta_{2,(k-1)} + \\ ,.... + p_k(t_1)\beta_{k,1} + p_{(k+1)}\beta_{(k+1),0} \quad (12)$$

for $k = 1, 2, 3, ..., n-2$.

In compliance with the above stated equations we will suggest normalizing condition for the sum of probability

$$\sum_{k=0}^{k=n-1} p_k = 1. \quad (13)$$

For permanent regime for time-dependent probabilities the following limit can be accepted according to [2,3]:

$$p_k = \lim_{t\to\infty} p_k(t). \quad (14)$$

The equations (9) to (13) will then be arranged as follows

$$p_0 = (p_0 + p_1)\beta_{1,0}. \quad (15)$$

For $k=1, 2, ..., n-2$ we will determine the probability p_k that the system contains k requests in the relation

$$p_k = (p_0 + p_1)\beta_{1,k} + p_2\beta_{2,k-1} + ... \\ + p_k\beta_{k,1} + p_{k+1}\beta_{k+1,0}, \quad (16)$$

where $k=1, 2, ..., n-2$.

If the service time is t and at the beginning of the service the system contains k requests, the probability that out of total number of requests $(n-k)$ out of system j requests will be returned into the system and $(n-k-j)$ will not be returned is

$$\binom{n-k}{j}\left(1-e^{-\lambda t}\right)^j\left(e^{-\lambda t}\right)^{n-k-j}. \quad (17)$$

As density of service time probability is $g(t)$, for $\beta_{k,j}$ we will obtain

$$\beta_{k,j} = \binom{n-k}{j}\int_0^\infty \left(1-e^{-\lambda t}\right)^j\left(e^{-\lambda t}\right)^{n-k-j} g(t)dt. \quad (18)$$

Relation (18) is applicable for the situation when service time is a random variable but continuous and has density $g(t)$. This enables us to count the probability $\beta_{k,j}$. Applying the equation (16) and standardizing condition (13) we will obtain an equation system which will enable us to determine p_0, i.e. when the system is empty. Employing relation (6) and the probability p_0 help us determine average waiting time of a request in a queue W. Average cycle C_y of each peripheral given by mean values consists of the following time intervals

$$C_y = 1/\lambda + W + \tau. \quad (19)$$

Average number of returns of the peripheral into the system can be calculated by

$$N = 1/C_y. \quad (20)$$

3. Conclusion

Modelling of a home access network by means of bulk service in a closed circuit which circulates constant number of requests constitutes a framework enabling us to determine performance and time characteristics of peripherals communicating with residential gateway regardless of technical and program equipment.

The presented results illustrate mutual dependence of the number of network peripherals and time characteristics determining operation of the network. Any changes in hardware or software structure of peripherals or residential gateway will result in changes of response time characteristics of the model.

Acknowledgment

This project was supported by the VEGA grant No. 1/0375/08 of Ministry of Education of the Slovak Republic.

Authors

IZABELA KRBILOVÁ graduated in 1967 with the Master's degree in Safety and Communication Engineering at the University of Transport and Communications in Žilina. Since 1969 she was employed at the Department of Information and Safety Systems. In 1979 she accomplished her dissertation thesis in the field of "Line Wire Communications Engineering". Since 1984 she is working as an associate professor at the Department of Control and Information Systems at the Faculty of Electrical Engineering, University of Žilina. Specialisation: application of queuing theory in communication systems. Reliability and diagnostics of complex systems.

VLADIMIR HOTTMAR graduated from the University of Žilina in 1975 with the Master's degree in Transport and Communications. At the beginning of his career he was employed at the Research Institute of Computers Techniques in Žilina, where he worked as a researcher in the Department of Microcomputers and Development Systems for 22 years, later being responsible for running the Research Department. Currently he is employed at the University of Žilina in the Department of Telecommunications in the position of an associate professor and is also involved in some research work. He is a project leader of the project VEGA. In 1999 he accomplished his PhD studies at the University of Žilina. Since May 2006 he has worked on the post of an associate professor at the University of Žilina.

BOHUMIL ADAMEC graduated from the University of Žilina in 2007 with the Master's degree in Telecommunications. Currently he is a PhD student at the University of Žilina in the Department of Telecommunications and Multimedia. His range of interest is home networking especially simulation and mathematical modeling of modern multimedia home networks.

References

[1] Mitrani, I.,
Modelling of computer and communication systems.
Cambridge University Press, 1987,
pp.73–74.

[2] Neuschl, Š. et al.,
Modelling and simulation.
SNTL, Praha, 1989, pp.304–354.
[3] Robertazzi, T. G.,
Computer Networks and Systems:
Queuening Theory and Performance Evalution.
Springer-Verlag Inc., New York, USA, 1990.
[4] Hottmar, V.,
Network model of the processor system,
Híradástechnika (Infocommunications Journal),
Selected Papers, Vol. LX, No. 12, 2005, Hungary.
[5] Peško, Š, Smieško, J.,
Stochastic models of operational analysis,
Published by the University of Žilina, 1999,
ISBN 80-7100-570-3.
[6] Hottmar, V.,
Model of a computational system.
Scientific studies of the ŽU, Electrotechn. Series 25,
1999, pp.11–21.,
ISBN 80-7100-716-1, ISBN 80-7100-913-X.
[7] Kalas, J.,
Markov's chains, published by
the Comenius University in Bratislava, 1993.
ISBN 80-223-0560-X.
[8] Zahariadis, T.,
Home Networking Technologies and Standards.
Artech House, London, 2003.
[9] Ungar, S.,
System and Architectural Requirements for
a Broadband Residential Gateway.
Request for Information, Bell Comm. Research,
July 24, 1997, pp.33–37.
[10] Lawrence, V.,
Digital Gateways for Multimedia Home Networks.
Telecommunication Systems Journal, August 2003.
[11] Zahariadis, T., Pramataris, K., Zervos, N.,
A Comparison of Competing Broadband
In Home Technologies.
IEE Electronics and Comm. Engineering J. (ECEJ),
August 2002, pp.133–142.
[12] Galko, M., Krbilová, I., Vestenický, P.,
Blocking Probability Influence on the Single-Channel
Service System Operation with Various Input Flows.
In TRANSCOM'97, Vol. 2, Žilina, 1997, pp.37–40.,
ISBN 80-7100-416-2.