We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Cross–Layer Design in Wireless Ad Hoc Networks with Multiple Antennas

Ehsan Soleimani-Nasab, Mehrdad Ardebilipour and Mahdi Kashiha
K.N. Toosi University of Technology
Iran

An ad hoc wireless network is a collection of wireless nodes that self-configure to form a network without the aid of any established infrastructure. Some or possibly all of these nodes are mobile. These networks are extremely compelling for applications where a communications infrastructure is too expensive to deploy, cannot be deployed quickly, or is simply not feasible. There are numerous potential applications for ad- hoc wireless networks, ranging from multi-hop wireless broadband Internet access, to sensor networks, to building or highway automation, to voice and video communication for disaster areas.

The lack of established infrastructure, the network and channel dynamics, and the nature of the wireless medium offer an unprecedented set of challenges in supporting demanding applications over ad hoc wireless networks. The wireless channel is inherently a broadcast medium, so transmissions from different nodes interfere with each other. The quality of wireless links vary over time and space due to interference, multipath fading, and shadowing refer to (setton et al., 2005).

Ad hoc networks are harder to design than wired networks because of problems that arise from the every nature of wireless communication. One of these problems, namely the hidden terminal that makes collision. To avoid collisions, a collision avoidance method could be used, as in the well known IEEE 802.11 DCF, which recommends the use of a bidirectional signaling flow made of Request-To-Send (RTS) and Clear-To-Send (CTS) frames before packet transmission, closing the data exchange with an Acknowledgement packet. This scheme forces other nodes in the proximity of the sender and the receiver to defer their own transmissions while a data exchange is in progress, even if they sense a free channel.

Fig. 1. shows a typical scenario of "hidden-terminal". Suppose that the B station is in the range of transmission of both A and C, but A and C do not feel another, and suppose that A is transmitting to B. According to the DCF protocol, if C has a packet to be sent to B, listens to the channel and it senses free, because it can not hear the transmission of A. Then begins to transmit, causing a collision at node B. Many authors are working to solve the hidden terminal problem.

In (Choudhury et al., 2006), the authors focused on purely directional transmission and designed multi-hop MAC (MMAC), a routing-aware protocol that bridges longer distances by both coordinating farther nodes using RTS/CTS exchanges over multiple hops and exploiting the higher gain and lower overall interference achieved by directional communications. In (Gatsis et al., 2010) the authors dealt with optimal cross-layer design for wireless ad hoc networks.

A networking-based approach is carried out in (Park et al., 2005) with MIMA-MAC, an access protocol specifically designed for ad hoc networks with up to two antennas per node. The devised MAC includes a contention-based and a contention-free period, used to set up links among receivers using two antennas to decode data coming from up to two transmitters using one antenna each. The small number of nodes considered and the constraint to use at most one antenna for transmission represent significant limitations.

In (Ramantan et al., 2005) authors proposed a set of integrated MAC, routing, neighbor discovery and signaling protocols for directional ad-hoc networks. In (Chen et al., 2006) authors proposed an access scheme to exploit multi packet reception with CDMA while meeting QOS requirement. Zhang and Lee (2008) toke an information-theoretic approach by defining throughput as the maximum mutual information between a received and a transmitted signal. They analyzed 802.11 multi user detection in one hop scenario.

In (Sundaresan et al., 2004), a centralized controller is able to estimate concurrent resource usage and to schedule links to exploit the benefits of MIMO such as Spatial Multiplexing (SM) and interference suppression refer to (paulraj et al., 2004), along with increased transmit rate. The final objective is a proportional fair scheduling of transmissions, the accounts for bottleneck links, and is achieved by graph coloring. An online algorithm is also designed. This last contribution, although interesting, makes some very strong assumptions on the PHY layer, e.g., that any transmission uses the full channel capacity and that signaling at the MAC level is perfect. Also, in (Hu and Zhang, 2004), some new ideas have suggested.

Unlike wireless ad hoc networks, wireless sensor networks, instead, are tiny objects that face a lot of constraints from the point of view of PHY capabilities, processing and memory resources, and most of all available battery energy. They are often designed for long-time operations, and thus require a careful design that grasps as many performance improvements as possible. In this field, a good cross–layer design could provide the ultimate resource for increasing lifetime without performance loss refer to (Madan et al., 2006).

From the MAC point of view, above works rely on the exchange of signaling messages among separate communications. Unlike IEEE 802.11 standard that uses in ad hoc networks we want the MAC to coordinate transmissions in order to favor parallel communications, while avoiding channel overload. Also, we want to drive the reception of SM signals so that wanted ones are sufficiently protected from interference, using a mechanism to prevent some nodes from transmitting if needed. In order to do this, we let the MAC use the knowledge of ongoing neighboring handshakes to decide whether or not to grant some requested transmissions, so that the interference cancellation capabilities of the MIMO receiver are properly exploited without its being overloaded refer to (Zorzi et al., 2006).

With cross layer design, physical layer at the symbol level and framed MAC layer on top of it lead to decrease in the error and increase in the network throughput. We use MIMO technique to improve MAC in ad hoc networks. MIMO techniques allow exploiting the presence of multiple antennas to improve transmission bit rate through spatial multiplexing or to improve the signal decoding efficiency through diversity reception and interference cancellation. In this study, we provide some framework and results on the reception performance of MIMO link in a multiuser scenario. The results show that the capture capability introduced by MIMO technology is significant and this should be taken into account when designing MAC protocols. In this study, we start the analysis of the performance of the PHY layer in a multiuser context and derive the implication that this

PHY layer would be on the design of higher layer protocols. We continue by designing a MAC layer that makes use of back-and-forth information exchanges with the PHY layer in order to perform multiuser detection. The whole process is driven in order to guarantee a satisfactory throughput and yet protect the wanted signals through active interference detection and cancellation. Results show that our offered protocol provides a large throughput improvement. This is due to the higher number of packets delivered to their final destination.
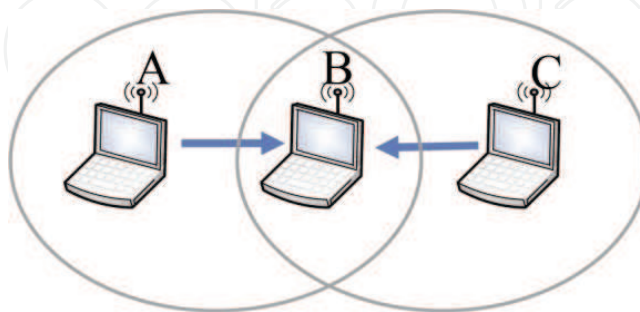
Fig. 1. Hidden terminal problem.

2. Physical layer model

2.1 Problem formulation

As a general line, consider that nodes with multiple antennas are arranged in the network where transmission takes place using packet radio communications. Transmitting nodes build streams of bits and (if necessary) encode them to combat channel impairments. Each user may select the number of antennas to use for transmission that is best suited to its needs. We make the assumption that, in each packet, the number of bits to be sent per transmitting antenna is constant for all users. Each time a multiple transmission has to be decoded, the receiver knows in advance the number of symbols to be simultaneously processed, along with the transmission duration of each of the incoming streams.

At the receiver, multiuser decoding is performed symbol-by-symbol, with a de-correlating layered space-time signal processing technique refer to (Sfar et al., 2003). The receiver is listening to the signals coming from K different users, $l = 1,...,K$, each using u_l antennas, and thus has to decode a total of $U = \sum_{l=1}^{K} u_l$ incoming symbol per time interval. Let $b = [b_1,...,b_U]^T$ denote the U - length symbol vector where each element is a symbol coming from one of the U transmitting antennas and superscript T denotes transposition. Let S be a matrix with columns containing spreading sequences, one column for each stream. Signals pass through the fading channel that we assume to be frequency non-selective, represented by the channel matrix $H = [h_1,...,h_P]$, where h_p is $1 \times K$ channel coefficient vector between the p-th receiver antenna and all K users. The received signal at antenna p can be written as:

$$r_p = SC_p b + n_p \tag{1}$$

where C_p denotes the complex diagonal channel matrix for the p-th antenna, $diag(h_a)$. The noise vector n_p is a complex valued zero mean Gaussian random N-vector with a covariance matrix $\sigma^2 I_N$, in which I_N denotes the $N \times N$ identity matrix, where N is length of spreading code for each user.

After the space code match filtering, we obtain the sufficient statistics vector Y_{MU} as:

$$Y_{MU} = \sum_{p=1}^{P} X_p^H r_p = R_{MU} b + n \qquad (2)$$

Where $R_{MU} = \sum_{p=1}^{P} X_p^H X_p$ is the $U \times U$ space cross-correlation matrix, with $X_p = SC_p, n = \sum_{p=1}^{P} X_p^H n_p$, H denotes the complex transpose operator. The receiving node may decide to estimate the channel for only a subset of the transmitting users, limiting the stream detection and cancellation to this subset. Thus, the sufficient statistics vector in (2) becomes a sum of two contributions, the first coming from decoded signals, and the other representing a interference term, namely

$$Y_{MU} = \sum_{p=1}^{P} X_p^H (r_p + X_p^{\text{int}} b_{\text{int}}) = R_{MU} b + n + I \qquad (3)$$

Where $I = \sum_{p=1}^{P} X_p^H X_p^{\text{int}} b_{\text{int}}$ is the space filtered interfering signal, involving the interference symbol b_{int} and the channel matrix towards interfering users C_{int} which receiver need not know. We report in Fig. 2. a flowchart description of detection algorithm
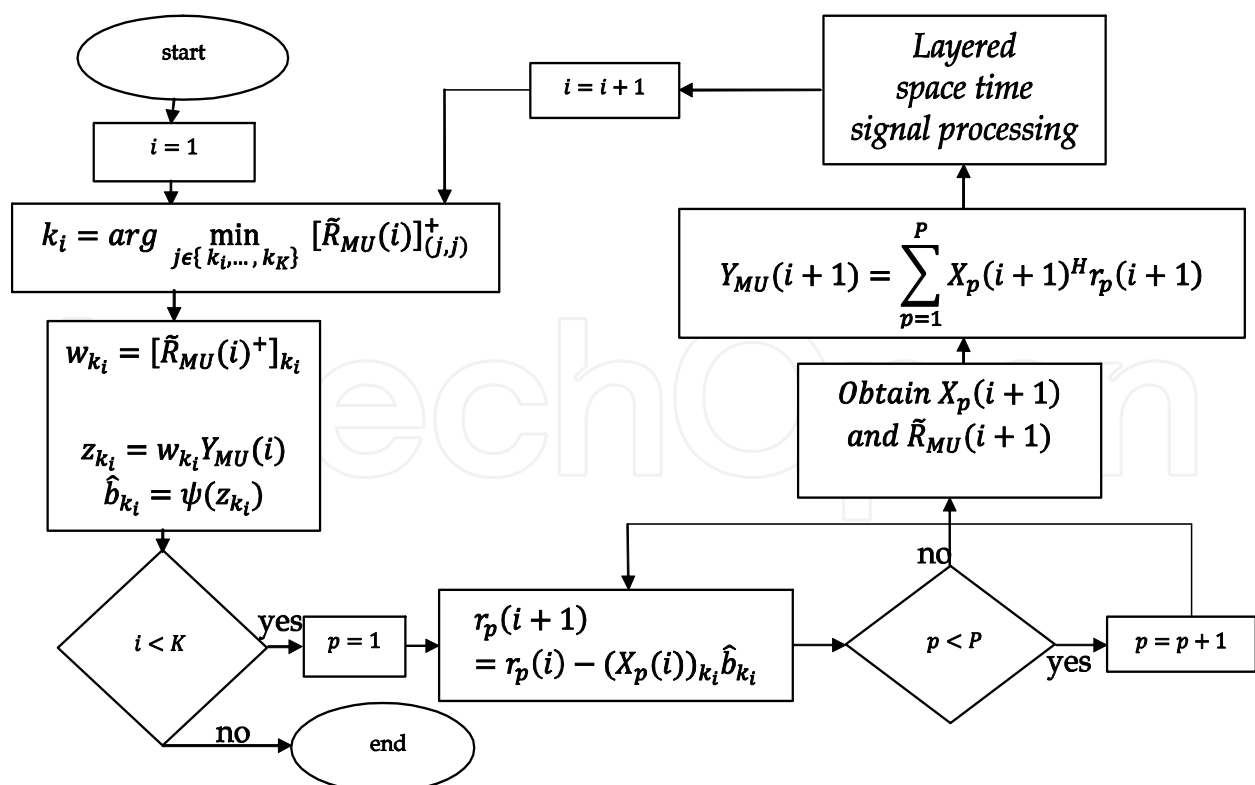

Fig. 2. Flowchart description of LAyered Space Time Multi User Detection (LAST-MUD) algorithm

The detection algorithm works on a single symbol each time and consists of U iterations. It implies pseudo-inverse calculus over R_{MU} to get $R_{MU}{}^+$ and reordering the received symbols according to their post–detection SNRs (including effects deriving from propagation from different distances). Iteration by iteration, the symbol with the maximum SNR is chosen and isolated from spatially multiplexed signals by linearly weighing the sufficient statistic vector Y_{MU} with a set of coefficients extracted from R_{MU}. The scalar value obtained by this process is fed into a decision block to yield the estimate of the transmitted symbol, and then the sufficient statistics vector Y_{MU} is updated by cancellation of the resulting estimate by striking out the $k_i - th$ column of $X_p(i)$ and the $k_i - th$ row and column of $R_{MU}(i)$. Iterative selection, decoding, and cancellation continue until all U symbols are extracted.

2.2 Physical layer simulations & results

In Fig. 3. we report a graph of bit error rate for all combinations of 4,10,14,16 and 22 users with one antenna each and a receiver with 6 and 8 antennas. The figure contains a performance comparison of BPSK modulation that also processed by taking the real part of R_{MU} during the decoding phase along with a QPSK modulation.

$$Y_{rT} = real(Y_{Tr}) = real(R_{Tr}b + n_{Tr}) = R_{rT}b + n_{rT} \qquad (4)$$
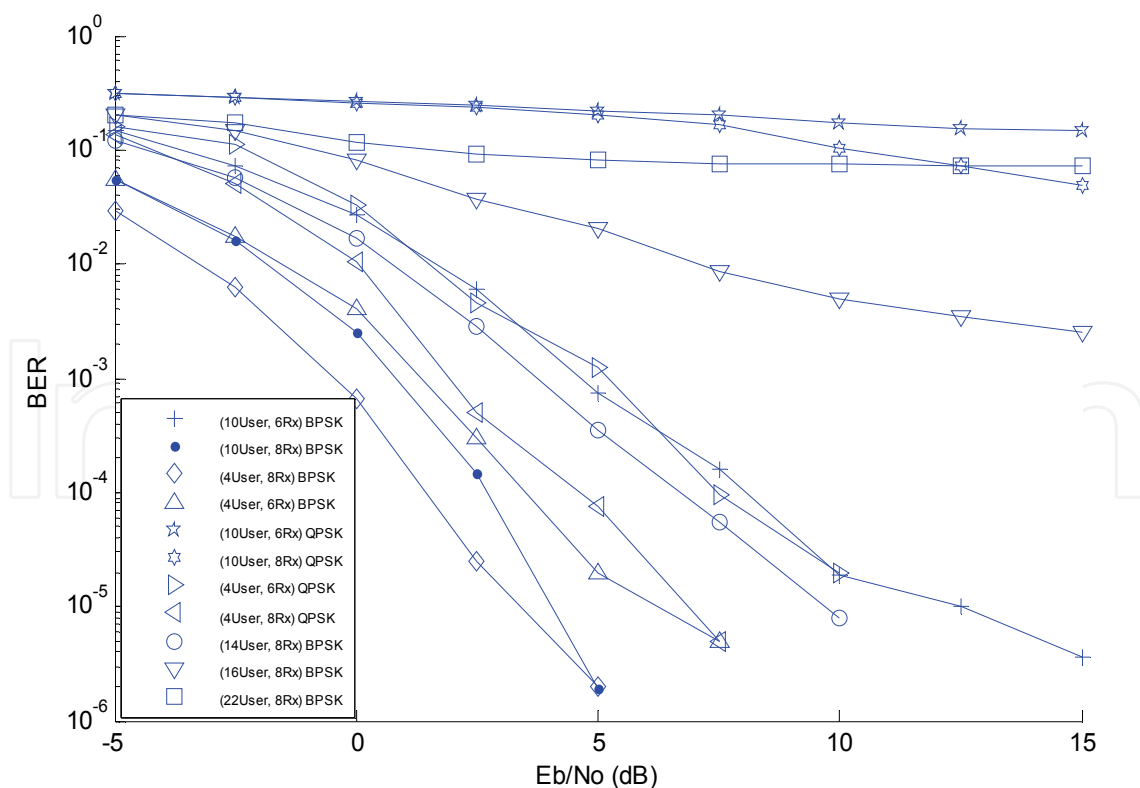

Fig. 3. Comparison of BERs as a function of SNR per receiver

In Fig. 3. and (4), when Y_{MU} is a real value we show our signal by rT and when Y_{MU} is a complex value we show our signal by Tr. As can be inferred from Fig. 3. rT-BPSK gives better results. This is not only a consequence of the constellation simplicity, but also of the fact that it is real. R_{MU}'s imaginary part brings into the detection process a further uncertainty element, namely the noise affecting the imaginary part that may impair the decision over symbols in a way that is unpredictable, due to the nonlinearity of the cancellation process. Fig. 3. also suggests that the loss in spectral efficiency due to the use of BPSK is easily recovered by the higher decoding performance of the system. For instance, with 14 incoming streams the BER for BPSK falls bellow 10^{-5} for 10dB SNR. Note in a more realistic ad hoc network scenario, where the nodes are randomly placed in the area of network, different average received powers would lead to even better performances.

In simulation, model is supposed Hata path loss model which states that a signal which propagates in distance d is multiplied by the constant A, $\frac{A}{d^{\beta}}$, where $A = 0.001$ and $\beta = 4$. Transmission rate and length of signaling packet are 7.5 Mb/sec and 25 byte respectively refer to (Soleimani-Nasab et al., 2009).

Fig. 4. shows signal interference plus noise ratio (SINR) per user for receiving stream by varying the distance, according to the number of antennas are used for transmission, interfered with 8 users are placed at 300m.

Fig. 5. is reproduced probability of errors in the RTS packets which were sent from 1 to 24 transmitters are placed at 100m and 8 interferers are placed at 300m which are sending packets simultaneously.
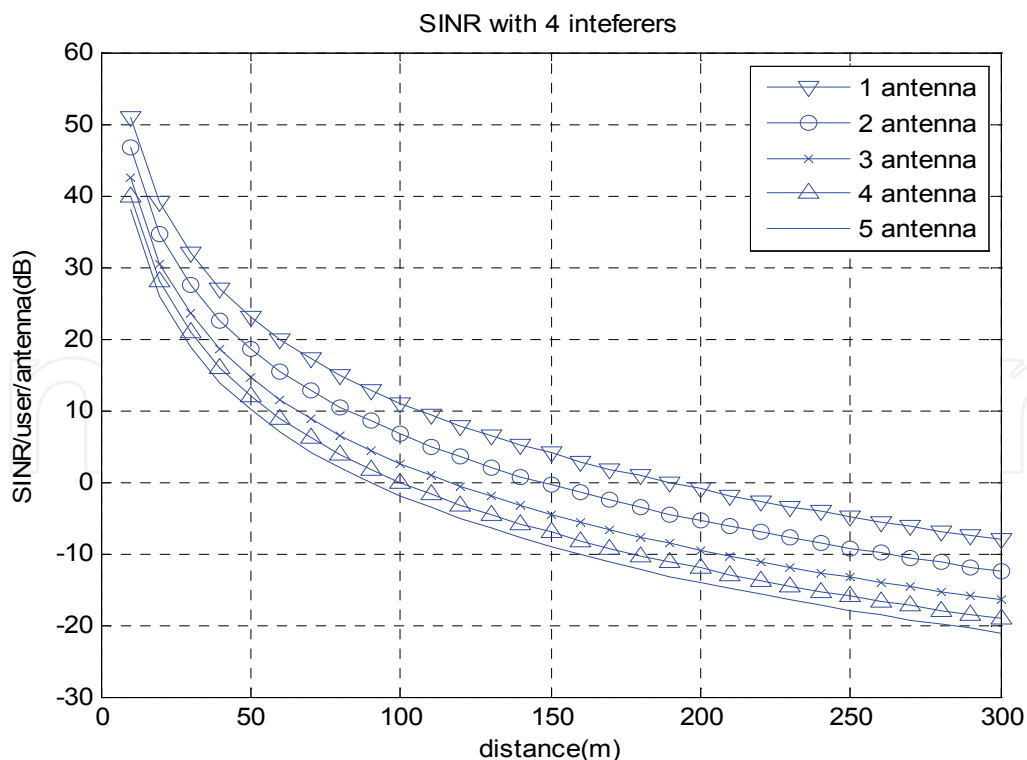

Fig. 4. SINR per user for receiving antenna by varying the distance
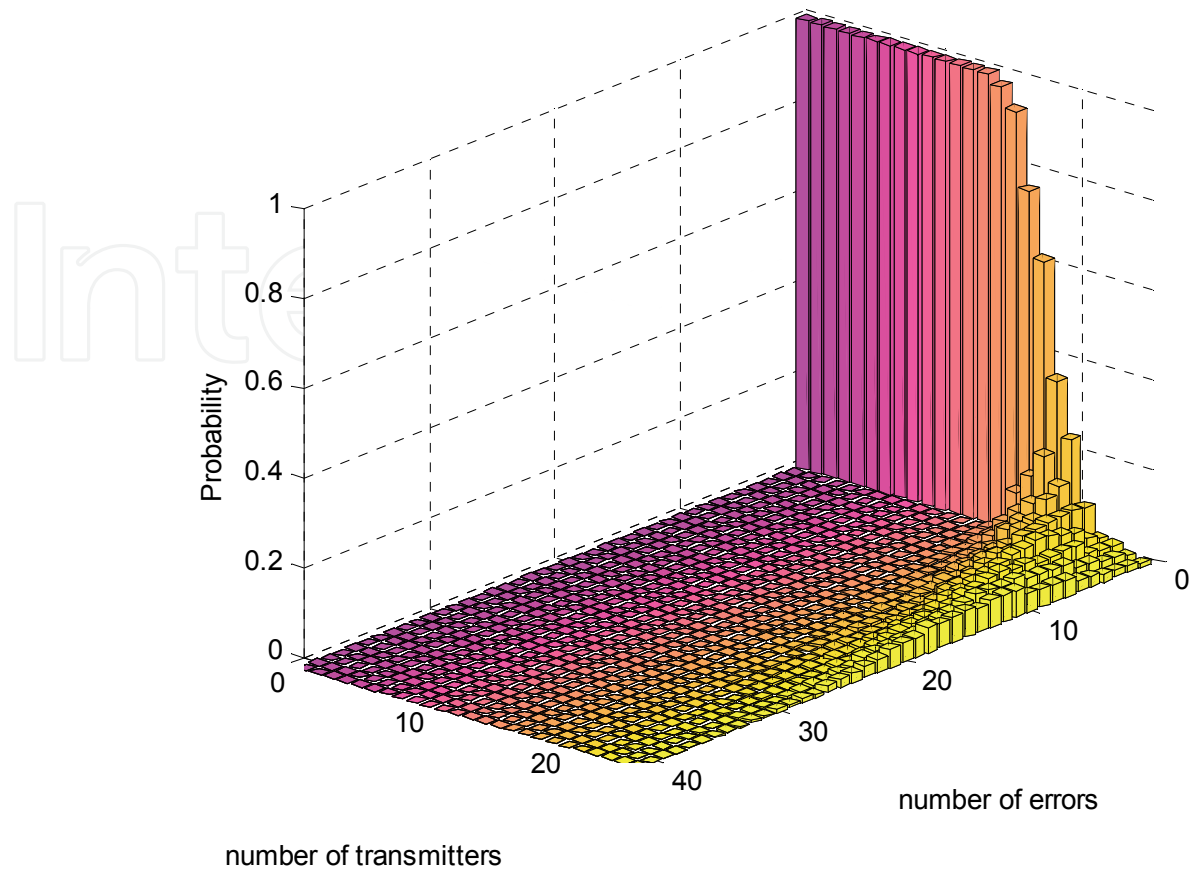
Fig. 5. Probability of errors in the RTS packets sent from 1 to 24 transmitters

Fig. 6. makes distributions of errors for data packets transmitted by 2 users so as each one with 4 antennas by the varying the distance, with 8 interferers which are placed at 300m. It is noted that each user is sending packet with rate equal to 30Mb/sec. The receiver with rate equal to 60Mb/sec is receiving a total of 1000-byte with 8 antennas in the slot. The transmitted power , which associated with any flow that has divided the packet, is 0.25/4 watt. The distances, which they are related to the charts, are: 50,100,110,120,130, 140, 150 and 200m. We observe that increase in distance leads to increase in error.

In Fig. 7. we show the probability of correct decoding RTS packets when they are received from 1 to 24 users simultaneously, corresponding to the probability of having 0 bit wrong, in the same above situation. The results show how to decode correctly a large number of handshake packets, which have the advantage of being short and being transmitted via a single antenna. The transmission via a single antenna gives the advantage of being able to concentrate all of power on a stream and loading a single signal in receiver. In fact, it has a very high probability of decoding up to 15 RTS packets which are simultaneously transmitted from 100m in the conditions of described interference. A station could receive multiple CTSs and decide itself how many antennas to use for spatially–multiplexed data transmission: as a rule of thumb, when more CTSs are heard, more streams will be sent in that part of the network, so more receivers will become overloaded, and fewer streams should be simultaneously sent by each transmitter.
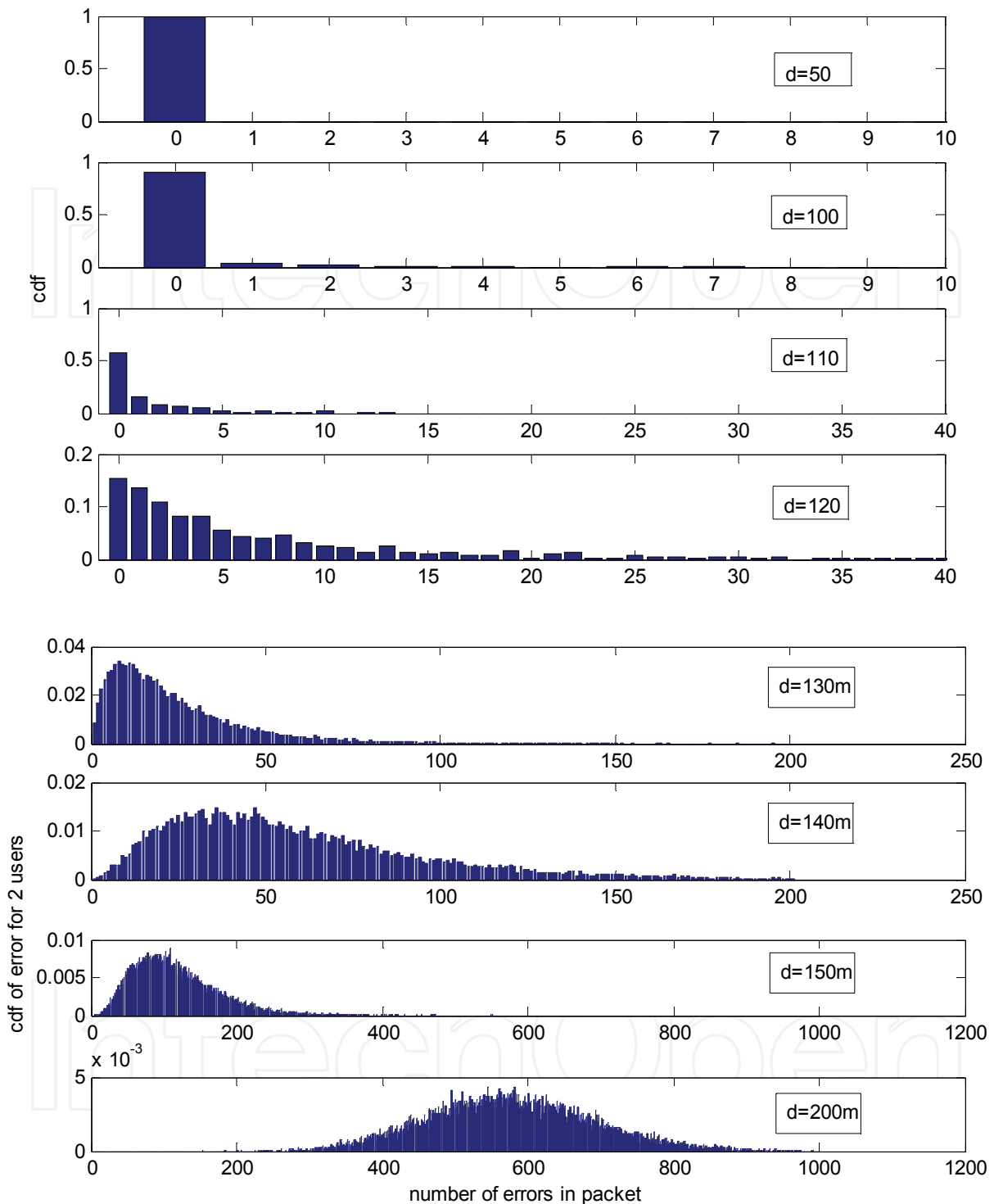
Fig. 6. Distributions of errors for data packets transmitted by 2 users each with 4 antennas by varying the distance

Fig. 8. shows the average error and standard deviation of the number of errors in the RTS packets, which are sent from 1 to 24 transmitters and are placed at 100m with 8 interferers which are placed at 300m are sending packets simultaneously. The standard deviation gives an idea of the magnitude, which distributes the number of errors: the error will increase by growth in the number of packets.
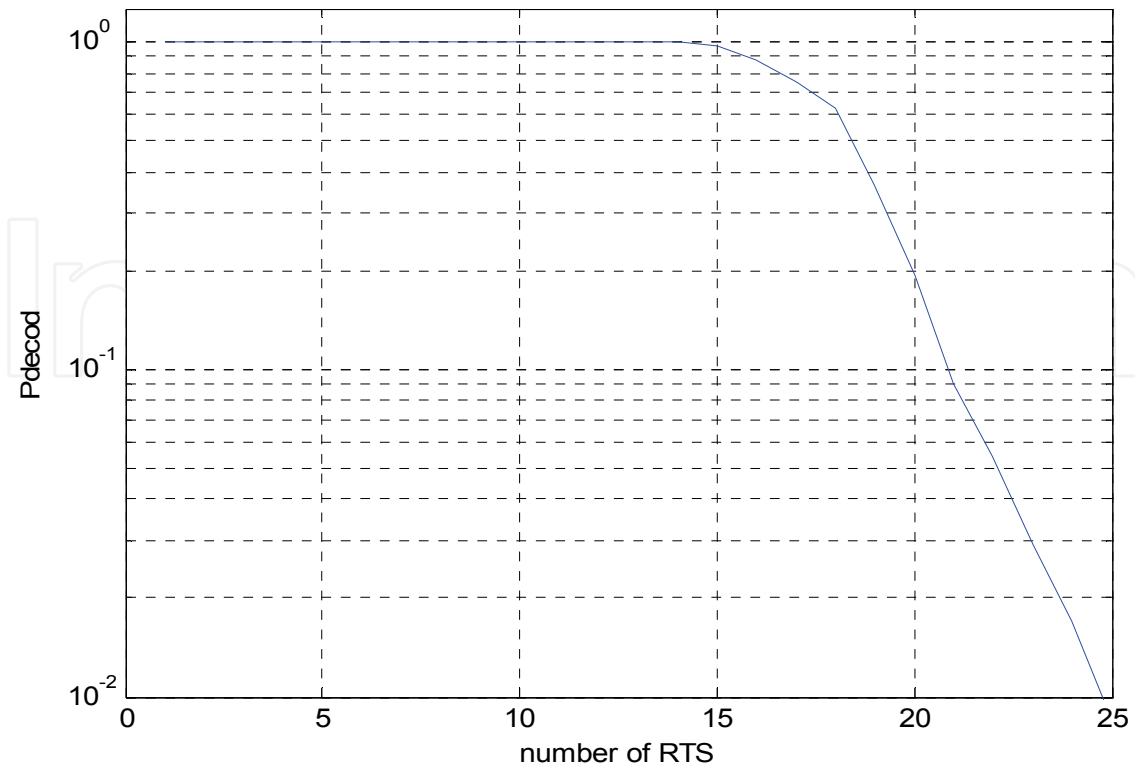
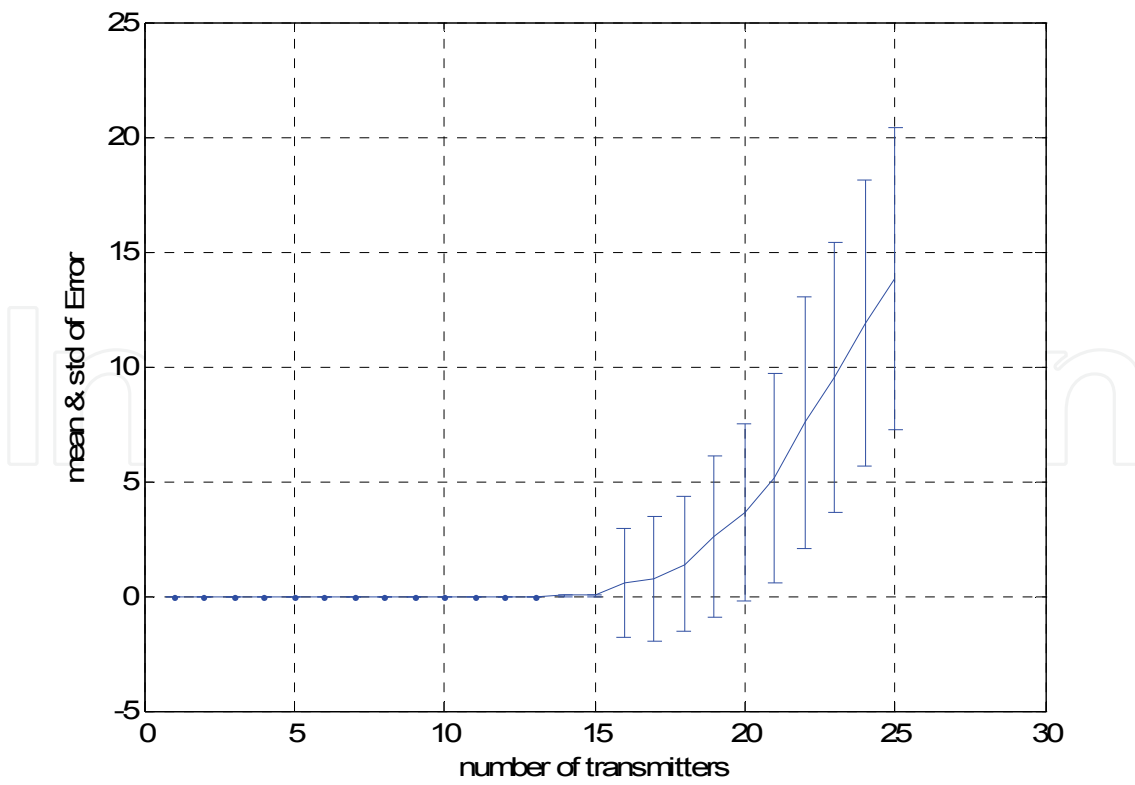Fig. 7. The probability of correct decoding for 1-24 RTS packets

Fig. 8. Mean and standard deviation of bit errors for 1-24 RTS packets

In Fig. 9. the average number of bit errors over two 500-byte packets split into two 4-streams is shown as a function of distance. We suppose each of two users is transmitting four 125-byte streams to a receiver with 8 antennas and 0, 4, 8 interferers are placed in 300m. As we can see, in the low load situation, there are no errors over a large range of distances. The error tolerance is very high up to a distance depending on the number of interfering users, i.e. those whose interference is not eliminated due to unknown channel state. In particular, two tx-rx links may continue to work at distances as far as 100-160m, depending on the interference level.
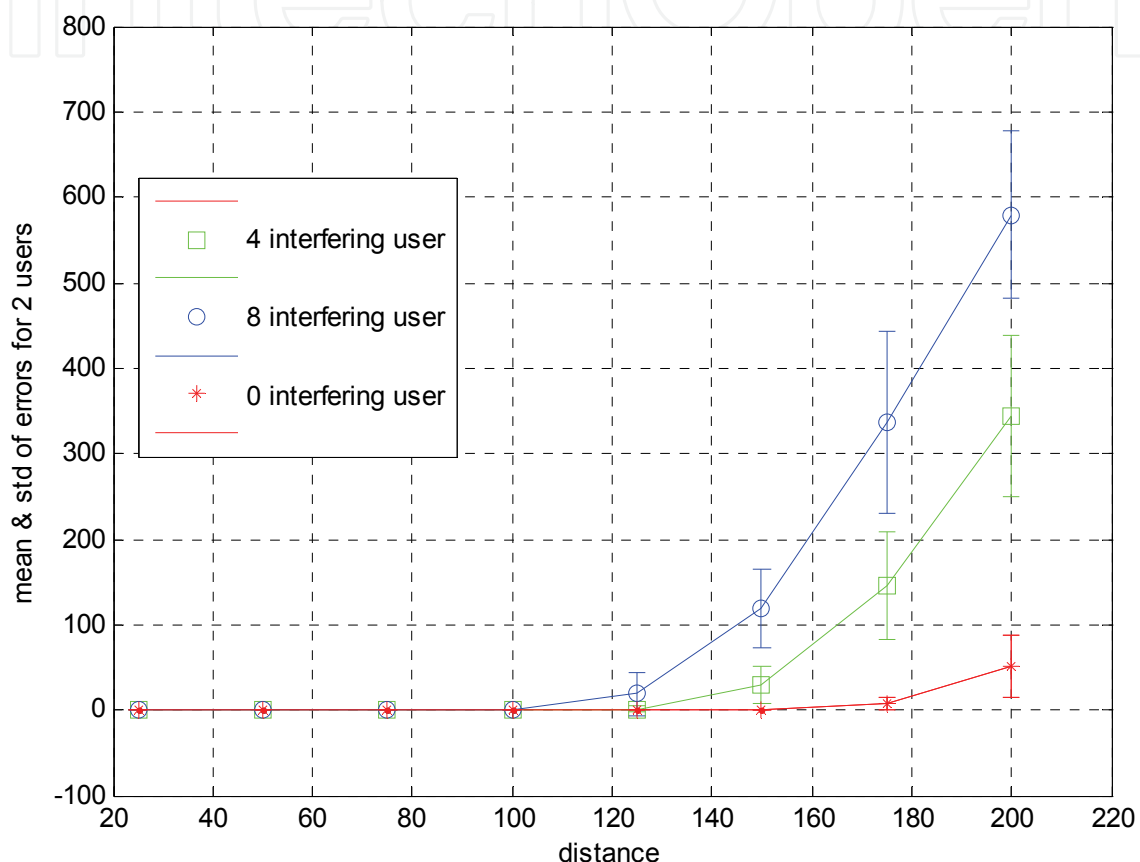

Fig. 9. Average and standard deviation of bit errors over two packets as a function of distance

Fig. 10. lists the CCDF (Complementary Cumulative Distribution Function, which expresses the probability that the number of errors are or exceed in this case, the figure reported in abscissa) the number of errors per transmitted DATA packet by 4 users and each has 4 antennas, with 8 interferers are placed in 300m. We see how the change in distance (20, 50, 70, 80 , 90 , 100m) causes a shift in distribution and the overload which is caused by the large number of streams, makes the performance of decoding very low even at limited distances. The probability that the number of errors is larger than zero for 20m, 50m and 80m is 0.009, 0.07 and 0.35 respectively.

Figs. 11 and 12, show the probability of the decoding of un-coded DATA packets are sent by one user and 2 users with 1, 2, 4, 6, 8 antennas respectively, equivalent to bit-rate of 7.5, 15, 30, 45, 60Mbps, and length of packets are 125, 250, 500, 750, 1000 bytes. For decoding purposes, we use the rate of 1/2 convolutional code is described by the octal coefficients
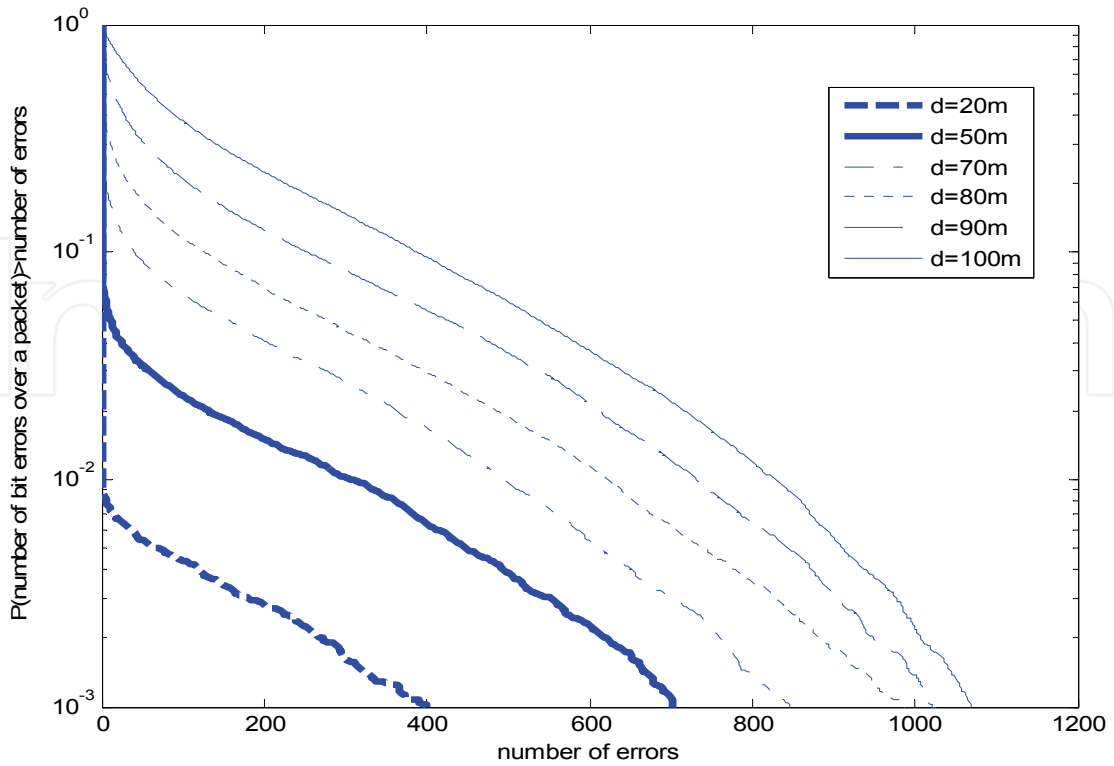
Fig. 10. Complementary cumulative distribution function of the number of errors
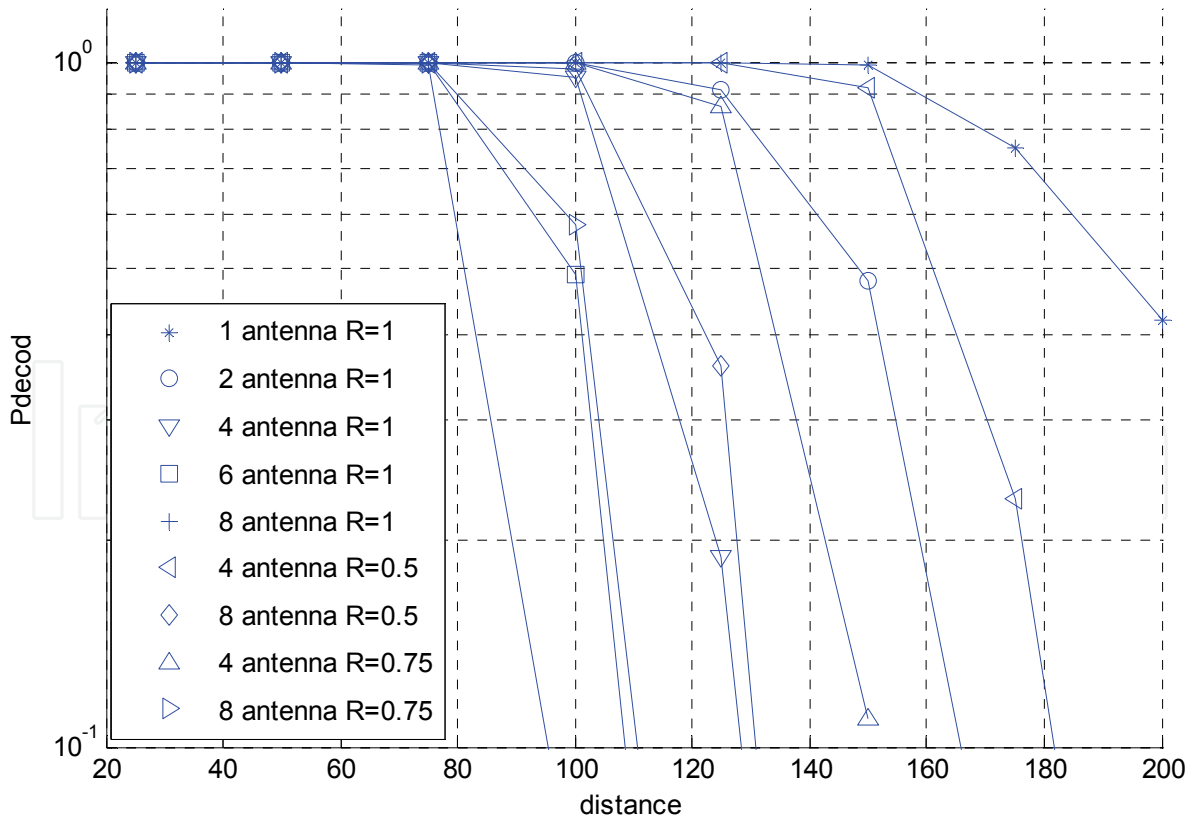

Fig. 11. Probability of correct packet reception for a single data transmission as a function of distance, with and without coding
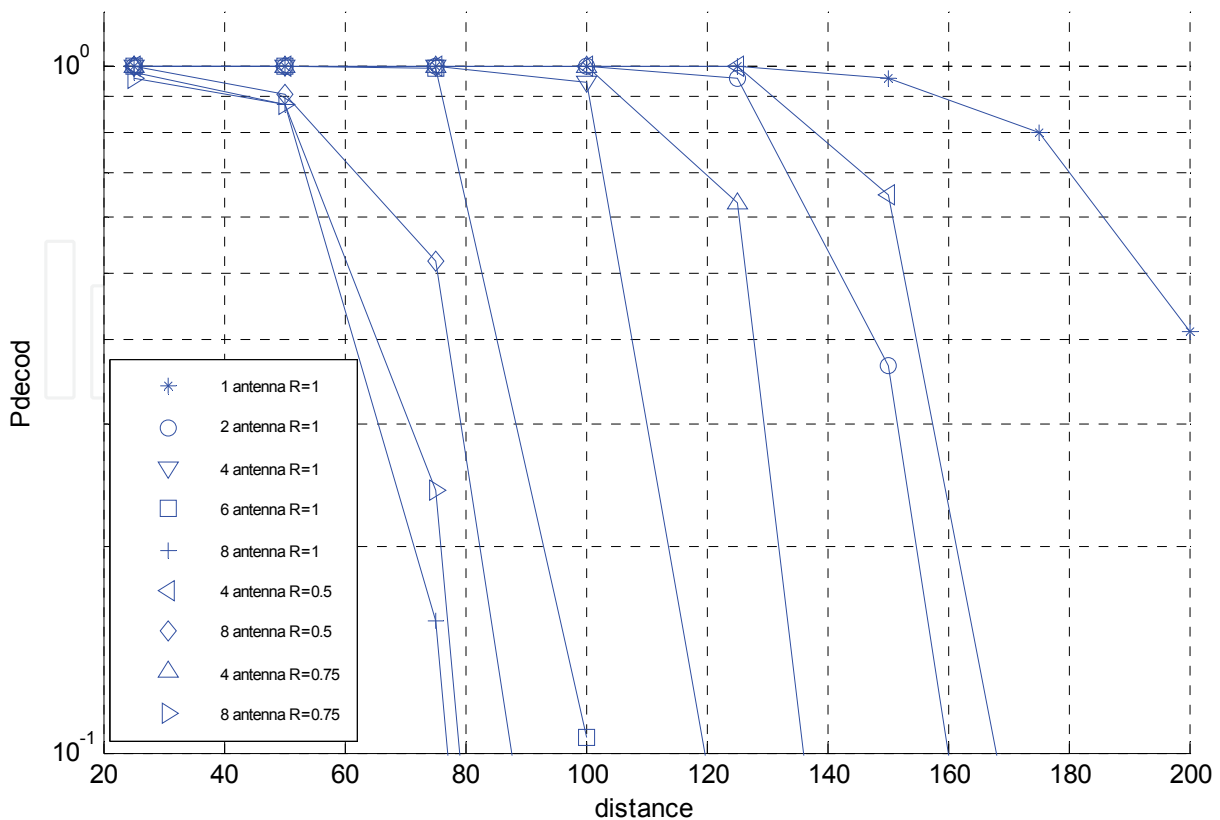
Fig. 12. Probability of correct packet reception for two users data transmission as a function of distance, with and without coding

$[133_8, 171_8]$ as specified, for instance, in the IEEE 802.11 standard. A 3/4 rate version of the code is obtained by puncturing the coded bits. As you can deduce, the distance in which an un-coded transmission becomes excessively error-prone varies as a function of the number of used antennas. The cases with one transmitter and two transmitters show a maximum reachable distance of about 125 and 100 meters respectively (when a single antenna is used) which falls to roughly 75 and 25 meters respectively when the complete set of available antennas are engaged in transmission. Where it has only one transmitter to full the capacity, we have a greater advantage to encode the data flow with further rate instead of reducing the number of antennas (even if we have more power to flow and earn diversity). This means that in low traffic conditions, coding makes it possible to reach farther distances at the price of an increased number of transmitting antennas. A MAC protocol should be able to exploit this favorable condition by forcing users to change adaptively their coding and antenna configuration, according to their own bit rate requirements and taking into account the adjacent nodes' status, which could be extrapolated from signaling packets. From above, if a node requires that at least an average percentage of its data transmission is correctly decoded, it may estimate (through RTS and CTS overhearing) how many its described receiver is loaded, the appropriate curve which corresponds to the required performance and distance to cover is selected from the graphs, hence it is necessary to establish the proper coding and spatial multiplexing scheme that would allow transmission at the desired successful probability, without overloading the receiver.

The information we get from this figure is that the coding cannot help anymore to reduce the interference from other data flow, which we have introduced, when target is to reach

farther distance. The system still has a very high performance even for a high number of transmitting antennas, if the distance from the receiver is kept below 25m but when the transmission distance increases, for seeing lesser interference it is better to send un-coded packets over fewer antennas. In addition, we infer that it would be preferable for a MAC protocol to split the longer packets into smaller units and transmit these units sequentially by using fewer antennas, somehow, the system load does not increase. This last result suggests that the use of channel coding (increasing the number of antennas) is not a very good choice. The lower transmit power and the increased receiver load tend to cancel the advantage which is introduced by the coding scheme. A similar problem would be found by using for example space–time codes, refer to (Jafarkhani, 2005; Alamouti, 1998 & Paulraj, 2003). Hence, in the following design, we decide to assume that no stream is actually coded. Our MAC protocol will focus on traffic control among adjacent nodes rather than bit rate and coding scheme adaptation.

Fig. 13. shows the bit-rate transmission versus distance. It is important to note that in the event of 2 users in transmission, the destination node is receiving data at double bit-rate in case of a single user.
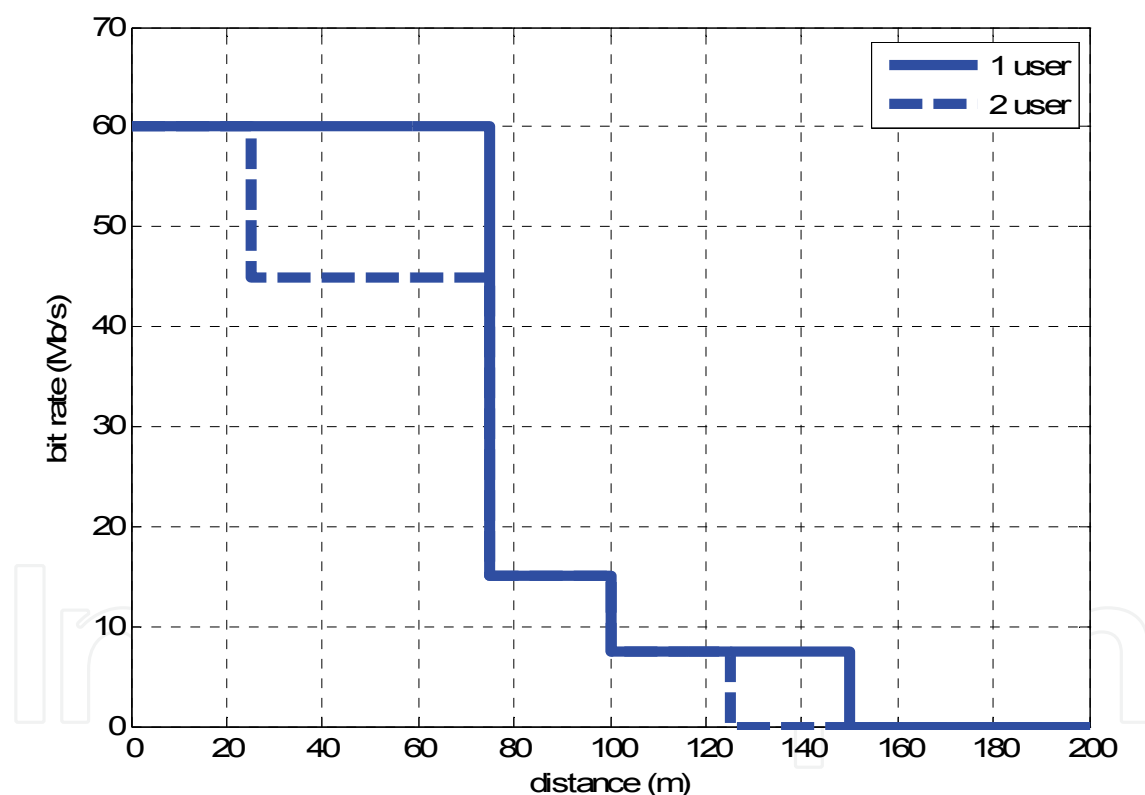

Fig. 13. Bit rate of data packets transmitted by 1 and 2 users by varying the distance

3. Cross layer MAC design for MIMO Ad Hoc networks

3.1 Introduction

The IEEE 802.11 protocol includes a specific mode called ad hoc. This mode operates according to the so-called Distributed Coordination Function (DCF). In turn, DCF defines two different modes, the basic mode (with random access after carrier sensing) and the

collision avoidance mode (with four-way handshaking before channel access). We know that preventing collisions would result in loss of data and waste of resource. In this section we want to introduce a good solution for hidden terminal problem in ad hoc network. With some channel knowledge, obtained through training sequences, receiver detects incoming streams separately. Each node have a limited capability of N_s^{max} sequence simultaneously. So the protocol must be aware of the tradeoff existing between the among of wanted data to detect and the interference protection granted to this data. In other word, without enough resources for interference cancellation, the receiver is not aware of interfering nodes nearby and so it can not estimate their channel and cancel them. Indeed, instead of blocking mechanisms, such as 802.11, we want to have simultaneous transmissions. We also want to exploit the spatial demultiplexing capability of MIMO processing.

In our approach, we consider that channel of nodes with a certain distance from receiver can be detected and cancelled and nodes with further distance and low received power can not be cancelled. In Fig. 14. we show the probability of correct receiving a data packet in the presence of interfering traffic versus the distance of the transmitter, for varying number of antenna used by the transmitter. We see that with a 90% minimum success ratio, a transmitter could reach 70m, 90m, 110m, using 8, 4 and 2 antennas respectively. It means that the maximum number of antennas allowed when transmitting to a set of receivers including corresponded neighbor. We use a framed communication structure, with four phases. Theses phases are designed according to standard sequence of messages in a collision avoidance mechanism, and are summarized as follows.
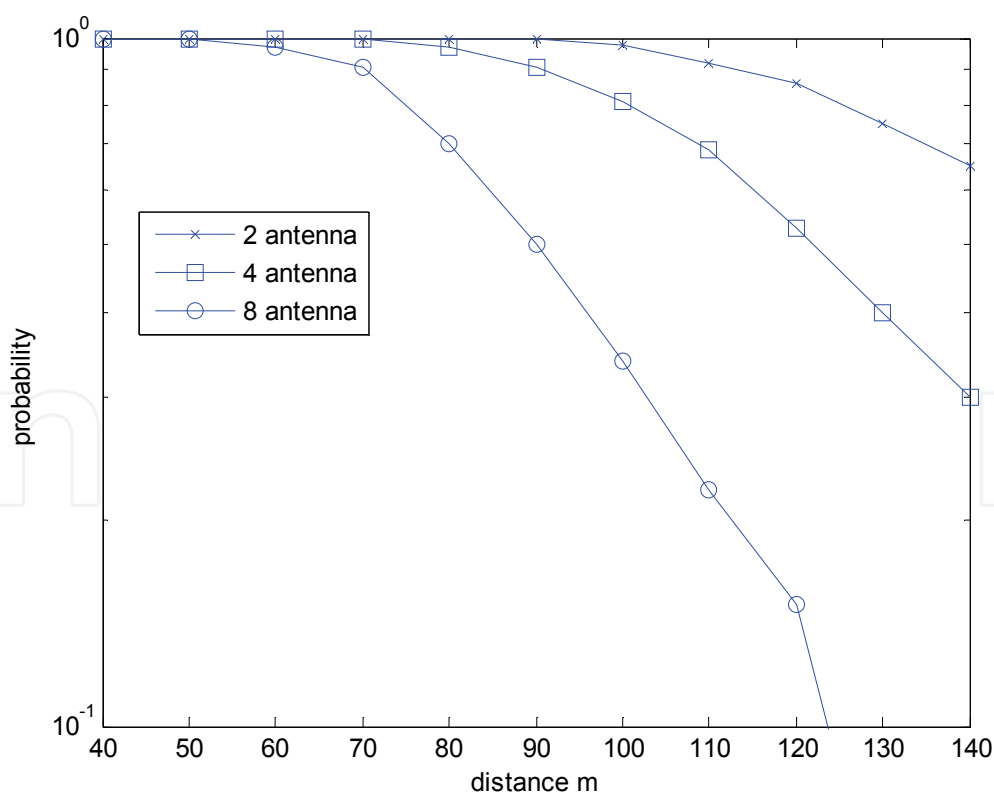

Fig. 14. Probability of correct receiving a data packet by varying the distance and number of transmitter antennas

Sending RTS packet: In this phase, all senders look into their backlog queue, and if it is not empty they compose transmission requests and pack them into a single RTS message. Each packet in the queue is split into multiple streams of fixed length, such that each stream can be transmitted through one antenna. Any RTS has to specify the number of streams to be sent simultaneously, in addition to the intended destination node. How to associate a destination node with a suitable number of transmit antenna depends on the degree of spatial multiplexing sought, as well as the local traffic intensity, thus the queue level of the sender. Any RTS may contain several such requests. Moreover, an RTS is always sent with one antenna and at full power. Each node selects number of antennas according to number of streams of current packet and keeps free other antennas for sending other packets.

Sending CTS packet: During this phase, all nodes that were not transmitters, themselves receive multiple simultaneous RTSs, and apply the reception algorithm of section 2 to separate and decode them. CTSs are also sent out using one antenna and at full power.   We use 4 schemes for receiving data and interfering streams to control the number of allowed transmitters and antennas.

Sending DATA packet: All transmitters receive CTSs and, after BLAST detection, they follow CTS indication and send their streams.

Sending ACK packet: After detection, all receivers evaluate which streams have been correctly received and send an ACK back to the transmitters. After the last phase the data handshake exchange is complete, the current frame ends and the next is started.

A random backoff is needed for nodes that do not receive a CTS, as otherwise persistent attempts may lead the system into deadlock. We make use a standard exponential backoff. Accordingly, before transmitting, node wait for a random number of frames, uniformly distributed in the interval $[1, BW(i)]$, where i tracks the current attempts, and $BW(i) = 2^{i-1}W$, with W a fixed backoff window parameter refer to (IEEE 802.11 Standard, 2007).

3.2 RTS and CTS sending schemes

To specify our MAC protocol, we need to introduce a simpler protocol for comparison. The definition of this protocol is necessary, since the approaches described in Section 2 can not be directly compared to our solution, because of either the absence of a specific MAC scheme refer to (chen and Gans, 2005), the optimization of MAC around some fixed PHY parameters such as the number of antenna refer to (Vang and Tureli, 2005), the diverse issue related to different modulation and signaling scheme refer to Hu and Zhang (2004), the attention devoted to achieving full diversity instead of full parallelism refer to (Hu and Zhang, 2004), or the idealized assumptions about a MIMO PHY level and MAC signaling refer to (Sundaresan et al., 2004) This protocol is meant as an example of how a layered networking solution would behave when set up on top of a SM-capable MIMO PHY level. Furthermore, it is directly comparable with our policies, as it can into account the PHY used (unlike (Sundaresan et al., 2004), that focuses on link capacity) and is sufficiently general not to depend on the number of antenna per node (unlike (Vang and Tureli, 2005)). When a node is granted access, it sends an RTS and waits for a CTS. With MIMO transmission, packets are divided in streams, each 125-byte long. To increase bit rate, streams are split in substreams, one per each available antenna and transmitted in parallel through all antenna. If a packet is formed of a number of 125-byte streams and N_a=8, each antenna will send one 125-bit substream per stream. Ack'ed substreams remove from the queue of node and

streams with errors are retransmitted. Indeed, Simpler protocol is a CSMA/CA protocol, just using a more powerful MIMO PHY layer.

3.2.1 RTS sending scheme

Consider that the set of neighbors of a given node s be denoted as $\mathcal{V} = \{v_1, v_2, \dots\}$. Let a_{sv_j} be the maximum number of antennas that s can uses when transmitting to any set of nodes that includes v_j. Suppose that node n is current node. At step $i = 1$, a request is created as fellows. The node reads the $k_1 = 1$ packet's destination, d_{k_1}, and the number of unsent streams, p_{k_1}. After that, node compares p_{k_1} with maximum antenna constraint, $a_{nd_{k_1}}$. If $p_{k_1} > a_{nd_{k_1}}$, the streams violate from maximum antenna constraint, hence forbidding any further spatial multiplexing. The request pair $(d_{k_1}, a_{nd_{k_1}})$ is inserted in the RTS packet.

If $p_{k_1} \leq a_{nd_{k_1}}$, the pair (d_{k_1}, p_{k_1}) is inserted in the RTS. Each node keeps indices of all packets selected for transmission in set S_i. The total number of antennas allocated until step i hold in $A(i)$. In the absence of interferes, node d_{k_1} could support $a_{nd_{k_1}} - p_{k_1}$ further antenna. So, the node goes to step 2 and searches its queue , until it finds a packet k_2 that maximum number of destination's antennas match the condition $a_{nd_{k_2}} \geq A(1)$. This means that the d_{k_2} can stand the transmission of the $A(1)$ streams from other node, in addition to its own. The transmitter sets $S_2 = S_1 \cup \{k_2\}$, calculates the number of streams allocated to packet k_2 as $M(2) = min\{min\{a_{nd_{k_1}}, a_{nd_{k_2}}\} - A(1), p_{k_2}\}$, that not violate the maximum number of antennas constraints $a_{nd_{k_1}}$ and $a_{nd_{k_2}}$ and $A(1)$ streams have been allocated. Then, it inserts in the RTS packet the pair $(d_{k_2}, M(2))$, and finally updates $A(2) = A(1) + M(2)$. If there is still antenna for transmission without saturating antenna constraints, algorithm goes to next step and so on. In general, at step i, the node searches the queue for a packet k_i with condition $a_{nd_{k_i}} > A(i-1)$. Then $S_i = S_{i-1} \cup \{k_i\}$, $M(i) = min\{min_{j \in S_i} a_{nd_{k_j}} - A(i-1), p_{k_i}\}$, and $A(i) = A(i-1) + M(i)$. The request $(d_{k_i}, M(i))$ is put in the RTS. The algorithm then goes to step $i + 1$ if and only if $min_{j \in S_i} a_{nd_{k_j}} > A(i)$ and a packet such that $a_{nd_{k_{i+1}}} > A(i)$ is found in the queue refer to (Casari et al., 2008). As an example consider Fig. 15. Another example with further request could be found in Fig. 16. In Fig. 17. we show a pseudo code of transmitter protocol.

Rx id	Number of streams	Maximum number of antennas
8	1	4
13	2	8
7	4	2
4	4	4
15	1	8

Request 1		Request 2		Request 3	
Rx id	streams	Rx id	streams	Rx id	streams
8	1	13	2	4	1

Fig. 15. An example of application of RTS sending scheme.

Rx id	Number of streams	Maximum number of antennas
8	1	4
7	2	4
13	1	8
4	1	2
15	4	8

Request 1		Request 2		Request 3		Request 4	
Rx id	streams	Rx id	streams	Rx id	streams	Rx id	streams
8	1	7	2	13	1	15	1

Fig. 16. Another example of application of RTS sending scheme with further request

transmitter protocol
// Initialize the step index , the number of allocated antennas A , the set of receivers S and the number of failures N_{fail}
$i = 1; A(0) = 0; S_0 = \phi; N_{fail} = 0$
// RTS phase: add users until class constraint are violated
While $min_{j \epsilon S_i} a_{nd_j} > A(i - 1)$ **do**
   // Is there a packet in the queue that complies with the current constraints?
   if $\exists$ a packet k_i s.t. $a_{nd_{k_i}} > A(i - 1)$ **then**
     // Add user as receiver
     $S_i = S_{i-1} \cup \{k_i\}$
     // Determine number of streams to send that does not violate any current class constraint
     $M(i) = min\{min_{j \epsilon S_i} a_{nd_{k_j}} - A(i - 1), p_{k_i}\}$
     $A(i) = A(i - 1) + M(i)$
     Insert request $(d_{k_i}, M(i))$ in RTS
   end if
end while
Send RTS
// Data phase: check CTS
if one or more CTS received **then**
   Send data streams according to CTSs
   $N_{fail} = 1$
else
   Backoff for b frames, b uniformly distributed in $[1, W.2^{N_{fail}-1}]$
   $N_{fail} = N_{fail} + 1$
end if
if ACK received then
   Mark all ACK'ed streams
   Remove from the queue all packets whose streams have been all ACK'ed
end if

Fig. 17. Pseudo code of transmitter protocol

3.2.2 CTS sending schemes

In this section we report 4 schemes for receiving data from transmitters. All of these schemes contain two set $\mathcal{W}$ and $\mathcal{U}$. The first set contains all requests directed to the node that names wanted request, the second set all other requests that names unwanted request. We knows that if p_k streams implies to transmitted, the receiver estimates channel of this streams. After that, number of available estimating resources is $N_s^{max} - p_k$. If $N_s^{max} - p_k > 0$ and exist any request in the node queue, process will be continued in the next step and so on.

SNR based receiver protocol: The node grants first highest power request in $\mathcal{W}$ and then considers all other requests in $\mathcal{W} \cup \mathcal{U}$, re-ordered by decreasing received power. In Fig. 18. we report a pseudo code of SNR based receiver protocol. In Fig. 19. an example of application of this protocol is showed.

SNR based receiver protocol
//Initialize number of trackable training sequences, N_s
$N_s = N_s^{max}$
// CTS phase: apply CTS policy
if one or more RTSs received **then**
  Create ordered sets $\mathcal{W}$ and $\mathcal{U}$
  Let $I_\mathcal{W}$ be the ordered set with the indices of the packets in $\mathcal{W}$
  Let $I_\mathcal{U}$ be the ordered set with the indices of the packets in $\mathcal{U}$
  //Grant at least one wanted request
  $i = I_\mathcal{W}(1)$
  Read source s_i and number of data streams p_i for the packet with index i
  Insert grant (d_i, p_i) in CTS
  $N_s = N_s - p_i$
  $I_\mathcal{W} = I_\mathcal{W} - \{i\}$
  // Manage other requests in order of decreasing received power
  While $N_s > 0$ & $(I_\mathcal{W} \neq \phi \text{ or } I_\mathcal{U} \neq \phi)$ **do**
    Let i be the request with the greatest power between $I_\mathcal{W}(1)$ and $I_\mathcal{U}(1)$
    $N = min\{p_i, N_s\}$
    $N_s = N_s - N$
    if $i \in I_\mathcal{W}$ **then**
      Insert grant (d_i, N) in the CTS
      $I_\mathcal{W} = I_\mathcal{W} - \{i\}$
    else
      $I_\mathcal{U} = I_\mathcal{U} - \{i\}$
    end if
  end while
end if
Send CTS
//Data phase: receive data streams
if Data streams received **then**
  De-multiplex streams and extract wanted ones
  Send ACK for correctly received streams belonging to requests in $\mathcal{W}$
end if

Fig. 18. Pseudo code of SNR based receiver protocol

First wanted based receiver protocol: In this protocol, a node gives priority to wanted transmission. If any estimating resources left , it then begins to consider unwanted requests. In Fig. 20. we report a pseudo code of first wanted based receiver protocol. In Fig. 21. an example of application of this protocol is showed.

Wanted based receiver protocol: In this case, the node grants the requests in $\mathcal{W}$ and does not consider $\mathcal{U}$ at all. In Fig. 22. we report a pseudo code of wanted based receiver protocol. In Fig. 23. an example of application of this protocol is showed.

SNR based receiver protocol without interference cancellation: This scheme operates as SNR based receiver protocol, but does not perform cancellation of interfering requests in $\mathcal{U}$. It means that only powerful interferes could be considered. In Fig. 24. we report a pseudo code of SNR based receiver protocol without interference cancellation. In Fig. 25. an example of application of this protocol is showed.
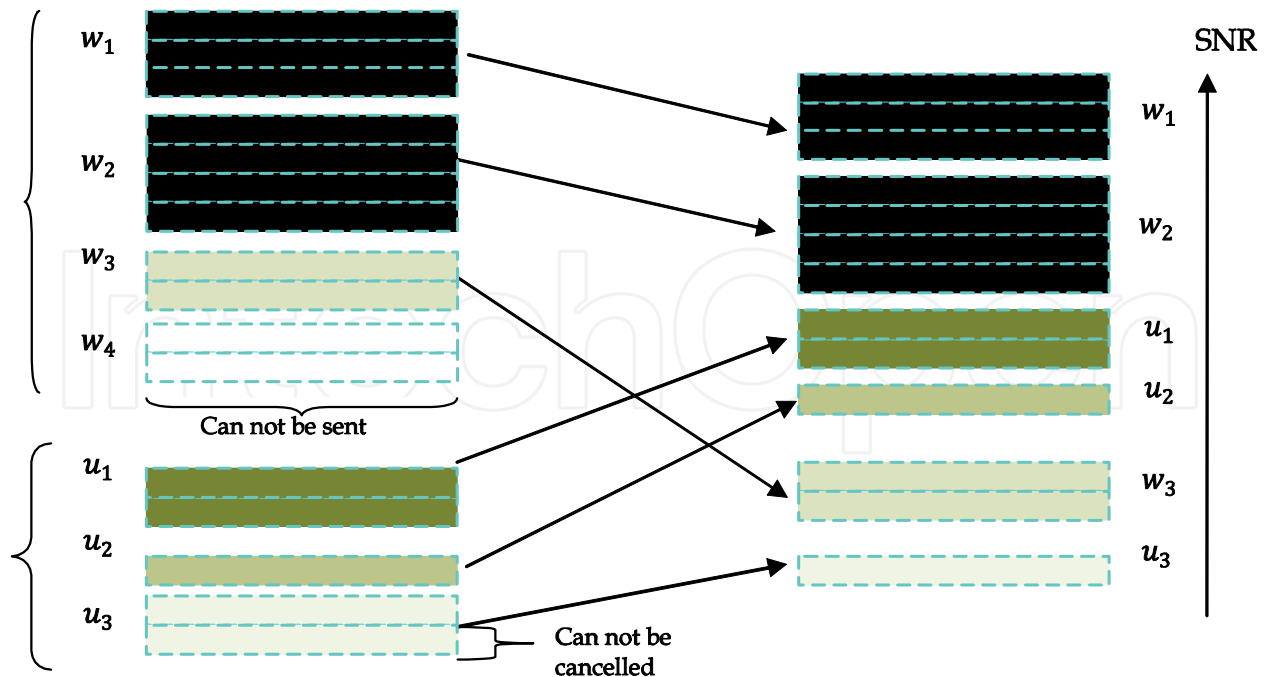
Fig. 19. An example of application of SNR based receiver protocol.

First wanted based receiver protocol
//Initialize number of trackable training sequences, N_s
$N_s = N_s^{max}$
// CTS phase: apply CTS policy
if one or more RTSs received **then**
   Create ordered sets $\mathcal{W}$ and $\mathcal{U}$
   Let $I_{\mathcal{W}}$ be the ordered set with the indices of the packets in $\mathcal{W}$
   Let $I_{\mathcal{U}}$ be the ordered set with the indices of the packets in $\mathcal{U}$
   //Grant at least one wanted request
   While $N_s > 0$ & $(I_{\mathcal{W}} \neq \phi)$ **do**
     $i = I_{\mathcal{W}}(1)$
     $N = min\{p_i, N_s\}$
     $N_s = N_s - N$
     Read source s_i and number of data streams p_i for the packet with index i
     Insert grant (d_i, N) in the CTS
     $I_{\mathcal{W}} = I_{\mathcal{W}} - \{i\}$
   end while
   While $N_s > 0$ & $(I_{\mathcal{U}} \neq \phi)$ **do**
     $i = I_{\mathcal{U}}(1)$
     $N = min\{p_i, N_s\}$
     $N_s = N_s - N$
     $I_{\mathcal{U}} = I_{\mathcal{U}} - \{i\}$
   end while
end if
Send CTS
//Data phase: receive data streams
if Data streams received **then**
   De-multiplex streams and extract wanted ones
   Send ACK for correctly received streams belonging to requests in $\mathcal{W}$
end if

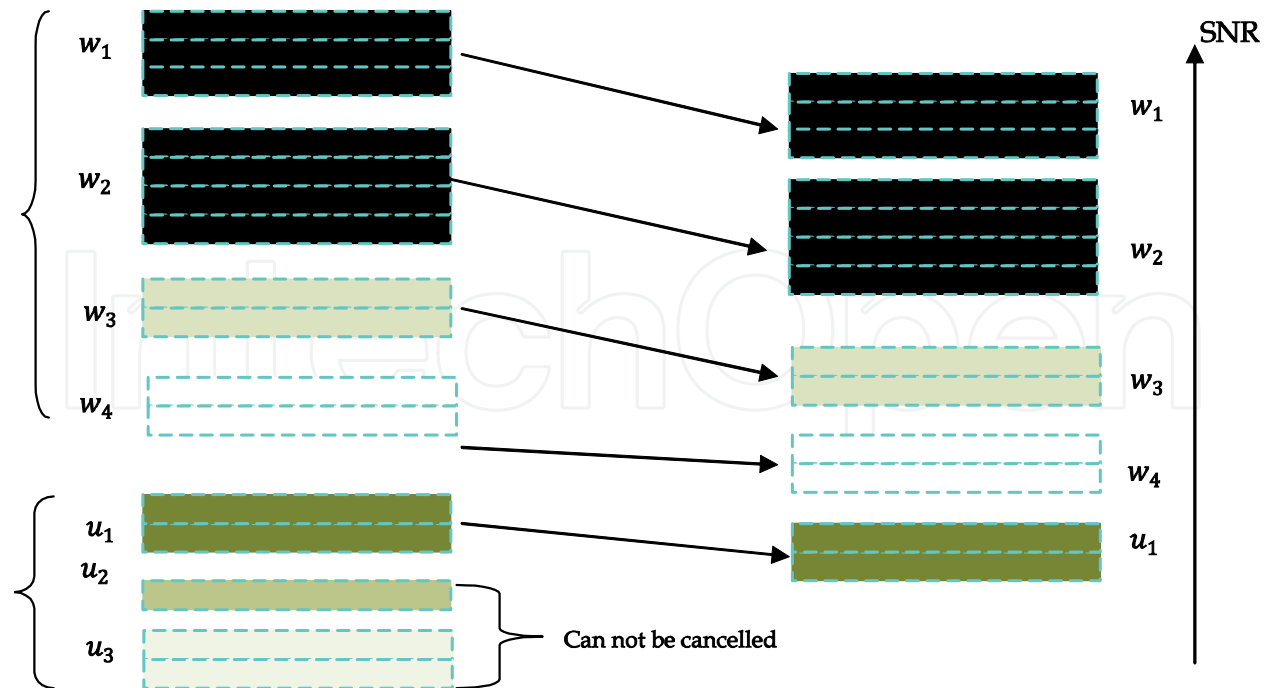Fig. 20. Pseudo code of first wanted based receiver protocol

Fig. 21. An example of application of first wanted based receiver protocol.

Wanted based receiver protocol
//Initialize number of trackable training sequences, N_s
$N_s = N_s^{max}$
// CTS phase: apply CTS policy
if one or more RTSs received **then**
  Create ordered sets $\mathcal{W}$
  Let $I_{\mathcal{W}}$ be the ordered set with the indices of the packets in $\mathcal{W}$
  //Grant at least one wanted request
  While $N_s > 0$ & $(I_{\mathcal{W}} \neq \phi)$ **do**
    $i = I_{\mathcal{W}}(1)$
    Read source s_i and number of data streams p_i for the packet with index i
    $N = min\{p_i, N_s\}$
    $N_s = N_s - N$
    Insert grant (d_i, N) in the CTS
    $I_{\mathcal{W}} = I_{\mathcal{W}} - \{i\}$
  end while
end if
Send CTS
//Data phase: receive data streams
if Data streams received **then**
  De-multiplex streams and extract wanted ones
  Send ACK for correctly received streams belonging to requests in $\mathcal{W}$
end if

Fig. 22. Pseudo code of wanted based receiver protocol.
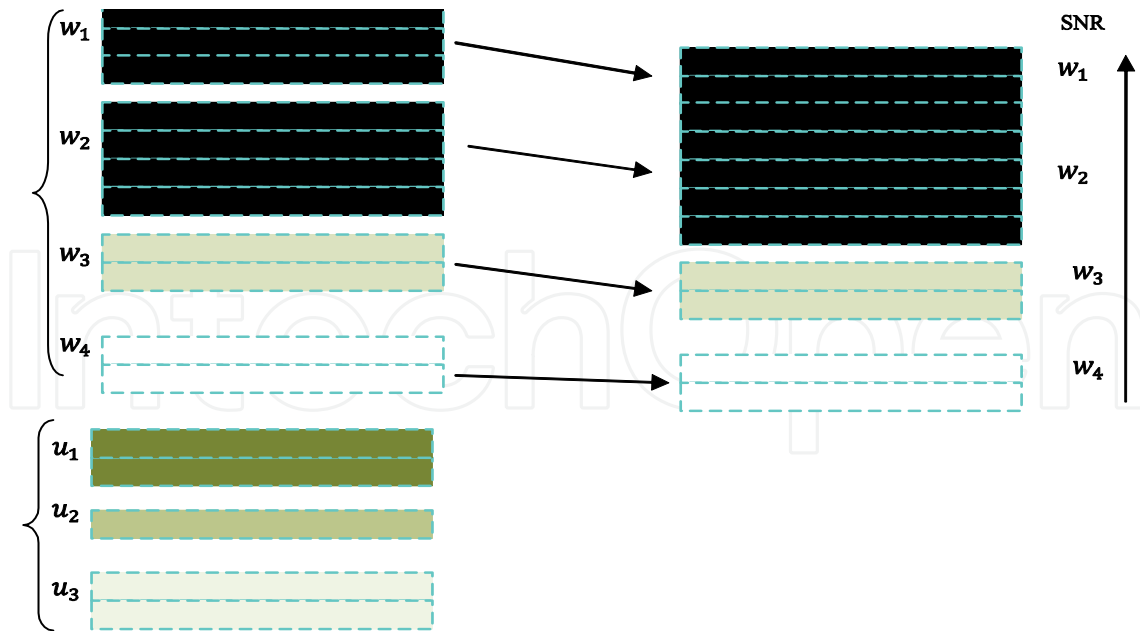
Fig. 23. An example of application of wanted based receiver protocol.

SNR based receiver protocol without interference cancellation
//Initialize number of trackable training sequences, N_s
$N_s = N_s^{max}$
// CTS phase: apply CTS policy
if one or more RTSs received **then**
   Create ordered sets $\mathcal{W}$ and $\mathcal{U}$
   Let I_w be the ordered set with the indices of the packets in $\mathcal{W}$
   Let I_u be the ordered set with the indices of the packets in $\mathcal{U}$
   //Grant at least one wanted request
   $i = I_w(1)$
   Read source s_i and number of data streams p_i for the packet with index i
   Insert grant (d_i, p_i) in CTS
   $N_s = N_s - p_i$
   $I_w = I_w - \{i\}$
   // Manage other requests in order of decreasing received power
   While $N_s > 0$ & ($I_w \neq \phi$ or $I_u \neq \phi$) **do**
      Let i be the request with the greatest power between $I_w(1)$ and $I_u(1)$
      $N = min \{p_i, N_s\}$
      $N_s = N_s - N$
      if $i \in I_w$ **then**
         Insert grant (d_i, N) in the CTS
         $I_w = I_w - \{i\}$
      end if
   end while
end if
Send CTS
//Data phase: receive data streams
if Data streams received **then**
   De-multiplex streams and extract wanted ones
   Send ACK for correctly received streams belonging to requests in $\mathcal{W}$
end if

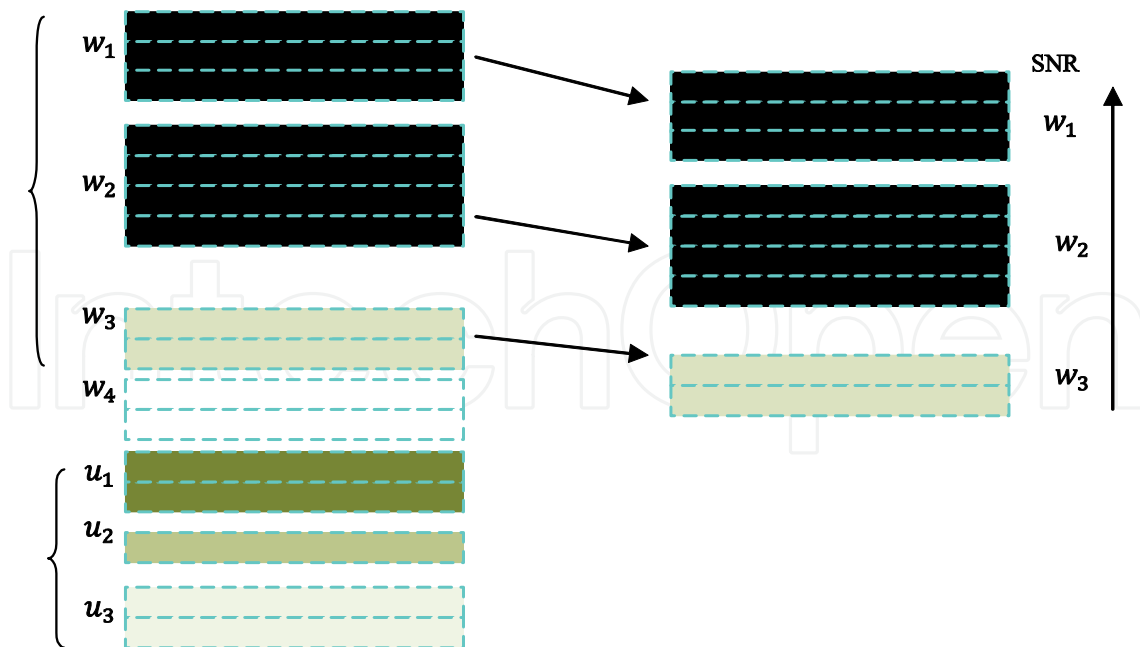Fig. 24. Pseudo code of SNR based without interference cancellation receiver protocol.

Fig. 25. An example of application of SNR based without interference cancellation receiver protocol.

CTS sending schemes are the only way to reduce data traffic in ad hoc network, since RTS/CTS are not used for channel reservation., but rather as an indication of intention /clearance to transmit, also both RTS and CTS sending schemes favor the creation of multiple point to point links, all potentially making use of SM. This is made possible by inserting multiple requests (grants) in the RTS (CTS), each composed of multiple streams. These schemes can operate on top of any PHY that successively detects multiple signals, cancels their contribution from the received signal. We choose V-BLAST as one such PHY, since it is a good representative and has recently received a lot of attention refer to (Zhang and Lee, 2008).

3. Network simulation setup & results

For evaluating our MAC scheme, we deploy 25 nodes randomly in a square area with 8 antennas each and nearest neighbors 25 m apart. Traffic is generated according to a Poisson process of rate λ packets per second per node. Each generated packet is made of k 125-bytes long streams, with k randomly chosen in the set {1, 2, 3, and 4}. Unsent packets are buffered. Each node has a finite FIFO queue where the packets are stored before being served. We also study the effect of convolutional coding on data packets using the standard 802.11 code refer to (IEEE 802.11 standard, 2007), W and $BW_{\max}$ are 1 and 32 respectively. For our simulation, we used the MATLAB.

Fig. 26. shows the average network throughput defined as a function of the offered traffic λ , defined as the number of correctly detected 125-byte streams per frame for all CTS sending schemes. We see that wanted based receiver protocol has bad performance, because it permits the sending of all requested streams and does not cancel any interferers. First wanted based receiver protocol have better performance than wanted based, because it has a way to cancel highest SNR interfering streams. Indeed, from network load 700, the amount

of requested traffic have not enough antenna for cancellation of unwanted signals and lead to decrease in the throughput. In the worst case, one wanted request protected against $N_s^{max} - 1$ strongest interferences and lead to best performance of SNR based receiver protocol.

Fig. 27. shows the average queue length as a function of the offered traffic λ for all CTS sending schemes. We see that first wanted based protocol because of lower throughput at network load larger than 800 does not allow sufficient packet sending. Also SNR based protocol have shorter queue length. We observe that other protocol reach to upper bound of delay. SNR based receiver protocol without interference cancellation has bad performance because it hasn't interference cancellation feature. Results show that the SNR based receiver protocol reach to best performance , as it has high throughput and throughput ratio, limited delay and queue length.

4. Conclusions

In this study, we combine MIMO multiuser detection at PHY layer with design of a protocol at MAC layer in a cross layer fashion simultaneously to have a better throughput for mobile ad hoc networks. As we can see in Fig. 26. this approach is able to support up to 12
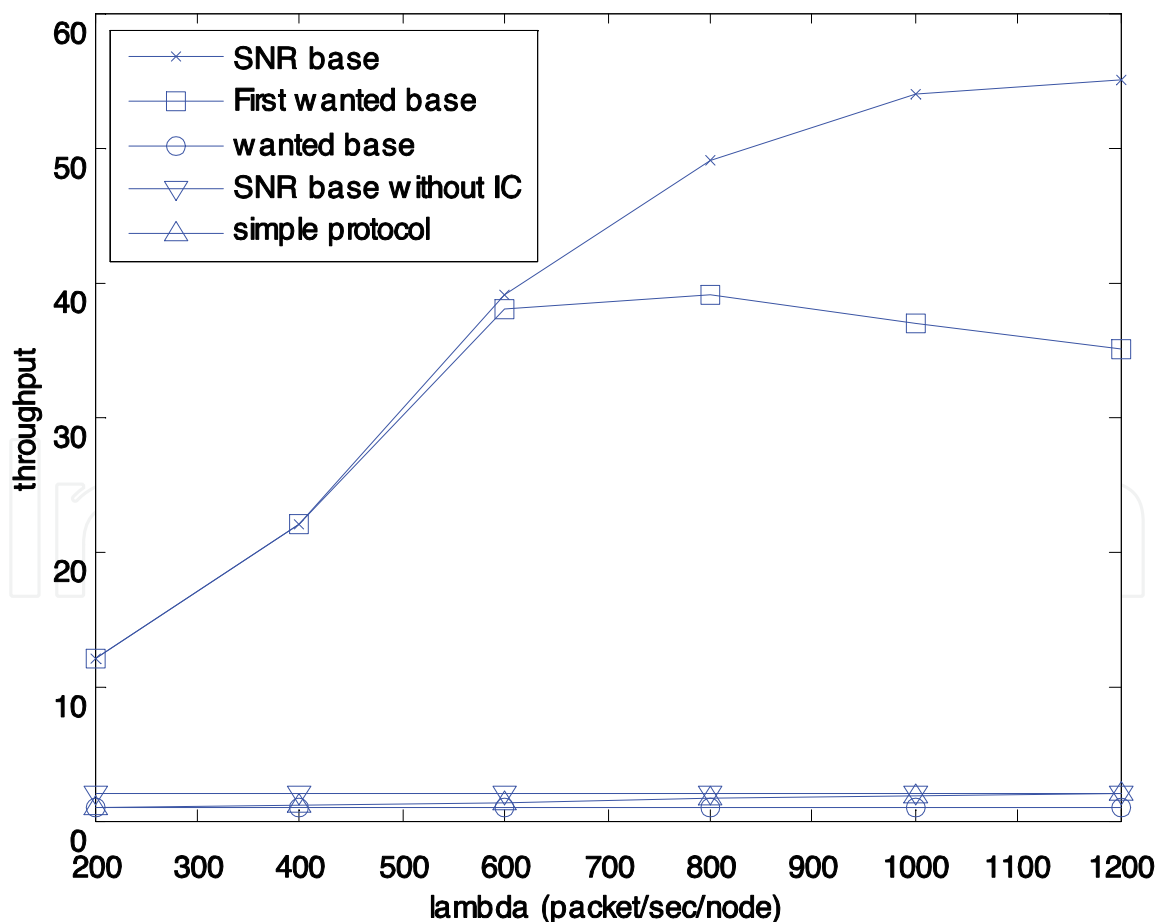

Fig. 26. Network throughput versus network traffic.

successful 125-byte streams per frame on average, which is larger than the maximum number of antennas per node, i.e., 8. This is a very interesting result. It substantiates the need for both a well-designed physical layer and a management protocol, and shows that the number of terminal antenna is a soft limit in MIMO ad hoc networks, if the effective rejection of multiple access interferences is provided. Also in Fig. 27 we show that average queue length is shorter than maximum length of queue, i.e., 120. Future work on this topic may be the extension to routing layer issues. Our scheme can be used on laptops that each one is considered as an ad hoc node and uses 8 antennas with 3 cm distance between the two adjacent antennas.
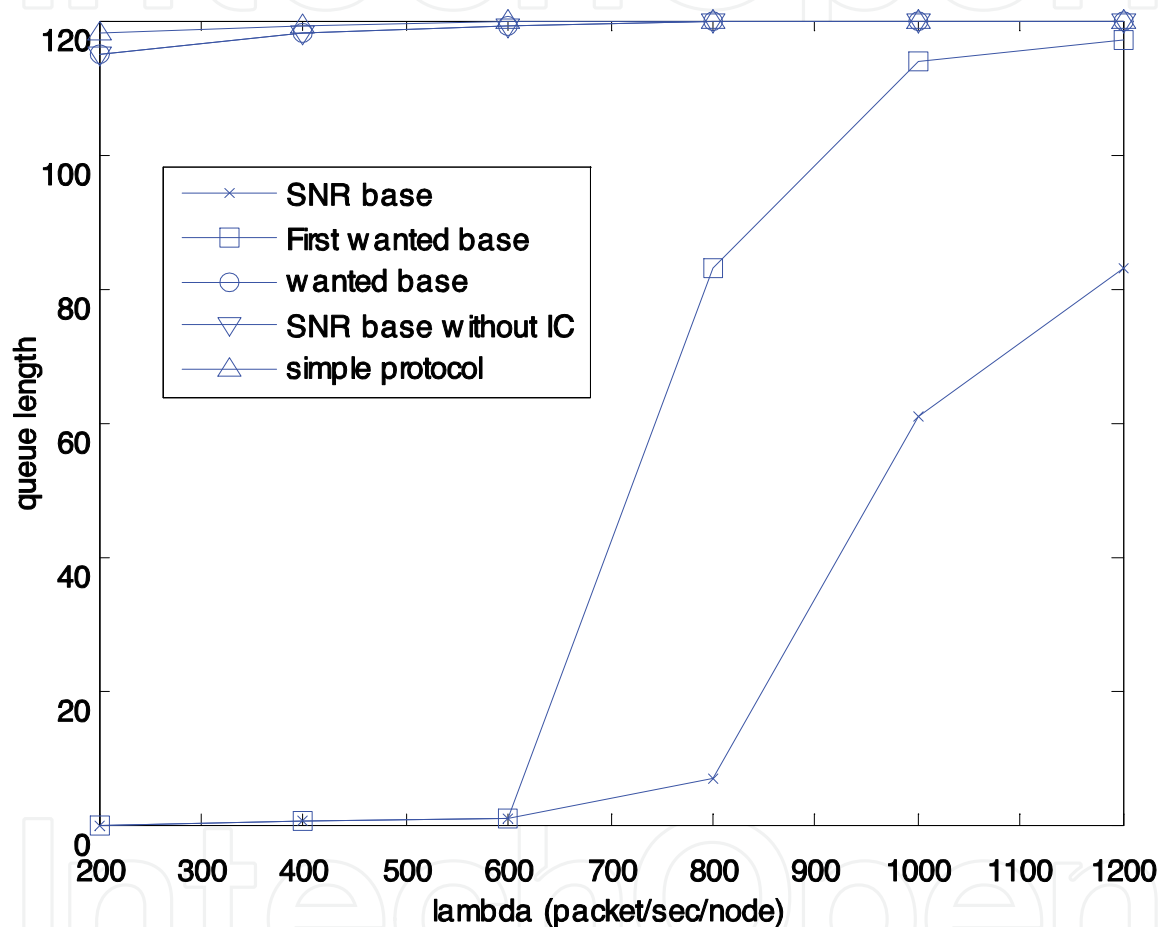
Fig. 27. Queue length versus network traffic.

5. References

Alamouti, S-M. (1998). A simple transmit diversity technique for wireless communication, *IEEE Trans Commun*, 16., pp. 1451-58

Casari, P., Levorato, M. & Zorzi M. (2008). MAC/PHY Cross-Layer Design of MIMO Ad Hoc Networks with Layered Multiuser Detection, *IEEE Transactions on Wireless Communications*, 7. 11., pp. 4596-4607

Chen, B. & Gans, M. (2005). MIMO communication in ad hoc networks, *Proceeding of IEEE VTC Conf*, Sweden, pp. 2434-38

Chen, H., Yu, F., Chan, H. & Leung, V. (2006). A novel multiple access scheme over multi-packet reception channels for wireless multimedia networks, *IEEE Trans Wireless Commun*, 6., pp. 1501-11

Choudhury, R-R., Yang, X., Ramantan, R. & Vaidya, N-H. (2006). On designing MAC protocols for wireless networks using directional antennas, *IEEE Trans Mobile Comput*, 5. pp. 477-491

Gatsis, N., Ribeiro, A. & Giannakis, G. B. (2010) Optimal resource allocation in wireless networks: algorithms and convergence, *IEEE Transactions on Wireless Communications*, submitted.

Hu, M. & Zhang, J. (2004). MIMO ad hoc networks: medium access control, saturation throughput, and optimal hop distance, *Journal of Commun Networks*, pp. 317-30

IEEE Standards Department (2007), ANSI / IEEE Standard 802.11, IEEE Press

Jafarkhani, H. (2005). *Space-Time Coding: Theory and Practice*, Cambridge University Press

Madan, R., Cui, S., Lal, S. & Goldsmith, A. (2006). Cross layer design for lifetime maximization in interference-limited wireless sensor networks, *IEEE Transactions on Wireless Communications*, 5. 11., pp. 3142-3152.

Park, M., Choi, S-H. & Nettles, S-M. (2005). Cross-layer MAC design for wireless networks using MIMO, *Proceeding of IEEE Global Commun Conf*, USA, pp. 938-942

Paularj, A., Nabar, R. & Gore, D. (2003). *Introduction to Space-Time Wireless Communication*, Cambridge University Press

Paulraj, A-J., Gore, D-A., Nabar, R-U. & Boleskei, H. (2004). An overview of MIMO communication: a key to gigabyte wireless, *Proceeding of IEEE*, 92., pp. 198-218

Ramantan, R., Redi, J., Santivanez, C., Viggins, D. & Polit S. (2005). Ad hoc networking with directional antenna: a complete system solution, *IEEE J Selected Areas Commun*, 23., pp. 496-506

Setton, E., Yoo, T., Zhu, X., Goldsmith, A. & Girod, B. (2005). Cross-layer design of adhoc networks for real-time video streaming, *IEEE Trans Wireless Commun*, 12., pp. 59-65

Sfar, S., Murch, R-H. & Letaief, K-B. (2003). Layered space-time multiuser detection over wireless uplink systems, *IEEE Trans Wireless Commun*, 2., pp. 653-668

Soleimani-Nasab, E. & Ardebilipour, M. (2009). Improve efficiency of ad hoc networks with MIMO communication and cross layer MAC design, *Procceeding of IEEE ICACT 2009*, South Korea, pp. 907-912

Sundaresan, K., Sivakumar, R., Ingram, M. & Chang, T-Y. (2004). Medium Access Control in ad-hoc networks with MIMO links: optimization consideration and algorithms, *IEEE Trans. Mobile Comput*, 3., pp. 350-65

Vang, D. & Tureli, U. (2005). Cross layer design for broadband ad hoc networks with MIMO-OFDM, *Proceeding of Signal processing Advances in Wireless Communication*, pp. 630-34

Zhang, J. & Lee, H-N. (2008). Throughput enhancement with a modified 802.11 MAC protocol with multi-user detection support, *Int J Electronics Commun*, 62., pp. 365-73

Zorzi, M., Zeidler, J., Anderson, A., Rao, B., Proakis, J., Swindlehurst, A.L., James, M. & Krishnamurthy, S. (2006). Cross-layer issues in MAC protocol design for MIMO ad hoc networks, *IEEE Wireless Commun Mag*, 13. 4., pp. 62-76
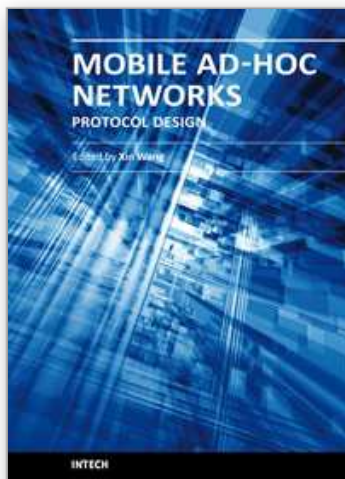
Mobile Ad-Hoc Networks: Protocol Design

Edited by Prof. Xin Wang

Being infrastructure-less and without central administration control, wireless ad-hoc networking is playing a more and more important role in extending the coverage of traditional wireless infrastructure (cellular networks, wireless LAN, etc). This book includes state-of-the-art techniques and solutions for wireless ad-hoc networks. It focuses on the following topics in ad-hoc networks: quality-of-service and video communication, routing protocol and cross-layer design. A few interesting problems about security and delay-tolerant networks are also discussed. This book is targeted to provide network engineers and researchers with design guidelines for large scale wireless ad hoc networks.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Ehsan Soleimani-Nasab, Mehrdad Ardebilipour and Mahdi Kashiha (2011). Cross–Layer Design in Wireless Ad Hoc Networks with Multiple Antennas, Mobile Ad-Hoc Networks: Protocol Design, Prof. Xin Wang (Ed.), ISBN: 978-953-307-402-3, InTech, Available from: http://www.intechopen.com/books/mobile-ad-hoc-networks-protocol-design/cross-layer-design-in-wireless-ad-hoc-networks-with-multiple-antennas

INTECH
open science | open minds