Dual Power Multiple Access with Multipacket Reception using Local CSI

Raymond Yim, *Member, IEEE*, Neelesh B. Mehta, *Senior Member, IEEE*, Andreas F. Molisch, *Fellow, IEEE*, and Jinyun Zhang, *Fellow, IEEE*

***Abstract*—Contention-based multiple access is a crucial component of many wireless systems. Multiple-packet reception (MPR) schemes that use interference cancellation techniques to receive and decode multiple packets that arrive simultaneously are known to be very efficient. However, the MPR schemes proposed in the literature require complex receivers capable of performing advanced signal processing over significant amounts of soft undecodable information received over multiple contention steps. In this paper, we show that local channel knowledge and elementary received signal strength measurements, which are available to many receivers today, can actively facilitate multipacket reception and even simplify the interference canceling receiver's design. We introduce two variants of a simple algorithm called Dual Power Multiple Access (DPMA) that use local channel knowledge to limit the receive power levels to two values that facilitate successive interference cancellation. The resulting receiver structure is markedly simpler, as it needs to process only the immediate received signal without having to store and process signals received previously. Remarkably, using a set of three feedback messages, the first variant, DPMA-Lite, achieves a stable throughput of 0.6865 packets per slot. Using four possible feedback messages, the second variant, Turbo-DPMA, achieves a stable throughput of 0.793 packets per slot, which is better than all contention algorithms known to date.**

***Index Terms*—Splitting algorithms, multiple access, contention, multiple packet reception, collision, power control, successive interference cancelation, receive signal strength indicator.**

I. Introduction

MULTIPLE access (MA) of nodes contending for a shared medium such as a wireless channel is a fundamental problem in wireless communications [1], [2]. A plethora of multiple access schemes such as ALOHA [3], carrier sensing multiple access [1], first-come-first-serve (FCFS) algorithm [4], and the part-and-try algorithm [5], [6] have been proposed and extensively studied. These schemes assume that the transmission is successful only if one packet is received by the destination at any time.

Multiple access algorithms that use Multiple Packet Reception (MPR), in which multiple packets – from single or multiple transmission attempts – are received and successfully separated by the receiver, are provably more efficient than ALOHA [7]–[10]. However, MPR often requires receivers that are capable of advanced signal processing. For example, by means of a polynomial phase-modulating sequence, the cyclostationarity of different received packets was used to color-code packets from multiple transmissions [9]. Signal separation was achieved in [11] using a rotational invariance technique. In Network-assisted Diversity Multiple Access (NDMA) [12], when k packets collide in a time slot, the network forces the transmitters to retransmit another $k-1$ times. So long as the channel changes sufficiently from one slot to another, these k consecutive transmissions allow the receiver to invert the channel matrix and recover all k collided packets. However, such channel variation can be difficult to ensure in low Doppler regimes. As can be seen, these algorithms also require receivers that can store and process significant amounts of soft information about signals received over multiple transmissions.

A more direct MPR approach uses successive interference cancellation (SIC) [13] to improve the throughput of multiple access [14]. For example, the SIC Tree Algorithm (SICTA) [14] stores soft information about the undecodable received signal whenever the receiver detects the presence of a message but cannot decode it successfully. This soft information improves the chances of decoding all the signals received thus far. When the receiver does eventually decode a packet, it subtracts its contribution from all previously stored received signals, and thereafter attempts to again decode them. The SICTA protocol is stable for arrival rates up to 0.693 packets/slot. This is substantially better than the First-come-first-serve (FCFS) binary tree algorithm, which becomes unstable when the packet arrival rate exceeds 0.487 packets/slot [4], [5]. However, like all other MPR schemes, SICTA requires the receiver to store soft information of the received signal of all previously undecodable messages. This also implies that decoding successively the possibly many packets that have collided over time can lead to long delays. Another important consideration is the feedback message size. Most protocols use a set of 2-bit feedback messages: "idle (0)", "success (1)", and "collision (e)" messages. Instead, SICTA's set of feedback messages consists of "0", "e", and, in addition, the number of packets that were finally resolved in the previous time slot. This number can be large, and requires allocation of more bits for feedback signaling.

In this paper, we propose a new and simple multiple access paradigm that uses *local channel state information (CSI)* at the transmitter to control the power received at the destination from each node (or, equivalently, the node's transmit power)

Manuscript received March 28, 2008; revised December 16, 2008; accepted February 24, 2009. The associate editor coordinating the review of this paper and approving it for publication was D. Gesbert.

R. Yim and J. Zhang are with Mitsubishi Electric Research Labs (MERL), Cambridge, MA, USA (e-mail: {yim, jzhang}@merl.com).

N. B. Mehta is with the Electrical Communication Eng. Dept., Indian Institute of Science (IISc), Bangalore, India.

A. F. Molisch is with the University of Southern California, Los Angeles, CA, USA. At the time of this work, N. B. Mehta and A. F. Molisch were with MERL, Cambridge, MA, USA. (e-mail: nbmehta@ece.iisc.ernet.in, Andreas.Molisch@ieee.org).

Part of this paper was presented at WCNC 2008, Las Vegas, NV, USA.

Digital Object Identifier 10.1109/TWC.2009.080425

so as to actively facilitate MPR. This local CSI can be easily obtained using channel reciprocity in time division duplex systems [2], and has been exploited in other multiple access schemes [15]–[17]. While the receiver still uses SIC, a key advantage of our approach is that it does not need to store signals from previous transmissions, which significantly reduces its memory and processing requirements. Instead, the receiver effectively utilizes elementary information from the total *received signal strength (power) indication (RSSI)* – a capability that is present in many commercial receivers already [18], [19]. As we show, not only is this paradigm more efficient than the best multiple access schemes known to date, but its receiver is also significantly simpler.

In particular, we propose a Dual Power Multiple Access (DPMA) algorithm in which the nodes transmit such that their received power takes one of two power levels. The key lies in setting the two power levels carefully so as to enable MPR using SIC at the receiver. As mentioned, DPMA *does not require* the receiver to store soft information of any of the undecodable signals over time – MPR is achieved simply by the use of successive interference cancellation of packets received *at the same time*. We introduce two versions of the DPMA algorithm, both of which operate over a 2-bit feedback channel. The first version, called *DPMA-Lite*, uses three feedback messages, as used in many contention algorithms, and a RSSI-capable receiver. Stability and delay analyses of both the algorithms are developed, and verified using simulations. As we shall see, depending on the dynamic range of the receiver, DPMA-Lite is stable for arrival rates up to 0.686 packets/slot; this is quite close to that of SICTA, which requires a more sophisticated receiver. We also introduce a more aggressive version called *Turbo-DPMA* that instead uses a set of four feedback messages, and is stable for a arrival rates up to 0.793 packets/slot. This is better than all the algorithms proposed in the literature to date. The proposed scheme works in systems with a single information sink. It has applications in uplink communication in cellular systems and in initial and periodic 'ranging' required for system entry and handover in WiMAX [20]. Another instance of a sink is the cluster head in a cluster of a wireless ad hoc or sensor network.

As mentioned, the use of local CSI to improve multiple access has been looked into previously. For example, in channel-aware ALOHA [15], each user transmits only if its channel gain exceeds a system-determined threshold. The Opportunistic ALOHA (O-ALOHA) protocol [16] sets the probability of transmission as a function of local channel knowledge. In [17], the time required for identifying the user with the highest priority through multiple access was substantially reduced by ensuring that the receive power levels were discrete. Splitting algorithms for capture were developed in [21]. However, all the above algorithms assume *single packet reception*, in which no packet is decoded when multiple nodes transmit simultaneously. While the use of multiple receive power levels has been considered in [22]–[24], the power levels were selected at random in each contention step and without adapting to feedback. SIC along with multiple power levels for multiple access has been considered in [25]. However, the different power levels were only used for simultaneous transmission of users with different bit error rate requirements. No feedback was assumed, and no collision resolution mechanism or RSSI measurement was used. To the best of our knowledge, DPMA is the first algorithm to use local CSI and RSSI to actively facilitate MPR and simplify receiver design.

The remainder of the paper is organized as follows. The system model is described in Sec. II. The DPMA-Lite and Turbo-DPMA algorithms are developed in Secs. III and IV, respectively. Section V contains the stable throughput and packet delay analyses of the algorithms, and is followed by simulations in Sec. VI and conclusions in Sec. VII.

II. System Model

We consider a wireless network consisting of a number of packet generating nodes that need to transmit packets to a message sink. The packets of each node are assumed to arrive for transmission at unique times. The packets are transmitted from the nodes in a time-slotted manner; it is assumed that all packets have the same size. Without loss of generality (wlog), the duration of a slot is set to unity. The channel power gain between transmitting node i and the message sink is denoted by h_i, and is assumed to be known at the transmitter (and nowhere else). This assumption is similar to the one made in channel-aware ALOHA [16], [26]. To facilitate analysis, we make the standard assumptions of a Poisson packet arrival process with a mean arrival rate (over all users) of λ, and that each new packet is generated at a unique node [4], [5], [14].

Let P_i denote the power received at the sink from node i. (We shall henceforth call it 'receive power'). The sink can decode the packet from node i successfully if its received signal to interference and noise ratio (SINR) exceeds a threshold:

$$\frac{P_i}{\sum_{j\neq i} P_j + \sigma^2} \geq \bar{\gamma}, \tag{1}$$

where σ^2 is the noise power and $\bar{\gamma} \geq 1$ is a threshold that depends on the modulation and coding used for the packet transmission [27]. Thus, a packet can be decoded successfully even when two or more users transmit simultaneously.

Consider now the specific case where every node i, which has local CSI, adjusts its transmit power so that its receive power, P_i, is either q_0 or q_1 (wlog, let $q_1 > q_0$). When two nodes each transmit a packet, one with receive power q_0 and another with q_1, *both* packets can be decoded successfully using SIC if

$$\frac{q_1}{q_0 + \sigma^2} \geq \bar{\gamma} \quad \text{and} \quad \frac{q_0}{\sigma^2} \geq \bar{\gamma}. \tag{2}$$

A checksum field in the packet enables the receiver to determine whether a packet has been decoded successfully.

The power level settings in (2) can be generalized to handle simultaneous transmissions by more than two contending users. Note that no packet can be decoded successfully if more than one user's receive power is q_1. However, if only one user's receive power is q_1, and if the power levels are set as follows:

$$q_0 = \sigma^2\bar{\gamma} \quad \text{and} \quad q_1 = \bar{\gamma}(aq_0 + \sigma^2) = q_0(a\bar{\gamma} + 1), \tag{3}$$

then the packet with receive power q_1 can be decoded so long as there are at most $\lfloor a \rfloor$ users with receive power at q_0. We

shall refer to a as the *adversary order*, as it ensures that a signal with receive power q_1 can overcome interference from up to $\lfloor a \rfloor$ users with receive power at q_0. Note that a can take any real value [17]. The assignment of receive power levels to contending nodes is described in the next section.

A. Controlling Receive Power Levels and Exploiting RSSI

It is the local channel knowledge that enables the transmitting node to control the receive power level. Each node can easily and locally compute its channel gain to the message sink by listening to a (predefined) pilot sequence that is periodically broadcast by the sink. For a target receive power P and an estimated channel gain h, a node transmits its message at power $P/|h|^2$. This technique is analogous to power control that is ubiquitous in second- and third-generation CDMA-based cellular systems. The mechanisms for enforcing discrete receive powers are the same in our case, though the motivation is subtly different. In power-controlled second-generation CDMA, it is essential that the received powers from all users are identical. In third-generation systems, several *discrete* receive power levels are foreseen (related to the fact that users with higher data rates need higher power). In our DPMA schemes, the different power levels are used for data sent at the same transmission rate.

The total receive power, specified by RSSI at the receiver, is the sum of receive powers of all received packets in a time slot. This measurement is made by many wireless receivers today. In our case, the receiver can extract useful side information from RSSI regarding the number of packets received at each of the two power levels since the receive power of each packet takes a limited set of values (q_0 and q_1). We will use this side information in the development of the DPMA algorithm in the following sections.

For clarity, we also define a quantity called the *Residual Receive Power (RRP)*, which can be derived from the RSSI after the receiver successively performs SIC. RRP is defined as the power of the received signal that remains after all decodable messages have been canceled from it. For example, if the receiver gets two packets, one at power q_1 and the other at power q_0, the RRP is on the order of the noise power, σ^2, as both packets will be successively decoded and canceled from the received signal. Consider another case in which the receiver gets three packets, one at power q_1 and two at power q_0, for $a \geq 2$. Then, it decodes the packet at q_1 successfully, and it fails to decode the remaining two packets at q_0. Therefore, the RRP is now $2q_0 + \sigma^2$. Finally, when no packet is received, the RRP is on the order of σ^2.

B. Practical Feasibility of Two Discrete Receive Power Levels

The discussion above in (2) and (3) used two power levels. Two such levels can be easily accommodated by receivers of existing systems. For example, if the minimum SINR threshold for successful decoding is $\bar{\gamma} = 10$ dB, it follows from (1) that the transmitter and receiver dynamic range should be at least 10 dB if the packet of higher received power is to be received successfully. In existing systems, the mobile station transmit power dynamic range is 35 dB in GSM systems [28] and 74 dB in third generation Wideband CDMA systems [18]. After accounting for variations in receive signal strength due to fading and near-far problem, one can reasonably assume that the receiver has about 20 dB of dynamic range. Thus modern wireless transmitters are easily capable of providing the dynamic range required to achieve the two power levels, as long as the adversary order a is not extraordinarily large.

While the proposed scheme can also be generalized to handle more than two power levels to deliver even better performance, this comes at the expense of a larger dynamic range requirement and a greater feedback overhead.

C. Relevant SIC Receiver Properties

For the case of the two receive power levels specified in (3), a SIC receiver that processes only the signal received in the current time slot exhibits the following properties, which shall be important in the algorithm development that follows.

- If only two packets are received, one with power q_0 and the other with power q_1, then both can be decoded.
- If only one packet is received with power q_0, then it can be decoded.
- If one packet is received with power q_1, then it can be decoded so long as no other packet is received with power q_1 and the number of packets with receive power q_0 does not exceed $\lfloor a \rfloor$. For example, if $a = 2.1$ (that is, $q_1 = q_0(2.1\bar{\gamma} + 1)$), and a packet A is received with power q_1 and two other packets are received with power q_0, then only packet A is received successfully.
- Otherwise, none of the received packets can be decoded.

A practical SIC receiver may cancel only $1 - \epsilon$ fraction of interference power, where $0 < \epsilon < 1$. This can be handled by increasing the power levels to $q_0 = \frac{\epsilon\sigma^2\bar{\gamma}^2 + \sigma^2\bar{\gamma}}{1 - \epsilon\bar{\gamma}^2 a}$ and $q_1 = \bar{\gamma}(aq_0 + \sigma^2)$. This leads to more stringent requirement on the dynamic range of the transmitter. It also caps the largest allowed value for a to $1/(\epsilon\bar{\gamma}^2)$. However, we will omit this non-ideality in the rest of this paper.

III. DPMA-Lite

The discussion so far has brought out the following two things: (i) controlling the receive power level of each user to two discrete values enables the receiver to decode up to two packets simultaneously, and (ii) valuable information can be derived from the RRP, and reflected in the feed back messages to better control the next stage of the contention process. Specifically, the RRP determines the receiver behavior as follows[1]:

1) RRP $< q_0 + \sigma^2$: This implies that the packets transmitted in the slot have been resolved. Hence, the receiver broadcasts the Resolved-All (RA) feedback message.
2) $q_0 + \sigma^2 \leq$ RRP $\leq q_1 + \sigma^2$: It implies that the packet with receive power at q_1 (if it was transmitted) was decoded successfully, and at least two packets were received at power q_0. Hence, the receiver broadcasts the Resolved-High (RH) feed back message.

[1]For simplicity of presentation, the RRP levels here ignore the noise power variations, which is justifiable for $\bar{\gamma} \gg 1$. The rules can be suitably modified otherwise.

TABLE I
CONTENTION SCENARIOS AND THE CORRESPONDING FEEDBACK MESSAGE FOR A DPMA-LITE RECEIVER.

Case	No. nodes at q_1	No. nodes at q_0	Success	RRP	Feed-back
a	0	0	N/A	$< q_0 + \sigma^2$	RA
b	0	1	Yes (1 pkt)	$< q_0 + \sigma^2$	RA
c	0	$[2, a\bar{\gamma} + 1]$	No	$[q_0 + \sigma^2, q_1 + \sigma^2]$	RH
d	0	$> a\bar{\gamma} + 1$	No	$> q_1 + \sigma^2$	RN
e	1	0	Yes (1 pkt)	$< q_0 + \sigma^2$	RA
f	1	1	Yes (2 pkts)	$< q_0 + \sigma^2$	RA
g	1	$[2, a]$	Yes (1 pkt)	$[q_0 + \sigma^2, q_1 + \sigma^2]$	RH
h	1	$\geq a + 1$	No	$> q_1 + \sigma^2$	RN
i	≥ 2	0	No	$> q_1 + \sigma^2$	RN
j	≥ 2	≥ 1	No	$> q_1 + \sigma^2$	RN

3) RRP $> q_1 + \sigma^2$: Finally, this case implies that the receiver could not decode any of the transmitted messages. Hence, it feeds back the Resolved-None (RN) message.

In Table I, we exhaustively list a number of scenarios that may occur at the receiver and the corresponding feedback. As can be seen, 2 feedback bits are required to send one of the three feedback messages (RA, RN, and RH).

Given this feedback, we now describe how each node behaves in subsequent time slots under DPMA-Lite. Briefly, DPMA-Lite makes the packets that have arrived in the past contend with each other (using different power levels as described below) and allows newer packets to contend only after having successfully resolved all the packets that contend. It also uses a time-limited gated access strategy [5], which limits the number of packets that contend.

A. Queuing, Gating and Contention Resolution Interval

When a new packet arrives at a user, it is stored in the user's local queue if the system is in the process of resolving the contention due to previously transmitted packets. The new packet is stored in the queue with its arrival time stamp. Consider the time slot in which the system clears the $(k-1)$-th contention. The k-th contention resolution interval (CRI) then begins at this time. Let b_k denote the number of backlogged time slots with unresolved packets at this time.

DPMA-Lite uses a time-limited gated access strategy [5], which allows only packets in a *maximum* interval of t_0 time slots to participate in the k-th CRI. That is, if b_k is smaller than t_0, then all unresolved packets (in the queues of all nodes) participate in the k-th CRI. Otherwise, only the packets with time stamps in the earliest t_0 time slots participate in the k-th CRI. The other packets remain in the queue until a future CRI. We refer to t_0 as the *gating interval* [5]. It will play an important role in optimizing the protocol's performance.

B. Formal Definition of DPMA-Lite Algorithm

For clarity, we first provide a formal definition of the DPMA-Lite algorithm and then explain the reasoning behind it. An example is also provided to illustrate its various possible steps.

To specify the algorithm, we first define the following terminology. Let $X = [x_{\min}, x_{\max})$ denote a contiguous time interval. We define the functions $H(X)$ and $L(X)$ to split the interval X into two equal-sized 'higher' and 'lower' intervals, respectively, as follows: $H(X) = [(x_{min} + x_{max})/2, x_{max})$ and $L(X) = [x_{min}, (x_{min} + x_{max})/2)$. Let U be a stack of unresolved contiguous time intervals.

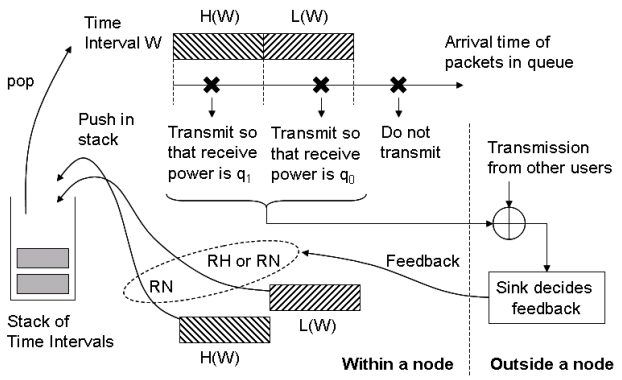

Fig. 1. A demonstration of the operations of DPMA-Lite at each node.

Let τ denote the current time slot number, and d denote the latest time stamp that was included in the previous CRI. At system initialization, we set $\tau = 1$ and $d = 0$, so that the packets with arrival time stamps in $[0, 1)$ have not entered any CRI.

At the beginning of each CRI, the algorithm computes the number of back-logged time slots $b = \tau - d$. As per the gating mechanism, the algorithm starts with $[d, d + \min(b, t_0))$ in stack U, so that all packets that arrived within this interval (of a duration of at most t_0 slots) participate in the CRI. Thereafter, we update d to $d + \min(b, t_0)$. At each time step of the CRI, all the transmitting nodes and the receiver (sink) implement the DPMA-Lite algorithm as follows. (Which part of the algorithm is implemented by whom will be clear from context.) Figure 1 demonstrates the operations of DPMA-Lite at each node except the initialization and termination steps.

- *Transmission rule:* Pop the last entered interval, W, from stack U. Every node with a packet arrival time stamp in the interval $H(W)$ transmits so that its receive power is q_1, and nodes with packet arrival time stamps in $L(W)$ transmit with receive power q_0.
- *Feedback generation:* The receiver determines its feedback based on the RRP as per Sec. III, and broadcasts it to all nodes.
- *Response to feedback:*
  1) If feedback = RA and $W \neq \emptyset$, then continue.
  2) If $W = \emptyset$ and feedback = RA, then terminate current CRI.
  3) If feedback = RH, then push $L(W)$ into stack U and continue.
  4) If feedback = RN, then push $L(W)$ and then $H(W)$ into stack U and continue.
- *At the end of a CRI:* The current time τ is updated to be the next time slot (which is also the slot in which the next CRI begins).

C. Explanation

DPMA-Lite is essentially a splitting algorithm that optimally gates/controls the number of packets that can be

TABLE II
ILLUSTRATIVE EXAMPLE FOR DPMA-LITE WHEN $a = 1$.

Slot	1	2	3	4	5	6	7
q_1	A,B,C	A	-	B	D,E	D	-
q_0	D,E	B,C	A	C	-	E	-
Feedback	RN	RN	RA	RA	RN	RA	RA

transmitted in a slot, and when once a number of packets collide in a slot, it splits the space of contending users into two parts that contend separately. Specifically, the response to different feedback messages can be explained as follows:

- A feedback of RA implies that every packet that was transmitted in the slot has been successfully resolved. Therefore, no packets remain in the interval W that was being handled by the slot. One can therefore proceed to resolve packets in the arrival time intervals that remain in the stack. If the stack is empty, then all packets in the current CRI have been resolved, and the next CRI begins in the next slot.
- A feedback of RH implies that at least two packets were received at q_0 and all packets from $H(W)$, which would have arrived with higher power q_1, have been resolved. Hence, in the next slot, only the nodes with packets in $L(W)$ contend.
- A feedback of RN implies that neither the packets in $L(W)$ nor in $H(W)$ were resolved. The algorithm therefore proceeds to resolve the intervals $L(W)$ and $H(W)$ separately, by pushing $L(W)$ and $H(W)$ on to the stack.

D. Example

We now illustrate how DPMA-Lite proceeds by means of an example, the parameters of which are artificially chosen to exercise the many scenarios defined in the algorithm. In Table II, we consider a specific scenario consisting of 5 nodes contending in a CRI, and an adversary order $a = 1$. Wlog, assume that their time stamps initially lie between 0 and 1, and algorithm starts with [0, 1) in its stack. Say, the arrival time stamps of these nodes, labeled A, B, C, D and E, are set as 0.2, 0.3, 0.4, 0.6, 0.7 respectively.

In the first slot, packets from nodes with time stamps that lie in the range $[0, 0.5)$, namely, A, B and C, arrive with receive power q_1. And, packets from remaining nodes whose time stamps lie in $[0.5, 1)$, namely, D and E, arrive with receive power q_0. This results in an RRP of $3q_1 + 2q_0 + \sigma^2$, which is larger than $q_1 + \sigma^2$. Thus the receiver feeds back *RN* to all nodes.

In slot 2, only the high power nodes of slot 1 (A, B, and C) transmit. Now A has receive power q_1 (its time stamp lies in $[0, 0.25)$), and B and C have a receive power q_0 (their time stamps lie in the range $[0.25, 0.5)$). Since $a = 1$, A cannot be decoded successfully, and the receiver feeds back RN again. In slot 3, only one node – the high power node A of slot 2 – transmits as only its time stamp lies in $[0.125, 0.25)$. It is received at power q_0 and is decoded successfully. Since RRP $= \sigma^2 < q_0 + \sigma^2$, and the receiver feeds back *RA*. In slot 4, both the low power nodes of slot 2, B and C, are resolved simultaneously as they are received at powers q_1 and q_0, respectively. (Their time stamps lie in $[0.25, 0.375)$ and $[0.375, 0.5)$, respectively). As the RRP is again less than $q_0 + \sigma^2$, another RA is fed back.

In slot 5, the low power nodes of slot 1 (D and E) transmit such that their receive power is q_1 (time stamps lie in $[0.5, 0.75)$), and no packet gets decoded. As the RRP is larger than $q_1 + \sigma^2$, the receiver feeds back a RN message. In slot 6, the high power nodes of slot 5 transmit. Now D and E are resolved simultaneously as they are received at power q_1 and q_0 respectively. (Their time stamps lie in $[0.5, 0.625)$ and $[0.625, 0.75)$, respectively). The RRP is less than $q_0 + \sigma^2$, and RA is fed back. Finally, in slot 7, the low power nodes of slot 5 transmit. However, since no node has a time stamp in $[0.75, 1)$, the slot is idle and RRP is less than $q_0 + \sigma^2$. The receiver feeds back RA, and terminates this CRI. A new CRI commences in the next slot.

IV. TURBO-DPMA

While DPMA exploits RSSI, it does not do so fully. This can be seen from the example shown in Table II. In slot 5, both D and E are received at q_1, and no packet is received at q_0. DPMA-Lite hence feeds back RN since $RRP > q_1$. This choice eventually leads to an empty slot in slot 7, which effectively lowers the maximum throughput that can be supported by the system. Being able to discriminate such a scenario can help increase the efficiency of the algorithm.

This can be done by checking whether the RRP less noise power is an integer multiple of q_1. If yes, then it is highly likely that no packet was received at q_0 because of the considerable gap that typically exists between the two power levels. For example, for $\bar{\gamma} = 10$ dB and $a = 1$, the odds that a sufficient number (10) of nodes transmitted at q_0 so as to cause the RRP to be an integral multiple of q_1 is of the order of 10^{-10}, which is small.

We now develop the Turbo-DPMA algorithm that actively exploits this fact. Before we do so, we discuss the implications of the unlikely event in which Turbo-DPMA is mistaken, i.e., it assumes that no q_0 receive power packet was received when several q_0 receive power packets were indeed received (such that the RRP less noise power is still an integral multiple of q_1). As we shall see, this scenario causes these packets to drop out of the current CRI. This has negligible impact since these packets recontend in the next CRI.

Turbo-DPMA therefore uses an additional feedback message Resolved-Low (RL) in addition to the three used by DPMA-Lite. To be specific, it generates its feedback messages from the RRP values as follows:[2]

1) $\mathbf{0} <$ RRP $< \mathbf{q_0} + \sigma^\mathbf{2}$: As in DPMA-Lite, the receiver therefore feeds back *RA*.
2) $\mathbf{q_0} + \sigma^\mathbf{2} \leq$ RRP $\leq \mathbf{q_1} + \sigma^\mathbf{2}$: As in DPMA-Lite, the receiver therefore feeds back *RH*.
3) RRP $\in \{\mathbf{mq_1} + \sigma^\mathbf{2} : \mathbf{m} \geq \mathbf{2}, \mathbf{m} \in \mathbb{Z}\}$: This implies that no packet is received at/near q_0, and the receiver cannot resolve the packets received at power q_1. The receiver therefore feeds back *RL*.
4) RRP $> \mathbf{q_1} + \sigma^\mathbf{2}$ and RRP $\notin \{\mathbf{mq_1} + \sigma^\mathbf{2} : \mathbf{m} \in \mathbb{Z}\}$: This implies that at least one message was received

[2]As in DPMA-Lite, the RRP intervals ignore the noise power variations for simplicity of presentation, which is justifiable for $\bar{\gamma} \gg 1$. The rules can be suitably modified otherwise.

with power q_1 and the receiver could not decode any of the messages. The receiver therefore feeds back *RN*.[3]

A. Formal Definition of Turbo-DPMA Algorithm

The formal definition of Turbo-DPMA algorithm is similar to that of DPMA-Lite in Sec. III-B, with the following one additional detail pertaining to the feedback message RL:

5) If feedback = RL, then push $H(W)$ into stack U and continue.

As in DPMA-Lite, in each time step of the CRI, all the transmitting nodes and the receiver (sink) implement the Turbo-DPMA algorithm. The full formal description is not repeated here due to space constraints. (As mentioned, in the unlikely event that a packet with time stamp τ is omitted in a CRI, it joins the next CRI by updating its time stamp to a uniformly chosen random value in the new interval.)

Table III exhaustively lists all the different scenarios that can be experienced at the Turbo-DPMA receiver, and the corresponding RRP and feedback message. Cases d, h, i and j are different from DPMA-Lite. In case i, Turbo-DPMA, unlike DPMA-Lite, can discern that no packet is received at q_0. In cases $d1$, $h1$ and $j1$, no packet is decoded successfully, the resulting RRP exceeds q_1 and it is not an integer multiple of q_1 after subtracting noise power. For these cases, the algorithm will feedback RN. In cases $d2$, $h2$ and $j2$, the RRP less noise power is an integer multiple of q_1. Therefore, Turbo-DPMA here incorrectly feeds back RL (i.e., it makes the wrong assumption that no packet is received at power q_0). As discussed above, the probability of these cases is extremely low.

For the example in Table II, the first four slots proceed in exactly the same manner as DPMA-Lite. However, in slot 5, Turbo-DPMA sends *RL* to the transmitters. This causes the collision resolution interval to end after slot 6 itself (unlike slot 7 for DPMA-Lite).

V. Stable Throughput and Delay Analyses of DPMA-Lite and Turbo-DPMA

A. DPMA-Lite Stable Throughput Analysis

The analysis assumes Poisson packet arrival processes, which imply that packets are uniformly distributed in the time interval. Consider the expected number of slots, L_n, required to resolve a collision involving n nodes. Clearly, when only zero or one packet is received in a slot, it takes exactly one slot to resolve the packet. Thus, $L_0 = L_1 = 1$.

When two packets are received in a slot, each packet can be received at power q_0 or q_1. The following are the four possible cases:

- *Both packets are received at q_1:* This case occurs with probability $\frac{1}{2^2}\binom{2}{0}$. The receiver feeds back *RN*. In subsequent slots, the high nodes and the low nodes contend separately and are thus resolved separately. Resolving the high nodes takes a duration L_2 because there are two packets to be resolved. Resolving the low nodes takes duration L_0, because there are no packets to be transmitted by low nodes.
- *Both packets are received at different power levels:* This case occurs with probability $\frac{1}{2^2}\binom{2}{1}$. In that case, both packets are resolved in the slot, and an *RA* message is fed back.
- *Both packets are received at q_0:* This occurs with probability $\frac{1}{2^2}\binom{2}{2}$, and the receiver feeds back *RH*.[4]

Hence, $L_2 = 1 + \frac{1}{2^2}\left(\binom{2}{0}(L_2 + L_0) + \binom{2}{1}0 + \binom{2}{2}L_2\right)$, which yields $L_2 = 2.5$.

The general case $n \geq 3$ consists of the following mutually exclusive cases:

- *$n - i$ nodes at level q_1 and i nodes at level q_0, with $0 \leq i < n-1$:* This case occurs with probability $\frac{1}{2^n}\binom{n}{i}$. It requires an average of L_{n-i} slots to resolve the packets in the high nodes and L_i to resolve the low nodes.
- *1 node at level q_1, and $n-1$ nodes at level q_0:* This occurs with probability $\frac{1}{2^n}\binom{n}{n-1}$. If the SINR for the high node is below the threshold $\bar{\gamma}$ (in other words, if $a < n-1$), then the receiver feeds back a RN message; in the subsequent timeslot, the high node is the only one allowed to transmit, and its packet is resolved. If the total power from the low nodes is below q_1, then the high node is immediately resolved, and a RH message is fed back. In either case, the $n-1$ low nodes are resolved subsequently, which requires L_{n-1} slots.
- *n nodes at level q_0:* If the total received power exceeds $q_1 + \sigma^2$ (i.e., if $a\bar{\gamma} + 1 < n$), then the receiver feeds back an RN message. In the subsequent slot meant for high nodes to contend, no transmission occurs since there are no high power nodes with messages, and the receiver feeds back RA. Subsequently, the low nodes are resolved, which takes duration L_n slots. If the total received power is lower than q_1, then the system immediately starts to resolve the low nodes.

Hence, for $n \geq 3$,

$$L_n = 1 + \frac{1}{2^n}\left(\left(\sum_{i=0}^{n-2}\binom{n}{i}(L_{n-i} + L_i)\right) + \binom{n}{n-1}(I_{\{a<n-1\}} + L_{n-1}) + I_{\{a\bar{\gamma}+1<n\}} + L_n\right), \quad (4)$$

where $I_{\{x\}}$ is the indicator function that is equal to one if x is true, and zero if x is false. After rearranging the equation, we arrive at the following recursion

$$L_n = \frac{\left(2^n + 1 + I_{\{a\bar{\gamma}+1<n\}} - nI_{\{a\geq n-1\}} + 2\sum_{i=1}^{n-1}\binom{n}{i}L_i\right)}{2^n - 2}. \quad (5)$$

When the packet arrival follows a Poisson process with mean arrival λ, and when the time interval (in units of the number of slots) to be included in a CRI is t, then the expected number of slots required to resolve a CRI is

$$R(\lambda t) = \sum_{n=0}^{\infty}\frac{(\lambda t)^n e^{-\lambda t}}{n!}L_n. \quad (6)$$

[3]Since the noise power is a random variable, in practice, the RN message should be fed back when the RRP values lie within an interval, or width of the order of σ^2, around the specified discrete values $mq_1 + \sigma^2, m \in \mathbb{Z}$.

[4]Strictly speaking, the feedback is *RN* if $a\bar{\gamma}+1 < 2$, or *RH* if $a\bar{\gamma}+1 \geq 2$. However, in all practical systems $a\bar{\gamma} \geq 1$; hence, a *RH* is always fed back.

TABLE III
THE SCENARIOS EXPERIENCED AT THE DPMA RECEIVER, AND THE CORRESPONDING FEEDBACK MESSAGE. (NOTATION USED: $S(x) = \{mx : m \in \mathbb{Z}\}$ AND $S_\sigma(x) = \{mx + \sigma^2 : m \in \mathbb{Z}\}$.)

Case	No. nodes at q_1	No. nodes at q_0	Success	RRP	Feedback
a	0	0	N/A	$< q_0 + \sigma^2$	RA
b	0	1	Yes (1 packet)	$< q_0 + \sigma^2$	RA
c	0	$[2, a\bar{\gamma} + 1]$	No	$[q_0 + \sigma^2, q_1 + \sigma^2]$	RH
$d1$	0	$> a\bar{\gamma} + 1, \notin S(a\bar{\gamma} + 1)$	No	$> q_1 + \sigma^2, \notin S_\sigma(q_1)$	RN
$d2$	0	$\geq a\bar{\gamma} + 1, \in S(a\bar{\gamma} + 1)$	No	$> q_1 + \sigma^2, \in S_\sigma(q_1)$	RL
e	1	0	Yes (1 packet)	$< q_0 + \sigma^2$	RA
f	1	1	Yes (2 packets)	$< q_0 + \sigma^2$	RA
g	1	$[2, a]$	Yes (1 packet)	$[q_0 + \sigma^2, q_1 + \sigma^2]$	RH
$h1$	1	$\geq a + 1, \notin S(a\bar{\gamma} + 1)$	No	$> q_1 + \sigma^2, \notin S_\sigma(q_1)$	RN
$h2$	1	$\geq a + 1, \in S(a\bar{\gamma} + 1)$	No	$> q_1 + \sigma^2, \in S_\sigma(q_1)$	RL
i	≥ 2	0	No	$> q_1 + \sigma^2, \in S_\sigma(q_1)$	RL
$j1$	≥ 2	$\notin S(a\bar{\gamma} + 1)$	No	$> q_1 + \sigma^2, \notin S_\sigma(q_1)$	RN
$j2$	≥ 2	$\in S(a\bar{\gamma} + 1) - \{0\}$	No	$> q_1 + \sigma^2, \in S_\sigma(q_1)$	RL

The following theorem describes the stability region of DPMA-Lite.

Theorem 1: The necessary and sufficient condition for stability is

$$\lambda < \frac{\lambda t_0}{R(\lambda t_0)}. \tag{7}$$

Proof: Let the backlog b_k be defined as the number of slots with unresolved packets in the system at the beginning of the k-th CRI. It is clear that b_k is a Markov process as b_k depends only on b_{k-1}. Due to the time-limited gated access design, all packets in the interval b_k enter the CRI when $b_k < t_0$; otherwise, only the packets in the first t_0 slots of the backlog enter the CRI. Hence, the expected number of backlogged slots in the next CRI, conditioned on b_k, is

$$\begin{aligned} \mathbb{E}[b_{k+1}|b_k < t_0] &= R(\lambda b_k), \\ \mathbb{E}[b_{k+1}|b_k \geq t_0] &= b_k - t_0 + R(\lambda t_0). \end{aligned} \tag{8}$$

For stability, we note that if b_k is a super-martingale whenever $b_k \geq t_0$, then the backlog is finite with probability one [29]. This holds true when the drift satisfies

$$\mathbb{E}[b_{k+1} - b_k | b_k \geq t_0] = R(\lambda t_0) - t_0 < 0, \tag{9}$$

which is equivalent to the condition in (7). ■

Equivalently, (7) can be restated as $\lambda < R^{-1}(t_0)/t_0$. However, doing so is not very useful because $R^{-1}(\cdot)$ is hard to obtain. (7) can be intuitively understood as follows. λt_0 is the expected number of packets entering a CRI when the maximum gating interval t_0 is used, and $R(\lambda t_0)$ is the expected number of time slots required to resolve λt_0 packets. Hence, $\lambda t_0 / R(\lambda t_0)$ is the rate at which packets are successfully decoded at the receiver when there is a significant backlog. Therefore, Theorem 1 states that stability is ensured when the arrival rate of packets into the system is less than the expected rate at which packets are decoded successfully by the receiver.

For a given adversary order a, we can numerically evaluate the stability region of the DPMA-Lite algorithm, in terms of t_0 and λ. Figure 2 shows the stability region, and shows the value of t_0 that gives the maximum value for λ. As a increases from 1 to 5, the maximum stable value of λ also increases, as expected, from 0.6517 when $1 < a < 2$, to 0.6791 when $2 < a < 3$, to 0.6854 when $3 < a < 4$, and to 0.6865 when $4 < a < 5$. The value of t_0 that leads to the maximum stable arrival rate changes from 2.476 to 2.551, 2.607, and 2.628 as a increases. This is expected since increasing a allows the receiver to resolve more cases; thus, the algorithm can become more aggressive in making more users contend.

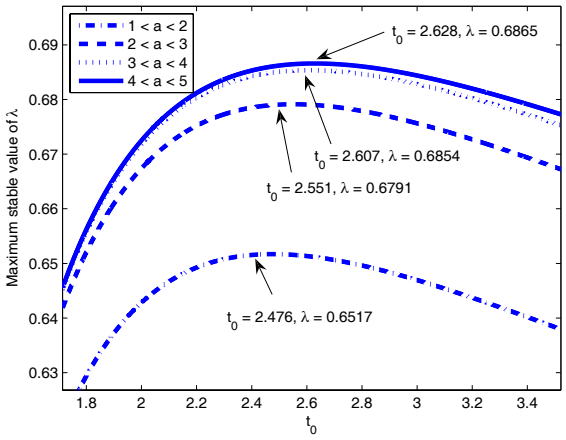

Fig. 2. The boundary of the stability region of DPMA-Lite for different values of adversary order a.

Using DPMA-Lite, we see that the maximum stable arrival rate is 0.6865 when $a \geq 4$. For an SINR threshold of $\bar{\gamma} = 10$, this implies that $q_1 \geq 41 q_0$, i.e., q_1 is 16 dB above q_0. Such a dynamic range can be readily supported by many receivers today. Even though the arrival rate of 0.6865 is marginally below the 0.693 result using the SICTA algorithm, DPMA-Lite is superior from an implementation complexity point of view since it does not require the receiver to store soft information of the received undecodable packets. Also, it only uses three feedback messages.

B. Turbo-DPMA Stable Throughput Analysis

The throughput analysis for Turbo-DPMA is very similar to that for DPMA-Lite above. Hereafter, we do not consider cases $d1$, $h1$, and $j1$, in which Turbo-DPMA incorrectly assumes that no packets were received at q_0, given that they are extremely unlikely events.

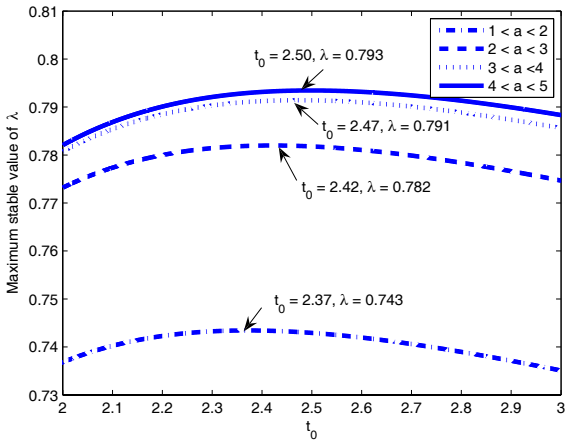

Fig. 3. The boundary of the stability region of Turbo-DPMA for different values of adversary order a.

When only zero or one packet is received in a slot, it takes exactly one slot to resolve the packet. Thus, $L_0 = L_1 = 1$. When two packets are received in a slot, the system behaves in the same way as for DPMA-Lite, except in the case that both packets are received at q_1. In that case the receiver feeds back RL (instead of RN). Since the system knows that only high nodes have packets to transmit, a duration L_2 (instead of $L_2 + L_0$ for DPMA-Lite) is required to transmit the packets. Hence, the total resolution time is

$$L_2 = 1 + \frac{1}{2^2}\left(\binom{2}{0}L_2 + \binom{2}{2}L_2\right), \tag{10}$$

which, when solved, leads to $L_2 = 2$ (which is lower than $L_2 = 2.5$ slots of DPMA-Lite).

Similarly, if $n \geq 3$ packets are transmitted, the only difference from DPMA-Lite occurs for the case in which all packets are received at level q_0. Here, RL is fed back instead of RN. Thus,

$$L_n = 1 + \frac{1}{2^n}\left(\binom{n}{0}L_n + 2\sum_{i=1}^{n-2}\binom{n}{i}L_{n-i} + \binom{n}{n-1}(I_{\{a<n-1\}} + L_{n-1}) + \binom{n}{n}L_n\right). \tag{11}$$

Rearranging the equation gives

$$L_n = \frac{1}{2^n - 2}\left(2^n - nI_{\{a\geq n-1\}} + 2\sum_{i=1}^{n-1}\binom{n}{i}L_i\right),\ n \geq 3.$$

The expected number of slots required to resolve a CRI given a collision resolution window of size t is given by (6), and the stability condition is the same as in Theorem 1.

For a given a, we can numerically evaluate the stability region of the DPMA algorithm in terms of t_0 and λ. Figure 3 shows the stability region boundary, and shows the value of t_0 that achieves it. As the adversary order, a, increases from 1 to 5, the maximum stable value of λ also increases, as expected, from 0.743 when $1 < a < 2$ to 0.793 when $4 < a < 5$. The corresponding value of t_0 that leads to the maximum stable arrival rate also increases from 2.37 to 2.50.

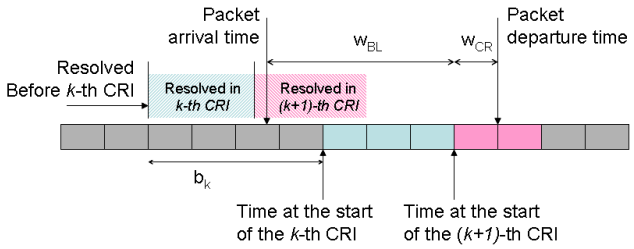

Fig. 4. Two components that contribute to packet delay: W_{BL} and W_{CR}.

C. Delay Analysis

Having analyzed the stable throughput region of the protocols, we now analyze the average delay observed by a packet that arrives in an arbitrary user's local queue. For brevity, we only show the derivation for Turbo-DPMA, with the derivation for DPMA-Lite being similar.

Consider a packet that arrives between the $(k-1)$-th and k-th CRI. Figure 4 shows the two time components that contribute to the packet delay: (i) the *backlog delay* W_{BL}, which is the time that a packet waits in a backlog before the CRI in which it is resolved begins and (ii) the *collision resolution delay* W_{CR}, which is the time during the CRI before the receiver decodes the packet successfully. We analyze these two components separately below.

1) Backlog Delay: As before, let b_k denote the number of slots backlogged at the start of the k-th CRI. In general, for an arbitrary t_0, the backlog b_k can be any positive value in the set $\mathcal{B} = \{n + mt_0 : n, m \in \mathbb{Z}\}$. For example, for $t_0 = 2.5$, a typical value for the gating interval, $\mathcal{B}$ is the set of all half integers. Since b_k is a super-martingale, b_k forms an ergodic Markov process, and we can find its steady-state distribution. To do so, we first compute below the transition probability, $p(b_{k+1}|b_k)$, of b_{k+1} given a backlog of b_k in the k-th CRI. Let $W_{b_k}(z) = \sum_{i=0}^{\infty} p(b_{k+1} = i|b_k)z^i$, denote the probability generating function of $p(b_{k+1}|b_k)$.

Let $p_i^{(n)}$ denote the probability of resolving n packets in exactly i slots, and let $Q_n(z)$ denote its probability generating function. Hence, $Q_n(z) = \sum_{i=0}^{\infty} p_i^{(n)} z^i$. Clearly, $p_0^{(n)} = 0$ for all $n \geq 0$. Since it takes exactly one slot to decode zero or one packet, we have $p_1^{(1)} = 1$ and $p_1^{(0)} = 1$, which implies that $Q_0(z) = Q_1(z) = z$.

For $n = 2$, the probability of resolving two packets in the first slot is $\frac{1}{2}$ (when both users transmit at different power levels). Otherwise, both packets need to be resolved all over again in future slots (since either RL or RH is fed back). Hence, $Q_2(z) = \frac{1}{2}z + \frac{1}{2}zQ_2(z)$, or

$$Q_2(z) = \frac{z}{2-z} = \sum_{i=1}^{\infty}\frac{1}{2^i}z^i. \tag{12}$$

For $n \geq 3$, we arrive at the following expression, which is similar to (11):

$$Q_n(z) = \frac{1}{2^n}z\left(\binom{n}{0}Q_n(z) + \sum_{i=1}^{n-2}\binom{n}{i}Q_{n-i}(z)Q_i(z) + \binom{n}{n-1}z^{I_{\{a<n-1\}}}Q_{n-1}(z) + \binom{n}{n}Q_n(z)\right). \tag{13}$$

This leads to the following $Q_n(z)$ recursion $Q_n(z) = \frac{\sum_{i=1}^{n-2}\binom{n}{i}zQ_{n-i}(z)Q_i(z)+\binom{n}{n-1}z^{1+I_{\{a\le n-1\}}}Q_{n-1}(z)}{2^n-2z}$, for $n \geq 3$.

Therefore, from the properties of a Poisson packet arrival process, the probability generating function of the backlog transition probabilities $p(b_{k+1}|b_k)$ can be written in terms of $Q_n(z)$ as:

$$W_{b_k}(z) = \begin{cases} \sum_{n=0}^{\infty} \frac{(\lambda b_k)^n e^{-\lambda b_k}}{n!} Q_n(z) & \text{if } b_k < t_0 \\ z^{b_k - t_0} \sum_{n=0}^{\infty} \frac{(\lambda t_0)^n e^{-\lambda t_0}}{n!} Q_n(z) & \text{if } b_k \geq t_0 \end{cases}. \tag{14}$$

For $b_k < t_0$, all the backlogged packets enter the k-th CRI, which implies that the start time of the next backlog interval coincides with the end time of the current backlog interval. In time b_k, the probability that n packets arrive is $\frac{(\lambda b_k)^n e^{-\lambda b_k}}{n!}$ (this follows from the Poisson arrival assumption). When $b_k \geq t_0$, the backlog packets of the $(k+1)$-th CRI are the ones that were not allowed to contend in the k-th CRI (i.e., they arrived during the $b_k - t_0$ slots that precede the k-th CRI) and over the duration of the k-th CRI. The duration of the k-th CRI depends on the number of packets that arrived over t_0 slots.

Recall that the coefficient of z^i in $W_{b_k}(z)$ equals $p(b_{k+1} = i|b_k)$. The steady state probability of the backlog duration l, denoted by $P_b(l)$, $l \in \mathcal{B}$, can now be evaluated from $p(b_{k+1} = i|b_k)$ using the global balance equation since the b_k is an ergodic Markov process [29].[5]

Given the steady state backlog duration probabilities $P_b(l)$, the expression for the expected backlog delay W_{BL} follows from the following theorem.

Theorem 2: The average backlog delay is

$$W_{BL} = \sum_{l<t_0} \tilde{P}_b(l)\frac{l}{2} + \sum_{l\geq t_0} \tilde{P}_b(l)\left(l - \frac{t_0}{2}\right), \tag{15}$$

where $\tilde{P}_b(l)$ is the probability of a packet arriving in the system when the backlog is l, and equals

$$\tilde{P}_b(l) = \frac{lP_b(l)}{\sum_{l'} l'P_b(l')}. \tag{16}$$

Proof: From the definition of W_{BL}, it follows that we do not need to consider packets that arrive after the first t_0 time slots of a backlog since these packets will be serviced only by subsequent CRIs. Therefore, for a backlog of length l, the arrival time, t, of the packet lies within $[0, \min\{l, t_0\}]$. Given the Poisson arrival assumption, it follows that t is uniformly distributed in $[0, \min\{l, t_0\}]$. The packet will have to wait for a time $l-t$ before the CRI in which it contends and is resolved begins. The average backlog delay is then $\int_0^{t_0}(l-t)\frac{1}{t_0}dt = l - \frac{t_0}{2}$, for $l \geq t_0$, and $\int_0^{l}(l-t)\frac{1}{l}dt = \frac{l}{2}$, for $l < t_0$. Hence, the result. ∎

2) Contention Resolution Delay W_{CR}: In general, the expected collision resolution delay depends on when the packet arrives in a particular collision resolution window, which is analytically difficult to compute exactly. However, this can be upper bounded by assuming that the packet of interest is the last to be resolved in a CRI. This depends only on the size of the collision resolution window. Hence,

$$W_{CR} \leq \sum_{l<t_0} \tilde{P}_b(l) \sum_{n=0}^{\infty} \frac{(\lambda l)^n e^{-\lambda l}}{n!} L_n + \left(\sum_{l\geq t_0} \tilde{P}_b(l)\right) \sum_{n=0}^{\infty} \frac{(\lambda t_0)^n e^{-\lambda t_0}}{n!} L_n. \tag{17}$$

Both the analytically evaluated and simulated values of the delay are shown in Fig. 5, and match each other well.

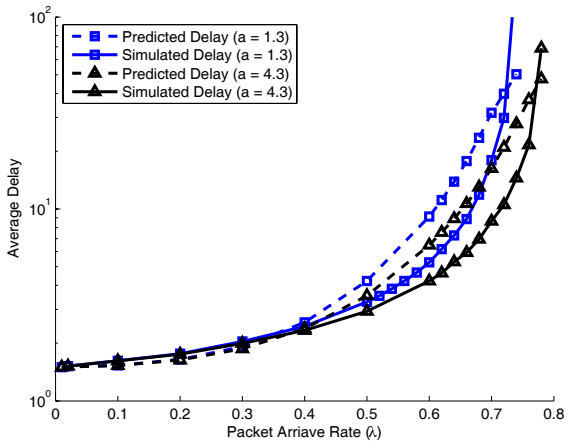

Fig. 5. Average delay of Turbo-DPMA as a function of the arrival rate, λ.

VI. Simulations

We confirm our analyses of the two DPMA algorithms through simulations. As mentioned, our simulation uses the *infinite nodes assumption* [4], where a new node is introduced for each new packets arriving at the system. The packet arrival follows a Poisson process with mean arrival rate λ, which is a simulation parameter. We assume perfect CSI at each transmitter so that the receive power of any packet is either exactly q_0 or q_1. The receiver noise power is assumed to be -100 dBm, and the decoding threshold $\bar{\gamma} = 10$ dB. Hence, $q_0 = -90$ dBm.

Figure 6 shows the average delay of the DPMA-Lite for a simulation consisting of 3×10^5 consecutive packets. The simulations use adversary orders of $a = 1.3$ and $a = 4.3$ (which set the values of q_1), and the respective optimal gating intervals of 2.476 and 2.628. In both cases, we see that the average delay remains low until the packet arrival rate, λ, approaches the maximum value for stability, which is 0.6517 for $a = 1.3$, and 0.6865 for $a = 4.3$.

Using the same simulation parameters, the average delay for Turbo-DPMA is shown in Fig. 5. Once again, the delay increases rapidly as the packet arrival rate approaches the maximum value for stability, which is 0.743 for $a = 1.3$, and 0.793 for $a = 4.3$. The figure also plots the evaluated average delay values calculated using the analytical results in Sec. V-C. We see a good match between the two. However, the match is not perfect for the following two reasons. First, the probability generating function $W_{b_k}(z)$ needs to be truncated to numerically evaluate the backlog transition probability.

[5]Since the backlogs are unbounded, the transition matrix is an infinite dimensional one. In practice, we truncate the transition matrix to a finite size to compute P_b.

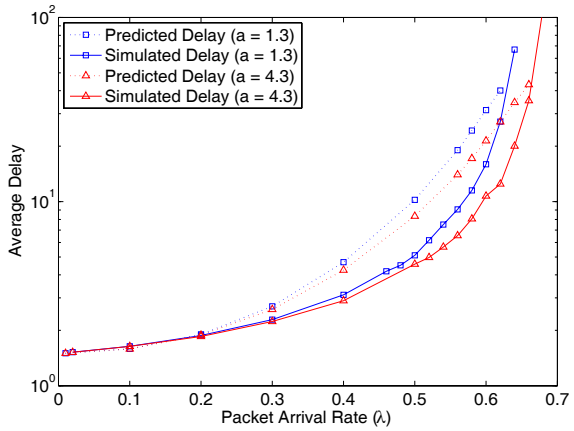

Fig. 6. Average delay of DPMA-Lite as a function of the arrival rate, λ.

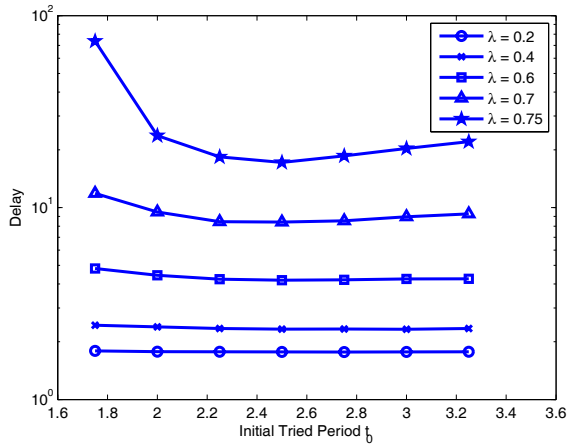

Fig. 7. Average delay of Turbo-DPMA as a function of The gating interval, t_0, for $a = 4.3$.

Second, as the arrival rate increases, the renewal time (when the all the packets in queue enter a particular CRI) becomes larger; thus, the simulation needs to cover significantly longer time intervals in order to obtain the correct average delay.

Finally, in Fig. 7, we examine the sensitivity of the Turbo-DPMA algorithm to the maximum gating interval t_0, at different arrival rate values, when $a = 4.3$. We see that when the network load is light, the average delay is fairly insensitive to the gating interval. Even when $\lambda = 0.6$, which is higher than the stable throughput of most contention algorithm, DPMA achieves an average delay of about 4.2 for a wide range of gating interval. We also see that the average delay is sensitive to t_0 only for packet arrival rates close to the stability region boundary.

VII. Conclusions

In this paper we introduced the concept of Active Multiple-Packet Reception (Active-MPR) for multiple access of several transmitters to a single common receiver that is capable of MPR. In Active-MPR, the transmitters use *local* channel state information to help improve the multiple-access performance, e.g., the stable throughput. We proposed the Dual Power Multiple Access (DPMA) algorithm that employs two discrete receive power levels to enable serial interference cancellation and, thus, successful MPR. Unlike other systems that employ MPR, DPMA did not need to store soft information at the receiver across multiple time slots.

We proposed two versions of DPMA. The more conservative variant of the algorithm, DPMA-Lite, guaranteed that all packets are received within a collision resolution interval. Using three feedback messages (the same number as in commonly known contention algorithms, but with different meaning), DPMA-Lite achieves a stable throughput of 0.6865 packets per slot for typical receiver dynamic ranges. We also proposed a more aggressive version of the algorithm, Turbo-DPMA, that uses four feedback messages. It achieves a stable throughput of 0.793 packets per slot, which is better than all previously known contention algorithms.

The algorithms have wide applicability for wireless networks, especially as more and more wireless receivers start using interference cancellation. Depending on the dynamic range of the receivers, we also envision generalizations of the algorithm to the case that three or more packets can be resolved simultaneously. While this would slightly increase the feedback overhead and the required dynamic range of the receiver, it would increase the stable throughput even more.

References

[1] L. Kleinrock and F. A. Tobagi, "Packet switching in radio channels—part I: carrier sense multipe-acess modes and their throughput-delay characteristics," *IEEE Trans. Commun.*, vol. 23, pp. 1400-1416, 1975.
[2] A. F. Molisch, *Wireless Communications*. Wiley-IEEE Press, 2005.
[3] N. Abramson, "The ALOHA system–another alternative for computer communications," in *Proc. Fall Joint Computer Conf., AFIPS*, p. 37, 1970.
[4] R. G. Gallager, "Conflict resolution in random access broadcast networks," *Proc. Air Force Office Scientific Research Workshop Commun. Theory Applications*, pp. 74-76, Sept. 1978.
[5] B. S. Tsybakov and V. A. Mikhailov, "Random multiple packet access: part-and-try algorithm," *Problemy Peredachi Infomatsii*, vol. 16, pp. 65-79, Oct.-Dec. 1980.
[6] M. Paterakis and P. Papantoni-Kazakos, "A simple window randomaccess algorithm with advantageous properties," *IEEE Trans. Inform. Theory*, vol. 35, pp. 1124-1130, Sept. 1989.
[7] S. Chez, S. Verdu, and S. Schwartz, "Stability properties of slotted Aloha with multipacket reception capability," *IEEE Trans. Automat. Contr.*, vol. 33, pp. 640-649, July 1988.
[8] L. Tong, Q. Zhao, and G. Mergen, "Multipacket reception in random access wireless networks: from signal processing to optimal medium access control," *IEEE Commun. Mag.*, pp. 108-112, Nov. 2001.
[9] A. G. Orozco-Lugo, M. M. Lara, D. C. McLernon, and H. J. Lemus, "Multiple packet reception in wireles ad hoc networks using polynomial phase-modulating sequences," *IEEE Trans. Signal Processing*, vol. 51, pp. 2003-2110, Aug. 2003.
[10] V. Naware, G. Mergen, and L. Tong, "Stability and delay of finite-user slotted aloha with multipacket reception," *IEEE Trans. Inform. Theory*, vol. 51, pp. 2636-2656, July 2005.
[11] R. Zhang, N. D. Sidiropoulos, and M. K. Tsatsanis, "Collision resolution in packet radio networks using rotational invariance techniques," *IEEE Trans. Commun.*, vol. 50, pp. 14–155, Jan. 2002.
[12] M. K. Tsatsanis, R. Zhang, and S. Banerjee, "Network-assisted diversity for random access wirless networks," *IEEE Trans. Signal Processing*, vol. 48, pp. 702-711, Mar. 2000.
[13] S. Verdu, *Multiuser Detection*. New York: Cambridge University Press, 1998.
[14] Y. Yu and G. B. Giannakis, "SICTA: a 0.693 contention tree algorithm using successive interference cancellation," in *Proc. IEEE INFOCOM*, pp. 1908-1916, Mar. 2005.
[15] X. Qin and R. Berry, "Exploiting multiuser diversity in wireless ALOHA networks," in *Proc. Allerton Conf. Commun., Control Computing*, Oct. 2001.
[16] S. Adireddy and L. Tong, "Exploiting decentralized channel state information for random access," *IEEE Trans. Inform. Theory*, vol. 51, pp. 537-561, Feb. 2005.

[17] R. Yim, N. B. Mehta, and A. F. Molisch, "Best node selction through distributed fast variable power multiple access," in *Proc. IEEE Int. Conf. Commun.*, pp. 5028-5032, May 2008.
[18] "Technical specification group radio access network; user equipment (UE) radio transmission and reception (FDD)," Tech. Rep. 25.101 (v7.7.0), 3rd Generation Partnership Project (3GPP), 2007.
[19] "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: Amendment 4: Enhancements for higher throughput," Tech. Rep. 20.5.5.7 (Draft version D2.05), IEEE 802.11 Working Group, July 2007.
[20] "Part 16: Air interface for broadband wireless access systems," Tech. Rep. Revision 2 Draft version D8, IEEE 802.16 Working Group, Dec. 2008.
[21] M. Sidi and I. Cidon, "Splitting protocols in presence of capture," *IEEE Trans. Inform. Theory*, vol. IT-31, pp. 295-301, Mar. 1985.
[22] J. J. Metzner, "On improving utilization in aloha networks," *IEEE Trans. Commun.*, vol. COM-24, pp. 447-448, Apr. 1976.
[23] W. Luo and A. Ephremides, "Power levels and packet lengths in random multiple access," *IEEE Trans. Inform. Theory*, vol. 48, pp. 46-58, Jan. 2002.
[24] J. Luo and A. Ephremides, "Power levels and packet lengths in random multiple access with multiple-packet reception capability," *IEEE Trans. Inform. Theory*, vol. 52, pp. 414-420, Feb. 2006.
[25] G. Mazzini, "Power division multiple access," in *Proc. IEEE Int. Conf. Universal Personal Commun.*, pp. 543-546, Oct. 1998.
[26] G. Ganesan and Y. Li, "Channel aware aloha with imperfect CSI," in *Proc. Globecom*, 2006.
[27] S. Cui, A. J. Goldsmith, and A. Bahai, "Energy-constrained modulation optimization," *IEEE Trans. Wireless Commun.*, vol. 4, pp. 2349–2360, 2005.
[28] "Technical specification group GSM/EDGE, radio access network; radio transmission and reception," Tech. Rep. 45.005 (v7.9.0), 3rd Generation Partnership Project (3GPP), 2007.
[29] G. Grimmett and D. Stirzaker, *Probability and Random Processes*. Oxford University Press, 3rd ed., 2001.

Raymond Yim (S'00-M'07) received his B.Eng. and M.Eng degrees in electrical and computer engineering from McGill University in Canada, and Ph.D. from Harvard University. In 2006-2007, he was an assistant professor in electrical and computer engineering at F. W. Olin College of Engineering, Needham, MA. Since 2008, he has been a research scientist at the Mitsubishi Electric Research Laboratories, Cambridge, MA, and has been involved in WiMAX development. His research interests include the design and analysis of cross-layered architectures and protocols for wireless communication networks, multiple access protocols, link adaptation, cooperative communications, smart antenna systems, interference management and intelligent transportation systems.

Neelesh B. Mehta (S'98-M'01-SM'06) received his Bachelor of Technology degree in Electronics and Communications Engineering from the Indian Institute of Technology, Madras in 1996, and his M.S. and Ph.D. degrees in Electrical Engineering from the California Institute of Technology, Pasadena, CA in 1997 and 2001, respectively. He was a visiting graduate student researcher at Stanford University in 1999 and 2000. He is now an Assistant Professor at the Dept. of Electrical Communication Engineering, Indian Institute of Science (IISc), Bangalore, India. He has worked as a research scientist in the Wireless Systems Research group in AT&T Laboratories, Middletown, NJ from 2001 to 2002; as a Staff Scientist at Broadcom Corp., Matawan, NJ, from 2002 to 2003; and as a Principal Member of Technical Staff at the Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, USA, from 2003 to 2007.

His research includes work on link adaptation, multiple access protocols, WCDMA downlinks, system-level performance analysis of cellular systems, MIMO and antenna selection, and cooperative communications. He is also actively involved in radio access network physical layer (RAN1) standardization activities in 3GPP. He is a TPC co-chair for the Transmission technologies track of VTC 2009 (Fall) and the Frontiers of Networking and Communications symposium of Chinacom 2008. He is a TPC member of several conferences such as Globecom, ICC, WCNC, WISARD, and VTC. He is on the editorial board of the IEEE Transactions on Wireless Communications, and is an executive committee member of the Bangalore chapter of the IEEE Signal Processing Society.

Andreas F. Molisch (S'89, M'95, SM'00, F'05) received the Dipl. Ing., Dr. techn., and habilitation degrees from the Technical University Vienna (Austria) in 1990, 1994, and 1999, respectively. From 1991 to 2000, he was with the TU Vienna, becoming an associate professor there in 1999. From 2000-2002, he was with the Wireless Systems Research Department at AT&T (Bell) Laboratories Research in Middletown, NJ. From 2002-2008, he was with Mitsubishi Electric Research Labs, Cambridge, MA, USA, most recently as Distinguished Member of Technical Staff and Chief Wireless Standards Architect. Concurrently he was also Professor and Chairholder for radio systems at Lund University, Sweden. Since 2009, he is Professor of Electrical Engineering at the University of Southern California, Los Angeles, CA, USA.

Dr. Molisch has done research in the areas of SAW filters, radiative transfer in atomic vapors, atomic line filters, smart antennas, and wideband systems. His current research interests are measurement and modeling of mobile radio channels, UWB, cooperative communications, and MIMO systems. Dr. Molisch has authored, co-authored or edited four books (among them the textbook *Wireless Communications*, Wiley-IEEE Press), eleven book chapters, more than 110 journal papers, and numerous conference contributions, as well as more than 70 patents.

Dr. Molisch is an editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and co-editor of special issues of several journals. He has been member of numerous TPCs, vice chair of the TPC of VTC 2005 spring, general chair of ICUWB 2006, TPC co-chair of the wireless symposium of Globecomm 2007, TPC chair of Chinacom2007, and general chair of Chinacom 2008. He has participated in the European research initiatives "COST 231," "COST 259," and "COST273," where he was chairman of the MIMO channel working group, he was chairman of the IEEE 802.15.4a channel model standardization group. From 2005-2008, he was also chairman of Commission C (signals and systems) of URSI (International Union of Radio Scientists), and since 2009, he is the Chair of the Radio Communications Committee of the IEEE Communications Society. Dr. Molisch is a Fellow of the IEEE, a Fellow of the IET, an IEEE Distinguished Lecturer, and recipient of several awards.

Jinyun Zhang received her B.Sc. degree in radio electronics from Tsinghua University, Beijing, China in 1970. Following her graduation, she was with Tsinghua University until 1985. She received her Ph.D. degree in electrical engineering from University of Ottawa, Canada in 1991. Dr. Zhang then joined Nortel Networks, where she held various management positions and engineering positions of increasing responsibility in the areas of digital signal processing, wireless communication and optical networks. Since 2001, Dr. Zhang has been the Manager of the Digital Communications & Networking Group at Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, USA. Currently, she is leading various new broadband wireless communications and networking research projects that include UWB, ZigBee ad hoc networking, MIMO, broadband multimedia home networking, wireless sensor networks, high speed WLAN, cooperative communications, WiMAX and next generation mobile communications systems.

Dr. Zhang has authored and co-authored more than 130 publications, invented and co-invented more than 100 patents and patent applications, and made numerous contributions to international wireless communications standards. Dr. Zhang is a Fellow of the IEEE and a member of the IEEE AP, BT, COMM, IT, ITS, LEO, SP, and VT Socieites. She serves as an Associate Editor of IEEE TRANSACTIONS ON BROADCASTING, an AdCom member of IEEE Broadcasting Technology Society, and has served as a Technical Program Committee member for various IEEE conferences.