OCP: Opportunistic Carrier Prediction for Wireless Networks

Chun-cheng Chen and Nitin Vaidya

University of Illinois at Urbana-Champaign

***Abstract*—In this paper, we propose Opportunistic Carrier Prediction (OCP) that jointly addresses exposed terminal and hidden terminal problems in wireless networks. OCP is based on the rationale that past interference information can be a good indicator for the outcome of future packet delivery. Therefore, each OCP sender maintains a summary of past interference information and opportunistically accesses the channel when it is confident that the packet transmission will be successful and cause no collision to other flows. To realize OCP, we propose (1) a novel data structure for each sender to summarize the interference information and (2) physical layer preemptive decoding scheme for each sender to collect the identities of the interferers. Through extensive evaluation, we show that OCP improves the system throughput by up to 170%, packet delivery success ratio by up to 400% in random topologies, while almost removing starvation in many settings.**

I. Introduction

Wireless medium access control is one of the most important research topics in the past decades. Since wireless channel is a shared medium, two nearby nodes accessing the wireless medium may cause interference to each other. Take Figure 1 for example, when node D sends a packet to F and A sends a packet to B simultaneously, F cannot receive D's packet correctly due to A's ongoing transmission in the proximity. IEEE 802.11 DCF [10], probably the most popular carrier sense multiple access (CSMA) wireless MAC protocol, adopts the carrier sensing mechanism so that a node transmits a data packet only if the sensed signal before the transmission is below a certain threshold called carrier sense threshold. In the above example, when A is transmitting, D will sense A's signal and wait until A finishes its transmission. Similarly, when flow D→F is active, A will remain silent. Thus, the transmission from D to F is safely protected from A's interference in IEEE 802.11 DCF.

Carrier sensing, however, does not always address the medium access problem properly. For example, although flows D→F and A→B in Figure 1 can not be reliable simultaneously, flows A→B and D→E *can* be simultaneously reliable since B and E are far away from the interference source D and A, respectively. If we want to protect flow D→F from the interference from A by allowing A and D to carrier sense each other, we cannot but have to sacrifice the concurrent transmission of A→B and D→E. In fact, it is also possible that interferers may not be in the proximity of wireless transmitters or that there could be obstacles separating transmitters and interferers (nodes A and C for example). In these scenarios, the interferers are hidden from the sender nodes, reducing the effectiveness of carrier sensing. Obviously, decreasing the carrier sense threshold improves the chances of detecting interferers farther away with the cost of silencing more flows in the proximity that could potentially be concurrent. On the other hand, increasing the carrier sense threshold allows more nearby flows to be concurrently active, while less number of interferers are silenced. Despite the fact that many research efforts [16], [17], [21], [23], [24] have been spent on tuning the carrier sense threshold to maximize the spatial reuse, the problem itself remains open.

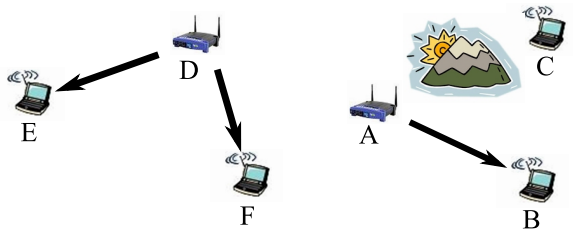

Fig. 1. Illustration of exposed terminal and hidden terminal problems in wireless networks. A and D are two exposed senders. C is a hidden interferer from A.

In this paper, we propose Opportunistic Carrier Prediction (OCP), a novel approach to allow each wireless sender to opportunistically access the medium. OCP's rationale is based on the observation that interference from the past can be a good indicator for the outcome of future packet delivery. Therefore, each sender maintains an empirical summary of *interference relationship* (who interferes my receiver and who is interfered by me) in the proximity. When the sender overhears that an interferer is in transmission or a flow that will be interfered by the sender's transmission is active, it defers its transmission until both the interfering sender and the interfered flow finish their transmissions.

To achieve the goal of OCP, we have to address the following challenges: First, how can a sender infer who is interfering its receiver and who is interfered by its transmission? Second, since each sender makes the medium access decision based on what it overhears on the channel, how can a sender efficiently extract what's going on from the wireless medium and update its channel access decision in a timely manner. Third, how do we ensure each sender correctly decodes the overheard information even in high network contention level with relatively low overhead? Finally, what is the channel access scheme if the sender does not overhear anything on the wireless medium?

In OCP, each sender infers the interference relationship by relating the overheard information on the channel to its receiver feedback of whether the previously sent packets are

correctly received. Senders further exchange information to complete the interference relationship. In order for a sender to efficiently extract on-going transmissions from the wireless medium, we insert a few bits right after the physical layer preamble. These bits are used as the flow identity that consists of sender/receiver identity pair so that after decoding the overheard packet preamble the sender can immediately extract the on-going flow information. Further, these bits are modulated using the most robust scheme available so that they can be correctly decoded in the presence of high level interference. In this paper, we assume a new physical layer decoding technique called *pre-emptive reception* that works as follows. The receiver first decodes the packet up to the receiver's identity. If the packet is destined to the receiver, it decodes the rest of the packet. Otherwise, it withdraws from the reception state. The benefit of pre-emptive reception is that once the sender finishes overhearing the on-going flow identity, it can switch to capture other flow transmissions that arise later, thereby collecting more information from the wireless medium. In case that the sender does not overhear on-going flow information from the channel, the standard carrier sensing multiple access (CSMA) is adopted. Therefore, OCP can be applied in conjunction with existing algorithms [16], [17], [21], [23], [24] that tune carrier sense threshold to optimality.

In summary, our contribution of the paper is: First, we propose a novel wireless medium access protocol (OCP) for each sender to dynamically learn from the history and infer the interferers' identities in the proximity to address both exposed and hidden terminal problems that have been long-haunted for decades. Second, although a technique similar to pre-emptive reception had been proposed by previous work [4], we believe we are the first to utilize this technique to empirically infer the interference relationship in wireless networks. Third, prior work CMAP [20] argued that carrier sensing is too conservative and proposed a medium access scheme that purely relies on the conflict map (or interference relationship). We show that simply relying on the interference relationship and blindly turning off carrier sensing does not help improve the throughput in the network with high contention. In particular, we will see that such scheme may degrade the throughput by up to 71% in random topologies. Finally, through extensive simulations, we show that OCP improves the system throughput over CSMA in random topologies of various contention levels by up to 170% and improves the packet delivery success ratio by up to 400%, while almost removing starvation in many settings.

The rest of the paper is organized as follows. We describe the most related work CMAP [20] in Section II. We further show in Section III that the partial packet recovery (PPR) [11] technique adopted by CMAP cannot decode the packet header/receiver in a highly contented network. We present OCP in Section IV and report the evaluation results in Section V. We compare OCP with various related works in the literature in Section VI, and conclude in Section VII.

II. CMAP Description

CMAP [20] argued that carrier sensing is too conservative and proposed to turn off carrier sensing and let each node access the medium based on the conflict map (interference relationship). They apply the partial packet recovery (PPR) technique [11] to empirically build the conflict map in each node's neighborhood. In particular, they append trailer and post-amble to each packet payload and include the flow identity into both header and trailer of each packet, as shown in Figure 2, based on the observation that when a collision happens, the header and trailer of the two colliding packets are usually intact and can be correctly decoded. Take Figure 1 for example, if A→B and D→F are active at the same time, although the two packets are colliding with each other, F can still decode the packet header sent by A and packet trailer sent by D to infer that flow A→B is interfering flow D→F. Once F infers such interference relationship, it publishes such information to sender D and all other neighboring nodes, thereby establishing the conflict map in the network. When next time A→B is active, D will defer its transmission to F until A→B is finished. Note that in such scheme, it is the receiver who infers the interference relationship. The sender accesses the medium based on the conflict map collected from its receiver and all other neighboring nodes.

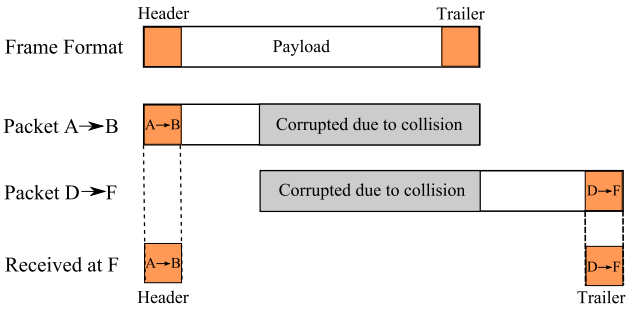

Fig. 2. F can decode the header sent by A and trailer sent by D to infer that A→B interferes D→F

III. Why CMAP Does Not Work In General

Although CMAP's observation in Figure 2 is true for a simple two-packet collision, we argue that this is generally not true in a bigger network. Consider a network with 10 or 20 flows. When carrier sensing is turned off, all nodes could send out packets and the collision will likely consist of complicated overlapping of packets in the air. The interferer's header/trailer might also be interfered by a 2nd, 3rd interferer, and so on[1]. Thus, whether their claim holds in general needs more justification. In particular, we want to know how likely the header and trailer can be successfully decoded when carrier sensing is turned off under high network contention level.

Before answering the above question, we categorize a packet collision into two groups: (1) Collision In the Beginning (CIB): the interference level is too high for the receiver to even start receiving the packet. For example, the collision at F in Figure 2 belongs to CIB. (2) Collision In the Middle (CIM): during the middle of receiving the packet, the interference from other nodes causes the receiver to drop the currently receiving packet. For example, if we reverse the transmission order of D→F and A→B in Figure 2, packet collision still happens at F, but it belongs to CIM now. Note that if the packet collision can be categorized into both CIB and CIM, we give preference to CIB. The reason will be clear later in the discussion.

[1]In their scheme, the header and trailer are modulated using the same rate as the payload.

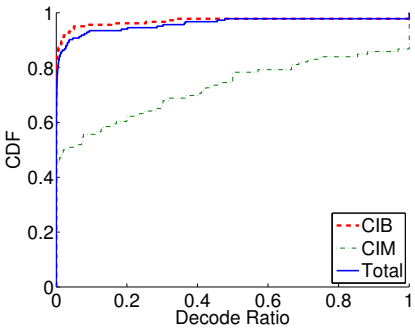

Fig. 3. CDF of the per-flow decode ratio for 10 random 20-flow topologies

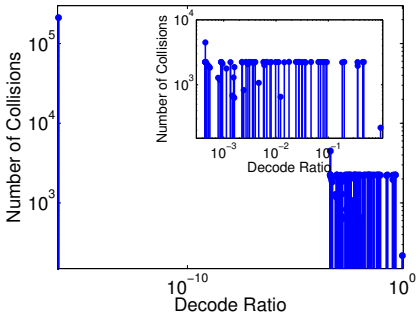

Fig. 4. Number of collisions for each decode ratio for 10 random 20-flow topologies

We implemented a more realistic physical layer capture model [15] in ns-2 simulator by considering the interference propagated from all other nodes. Whether a packet can be captured/received depends on the modulation scheme (transmission data rate) and the corresponding SINR value. We randomly generate 10 topologies, each with 20 distinct backlogged flows in a 1000m x 1000m area. More detailed setting can be found in Section V.

Although CMAP assumes that it is the header/trailer *closet* to a packet collision contain the interferer information and can be decoded, in the simulation we only require that the header/trailer of *any* packet overlapping a collided packet be decoded in order to identify the interferer. If the receiver can decode both the interferer's identity and the sender's identity in a packet collision, we say the collision is *decodable*. The ratio of the number of decodable collisions to the number of total collisions is called the *decode ratio*. We ask the question that under such network contention as described above, can we still decode the interferer information and infer the conflict relationship?

In Figure 3, we plot the cumulative distribution function (CDF) of the decode ratio for each flow in the 10 random 20-flow topologies. For CIB, the decode ratio is mostly under 46%. What's worse, more than 80% of the flows have decode ratio 0%. For the case of CIM, 45% of the flows have decode ratio 0%. Although 15% of the CIM flows have 100% decode ratio, the majority of the collisions are categorized into CIB. As a result, when combining CIB and CIM, the overall CDF of the decode ratio is not much different from that of CIB. We note here that for CIM, the header of the sender is likely to be correctly decoded and the receiver only needs to recover the trailer to decode the interferer's identity. This explains why the decode ratio for CIM is higher than that for CIB.

Since there are still tiny portion (2%) of flows with good decode ratios as shown in Figure 3, one question is whether this may actually help the flows suffering from hidden/exposed terminal problem. In Figure 4, we plot the number of collisions for each of the decode ratios over all of the 10 random topologies. As we can see (note the log-scale of x-y axes), the majority of the collisions have decode ratio 0%, therefore can not rely on partial packet recovery (PPR) to decode the header/trailer to identify the interferer's identify. What's worse, those flows with high decode ratio (therefore can apply PPR to recover the header/trailer) do not suffer from severe packet collisions.

The following points summarize our findings: First, CMAP only searches for the closest decodable header/trailer to identify who is the interferer. In reality, when there are multiple interferers, the interferer's header/trailer may also be interfered by other nodes, the true interferer which contributes the most to a collision may not be the one that is decoded by CMAP. The above experiments ignore such miscalculation, and it is clear to see that header/trailer still does not usually survive in a packet collision. Second, CMAP modulates the header and trailer using the same rate as the payload. As we have seen from the above study, such scheme does not help decode the header/trailer when multiple interferers come into play. One simple fix is to modulate the header/trailer using the lower, more robust date rate. A back-of-the-envelope calculation, however, shows that the incurred overhead (24-byte trailer plus 24-byte postamble as proposed in CMAP), when transmitted at 1 Mbps, consists of more than 35% of a regular data transmission at 11 Mbps. Third, since the majority of the collisions can be categorized into CIB, if we can address the CIB well, most of the collisions could be avoided. Further, since for CIB the interferers' identities have already been transmitted over the air before the interfered packet being sent, it is not necessary to place the responsibility at the receiver to decode both the header and trailer of a packet collision[2] in order to infer the interference relationship.

IV. Opportunistic Carrier Prediction

The analysis in Section III sheds light on the design of OCP. Since for most of the collisions the interferers' identities have been transmitted over the air before the interfered packet is sent, the sender can make more prudent decisions for channel access by carefully observing what's going on in the air. The main idea of OCP is to build a mapping between the overheard flow information at the *sender* side and the corresponding packet delivery success ratio (SR). In order to build the mapping, each sender in the network tries to overhear packets in the air and extracts the information of what flows (transmitter-receiver pairs) are active. All the *currently active* flows overheard by a sender are used to represent the current *channel status*. Each sender builds the mapping by relating the channel status to the SR based on its receiver feedback of whether the previously sent packets are correctly received or not. We summarize OCP as follows:

[2]Correctly decoding the packet already requires much computation effort and we believe the receiver should not be made unnecessarily complicated.

- Each node is set to promiscuous mode and overhears ongoing transmissions continually (§IV-A). Before a node transmits a data packet, all the currently active flows that are overheard are recorded as the channel status identification (CSID).
- After the node transmits the data packet, it updates the success ratio (SR) of the corresponding CSID based on whether the data is received by the receiver or not (§IV-B).
- Each sender also periodically broadcasts the identities of its interferers so that when the interferers receive the packet and realize they are interfering some flow, they will yield to the flow when it is active (§IV-C).
- Each sender node continually overhears currently active flows and updates its CSID accordingly. The sender uses the CSID to consult the CSID-SR mapping before transmitting the data. The sender node accesses the channel only when no flows will be interfered and the CSID corresponds to a high success ratio, say, larger than 50% (§IV-D).

A. Pre-emptive Reception and CSID Collection

In traditional wireless reception process, a receiving node always finishes receiving the entire packet even though the packet may be destined to other nodes. Such *non-preemptive* reception is currently implemented in most of the wireless receivers. However, in OCP, we propose that physical layer supports a *pre-emptive* reception capability. The idea is that when the node receives the packet up to the receiver's identity in the packet header, depending on whether the packet is destined to the receiver, it decides to receive the rest of the packet or not. If the packet is destined to the receiver node, it receives the rest. Otherwise, it withdraws from the reception state. Such pre-emptive reception technique is mainly used for each sender node to more efficiently overhear the ongoing transmissions in the air. Once the sender decodes the sender/receiver identities in the overheard packet, it can switch to capture other ensuing flow transmissions, thereby collecting more information in the air.

One approach for preemptive reception is for physical layer to simply decode the bits all the way up to the receiver's MAC address in the header, but this inevitably forces receiver to also decode other fields in the PLCP and MAC header. A better approach for preemptive reception could be inserting a few bits right after the PLCP preamble serving as the sender/receiver identity. In this case, nodes will need to negotiate to ensure each one has distinct identity in its two-hop neighborhood. Another approach could be simply moving the MAC address to right after the PLCP preamble. Since MAC addresses are distinct, there is no need for negotiation. In this paper, we adopt the latter approach. The detailed frame format is shown in Figure 5. By overhearing the *TransmitterID* and *ReceiverID*, the node knows what are the currently active flows in the air. Note that *Length* is used to indicate how long the overheard flow will last. By receiving these three pieces of information, each overhearing node knows exactly which flows will be active until when. All the three fields are modulated using the most robust scheme available so that they can be more easily captured along with existing interference.

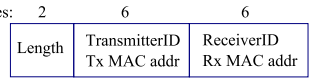

Fig. 5. Frame format inserted after the physical layer preamble

B. CSID-SR Mapping

Before a sender transmits DATA to the receiver, it records all the currently overheard flows. The sender concatenates the recorded flow IDs, each consisting of (*TransmitterID*, *ReceiverID*) pair, to represent the channel status (CSID) before the transmission[3]. Once the DATA is sent out, the sender waits for the ACK to update the packet delivery success ratio (SR) of the corresponding CSID. Note that the ACK may be lost even though the DATA is correctly received at the receiver. To avoid such false negative events, we redefine the ACK so that each ACK selectively acknowledges the previously received packets, which can be easily implemented using a simple bitmap.

To implement the CSID-SR mapping, we maintain a list of SR records (*numSuc, numFail, updTime*) containing the number of successful transmissions, number of failed transmissions, and the last time the record was updated. Note that a node may send packets to multiple receivers, and the packet delivery success ratio may be different for different receiver even with the same CSID. Take Figure 1 for example, when A→B is active, the corresponding CSID for both flow D→E and D→F at sender D is A→B. But, as we have known earlier, the same CSID (A→B) would give totally different prediction result at sender D for the two flows D→E and D→F. Thus, each SR record must be indexed by (*CSID, Rcvr*) where *Rcvr* is the receiver of the corresponding flow. When the transmitter receives the ACK (does not receive the ACK for the entire bitmap window), it increments the *numSuc* (*numFail*) field. The packet delivery success ratio can be easily derived from *numSuc* and *numFail*. Every time the CSID-SR mapping is consulted, we require that *numSuc* + *numFail* > 1; otherwise, the channel is considered idle due to insufficient number of data points.

A node may temporarily move away from its sender or the channel quality may occasionally be bad, causing the SR to drop to a low value and preventing the sender from accessing the channel ever again. To address such transient events, we must age out the stale data so that senders can intermittently poll the medium and access the channel when channel quality becomes good. Each time the CSID-SR mapping is accessed, we age out the corresponding SR record by multiplying the *numSuc* and *numFail* by the aging factor α,

$$\alpha = \begin{cases} 1 - \frac{t - updTime}{T_{window}} & \text{if } t - \textit{updTime} < T_{window} \\ 0 & \text{Otherwise} \end{cases} \quad (1)$$

In our implementation, we set T_{window} to be 5 seconds.

C. Handling Dominating Interferers

From the CSID-SR mapping, each node can easily infer what are the interferers. For example, flow 0→1 in the asymmetric two-flow topology shown in Figure 6 is suffering from the interference from node 2. Flow 2→3, on the other hand, always succeeds in packet transmission. When node 0 examines the CSID-SR mapping, it will find that the success ratio of (2→3, 1)

[3]We do not distinguish the order of flow IDs if multiple flows are overheard.

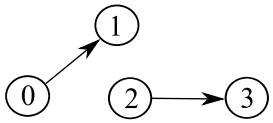

Fig. 6. Flow 0→1 suffers from the interference from node 2; while node 2 always sends packet to 3 successfully. Node 0 will inform node 2 that it's being interfered.

is low (less than 50%) and identify 2 as the interferer. After each transmitter node infers who are the interferers from the CSID-SR mapping, it periodically sends out packets using the most robust modulation to tell its neighbors who are the interferers. The broadcast packet contains a list of (*interfererID, TxID, RxID*) where *interfeferID* is the identity of interferer and *TxID*/*RxID* is the transmitter/receiver ID of the interfered flow. In the above example, when node 2 receives the broadcast packet containing $(2, 0, 1)$, it knows that flow 0→1 suffers from its interference. When next time node 2 overhears that 0→1 is transmitting, it will yield to flow 0→1 until the transmission is finished[4].

When the sender examines the CSID-SR mapping, it's possible that the record with low success ratio corresponds to the CSID consisting of multiple flows. In this case, the sender does not know the interference is due to which node(s). In our implementation, we only report 1st-order interferers, i.e. the TransmitterID in CSID that consists of only one flow whose SR is less than 50%. A more advanced approach could be to attribute the interference to the flow with the strongest receive signal strength (RSS) value. More information regarding the interferers' identities could also be extracted through mining the entire CSID-SR mapping. We leave this as our future work.

D. Opportunistic Channel Access

Each sender continually overhears the on-going flows and updates its CSID accordingly. Each time the CSID is changed, the sender updates its prediction of the channel status. The prediction consists of three parts. First, if a flow that will be interfered by the sender's transmission is active (appears in the CSID), then the sender predicts the channel busy. Second, if the CSID corresponds to a success ratio less than 50%, the sender predicts channel busy; otherwise, it predicts channel idle. Finally, if the sender does not overhear any CSID, then it falls back to the standard carrier sensing. To incorporate existing backoff mechanism into OCP, the backoff timer needs to be suspended (released) whenever the channel is predicted busy (idle). Therefore, we are guaranteed that after the backoff is finished, the DATA is always transmitted when channel is predicted idle. The detail of the prediction process is shown in Figure 7.

Note that rather than totally relying on the conflict mapping as what is done in CMAP[20], the above prediction process ensures that the sender accesses the medium only in an opportunistic manner, i.e. when it is confident that accessing the medium will likely be successful and cause no collision to other flows. In Section V, we will see that blindly turning off carrier sensing suffers from significant throughput loss by up to 71% in random topologies.

[4]This is possible since node 2 knows the transmission duration of 0→1 from the *Length* field.

predict(*CSID*)
1: **if** there is any flow in CSID that is interfered by me **then**
2: return BUSY
3: **if** CSID contains no flow information **then**
4: **if** interference $>$ CSThresh **then**
5: return BUSY
6: **else**
7: return IDLE
8: **if** CSID-SR mapping does not contain the record for the CSID **then**
9: return IDLE
10: **if** success ratio of CSID $>$ 0.5 **then**
11: return IDLE
12: **else**
13: return BUSY

Fig. 7. Pseudo-code for predicting the channel status at the sender node

V. Performance Evaluation

We implement OCP in ns-2 simulator. Existing ns-2 does not allow for packet capture even when the SINR of one packet is much larger than the other. Therefore, we implement a more realistic capture model [15] by considering the propagated interferences from all other nodes in the network. A packet can be received only if the signal to interference and noise ratio (SINR) is larger than the predefined threshold and the signal is above the sensitivity level (receive threshold). We set the SINR threshold according to the measurement study in [1] and receive threshold according to [2] so that the corresponding receive range is 232m for 11 Mbps data rate and 550m for 1 Mbps data rate (for modulating CSID).

The methodology to evaluate the efficacy of OCP is as follows. First, we study a simple random topology setting. We randomly place 5 distinct flows, each running a *backlogged* CBR traffic in a 600m by 600m area. Since CSID is modulated using the lowest data rate, all nodes are likely to receive each other's CSID in this setting. Second, we evaluate OCP's performance in a more complicated random topology setting. We place 20 distinct flows in a 1000m by 1000m area. In such topologies, hidden terminals may exist. Furthermore, with 20 flows, the contention level will be more variant and thus more difficult for a node to infer the interference relationship. For each of the above two settings, we randomly generate 50 topologies and compare the performance between OCP and the popular IEEE 802.11 protocol that is based on CSMA. Unless otherwise stated, binary exponential backoff is turned off so that each node maintains the same contention level during the evaluation.

We try to answer the following questions in the next few sections: (§V-A) Should we turn off carrier sensing and purely rely on the inferred interference relationship as done in CMAP [20]? (§V-B) Can OCP improve the throughput over existing CSMA mechanism? (§V-C) Can OCP improve the packet delivery success ratio over existing CSMA? (§V-D) Can OCP alleviate starvation in the network?

A. Carrier Prediction Should be Opportunistic and Carrier Sensing Should NOT be Turned Off

As discussed in Section II, CMAP [20] argued that carrier sense (CS) is too conservative and proposed to rely on the

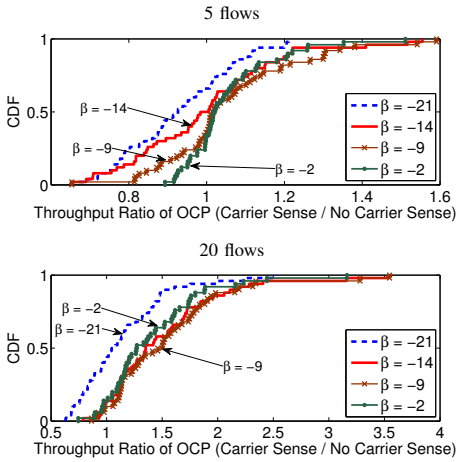

Fig. 8. CDF of the throughput ratio of OCP with carrier sense to OCP without carrier sense for 50 random topologies at β = -21, -14, -9, -2. The figure will be more intelligible if it's color-printed.

inferred conflict mapping rather than on carrier sensing. In this section, we ask the question that if we can infer the interference relationship, shall we turn off carrier sensing all the time as proposed in CMAP [20]?

We first define β as the carrier sense threshold normalized by the sensitivity of receiving a packet, i.e. β = *CSThresh / RxThresh*. Since an OCP-enabled sender falls back to carrier sense when it does not overhear any CSID, we vary the carrier sense threshold which OCP-enabled senders fall back to. In particular, we set the carrier sense threshold so that the corresponding carrier sense range is 768m, 512m, 384m, and 256m. The corresponding β value is -21, -14, -9, -2. Now, for each of the 50 random topologies, we compare the throughput of OCP for which carrier sense threshold set to these β values with the throughput of OCP for which carrier sense is disabled.

Figure 8 shows the cumulative distribution function (CDF) of the throughput ratio (throughput of OCP with carrier sense / throughput of OCP without carrier sense) for each of the 50 random topologies. As we can see for the 5-flow random topologies, when the carrier sense range is 768m (β = -21), 60% of the topologies for which carrier sensing is turned on perform worse than simply turning off the carrier sensing. The largest throughput ratio is 1.2, and smallest throughput ratio is only 0.2. Indeed, adopting carrier sensing in this case is too conservative as claimed by [20]. However, as CS range decreases (β increases), the throughput ratio improves. At $\beta = -2$, CS-enabled OCP performs better than CS-disabled OCP for 75% of the 50 topologies. When the contention level increases, it is more and more difficult to correctly decode the CSID to infer the interference relationship. Blindly contending for the channel with insufficient information will likely result in collision. This can easily be seen in the plot of random 20-flow topologies in Figure 8. In this case, CS-enabled OCP outperforms CS-disabled OCP for more than 60% of the topologies for $\beta = -21$ and more than 90% of the topologies for $\beta = -14, -9, -2$. From Figure 8, there is always a carrier sense setting such that CS-enabled OCP performs much better than CS-disabled OCP for 5-flow and 20-flow random topologies. Thus, we argue that OCP should be opportunistic and carrier sense should NOT be turned off.

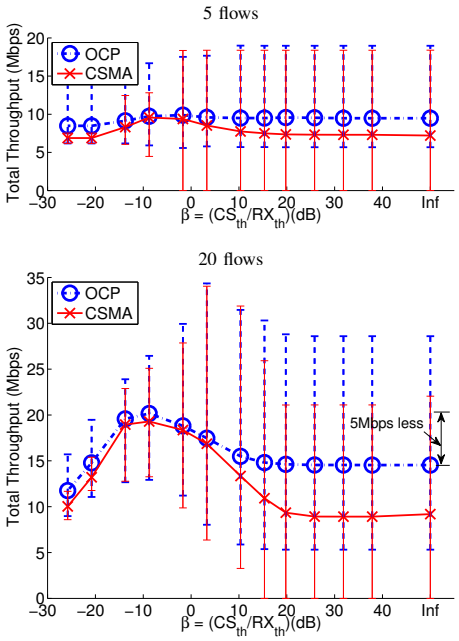

Fig. 9. Mean, Max, and Min total throughput of 50 random topologies for OCP and CSMA at different CS_{th}

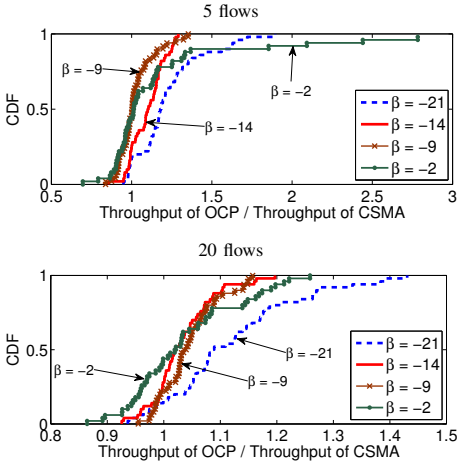

Fig. 10. Throughput ratio of OCP to CSMA over 50 random topologies at β = -21, -14, -9, -2. The figure will be more intelligible if it's color-printed.

As we have shown in Section III, relying on partial packet recovery (PPR) [11] to decode the header and trailer of two colliding packets results in low per-flow decode ratio, thereby not helping to infer the interference relationship in the network with such high contention level. Since CMAP relies on PPR to infer the interference relationship and aggressively turns off the carrier sensing in all cases, we argue that CMAP performs at best as good as CS-disabled OCP.

B. OCP Improves Throughput

In this section, we try to answer how much throughput improvement OCP has over CSMA. Since an OCP-enabled sender falls back to carrier sense when it does not overhear any CSID, we compare the performance of OCP and CSMA at varied carrier sense threshold to evaluate how much gain OCP brings.

Figure 9 shows the max, min, and average throughput of

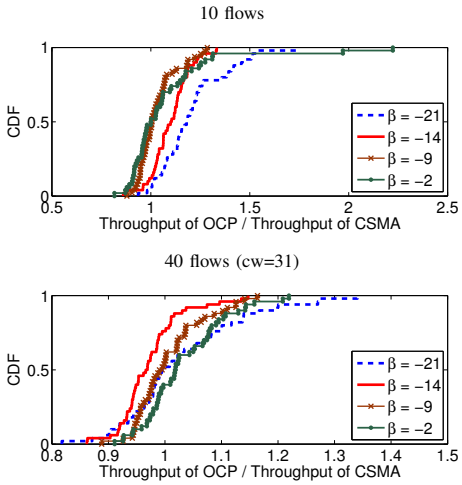

Fig. 11. Throughput ratio of OCP to CSMA over 50 random topologies at β = -21, -14, -9, -2. The figure will be more intelligible if it's color-printed.

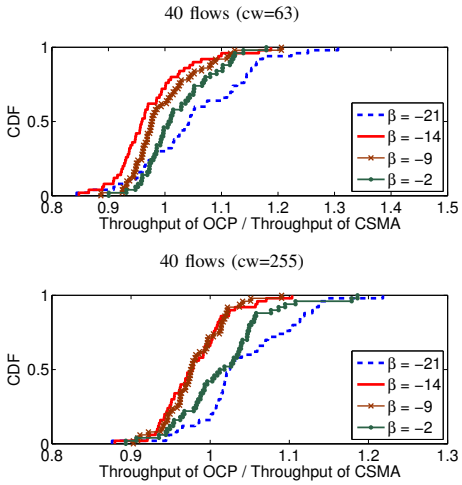

Fig. 12. Throughput ratio of OCP to CSMA over 50 random topologies at β = -21, -14, -9, -2. The figure will be more intelligible if it's color-printed.

the 50 random topologies for varied β values. We see that when carrier sense threshold is small, all the nodes can hear each other and the network becomes essentially a single hop network. In this case the total system throughput for CSMA does not vary much. But when carrier sense threshold is set to larger values (β >10), the throughput for CSMA could vary from 0 Mbps to as large as 20 Mbps. An OCP sender, however, accesses the medium whenever there is an opportunity. Therefore, when carrier sense threshold is low (β =-26), i.e. network is essentially single-hop, OCP sender nodes can still grab the opportunity to boost the throughput to 14.5 Mbps for 5 random flows and 15.5 Mbps for 20 random flows. Even at the optimal carrier sense threshold (β =-9), the average throughput of OCP outperforms that of CSMA. Furthermore, at larger carrier sense thresholds, OCP improves not only the average total throughput by up to 67% but also the min total throughput from 0 Mbps to at least 5.5 Mbps.

Comparing the two plots in Figure 9, we see that OCP's average total throughput varies more significantly at different β values for 20 random flows. For example, the average total

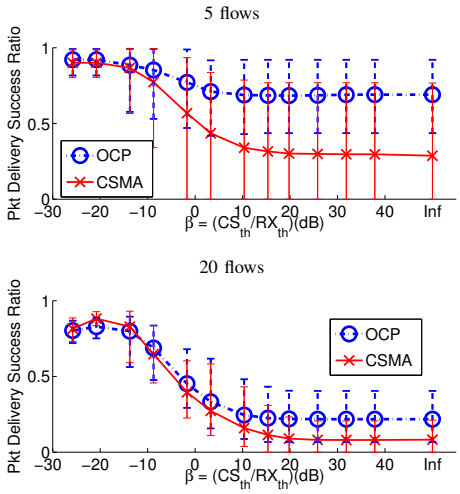

Fig. 13. Mean, Max, and Min packet delivery success ratio of 50 random topologies for OCP and CSMA at different CS_{th}

throughput of OCP is 12 Mbps at $\beta = -26$, 20 Mbps at $\beta = -9$, and 15 Mbps at β = INF (no carrier sense). Again, if we simply turn off carrier sense all the time, we may perform OK for 5 random flows in terms of average total throughput, but we could lose up to 5 Mbps for 20 random flows compared with OCP with optimal carrier sensing. Although finding the optimal carrier sense threshold is out of the scope of this paper, we point out that the effect of OCP may need to be taken into consideration when tuning carrier sense threshold to its optimality.

We further compare the throughput improvement of OCP over CSMA for each of the 50 random topologies. Figure 10 shows the CDF of the throughput ratio of OCP to CSMA for each topology for varied β values. Let's first see the 5-flow case. For β = -9 and -2, OCP outperforms CSMA for only around 50% of the topologies. This is because carrier sense thresholds at these two values are already optimal (see Figure 9) and there is not much space left for OCP to opportunistically access the medium. When there is no such opportunity, OCP consumes more overhead and results in around 12% throughput loss. When β = -21 and -14, CSMA becomes more conservative and OCP is able to exploit the opportunity of flow concurrency and improves the throughput for more than 80% of the topologies. When we place more flows (the 20-flow plot in Figure 10) in a larger area, the contention level varies more in the network and the opportunity for concurrent transmission is more likely to occur. Indeed, OCP outperforms CSMA for more than 85% of the topologies for β = -21, -14, and -9 and 62% of the topologies for β = -2.

C. Does OCP Improve Packet Delivery Success Ratio?

In this section, we evaluate OCP's performance in link layer packet delivery success ratio. We plot the max, min, and average success ratio for the 50 random topologies for OCP and CSMA with varied β value in Figure 13. For both 5-flow and 20-flow random topologies, OCP and CSMA's success ratios decrease when carrier sense threshold increases (β increases). This is because by setting to large carrier sense threshold each node contends for the medium more aggressively and ignores near-by transmissions. As a result, for random 5 flows CSMA's

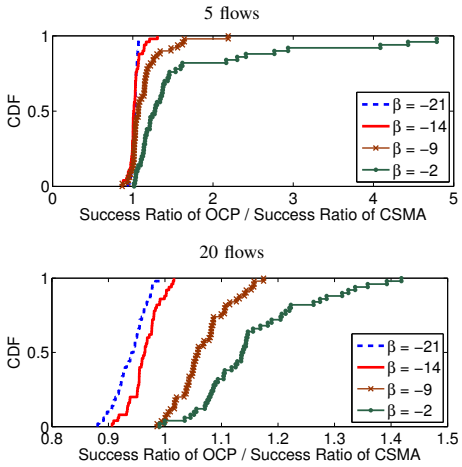

Fig. 14. CDF of the ratio of packet delivery success ratio of OCP to that of CSMA over 50 random topologies at β = -21, -14, -9, -2

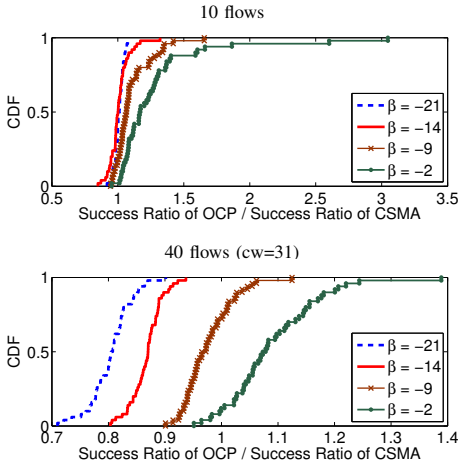

Fig. 15. CDF of the ratio of packet delivery success ratio of OCP to that of CSMA over 50 random topologies at β = -21, -14, -9, -2

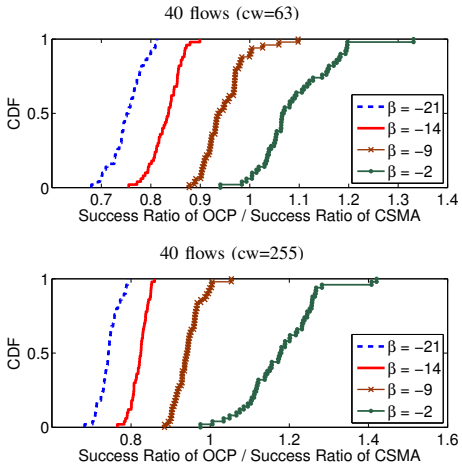

Fig. 16. CDF of the ratio of packet delivery success ratio of OCP to that of CSMA over 50 random topologies at β = -21, -14, -9, -2

average success ratio decreases from 90% to 30% while OCP improves it to 92% and 73% respectively. For random 20 flows, CSMA's average success ratio decreases from 87% to 9%, while OCP from 83% to 23%. Comparing the 5-flow plot with 20-flow plot in Figure 13, we see that OCP does not improve the success ratio in the 20-flow plot as much as in 5-flow plot. There are two reasons. First, the contention level for 20 flows could be much more intensive than 5 flows, causing more packet collisions. Second, with 20 flows, it is more likely for a sender node to decode a non-interferer while missing the packet of the true interferer when both are active. As a result, both CSMA and OCP have lower success ratios for 20 random flows than for 5 random flows.

More interestingly, OCP does not improve the success ratio over CSMA for 20 random flows when β is small. This is because when β is small, the network is essentially one-hop (nodes can sense the transmission of each other) and the success ratio for CSMA is significantly increased to more than 80%. Since we encourage nodes to contend for the channel whenever the packet delivery success ratio is larger than 50%, OCP senders become more aggressive in channel access, thereby not improving the success ratio. Despite that OCP decreases the success ratio when $\beta < -10$ for 20 random flows, it improves the average total throughput by 2 Mbps (20-flow plot in Figure 9). To illustrate further, we compare the success ratio of OCP and CSMA for each of the 50 random topologies and draw the CDF for the ratio of success ratio in Figure 14. For both 5-flow and 20-flow random topologies, OCP has significant improvement over CSMA for β = -9 and -2. On the other hand, for β = -21 and -14, it does not perform significantly better for 5-flow random topologies and performs even worse for 20-flow random topologies. If we compare Figure 10 with Figure 14, we immediately see that while OCP performs worse for β = -21 and -14 in success ratio, it is at these two β values that OCP performs much better in terms of throughput. On the other hand, OCP outperforms CSMA for β = -9 and -2 in success ratio, its throughput improvement becomes mediocre. The reason is because an OCP sender contends for the medium when it sees an opportunity under the condition that the packet delivery success ratio is at leat 50%. Such opportunistic approach may sometimes reduces packet delivery success ratio trading for more throughput. On the other hand, when there is not much opportunity to improve the throughput (β = -9 and -2 in Figure 10), OCP senders become more conservative rather than blindly access the channel, thereby improving the packet delivery success ratio (Figure 14).

D. OCP Alleviates Starvation

Recall that the design purpose of OCP is for improving the concurrency of flow transmissions in various contention levels. We try to understand whether OCP can also mitigate the starvation of the flows. We define that a flow is starved when it gets no throughput during the entire simulation. For ease of exposition, let's see one random 5-flow topology in Figure 17. In such a topology, node 13 suffers from the interference from node 8 due to the close distance between them. Node 12, on the other hand, suffers from the interference from node 5 when

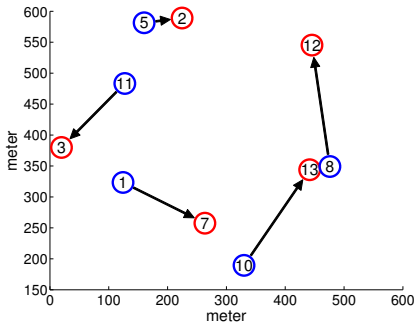

Fig. 17. One of the 50 random 5-flow topologies

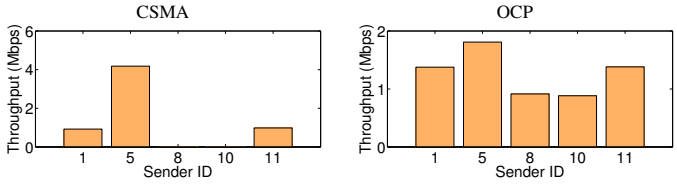

Fig. 18. Throughput profile of the 5-flow random topology in Figure 17

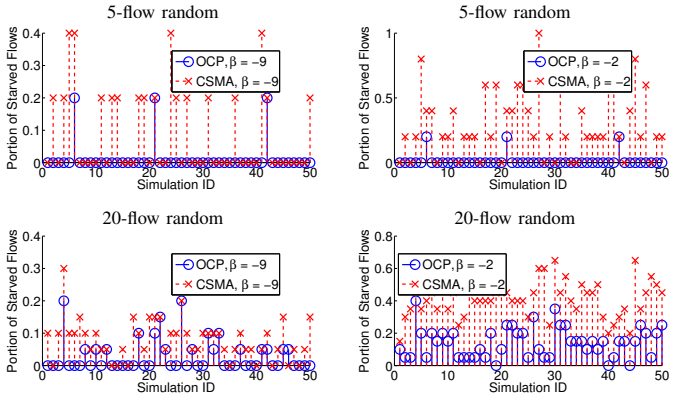

Fig. 19. Portion of flows that are starved for 5-flow and 20-flow random topologies at β = -9 and -2

both node 8 and 5 are active at the same time[5]. In such a topology, if β is set to -9 (the optimal value for 5-flow random topologies in general), both flows 8→12 and 10→13 will be severely starved for CSMA, as shown in Figure 18. On the other hand, OCP totally removes the starvation of the two flows and improves the fairness among the flows.

To further show that the above illustration is not a niche example, we compare OCP and CSMA by plotting in Figure 19 the portion of starved flows for each of the 50 random topologies for β = -9 and -1. We do not observe much starvation on other smaller β values and thus ignore those plots. Clearly, OCP almost completely removes the starvation for 5-flow random topologies and significantly reduces the starvation for 20-flow random topologies. Interestingly, for the 27-th 5-flow random topology at β = -2, all the 5 flows are completely starved in CSMA while OCP totally removes the starvation in this case.

E. Typical Exposed/Hidden Terminal Topologies

As shown in Figure 20, we place four nodes in the network. In this topology, node 0 and node 2 are two exposed senders and can sense each other's transmission. In this topology,

[5]Whether the packet can be received depends on the SINR. In this case, node 2 can receive the packet from node 5 even when node 5 and node 8 are both active simultaneously.

Fig. 20. Symmetric two-flow topology, node 0 and 2 are two exposed senders

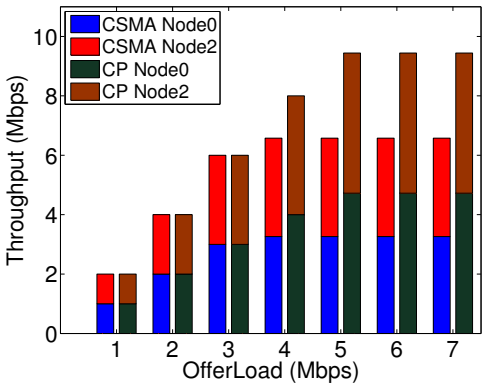

Fig. 21. Throughput of two exposed senders for varied offer load

Fig. 22. Asymmetric two-flow topology, sender 2 is hidden from sender 0

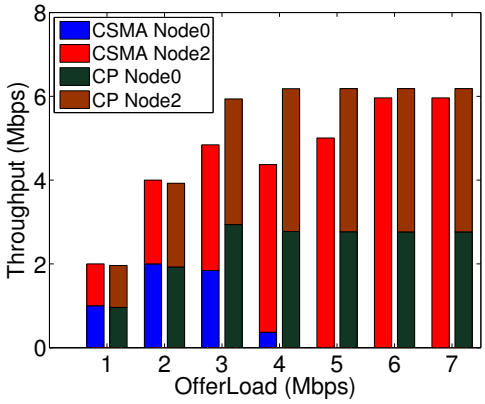

Fig. 23. Throughput of asymmetric hidden-exposed flows for varied offer load

since receiver 1 and 3 are far away from their corresponding interferers 2 and 0 respectively, flow 0→1 and 2→3 can be active concurrently. However, existing CSMA protocol does not allow such concurrency since node 0 and 2 can hear each other. As a result, at most one flow can be active at any given point of time. In deed, in Figure 21, we see that when the offer load of both flows increases, flow 0→1 and 2→3 are able to achieve fair throughput. But the total throughput can not go beyond 6.3 Mbps, the max throughput of a single flow. By introducing opportunistic medium access, node 0 and node 2 can access the channel concurrently. Therefore, the total system throughput is improved to 9.4 Mbps.

We then evaluate the performance of OCP for the topology shown in Figure 22. Since node 2 is in the proximity of node 1, it can potentially interfere node 1's reception. Further, node 2 is hidden form node 0 and existing carrier sensing does not help node 0 to detect the existence of node 2. One possible solution is to simply reduce the carrier sense threshold at node 0. But this also forces node 0 to be silent when an exposed flow that can be concurrently active occurs. OCP on the other hand, allows node 0 to capture the CSID of flow 2→3 and

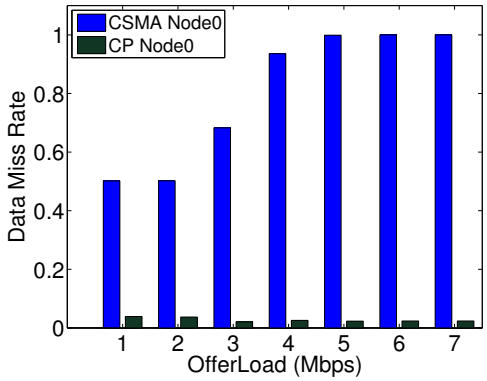

Fig. 24. Data miss rate of asymmetric hidden-exposed flows for varied offer load

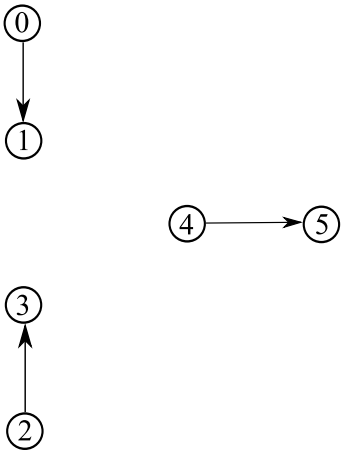

Fig. 25. Asymmetric hidden-exposed 3-flow topology. In this scenario, node 4 is hidden from both node 0 and 2

yield to such transmission without giving up the opportunity of concurrency with exposed flows. As shown in Figure 23, we can see that as the offer load of two flows increases, node 0's throughput reaches its maximum at 2 Mbps, then quickly drops to zero. On the other hand, OCP allows node 0 to intelligently compete with its interfering flow node 2→3 and achieves almost fair throughput among the two flows. Note also that OCP's total throughput of the two flows outperforms that of CSMA for most of the offer loads. Figure 24 shows the data miss rate of the two flows. Since receiver 3 will not be interfered by all potential interferers, we only plot the data miss rate for flow 0→1. Clearly, due to the hidden terminal node 2, node 0 does not benefit from CSMA and thus blindly access the channel. On the other hand, with the help of OCP, node 0 and node 2 cooperatively access the channel and reduces node 0's data miss rate to be less than 5%.

F. Topology with Dominating Interferers

We next place 6 nodes as shown in Figure 25. In this scenario, node 4 is hidden from both node 0 and 2. Therefore, node 0 and 2 do not know the existence of such an interferer that can affect the packet reception at their respective receiver node 1 and 3. So, as the offer load increases, we expect that node 4 will gradually grab the channel and dominating all the transmissions in the air. As shown in Figure 26, when the offer load increases, the throughput of flow 0→1 and 2→3 for CSMA peak at 2-Mbps offer load. After that, their

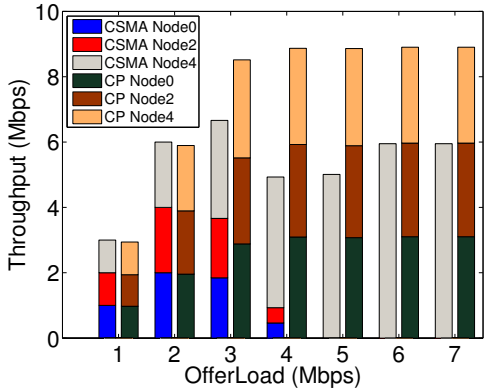

Fig. 26. Throughput of asymmetric hidden-exposed 3-flow for varied offer load

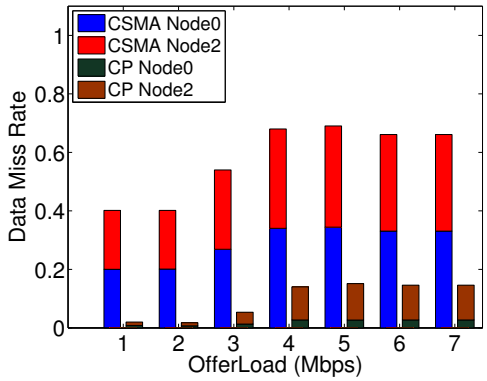

Fig. 27. Data miss rate of asymmetric hidden-exposed 3-flow for varied offer load

respective throughput decreases to zero at 5-Mbps offer load, while flow4→5 dominates the medium usage. Note that a carefully CSMA scheme in this scenario only allows at most one flow be active. As a result, the total throughput does not exceed 6.5 Mbps, the max throughput of a single flow. OCP, on the other hand, ensures the three flows to fairly share the medium so that each flow has around one third of the total throughput. Further, in this scenario, flow 0→1 and 2→3 can be active simultaneously as long as flow 4→5 is not active. OCP is not only able to allow concurrency of such two flows but also intelligent coordinate between all the three flows so that they fairly share the medium, despite the asymmetry of the topology. By allowing concurrency of flow 0→1 and 2→3, the total throughput of OCP increases by 33% than that of CSMA. We also draw the data miss rate for flow 0→1 and 2→3 in Figure 27. As we can see, OCP reduces the data miss rate of the two flows from up to 70% to 15% at high offer loads, and from 40% to 3% at low offer loads.

VI. Related Work

Many protocols have been proposed for medium access control in wireless networks. MACA [13], MACAW [3], and FAMA [8] are the earlier proposals for handling exposed/hidden terminal problems. They mainly designed floor acquisition schemes through the exchange of specific control packets (RTS, CTS, DS, ACK etc) to selectively silence wireless nodes in the network for interference avoidance. IEEE 802.11 DCF [10] is probably the most poplar CSMA/CA protocol. It adopts not only physical carrier sensing but also virtual carrier sensing

(RTS/CTS) so that two nodes in a cell that are two hops away know the existence of each other through RTS/CTS floor acquisition. However, since RTS/CTS incurs at least 37% and 29% overhead for 11Mbps 802.11b and 54Mbps 802.11a/g respectively [5], virtual carrier sensing is turned off by default in practice.

Since physical carrier sense is adopted in IEEE 802.11, many works have studied tuning the optimal carrier sense threshold to maximize the spatial reuse. [23] derived theoretical estimation of the optimal carrier sense threshold based on SINR interference model. They also proposed a distributed algorithm adapting carrier sense threshold in [24]. However, the above studies ignored the impact of MAC overhead in the analysis and it has been shown [21] that the aggregate throughput could suffer from a significant loss if MAC overheard is not considered properly. [16] prosed an enhanced carrier sensing mechanism by adapting the EIFS duration based on the length of packet types (RTS, CTS, DATA, ACK) observed on the medium. [17] adapts carrier sense threshold based on transmitter-receiver distance. [12] experimentally verified the efficacy of carrier sense and identified existing problems of carrier sense. All the above works proposed the solutions within the context of carrier sensing, while we go one step further to incorporate carrier sense as part of our medium access scheme.

Besides controlling the carrier sense threshold, other works have studied controlling the modulation rate [6], [9], [19], transmission power [18], or a combination of them [7], [14], [22] to allow for more concurrent active flows in one-hop or multi-hop wireless networks. [9] proposed to utilize RTS/CTS control packet and let the receiver decide the modulation scheme for the next coming DATA packet. [19] proposed to further opportunistically transmit more DATA packets when the channel condition at the receiver is good. [6] considered the interference pattern and jointly controlled modulation scheme and frame size to exploit medium access opportunities. POWMAC [18] inserted an interference margin in the CTS packet to tolerate certain amount of interference at the receiver, thereby increasing the number of concurrent active flows. Finally, [7], [22] jointly control the transmission power and carrier sense threshold, [14] proposed to tune modulation scheme, transmission power, and carrier sense threshold all together to improve spatial reuse in multi-hop wireless networks. In this paper, we only focus on the carrier sensing aspect for spatial reuse. We leave it as future work for incorporating modulation scheme or power control into the OCP framework.

VII. Conclusion

Since wireless medium is a shared resource, how to control the medium access scheme to reduce interference and increase spatial reuse is a crucial topic in wireless networks. In this paper, we have presented OCP for each sender to opportunistically access the wireless medium. OCP is based on the rationale that the past interference information could be used as an indicator for future packet delivery outcome. An OCP-enabled sender accesses the medium only when it is confident that the channel access will likely to succeed and cause no collision to other flows. We propose a novel CSID-SR mapping to allow for interference inference at the sender side even under high network contention. Further, we have shown that such medium access scheme needs to be done *opportunistically*. An OCP node must fall back to carrier sensing when there is no information overheard in the air, or there will be significant throughput degradation. In OCP, each receiver only needs to focus on correctly decoding the packet since OCP is a purely sender-side interference inference medium access scheme. Compared with CSMA, we have shown that OCP significantly improves the throughput, packet delivery success ratio, and alleviates starvation in various random topologies with different contention levels.

VIII. Acknowledgement

We thank Guanfeng Liang for the fruitful discussion. This research is supported in part by Vodafone Fellowship and US Army Research Office grant W911NF-05-1-0246. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

References

[1] A. Akella, G. Judd, P. Steenkiste, and S. Seshan. Self management in chaotic wireless deployments. In *Proceedings of ACM MobiCom*, 2005.

[2] S. Bansal, R. Shoreyy, and A. Kherani. Performance of tcp and udp protocols in multi-hop multi-rate wireless networks. In *Prof. of IEEE WCNC*, 2004.

[3] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang. MACAW: A medium access protocol for wireless LANs. In *Proceedings of ACM SIGCOMM*, 1994.

[4] A. Chan and S. C. Liew. Merit of phy-mac cross-layer carrier sensing: A mac-address-based physical carrier sensing scheme for solving hidden-node and exposed-node problems in large-scale wi-fi networks. *LCN*, 0, 2006.

[5] C. Chen and H. Luo. The case for heterogeneous wireless MACs. In *Proceedings of HotNets*, 2005.

[6] C. Chen, H. Luo, E. Seo, N. Vaidya, and X. Wang. Rate-adaptive framing for interfered wireless networks. In *Proceedings of IEEE INFOCOM*, 2007.

[7] J. A. Fuemmeler, N. H. Vaidya, and V. V. Veeravalli. Selecting transmit powers and carrier sense thresholds in csma protocols for wireless ad hoc networks. In *Proc. of WICON06*, 2006.

[8] C. L. Fullmer and J. Garcia-Luna-Aceves. Solutions to hidden terminal problems in wireless networks. In *Proceedings of ACM SIGCOMM*, 1997.

[9] G. Holland, N. Vaidya, and P. Bahl. A rate-adaptive MAC protocol for multi-hop wireless networks. In *Proceedings of ACM MobiCom*, 2001.

[10] IEEE Computer Society. Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. IEEE standard 802.11, 1999.

[11] K. Jamieson and H. Balakrishnan. PPR: Partial Packet Recovery for Wireless Networks. In *ACM SIGCOMM*, 2007.

[12] K. Jamieson, B. Hull, A. K. Miu, and H. Balakrishnan. Understanding the real-world performance of carrier sense. In *Proceedings of ACM SIGCOMM E-WIND Workshop*, 2005.

[13] P. Karn. MACA: A new channel access method for packet radio. In *Proceedings of IEEE Computer Network Conference*, 1990.

[14] T.-S. Kim, H. Lim, and J. C. Hou. Improving spatial reuse through tuning transmit power, carrier sense threshold, and data rate in multihop wireless networks. In *Proc. of MobiCom06*, 2006.

[15] A. Kochut, A. Vasan, A. U. Shankar, and A. Agrawala. Sniffing out the correct physical layer capture model in 802.11b. In *ICNP '04*.

[16] Z. Li, S. Nandi, and A. Gupta. Improving mac performance in wireless ad-hoc networks using enhanced carrier sensing (ecs). In *In Proc. of IFIP NETWORKING*, 2004.

[17] N. A. M. Maung, T. Noguchi, and M. Kawai. Maximizing aggregate throughput of wireless ad hoc networks using enhanced physical carrier sensing. 2008.

[18] A. Muqattash and M. Krunz. A single-channel solution for transmission power control in wireless ad hoc networks. In *Proc. of MobiHoc04*, 2004.

[19] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly. Opportunistic media access for multirate ad hoc networks. In *Proceedings of ACM MobiCom*, 2002.

[20] M. Vutukuru, K. Jamieson, and H. Balakrishnan. Harnessing Exposed Terminals in Wireless Networks. In *Proceedings of NSDI*, 2008.

[21] X. Yang and N. H. Vaidya. On the physical carrier sense in wireless ad-hoc networks. In *Proceedings of IEEE INFOCOM*, 2005.

[22] X. Yang and N. H. Vaidya. Spatial backoff contention resolution for wireless networks. In *Proceedings of IEEE WiMesh*, 2006.
[23] J. Zhu, X. Guo, L. L. Yang, and W. S. Conner. Leveraging spatial reuse in 802.11 mesh networks with enhanced physical carrier sensing. In *Proceedings of IEEE ICC*, 2004.
[24] J. Zhu, X. Guo, L. L. Yang, W. S. Conner, S. Roy, and M. M. Hazra. Adapting physical carrier sensing to maximize spatial reuse in 802.11 mesh networks: Research articles. *Wirel. Commun. Mob. Comput.*, 4(8):933–946, 2004.