On the Transport Layer Throughput in Cognitive Mobile Ad Hoc Networks using Soft Sensing

Mohamed Naguib, Yahya Z. Mohasseb Hisham Dahshan
Department of Communications, The Military Technical College, Cairo, Egypt, 11331.
Email: {mnaguib, mohasseb}@ieee.org, hdahshan@mtc.edu.eg

***Abstract*—This paper investigates the effect of cognitive operations in the physical (PHY) layer on the throughput of the transport layer in Cognitive Mobile Ad Hoc Networks (CoMANETs). The focus is on the use of soft sensing to enhance secondary users capacity in the PHY layer. A new transport layer protocol is proposed which updates the Congestion Window (CW) efficiently according to the power and rate adaptation strategy of the soft sensing scheme. The expected throughput is derived and compared to the case where the transport layer is oblivious to the use of soft sensing, as well as to a benchmark case from the literature where only hard sensing is used. Results show significant throughput improvements due to soft sensing in general. These improvements increase with proper adaptation of the transport layer protocol.**

I. Introduction

CoMANETs are an effective solution to the scarcity of the wireless resources since they enable Secondary Users (SU) to transmit on vacant portions of the spectrum under an average interference constraint with the Primary User (PU) [1].

Transport layer protocols used in fixed networks (e.g. [2]) cannot be applied directly to Mobile Ad-hoc Networks (MANETs) nor to CoMANETs. These protocols are designed to detect congestion by measuring the Round Trip Time (RTT). Once the RTT exceeds a threshold, a Retransmission Time Out (RTO) event causes the TCP to reduce its CW to one. However, both in MANETs and CoMANETs packet loss due to fading or user mobility could cause RTO without actual congestion. Furthermore, it can occur in CoMANETs due to cognitive operations at intermediate nodes, e.g. spectrum sensing or spectrum switching, or due to SU interference with PU transmission due to mis-detection. If the source reduces the CW to one in all previous cases, the number of successful transmitted packets, and consequently the average throughput, decrease. Many Protocols were proposed to enhance the transport layer performance in MANETs [3]–[6]. In particular, the Bandwidth Delay Product (BDP) and the minimum RTT are used to judiciously update the CW in [5] and cross-layer cooperation and feedback information are used in [6] to improve the performance of TCP over MANETs.
Neither TCP protocols [2] nor its MANETs counterparts [3]–[6] are designed to tolerate the periodic traffic interruptions caused by channel sensing or switching in CoMANETs. Therefore, development of CoMANETs compatible protocols attracted significant attention [7]–[10]. Perhaps the earliest work appeared almost a decade ago in an earlier version of [7], where a modified version of TCP targeted Cognitive Radio Ad Hoc Network (TCP-CRAHN). TCP-CRAHN used explicit notifications from intermediate nodes and destination to differentiate between transmission disconnection during sensing and other causes. However, it did not address the variability of bandwidth available after channel switching. This issue was addressed in [9], where the proposed protocol, TCP-COBA, updates the CW based on the BDP and the available buffer space in the relay nodes when switching channel. In [8], an enhancement of TCP Westwood [11] bandwidth estimation function is proposed to properly modify the CW during spectrum sensing at the destination node. However, intermediate nodes are not taken into account which makes it prone to buffer overflow when these nodes sense the spectrum. A TCP-Friendly Rate Control protocol for Cognitive Radio (TFRC-CR) is proposed in [12] which continuously adjusts the transmission rate using equations which incorporate FCC spectrum databases to obtain side information on the PU. Other non-TCP based transport protocols have been proposed for Cognitive Radio based Sensor Networks, e.g. [10], [13] which focus more on the energy efficiency issues associated with these networks.

Soft-sensing has long been proved to improve the rate for the SU, by adapting its transmission power or rate according to the energy sensed on the channel [14]. Despite the attention it has received, its impacts on transport protocols have not been investigated. In this paper, we take the first steps towards designing a Soft-Sensing Aware TCP protocol (SSA-TCP) that adapts to the different states of the SU in such a system. The protocol uses explicit notifications to update CW based on the link state, and rate adaptation during spectrum sensing as in [7]. We analyze the throughput of this protocol and compare it to two benchmarks. First, the case where the transport layer is unaware of the soft sensing operations on the PHY layer and assumes that the rate is fixed during all transmissions. Second, the case where hard sensing is used with TCP-CRAHN. Our results show that there is a potential for throughput increase in a wide variety of cases under the use of soft-sensing, even if the transport layer is not responsive to this type of sensing.

Following this introduction, the SSA-TCP is presented in Section II and its throughput is derived in Section III. Section IV presents the performance evaluation, and finally Section V concludes the paper.

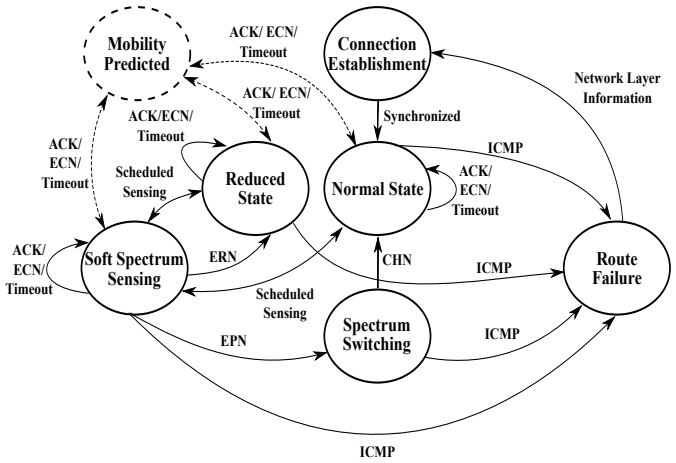

Fig. 1: Finite state machine model of the SSA-TCP.

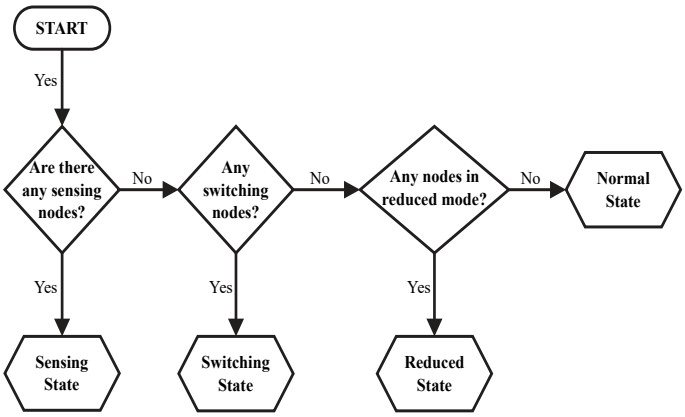

Fig. 2: Determining the *end-to-end* link state from states of the nodes.

II. The Soft Sensing Aware - TCP protocol (SSA-TCP)

A. Notation and SSA-TCP State Diagram

Before going into the detailed description of the protocol states, it is important to differentiate between the states of the intermediate nodes and the state of the transport layer connection, which describes the *end-to-end* link state. Each node can be either in sensing, normal, reduced, or switching states with probabilities p_s, p_n, p_r and p_{sw} respectively. At the transport layer, the *end-to-end* link will be in any of the above mentioned states with probabilities P_s, P_n, P_r and P_{sw} respectively. As illustrated in Figure 2, the *end-to-end* link state is determined from knowledge of the individual states of all intermediate nodes by applying the following rules sequentially: 1) The *end-to-end* link is considered to be in the *Sensing* state if any node along the route is performing sensing. 2) If none of the nodes is in sensing state, the link will be in *Switching* state if any node is in switching state. 3) The link will be in the *Reduced* state if any node is in the reduced state but all nodes are neither sensing, nor switching. 4) Finally, the link is in *Normal* state if all intermediate nodes along the route are in normal state. As will be detailed in the sequel, we focus on the throughput of the *end-to-end* link provided the connection is already established. Therefore, we do not consider the probabilities of other states mentioned in Section II-F. Figure 1 shows the state diagram for the *end-to-end* connection. The probability of an event $\mathbf{A}$ is denoted by $\mathbb{P}(\mathbf{A})$.

B. Soft Spectrum Sensing

Soft Sensing is performed periodically. The sensing time, t_s is determined based on a trade-off between increasing throughput by decreasing t_s, and increasing the reliability of the sensing outcome by increasing t_s. As a simplifying assumption, we assume here that t_s is the same for all nodes. However, we assume that the start of the sensing period is randomly chosen by each node upon deployment. The *Connection Establishment* phase ensures that the source is aware of the sensing cycles of all intermediate nodes, and is able to determine the state of the link accordingly. The link is virtually disconnected at the node performing sensing, because the SSA-TCP source does not cease the transmission. Consequently, a flow control is needed to prevent buffer overflow at the node preceding the sensing node. The CW is updated to the $min(cwnd, \mathcal{R}_{i-1})$, where $cwnd$ is the latest updated value of CW during *Normal* or *Reduced* state, and $\mathcal{R}_{i-1}$ is the residual buffer space at the node preceding the sensing node. In a *Soft Sensing* scheme, the transmission power $\mathcal{P}$ is variable and depends on the received signal strength ξ from the channel. Here, we consider a 3-level power adaptation strategy that determines $\mathcal{P}$ and the following link state based on ξ using two thresholds, λ_L and λ_H with $\lambda_L < \lambda_H$. At any node If $\xi \geq \lambda_H$, then $\mathcal{P} = 0$, and the node sends an Explicit Pause Notification (EPN) to the source and transitions into switching state, causing SSA-TCP to transition into *Spectrum Switching* state according to Section II-A. If $\xi \leq \lambda_L$, then $\mathcal{P}$ is set to $\mathcal{P}_{max}$, and the node transitions into normal state, causing SSA-TCP transitions into *Normal* state. If $\lambda_L < \xi < \lambda_H$, then $\mathcal{P}$ is set to $\mathcal{P}_{min}$ to satisfy the average interference constraint as in [14], the node transitions into reduced state. Subsequently, it sends an Explicit Reduction Notification (ERN) to the source, which contains the reduced transmission rate R_r, causing SSA-TCP to transition into the *Reduced* state.

C. Normal State

SSA-TCP transitions into *Normal* state if all nodes in the link are in normal state, which is determined by not receiving EPN or ERN from any node at the end of its sensing time. It is assumed that each node has a fixed normal transmission rate R for a fixed normal time t_n. In this state, the same flow control as in [2] is used. So, the CW of SSA-TCP is increased in response to receiving ACKs. The RTO events and Explicit Congestion Notification (ECN) as in [7] is used to inform the source about congestion events, and the source must decrease its CW to reduce number of dropped packets. In order to control the CW in the following *Sensing* state, each intermediate node sends its residual buffer space $\mathcal{R}_i$ to the source by piggybacking it over the ACK, where, $\mathcal{R}_i$ is the

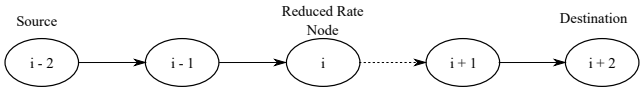

Fig. 3: A multi-hop cognitive mobile ad hoc network.

residual buffer space at each node which is used to control the CW during the *Sensing* state. According to the spectrum sensing schedule SSA-TCP transitions into *Sensing* state.

D. Reduced State

As explained in Section II-A, the link enters into the *Reduced* state if at least one node is in the reduced state, and none of the nodes is in sensing nor in switching state. This is indicated by any node sending an ERN at the end of the sensing state. Moreover, since the SU reduces power to $\mathcal{P}_{min}$, the Signal to Interference plus Noise Ratio (SINR), denoted by γ is decreased, which forces the PHY layer to reduce its transmission rate to R_r in order to preserve the same probability of error, where R_r is extracted from the ERN packet. Consequently, The transmission time, T_D, and the buffering time T_b increase. This increases RTT and decreases throughput. In order to avoid congestion, the transport layer limits its CW to ζ which equal the maximum allowable number of packets to be transmitted during the reduced state interval.

$$\zeta = \frac{R_r \times t_r}{S} \tag{1}$$

where, t_r is the time to be in the reduced state, and S is the data packet size. SSA-TCP transitions into *Sensing* state at the next scheduled sensing event.

E. Spectrum Switching

The SSA-TCP enters this state as explained in Section II-A. During this state transmission is ceased and the two affected nodes scan the spectrum for a free channel, and negotiate to operate on the new channel during a switching time t_{sw} which is assumed to be fixed. After the negotiation is done, the node turns back to its normal state, and a new channel message, called CHN, is sent to the source to update its status as normal. SSA-TCP transitions into *Normal* state on receiving a CHN message and ensures that all other nodes are in normal state.

F. Other States

Connection Establishment is performed using the standard 3-way handshake of TCP NewReno [2], which employs SYN, SYN-ACK, and ACK packets. However, each node in the path appends its ID, a timestamp, and $\{t_i, t_s\}$ to the Synchronization (SYN) packet, where, t_i is the remaining time before next sensing operation, and t_s is the time spent to perform sensing. After collecting this information from all nodes along the route, the receiver responds to the SYN packet by piggybacking the collected information over the SYN-ACK packet and sending it to the source. The source then deduces the sensing schedule. SSA-TCP transitions into the *Normal* state once the handshake is completed successfully. In the *Normal*, *Reduced*, and *Sensing* states the link may be virtually at the *Mobility Predicted* state. TCP transitions into this state if any node in the path set a Mobility Flag (MF) which is appended to an ACK packet, a node sets the MF based on the predicted ξ of the next node, which is estimated using the prediction framework used in [7]. TCP transitions into *Route Failure* state if any node in the link sends an ICMP packet to the source, and then the connection must be initiated again.

III. TCP Throughput Analysis

In this section the throughput of the proposed SSA-TCP is derived. Two additional scenarios are considered for comparison. Section III-B assumes that soft sensing is used in the PHY layer, but the transport layer is *unaware* of the associated rate reduction and does not adapt its transmission, thereby reducing throughput. Section III-C, derives the throughput for a hard sensing scheme to show whether soft sensing has an inherent advantage. We consider a TCP connection with k nodes in the *end-to-end* link, maximum number of retries defined in CSMA/CA is L, and bit error rate ε.

A. SSA-TCP Analysis

The averaged throughput can be estimated by the average number of bits transmitted in all states divided by the total average time:

$$B = \frac{B_n P_n t_n + B_r P_r t_r + B_s P_s t_s + B_{sw} P_{sw} t_{sw}}{P_n t_n + P_r t_r + P_s t_s + P_{sw} t_{sw}} \tag{2}$$

where, B_n, B_r, B_s, B_{sw} are the transport layer throughput in the *Normal*, *Reduced*, *Sensing* and *Switching* states respectively. The probability that the *end-to-end* link is in the ***Sensing*** state is given by:

$$P_s = 1 - (1 - p_s)^k \tag{3}$$

where, p_s is the ratio of the sensing time to the sensing period. The throughput during *Sensing* state will differ according to the preceding state. In particular, the throughput after the normal state, denoted here by $B_{s,n}$, is given by [7]:

$$B_{s,n} = \min\left\{\left(\sum_{i=2}^{k} \mathcal{R}_{i-1} \frac{1}{k t_s}\right), B_n\right\} \tag{4}$$

Similarly, if the sensing state comes after a reduced state then $B_{s,r}$ is given by:

$$B_{s,r} = \min\left\{\left(\sum_{i=2}^{k} \mathcal{R}_{i-1} \frac{1}{k t_s}\right), B_r\right\} \tag{5}$$

where, $\mathcal{R}_{i-1}$ is the residual buffer size of the node preceding the one performing sensing. From (4) and (5), it is possible to calculate the average throughput during *Sensing* as:

$$B_s = P_n . B_{s,n} + P_r . B_{s,r} \tag{6}$$

The probability that the *end-to-end* link is in the ***Normal*** state is the probability that all individual nodes are simultaneously in the normal state and is given by:

$$P_n = p_n^k \tag{7}$$

where p_n is the probability of a single node being in normal mode. This occurs if a node is not sensing and is transmitting at the normal rate, rather than the reduced one. In turn, this means that the previous sensing determined that $\xi \leq \lambda_L$. Therefore, $p_n = \pi_n(1-p_s)$, where $\pi_n = \mathbb{P}(\xi \leq \lambda_L)$. The throughput can be calculated using Equation (8) as in [7]

$$B_n = \frac{S}{RTT}\sqrt{\frac{3}{2p}} \tag{8}$$

where, $RTT = \bar{n}k(T_D + T_{ACK})$, T_D is the transmission time, $\bar{n}$ is the average number of MAC retransmissions and T_{ACK} is the ACK transmission time. Packet loss at link-layer occurs either due to interference or channel errors. Interference occurs due to the PU mis-detection causing packet loss with probability $p_I = p_m \times P_n$, where $p_m = \mathbb{P}(\xi \leq \lambda_L$ .PU is ON) is the probability of PU mis-detection in at least one node. The probability of packet loss due to channel errors is $p_{ch} = 1-(1-\varepsilon)^S$. Therefore, the probability of packet loss at link-layer p_e is given by:

$$p_e = p_I + p_{ch} - p_I \times p_{ch} \tag{9}$$

This enables us to write $\bar{n} = min\left(\sum_{i=0}^{L-1} ip_e^i(1-p_e), L\right)$ Finally, the probability of *end-to-end* packet loss is given by:

$$p = 1-(1-p_e^L)^k \tag{10}$$

The probability that the *end-to-end* link is in ***Reduced*** state is:

$$P_r = \sum_{i=1}^{k}\binom{k}{i}p_r^i(1-p_s-p_{sw}-p_r)^{k-i} \tag{11}$$

where, $p_r = \pi_r(1-p_s)$ and $\pi_r = \mathbb{P}(\lambda_L < \xi < \lambda_H)$. The throughput is:

$$B_r = \frac{S}{RTT_r}\sqrt{\frac{3}{2\acute{p}}} \tag{12}$$

where, $RTT_r = RTT + T_b$, and the buffering time T_b is calculated from the transmission time T_D using: $T_b = T_D \times \nu$, and ν is number of packets in the buffer of node i in Figure 3, $T_D = \frac{S}{R_r}$, $R_r = W \times \log(1+\gamma)$, and W is the channel bandwidth. Assuming that $T_b = 0$ for all nodes except the reduced rate node. When node i reduces its rate to maintain its probability of error, ε, the network is congested at this node. The estimated number of packets in the buffer of node i is calculated as follows:

$$\nu = max(cwnd - \zeta, 0). \tag{13}$$

Because SSA-TCP updates its $cwnd$ to ζ, then ν will be equal to 0. In this state $\acute{p}$ can be calculated as in the *Normal* state, but with different probability of interference with PU which is given by $\acute{p}_I = \acute{p}_m \times P_r$, where $\acute{p}_m$ is the probability of at least one node mis-detects PU in the *Reduced* state which is given by $\mathbb{P}(\lambda_L < \xi < \lambda_H$. PU is ON).

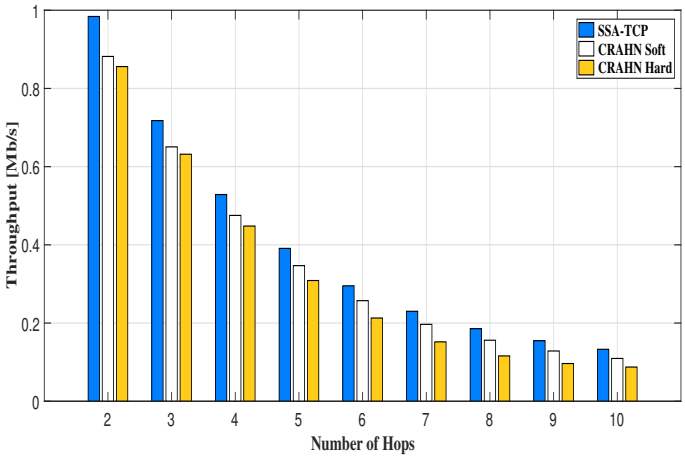

Fig. 4: Throughput versus number of hops in the network.

The probability that the *end-to-end* link in the ***Switching*** state is given by:

$$P_{sw} = \sum_{i=1}^{k}\binom{k}{i}p_{sw}^i(1-p_s-p_{sw})^{k-i} \tag{14}$$

where, $p_{sw} = \pi_{sw}(1-p_s)$, and $\pi_{sw} = \mathbb{P}(\xi \geq \lambda_H)$. Here, $B_{sw} = 0$ since the source ceases transmission.

B. Soft Sensing Unaware protocols

If the transport layer is designed for hard sensing only, *e.g.* TCP CRAHN, all previous states will apply except the *Reduced* state, since it is unaware of PHY layer activity at this state. Consequently, the previous analysis is applicable after some modification to the *Reduced* state. Here, $\nu = cwnd - \zeta$ may be greater than zero, because the transport layer does not update its $cwnd$ based on PHY layer information, which is still updated as in the *Normal* state in Section II-B. This optimistic view of PHY layer rate causes excessive delay and reduces throughput.

C. Hard Sensing protocols

In this case, there is only one threshold of $\xi = \lambda_L$ which is used to decide whether to switch to normal or switching state. Since there is no *Reduced* state, the same analysis applies as in Section III-A, but with $B_r = 0$, and $P_r = 0$. The resulting throughput is:

$$B = \frac{B_n P_n t_n + B_s P_s t_s + B_{sw}\acute{P}_{sw}t_{sw}}{P_n t_n + P_s t_s + \acute{P}_{sw}t_{sw}} \tag{15}$$

Therefore, $\acute{p}_{sw} = \acute{\pi}_{sw}(1-p_s)$, and $\acute{\pi}_{sw}$ is $\mathbb{P}(\xi \geq \lambda_L)$.

IV. Performance Evaluation

In this section, the performance evaluation of the SSA-TCP in comparison with soft sensing unaware protocols and CRAHN is presented. As in [14], it is assumed that, given the state of the PU, the received signal strength is approximately normally distributed with $\xi \sim \mathcal{N}(\mu_i, \sigma_i)$ with $\sigma_1^2 = 160$,

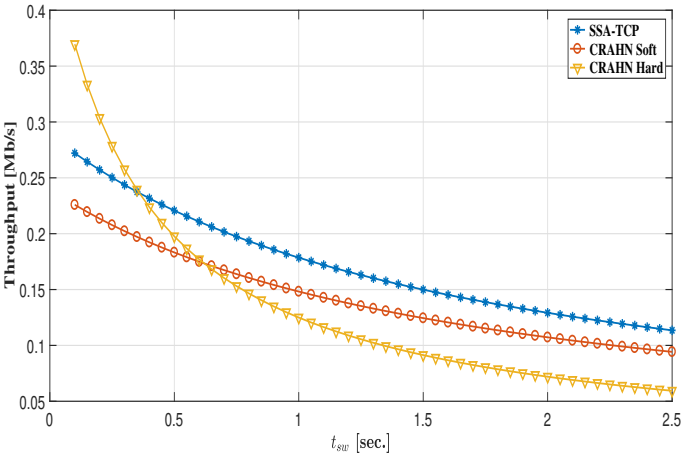

Fig. 5: Throughput versus switching time in seconds.

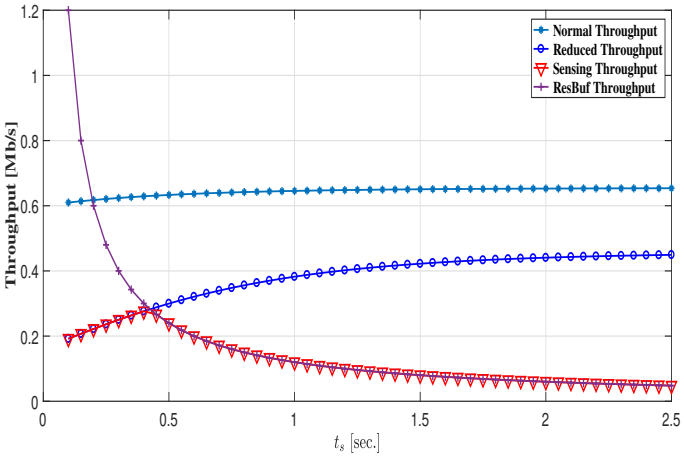

Fig. 6: Throughput versus sensing time in seconds.

$\mu_1 = 40$ when PU is on, and $\sigma_0^2 = 40$, and $\mu_0 = 20$ when PU is off. Given these distributions, and the probability of PU transmitting (P_{on}), it is straightforward to calculate the probabilities of ξ falling in the different decision regions. More specifically, we are able to find π_n, π_r, π_{sw} and p_m for the SSA-TCP, as well as $\acute{\pi}_{sw}$, $\acute{p}_m$ used in Section III-C. The following parameters are used in the sequel to illustrate the relative performance of the three protocols: $k = 10$, $P_{on} = 0.5$, $\lambda_L = 36$, $\lambda_H = 50$, $t_s = 0.3$ sec., $t_n = 5$ sec., $t_r = 2.5$ sec., $t_{sw} = 1.4$ sec., $R = 4$ Mb/s, $R_r = 2.8$ Mb/s, $\varepsilon = 7.5 \times 10^{-5}$, $\mathcal{R}_i = 10$ packets, and $\nu = 6$ packets. These parameters are not prohibitive and serve just as an illustrative example to characterize the model. Other parameters, including T_D and T_b, can be calculated from these parameters as explained in the previous Section.

Figure 4 shows, regardless of the number of hops, SSA-TCP achieves the best throughput followed by CRAHN-soft (cf. Section III-B) then CRAHN-hard (cf. Section III-C). SSA-TCP achieves a gain of up to 15% over CRAHN-hard and up to 13% compared to CRAHN-soft. Soft sensing aware protocols continue transmitting albeit with reduced rate instead of switching channels for longer switching time. The advantage of SSA-TCP over CRAHN with soft sensing is due to the fact that it updates its CW based on the PHY layer information. The figure also shows that the throughput decreases for all protocols with increasing number of hops. This is due to increasing packet loss as k increases as evident in Equation (10). It is also a result of each node adding sensing, switching, or buffering time which increases RTT and consequently decreases throughput.

Figure 5 presents throughput against switching time, t_{sw}. When t_{sw} is very small ($t_{sw} < 0.35$ sec.), CRAHN-hard achieves higher throughput than both SSA-TCP and CRAHN-soft. Indeed, for such small values it is better to search for a free channel than to suffer possible interference for intermediate values of ξ. With increasing t_{sw}, SSA-TCP throughput becomes higher than CRAHN-hard because transmitting with reduced rate and PHY layer adaptation becomes better than ceasing transmission for longer time. For $t_{sw} \geq 0.65$ sec., CRAHN-soft surpasses CRAHN-hard sensing due to its ability to transmit if the signal strength is below λ_H.

In order to analyze the effect of sensing time, t_s, on the average throughput, we begin by quantifying its effect on the contributing terms in Equations (4) and (5). To this end, Figure 6 shows the effect of t_s on throughput in both *Normal* and *Reduced* states; B_n and B_r respectively. It also shows its effect on the throughput due to residual packets in the preceding node, which is captured by the first term in the right hand side of both equations. As shown in the figure, increasing sensing time decreases the probability of interference, and consequently increases the throughput in *Normal*, *Reduced* states. However, the residual term decreases with increasing t_s. The figure also shows $B_{s,r}$ calculated as the minimum from Equation (4), which is shown to increase up to $0.5sec.$ and then decrease.

The impact of sensing time t_s on the average throughput calculated from Eq. (6) is illustrated in Figure 7. It is not surprising to see that B_s follows the same trend of its two components; $B_{s,n}$ and $B_{s,r}$, illustrated by the previous Figure. Notably, the figure shows that the relative performance of the SSA-TCP is superior to both CRAHN-soft and CRAHN-hard. Furthermore, it also asserts the usefulness of soft sensing even if the transport layer is unaware of it use. This is evident from comparing CRAHN-soft to CRAHN-hard.

Figure 8 shows the effect of changing the reduced rate R_r relative to the normal rate R. Increasing R_r increases the average throughput in case of using soft sensing. In CRAHN with hard sensing, the average throughput is constant as there is no reduced rate in this case. To ensure that using the soft sensing scheme is better than hard sensing scheme, the following inequality with $\rho = \frac{R_r}{R}$ must be satisfied.

$$\rho \geq 1 - \frac{t_{sw}}{t_r} \tag{16}$$

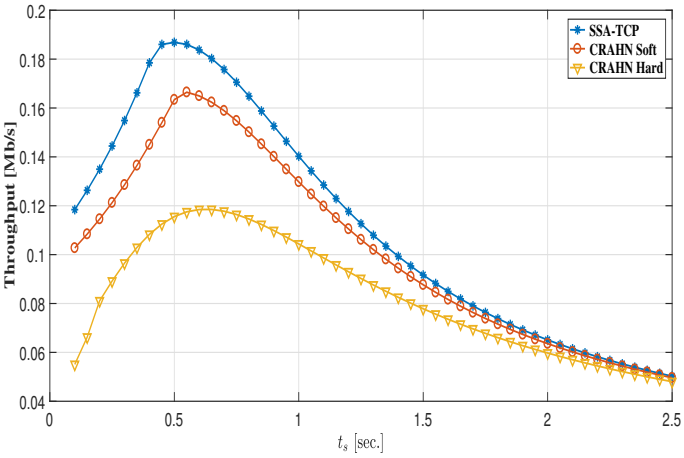

Fig. 7: Throughput versus sensing time in seconds.

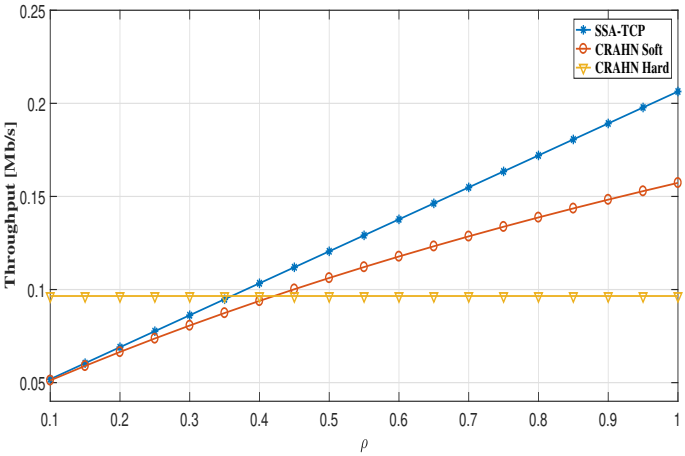

Fig. 8: Throughput versus ρ: ratio of reduced rate to normal rate

It is noticed from all figures that applying soft sensing increases the average throughput of the transport layer. The reason is that using soft sensing enables the SU to transmit even if the PU is on but with reduced rate in order not to interfere with the PU. The SSA-TCP has higher throughput than TCP CRAHN with soft sensing, and TCP CRAHN with hard sensing because the SSA-TCP effectively adapts its CW based on the PHY layer information.

V. Conclusions

A transport layer protocol for CoMANETs using soft sensing has been proposed. In the proposed protocol, the PHY layer of the SU adapts its transmission rate to maximize the capacity. In order for the SU in the proposed protocol to avoid interference with PU, it adapts its CW according to the reduced rate suitable for transmitting with minimum power. Extensive analysis of the average throughput of the proposed SSA-TCP in comparison with soft sensing unaware and hard sensing protocols (CRAHN) has been presented. Results show that the proposed SSA-TCP has higher throughput than CRAHN with soft sensing and CRAHN with hard sensing.

References

[1] I. F. Akyildiz, W. Y. Lee, and K. R. Chowdhury, "CRAHNs: Cognitive Radio Ad Hoc Networks", *Ad Hoc Networks*, vol. 7, no. 5, p 810 – 836, 2009.

[2] T. Henderson, S. Floyd, A. Gurtov, and Y. Nishida, "The NewReno Modification to TCP's Fast Recovery Algorithm", RFC 6582, IETF, April 2012.

[3] Y. Seok, Y. Park, and Y. Choi, "Vertical and Horizontal Flow Controls for TCP Optimization in the Mobile Ad Hoc Networks", in *2003 IEEE* 58^{th} *Vehicular Technology Conference. VTC 2003-Fall*, vol. 4, p 2635–2639, Oct 2003.

[4] H. Marques, J. Leguay, H. Khalif, V. Conan, and D. Lavaux, "Transparent IP Proxy for Tactical Ad Hoc Networks", in *MILCOM 2013 - 2013 IEEE Military Communications Conference*, p 842–847, Nov 2013.

[5] L. A. Grieco and S. Mascolo, "Performance Evaluation and Comparison of Westwood+, NewReno, and Vegas TCP Congestion Control", *SIGCOMM Comput. Commun. Rev.*, vol. 34, p 25–38, Apr. 2004.

[6] K. Sundaresan, V. Anantharaman, H. Y. Hsieh, and A. R. Sivakumar, "ATP: A Reliable Transport Protocol for Ad Hoc Networks", *IEEE Transactions on Mobile Computing*, vol. 4, p 588–603, Nov 2005.

[7] K. R. Chowdhury, M. D. Felice, and I. F. Akyildiz, "TCP CRAHN: A Transport Control Protocol for Cognitive Radio Ad Hoc Networks", *IEEE Transactions on Mobile Computing*, vol. 12, p 790–803, April 2013.

[8] D. Sarkar and H. Narayan, "Transport Layer Protocols for Cognitive Networks", in *2010 INFOCOM IEEE Conference on Computer Communications Workshops*, p 1–6, March 2010.

[9] K. Tsukamoto, S. Koba, M. Tsuru, and Y. Oie, "Cognitive Radio-Aware Transport Protocol for Mobile Ad Hoc Networks", *IEEE Transactions on Mobile Computing*, vol. 14, p 288–301, Feb 2015.

[10] A. O. Bicen, O. Ergul, and O. B. Akan, "Spectrum-Aware and Energy-Adaptive Reliable Transport for Internet of Sensing Things", *IEEE Transactions on Vehicular Technology*, vol. 67, p 2359–2366, March 2018.

[11] S. Mascolo, C. Casetti, M. Gerla, M. Y. Sanadidi, and R. Wang, "TCP Westwood: Bandwidth Estimation for Enhanced Transport over Wireless Links", in *Proceedings of the* 7^{th} *Annual International Conference on Mobile Computing and Networking*, MobiCom '01, (New York, NY, USA), p 287–297, ACM, 2001.

[12] A. K. Al-Ali and K. R. Chowdhury, "TFRC-CR: An Equation-based Transport Protocol for Cognitive Radio Networks", *Ad Hoc Networks*, vol. 11, no. 6, p 1836 – 1847, 2013.

[13] Y. B. Zikria, S. Nosheen, F. Ishmanov, and S. W. Kim, "Opportunistic Hybrid Transport Protocol (OHTP) for Cognitive Radio Ad Hoc Sensor Networks", *Sensors*, vol. 15, no. 12, p 31672–31686, 2015.

[14] S. Srinivasa and S. A. Jafar, "Soft Sensing and Optimal Power Control for Cognitive Radio", *IEEE Transactions on Wireless Communications*, vol. 9, p 3638–3649, December 2010.