HYMAC: an Intelligent Collision Avoiding Dynamic MAC for Wireless Sensor Networks

Adrian Udenze
Dept. of Electronics and Computer Engineering,
Unizik, Awka.

Abstract - **HYMAC is a novel intelligent MAC protocol based on a multi-agent reinforcement learning framework. HYMAC combines a number of desirable attributes for a Wireless Sensor Network MAC protocol including collision avoidance, ability to adapt to changes in network traffic conditions, learning from the environment and making autonomous decisions as well as having minimal computational overheads. Simulation results show that HYMAC outperforms Decentralised MAC, a state of the art MAC protocol, in terms of power efficiency and latency in dynamic traffic conditions.**

Keywords - Reinforcement Learning, Media Access Control, Power Management, Scheduling, Wireless Sensor Networks.

I. INTRODUCTION

Numerous Media Access Control protocols have been developed for Wireless Sensor Networks (WSN). Each one developed to improve on previous ones with regards latency, power consumption and computational overheads. WSN nodes have to stay awake to communicate however staying awake is power intensive and so nodes must shut down their radios during idle periods to save power. Shutting down their radios or going into a sleep state means that nodes cannot communicate thereby increasing packet delays if packets are queued while nodes are sleeping. This system of adopting an active period where nodes communicate and a sleep period where nodes have their radios off termed duty cycling [1] has been shown to be effective in balancing the need for reduced power consumption and acceptable traffic delays. The problem then becomes deciding what the duty cycle should be i.e. how long to stay in a given state. Where traffic conditions are known a priori, the problem is simplified and nodes can have their duty cycles preset by a user. Where traffic conditions are not known a priori then a new tack is required. Artificial intelligence in particular Reinforcement Learning (RL) based agents have been shown to be effective in learning network conditions online and adapting their behaviours autonomously. Thus, a node equipped with such an agent can learn traffic conditions online and adapt its duty cycle accordingly. The work by [2] present simulation results for a RL based MAC, that is able to learn duty cycles online, and a static duty cycled node SMAC [3] under dynamic traffic conditions. The results show a significant increase in power efficiency and decreased latency for the RL based MAC.
WSN MACs as well as sorting out an appropriate duty cycle have to contend with collisions. Collisions occur when two or more nodes communicate over the same channel at the same time thus interfering with each other. RLMAC copes with collisions by adopting Collision Sense Multiple Access (CSMA) [4, 5 and 6] where a node backs off for a random amount of time if a collision is detected before attempting to transmit. This technique effectively deals with a collision when it arises but does nothing to prevent them from happening. Duty cycle based MACs like RLMAC and SMAC are prone to collisions as nodes wake up at the same time following a sleep period and attempt to communicate. When traffic load is heavy and contention for media access heavy, these collisions can become significant leading to increased delays and increased power wastage due to retransmissions. Thus a strategy that is able to synchronise and desynchronise nodes is desirable. Communication in a WSN can be reduced to the two topologies shown in figure 1. The linear topology typifies communication where nodes have to synchronise their duty cycles such that they are all awake at the same time to receive and transmit messages. Nodes have no neighbours on the same hop so synchronisation does not lead to collisions. For the star topology, nodes have neighbours on the same hop and so de-synchronisation is required so that nodes wake up at different times to communicate with the sink node. Decentralised MAC [1] DECMAC does just this. By synchronising and desynchronising nodes, simulation results show increased power efficiency and reduced latency when compared to nodes without this mechanism such as RLMAC. On the other hand however, DECMAC is not able to adjust to varying network conditions as it makes no attempt to learn duty cycles online, instead duty cycles are prefixed by the user.
Hybrid MAC (HYMAC) has been developed to combine the duty cycle learning mechanism of RLMAC with the collision avoidance mechanism of DECMAC to form a new MAC that is able to cope with dynamic traffic conditions and synchronise and desynchronise nodes such that collisions are minimised. Simulation results show that latency and power efficiency results are better for HYMAC when compared to DECMAC.

The rest of the paper is organised as follows. First a literature review is presented of relevant research in MAC protocols. Then HYMAC is presented and discussed followed by a discussion of simulation results. Conclusions come at the end of the paper.

Figure 1: Star and Linear topologies
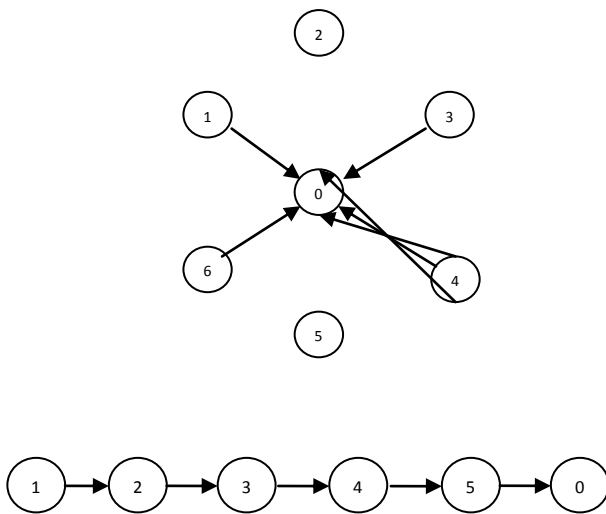
II. LITERATURE REVIEW

MAC protocols can be distinguished into contention and schedule based protocols; and hybrid protocols where the best qualities of contention and schedule based protocols are combined. Scheduled MAC protocols are predominantly TDMA based, where competing nodes are given time slots for channel access. Works presented in [7, 8, 9, 10 and 11] all give examples of TDMA based scheduled MAC protocols. The key strength of TDMA based MAC is high energy efficiency and the potential of eliminating collisions. These strengths are however at the expense of limited scalability and adaptability to changes in the network. Also, control is often centralized in order to limit control packet overhead, which leads to a hierarchical organization of nodes or cluster formations. The result of which is a limited support of peer to peer communication. A CDMA alternative to TDMA scheduled protocols is discussed in [12] however, CDMA is predominantly used in cellular networks due to Multiple Access Interference (MAI).

Contention based MAC protocols do not attempt to eliminate collisions completely but aim to avoid and recover from them when they occur, examples include ALOHA [13], CSMA [14] and other variations such as CSMA/CA [4, 5, 6]. Contention based protocols have increased flexibility and scalability; in addition, nodes can operate on a peer to peer basis. Control overhead is also significantly reduced compared to scheduled protocols. However, the fundamental disadvantage of contention based protocols is energy inefficiency. Collisions may occur, nodes overhear other nodes and idle listening prevention is not implicit. PAMAS [15] attempts to reduce overhearing by introducing duty cycles. A separate signalling channel is used to broadcast a busy signal when the channel is being used thus preventing other nodes from transmitting and also, nodes can turn off their radios for the duration of the transmission reducing overhearing.

Hybrid protocols attempt to combine the desirable qualities of schedule and contention based protocols. A hybrid CDMA / FDMA protocol is presented in [16] in which the authors use FDMA to combat MAI in CDMA protocols. A hybrid CDMA / TDMA protocol is presented [17], nodes maintain a TDMA based schedule for each communication link and CDMA is used to prevent collisions due to neighbouring nodes communicating simultaneously over other links. Bluetooth, [18], is another hybrid CDMA / TDMA protocol where collisions within clusters are avoided using CDMA and communication between cluster heads use TDMA.

Duty cycle based techniques can also be thought of as a hybrid between scheduling and contention based protocols. The general form is for neighbouring nodes to maintain schedules of active and sleep times. During sleep times, nodes power off their radios and go to sleep. In the active period, CSMA/CA is used for media access. The IEEE 802.11 protocol, LANWAN [6] has a power saving mode in which nodes adopt a duty cycle. Nodes are required to wake up at fixed points in time to listen out for beacons which contain messages to nodes for which messages are queued. Variations of this technique aim to improve the protocol by reducing communication overheads. Studies in [19] and [20] analyse and discuss impacts of the beacon interval and window sizes on power saved. The IEEE 802.11 protocol also includes MAC in the form of carrier sensing where nodes exchange Request to Send / Clear to Send (RTS/CTS) messages before proceeding with packet transmissions.

An improvement on the IEEE 802.11 protocol is the S-MAC protocol [3], where nodes form virtual clusters around common schedules. By maintaining different schedules for each cluster of nodes, only nodes within a cluster have to wake up at the schedule determined times. This is in contrast to the IEEE 802.11 protocol, where all nodes have to wake up at fixed points in time. T-MAC [21] improves on S-MAC by providing a timeout mechanism in which nodes following a short period of listening for a RTS/CTS exchange can go to sleep if none is detected. As such, more energy is saved compared to the fixed active and sleep duty cycles of S-MAC. P-MAC [22] improves on T-MAC and S-MAC by tailoring each node's duty cycle to its rate of traffic. Thus nodes with higher rates of traffic have longer duty cycles compared to those with less traffic, thereby improving throughput within the network. RLMAC [2] is a dynamic protocol that, similar to P-MAC, tailors each node's schedule according to its traffic and that of its neighbours. In contrast to P-MAC however, RLMAC does not require information of traffic flow in advance of scheduling. The RLMAC protocol learns by interacting within the network environment near optimal duty cycles which makes it able to adapt to dynamic events in a network. In [1] the authors present another RL based protocol (DECMAC) that learns optimal slots within a time frame of when to transmit and receive messages. The result of which is a reduction in number of collisions and retransmissions thus improving throughput.

III. HYMAC

A. Introduction

HYMAC is a novel MAC protocol that improves on DECMAC. Duty cycles in DECMAC are manually set which implies that traffic conditions are known a priori. For networks in which conditions are not known a priori, DECMAC duty cycles will not be optimal thus leading to either increased latency due to nodes going to sleep too early or increased idle listening due to nodes staying awake for too long.

The design objectives of HYMAC were to develop a MAC protocol able to avoid collisions between nodes thus reducing energy wasted in retransmissions and also reduce delays as well as reduce power wasted idle listening where a node's radio is on but neither receiving nor transmitting packets. Attention is also placed on networks in which events are dynamic and not known a priori. Duty cycling has been proven as discussed in the review section to be effective in reducing idle listening. In addition nodes equipped with RL based agents have been shown to be effective in coping with dynamic events in a network. HYMAC adopts both proven techniques. Time is divided into frames and each frame divided into slots. A node may be awake for a period within a frame and spend the rest of the time sleeping after which the cycle repeats. HYMAC is based on a RL agent and it is the agent's job to make decisions on which group of consecutive slots within a frame to stay active in. To do this the agent must decide on a start slot or wake up time and an end slot or sleep time. Put in another form a HYMAC agent decides which time slot to wake up in and how long to stay awake for. Furthermore, for a successful transmission to occur two nodes have to cooperate thus two agents have to cooperate.

B. HYMAC reward

The reward mechanism for a RL agent is the driving force to rational behaviour. A comprehensive guide to RL is presented in [23]. Rewards, actions update rules and Q values have also been discussed in [2 and 1]. A HYMAC agent maintains a Q value Q^i for each slot in a frame Q_s^i. The value of the slot depends on four events taking place, receiving, transmitting, overhearing and idle listening. If a node receives a message during a given time slot it receives a reward of 1, if it transmits a message it receives a reward of 1, if it overhears a transmission it receives 0 and if nothing happens it receives 0.

The Q values are summed for each consecutive slot in a window which drives a node to take actions that lead to slots in which transmission or reception takes place, equation 1.

$$\hat{Q} = \sum_{j=0}^{C} Q_{s+j}^i \qquad (1)$$

Next, this value is made a ratio of the total window size to encourage nodes to select windows in which all the time is spent either receiving or transmitting. Thus if a node is active for C time slots, Q value is given below in equation 2

$$Q = \hat{Q}^w/C \qquad (2)$$

w above is a weighting factor which determines the preference of delay over power efficiency. The higher the weighting the more nodes are encouraged to stay awake and transmit as many packets as possible.

The first two reward mechanisms combined incentivises nodes to work together to achieve a common goal thus cooperation is achieved and delay reduced. The second two rewards discourage nodes from taking actions that lead to power wastage. To begin with, the wake up slot of nodes and the active period within the frame are randomly initialized after which Q values are update using the update rule proposed in [1] for a stateless system:

$$Q_s^i \leftarrow (1 - \alpha).\breve{Q}_s^i + \alpha.r_{s,e}^i \qquad (3)$$

$r_{s,e}^i$ is the reward received at slot s following an event, α is a learning rate which is left constant to take into account dynamic events in the network. A greedy approach is adopted where nodes choose actions that have the highest Q values with a high probability. Eventually Q values converge such that actions taken are optimal i.e.

$$A' = \arg\max \hat{Q}^w/C \qquad (4)$$

IV. SIMULATION RESULTS

Simulations were carried out to investigate power efficiency and latency of HYMAC against DECMAC in a network where traffic rates are not known a priori. DECMAC duty cycles have to be preset as in SMAC whereas HYMAC learns optimal duty cycles on line. By keeping a constant learning rate and making the reward a function of the ratio of time spent productively to total time spent active, HYMAC can adapt to changes in the network thus saving power and reducing latency in heavy traffic conditions. To this end two topologies were investigated, namely, linear topology where packets are generated at one end of a chain of five nodes before the sink node and a star topology where nodes within a sink's vicinity compete for channel access, figure 1.

The simulation parameters used were as follows. A frame is 1sec long and each frame is split into 100 slots of 10ms. A reward of 1 unit is received for a successful transmission or reception in a time frame and 0 otherwise. The duty cycle of DECMAC was set to set to different cycles and data rates varied between 1 packet of 50 bytes every second and 10 packets of 50 bytes every second to simulate different rates of traffic. Transmission time for each packet was set to 20ms. Simulations were carried out for 5000 seconds and results averaged over 50 cycles for each topology. A learning rate of 0.1was used for the learning algorithm. The weight w was set to 1.1 to reflect a slight emphasis on reduced latency. A sample of the simulation results where DECMAC has the duty cycle set to 40% and data rates varied between 1 and 10 packets per second is presented below. Latency results are presented along with

energy efficiency results measured in number of transmission and receptions against energy consumed. Energy efficiency results for the linear topology are presented in figure 2. It can be seen that for low traffic rates, HYMAC outperforms DECMAC significantly. This is because DECMAC stays awake for longer than is necessary whereas HYMAC learns an optimal duty cycle thus saving power by going to a low power mode when messages have been exchanged. For higher traffic rates DECMAC and HYMAC perform similarly. Figure 3 shows results for packet delay between DECMAC and HYMAC. At low traffic rates both have the same performance as both are up for long enough to service requests. As traffic rates increase DECMAC reaches its optimal performance, at 5 packets per second, and then delays start to build up as DECMAC goes to sleep too early thus causing a backlog of un-serviced requests. HYMAC on the other hand is able to adjust its duty cycle appropriately reducing delays. Figure 4 shows the power efficiency performance of DECMAC against HYMAC for a star topology. Again for low traffic rates DECMAC is inefficient as it stays up for too long with its fixed duty cycle. As rates increase efficiency starts to build up and performance matches that of HYMAC which is able to adopt its duty cycle autonomously. Figure 5 shows delay results and again HYMAC outperforms DECMAC at high traffic rates due to HYMAC's ability to adjust its duty cycles.
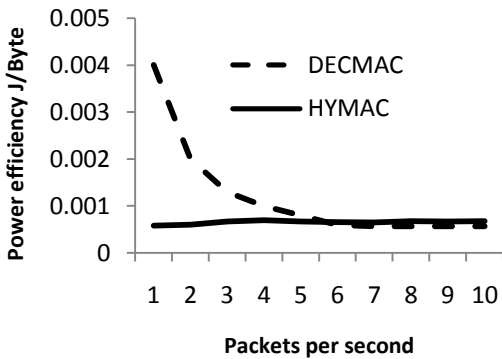
Figure 3: Linear topology delay

Figure 4: Star topology power efficiency
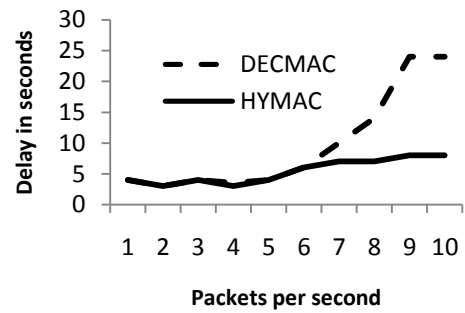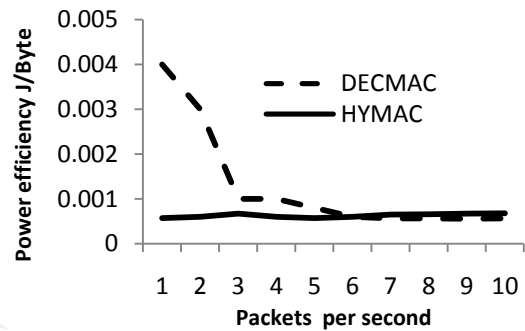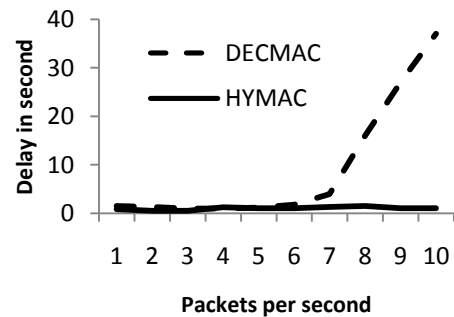

Figure 5: Star topology delay

Figure 2: Linear topology power efficiency

V. CONCLUSION

A novel MAC protocol HYMAC was developed to cope with dynamic WSNs where events are not known a priori and are also dynamic. HYMAC has a reward function that encourages nodes to act together to successfully exchange messages, to adapt a nodes duty cycle to changes in rates of traffic as well as synchronise and desynchronise nodes so that delay in transmitting packets as well as energy consumed are minimised. Simulation results were presented which showed better latency and power efficiency results for HYMAC when compared to DECMAC. The work was carried out for networks with periodic data patterns, future work will be aimed at investigating the performance of HYMAC in event driven networks.

REFERENCES

[1] Mihaylov, M., Le Brogne, Y., Tuyls, K., and Nowe, A. (2012), 'Decentralised reinforcement learning for energy-efficient scheduling in wireless sensor networks', International Journal of Communication Networks and Distributed Systems, Vol9, no 3/4, pp 207-224.

[2] Zhenzhen, L. and Itamar, E. (2006), 'RL-MAC: a reinforcement learning based MAC protocol for wireless sensor networks', International Journal of Sensor Networks, Vol. 1, Issue 3/4 September 2006, pp. 117-124

[3] Ye, W., Heidemann, J. and Estrin, D. (2004) 'Medium access control with coordinated adaptive sleeping for wireless sensor networks', IEEE/ACM Transactions on Networks, Vol. 12, No. 3, pp.493–506.

[4] Karn, P. (1990) 'MACA - A new channel access method for packet radio' In. Proceedings of Amateur Radio Ninth Computer Networking Conference, ARRL, pp. 134-140.

[5] Bharghavan, V., Demers, A., Shenker, S. and Zhang, L. (1994) 'MACAW: A media access protocol for wireless LANs', Proceedings of the conference on Communications architectures, protocols and applications, London, pp. 212 – 225.

[6] LAN WAN Standards Committee of the IEEE Computer Society (1997) 'Wireless LAN medium access control (MAC) and physical layer (PHY) specification' IEEE, New York, NY, USA, IEEE Std 802.11-1997.

[7] Hohlt, B., Doherty, L. and Brewer, E. (2004) 'Flexible power scheduling for sensor networks' In Proceedings of the third international symposium on Information processing in sensor networks, pp.205 – 214.

[8] Hohlt, B., and Brewer, A. (2006) 'Twinkle: Network power scheduling for tinyos applications" In IEEE international conference on distributed computing sensor systems, San Francisco, USA, 2006.

[9] Arisha, K. A., Youssef, M. A. and Younis, M. F. (2002) 'Energy-aware TDMA based MAC for sensor networks', Proceedings of IEEE IMPACCT 2002, NY, USA.

[10] Sichitiu, M. L. (2004) 'Cross-layer scheduling for power efficiency in wireless sensor networks' In IEEE Infocom. 3, pp. 1740-1750.

[11] Heinzelman, W. R., Chandrakasan A. and Balakrishnan, H. (2000) 'Energy-efficient communication protocols for wireless microsensor networks', In proceedings of the Hawaii International Conference on Systems Sciences.

[12] Proakis, J. G., Salehi, M. and Bauch, G. (2003) 'Contemporary Communication Systems using MATLAB and Simulink' 2nd ed. Southbank, Vic. Thomson/Brooks/ Cole.

[13] Abramson, N. (1970) 'The ALOHA system – Another alternative for computer communications', Proceedings of the November 17-19, 1970, Joint Computer Conference Houston, Texas, pp. 281-285.

[14] Kleinrock, L. and Fouad, T. (1975) 'Random access techniques for data transmission over packet-switched radio channels' Proceedings of the national computer conference and exposition, Anaheim, California, pp. 187-201, May 19-22.

[15] Singh, S. and C. S. Raghavendra, C. S. (1998) 'PAMAS: Power aware multi-access protocol with signalling for ad hoc networks', ACM Computer Communications Review, vol. 28, no. 3, pp. 5–26.

[16] Liu, B. H., Chou, C. T., Lipman, J. and Jha, S (2005) 'Using frequency division to reduce MAI in DS-CDMA wireless sensor networks',Wireless Communications and Networking Conference, IEEE Vol. 2, 13-17 March, pp. 657 – 663.

[17] Sohrabi, K. and Pottie, G. J. (1999) 'Performance of a novel self-organizing protocol for wireless ad hoc sensor networks', In proceedings of the IEEE 50th Vehicular Technology Conference, pp 1222-1226.

[18] Bluetooth SIG Inc., (2001) 'Specification of the Bluetooth system: Core', http://www.bluetooth.org/.

[19] Jung, E. S. and Vaidyan N. (2002) 'Energy efficient MAC protocol for Wireless LANs.In proceedings of IEEE Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM, 3, pp. 1756 – 1764.

[20] Woesner, H., Ebert, J. P., Schlager, M. and Wolisz, A. (1998) 'Power-Saving Mechanisms in Emerging Standards for Wireless LANs: The MAC Level Perspective' IEEE Personal Communications, Vol. 5, issue 3, pp.40-48.

[21] Dam, T.V. and Langendoen, K. (2003) 'An adaptive energy efficient MAC protocol for wireless sensor networks', SenSys '03: Proceedings of the first International Conference on Embedded Networked Sensor Systems, New York, NY: ACM Press, pp.171–180.

[22] Zheng, T., Radhakrishnan, S. and Sarangan, V. (2005) 'Pmac: an adaptive energy-efficient MAC protocol for wireless sensor networks', Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05), pp.65–72.

[23] Sutton, R. S. and Barto, A. G. (1998) Reinforcement Learning: An Introduction, Cambridge MA: MIT Press.