

01 Apr 2009

Dynamic Channel Allocation in Wireless Networks Using Adaptive Learning Automata

Behdis Eslamnour

Maciej Jan Zawodniok
Missouri University of Science and Technology, mjzx9c@mst.edu

Jagannathan Sarangapani
Missouri University of Science and Technology, sarangap@mst.edu

Follow this and additional works at: https://scholarsmine.mst.edu/ele_comeng_facwork



Part of the [Computer Sciences Commons](#), and the [Electrical and Computer Engineering Commons](#)

Recommended Citation

B. Eslamnour et al., "Dynamic Channel Allocation in Wireless Networks Using Adaptive Learning Automata," *Proceedings of the IEEE Wireless Communications and Networking Conference, 2009*, Institute of Electrical and Electronics Engineers (IEEE), Apr 2009.

The definitive version is available at <https://doi.org/10.1109/WCNC.2009.4917686>

This Article - Conference proceedings is brought to you for free and open access by Scholars' Mine. It has been accepted for inclusion in Electrical and Computer Engineering Faculty Research & Creative Works by an authorized administrator of Scholars' Mine. This work is protected by U. S. Copyright Law. Unauthorized use including reproduction for redistribution requires the permission of the copyright holder. For more information, please contact scholarsmine@mst.edu.

Dynamic Channel Allocation in Wireless Networks using Adaptive Learning Automata

Behdis Eslamnour

Department of Electrical and
Computer Engineering
Missouri University of Science and
Technology
Rolla, MO, USA
ben88@mst.edu

Maciej Zawodniok

Department of Electrical and
Computer Engineering
Missouri University of Science and
Technology
Rolla, MO, USA
mjzx9c@mst.edu

S. Jagannathan

Department of Electrical and
Computer Engineering
Missouri University of Science and
Technology
Rolla, MO, USA
sarangap@mst.edu

Abstract—The bandwidth utilization of a single channel-based wireless networks decreases due to congestion and interference from other sources and therefore transmission on multiple channels are needed. In this paper, we propose a distributed dynamic channel allocation scheme for wireless networks using adaptive learning automata whose nodes are equipped with single radio interfaces so that a more suitable channel can be selected. The proposed scheme, Adaptive Pursuit Reward-Inaction, runs periodically on the nodes, and adaptively finds the suitable channel allocation in order to attain a desired performance. A novel performance index, which takes into account the throughput and the energy consumption, is considered. The proposed scheme is adaptive in the sense that probabilities in the each step are updated as a function of the error in the performance index. The extensive simulation results in static and mobile environments provide that using the proposed scheme for channel allocation in the multiple channel wireless networks significantly improves the throughput, drop rate, energy consumption per packet and fairness index.

Index Terms— *adaptive reward-inaction, channel allocation, learning automata, wireless ad hoc sensor networks.*

I. INTRODUCTION

IT is widely believed that the wireless networks are being limited by the lack of the available spectrum, and at the same time the spectrum is not efficiently utilized. Spectrum utilization can be improved using spatial techniques, frequency, modulation techniques, *etc.* As a consequence, newer concepts such as software-defined radios and cognitive radios were made possible [1]. While the cognitive radios are not limited to spatial and temporal spectrum utilization, the spatial channel reuse approach in wireless networks has been vastly investigated [2] - [6].

The bulk of the research on multiple channel allocation is notably done for mesh networks [3], WLANs with infrastructure [4], cellular networks [6] and cognitive radio networks [5]. The multi-channel allocation problem has been investigated for the networks in which the nodes are equipped with either multiple-radio interface [7] or single-radio interface [2][4][8]. In the single-radio approach, the radios switch between the channels frequently in order to minimize

interference and collision between the simultaneous transmissions in the same communication range. Usually in this approach, all the nodes periodically switch to a common channel for channel co-ordination, and then switch to different data channels to conduct the simultaneous transmissions. Therefore the switching delay (80-100 μ s [2]) becomes one of the overheads increasing the network end-to-end delay. Additionally, synchronization is required in these schemes.

In the case of multiple-radio interface approach, usually one interface is dedicated to the control signals, and the remaining channels are allocated for simultaneous transmission of data thus increasing temporal and spatial spectrum utilization and not requiring synchronization. Further, utilizing multiple radios reduces the need for frequent channel switching, and hence the switching overhead is significantly less than that in the single-radio approach. However, the cost of additional radios and their energy consumption must be taken into account.

By contrast, in this paper, we propose a distributed dynamic channel allocation scheme for wireless networks and in particular wireless sensor networks whose nodes are equipped with single radio interface due to their low cost requirement. Therefore, synchronization is required in this scheme. The periodic nature of this algorithm makes it dynamic and enables the channel allocation to adapt to the topographic changes, possible loss of some channels, mobility of the nodes, and the traffic flow changes. The adaptive pursuit reward-inaction learning algorithm runs periodically on the nodes, and adaptively finds the optimum channel allocation that provides the desired performance (or closest to the desired performance). Unlike the linear and nonlinear schemes in which the reward and penalty values were functions of the probabilities, we examine an adaptive updating scheme in which the reward and penalty values are functions of the error between the desired and the estimated performance of the current channel allocation. By selecting realistic desired performance metric, the convergence of the algorithm is guaranteed.

II. METHODOLOGY AND ALGORITHM

A. Methodology

In the proposed algorithm, the nodes periodically switch between the control stage, T_c , and data transmission stage, T_d (See Figure 1). Each data transmission period, T_d , is comprised of the individual time slots, T_s . As an initial assumption, we consider peer-to-peer networks in which all nodes are equipped with a single radio. We also assume that routes have been established by a proactive routing protocol such as optimal link state routing (OLSR) [12] or optimal energy delay routing (OEDR) [13]. During T_c , all nodes are on one common channel to communicate the control signals. It is possible that one or more of the channels get highly affected by external interference and the network would lose these channels temporarily or permanently.

In order to maintain the network connectivity in the sense of exchanging the control signals, we propose having a unique sequence of all the channels. In the event of a loss of a control channel, the nodes would try the next channel in the sequence as the control channel during T_c . The control signal carries the schedule of the time slots for the links in the subsequent data transmission period. During the time scheduling, groups of non-intersecting links are scheduled for each T_s time slot. Also broadcast communications and route discovery are performed during T_c period. After the T_c stage, the data transmission stage, T_d , begins. During each T_s time slot of T_d , channels are allocated to the links previously assigned to the T_s . The channel allocation algorithm is an iterative algorithm during which the channel allocation is refined. Due to the iterative nature of the algorithm, each T_s is divided into smaller time slots, T_{mini} , separated by T_g – guard bands. The probabilities and parameters of the channel allocation algorithm are updated for each link from one T_{mini} to the next.

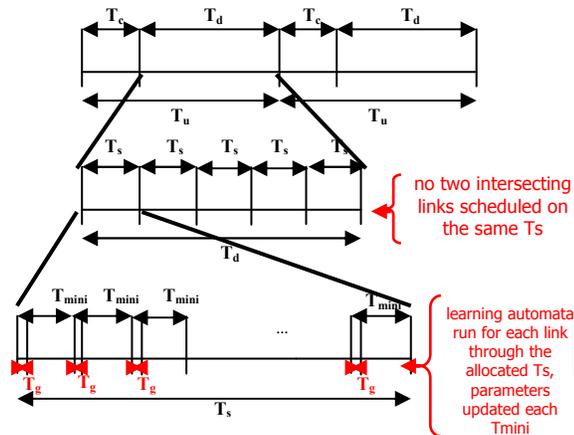


Figure 1. Control and data time slots within the data transmission period.

By periodically repeating the T_c and T_d stages, the channel allocation becomes dynamic. In addition, the network can adapt to the topographic changes, mobility of the nodes, and the changes in the traffic flow. Also in the event of control channel, C_c , loss the next channel in the sequence will be used as the control channel. It must be noted that this sequence is a

common knowledge among all the nodes in the network. Any eligible external node that tries to join the network would send out join-request signals periodically and listen in the intervals. It would be able to join the network during one of the T_c periods, and obtain the sequence and other necessary information about the network.

We also propose using the control channel as one of the available channels for data transmission during the T_d period. By utilizing this additional channel during T_d instead of dedicating it to the control signals and using it only during T_c , the spectrum utilization can be increased.

B. Algorithm

During each T_s , the learning algorithm is run on each transmitter node, i , separately. We first use the Adaptive Pursuit Reward-Inaction (PRI) which is an extended version of Distributed PRI [9], [10]. Unlike the DPRI, in the Adaptive PRI scheme the update value, $\theta(k)$, of the probabilities is not a constant anymore. The update value of the probability is now a function of the error, $\Delta(k)$, of the performance metric.

We chose DPRI algorithm because of the faster convergence provided by it [9]. The Adaptive PRI algorithm is presented in Section B.1. However, it appears that depending on the conditions that determine whether the environment response is satisfactory or unsatisfactory, the channel allocation on some links might always result unsatisfactory response. This would result in ‘left-out’ links, whose channel selection probabilities are not updated due to the ‘reward’ property of the algorithm.

In order to eliminate this issue, we propose the Adaptive Pursuit Reward-Penalty (PRP) learning scheme. The ‘reward’ behavior of this scheme is the same as the Adaptive PRI. On the other hand, in the case of unsatisfactory environment response for a channel selection, the probability of selecting that channel (if that channel is not the channel with the highest performance among the channels) is decreased, and the probabilities of selecting the other channels are increased. Although this scheme eliminates the ‘left-out’ links problem, it has a rather slower convergence because of increasing the probabilities of some of the non-optimal channels in the ‘penalty’ scheme.

The performance metric of the network used in this paper was defined as $\phi^* = \left(\frac{H}{E}\right)_{desired}$ where H is the desired percentage

of the successful transmissions and E refers to the desired consumed energy per one successful packet transmission. By this definition, the unit of the performance metric ϕ^* becomes packets/joule. Therefore, by selecting a realistic desired performance metric, the objective is to find the optimum channel allocation that provides a higher performance in terms of throughput defined in terms of a target value. A large value of ϕ^* indicates successful transmission of more packets. Hence, this performance metric covers both the throughput and the energy efficiency of the network.

The nonlinear pursuit reward-inaction scheme is given by:

1) Initially, the probability of selecting any of the channels, j , on any node, i , $p_i^j(0)$, is set to $1/N$, where N is the number of available channels.

2) Select a channel according to the probability distribution, $p_i^j(k)$. Transmit packets during the transmission interval.

3) Based on the measured feedback, update $J_i^j(n)$, $L_i^j(k)$ and $e_i^j(k)$. $J_i^j(n)$ is the percentage of successful transmissions on node i while using channel j , and $L_i^j(k)$ is the number of times that channel j was selected for node i from time 0 till k .

4) If $L_i^j(k) \geq M$, update $\hat{H}_i^j(k)$, $\hat{E}_i^j(k)$ and $\hat{\phi}_i^j(k)$ and continue on step 5. Otherwise, go to step 7.

$\hat{H}_i^j(k)$ is the average estimated throughput over a window of M , $\hat{E}_i^j(k)$ is the average estimated consumed energy over a window of M , and $\hat{\phi}_i^j(k)$ is the estimated performance of channel j for node i at time k .

$$\hat{H}_i^j(k) = \frac{1}{M} \sum_{n=L_i^j(k)-M+1}^{L_i^j(k)} J_i^j(n) \quad \hat{E}_i^j(k) = \frac{1}{M} \sum_{n=L_i^j(k)-M+1}^{L_i^j(k)} e_i^j(n) \quad \hat{\phi}_i^j(k) = \frac{\hat{H}_i^j(k)}{\hat{E}_i^j(k)}$$

$$5) \beta_i^j(k) = \begin{cases} 0, & \text{if } \frac{\phi^* - \hat{\phi}_i^j(k)}{\phi^*} < \delta \\ & \text{(satisfactory response)} \\ 1 & \text{otherwise} \\ & \text{(unsatisfactory response)} \end{cases} \quad (1)$$

where $\beta_i^j(k)$ is environment response for selecting channel j by node i at time k .

$\left\{ \begin{array}{l} \text{if } \beta_i^j(k) = 0, \text{ the automaton will be rewarded} \\ \text{if } \beta_i^j(k) = 1, \text{ the automaton will not be rewarded} \end{array} \right.$

6) Detect the channel index, \hat{m}_i , that provides the best estimated performance, $\hat{\phi}_i^{\hat{m}_i}(k)$. Update the probabilities if the environmental response was satisfactory.

$$\text{If } \beta_i^{\hat{m}_i}(k) = 0, \begin{cases} p_i^{\hat{m}_i}(k+1) = 1 - \sum_{j=1, j \neq \hat{m}_i}^N p_i^j(k+1) \\ p_i^l(k+1) = p_i^l(k) - \theta(k) \quad \forall l \neq \hat{m}_i \end{cases} \quad (2)$$

$$\text{where } \theta(k) = \begin{cases} \gamma \cdot |\Delta(k)| / \phi^*, & \text{if } -\delta < \Delta(k) / \phi^* \\ \lambda \cdot |\Delta(k)| / \phi^* & \text{otherwise} \end{cases}$$

such that $0 \leq \theta(k) < 1$ and $\Delta(k) = \phi^* - \hat{\phi}_i^{\hat{m}_i}(k)$,

7) Continue to the next iteration, step 2.

Next, the proof of convergence of the algorithm is presented. The theorems and proofs follow the general method used in [9]. Theorem I establishes that for each node that is running the algorithm, if after a certain time, the channel allocation results in a better performance for one channel compared to the other channels, the probability of selecting that channel approaches one. Theorem II establishes that for each node and each channel, there exists a time that the channel has been selected by the node for at least M times. This guarantees having the average values of the throughput,

delay and consumed energy, which are required for the calculation of the performance.

Theorem I: Suppose there exists an index m_i and a time instant $k_0 < \infty$ such that $\hat{\phi}_i^{m_i}(k) > \hat{\phi}_i^j(k)$ for all j such that $j \neq m_i$ and all $k \geq k_0$. Then there exists γ_0 and λ_0 such that for all resolution parameters ($\gamma < \gamma_0, \lambda < \lambda_0$), $p_i^{m_i}(k) \rightarrow 1$ with probability 1 as $k \rightarrow \infty$.

Proof: See Appendix A.

Theorem II: For each node i and channel j , assume $p_i^j(0) \neq 0$. Then for any given constant $\delta_0 > 0$ and $M < \infty$, there exists $\gamma_0 < \infty, \lambda_0 < \infty$ and $k_0 < \infty$ such that under the discrete pursuit reward-inaction algorithm, for all learning parameters $\gamma < \gamma_0$ and $\lambda < \lambda_0$ and all time $k > k_0$:

Pr{each channel chosen by node i more than M times at time k } $\geq 1 - \delta_0$.

Proof: See Appendix A.

IV. SIMULATION RESULTS AND DISCUSSIONS

In this section, we present the numerical results of running the adaptive PRI learning algorithm on a set of peer-to-peer wireless networks with varying traffic, mobility, and number of nodes using network simulator NS-2. The networks are consisted of 50 single-radio wireless nodes located in an area of 100m×100m, while the communication range of the nodes are at 250m. As a result, a dense network topology is created where a single channel is not able to provide sufficient quality of service (QoS). Traffic is generated by a constant bit rate (CBR) sources with data rates equal to 2 Mbps and packet size equal to 1024 bytes. The simulations considered networks with up to 11 orthogonal channels whose bandwidth is set to 11 Mbps. The objective of the multi-channel protocol is to allocate the available channels to the links such that the performance converges to a desired value as defined in (0).

The target value ϕ^* and the updated parameters were set for different scenarios such that the desired performance is achievable. The nodes start without preferred channel and switch between channels until they find the one that provides the desired performance. The width of the moving average window, M , was selected to be 5.

A. Static Scenario

This simulation scenario considers single time slot duration, T_s , where all nodes are contending for the channels. The network topology is static for the whole simulation duration in order to observe the convergence time of the presented schemes.

Figure 2 illustrates an example of channel switching and allocation using the Adaptive PRI for a randomly selected simulation with 50 nodes and 10 channels. Initially, the flows randomly switch between all available channels since each link starts with equal probability of selecting the channels. When the nodes collect statistical results from the initial iterations, they evaluate the performance for each channel and

start updating the channel selection probabilities. Over time the nodes learn if the initial channel selection is successful. If the desired performance is not achieved, they will switch to other channels and evaluate alternative channel allocations. Once the desired performance is met the nodes reinforce the channel selection by adjusting corresponding probabilities. Afterwards, the channel switching stops since nodes find the adequate channels thus resulting in collision and packet drop.

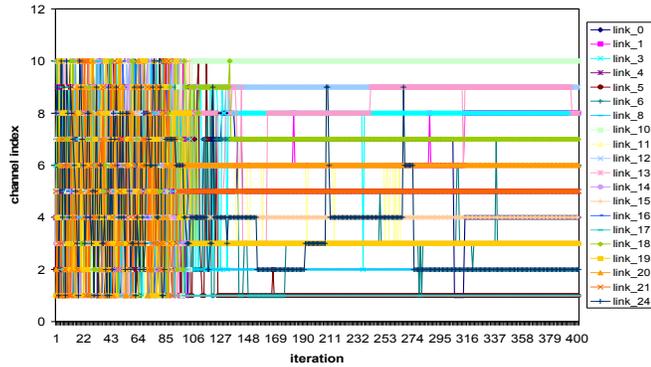


Figure 2. The converged channel allocation for the 21 links in a network of 50 peer-to-peer nodes (25 links), using the Pursuit Reward-Inaction learning automata.

The throughput (not shown) is low when the nodes frequently switch during convergence phase since often two or more nodes will select the same channel thus resulting in collision and packet drop. Once the appropriate channel allocation is found, the channel switching stops and the throughput increases to the maximum level.

B. Static Scenario – starting flows at different times

The learning algorithm was run on the networks of 50 nodes with up to 11 orthogonal channels. Three flows start at second 2, then seven more flows start at second 3 and finally fifteen more flows start at second four. The standard 802.11 protocol was also run on the networks to compare its performance to the performance of the learning algorithms. This was done by a) using a single channel, and b) using 10 channels and randomly allocating them to the links. For each case, the simulation was repeated using 10 random scenarios, and the average of the 10 repeated simulations were used in result analysis. The achieved throughput by applying the different methods is presented in Table I.

It is noticed that as the number of channels used in the Adaptive PRI learning schemes is increased, the throughput is significantly increased compared to the single-channel 802.11 scenario. The increased throughput is provided by the additional capacity of the additional channels. For the case of 25 flows, the Adaptive PRI with 10 data channels provides an improvement of 13 times in throughput compared to a single-channel 802.11. When there are 25 flows in the network and only one channel is provided, the network is so congested that it provides a throughput of only 3 for the 25 flows.

However, when the Adaptive PRI is used on 10 channels, it provides a higher capacity though not the capacity required to

eliminate the congestion. The capacity provided by the 10 channels is almost 10×capacity of each channel. The capacity of each channel for data packets in 802.11 is almost half of the channel bandwidth. We had chosen a standard channel bandwidth of 11Mbps in the simulations. Therefore the total throughput of 39.58 Mbps is reasonable compared to the total capacity of almost 50 Mbps, since there is a noticeable congestion in the network. Also for the same case of 25 flows, PRI with 10 data channels provides an improvement of 1.22 times in throughput over random allocation of 10 channels. Using the Adaptive PRI algorithm for the networks of 6 nodes and 20 nodes, the maximum possible throughput (6 Mbps and 20 Mbps, respectively) can be achieved by utilizing 3 and 10 channels respectively, which will allocate a different channel to each link. However, for the network of 50 nodes saturation and high drop rate are inevitable, although the throughput is improved significantly by increasing the number of channels. As the number of nodes in the network increase, the number of contending nodes during the time slot, T_s , and mini slot, T_{mini} , increases. This can result in a case that some nodes do not get any chance to transmit during T_{mini} . Hence with a performance much smaller than the desired performance (i.e., unsatisfactory environment response), due to the “reward” characteristic of the learning algorithm, probabilities of channel selection would not be updated for them.

Table I also presents the drop rate and energy consumption in the network using the different methods of channel allocations, and different number of channels. The results show that for the networks of 3 and 10 flows, the drop rate is significantly reduced by utilizing the Adaptive PRI learning scheme and more number of channels. The drop rate for the network of 25 flows is also reduced, but not as much as it was for the networks with smaller densities. This is due to the fact that the network is so dense and the number of contending nodes is so high that the saturation is inevitable. It can be noticed by using the Adaptive PRI channel allocation and 10 data channels, in the worst case scenario (greatest number of flows), the drop rate is reduced by 78.38% compared to when using a single-channel 802.11. For the same case of 25 flows, PRI with 10 data channels provides a 44.78% reduction on drop rate over random allocation of 10 channels.

The results also show that using the PRI learning scheme and increasing the number of data channels significantly improves the energy consumption per packet. It can be noticed that by using PRI channel allocation and 10 data channels, in the worst case scenario (greatest number of flows), the energy consumption is reduced by 90.25% compared to when using a single-channel 802.11. Also using PRI with data channels reduces the energy consumption by 12.33%. For the same case of 25 flows, PRI with 10 data channels provides a 12.33% reduction in energy consumption per packet over random allocation of 10 channels.

Another performance metric that was used for evaluating the channel allocation schemes was fairness index [11]. Table I also presents the fairness index provided by using the different methods of channel allocations, and different number of

channels. The results show that using the Adaptive PRI learning scheme and increasing the number of data channels improves the fairness index – especially when there are greater number of flows. It can be noticed that by using the Adaptive PRI channel allocation and 10 data channels, in the worst case scenario (greatest number of flows), the fairness index is

increased by 3.7 times compared to when using a single-channel 802.11. Also using the Adaptive PRI with 10 data channels increases the fairness index by 1.28%. For the same case of 25 flows, the Adaptive PRI with 10 data channels provides a 1.28% improvement in fairness over random allocation of 10 channels.

TABLE I. PERFORMANCE OF CHANNEL ALLOCATION SCHEMES.

	Throughput (Mbps)			Drop rate(Mbps)			Energy consumption (joules/packet)			Fairness index		
	3 flows	10 flows	25 flows	3 flows	10 flows	25 flows	3 flows	10 flows	25 flows	3 flows	10 flows	25 flows
802.11 – single data channel	4.20	3.89	3.00	0.77	15.98	47.00	0.00215	0.00807	0.01969	0.8028	0.4443	0.2157
PRI – 3 data channels	6.12	12.44	12.19	0	5.82	38.80	0.00125	0.00235	0.00521	0.9716	0.8337	0.5129
PRI – 10 data channels	6.15	20.57	39.58	0	0	10.16	0.00109	0.00130	0.00192	0.9824	0.9531	0.8022
802.11 – 10 data channels, random channel allocation	6.20	18.80	32.53	0	0.65	18.40	0.00105	0.00142	0.00219	0.9811	0.9475	0.7921

C. Mobile Scenario

TABLE II. PERFORMANCE OF PRI WITH NODE MOBILITY

	PRI, 10 data channels				
	Static (0 m/s)	5 m/s	10 m/s	15 m/s	20 m/s
Throughput (Mbps)	84.31	83.68	82.96	81.84	79.44
Drop rate (Mbps)	13.35	14.10	14.62	15.71	17.78
Energy consumption (joules/packet)	0.00173	0.00174	0.00174	0.00176	0.00181
Fairness index	0.7066	0.6975	0.6900	0.6868	0.6636

In Section IV.B (static scenario) we mentioned the assumption of a static network topology during T_s . In this section we examine a case that the network topology undergoes changes during the T_s period. We consider a larger network (1000mx1000m) and greater number of flows (50 flows, i.e. 100 peer-to-peer nodes). Then the behavior of the single-channel 802.11, randomly allocated 10 channels using 802.11, and the Adaptive PRI learning scheme in the case of mobility of the nodes were examined. For four different values of maximum speed (5, 10, 15, and 20 m/s) and also static case (0 m/s), 10 random scenarios were generated and the average of these repeated simulations were used for comparison. Table II presents the results for using the Adaptive PRI and 10 channels. The speed change does not show a significant effect on the performance. However, in general, these larger network scenarios with a higher traffic flow show a lower performance compared to the static case (Section IV.B).

TABLE III. PERFORMANCE OF DIFFERENT SCHEMES WITH NODE MOBILITY

	10 m/s	
	802.11 - single channel	802.11 – 10 data channels, randomly allocated
PRI – 10 data channels		

Throughput (Mbps)	15.51	69.97	83.68
Drop rate (Mbps)	80.43	26.92	14.10
Energy consumption (joules/packet)	0.008398	0.001940	0.001735
Fairness index	0.2169	0.6263	0.6975

By using the Adaptive PRI learning scheme, the throughput, drop rate and energy consumption show a significant improvement compared to the case that 802.11 is used with randomly allocated 10 data channels (Table III). Also compared to the single-channel 802.11, both Adaptive PRI and 802.11 over randomly allocated 10-data channel are performing significantly better.

The throughput is improved by 19.6%, the drop rate is reduced by 47.6%, the energy consumption per packet is reduced by 10.6% and the fairness index is improved by 11.4%. Also compared to the single-channel 802.11, both Adaptive PRI and 802.11 over randomly allocated 10-data channel are performing significantly better.

V. CONCLUSIONS

In this paper we propose a distributed dynamic channel allocation algorithm for wireless networks whose nodes are equipped with single radio interface. The periodic nature of the algorithm makes it dynamic and enables the channel allocation to adapt to the topographic changes, possible loss of some channels, mobility of the nodes, and the traffic flow changes. The Adaptive Pursuit learning algorithm runs periodically on the nodes, and adaptively finds the optimum channel allocation that provides the desired performance while the convergence of the algorithm is guaranteed. The simulation results for static and mobile networks of different densities and data channels demonstrate that a significant improvement is achieved in throughput, drop rate, energy consumption per packet, fairness index when compared to the single-channel. 802.11 and random allocation of the channels.

VI. ACKNOWLEDGMENTS

The authors would like to thank the AFRL Contract and Intelligent Systems Center that supported this research in part.

Appendix A

Proof of Theorem I: From the definition for discrete pursuit reward-inaction, we know that if m_i satisfies

$$m_i = \arg \max_j \hat{\phi}_i^j(k), \text{ and } \hat{\phi}_i^{m_i}(k) = \max_j \hat{\phi}_i^j(k).$$

then $\hat{\phi}_i^{m_i}(k) > \hat{\phi}_i^j(k)$ for all $j \neq m_i$ and all $k \geq k_0$.

Therefore, for all $k > k_0$,

$$p_i^{m_i}(k+1) = \begin{cases} 1 - \sum_{j=1, j \neq m_i}^N (p_i^j(k) - \theta(k)), \\ \quad \text{if } \beta_i^j(k) = 0 \quad (\text{w.p. } \zeta_i^{m_i}(k)) \\ p_i^{m_i}(k) \\ \quad \text{if } \beta_i^j(k) = 1 \quad (\text{w.p. } 1 - \zeta_i^{m_i}(k)) \end{cases}$$

If $p_i^{m_i}(k) = 1$, then the ‘‘pursuit’’ property of the algorithm trivially proves the result.

Assuming that the algorithm has not yet converged to the m_i th channel, there exists at least one nonzero component of $\mathbf{P}_i(k)$, $p_i^q(k)$, with $q \neq m_i$. Therefore we can write $p_i^q(k+1) = p_i^q(k) - \theta(k) < p_i^q(k)$. Since $\mathbf{P}_i(k)$ is a probability vector, $\sum_{j=1}^N p_i^j(k) = 1$, and $p_i^{m_i}(k) = 1 - \sum_{j=1, j \neq m_i}^N p_i^j(k)$.

Therefore, $1 - \sum_{j=1, j \neq m_i}^N (p_i^j(k) - \theta(k)) > p_i^{m_i}(k)$.

As long as there is at least one nonzero component, $p_i^q(k)$ (where $q \neq m_i$), it is clear that we can decrement $p_i^q(k)$ and increment $p_i^{m_i}(k)$ by at least $\theta(k)$.

$$\text{Hence, } p_i^{m_i}(k+1) = p_i^{m_i}(k) + c(k) \cdot \theta(k)$$

where $c(k) \cdot \theta(k)$ is an integral multiple of $\theta(k)$, and $0 < c(k) < N$, and

$$\theta(k) = \begin{cases} \gamma \cdot \frac{|\Delta(k)|}{\phi^*}, & \text{if } -\delta < \frac{\Delta(k)}{\phi^*} \\ \lambda \cdot \frac{|\Delta(k)|}{\phi^*}, & \text{otherwise} \end{cases}$$

Therefore we can express the expected value of $p_i^{m_i}(k+1)$ conditioned on the current state of the channel, $\mathbf{Q}(k)$, ($\mathbf{Q}(k) = \{\mathbf{P}_i(k), \phi_i(k)\}$) as follows

$$\begin{aligned} E[p_i^{m_i}(k+1) | \mathbf{Q}(k), p_i^{m_i}(k) \neq 1] &= \\ \zeta_i^{m_i}(k) \cdot [p_i^{m_i}(k) + c(k) \cdot \theta(k)] &+ (1 - \zeta_i^{m_i}(k)) \cdot p_i^{m_i}(k) \\ &= p_i^{m_i}(k) + \zeta_i^{m_i}(k) \cdot c(k) \cdot \theta(k) \end{aligned}$$

Since all the previous terms have an upper bound of unity, $E[p_i^{m_i}(k+1) | \mathbf{Q}(k), p_i^{m_i}(k) \neq 1]$ is also bounded,

$$\sup_{k \geq 0} E[p_i^{m_i}(k+1) | \mathbf{Q}(k), p_i^{m_i}(k) \neq 1] < \infty.$$

Thus we can write

$$\begin{aligned} E[p_i^{m_i}(k+1) - p_i^{m_i}(k) | \mathbf{Q}(k)] &= \\ \zeta_i^{m_i}(k) \cdot c(k) \cdot \theta(k) &\geq 0, \quad \text{for all } k \geq k_0 \end{aligned}$$

implying that $p_i^{m_i}(k)$ is submartingale. By submartingale convergence theorem, the sequence $\{p_i^{m_i}(k)\}_{k \geq k_0}$ converges.

Hence, $E[p_i^{m_i}(k+1) - p_i^{m_i}(k) | \mathbf{Q}(k)] \rightarrow 0$ w.p.1, as $k \rightarrow \infty$.

This implies that $\zeta_i^{m_i}(k) \cdot c(k) \cdot \theta(k) \rightarrow 0$ w.p.1. This in turn implies that $c(k) \rightarrow 0$ w.p.1 ($\theta(k) \rightarrow 0$ w.p.1), which means there is no nonzero element in $\mathbf{P}_i(k)$ except for $p_i^{m_i}(k)$ (or $\Delta(k) \rightarrow 0$).

Consequently, $\sum_{j=1, j \neq m_i}^N p_i^j(k) \rightarrow 0$ w.p.1 and

$$p_i^{m_i} = 1 - \sum_{j=1, j \neq m_i}^N p_i^j(k) \rightarrow 1 \quad \text{w.p.1} \quad \blacksquare$$

Proof of Theorem II: Omitted. \blacksquare

REFERENCES

- [1] Joseph Mitola III. *Cognitive Radio An Integrated Agent Architecture for Software Defined Radio*. PhD thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 2000.
- [2] Bahl, P., Chandra, R., and Dunagan, J., ‘‘SSCH: slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks,’’ *Proc. of the 10th Annual international Conference on Mobile Computing and Networking* (Philadelphia, PA, USA, September 26 - MobiCom '04. ACM, New York, NY, pp. 216-230, Oct 2004.
- [3] Alicherry, M., Bhatia, R., and Li, L. 2005. Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks. In *Proceedings of the 11th Annual international Conference on Mobile Computing and Networking* (Cologne, Germany, August 28 - September 02, 2005). MobiCom '05. ACM, New York, NY, pp.58-72.
- [4] Mishra, A., Banerjee, S., and Arbaugh, W., ‘‘Weighted coloring based channel assignment for WLANs,’’ *SIGMOBILE Mob. Comput. Commun. Rev.* 9, 3 pp.19-31, July 2005.
- [5] Nie, N.; and Comaniciu, C., ‘‘Adaptive channel allocation spectrum etiquette for cognitive radio networks,’’ *Proc. of the First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, vol., no., pp.269-278, 8-11 Nov. 2005.
- [6] Felegyhazi, M.; Cagalj, M.; Bidokhti, S. S.; and Hubaux, J.-P., ‘‘Non-Cooperative Multi-Radio Channel Allocation in Wireless Networks,’’ *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, vol., no., pp.1442-1450, May 2007.
- [7] Kyasanur, P.; and Vaidya, N.H., ‘‘Routing and interface assignment in multi-channel multi-interface wireless networks,’’ *Proc. of Wireless Com. and Networking Conf.*, vol.4, pp. 2051-2056, March 2005.
- [8] Jungmin So and Nitin Vaidya, ‘‘Routing and channel assignment in multi-channel multi-hop wireless networks with single-NIC devices,’’ *Technical Report, University of Illinois-Urbana Champaign*, Dec. 2004.
- [9] Oommen, B.J.; and Lanctot, J.K., ‘‘Discretized pursuit learning automata,’’ *IEEE Transactions on Systems, Man and Cybernetics*, vol.20, no.4, pp.931-938, Jul/Aug 1990.
- [10] Haleem, M.A.; and Chandramouli, R., ‘‘Adaptive downlink scheduling and rate selection: a cross-layer design,’’ *IEEE Journal on Selected Areas in Comm.*, vol.23, no.6, pp. 1287-1297, June 2005.
- [11] Jain, R., D. Chiu, and Hawe, W., ‘‘A Quantitative Measure Of Fairness And Discrimination For Resource Allocation In Shared Computer Systems’’, DEC Research Report TR-301, September 1984.
- [12] Clausen, T., and Jacquet, P., ‘‘Optimized Link State protocol,’’ 2003.
- [13] Regatte, N., and S. Jagannathan ‘‘Optimized Energy-Delay Routing in Ad Hoc Wireless Networks,’’ *Proc. of the World Wireless Congress*, 2005.