An Online Learning Method to Maximize Energy Efficiency of Cognitive Sensor Networks

Ali Valehi, and Abolfazl Razi, Senior Member, IEEE

***Abstract*—In this paper, an online reinforcement learning method based on *particle swarm optimization* (PSO) is proposed to maximize transmission energy efficiency of *cognitive radio sensor networks* (CRSN) by regularizing packet lengths. The idea is to simulate a set of artificial channels to find a hypothetical channel that behaves almost equivalently to the actual channel by minimizing a properly developed loss function. This method eliminates the need for a separate offline channel estimation. The results show an improvement of respectively 40% and 20% in the energy efficiency of the proposed method compared to the constant packet length and MLE based offline channel estimation. Also, the PSO based optimization method outperforms similar evolutionary methods. This framework is general and can be used to optimize a desired parameter in cognitive radio networks.**

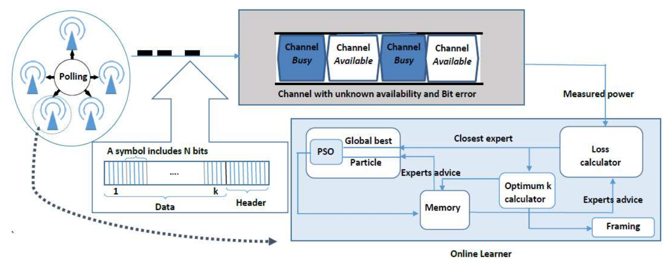

Fig. 1: System model: A cluster of cognitive sensors combine measurement samples into *data packets* and send them in vacancies of a shared channel. Using an online learning method, packet lengths are regularized to maximize energy efficiency. A polling method is used to coordinate sensors' access to the channel to avoid inter-sensor interference.

I. Introduction

Recently, the idea of using *cognitive radio sensor networks* (CRSN) by leveraging cognizant spectrum access to implement low-cost sensor networks has witnessed an unprecedented attention [1]. An important challenge of CRSN is optimizing transmission energy efficiency (EE) to prolong lifetime of tiny sensors with limited power sources. Improving EE also reduces the environmental impacts of communication systems, as a key goal of *green communications* [2].

In this paper, we study the problem of maximizing EE for a network of sensors as unlicensed secondary users (SU) which transmit their data packets in vacancies of a shared channel following an *interweave-based cognition* [3]. In [4], the authors proposed to optimize EE of CRSN through regularizing packet lengths based on the current channel conditions. However, this approach assumes a prior knowledge of the channel statistics for SUs, which is not realistic in practical applications. On the other hand, using channel estimation methods requires long training times and hence imposes additional transmission delays. Channel prediction methods are used recently to accelerate this process by using the history of channel state information (CSI) to predict the properties of next few time slots using two exploitation and exploration phases [5], but is not suitable for extremely dynamic situations. To eliminate the need for additional exploitation phase, reinforcement learning can be used to learn channel conditions over time [6]. However, the learning outcomes typically exhibit considerable lag from the actual channel variations and hence fail in timely reaction to abrupt channel variations.

To address the above mentioned issues, here we propose a novel online learning method that predicts a channel's EE behavior, instead of directly learning channel CSI. The idea is to simulate a set of artificial channels and utilize a *particle swarm optimization* (PSO)-based method that converges rapidly to a hypothetical channel which behaves almost equivalently to the actual channel in terms of EE. Packet length regularization is incorporated into the optimization stage and hence the system adaptively adjusts packet lengths to retain maximal EE. This PSO-based learning method can be used to optimize a desired performance metric of secondary networks such as power consumption, delay, and data throughput without a prior knowledge about the primary network. It eliminates the need for constant channel estimation [7]. Further, this software-based method provides a quick adaption to the channel variations at relatively low computational cost without imposing additional delay and energy consumption to the system.

II. System Model

The system model as depicted in Fig. 1 comprises a cluster of cognitive sensors as SUs that collect measurement information and send it to a designated destination through a shared channel owned by a network of licensed primary users (PU). Here, we use interweaved cognition, where SUs utilize the channel only if it is not in use by PUs. Also, SUs abandon their current transmission sessions as soon as the common channel seized by PUs to make the secondary network totally absent from the primary network's perspective [8]. Following the commonly adopted assumption of independent interval among consecutive samples, here we assume that the N-bit measurement samples are generated according to a Poisson process of rate λ. A sequence of k consecutive samples are combined to form a *data packet* of length $l(k) = kN + H$ bits, where H is the packet header. We utilize coordinated medium access control (e.g. polling-based channel access) among the SUs to avoid inter-sensor interference. This assumption provides the convenience of modeling a system of M users with input traffic λ' as a single-user with input traffic $\lambda = M\lambda'$. We assume zero error tolerance with selective *automatic repeat request* (ARQ) retransmission mechanism to ensure that an error-free copy of each packet is delivered to the destination.

A SU's packet is discarded due to two main reasons including channel errors and loss of channel when a PU seizes the channel. The channel utilization process is modeled as a sequence of intervals with alternating *busy* and *available* states, where the intervals follow the commonly used exponential distribution [9] with means u and v, respectively (i.e. $T_{2i} \sim \exp(u)$, $T_{2i+1} \sim \exp(v)$ for $i = 0, 1, 2, \dots$).

This work concerns the fundamental question of what is the optimal number of samples in each data packet (denoted by k), such that the EE is maximized noting the contradictory impacts of k on EE. Longer packets increase the packet error rate, $\beta_p(k) = 1-(1-\beta)^{kN+H}$, where β and β_p are the bit and packet error probabilities, respectively. Also, longer packets increase the probability of channel loss during a transmission session. Therefore, a large k declines EE through increasing the retransmission rate. On the other hand, a larger k improves EE by reducing the packet overhead ratio $H/(kN + H)$. In particular, we are interested in developing an efficient and near-optimal method that regularizes k_t for iterations $t = 1, 2, 3 \dots, n_T$ based on the current system conditions such that it converges to the maximizer of EE(k), where no prior knowledge about the channel conditions (u, v, β) is available to the secondary system.

$$k_t \to k* = \underset{k}{\operatorname{argmax}}\{EE(k)\} \quad \text{as } t \to \infty. \tag{1}$$

III. Energy Efficiency

In this work, we follow the popular definition of EE [10] as the average number of successfully transmitted bits per unit energy consumption. It is common to split consumed energy into two parts including P_p (the power used per packet for packet formation, queuing, channel selection, etc [11]) and P_b (power used per bit for the actual transmission power). Consequently, we have:

$$EE(k, ch) = \frac{kN}{P_b(\mathbb{E}[R_d l_d(k)] + l(k))\mathbb{E}[R_e] + P_p}, \tag{2}$$

where, R_d is the number of retransmissions caused by the channel loss until sending a copy of the packet, $l_d(k)$ is the average number of bits sent in an unsuccessful transmission attempt before the channel loss, R_e is the number of retransmissions due to channel errors and $l(k)$ is the number of bits in a complete packet. If the channel parameters are known, we can find an explicit expression for $EE(k, ch)$ by characterizing R_d, and R_e in terms of channel statistics (u, v, β) and k. Before detailing the proposed method, we simplify (2) to

$$EE(k, ch) = \frac{kN}{P_b(\mathbb{E}[R_d]\mathbb{E}[l_d(k)] + l(k))\mathbb{E}[R_e] + P_p}, \tag{3}$$

noting that R_d and $l_d(k)$ are conditionally independent for a given k. Exponential distribution of channel intervals implies a uniform distribution for the intersection of a packet and channel transition epoch, which in turn yields $\mathbb{E}[l_d(k)] = l(k)/2 = (KN + H)/2$. Also, R_e follows a geometric distribution with success probability $\alpha_p(k) = (1 - \beta_b)^{kN+H}$ and hence we have $\mathbb{E}[R_e] = 1/\alpha_p(k)$. However, finding $\mathbb{E}[R_d]$ is more involving and we have to consider different scenarios for a

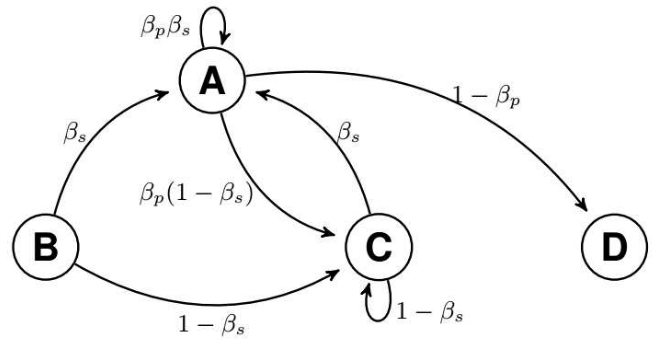

Fig. 2: The system dynamics (in terms of the channel status when the SU attempts to transmit a packet) is modeled as a Markov chain.

packet transmission abortion. We note that in a queued system, the departure epoch of a packet depends on the previous packet departure. We model the system dynamics with a Markov chain as depicted in Fig. 2, where the transitions show the state of the next transmission attempt with respect to the current attempt. We define the following four states:

- A: the channel is *available* and the remaining portion of the current interval is sufficient to send a packet.
- B: the channel is *busy*.
- C: the channel is *available* but the remaining of the current interval is not sufficient to send a packet.
- D: Absorbing state, an error-free copy of the packet is delivered to the destination.

We develop the following state transition matrix T based on a set of probabilities including the packet error probability (β_p) and the probability of the channel being sufficient for a packet transmission ($\beta_s = \Pr(T > s_b) = e^{\frac{-s_b}{v}}$), where $s_b = \frac{kN+H}{R_{ch}}$ is the time required to send a packet.

$$\mathrm{T} = \begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{cccc} A & B & C & D \end{array} \\ \begin{pmatrix} \beta_p\beta_s & 0 & \beta_p(1-\beta_s) & (1-\beta_p) \\ \beta_s & 0 & (1-\beta_s) & 0 \\ \beta_s & 0 & (1-\beta_s) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{array} \tag{4}$$

Likewise, the initial probability for the first packet and the packets which arrive after the absorbing state, denoted by Π_0, can be expressed as follows:

$$\begin{aligned} \Pi_0^T &= \begin{bmatrix} p_{A_0} & p_{B_0} & p_{C_0} & p_{D_0} \end{bmatrix} \\ &= \begin{bmatrix} \frac{v}{u+v}\beta_s & \frac{u}{u+v} & \frac{v}{u+v}(1-\beta_s) & 0 \end{bmatrix} \end{aligned} \tag{5}$$

The expected number of passing through each state before ending up with the absorbing state can be found using a submatrix of transition matrix (after excluding the row and column of the absorbing state) denoted by Z [12]. We first obtain the fundamental matrix of (4) denoted by Q from Z, as follows:

$$Q = (I - Z)^{-1} = \begin{bmatrix} \frac{-1}{\beta_p - 1} & 0 & \frac{-\beta_p(\beta_s - 1)}{\beta_s - \beta_p\beta_s} \\ \frac{-1}{\beta_p - 1} & 1 & \frac{-(\beta_s - 1)}{\beta_s - \beta_p\beta_s} \\ \frac{-1}{\beta_p - 1} & 0 & \frac{-(\beta_p\beta_s - 1)}{\beta_s - \beta_p\beta_s} \end{bmatrix} \tag{6}$$

Noting the fact that passing through each step except the absorption state corresponds to dropping a packet, we can find the expected number of discarded packets as:

$$\begin{aligned}
\mathbb{E}[R_d] &= \sum \text{Initial probability of each state} \\
&\times \text{expected number of passing through each state} \\
&= \Pi(1)Q(1,3) + \Pi(2)Q(2,3) + \Pi(3)Q(3,3) = \\
&= \frac{v}{u+v}\beta_s\Big(\frac{-\beta_p(\beta_s-1)}{\beta_s-\beta_p\beta_s}\Big) + \frac{u}{u+v}\Big(\frac{-(\beta_s-1)}{\beta_s-\beta_p\beta_s}\Big) \\
&\quad + \frac{v}{u+v}(1-\beta_s)\Big(\frac{-(\beta_p\beta_s-1)}{\beta_s-\beta_p\beta_s}\Big).
\end{aligned} \tag{7}$$

IV. Online Learning Method

Here, we propose an online learning method with expert advice that converges to the maximizer of (3), for unknown (u, v, β). The idea is using PSO, to maximize a function with multiple local optimums by starting from a large number of initialization points and moving towards directions where a desired cost function for each particle and for the whole system is minimized.

A. Initialization: Firstly, a predefined number of particles, $o_i = (u_i, v_i, \beta_i), i = 1, 2, \dots n$, are randomly generated. Here, u_i, v_i, β_i are hypothetical expected values of (u, v, β) form o_i's perspective. Also, an initial value for k is chosen (k_1).

B. Try: In each segment s, the system collects $m.k_s$ samples and transmits m packets of length $l(k_s)$, then the empirical EE is calculated based on the most recent L packets as $EE(k_s, ch) = kLN/\Big(P_b\big(\sum_{j=1}^{L}[R_d(j)l_d(k_s, j)] + l(k_s)\sum_{j=1}^{L}[R_e(j)]\big) + LP_p\Big)$.

C. Optimization: Then, a PSO-based optimization algorithm is executed. Each particle o_i, calculates the hypothetically expected energy efficiency $EE(k_s, o_i^t)$ based on the previously chosen k_s and its own parameter set $(u_i^t, v_i^t, \beta_i^t)$, where postscript t denotes the iteration number. Subsequently, a loss function $d^t(i) = |EE(k_{s i}, o_i^t) - EE(k_s, ch)|$ is calculated for each particle. Then, an iterative PSO-based algorithm is executed based on two input sets including i) experts' advice as particles and ii) the loss function based on empirical energy efficiency. At each iteration t, each particle i moves towards a direction which is the linear combination of i) its previous motion at iteration $t-1$, ii) its *best local position*, denoted by $o_b^t(i)$ with a locally minimal loss function $b(i)$, and iii) the *global best* particle, denoted by o_g. More formally, we have:

$$\begin{aligned}
&o_i^{t+1} = o_i^t + V_i^t dt, \\
&V_i^t dt = \delta(o_b(i) - o_i^t) + \beta(o_g - o_i^t) + \gamma(o_i^t - o_i^{t-1}),
\end{aligned} \tag{8}$$

where δ, β, γ are tuning parameters. The motions continue until the algorithm converges to a *global best* or for a predefined number of iterations. Finally, $k_{s+1} \leftarrow \arg\min_k EE(k, o_g)$ is chosen as the best k for the next segment. The pseudo code of the proposed algorithm is presented next. The complexity of this algorithm is $O(n_T n)$, which is linear in the number of iteration n_T and particles n, therefore computationally feasible in practical systems. The convergence to a global optimum is ensured by choosing sufficiently large n and distant particles.

Algorithm 1: Optimum packet length estimation algorithm

Input: ,H,N, $EE(k, ch)$,n_T,m,L,n
Output: k_s, for $s = 2, 3, 4, \dots,$
begin
  Init: initialize k_1, initialize o_i with random (u_i,v_i,β_i)
  for *segment:s* = *1* **to** ∞ **do**
    Send m packets of length k_s
    $EE(k_s, ch) \leftarrow$ Measure empirical EE
    Init: $b(i) = \infty$ for $i = 1, ..., n$
    for $t = 1$ **to** n_T **do**
      for $i = 1$ **to** n **do**
        $d^t(i) = |EE(k_s, o_i^t) - EE(k_s, ch)|$
        if $d^t(i) < b(i)$ **then**
          $b(i) \leftarrow d^t(i)$ (update local best)
          $o_b(i) \leftarrow o_i^t$
      $g \leftarrow \arg\min_i b(i)$ $i = 1, .., n$ (global best)
      for $i = 1$ **to** n **do**
        $o_i^{t+1} \leftarrow$
        $o_i^t + \delta(o_b(i) - o_i^t) + \beta(o_g - o_i^t) + \gamma(o_i^t - o_i^{t-1})$
    $k_{s+1} \leftarrow \arg\min_k EE(k, o_g)$

V. Numerical results

In this section, we evaluate the performance of the proposed algorithm based on the system model provided in section II. System parameters are set to $\lambda = 1$, $N = 16$, $H = 80$, $P_b = 0.01$, $P_P = 0.02$, $R_{ch} = 100$ bps, $L = 300$, and $m = 30$ unless otherwise specified. We also assume that the channel parameters are subject to abrupt changes to simulate the worst-case scenario. The system starts with a random k_1. At each segment s, m packets of length $l(k_s)$ is sent. The proposed online learning method is executed based on the resulting EE for the last $L = 300$ transmitted packets to identify the optimal k_{s+1} for the subsequent segment and this cycle continues. The training and a transmission segment overlap in $min(m, L) = 30$ packets. The ratio of m/L makes a balance between the accuracy of empirical EE and the agility of the algorithm in responding to channel variations. We use randomly initialized seed particles for each segment in order to provide flexibility in accommodating channel variations from one segment to another. We also include the best particle of the previous segment as one of the seed particles for a faster convergence in the absence of channel variation.

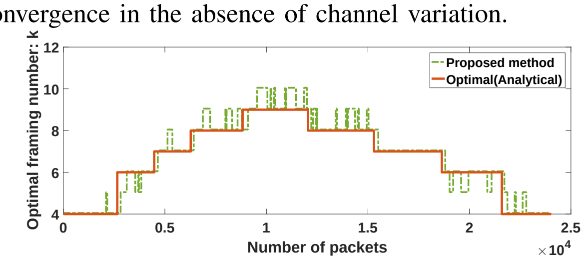

Fig. 3: Optimum framing parameter (k) found by the proposed method for the SU for time-varying channel availability rate ($v/u \in [1 : 0.1 : 3]$, $\beta = 10^{-4}$).

TABLE I: Evolution of the best particle for channel parameters ($\beta = 10^{-4}$,$v/u = 2$)

Iteration	Channel v/u	β	k_{opt}	EE
2	1.18	1.6e-4	9	0.281
5	1.1231	1.7264e-4	9	0.320
10	1.1193	1.7267e-4	8	0.323
100	1.1193	1.7268e-4	8	0.323
True Channel	2	1e-4	8	0.325

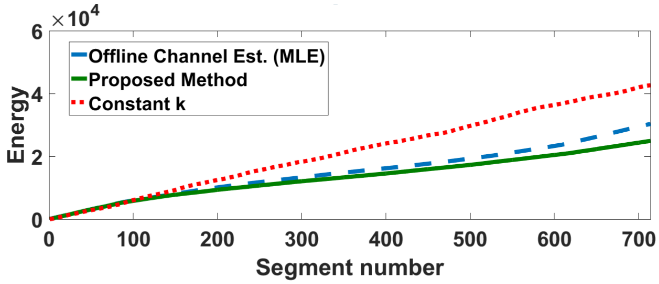

Fig. 4: Accumulated energy consumption for the system with constant k, MLE-based offline learning and the proposed method.

The basic operation and convergence of the algorithm is presented in Fig. 3. The particles start from a random set of hypothetical channel parameters (β, u, v), but quickly converge to an artificial channel which yields an EE close to the empirical EE, hence it can be used to determine the optimal k value. The green curve shows the suggestive packet length k by the best particle at the end of each round of algorithm, and the red curve is the optimal value of k based on the actual channel parameters. The convergence is regardless of the initialization points, hence the algorithm accommodates abrupt channel condition changes. The most deviation from the optimal k we noticed is 1, which is due to the discretizing error, when solving the equation (1) for the best particle.

It is interesting that the parameters of the best particle (β, u, v) may or may not match the actual channel, since infinitely many combinations of (β, u, v) in (3) yield the same EE. This concept is shown in Table I, where the best particle converges to $k_{opt} = 8$ in 10 iterations, but represents channel parameters ($\beta = 1.727 \times 10^{-4}$,$v/u = 1.119$) which are totally different than the true channel values ($\beta = 10^{-4}$,$v/u = 2$). The resulting EE is within 1% from the actual one in terms of MSE. This is the main contrast of the proposed method with directly estimating the channel parameters. Fig. 4 compares the accumulated energy consumption of the system under the same conditions for three methods including i) constant k customized to the initial channel conditions, ii) optimal k based on the offline maximum likelihood estimation (MLE) of the channel parameters, and iii) the proposed method. It is shown that the proposed method outperforms both methods (20% over the MLE method and 40% over the constant k). The superiority of the proposed method which adapts to the dynamic channel conditions over the constant k is trivial. The result of the proposed method overcomes the MLE method, since a fairly large amount of packets are required for the accurate estimation of all channel parameters.

Finally, Fig.5 compares the result of the proposed PSO-based method with similar online learning approaches using two different evolutionary methods *differential evolution*(DE) and *simulated annealing* (SA) in terms of statistical parameters: mean (μ) and variance (σ) of the optimal k over 100 different runs. It is seen that the PSO-based optimization engine outperforms other evolutionary methods since multiple particles can cover a wider range of channel conditions and are more successful in finding the global optima.

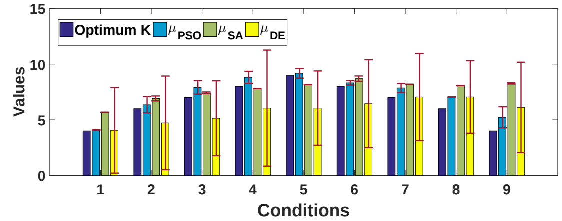

Fig. 5: Performance comparison of online learning methods using PSO, DE, and SA methods. The mean and standard deviation of k is shown for 100 runs.

VI. Conclusions

In this paper, a novel approach is proposed to adjust packets lengths of secondary users based on dynamic channel conditions without directly estimating the shared channel parameters. The idea is to use an online learning method based on PSO to adjust packet lengths such that the deviation of resulting energy efficiency from the empirically obtained value is minimized. This novel method eliminates the need for estimating channel parameters directly and shows 20% less transmission power consumption compared to MLE-based offline channel estimation and 40% compared to constant packet lengths. The near-optimal performance of this method is evident by catching up with the analytically derived parameters after a few iterations. This method is general and can be used for similar optimization problems in cognitive networks.

References

[1] O. B. Akan, O. B. Karli, and O. Ergul, "Cognitive radio sensor networks," *IEEE Network*, vol. 23, no. 4, pp. 34–40, July 2009.

[2] R. Mahapatra, Y. Nijsure, G. Kaddoum, N. U. Hassan, and C. Yuen, "Energy efficiency tradeoff mechanism towards wireless green comm: A survey," *IEEE Comm. Surveys Tut.*, vol. 18, no. 1, pp. 686–705, 2016.

[3] A. Goldsmith, S. A. Jafar, I. Maric, and S. Srinivasa, "Breaking spectrum gridlock with cognitive radios: An information theoretic perspective," *Proceedings of the IEEE*, vol. 97, no. 5, pp. 894–914, May 2009.

[4] A. Valehi and A. Razi, "Maximizing energy efficiency of cognitive wireless sensor networks with constrained age of information," *IEEE Trans. Cognitive Comm. Netw.*, no. 99, pp. 1–1, Sep. 2017.

[5] J. Lunden, V. Koivunen, and H. V. Poor, "Spectrum exploration and exploitation for cognitive radio: Recent advances," *IEEE signal processing magazine*, vol. 32, no. 3, pp. 123–140, 2015.

[6] F. M. Al-Turjman, "Information-centric sensor networks for cognitive iot: an overview," *Annals of Telecom.*, vol. 72, no. 1, pp. 3–18, Feb 2017.

[7] X. Chen, H. Zhang, A. B. MacKenzie, and M. Matinmikko, "Predicting spectrum occupancies using a non-stationary hidden markov model," *IEEE Wireless Comm. Let.*, vol. 3, no. 4, pp. 333–336, Aug 2014.

[8] A. Patel, M. Z. A. Khan, S. N. Merchant, U. B. Desai, and L. Hanzo, "The achievable rate of interweave cognitive radio in the face of sensing errors," *IEEE Access*, vol. 5, pp. 8579–8605, 2017.

[9] M. Wellens, J. Riihijarvi, and P. Mahonen, "Evaluation of adaptive mac-layer sensing in realistic spectrum occupancy scenarios," in *IEEE Symp. New Frontiers in Dynamic Spectrum*. IEEE, 2010, pp. 1–12.

[10] A. Razi, F. Afghah, and A. Abedi, "Channel-adaptive packetization policy for minimal latency and maximal energy efficiency," *IEEE Trans. Wireless Comm.*, vol. 15, no. 3, pp. 2407–2420, March 2016.

[11] M. C. Oto and O. B. Akan, "Energy-efficient packet size optimization for cognitive radio sensor networks."

[12] C. M. Grinstead and J. L. Snell, *Introduction to probability*. American Mathematical Soc., 2012.