HAL
archives-ouvertes.fr

Competition: Channel Exploration/Exploitation Based on a Thompson Sampling Approach in a Radio Cognitive Environment

Arash Maskooki, Viktor Toldov, Laurent Clavier, Valeria Loscrí, Nathalie Mitton

► **To cite this version:**

Arash Maskooki, Viktor Toldov, Laurent Clavier, Valeria Loscrí, Nathalie Mitton. Competition: Channel Exploration/Exploitation Based on a Thompson Sampling Approach in a Radio Cognitive Environment. EWSN - International Conference on Embedded Wireless Systems and Networks (dependability competition), Feb 2016, Graz Austria. <hal-01249135>

HAL Id: hal-01249135

https://hal.inria.fr/hal-01249135

Submitted on 30 Dec 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Competition: Channel Exploration/Exploitation Based on a Thompson Sampling Approach in a Radio Cognitive Environment

Arash Maskooki[1], Viktor Toldov[1,2], Laurent Clavier[2,3], Valeria Loscrí[1], Nathalie Mitton[1]
[1]Inria; [2]Université Lille 1, IEMN UMR CNRS 8520, IRCICA USR CNRS 3380; [3]Institut Mines-Télécom, Télécom Lille
[1]firstname.lastname@inria.fr,[2]firstname.lastname@iemn.univ-lille1.fr, [3]firstname.lastname@telecom-lille.fr

Machine learning approaches have been extensively applied in interference mitigation and cognitive radio devices. In this work, we model the spectrum selection process as a multi-arm bandit problem and apply Thompson sampling, a fast and efficient algorithm, to find the best channel in the shortest time interval. The learning algorithm will work on top of a network layer to efficiently route the event information to the sink.In this work we address the problem on two layers: channel decision layer and network layer. Channel decision layer formulates the channel selection as a multi-arm bandit problem. Thomson sampling [2] is a classical solution in Bandit problems. We adapt Thompson sampling to our context and implement it in hardware in the channel selection layer. The network layer uses the results of this channel selection to route data on the specified channel.

We formulate channel selection and exploration dilemma as a multi-arm bandit problem. Then, we describe an efficient and simple learning algorithm for the channel selection process.

In a multi-arm bandit problem, an agent tries to obtain as much reward as possible by playing the most rewarding arm among N arms. However, each arm rewards randomly upon being played according to an unknown distribution. Hence, the objective is to minimize exploration to find the most rewarding arm. A policy A is an algorithm that defines the actions of the agent usually based on the previous observations. We assume n_j to be the number of times j^{th} arm has been played after n steps and μ_j to be the expected reward of j^{th} arm. In other words, channel j is found available in average $\mu_j n_j$ times in n_j measurements. μ_j is associated with the statistics of other networks.

To keep track of the other networks statistics, an expiration time can be defined to trigger the search for a new channel. Another criterion to trigger the search for a new channel is to define a threshold for the number of consecutive unsuccessful channels access after which the user will search for a new channel.

Thompson sampling [2] is best understood in Bayasian context. Assume we observed S_j, the observation vector, after accessing channel j, n_j times. Assuming Bernoulli distribution for each access trial with parameter μ_j, the parametric likelihood function for observation vector S_j is as follows,

$$p_j(S_j|\mu_j) = {\mu_j}^{t_j}(1-\mu_j)^{n_j-t_j}, \tag{1}$$

where t_j is the number of successful transmissions on j^{th} channel in n trials. Without loss of generality, we use Beta distribution as the prior for the distribution of parameter μ_j. This is because Beta distribution is conjugate prior for the likelihood function in Equation (1) which simplifies the derivations [3]. Using Bayes rule we can write,

$$p_j(\mu_j|S_j) = \frac{p_j(S_j|\mu_j)\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}{\mu_j}^{\alpha-1}(1-\mu_j)^{\beta-1}}{p_j(S_j)}, \tag{2}$$

where,

$$\Gamma(\alpha) = \int_0^\infty x^{\alpha-1}e^{-x}dx \tag{3}$$

and α and β are the shape parameters of the Beta distribution; as we assume no prior information on μ_j we initialize $\alpha = \beta = 1$ which yields uniform distribution. Substituting (1) in (2) yields,

$$p_j(\mu_j|S_j) = \frac{\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}}{p_j(S_j)}{\mu_j}^{t_j+\alpha-1}(1-\mu_j)^{n_j-t_j+\beta-1}. \tag{4}$$

$\alpha' = t_j + \alpha$ and $\beta' = n_j - t_j + \beta$ can re-write (4) as:

$$p_j(\mu_j|S_j) = C{\mu_j}^{\alpha'-1}(1-\mu_j)^{\beta'-1} \tag{5}$$

Substituting the normalizing factor C we obtain,

$$p_j(\mu_j|S_j) = \frac{\Gamma(\alpha'+\beta')}{\Gamma(\alpha')\Gamma(\beta')}{\mu_j}^{\alpha'-1}(1-\mu_j)^{\beta'-1}, \tag{6}$$

which is the beta distribution with parameters α' and β',

$$p_j(\mu_j|S_j) = beta(\alpha', \beta'). \tag{7}$$

Thompson sampling Channel selection algorithm is described in Algorithm 1.

Algorithm 1 Thompson Sampling

Parameters: K: total number of accessible channels
j: channel index
n: total number of channel access
s_j: current state of the channel j
t_j: number of successful transmissions so far
$\bar{x}_j$: empirical mean of the channel j states,
α: *a priori* (*beta* distribution) model parameter
β: *a priori* (*beta* distribution) model parameter
α': *a posteriori* (*beta* distribution) model parameter

$$\alpha'_j = t_j + \alpha_j \tag{8}$$

β': *a posteriori* (*beta* distribution) model parameter

$$\beta'_j = n_j - t_j + \beta_j \tag{9}$$

TRANSMIT(): Packet transmission function
Initialization:
for all j **do**
    if channel j is busy **then**
        $s_j = 0$
    else
        $s_j = 1$
    end if
    update t_j, n_j, α'_j and β'_j
end for

while True **do**
    for all j **do**
        sample $r_j \sim beta(\alpha'_j, \beta'_j)$
    end for
    $m = \arg\max\{\bar{r}_j\}$
    if channel m is busy **then**
        $s_m = 0$
    else
        $s_m = 1$
        TRANSMIT()
    end if
    update t_j, n_j, α'_j and β'_j
end while

As there is no prior information about the channels, we set $\alpha = 1$ and $\beta = 1$ which yields uniform distribution in $[0, 1]$.

we implemented the in TelosB nodes to measure its performance in real-time. In our setup, we used 3 pairs of laptops occupying 3 orthogonal Wi-Fi channels, 1, 6 and 11 overlapping with standard 802.15.4 channels, 12, 17 and 22 respectively. The traffic was generated using "Distributed Internet Traffic Generator" [1] in single flow mode with packet size 500 bytes which is the average packet size on the Internet [5]. Two TelosB nodes were programmed in Contiki operating system one with a learning algorithm and one as oracle fixed on the best channel. To generate samples of Beta distribution used in Thompson sampling algorithm we used "GEN_SEQUENCE" open-source library [4].

To monitor the availability rate of the channel we programmed one TelosB node as monitor which just sampled the channel. The availability rate obtained as the average number of samples the channel is detected available. The RSSI sensitivity of TelosB node was set to $-40dbm$. This relatively high threshold was set to suppress the RSSI received from other networks present in the building. With this sensitivity, the monitor node registers approximately 90% availability rate for the channels. The availability rate of channel 6 drops to approximately 40% when the traffic generator is activated at $2000 pkt/sec$ and packet size of 500 bytes. The availability rate of the channel 1 drops to approx. 60% when the traffic generator occupies the channel with $500 pkt/sec$. Channel 11 is left without traffic although the server and client were connected. The monitor shows approx. 90% availability on the channel. Note that the channel occupancy rate is affected by our traffic, other networks traffic and noise. However, it was roughly constant during the experiment at the given rates.

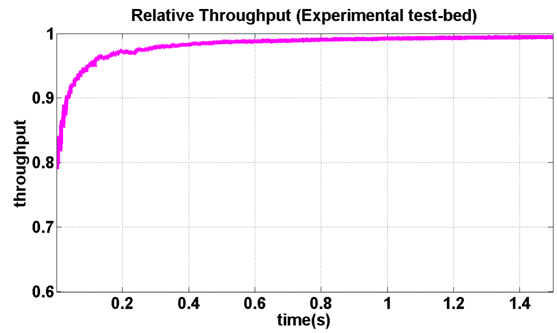

Figure 1: Relative throughput on three accessible channels using Empirical real-time test-bed.

We programmed two Telosb nodes; one as an oracle which always operated on the channel with the best availability rate. The other node was programmed with the implementation of a learning algorithm to find and use the best channel. In our results, we considered an available channel as a successful transmission. In reality, the packet transmission can be disrupted in the middle of the transmission and cause the transmission to fail. However, the collision will affect the throughput results of all algorithms including the oracle the same way. Hence the comparison results would not be affected.

In each set of experiments, we performed 3 experiments where occupancy rate of the channels were permutated. We repeated each experiment 3 times. The relative throughput of each algorithm in each experiment is divided by the oracle performance of the best channel and then averaged over all the experiments for each algorithm. Figure 1 shows the performance of the algorithm. As seen in the figure, the learning algorithm finds the best channel and reaches about 99% of the throughput of the oracle in 0.8 seconds.

Results show that Thompson sampling formulation achieves high average throughput as it spends less time on exploring the channels and converges to the best channel fast.

1 References

[1] A. Botta, A. Dainotti, and A. Pescapè. A tool for the generation of realistic network workload for emerging networking scenarios. *Computer Networks*, 56(15):3531–3547, 2012.

[2] O. Chapelle and L. Li. An empirical evaluation of thompson sampling. In *Advances in neural information processing systems*, pages 2249–2257, 2011.

[3] P. Diaconis, D. Ylvisaker, et al. Conjugate priors for exponential families. *The Annals of statistics*, 7(2):269–281, 1979.

[4] K. Karplus. Online library.

[5] K. Thompson, G. J. Miller, and R. Wilder. Wide-area internet traffic patterns and characteristics. *Network, iEEE*, 11(6):10–23, 1997.