EVALUATION OPTIMIZATION TECHNIQUES FOR SOFTWARE DEFINED RADIO – COGNITIVE RADIO SYSTEM PERFORMANCE

Nikhil Challa , nikhil87@vt.edu; Jeong-O Jeong, jeongo9@vt.edu; Carl Dietrich, cdietric@vt.edu; Tim Newman , trnewman@vt.edu; Jeffrey Reed, reedjh@vt.edu;
Bradley Department of Electrical and Computer Engineering,
Virginia Polytechnic and State University,
Blacksburg, VA 24060

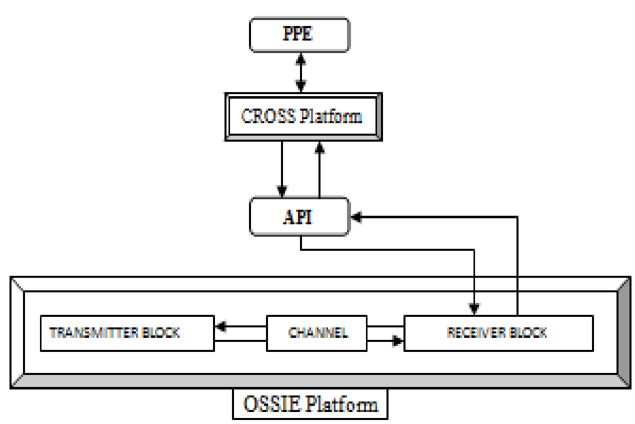
ABSTRACT

This paper documents optimization techniques that were implemented on the Open Source SCA Implementation Embedded (OSSIE) and Cognitive Radio Open Source System (CROSS). The fundamental objective of this development is adapting the radio to variability in channel characteristics without resorting to predefined mathematical expressions. Many of existing radio systems uses complex mathematical expressions or predefined relationships to estimate ideal parameter values for best performance. By reducing this complexity we can also speed up the radio's adaptability. Also due to the flexibility we have incorporated in the system, there are a few assumptions that have to be considered to make the system robust. This approach works independent of the mathematical knowledge of the Parameters and Utilities, but rather works on real time estimates and observations. This paper details the experimental results observed from implementing the platforms with the Universal Software Radio Peripheral (USRP). Further possible research developments are also covered towards the end of the paper.

1. INTRODUCTION

We consider a system in which an OSSIE based Software Defined Radio (SDR) is responsible for signal processing and physical layer modifications, and is interfaced with the CROSS cognitive radio architecture to create a cognitive radio that implements some level of intelligence. By integrating both the platforms, we were able to boost the radio performance. Also by implementing the property of weights, we were able to adjust the system to various requirements that may change from case to case. Parameters of the data blocks at the receiver that have constraints against them are termed as "utilities" and those that do not have any constraints are termed as "observables". We have taken into consideration only one utility constraint for a given system, and also opened up possibilities to involve more constraints. In this paper we discuss possible ways to reduce computation time in the cognitive engine and thereby speed up optimization. A few approaches are discussed in more detail to understand the ways of speeding up computation time. A high level view of the system is shown in Figure 1. Data constantly flow from the transmitter to the receiver and control information is transmitted in the reverse channel. The API is a component, a part of the CROSS platform responsible for extracting observables/utilities from the OSSIE components and also sending back control information from the CROSS platform. Based on the decoded data blocks at the receiver and observing the characteristics of the data packets, the API will pass relevant information to CROSS. Based on the updates and the past performance, CROSS will reply back with the control information which will be sent in the reverse channel. In order to integrate OSSIE and CROSS, it will be important to consider the limitations of both platforms. This paper has equally focused on both platforms by focusing on the signal processing blocks in OSSIE and the embedded intelligence in CROSS. Based on weights for each parameter and utility, the total weight is what determines the best parameter settings. In our application, CROSS uses the Case Based Reasoning (CBR) approach to obtain the optimal parameter values to improve system performance.

Figure 1. Block Diagram



2. PLATFORMS

2.1. OSSIE

OSSIE is an open source SDR [1] development effort based at Wireless@VT. OSSIE provides a freely available SCA-based framework and tools for prototype development that will help demonstrate the performance of this application. OSSIE can be compared to a tunable radio where each block can be separately tuned. A block can be considered as a physical characteristic of a signal like modulation scheme, transmit power, coding scheme, etc. By separating the blocks based on certain properties of the signals, it becomes easier for the cognitive radio to tune the properties.

2.2 CROSS

CROSS is a research project at Wireless@VT that is developing an open source Cognitive Radio architecture [2]. CROSS provides a very stable platform to facilitate communication to any external platforms and also embed any code to perform intelligible processing on the received information from the external source. The CROSS can be considered as the brain of the system and is also responsible for most of computational complexity and delay. Instead of manually tuning the parameter blocks in the SDR, we can leave this function to the CROSS. Also, CROSS has the capability to create APIs to integrate with other platforms.

3. SYSTEM OVERVIEW

3.1. Learning Mechanisms

The advantage of this system over many others is the ability to self-learn the basic relationship between parameters and utilities. It is required that the parameter values are assigned in a known order e.g., increasing transmit power or modulation order. Since we have the expertise/knowledge to relate most of the parameters with utilities, we can use this advantage to reduce computation time. The system over the training period will estimate the approximate relationship between the parameter and the utility and verify if this relationship is valid over its entire domain. The Trend (Tr) represents the direction of improved performance and the exact expression is listed as Equation (1). If the Trend is not consistent then it will register that parameter as a nonrelated parameter, else as either directly proportional or inversely proportional. It is not necessary to know the exact relationship, but the proportionality over its entire range is an important factor. If the system is aware of just this property it will be aware of the direction of increasing performance or efficiency. This can greatly reduce the computational complexity, and we can also neglect the complex mathematical equations to relate the parameters and utilities. The SDR will be tuned based on the past experiences (section 3.3), proportionality estimated and the current state of the system and channel. At a time it is subjected only to one utility constraint. An approach for multi-utility constraint is described later in section 3.2.

$$Tr = abs(M_y) = \left|\frac{\Delta y}{\Delta x}\right| = \left|\frac{y1-y0}{x1-x0}\right| \qquad \text{..... (1)}$$

In the case of nonrelated parameters, it is necessary for the cognitive radio to estimate its performance for all possible combinations. It will be discussed in section 4.

3.2. Optimization

There are multiple possible Parameter Value (P_i) combinations to satisfy a given constraint since the domain of working space is relatively large. In order to reach the most optimal state, it would be necessary to distinguish one parameter from the other by using the concept of weights. In order bring in flexibility to expand the areas of implementation or vary it with user preferences; we define weights (Pw_i, Uw). By using weights, we can distinguish the flexibility of one parameter over the other. Weights can also represent the actual cost for variability. For example, it might be costly to change the modulation scheme as compared to increasing the transmit power, in terms of hardware complexity. Also, increasing the coding gain can adversely affect the data rates, in case data rate is a priority. Another example would be in the case of energy efficient systems, we would give more weight to Transmit power. Also, the utility is given weights to prioritize its deviation from desired value. In many cases, perfect performance is not desired since it would increase resource consumption. An ideal example would be to consider BER to be about 0.2 instead of 0. This has been described with an experimental example in section 4. A Penalty has been included to make sure the system doesn’t go below/above the restricted Utility Target (U_t) based on the requirement. Penalty variable has a value of 1 if Utility is within the domain limits else will have a value much greater than one. This will force the system to meet the specifications regarding the utility range. The parameter values which result in least Total Weight (TW) will be chosen as current values. Equation (2) gives the actual equation used for optimization.

$$TW = P_i \text{ x } Pw_i + |(U - U_t) \text{ x } Uw \text{ x } Penalty| \qquad \text{…. (2)}$$

This approach is applicable to systems that rely on almost unpredictable environment characteristics like environment noise and fading channels. Since this approach relies on learning mechanisms which are not robust for all scenarios, it will be subjected to some limitations which will be addressed later in section 3.6.

There are several factors that contribute to the complexity of an optimization problem like the number of multiple decision variables, uncertainty in channel characteristics and number of simultaneous constraints to be satisfied. We have mitigated most of the complications by not including the exact dependency between utility and parameter and adhering to the concepts of step size and single parameter tuning.

In most cases, the channel undergoes slow variability and hence will not spin the system completely out of the stable domain. The approach used to optimize the system parameters to reduce total weight is based on Equation (3).

If $P_n/P_m > Pw_m/Pw_n$, $P_m = P_m + 1$ else $P_n = P_n + 1$ …. (3)

Considering an example where $Pw_m = 1$, $Pw_n = 2$, with P_m and P_n to be incremented as per weights. The ratio of Pw_n to Pw_m is 2. The incremental order for P_m and P_n based on ratio of Pw_m and Pw_n is shown in Table 1.

Table 1. Incremental order of Parameter Values

P_m	P_n	P_m/P_n
1	1	1
2	1	2
(3) => 2	(1) => 2	1
3	2	1.5
4	2	2
(5) => 4	(2) => 3	1.33
5	3	1.66

In this model we use pair wise optimization, as in we compare just two parameters at a time. It would be obvious to start with the parameters that have the minimum weights, since they can be tuned with least cost or TotalWeight variability. Another factor which determines the order of tuning is the number of valid values. The valid values would depend on min, max and step size, which also depends on SDR capability and user requirements.

3.3. Case Based Reasoning

The Case Based Reasoning Cognitive Engine (CBR-CE) developed by Dr. Tim Newman at Virginia Tech was used as a Cognitive Radio platform for evaluating the system's performance. CBR is an imitation of human learning where experiences can be used for understanding and problem solving process [3,4]. The system relies on past parameter values and its corresponding utility value which is registered by the CBR module and the current state. A table entry is maintained which stores the observed utility values and the receiver corresponding to the tuned parameter values at the transmitter. Since the system has just one table entry to estimate its next optimal parameter values, it greatly reduces the computation time. The other variables that are required are the Trend and PoC (Percentage of Change, section 4). As the channel changes, there is variation in the corresponding utility value and thus the table entries are updated accordingly. This keeps the system updated for further optimization if required. Instead of updating the entire table, we are concentrated on updating only the local entries, since the parameters can only undergo step size variation.

3.4. Error Tolerance

One of the contributing factors for errors is the distribution of the observed value of the utility over the actual value. In order to mitigate the distribution, we average the utility value over 100 or more transmission blocks, or consider each block to consist of large number of samples. This maintains the observed utility value with minimal deviation, thus making the cognitive engine less prone to distribution errors. Also due to the concept of step size, it avoids the system from setting the parameter values beyond domain of stability. The value of the step size is left to the user to improve flexibility. The disadvantage of fixed step size is the slowness of the system to reach an optimal state for cases in which the channel undergoes drastic variations.

3.5. Multi-utility constraints

In order to subject the system to multi-utility constraints, it may be required that CROSS platforms be running in parallel with multiple SDRs with each SDR running on different utility constraint. The parameter properties which include slope, min, max, and current values can then be shared, and by using simple logic, the most optimal parameter settings can be estimated. There will be degradation in system performance in order to fulfill multiple constraints. Consider a case where Data rate is a utility for one system and BER (bit error rate) for the other. The constraint on the data rate will affect the modulation order. Similarly, the transmit power for BER and therefore there will have to be a balance between them. It will be required that both the radio systems maintain the same set of parameters. The various possibilities that may result by sharing the table entries are listed as Equations (1a) to (5a)

$P_1 > a$ & $P_2 < b$ where $a < b$ ----- (1a)
$P_1 < a$ & $P_2 > b$ where $a > b$ ----- (2a)
$P_1 > a$ & $P_2 < b$ where $a > b$ ----- (3a)
$P_1 < a$ & $P_2 > b$ where $a < b$ ----- (4a)
$P_1 < a$ & $P_2 < b$ where $a < b$ or $a > b$ ----- (5a)
$P_1 > a$ & $P_2 > b$ where $a < b$ or $a > b$ ----- (6a)

Equation (1a) can be read as “Parameter from system 1 is of value ‘a’ and is subject to constraint in increasing order that is, as P_1 increases, the utility changes in direction of satisfying the constraint”. In case (1a) and (2a), the parameter has the valid domain for variability and can be easily adjusted, say by taking mean of ‘a’ and ‘b’. In case (3a) and (4a), there are no overlapping domains, and hence these cases can only be solved by using other parameters. The system will not approach the solution if there is no other parameter. This case can generally be avoided by including more SDR blocks and hence incorporate more flexibility. In case (5a) the obvious solution would be to select the lower of the two values. Similarly for case (6a) it will be the higher of the two. The limitation we are facing currently is subjecting the system to multiple utility constraints, and we intend to use the approach described in this section.

3.6. Limitations

This system is developed on certain grounds which consider the limitations of the two platforms. Due to the flexible nature of the system and ability to give user the choice or preference in order to make it feasible for various scenarios, it is important to discuss the limitations and the possible solutions.

In case of determining the proportionality, an important consideration would be the dependence of the utility on multiple factors/parameters rather than independent one to one relationship. One point to consider is that though the slope between a Parameter and a Utility may vary depending on values of the other parameters, in most cases the proportionality will be maintained, which is what is important for determining the direction of improved performance.

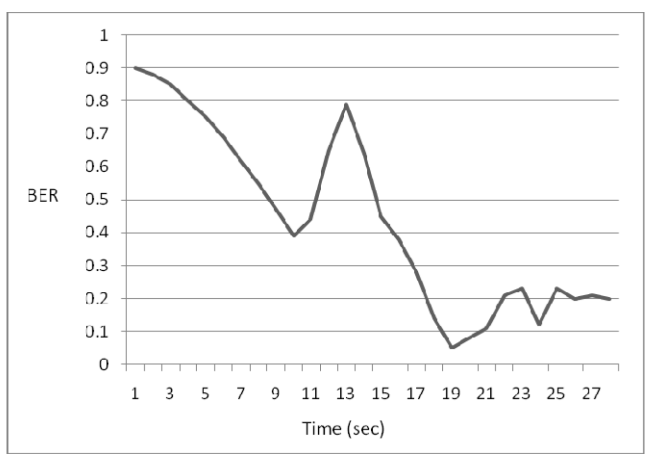
Since we do not rely on predefined equations and we want to reduce the computational complexity, it would make sense to adhere to fix step size. By keeping the step size fixed based on the flexibility provided by SDR, we can create perfect integration with CROSS, without resulting in crossing over to unstable or restricted domain values. Also the minimum and maximum value of the parameter is provided by SDR, so that the cognitive radio is aware of its lower and upper limits. The values of the Parameter are provided such that they have almost linear characteristics with the utility or at least maintain the same proportionality over the entire range of values. For example we are fully aware that increase in Transmit power will decrease the BER and hence decide to designate Transmit power as a Parameter or an SDR knob and BER as a utility. Similarly, the same approach can be applied for Modulation scheme and Bandwidth and other such cases. It may be required that the parameter values are such that there is constant change in utility value so that the relationships can be estimated accurately.

During the learning process, it is required that the system doesn’t undergo drastic changes that might invert the utility value trend over very short duration. This will make the cognitive radio to classify that parameter as unrelated. The linear characteristic in turn increases optimization time.

4 EXPERIMENTAL RESULTS

The time it takes to reach an optimal state depends on various factors such as number of parameters, step size for each parameter, and number of steps for each parameter. We experimented a simple case on USRPs to validate the results in real world scenarios. USRP is a low cost RF front end with a flexible platform [5]. Each block consists of about 128 bits and the estimated time for a batch of 100 block transmission (1 block set) is about 1 second. Consider a system with three parameters, Transmit power (Tp, 10 settings), Modulation Scheme (Ms, 4 settings) and Coding Gain (Cg, 4 settings), and the utility being BER, with a target of 0.2. The other parameters that are flexible are the PoC, equated to 1; and the weights are allotted as 1 : 2 : 3 (Tp : Ms : Cg). The PoC represents the percent change in utility value that can be considered as reliable for relationship estimation. This is to avoid misunderstanding with minimal variability due to randomness of channel. Coding Gain was provided with simple redundancy or repetition of bits. This can be replaced with more complex coding techniques like Turbo Coding. It would take at least 18 block sets (10 + 4 + 4) to validate the trend of each parameter individually and sequentially with the utility. The plot in Figure 2 shows the optimization curve we have observed.

Figure 2. BER Plot



The degradation in performance around the 13 second time frame is due to the system learning mechanism to realize that the increase in modulation scheme from BPSK to QPSK to 16QAM to 64QAM is actually degrading performance, or increasing BER. Once all the parameters have been verified (Time = 19s), the radio is functioning in the highest performing (lowest BER) state, but there is increase in cost (shown by TotalWeight) due to undesired higher performance. It uses the concept of weight ratio (section 3.2.), to optimize the system. The system is at a near optimal state at about 23 sec. Once the system has reached this state, only a few seconds are required for the system to adapt to maintain this state even in varying channel conditions, in most cases just 1 sec. In cases where the parameter utility relationship is not defined or inconclusive, the system uses the other parameters to optimize the system. Once that is completed, it will check the weights for all remaining possible combinations with inconclusive parameters. Due to limitations from the USRPs and the platforms, we are facing constraints with the data rates, but we believe that the speed of transmission of the blocks can be increased. This will reduce the time frame for the cognitive engine to perform computations, thus favoring this approach. Considering the current data rates in mobile devices to be about 1Mbps, this will reduce the feedback time to about 1ms and it would be a good approach to reduce the computation requirement from the receiver side. The approach can also be expanded for bidirectional data transfer where both control information and data are being transmitted in both directions. The redundancy in control information can be increased to reduce the effect of channel.

5 CONCLUSIONS

Effort has been focused on minimizing the computation time and making the API as flexible as possible. The reduction in the computation time can facilitate faster feedback and thus faster optimization of radio resources to cope with varying channel characteristics. We have also discussed the possibility for bidirectional flow optimization. Also the parameter types can be expanded to include multiple antennas or MIMO applications, Pulse Shaping, and other such signal characteristics. On the utility side, we can replace BER with Bandwidth or Data rate, or specific QoS requirements. By varying the Weights, we can provide QoS or user preferences.

6 REFERENCS

[1] Software Defined Radio Source Codes and information : http://www.ossie.wireless.vt.edu/

[2] Cognitive Radio Source Codes and information : http://www.cornet.wireless.vt.edu/

[3] J. L. Kolodner and D. Leake. A tutorial introduction to case-based reasoning. In D. Leake, editor, Case-Based Reasoning: Experiences, Lessons and Future Directions, chapter 2, pages 31–65. MIT Press, Cambridge, MA., 1996.

[4] S. Shiu and S. K. Pal. Foundations of Soft Case-Based Reasoning. Wiley Series on Intelligent Systems.Wiley-Interscience, Hoboken, NJ, 2004.

[5] http://www.ettus.com