AN EVALUATION OF COSTS AND BENEFITS OF AN ONTOLOGY AND POLICY BASED COGNITIVE RADIO

Shujun Li; Mieczyslaw M. Kokar; Jakub Moskal
Department of Electrical and Computer Engineering, Northeastern University, Boston
li.shuj@neu.edu, kokar@coe.neu.edu, jmoskal@ece.neu.edu

ABSTRACT

Cognitive radio is expected to have the capability to sense the changes of the environment and adapt its communications parameters accordingly. Ideally, adaptation shall be done autonomously, yet collaboratively. In this approach, an ontology is developed to capture the core concepts in wireless communications. This ontology is written in a formal language (Web Ontology Language) that has computer interpretable semantics and can be processed by an inference engine. The ontology is shared by the transmitter and receiver and provides a common knowledge base for them. Using the vocabularies provided by the ontology, a set of rules is developed to implement the adaption policies and a set of control messages are developed to enable collaboration between the transmitter and receiver. The interpretation and execution of the policies and incoming control messages are done in the inference engine. This paper focuses on evaluating the costs and benefits of the ontology and policy based cognitive radio in terms of time delay, communications overhead and performance improvement.

Keywords: Ontology, Policy, Cognitive radio, Adaptation

1. INTRODUCTION

Cognitive radio is expected to have the capability to sense the changes of the environment and adapt its communications parameters accordingly. Ideally, adaptation shall be done autonomously, yet collaboratively.

Autonomous adaptation means that the changes of the operating parameters can be implemented without human intervention. There are different levels of adaptation in radios: (1) At the low level, the adaptation algorithm is built into hardware. For instance, in 802.11a, radios are able to sense the bit error rate and then adapt the modulation to a data rate and the forward error correction (FEC) such that the bit error rate can be controlled at an acceptable low level. This algorithm is implemented in application-specific integrated circuit (ASIC) chips [2]. (2) At an intermediate level, the adaptation is software-defined. One way to achieve it is to hard code the adaptation algorithm into the radio. The shortcoming of this approach is that an algorithm hard-coded into the radio forms an inseparable part of the radio's firmware. Another way is to write the adaptation algorithm into a set of policies that control the radio behavior. This approach separates the adaptation policies from the implementation and thus exhibits more flexibility on the modification of the adaptation algorithm. (3) At the high level, the radio is able to learn from its experience and adapt its parameters without human intervention.

Collaborative adaptation involves collecting information from other radios and interacting with them to achieve a given goal. It requires (1) a proper way to exchange control messages between the radios, and (2) a proper way to interpret and execute the incoming control messages. Control messages shall be capable of expressing many more aspects than the current protocols can provide. For instance, instead of querying for a scalar parameter, cognitive radio shall also be able to query for more complicated information, such as the structure of a radio component or the finite state machine of a component. Additionally, the way to interpret and execute the incoming control messages is expected to be flexible and efficient. There are three possible ways to achieve collaborative adaptation [3]. The first way is to develop a communications protocol capable of expressing a wide range of aspects in wireless communications. On the one hand, it would increase the size of the header of the physical layer packets; on the other hand, it would be limited by the size of the header and the types of information that could be included in the header. The second way is to define a large vocabulary of control messages expressed in XML and including such messages in the payload of the packet. This approach provides more flexibility in that it can express more complicated information, however, it would require a XML schema to provide the description of the XML structure and procedural code to interpret the control messages written in XML. The third way is to give radios a formal language with computer-interpretable semantics in which any control message can be encoded, provided that it can be expressed in terms of ontology shared by the radios. This approach does not require a separate procedural code to interpret each type of control messages; instead it requires a generic interpreter, i.e. an inference engine (reasoner), to process the control messages written in a formal language such OWL (Web Ontology Language) or RDF (Resource Description Framework).

In [1], we proposed an ontology and policy based approach to enable autonomous and collaborative adaptation in cognitive radio. In this approach, the adaptation policy is written in a set of rules and the control messages are expressed in a formal language that can be interpreted and executed by an inference engine. This paper focuses on evaluating the costs and benefits of this ontology and policy based approach in terms of time delay, communication overhead, and performance improvement.

The rest of this paper is organized as follows. Section 2 reviews the ontology and policy based cognitive radio presented in [1]. Section 3 provides an evaluation of costs and benefits of the ontology and policy based approach. Conclusions and future work are in Section 4.

2. ONTOLOGY AND POLICY BASED COGNITIVE RADIO: AN OVERVIEW

2.1. An Example: Link Adaptation

In [1], we used a link adaptation use case to show how to achieve autonomous and collaborative adaptation of cognitive radio parameters using ontology and policy based approach. In this use case, there is a transmitter-receiver pair. The goal is to maximize the power efficiency, i.e. the information bit rate per transmitter watt of power. This is attained by fine-tuning the parameters in the transmitter and the receiver.

The list of tunable parameters (knobs) and observable parameters (meters) is shown in Table 1.

The goal is to maximize the following objective function (Unit: $Gbit / watt \cdot \sec$):

$$\boxed{objFunc = \frac{0.512}{10^{\frac{PowdB}{10}}\left[\frac{528\cdot(1+\frac{m}{2^m-m-1})}{v} + trainPeriod\right]}}$$

The constraints are listed as follows (the derivation of the constraints can be found in [1]):

- $10dB \le mSNR \le 15dB$.
- $PowdB \le 0dB$
- $\Delta PowdB = \Delta mSNR$.
- $3 \le m \le 7$.
- $v \ge 1$
- $\Delta v = \Delta mSNR / 6$.
- $5\cdot(M + N1 + N2) \le trainPeriod \le 10\cdot(M + N1 + N2)$

Table 1. Parameters of Transmitter and Receiver

Tx	PowdB	Transmission Power (Unit:dBm)	Knob
	trainPeriod	Length of training sequence (Unit: channel symbol)	Knob
	m	Index of (2^m-1, 2^m-1-m) Hamming Code	Knob
	v	Integer of QAM modulation, 4^v is the size of QAM constellation	Knob
	Payload	Size of payload field. Set payload=128 bytes	Fixed
	fracSpacing	Number of samples per symbol. Set fracSpacing=2	Fixed
	sampleRate	Number of samples per second. Set sampleRate=1000	Fixed
Rx	M	Number of feedback taps	Knob
	N1	Number of precursor feedforward taps	Knob
	N2	Number of postcursor feedforward taps	Knob
	mSNR	Mean Signal-to-Noise Ratio	Meter

In addition, M, $N1$, $N2$ may influence $mSNR$: the $mSNR$ will increase with M, $N1$, $N2$, until a sufficiently large equalizer for the multipath is achieved. After that point, increasing the equalizer dimensions will have no effect, except to increase the shortest possible training sequence.

2.2. Ontology and Policy

Ontology is a formal, explicit specification of a set of concepts in a specific domain and the relationships between these concepts. We developed a Cognitive Radio Ontology that covers the core concepts from the PHY and MAC layers of wireless communications. The ontology is written in OWL/RDF and includes the following top-level classes: (1) Object, (2) Process, (3) Quantity, (4) Value, and (5) UnitOfMeasure. The documentation of this ontology is published in [6] and [7]. Using the classes and properties defined in this ontology, we can develop policies to control the behavior of the radio.

Policy is a set of rules in the form of "IF-THEN". In our implementation, we used BaseVISor as the inference engine, thus the policies in the link adaptation are written in the BaseVISor syntax.

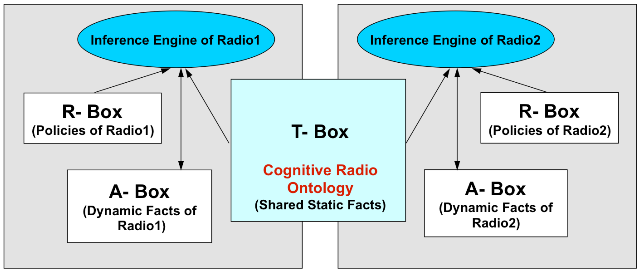

Figure 1 Ontology and Policy in Cognitive Radio

Figure 1 shows how to combine ontology, policy and inference engine in Cognitive Radio. T-Box contains the basic terms of the domain and includes the definitions of classes and properties as defined in the Cognitive Radio Ontology. T-Box is shared by the radios as a common knowledge base. R-Box contains the policies/rules, describing how to react to different situations. A-Box contains the facts that are only available when the radio is operating; they are instances of the classes in the T-Box and are generated by the system in run-time.

2.3. Control Model

There are two types of parameters in cognitive radio: *knobs* and *meters*. Knobs are the adjustable parameters that affect the performance of the radios; meters are the measurable parameters that can be observed and reflect the performance of the radios. Assume that there is a transmitter-receiver pair: radio A and radio B. Radio A sends a data packet to radio B. A simple control model would work in this way: radio B collects the knobs (e.g. transmitter power of radio A or hamming code index of radio B), the meters (e.g. signal-to-noise ratio measured at radio B) and other sensed information from the environment and computes the Quality of Service metric (e.g. the power efficiency). Then the overall Quality of Service is sent to the controller of radio B as a feedback. The controller then evaluates whether the goal is achieved (e.g. whether the power efficiency is within an acceptable range). If not, then the controller computes a new set of values for the knobs for the next transmission (e.g. a new value of the transmitter power at radio A) in order to achieve the goal.

2.4. Architecture

We implemented the link adaptation on GNU/USRP radios. GNU Radio is a free software development toolkit that provides signal-processing blocks to implement software radios using external RF hardware and processors. The Universal Software Radio Peripheral (USRP) is a high-speed USB-based board that enables general-purpose computers to function as software radios.

The architecture of the ontology and policy based cognitive radio is shown in Figure 2. There are five parts in this architecture.

Radio platform, i.e. the GNU radio, provides all the digital signal processing, software control, and the interfaces to interact with the RF, sensors, information source/sink and the policy reasoner.

System Strategy Reasoner (SSR) is an internal inference engine of the cognitive radio. It is the controller that gets feedback from the internal and external environment and forms strategies to control the operation of the radio.

Data In/Out (IO) is used to handle the incoming and outgoing messages. All the incoming messages from the RF are first processed by the Radio Platform and then passed to the Data IO. If it is a control message, Data IO passes it to Monitor Service; if it is a data message, Data IO will pass it to the Application layer. Similarly, the outgoing control message generated by the SSR and the data message generated by the Application layer is merged in a buffer in Data IO, then passed to the Radio Platform, and finally sent out through the RF.

Monitor Service (MS) is used to pass control message between SSR and Data IO. MS unwraps the outer part of the control message and passes the content to SSR. The content of the control message is written in OWL/RDF and thus can be processed by the inference engine in SSR. The outer part of the control message is defined by the FIPA ACL message structure and specifies the type of the control message. The details of the FIPA ACL message structure will be given in Section 2.5.

LiveKB provides a generic GET/SET API that allows the reasoned to access and adjust the parameters of the radio. The details of LiveKB are discussed in [4].

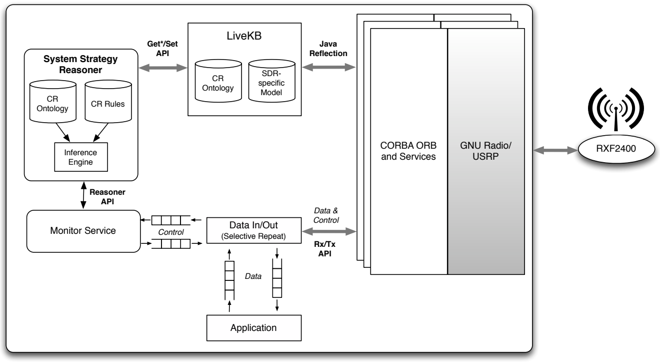

Figure 2. Implementation Architecture

2.5. Message Structure

FIPA Agent Communication Language (ACL) provides a standard set of message structures and message exchange protocols that support interoperability among agents and agent-based applications. In our implementation, we use the FIPA ACL to construct control messages.

A control message contains two parts: (1) a set of message parameters defined by FIPA ACL, and (2) the content defined by a FIPA ACL content language, e.g. OWL/RDF. The FIPA ACL defines 22 types of control messages such as query, request, inform, agree, etc. [5].

2.6. Adaptation Process

The link adaptation is accomplished by executing the policy rules for which the pre-conditions are satisfied. Figure 3 shows an example of the adaptation process in which Radio B queries for radio A's current transmitter power, then executes the adaptation policy to compute the optimized values of the parameters for the next transmission, and then requests radio A to change its parameter accordingly.

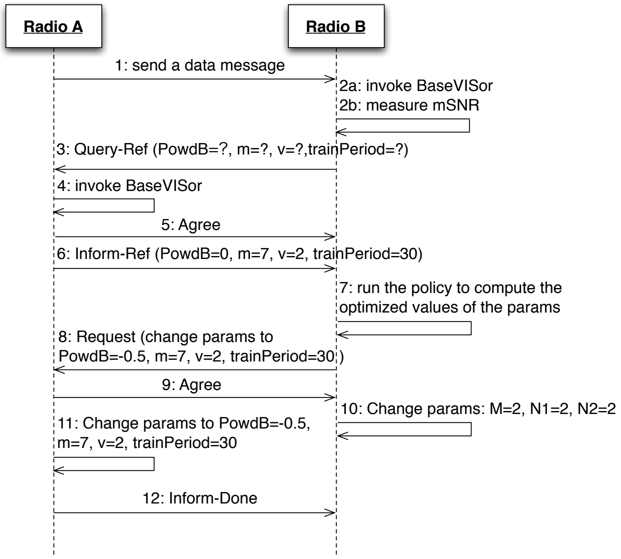

Figure 3. Sequence Diagram of Link

3. EVALUATION

The goal of the link adaptation use case is to maximize the power efficiency, subject to a set of constraints. In the case when there is no link adaptation, the values of the knobs of the transmitter and receiver are fixed, i.e. the radios keep using the initial values of their parameters during the transmission and there is no change of the parameters unless the user manually changes it. Thus the power efficiency remains at the same level while the *mSNR* fluctuates as the channel environment changes. Take the transmitter power as an example, if the initial transmitter power is set to a high level, then the radio may waste energy when the channel environment is "good" and *mSNR* is very high. In such a case, the transmitter can use a lower transmitter power to increase the power efficiency yet still maintain the *mSNR* in an acceptable range. Conversely, if the initial transmitter power is set to a low level, then it may lead to an increase of lost packets or corrupted packets when the channel environment is "bad" and *mSNR* is very low. In such a case, the transmitter shall use a higher transmitter power in order to bring the *mSNR* back to an acceptable range. Thus, it is necessary to adapt the radio parameters to the change of the channel environment. This can be achieved by the approach described in Section 2. In order to evaluate the benefits and costs of the ontology and policy based radio adaptation, this section assesses the performance improvement, time delay, and communications overhead, respectively.

3.1. Performance Improvement

In our experiment, radio A is the transmitter and radio B is the receiver, both of them are operating in half-duplex mode. In each run, radio A sends an image of 10000 pixels to radio B. Each pixel is sent as an individual packet and the size of each packet is the same. Assume that the initial transmitter power is 15dBm, in the case when there is no adaptation, radio A uses the same transmitter power to send all the 10000 packets. In the case when there is adaptation, radio A uses the initial transmitter power to send the first few packets until the *mSNR* measured at radio B is out of range. Then radio B will triggers the adaptation policy, computes a new value of the transmitter power, then requests radio A to change its transmitter power accordingly. The power adaptation process continues until radio A finishes sending all the packets. We change the initial transmitter power from -37dBm to 15dBm with uneven intervals. For each initial transmitter power, we run the experiments for 10 times for the case without adaptation and another 10 times for the case with adaptation. Then we compute the average power efficiency, mean signal-to-noise ratio and average corrupted packet rate for each case.

Figure 4-6 shows the comparison results of the communications link with adaptation and without adaptation, in terms of mean signal-to-noise ratio, power efficiency and corrupted packet rate. All the x-axes are the initial transmitter power.

It can be seen that (1) when the initial transmitter power is smaller than -10 dB, the use of adaptation can yield smaller power efficiency, but the corrupted packet rate is smaller due to higher *mSNR*. Smaller corrupted packet rate means that there will be less traffic imposed to the network because the radios have less need to re-send the packets. (2) When the initial transmitter power is larger than -10dB, the use of adaptation will increase the power efficiency, yet it will not increase the corrupted packet rate, i.e. in Figure 7, when initial transmitter power is larger than -10dB, the blue line ("with adaptation") and the red line ("no adaptation") are almost at the same level.

3.2. Time Delay

As shown in Figure 3, in our implementation of the link adaptation use case, the radio is able to generate five types of messages: Query-Ref, Agree, Inform-Ref, Request, and Inform-Done. To make the case simpler, we assume that radio A always agrees to an incoming query or request. To implement this, MS generates an "Agree" when it receives a

"Query-Ref" or "Request", rather than passing it to the SSR and let the inference engine make the decision. All the other four types of control messages are generated by the inference engine. In order to evaluate the time delay imposed to the system due to the use of ontology and policy based approach, we measured the response time needed to generate each type of control message in the inference engine. The response time depends on the type of control message and the size of the search space, i.e. the number of facts (triples) in the knowledge base. For evaluation purposes, we created five ontologies with different sizes: each of which was used as the T-Box shown in Figure 1. For instance, we use the ontology with 500 triples as the T-Box, then we run the sequence shown in Figure 3 for 50 times and measure the average response time for each control message generated by the inference engine. Then we run the experiment again using the ontology with 1000 triples, 1500 triples, 2000 triples and 2500 triples. Figure 7 shows the average response time of each control message type for T-Box with different size along with the standard error for each one. It can be seen that: (1) the response time to generate "Query-Ref" and "Inform-Ref" increases proportionally to the size of the T-Box. (2) The response time to generate "Request" and "Inform-Done" does not increase as the size of the T-Box increases. In addition, it is much less than the time to generate "Query-Ref" and "Inform-Ref".

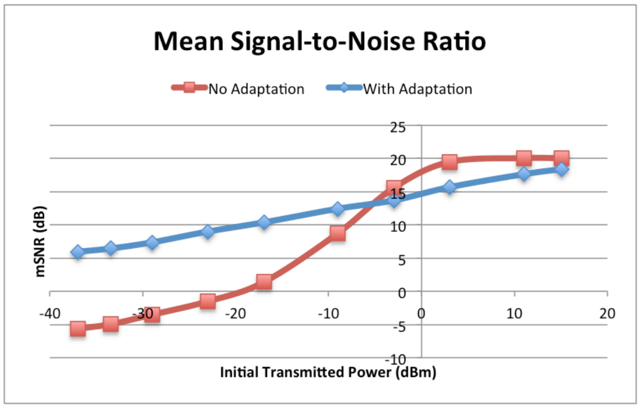

Figure 4 Performance Evaluation (1): Mean SNR

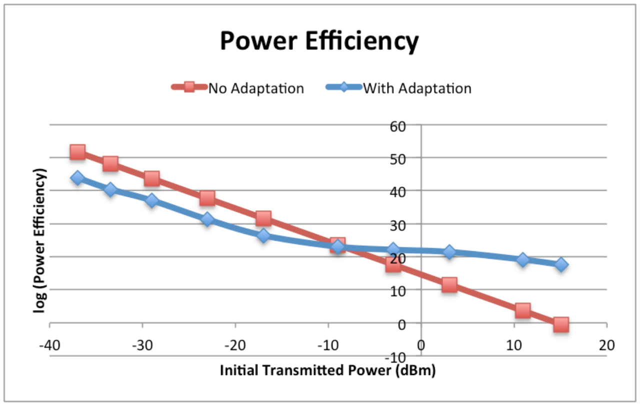

Figure 5 Performance Evaluation (2): Power Efficiency

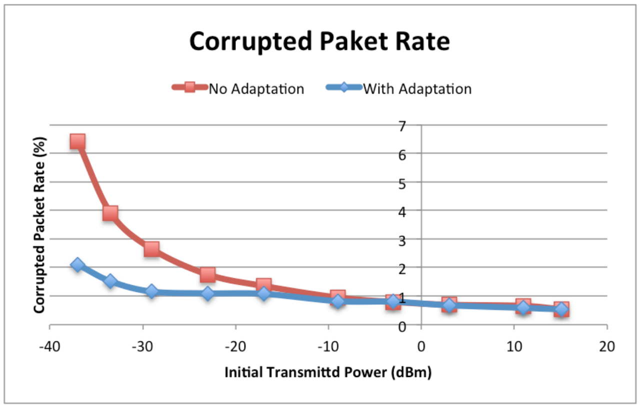

Figure 6 Performance Evaluation (3): Corrupted Packet Rate

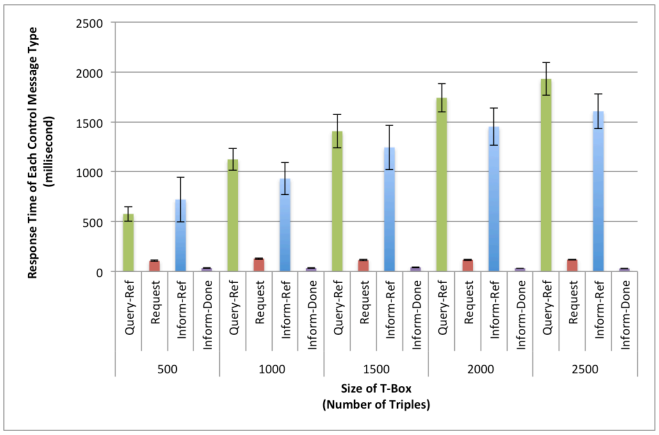

Figure 7 Response Time of Each Control Message

3.3. Communications Overhead

Assume that we need to represent the value of a parameter in the fixed protocol approach and there are 3000 tunable parameters and the value of each parameter is of Double type, then we need 12 bits to represent the name of each parameter, and another 64 bits to represent the value of each parameter, i.e. in the naïve way, each parameter requires 10 extra bytes in the header.

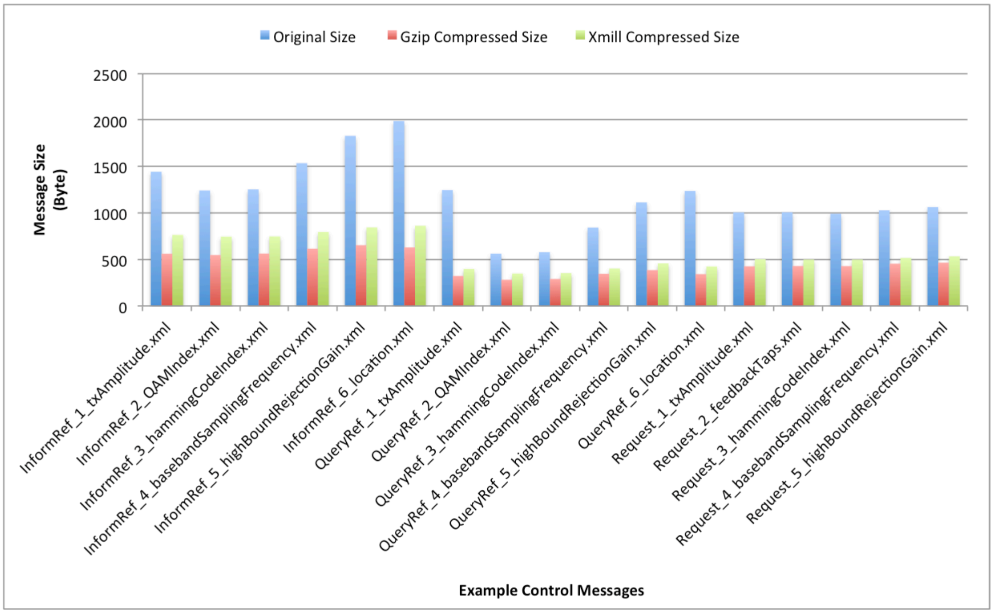

Figure 8 Communications Overhead of Ontology and Policy Based Approach

In the ontology and policy based approach, such information is represented in OWL/RDF as a control message and is put into the payload of the packet rather than the header. It requires more bytes than the fixed protocol approach because control messages are written in OWL/RDF. To evaluate how much communications overhead is imposed to the network by the ontology and policy approach, we created 17 different control messages, each of which represents either a quest, a request, or an inform of a parameter in the radio. Figure 8 shows the original size and the compressed size of each control message. Two compression tools were used: Gzip and Xmill. It can be seen that Gzip has a higher compression ratio than Xmill. In the case when one parameter needs to be represented in the OWL/RDF control message, it requires 455 extra bytes in the payload.

5. CONCLUSIONS AND FUTURE WORK

In summary, we evaluate the costs of benefits of the ontology and policy based cognitive radio in terms of time delay, communications overhead and performance improvement. Though the use of ontology and policy based adaptation leads to some time delay of the system and communications overhead to the network, it improves the link performance and provides more flexibility to enable autonomous and collaborative adaptation of cognitive radio. In the future, we will further investigate the inference capabilities of the ontology and policy based approach and compare this approach with a non-semantic approach, such as XML.

6. ACKNOWLEDGEMENT

This work has been partially supported by a DARPA contract through System Planning Corporation (Agreement No. SRA-0775). The views, opinions, and/or findings contained in this article/presentation are those of the author/presenter and should not be interpreted as representing the official views or policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the Department of Defense.

7. REFERENCES

[1] S. Li, M. M. Kokar, D. Brady, J. Moskal, "Collaborative Optimization of Cognitive Radio Parameters Using Ontology and Policy Based Approach", SDR Forum Technical Conference, Dec. 2010.

[2] J. Polson, B. A. Fette (Ed.), "Cognitive Radio Technology", Chapter 4 "Cognitive Radio: The Technologies Required", Elsevier, August 2009.

[3] S. Li, M. M. Kokar, J. Moskal, L. Pucker "Now Radios Can Understand Each Other: Modeling Language for Mobility", URL: http://data.memberclicks.com/site/sdf/MLM2011.pdf, Mar. 2011.

[4] J. Moskal, M. M. Kokar, S. Li, "Interfacing A Reasoner with An SDR Using A Thin, Generic API: A GNU Radio Example", SDR Forum Technical Conference, Dec. 2010.

[5] FIPA Communicative Act Library Specification, available at http://www.fipa.org/specs/fipa00037/SC00037J.pdf.

[6] Wireless Innovation Forum MLM Working Group, "Description of Cognitive Radio Ontology v.1.0", WINNF-10-S-0007, available at http://groups.winnforum.org/d/do/3370, 2010

[7] S. Li, M. M. Kokar, D. Brady, "Developing an Ontology for the Cognitive Radio: Issues and Decisions", SDR Forum Technical Conference, Dec. 2009.