MAPPING SPECTRUM CONSUMPTION MODELS TO COGNITIVE RADIO ONTOLOGY FOR AUTOMATIC INFERENCE

Yanji Chen (chen.yanj@husky.neu.edu)[1], Mieczyslaw M. Kokar (m.kokar@neu.edu)[1], Jakub J. Moskal (jmoskal@vistology.com)[2], and Durga Suresh (d.suresh@neu.edu)[1]

[1]Northeastern University, Boston, MA, USA
[2]VIStology, Inc., Framingham, MA, USA

ABSTRACT

Radio frequency spectrum management plays a critical role in various domains, including government, military, industrial and personal communications. Current methodology of spectrum management relies primarily on licensing, i.e., giving control over a specific part of the spectrum to a limited number of providers, who then ensure that there is no interference among the communicating radios. This approach, however, leads to an underutilization of the spectrum.

To address these issues, various dynamic spectrum access and management approaches have been investigated. Most, if not all, approaches rely on an exchange of spectrum information among the various communication nodes. In this paper we are using the Model-Based Spectrum Management approach based on the creation and exchange of Spectrum Consumption Models that are expressed in Spectrum Consumption Modeling Markup Language (SCMML) that uses the eXtensible Markup Language (XML) schema definition. We have mapped SCMML to Web Ontology Language (OWL) – the formal language used in the Semantic Web. We show that it is possible to add axioms to such an OWL representation of SCMML and then use the resulting representation to infer facts that were only implicitly represented in the data. The paper discusses two spectrum management related use cases to explain the approach and its potential.

1. INTRODUCTION

Nowadays, spectrum assignment is implemented mainly through a static, reserve-based approach in which spectrum is licensed to providers who manage the spectrum by dynamic assignments to the subscribers (primary users). This approach, unfortunately, in some situations results in the underutilization of the spectrum, as documented in various studies, e.g., [1].

One of the possible approaches to improve the situation with the efficiency of spectrum utilization is to allow for an opportunistic access to spectrum, i.e., allow unlicensed users to access the spectrum where and when it is not used by the primary users. In this kind of spectrum sharing, the rights of the primary users must be preserved, but also the potential uses of the spectrum by the opportunistic users must be specified by policies.

Policies can be specified, represented and executed in many different ways. One of the possibilities is to establish a standard language for describing policies. Additionally, if the language is formal, then the execution of the policies can be achieved by using cognitive engines that are capable of interpreting such policies automatically. This kind of language is being standardized by the IEEE 1900.5 Working Group. The work of this group has resulted in the publication of the requirements for such a policy language [2]. Currently, this Working Group is preparing a standard (IEEE 1900.5.1) for a policy language that partially satisfies the requirements specified in [2]. Additionally, this group is working on another related standard (IEEE 1900.5.2), which will capture the specification of Spectrum Consumption Models [3].

Spectrum Consumption Models (SCMs) are data structures by spectrum management policies. They are inherent to the approach called Model-Based Spectrum Management (MBSM) [4, 5]. The intent of the MBSM approach is to concentrate on the consumption of spectrum, provide computational methods for assessing compatibility among models, serve as a loose coupler for spectrum management systems and enable further extensions of spectrum use and sharing. In other words, a consumption model (expressed as an SCM in SCMML) needs to be compatible with the policy, which also makes use of SCMs to describe the authorization for an allowed use of the spectrum.

The paper firstly discusses the mapping of the SCMML XSD schema to an ontological representation. After that, two use cases are demonstrated in order to show the feasibility of the approach in which SCM descriptions are converted to the OWL language [6], which in turn is used by automatic inference to derive decisions regarding the satisfiability of the spectrum use policies. The first use case is related to querying reported transmitter movements based on location - whether the transmissions from these transmitters are acceptable in the area defined by the policy. The second use case shows that an inference engine can infer whether the receiver can tolerate the interfering signal from a remote transmitter. The paper ends with conclusions and suggestions for future research.

2. MAPPING SCM TO OWL

While expressing SCMs in SCMML is a step towards the formalization of data models - providing a formal syntax for such models - these representations cannot be processed by automatic inference engines due to the lack of formal semantics for XML. Instead, SCMs can only be processed by procedural code written for a specific XSD schema. Any updates to XSD would require new code for interpreting the newly introduced tags. In order to avoid such a kind of situation we are advocating the use of a language with formal, computer processable semantics, the Web Ontology Language (OWL) [6].

As one of the steps in this direction, we developed a tool that can automatically transform any XML schema definition to OWL. The conversion result of the SCMML XSD file to OWL is called here the SCMML ontology (SCMMLO). The justification for calling this an ontology is that the XSD schema captures some of the domain knowledge, including some classifications, individuals as well as some relations. In its creation, the OWL generation code maps the hierarchical structure of the XSD into the classes, properties (relations) and individuals in the resulting ontology.

Nevertheless, the structural information represented in SCMMLO still does not have much of the semantic richness. It primarily captures the tree structure of the XSD schema. In order to enhance the semantics of this ontology, we are embedding it in a more comprehensive ontological structure shown in Figure 1. This figure shows the *import* relationship among the ontologies. According to this structure, ontologies in the bottom layer can reuse components from the upper layer and still can add restrictions/components without making any modifications to the upper layer.

As shown in Figure 1, the SCM ontology (SCMO) is added to the structure. SCMO provides the glue that connects SCMMLO to the whole structure. Additionally, SCMO imports Cognitive Radio Ontology (CRO) developed at the Wireless Innovation Forum [7]. This figure also shows Nuvio, a foundational ontology developed by Northeastern University and VIStology. Then the use cases discussed in this paper can be represented in SCMO, as shown at the bottom of this figure.

The SCMO ontology includes a number of axioms that link SCMMLO and CRO. Table 1 shows the types of axioms in SCMO inserted for merging SCMMO and CRO; the "Category" column lists the type of axioms added, while the "Examples" column shows some of the examples of axioms for each category.

The Location and Power Margin use cases were represented in SCMO (to be discussed later in the paper). The scenarios for these use cases were added and then inference over such representations was carried out using BaseVISor [8], one of the inference engines for ontologies. At this point we could use a standard query language (SPARQL [9]) to query the inference results. In our experiments, however, we used the BaseVISor rules syntax to represent queries and to obtain the results.

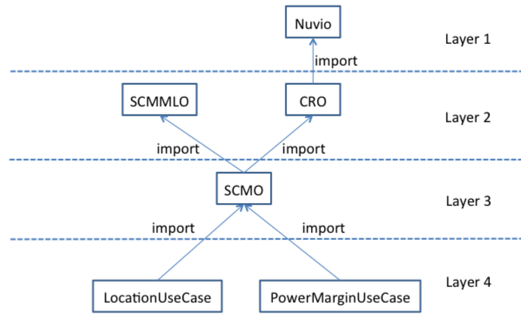

Figure 1: SCMML ontology in the context.

3. SCM AND SCMML

3.1. Why SCM?

3.1.1. Problem Statement

Spectrum management is supported by various tools. For instance, the MITRE report [4] states that at least 46 different tools are in use, presumably more than one tool by a particular spectrum manager. In this situation, and considering that agreements on the use of spectrum may require negotiations among the different stakeholders (and thus different tools), the interoperability among the tools is very desirable, if not necessary. In particular, such tools need to exchange models of the spectrum situations. As illustrated in Figure 2 [4], data and models of terrain and propagation effects are stored in tools embedded in different systems. Managers make spectrum allocation decisions by using the tools as well as their knowledge of the operational use of system and other systems operating in the same frequency band. The decision (left hand side) communicated to the other tool (right hand side) includes some data that fails to capture the knowledge created in the planning process (known to the user of the first tool). In this scenario, the manager that uses the second tool is left in the dark – getting just the results does not provide any information on how to interpret how the decision was made. To resolve this gap requires a transfer of data and information between the tools. For this, a unified set of models that eliminate incompatibility among systems and transfer of the knowledge between systems and spectrum managers is desirable. The intent of SCMs is to fill this gap.

Another issue resides in the sharing of data among stakeholders. Communicating parties often do not want to fully share data with others or trust the results of other parties' analysis. Each party may have its own rules on deciding what constitutes interference. Therefore, negotiating the access to spectrum plays a critical role for achieving a more permanent agreement based on operation conditions.

Table 1: Embedding relations.

Components	Category	Examples
Class	Add subclass restriction	SCMMLO:Circle rdfs:subClassOf Nuvio:Object
Class	Add equivalent class	SCMMLO:Transmitter owl:equivalentClass CRO:Transmitter
Property	Add sub property	SCMMLO:hasBand rdfs:subPropertyOf Nuvio:compositeOf
Property	Add domains/ranges	SCMMLO:hasAltitude rdfs:domain SCMMLO:Point
Property	Add domains/ranges	SCMMLO:hasAltitude rdfs:range SCMMLO:Distance

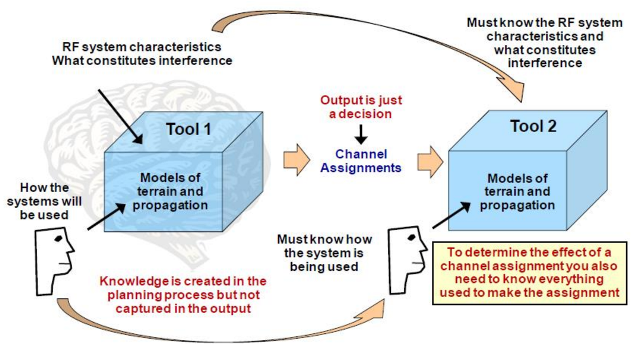

Figure 2: Limited flow of information.

3.1.2. *Advantages of SCM*

To address the issues mentioned above, Figure 3 (from [4]) shows the effect of SCMs on spectrum management.

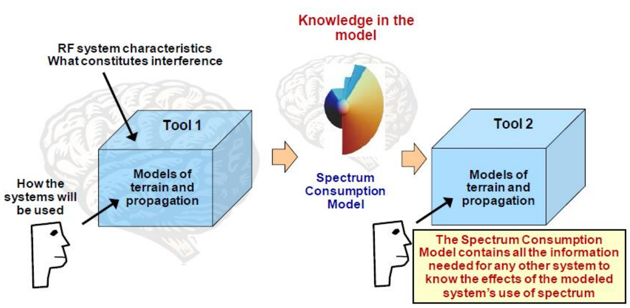

Figure 3: SCMs in spectrum management.

As can be seen from this figure, spectrum management tools exchange SCMs, not just decisions. SCMs represent models that capture relevant aspects of a system's use of spectrum. According to [4], these models provide unambiguous definitions of the extent to which a system emits radiation, what would cause harmful interference with that system's operation, and under what circumstances. They describe specific uses of spectrum as opposed to the general characteristics of systems captured in system data. Therefore, they provide a means to capture and use the judgment of mission planners and spectrum managers [4]. More specifically, the benefits for SCM mainly lie in the following three aspects.

Focus on spectrum consumption: Management tools and spectrum managers do not have to share details about the RF components of systems and about specific system missions [4]; the model captures these details abstractly.

Compute compatible reuse: A SCM created by one system provides sufficient information to allow other management tools to compute compatible reuse [4]. Therefore, the assessment of compatibility is the same anywhere across the entire SM system for the same models.

Serve as a loose coupler: An effective loose coupler standardizes a small portion of a system at the intersection of what must be shared between the layers and across the layers [4]. SCM serves as a loose coupler among SM systems and RF systems because it provides a means of sharing the data necessary at their intersection. The shared data consist of models of spectrum consumption and the attendant computations used with these models to arbitrate compatibility.

4. CONVERTING SCMML TO SCMMLO

As was mentioned earlier in this paper, in order to utilize the work performed within the SCM standardization effort, we convert (automatically) the SCMML XSD schema to OWL. Since SCMML is a work in progress, modifications to its schema are likely; each such modification would require the repeat of the conversion process. Automatic conversion will allow for keeping the OWL representation up to date with respect to the latest (newer) versions of the SCMML schema. To this end, we used a tool developed by VIStology, which can take an arbitrary XSD schema and turn it into an OWL ontology. In the terminology of the Semantic Web (cf. [10]), the part of the ontology that holds the concepts of the ontology (classes and properties) is called the TBox. The part that holds the instance data, on the other hand, is called the ABox. The tool thus produces not only the TBox, but also an instance generator, which can convert XML documents to OWL ABox, conforming to the generated TBox. The generated TBox is produced according to the patterns described in [10] and maintains majority of the integrity constraints expressed in the schema. Because the tool is independent of a particular schema, it can be applied to any new version of the SCM XSD.

5. USE CASES

In order to show how SCMO can be used in the domain of spectrum management, two use cases are demonstrated in this section. The first one is related to dynamic spectrum management based on location and spectrum allocation restrictions for different stakeholders. The second use case deals with assessing whether signals from specific transmitters might be harmful to receivers, before actual transmission occurs. For both use cases, the models that are built upon SCMO, rules that can assert more facts, and the results obtained by using the BaseVISor inference engine are described. In the end, potential advantages of using our method, as opposed to other methods, such as using an imperative programming language to directly implement the interpretation procedures for the particular tags of XML representations of SCMs, are discussed.

5.1. Location Use Case

5.1.1. Scenario Description

In cellular networks, spectrum allocation is done by the process of back and forth communication via a control channel between transmitter and base station through which a spectrum band is allocated to the mobile station by the base station. However, some spectrum may not be used sometime to avoid interfering other radios operating in the same area, e.g., primary users. In order to make a decision on which mobile stations can use the particular spectrum, the base station needs to have some information about the mobile station and its needs. For exchanging information about spectrum consumption and needs, the communication nodes can use SCMs described earlier in this paper. An SCM can serve either as a constraint or as an authorization, i.e., it can specify what frequency bands can/cannot be used in what subareas within the transmission scope of a base station. This information can be stored in, or retrieved from, a database of the corresponding base station. Thus the SCM provides information that is useful for making spectrum allocation decisions by the base station. In its inference, the base station makes use of some facts related to spectrum availability in a particular location, e.g., information about the transmitters/emitters located in the area, or definitions of subareas and regulatory policies applicable to the specific subareas. The following shows a typical scenario.

Assume that the cube in Figure 4 shows the coverage of the base station (base station is not shown here) for this scenario. The policy for this space is shown in Table 2.

Table 2: Spectrum policy for the scenario.

	A	B	C	D
Cube	-	-	+	+
Cylinder	-	-	+	-
Outside	-	-	-	-

Suppose that currently there are four secondary transmitters, marked as A, B, C, and D in this figure. A is outside of both spaces (cylinder and small cube), B is within the cylinder, C is in the intersection of the cylinder and the small cube, and D is within the small cube. Since A is outside of both spaces, it cannot transmit since none of the secondary users can use the spectrum in this region. Our expectation is that our model can get the result that C and D can use their given spectrum in the small cube. Also, C can use its spectrum in the cylinder as well. As for the others, either a reallocation procedure is needed (for B in this scenario) or a termination of the use of a given spectrum should be processed (for A in is this scenario).

5.1.2. Structure of the Model

Figure 5 shows how the Location use case is captured in the ontology. This figure is generated from an OWL file using Ontoviz plug-in embedded in Protégé 3.5 [11, 12]. The rectangles represent classes in OWL, such as LocationUseCase:Shape, SCMML:Point, etc. I.e., the former represents the class Shape (collection of shapes), while the latter represents the class Point (i.e., any point in 3D space). The prefix before the colon represents which ontology the class comes from. For instance, SCMML:Point means the class named Point is originated from the SCMML ontology. The arrows that connect classes represent relationships between elements of the classes. They are either relations between sets of objects (instances of the classes), called object properties, or subclass relations, where the semantics of such relations is defined by the subset relation (class A is subclass of class B means that each individual of class A is also an individual of class B). Individuals are not shown in this figure because of the space limitation.

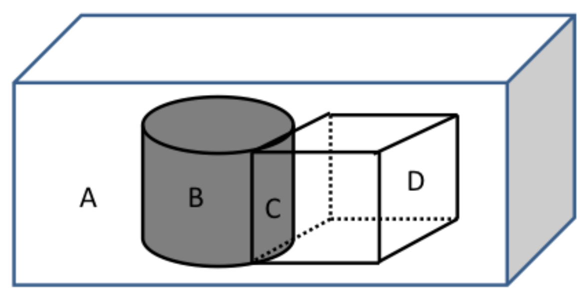

Figure 4: Location use case: Scenario.

The two central concepts represented in Figure 5 are Cylinder and Cube. They both are subclasses of ThreeDShape – Cylinder directly, while Cube indirectly via Polyhedron. The two shapes are constructed using their bases – Circle for Cylinder and Rectangle for Polyhedron. The circle is defined by a point (connected via hasCenter) and a radius (linked via hasRadius). The polygon is defined by vertices (linked via hasVertex). The bases are connected to the 3D shapes via the hasBase property. The third dimension is then added via the hasHeight property which links

3D shapes with the class Distance. All of these representations follow the same structure as the XSD schema that defines the SCMML. In other words, this representation is compatible with the representation promoted by the MBSM approach.

5.1.3. Policies and Rules

Since the inference engines embedded in Protégé lack the capability of quantitative computation, as well as due to their restrictions with respect to the representation of policies and rules, we used the BaseVIsor inference engine that can represent rules in its own (RDF-like) syntax, where the rules can also invoke procedural attachments (calls to procedures written in Java). Such procedural attachments are needed for performing quantitative operations. BaseVISor rules are based on an ontology that BaseVISor imports. The following describes the rules that were used in our experiments, along with a short explanation of their meaning. The syntax of the rules is not shown due to space limitations.

Rule 1: Detect whether a cube contains a point where a transmitter is located. A point has altitude, latitude and longitude coordinates. The representation of cube is shown in Figure 5. Therefore, the rule can infer whether a point is within or outside a cube by comparing the corresponding coordinates of both the point and the coordinates of the vertices and the height of the cube.

Rule 2: Detect whether a cylinder contains a point where a transmitter is located. The idea is similar to that of Rule 1. A cylinder is modeled by a circular base, that in turn is modeled by a circle center point and a radius, as well as the height (distance). Therefore, if a point is within a cylinder, its height does not exceed the height of the cylinder. Also, the horizontal distance between a point and circle center point of the base does not exceed its radius. To implement this rule we had to use a procedural attachment to perform the quantitative operations required by this rule.

Rule 3: Infer which transmitters can transmit in their current locations. This rule uses the results of the previous two rules to infer whether the transmitters are allowed to transmit according to the current policies associated with the space regions defined by the cube and the cylinder. The interpretation of the permissions in this case is that if a transmitter does not get a permission to transmit, it cannot transmit.

5.1.4. Inference Result

The result of running BaseVISor (a capture of the Terminal window) is shown in Figure 6. The result is, as expected, that both transmitters can transmit when they are in the cube, but only C can transmit in the cylinder.

```
TransmitterC can transmit in TestCube
TransmitterC can transmit in TestCylinder
TransmitterD can transmit in TestCube
```

Figure 6: Inference result for Location use case.

5.2. Power Margin Use Case

5.2.1. Scenario Description

When two radios communicate via a channel shared by other nodes, a multiple access protocol must be utilized. Contention protocols resolve a collision after it occurs, while collision-free protocols ensure no collision. However, the former requires a lot of back and forth communication to deal with collisions while the latter reduces channel utilization ratio when transmission overload is not high. Therefore, it is necessary for a transmitter to detect whether a signal may interfere with receivers that are not target receivers before actual transmission occurs. In this paper we use ontology based inference in order to address these issues. Here we discuss a Power Margin use case to explain our approach. In this use case, a request is required from the transmitter to the base station on whether the transmitter is allowed to transmit, before transmitting. The base station then returns a decision derived by its inference engine. The inference is based on the ontology that contains the knowledge about the domain and the current situation. Figure 7 shows a typical scenario.

As shown in this figure, this scenario includes four transmitters, marked as TA, TB, TC and TD, respectively. Additionally four receivers are marked as RA, RB, RC and RD, respectively. Suppose TA is transmitting to RA and TB is transmitting to RB. If TC wants to transmit to RC and TD to RD at the same time, we want to know whether these transmissions may interfere with the other receivers that are not target receivers for these transmitters.

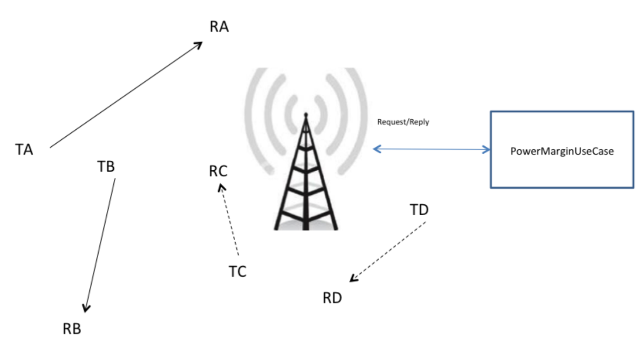

Figure 7: Power Margin use case scenario.

5.2.2. Relevant MBSM concepts

Several concepts are proposed in the MBSM approach to deal with this problem [4].

Spectrum mask: Spectrum mask specifies the power-spectrum density vs. frequency, relative to the total power of a transmitter. In the scenario considered here, power spectrum density varies by location since the signal attenuates with propagation and varies with terrain.

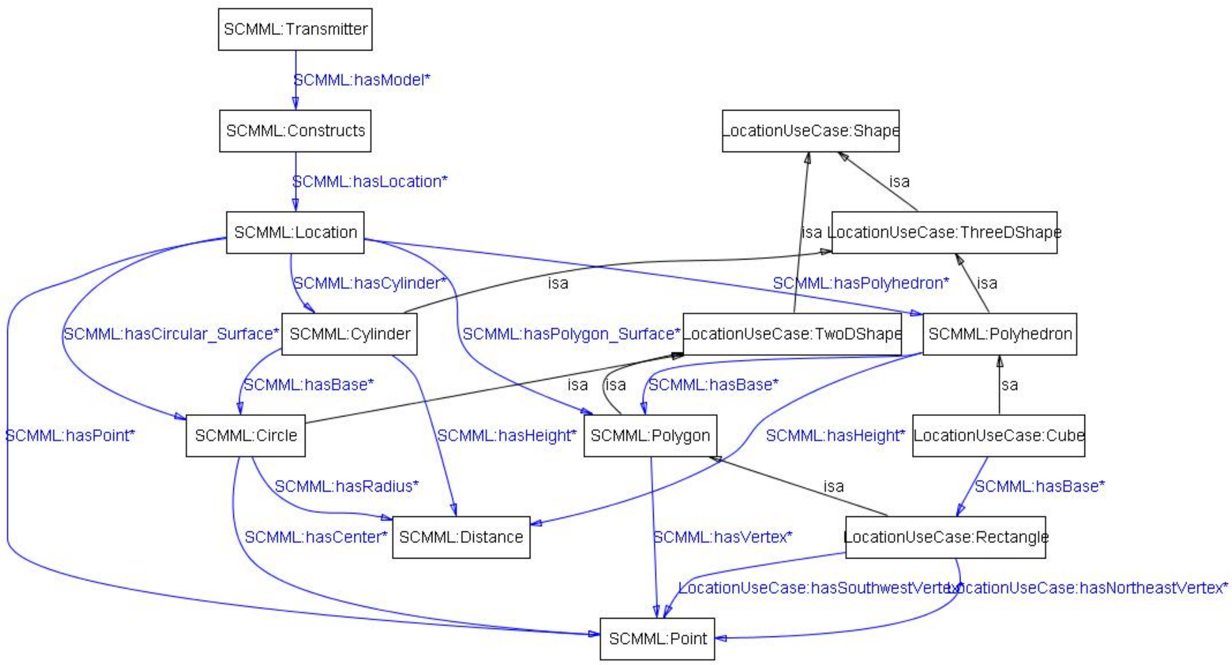

Figure 5: Representation of Location use case in ontology.

Underlay mask: Underlay mask specifies the power spectrum density of a signal that a receiver can tolerate from a remote interfering transmitter; it is represented as a function of relative power versus frequency. It defines the maximum power level of the anticipated interference at the receiver as a function of frequency.

Power margin: Power margin is the minimum power adjustment that would have to be made to make the spectrum power mask and the underlay mask meet. When the masks are represented by inflection points, power margin is the adjustment that would have to be made across the overlapping inflection points that would cause the two masks meet.

Both spectrum mask and underlay mask are usually represented as piecewise linear graphs of power spectrum density. Figures 8 to 11 show spectrum masks of transmitters TA, TB, TC and TD at the the locations of receivers RA, RB, RC and RD, respectively, the underlay masks of receivers RA, RB, RC and RD, respectively, as well as the power margins.

Power margin can be computed using the total power method or the maximum power spectrum density method. In our implementation of this use case we used the latter method. This method is used when multiple interferers do not coordinate their interference with each other (as was assumed in our use case) to ensure that they collectively stay within the limits of allowed interference. For this, each of the transmitters must stay within the limits of the underlay mask.

In this method, first, both the underlay mask and the interfering signal's spectrum mask are converted to the same resolution bandwidth. For instance, in our scenario the bandwidth of the spectrum mask and the underlay mask were 1kHz and 10kHz,

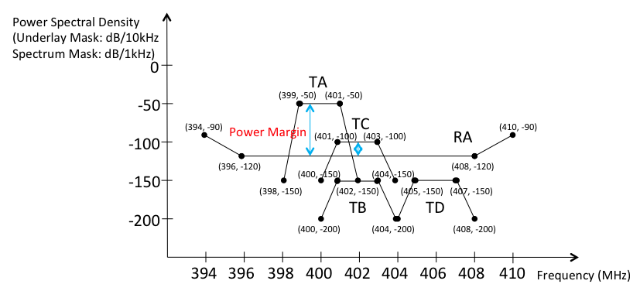

Figure 8: Spectrum masks vs. the underlay mask of RA.

respectively. Second, we had to infer the value of the power margin.

Following this procedure, we can estimate that TC would interfere RA (seen from Figure 8) and RD (seen from Figure 11), while TD would interfere RB (seen from Figure 9), i.e., both transmitters TC and TD should not transmit signal to avoid interfering the receivers that are not their targets.

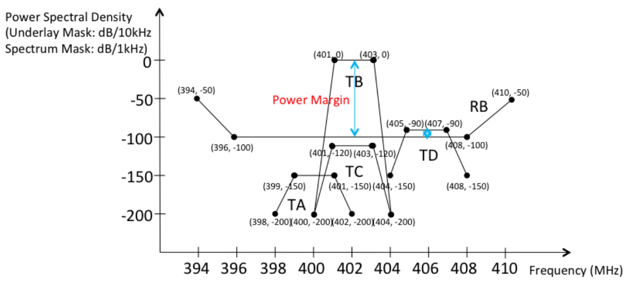

Figure 9: Spectrum masks vs. the underlay mask of RB.

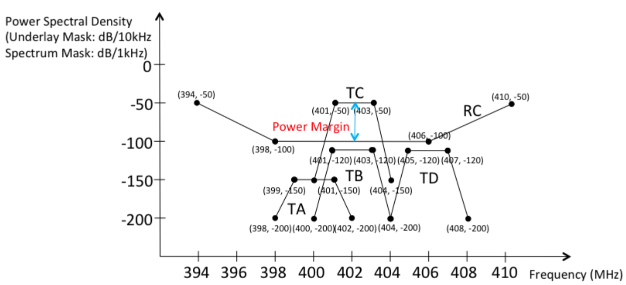

Figure 10: Spectrum masks vs. the underlay mask of RC.

5.2.3. *Structure of the Model*

Based on the above analysis, the Power Margin use case ontology was developed (that imports SCMO) to which then some details were added. Similarly as for the Location use case, Figure 12 shows the classes and the relationships among the classes. Again, individuals are not shown in this figure because of the space limitation of the paper.

5.2.4. *Policy and Rules*

The solution of this problem requires inferring whether sufficient power margins exist for the redeiver-transmitter pairs. To implement this kind of inference we had to add one rule.

Rule 1: Detect whether power margins between spectrum masks and underlay masks exist for each receiver-transmitter pair.

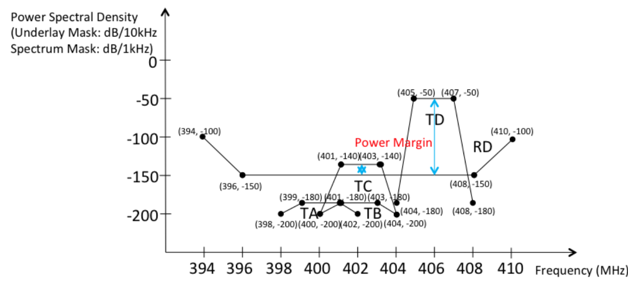

Figure 11: Spectrum masks vs. the underlay mask of RD.

This rule can be specified as follows. For all transmitters and receivers, find all the transmitter-receiver pairs s.t., the receiver is not the target receiver for the receiver (since we are only concerned about interfering signals), and s.t., sufficient power margin exists for each such pair. The implementation of the rule was based on the comparisons of the appropriate inflection points that represented the underlay and the spectrum masks.

5.2.5. *Inference result*

Similarly as for the Location use case, the result of running BaseVISor (a capture of the Terminal window) is shown in Figure 13. The result is, as expected, that transmitter TC would interfere RA and RD, while TD would interfere RB. Therefore, both transmitters should not transmit at this time.

```
TransmitterC will interfere ReceiverA
TransmitterC will interfere ReceiverD
TransmitterD will interfere ReceiverB
```

Figure 13: Inference result for Power Margin use case.

6. USING OWL VS. IMPERATIVE LANGUAGES

Both OWL and rule languages fall in the category of *declarative programming*. Unlike the case for *imperative* languages (e.g., C, C++, Java), the code in declarative languages does not include any “control knowledge”, i.e., the code does not include statements that explicitly direct the thread of execution to specific blocks of code, when some conditions are satisfied or states are reached. Instead, OWL code contains collections of facts, some of them representing generic knowledge, e.g., that each radio must have a transmitter and a receiver, or more specific knowledge about specific radios and their components. Additionally, the facts are interlinked via properties (relations), thus resulting

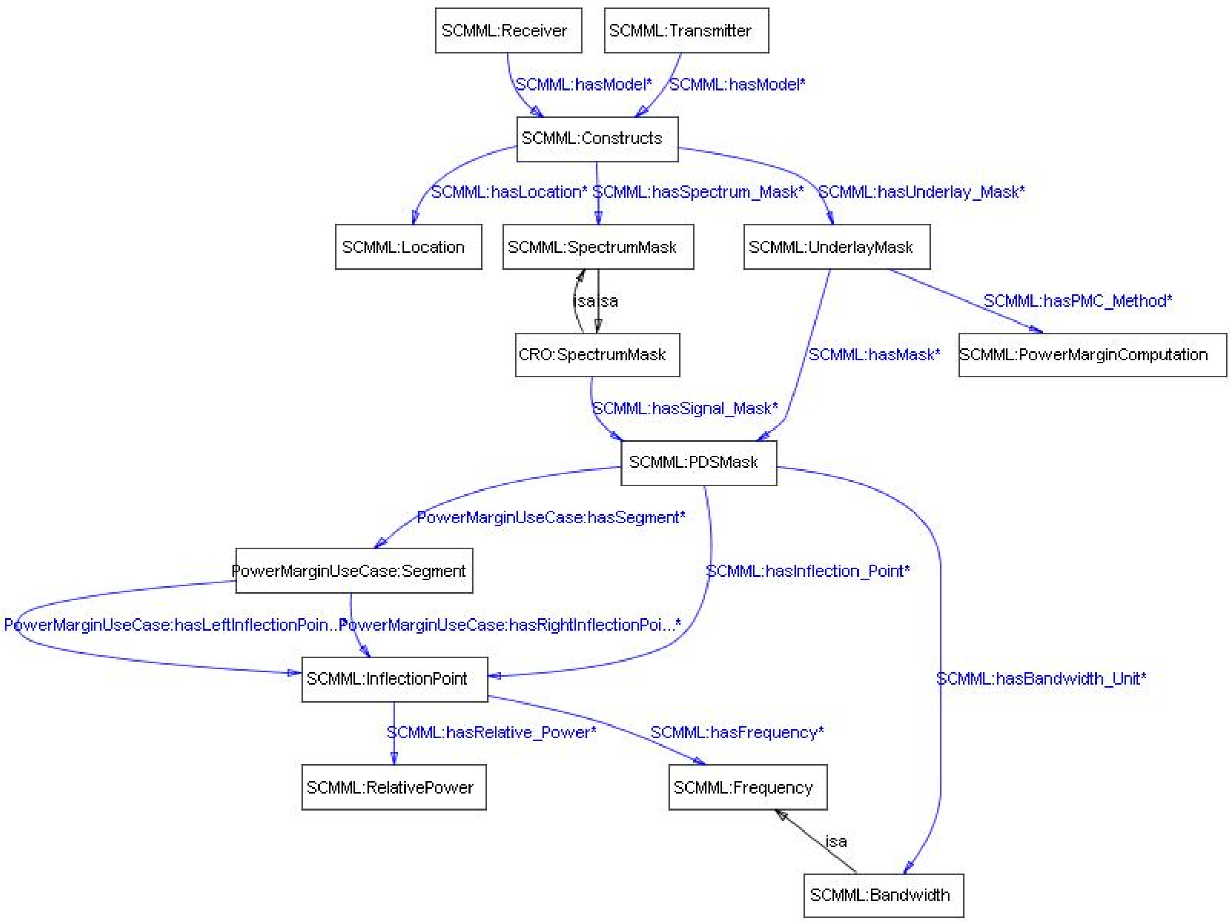

Figure 12: Representation of Power Margin use case in ontology.

into a graph representation. The execution of declarative code is carried out by an inference engine, which is specific to the language, but not to any declarative information represented in either OWL or rules.

In imperative languages the programmer needs to write code that will make use of some information stored, e.g., in databases, which will capture the functionality desired by the user. Thus the user (customer) provides specifications, which then are turned into code by developers. The code development process is rather long and cumbersome. In declarative language the development of such code is either unnecessary or at least limited. I.e., the user needs to formulate *queries*, which then are executed by a query processing engine, which in turn invokes an inference engine. The query engine, again, is associated with the language and not with the information specific to the information being queried (although the queries must be formulated in the form that the query language allows).

The good feature of the declarative programming approach is that once a system is implemented, modifications can be implemented rather easily, since only the declarative knowledge needs to be modified, while the procedural code (the query and inference engines) remain the same. Additionally, the user does not need to know how to invoke specific parts of the knowledge, since this is covered by the functionality of the engines.

The approach presented in this paper is expected to take advantage of these features of declarative programming. Among others, it is expected that the OWL and rules engines will be able to process the large variety of possible SCM models representing various communication scenarios. Additionally, the ontologies that are part of this approach, will serve as vehicles for interoperability of spectrum management systems. Also, the flexibility of the querying process will allow spectrum managers to query the various aspects of the models they exchange, not just a limited set of questions allowed by the GUI of the spectrum management systems.

7. CONCLUSION AND FUTURE WORK

The focus of this paper is on the conversion of SCM models expressed in XML to OWL and rules and using such representations in the process of spectrum management. The main point was to show that such an approach can be implemented and that an inference engine can derive desirable conclusions. This kind of question is important since OWL and its logic-based inference is geared towards discrete facts rather than quantitative operations that are needed for the spectrum management domain. Both of the use cases discussed in this paper required such quantitative operations. Those operations were successfully implemented via procedural attachments that were invoked by the inference engine executing the rules. In this sense, the feasibility of the approach was demonstrated. For both use cases, the inference engine was deriving correct conclusions, some of which were shown in the paper.

The next step in the analysis of the approach proposed in this paper needs to be a deeper evaluation of both the scope of applicability and efficiency of such an ontology-based spectrum policy management. This would require collecting a larger set of various use cases, development of extensions to both the ontology and the rule base required to run the use cases, and collecting metrics. In particular, it is important to focus on the metrics related to the effort needed for the implementation of each use case and the efficiency of the solution in terms of the speed of execution, system requirements and the accuracy of the decisions obtained. Additionally, the approach needs to be compared to a more traditional, imperative language based, implementations of the same use cases.

REFERENCES

[1] FCC, "Report of the spectrum efficiency working group," Federal Communications Commission, Tech. Rep., 2002.

[2] "IEEE Standard for Policy Language Requirements and System Architectures for Dynamic Spectrum Access Systems. IEEE Std 1900.5^{TM}-2011," IEEE Communications Society, 2011.

[3] IEEE, "IEEE DySPAN 1900.5.2 standard," 2015, http://grouper.ieee.org/groups/dyspan/5/index.htm.

[4] J. Stine and S. Schmitz, "Model based spectrum management," The MITRE Corporation, Tech. Rep., 2013.

[5] L. Grande, M. Sherman, H. Zhu, M. M. Kokar, and J. Stine, "IEEE DySPAN 1900.5 Efforts To support Spectrum Access Standardization," in *2013 IEEE Military Communications Conference*. IEEE, 2013, pp. 1750–1755.

[6] W3C, "OWL 2 Web Ontology Language Document Overview," 2009, http://www.w3.org/TR/owl2-overview/.

[7] WinnF, "Description of cognitive radio ontology," Wireless Innovation Forum MLM Working Group, Tech. Rep. WINNF-10-S-0007, 2010, http://groups.winnforum.org/d/do/3370.

[8] C. Matheus, M. M. Kokar, and R. Dionne, "A demonstration of formal policy reasoning using an extended version of BaseVISor," in *IEEE Workshop on Policies for Distributed Systems and Networks, POLICY 2008*, 2008.

[9] E. Prud'hommeaux and A. Seaborne, "Sparql query language for rdf. w3c working draft," 2006, http://www.w3.org/TR/rdf-sparql-query/.

[10] in *The Description Logic Handbook*, F. Baader, D. McGuinness, D. Nardi, and P. Patel-Schneider, Eds. Cambridge University Press, 2003.

[11] "Protege," 2013, Stanford Center for Biomedical Informatics Research: protege.stanford.edu.

[12] "Ontoviz plug-in for Protege," Stanford Center for Biomedical Informatics Research: http://protegewiki.stanford.edu/wiki/OntoViz.