A POLICY REASONER FOR POLICY-BASED DYNAMIC SPECTRUM ACCESS

Behnam Bahrak
(bahrak@vt.edu)

Dept. of Electrical & Computer Engineering
Lab for Advanced Research in Information Assurance & Security (ARIAS)
Virginia Tech

Dec. 3, 2009

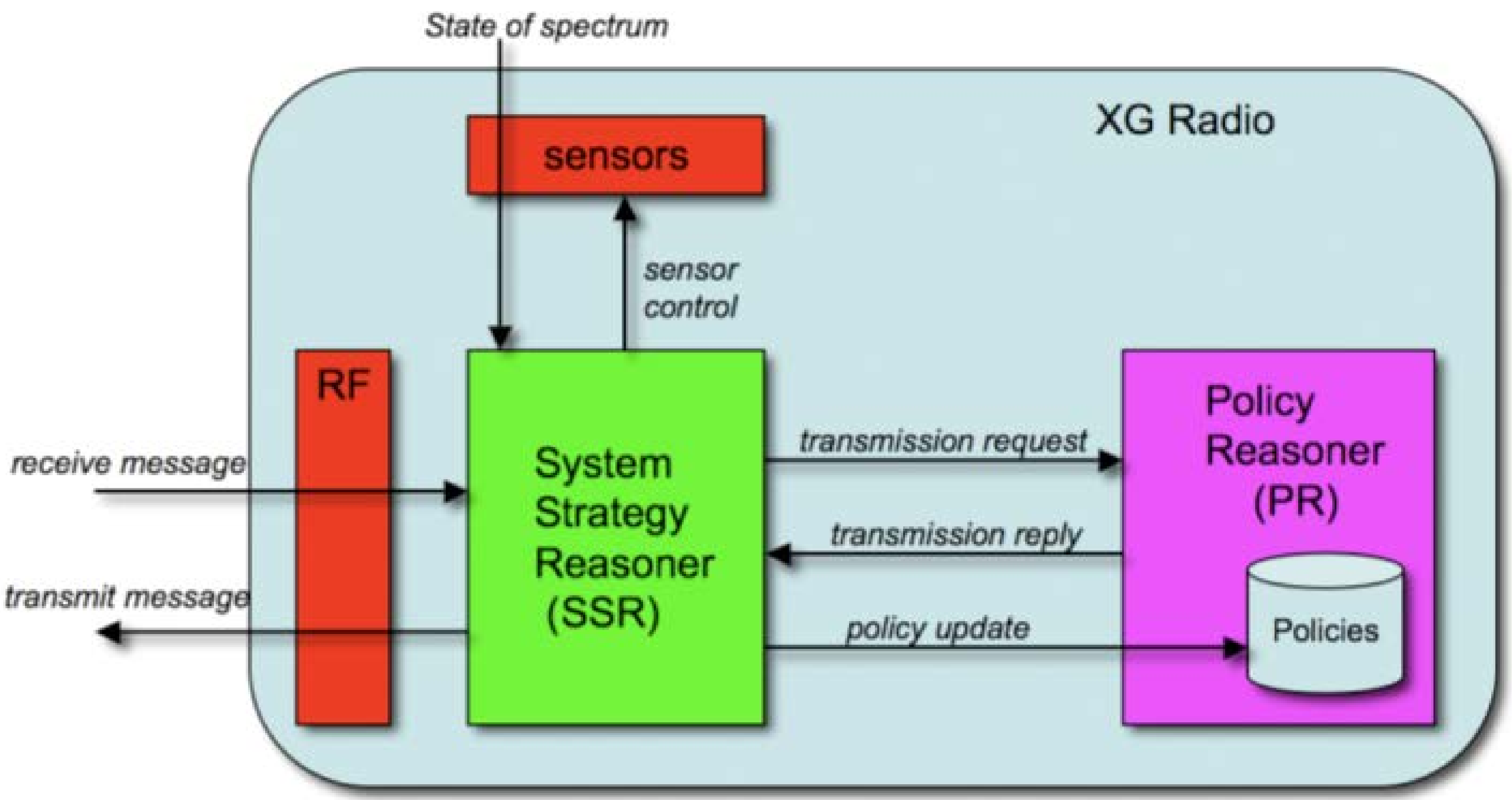
VirginiaTech
Invent the Future

Agenda

- Introduction to XG radio architecture
- Multi Terminal Binary Decision Diagrams
- The structure of the proposed policy reasoner
- Reasoning algorithms
- Conclusion & Future work

XG Radio Architecture (1)

•The neXt Generation (XG) program is a technology development project sponsored by DARPA's Strategic Technology Office.



XG Radio Architecture (2)

- **Sensors**: An XG radio needs sensors to sense its environment and discover the unused spectrum.

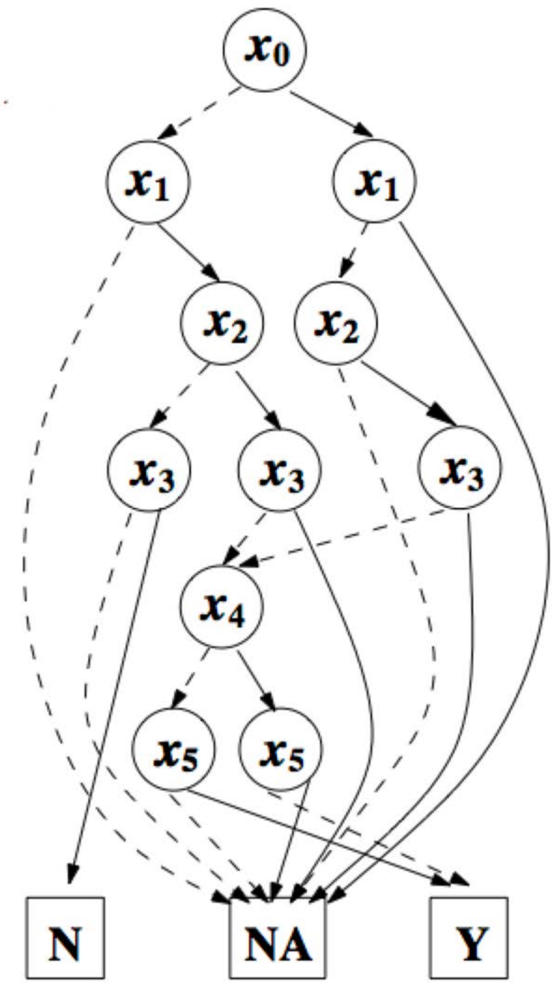
- **Radio Frequency (RF)**: The RF part of an XG radio is used to transmit and receive various signals.

- **System Strategy Reasoner (SSR)**: The SSR controls the radio's transmission. It builds optimal transmission request based on sensor data and its current state.

- **Policy Reasoner (PR)**: The PR receives the transmission requests that the SSR builds and evaluates them using the currently active set of policies, checking them for policy conformance.

Multi Terminal Binary Decision Diagrams

- A Binary Decision Diagram (BDD) is a data structure that is used to represent a Boolean function.
- BDDs are rooted, directed, and acyclic graph, which consists of decision node and terminal nodes called 0-terminal and 1-terminal.
- Each decision node *u* is labeled by a Boolean variable *(var(u))* and has two child nodes called low child *(low(u))* and high child *(high(u)).*
- MTBDD is the same as BDDs with the difference that MTBDDs have more than two terminal nodes

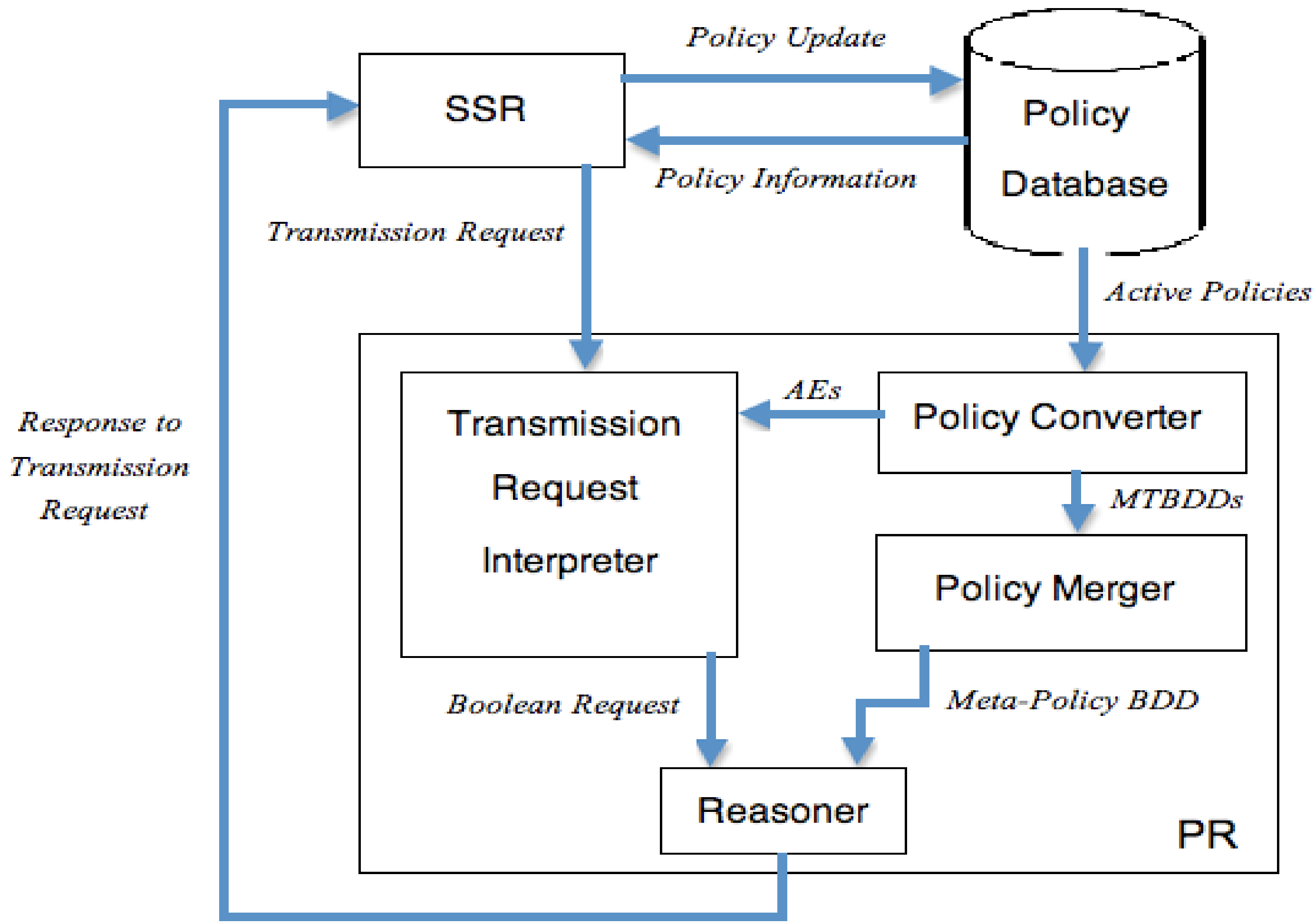


Why MTBDDs?

- We have compared different kind of representation for policies.
- *MTBDDs* have proven to be a simple and efficient representation for XML policies.

	Representation	Policy Combining	Conflict Analysis	Constraint Computing
Alloy	poor scalability/ limited tools	NA	simple	good
LPS	NA	NA	Complicated	NA
PROMELA	poor scalability	NA	NA	NA
RMM	no tools	NA	NA	NA
VDM	poor scalability		NA	NA
DNF/CNF	poor scalability	weak	NA	good
MTBDD	good scalability/ many tools	powerful	simple	good

Strructure of Our Policy Reasoner



Virginia Tech
Invent the Future

Policy Converter (1)

- The Policy Converter converts the XML cognitive radio active policies into MTBDDs.
- A cognitive radio policy in XML format can be transformed into a Boolean expression.
- Using the Boolean encoding, a policy *P* can be transformed into an MTBDD.

Policy Merger

- Policy merger combines all the active policy MTBDDs that policy converter has made to build a single meta-policy MTBDD.
- The policy merger in our policy reasoner uses the *deny-override* rule to resolve conflicts.
- *Deny-override* rule denies a request if the request is denied by any of the currently active policies.

Transmission Request Interpreter

- In order to respond to the transmission request, the Reasoner requires the transmission request to be compatible with the meta-policy MTBDD.

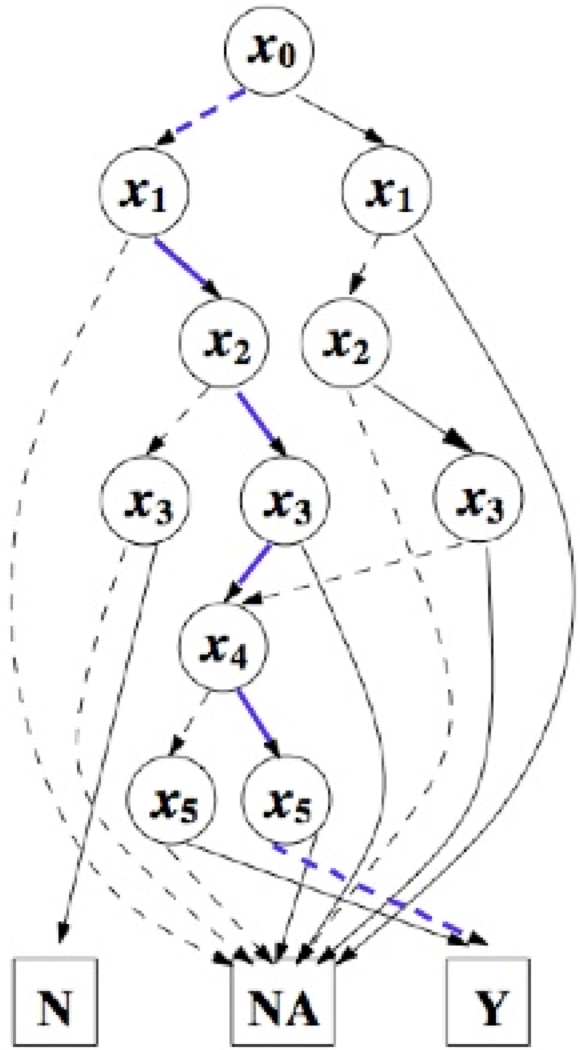
- The transmission request interpreter evaluates the transmission request and assigns a value to each of the Boolean variables of the meta-policy BDD

Reasoner

- **The Reasoner:** The Reasoner sends one of the three possible responses to the SSR:
- **Yes:** The transmission is allowed.
- **Incomplete:** The transmission request is incomplete, but will be allowed if the SSR applies the appropriate opportunity constraints to its transmission request.
- **No:** The transmission is not allowed, but if the SSR applies a set of appropriate changes to its transmission request, the transmission will be allowed.

Reasoning Algorithms (1)

- We simply follow the path indicated by the transmission request.
- If the path ends with 'Y' terminal node, the transmission request is allowed and no additional reasoning is required.



Reasoning Algorithm (2)

- If the transmission request is incomplete, the Tx request interpreter will assign values to AEs that are specified in the transmission request and the reasoner simplifies the meta-policy BDD.

- The reasoner uses **FindPath** algorithm to find an opportunity constraint such that if the SSR satisfies that constraint in its transmission request, the policy reasoner will permit the transmission request.

Reasoning Algorithms (3)

- When the transmission request is denied, the policy reasoner computes the opportunity constraint by using **FindBestPath** algorithm.

- An optimal returned path should have the following properties:

  (1) terminates with the Y terminal node.

  (2) shares the maximum number of edges with the transmission request's path.

Future Work

- We plan to extend the policy reasoner architecture to include additional modules and functionalities.
- A policy preprocessing module that removes invalid paths within the meta-policy BDD
- We also plan to devise an algorithm for computing opportunity constraints that can process *weighted* transmission request parameters.
- The implementation of our policy reasoner is ongoing and we plan to carry out a quantitative analysis of the policy reasoner.

Conclusion

- We described the architecture of a new policy reasoner.
- Our device-independent policy reasoner processes policies represented as MTBDDs.
- It detects and resolves policy conflicts.
- The proposed policy reasoner also has the ability to process underspecified or invalid transmission requests.
- We have contrived two new algorithms for computing the opportunity constraints.