Harrold, T.J., Faris, P.C., & Beach, M.A. (2008). Distributed spectrum detection algorithms for cognitive radio. In *The IET Seminar on Cognitive Radio and Software Defined Radios: Technologies and Techniques* Institution of Engineering and Technology (IET).
<http://hdl.handle.net/1983/1252>

Peer reviewed version

[Link to publication record in Explore Bristol Research](#)
PDF-document

University of Bristol - Explore Bristol Research

General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available:
<http://www.bristol.ac.uk/red/research-policy/pure/user-guides/ebr-terms/>

Distributed Spectrum Detection Algorithms for Cognitive Radio

T.J. Harrold, P.C. Faris and M.A. Beach

Thursday 18th September 2008

Background and Introduction

- For Cognitive Radio networks
 - There is a need to identify the White Space in the spectrum
 - The capabilities of the terminals could allow them to perform spectrum sensing to achieve this
 - The sensing task could be implemented more accurately/efficiently by teaming of terminals
- This technique introduced in this work
 - Decides whether a single channel is occupied
 - Shares results between CR nodes to improve performance

Background and Introduction

Spectrum Occupancy and White Space

Spectrum Sensing Challenges

- Maintaining an up to date picture of spectrum occupancy is difficult
 - Transmitters may be agile
 - Path loss may suffer temporal changes between transmitter and sensor
 - Transmitter may be temporarily hidden due to shadowing
- Getting it wrong
 - False detection: lost re-use opportunity
 - Missed transmission: potential interference

Distributed Sensing

Distributed Sensing

$$Q = [X_1 \quad \dots \quad X_N] \begin{bmatrix} D_1 \\ \vdots \\ D_N \end{bmatrix} + [Y_1 \quad \dots \quad Y_M] \begin{bmatrix} T_1 \\ \vdots \\ T_M \end{bmatrix} + S \times Z$$

M	Number of previous results/time-steps included
N	Number of neighbouring nodes included
X_n	Sensing result from neighbouring node n [+1,-1] Eg. $D_i = 1000/d_i$, $D_i \leq 1$
D_n	Weighting factor according to distance applied to neighbouring node n
Y_m	Result from m time-steps previously
T_m	Weighting applied to previous result m
S	Weighting applied to the node's own result
Z	Node's own result [+1,-1]
Q	Final result [positive, negative]

Weighted Algorithm

- Various Trade-offs exist
 - How many neighbour nodes to include?
 - How to weight the importance of neighbour nodes' decisions?
 - How to weight the importance of the own node's decision
 - How to weight historic results
- Factors
 - Extra control traffic required
 - Accuracy of results and false-positives

Simulations

- MATLAB simulations to test algorithm performance
 - Sensor nodes deployed randomly
 - Aim is to test the probability of detecting a transmission
 - Compare single node vs distributed algorithm
- 5 Different transmitter types to detect
 - Distinguished by transmitter power
- 3 scenarios
 - Simulation area
 - Path loss exponent
 - Shadowing variance

Simulations

Performance – Single Node Sensing

Performance – With Distributed Detection

Sensitivity – Node Density

- Dense Urban Scenario – 1km²

Sensitivity – Number of Neighbours

Conclusions

- Sharing of single channel sensing information
 - Can greatly improve detection accuracy
 - >99% accuracy has been shown in these simulations
 - Not so good for lower power transmissions in highly shadowed propagation environments
- Applications and further work
 - Allocation of sensing task for multiple channels
 - WiMAX bandsharing with swept radar
 - Real-time updating
 - Spectrum Access

- Thank you for your attention