Combining Cognitive Radio and Software Radio Approach for Low Complexity Receiver Architecture

27. February 2007

Edmund Coersmeier

NOKIA
Connecting People

Overview

- Motivation
- Cognitive Radio Enhancement
- Software Radio Complexity
- Conclusion

Motivation

Connecting People

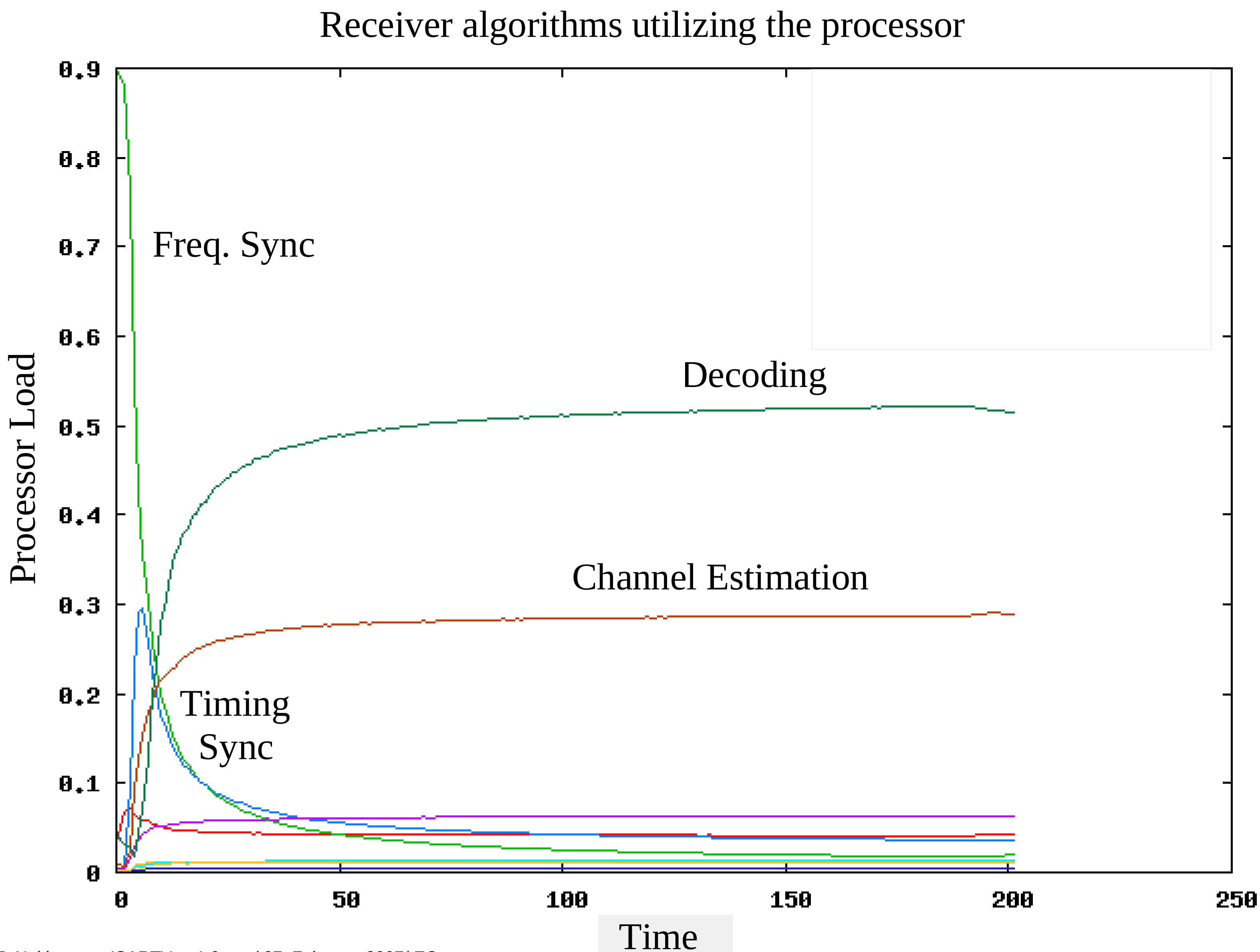
Requirement when combining Cognitive and Software Radio

- Cognitive Radio for spectrum efficiency
  - analyzing user application
  - definition of wireless requirements
  - spectrum scanning
  - definition of radio characteristics

- Software Radio
  - adjusts transmitter and receiver algorithms
  - transforms algorithms to an applicable architecture
  - maps the architecture on available processor platform
  - balances between different, parallel operating radios

- To achieve efficient receiver implementations Software Radio requires
  - strong flexibility in terms of
    - algorithm complexity
    - power consumption
  - *support from Cognitive Radio*

Processor load as measure for algorithms complexity



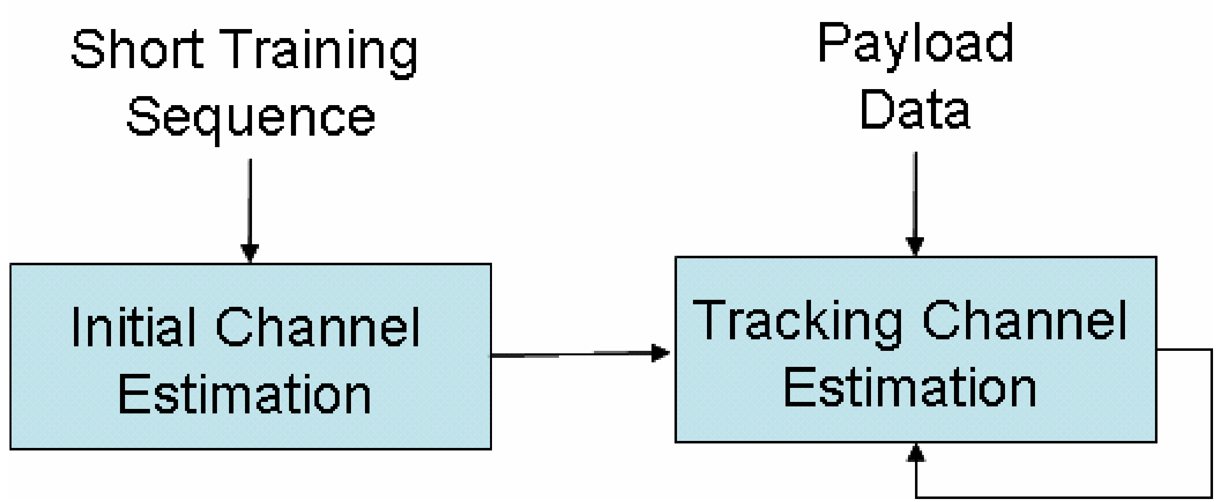
Cognitive Radio Enhancement

Channel-State Estimation

- Channel-State Estimate to judge about channel capacity

- Semi-blind training
  - Supervised training mode via short training sequence
  - Tracking via data feedback

- Rate feedback to transmitter to setup
  - data rate
  - transmit-power control



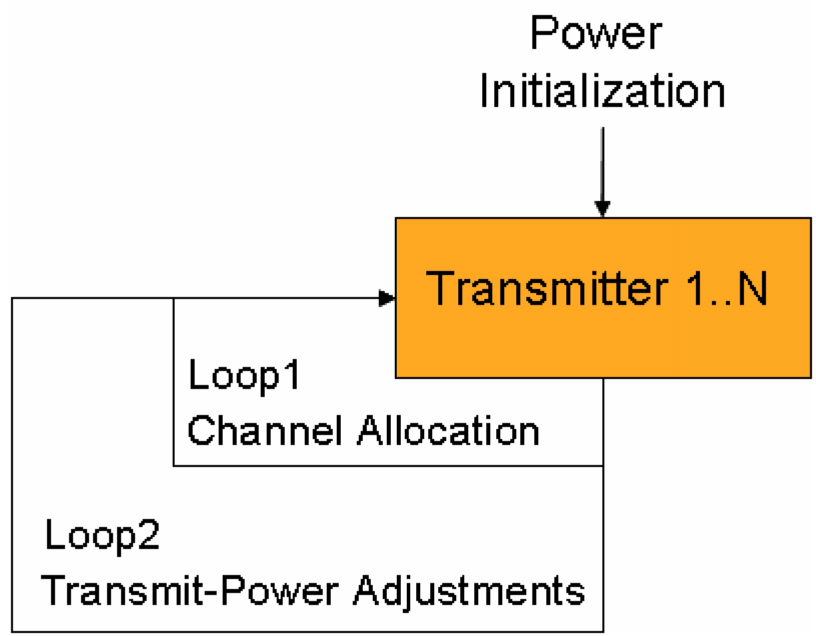
Transmit Power Control

- Power initialization

- Inner Loop
  - Allocation of a number of channels

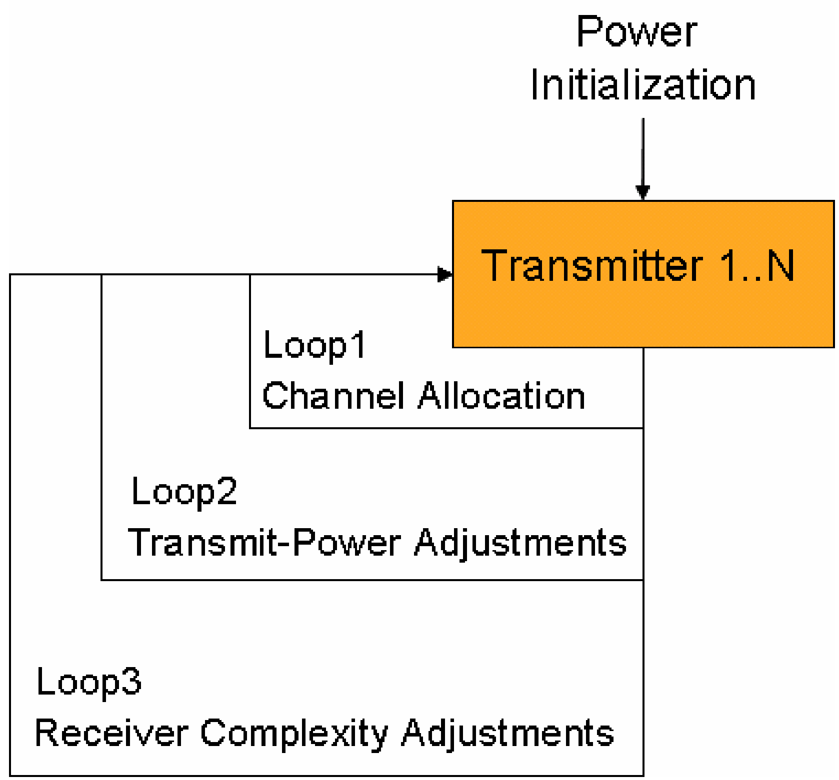
- Outer Loop investigates achieved data rate
  - exceeding
  - matching
  - undershooting

- Outer Loop adjusts the transmit power of each transmitter
  - All transmitters run from data-rate perspective with optimal transmit power
  - *What is about the receiver complexity?*



Cognitive Radio Enhancement

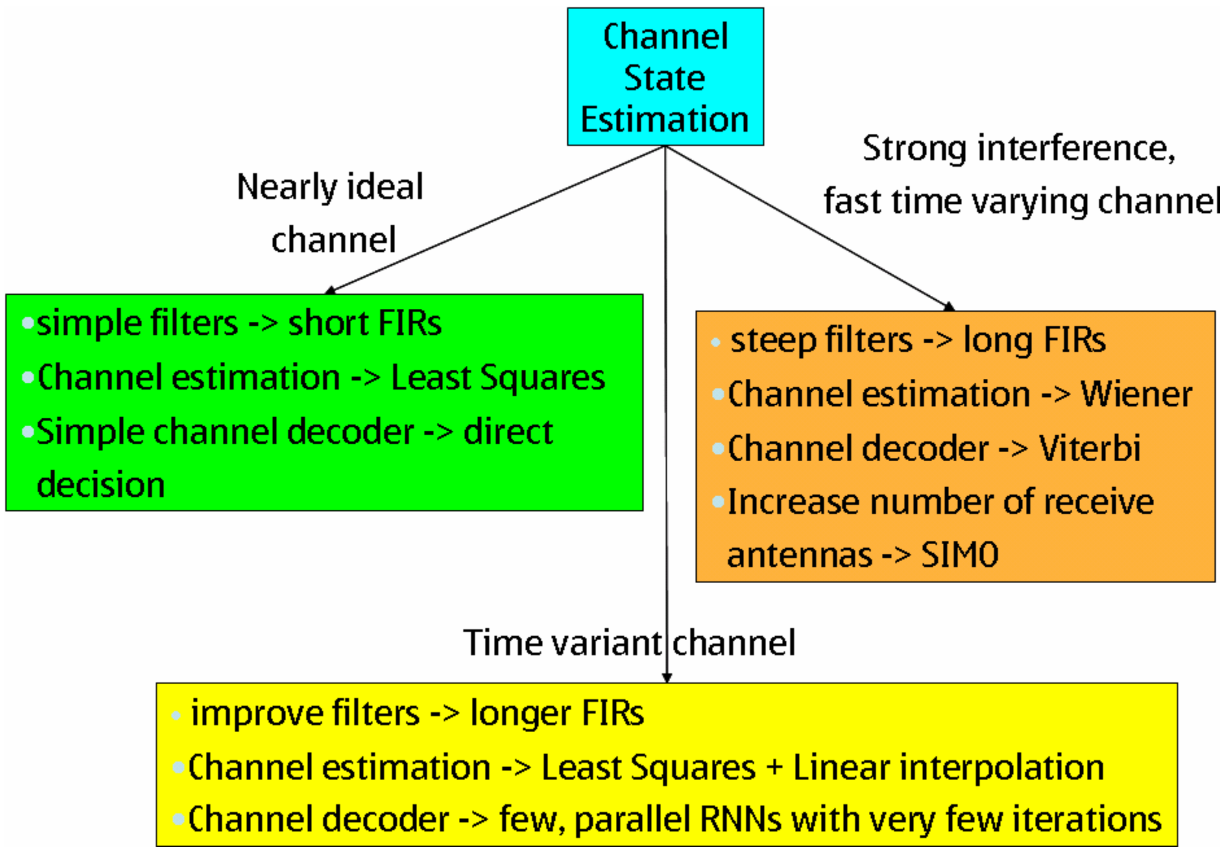
- Each receiver includes an option to ask for low receiver complexity
  - Transmit-power increase
  - High quality channel selection
- Transmit-power increase
  - Other transmitters reduce power
  - Other receivers increase complexity
- High quality channel selection
  - Find a better fitting free channel
  - Exchange already allocated channels



Software Radio Complexity

Receiver Algorithms with different Complexities

- Different receiver complexities based on channel-state estimation
- Receiver complexities can change at any time



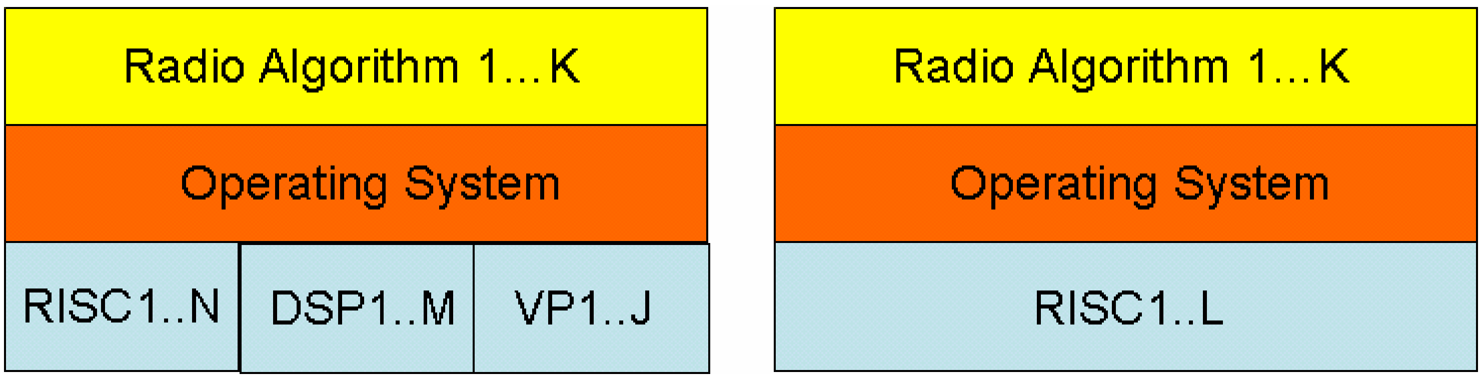
NOKIA Connecting People

Receiver Scaling through Multi-Processor Platforms

- Pure hardware-optimized design can be replaced by multi-processor platform
- Several radios and their algorithms run in parallel

For parallel programming a significant change of

mathematics for radio algorithms might be required



Example of Parallel Radio Algorithms – Channel Decoding

- Viterbi
  - high signal processing performance
  - optimal for hardware implementation
  - sub-optimal for software radio approach
  - difficult to parallelize

- Recurrent Neural Networks
  - do not outperform Viterbi signal processing performance
  - similar mathematics as adaptive filters
  - easy to parallelize several networks

$$\min_{\mathbf{w}(n)} \left\| e(n) \right\|_2^2 = \min_{\mathbf{w}(n)} \left\| \mathbf{X}(n)\mathbf{w}(n) - \tilde{y}(n) \right\|_2^2$$

Adaptive Filter

$$\min_{\mathbf{c}} \left\| \mathbf{e} \right\|_2^2 = \min_{\mathbf{c}} \left\| \mathbf{r} - \mathbf{c} \right\|_2^2 = \min_{\mathbf{a}} \left\| \mathbf{r} - \mathbf{G}^T \mathbf{a} \right\|_2^2$$

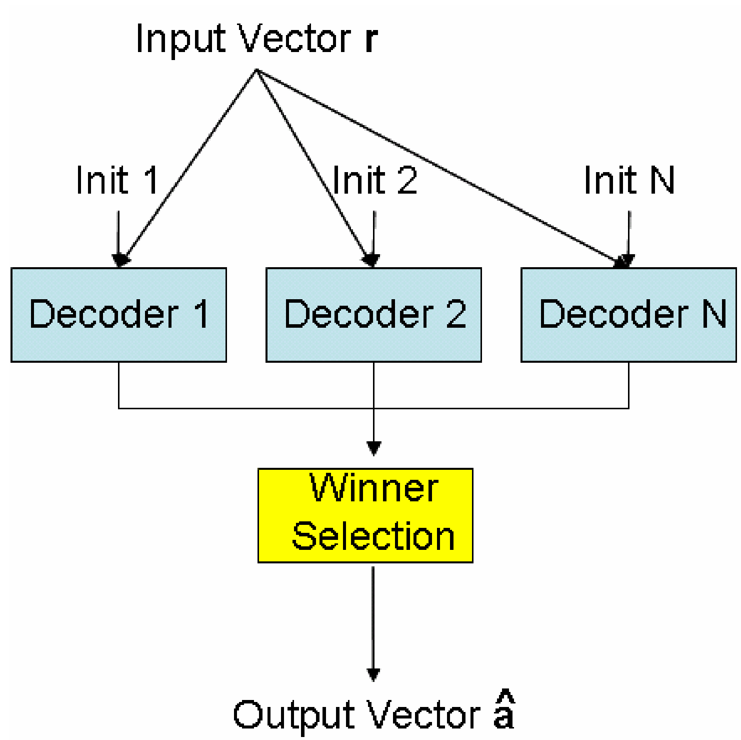
Recurrent Neural Network

NOKIA
Connecting People

N Parallel Channel Decoders

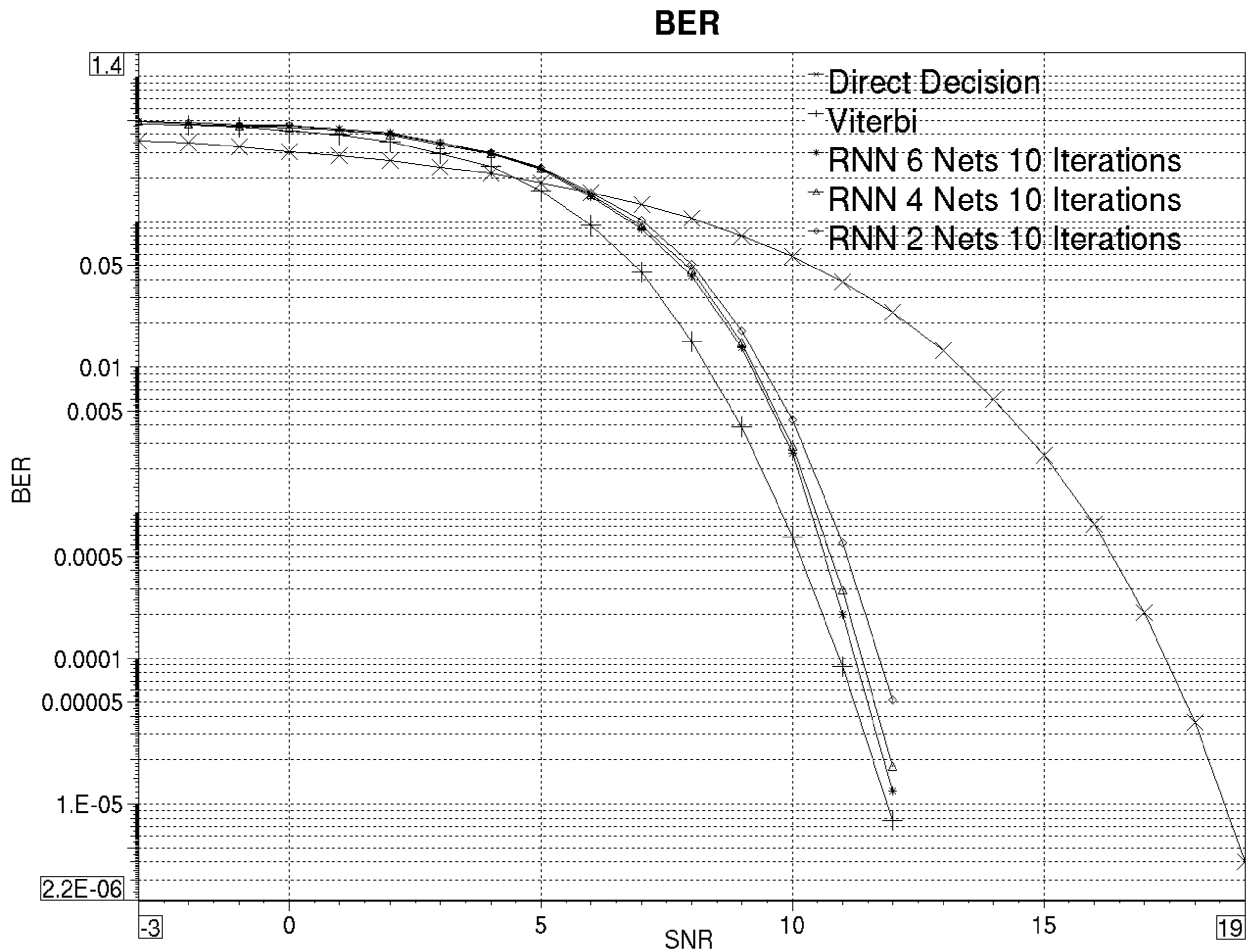
- Run several networks in parallel
- The more networks, the higher the channel decoding performance

- Research topic – optimize complexity of each channel decoder network



Simulation Results

- Number of RNN networks can be adjusted to channel quality



Connecting People

Conclusion

- Software Radio needs to handle several radios in parallel

- Cognitive Radio should support receiver complexity reduction

- Parallel processor platform should be able to replace optimized hardware

- *Mathematics for radio algorithms might change to enable parallel programming*

