

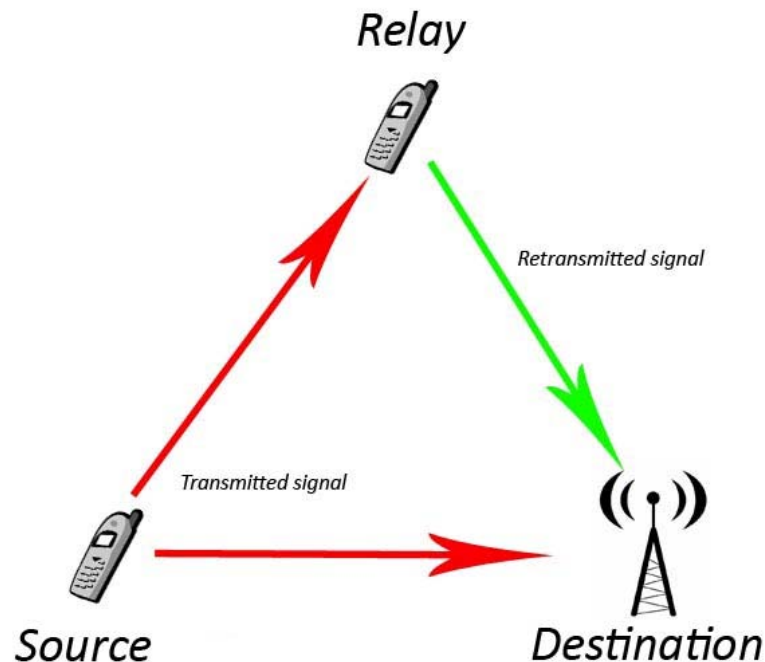
Cooperative Relaying (Matlab Prototype)

Green Group

2008-04-30

Project Description

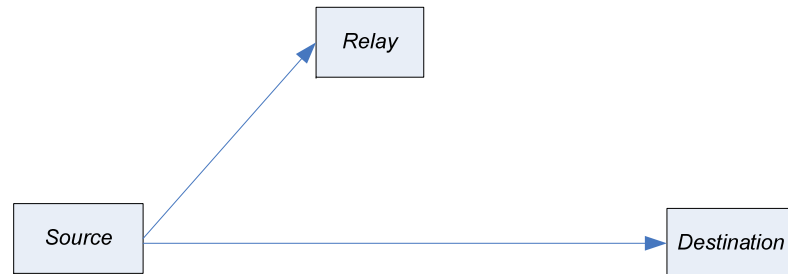
- We will implement the 3-node communication system including Source, Relay, and Destination.



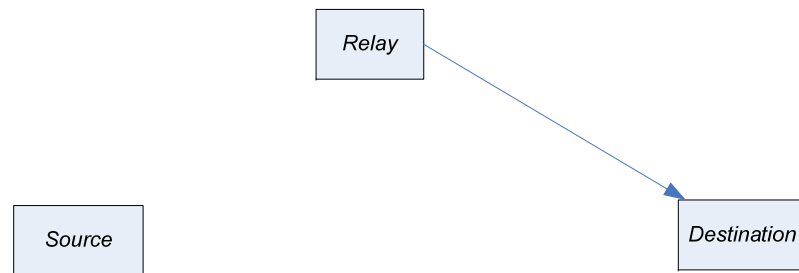
Project Description

- Half duplex, Orthogonal System

- First time slot



- Next time slot



Project Description

- Three different types of relaying protocols
 - Amplify-and-forward
 - Decode-and-forward
 - Detect-and-forward

Matlab Prototype

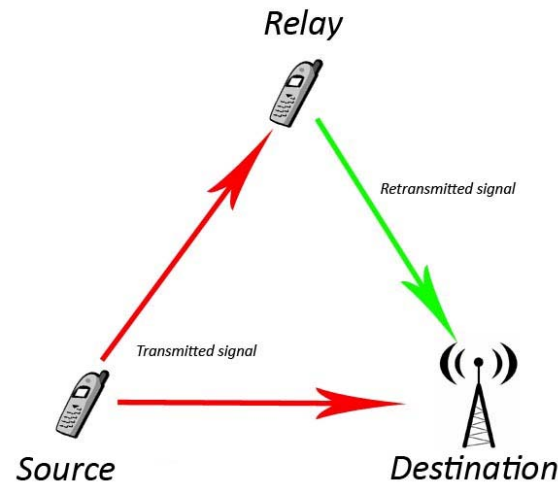
- The Matlab prototype in our project consists of two parts:
- First, the Matlab simulation of the three-node communication system
- Second, the Matlab simulation of the SISO system using fading channel on DSP

Matlab Prototype

- We will provide results with the BER performance, and some analyses of those results.
- Also a comparison of BER performance of 3 different relaying protocols will be provided.

3-node communication system

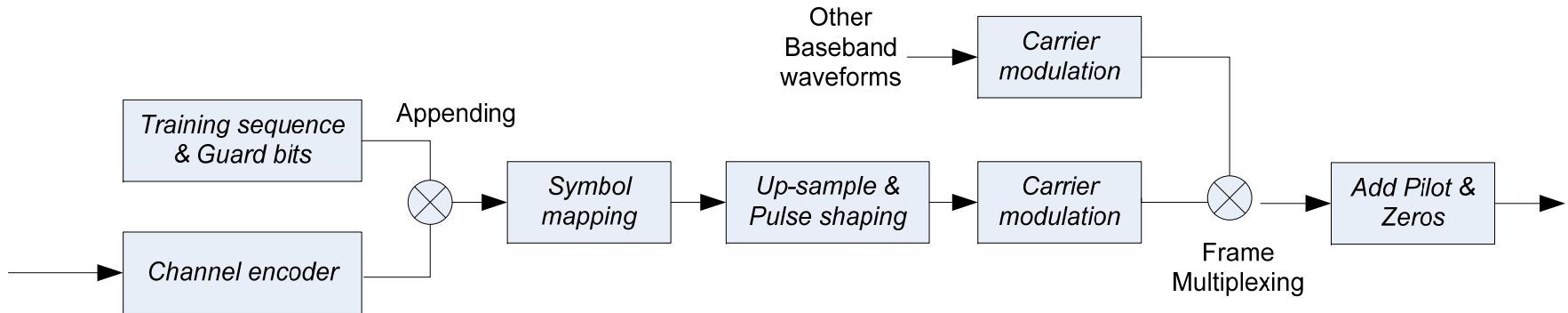
- The basic idea is based on 2-node communication system (SISO system).
- “Relay” works cooperatively with Source.
- → Spatial Diversity



SISO system



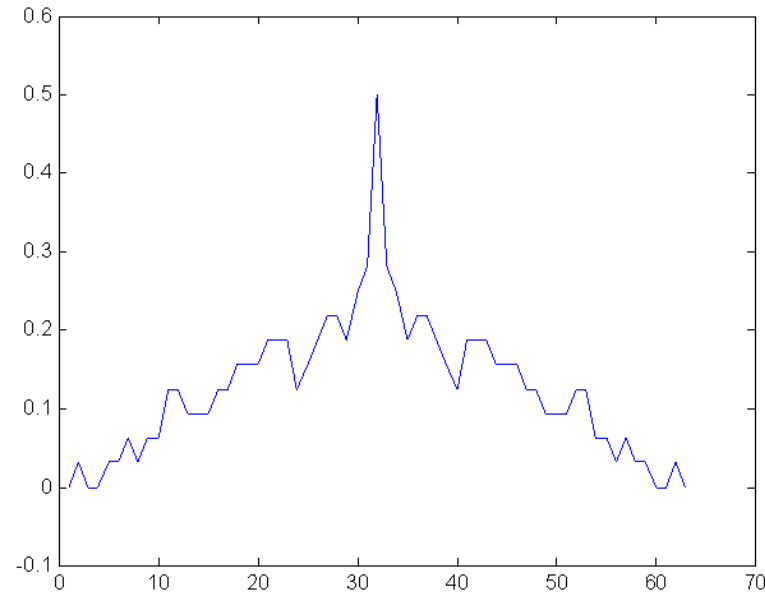
Transmitter



- M-sequence for training bits
- Convolutional code
- QPSK
- Upsampling by factor 8
- Raised cosine pulse shape with rolling factor 0.22

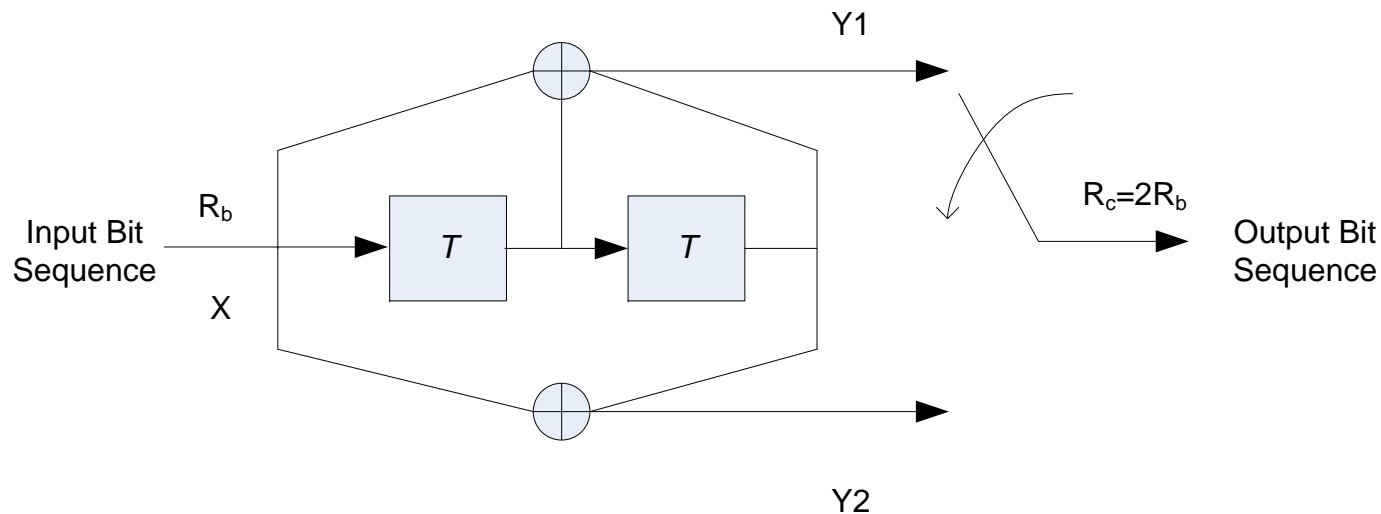
Transmitter

- Training sequence with a good ACF property

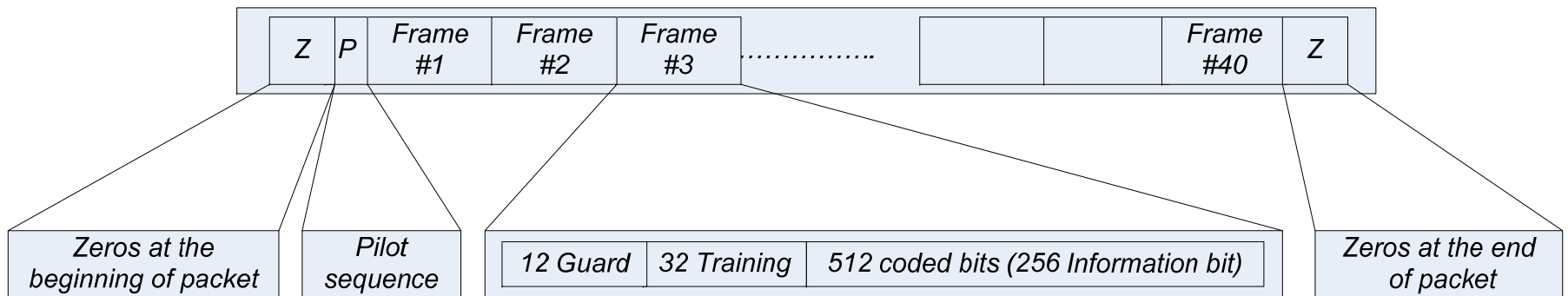


Transmitter

- 1/2 rate convolution channel encoding

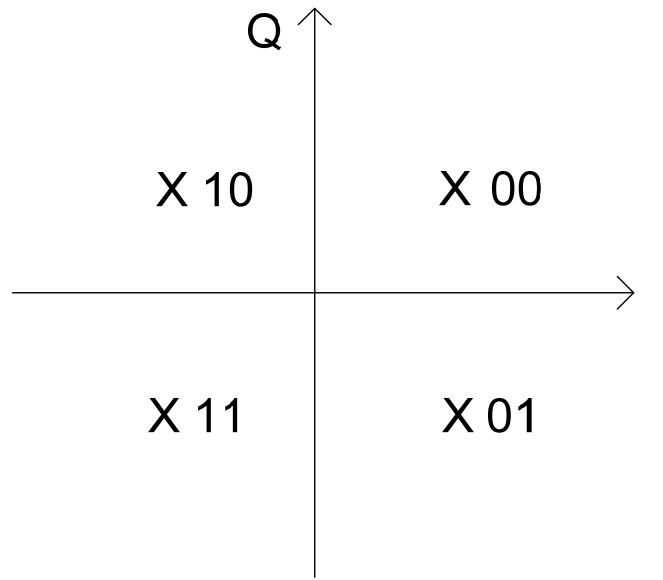


Frame structure



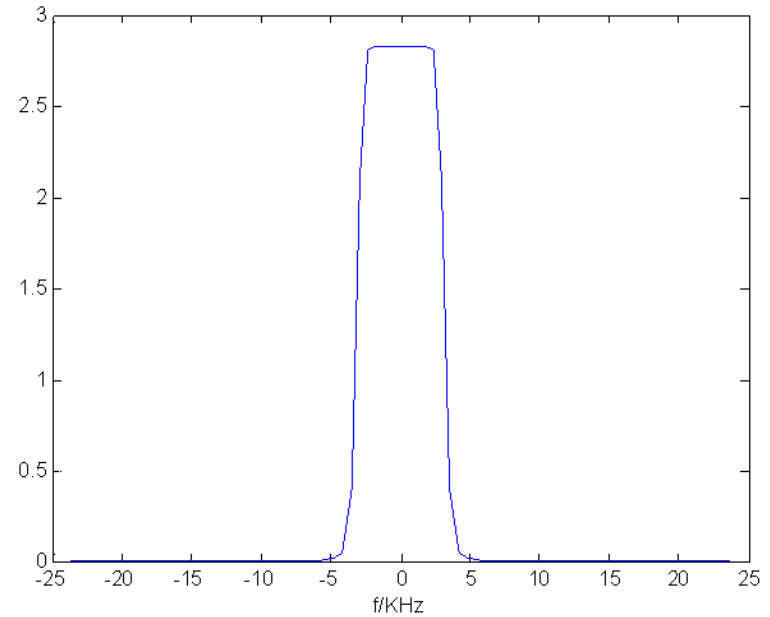
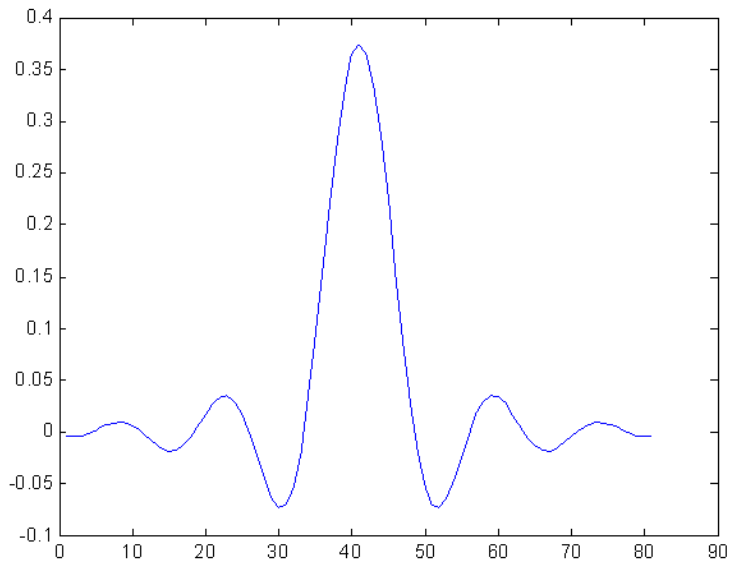
Transmitter

- Symbol mapping :
QPSK with gray mapping



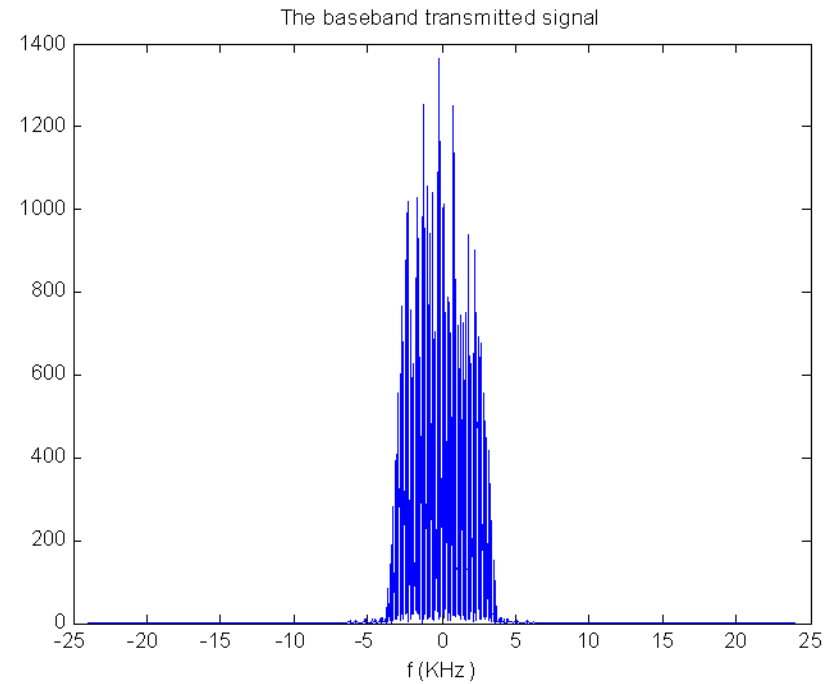
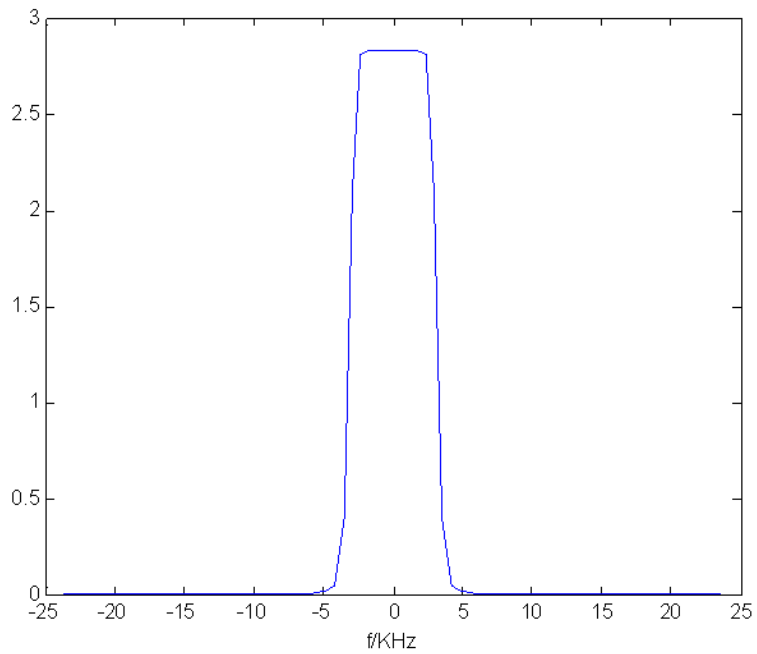
Transmitter

- Pulse shaping



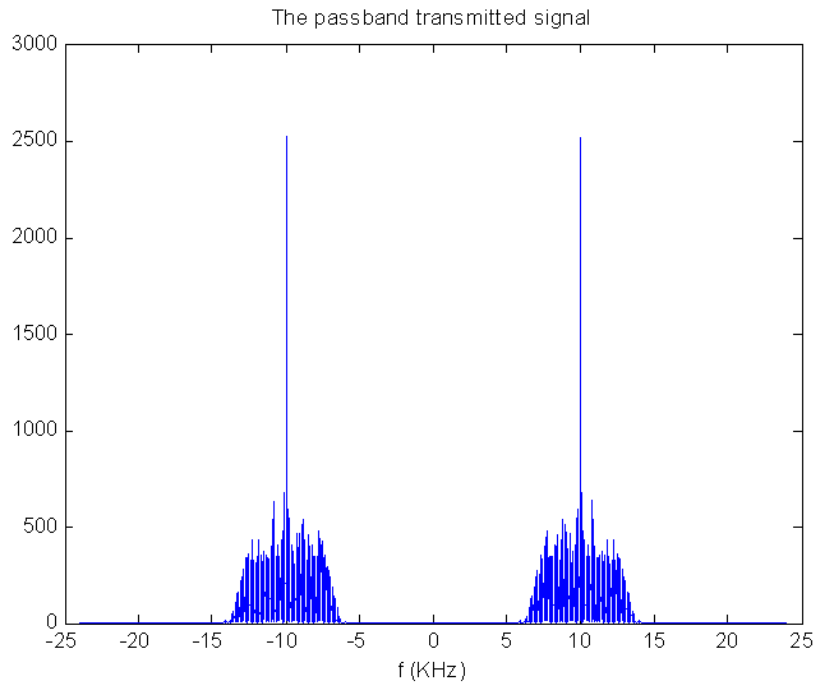
Transmitter

- Pulse shaping



Transmitter

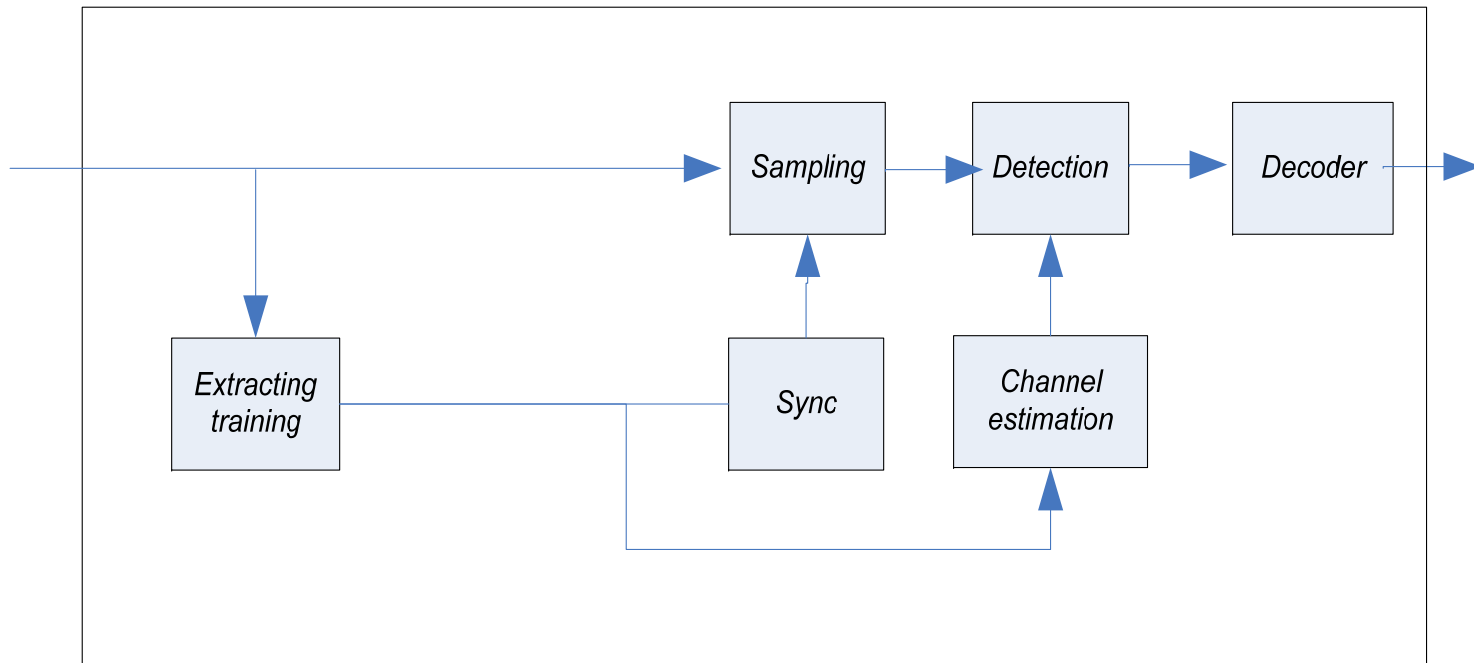
- Upconversion



Channel

- Baseband Rayleigh fading channel with Doppler frequency shift 0.5 Hz by using Jake's model

Receiver



Receiver

- **Synchronization**

Cross-correlation between the received signal and a locally generated time-shifted replica of the training sequence.

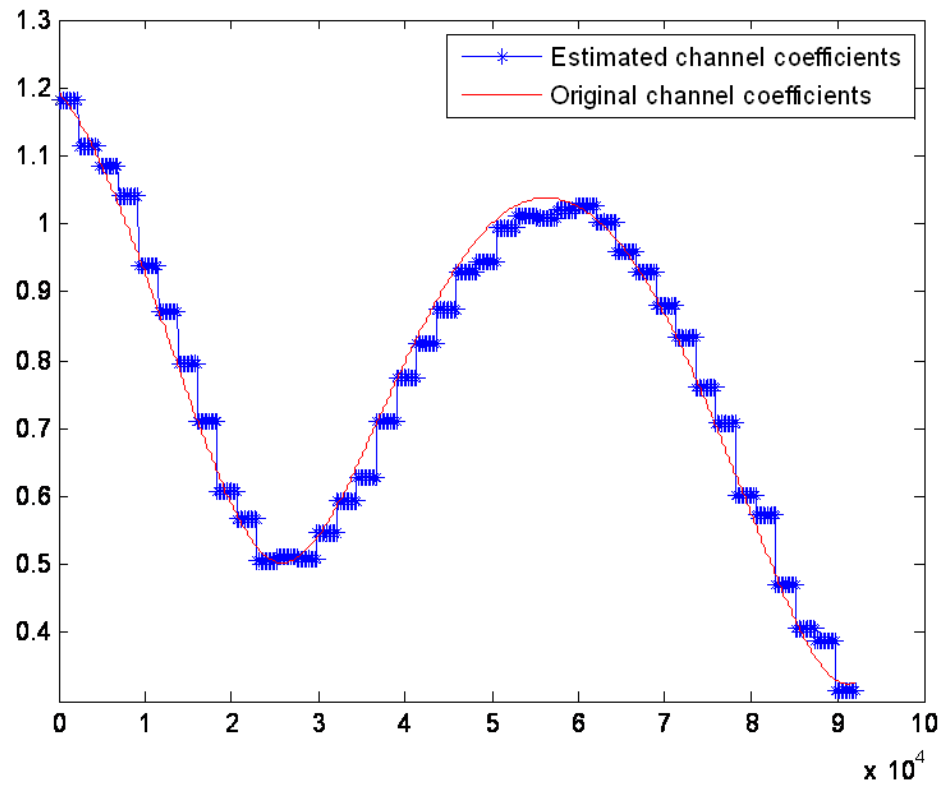
Receiver

- **Channel estimation**

Calculated based on Least-square criteria by using the received signal during the training sequence

Receiver

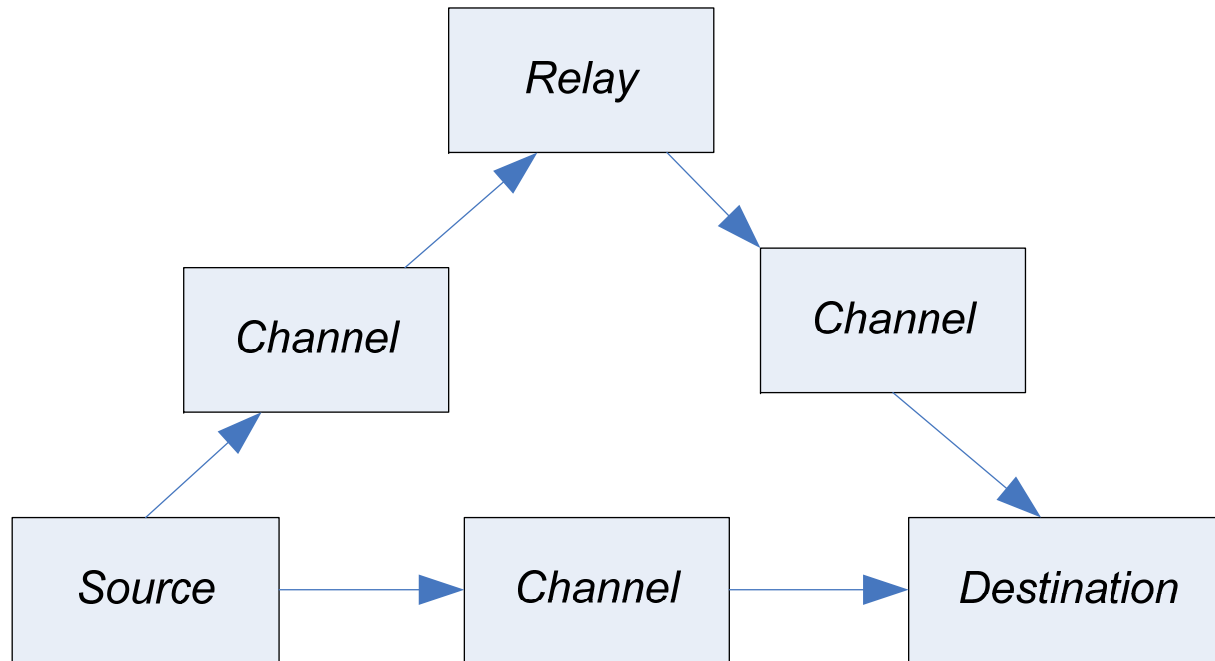
- Channel estimation



Receiver

- **Three kinds of detectors**
- **ZF (Zero Forcing)**
- **MMSE (Minimum Mean Square Error)**
- **ML (Maximum Likelihood)**

3-node communication system



3-node communication system

- **Relaying protocols**

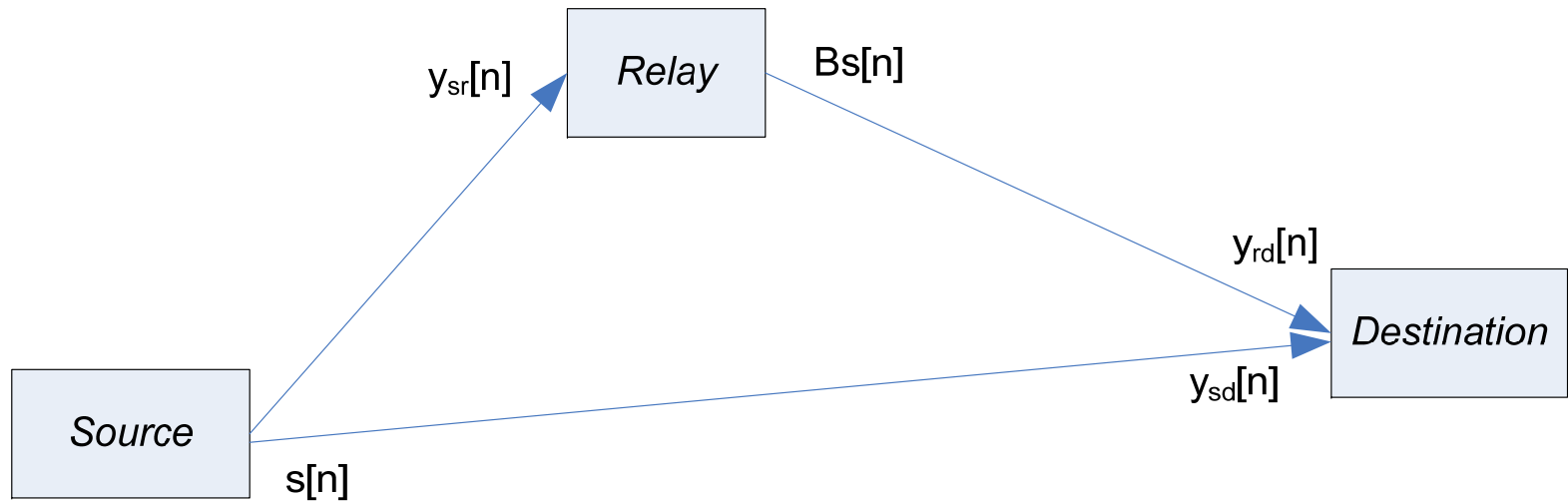
Amplify-and-forward

Decode-and-forward

Detect-and-forward

3-node communication system

Amplify-and-forward

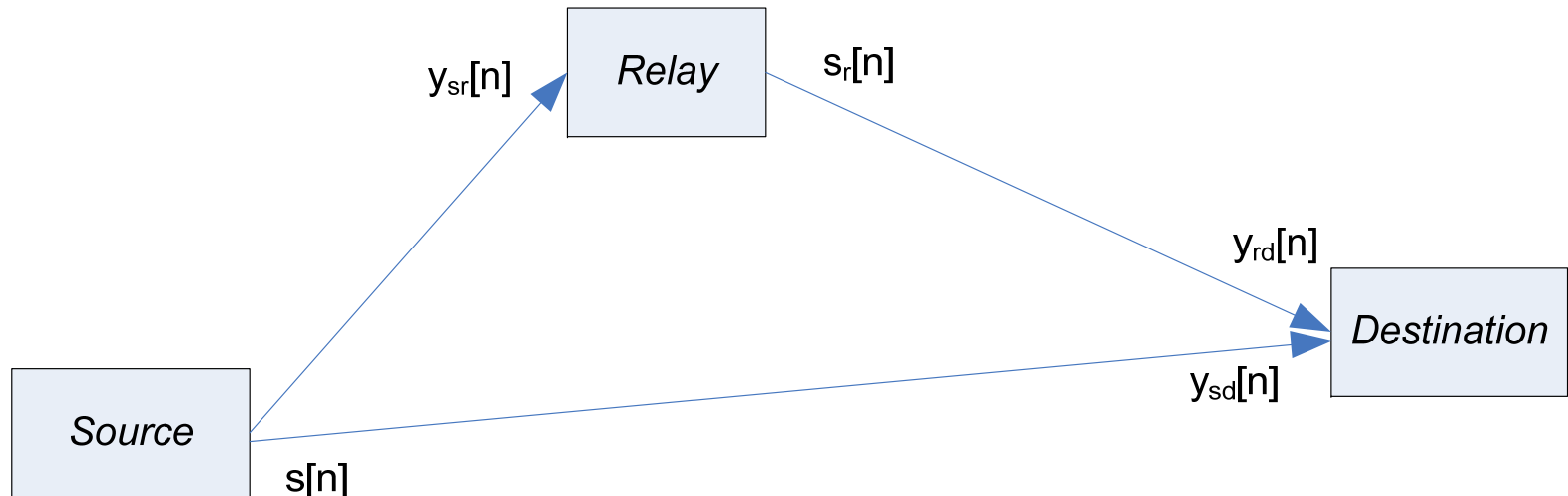


$$\beta \leq \sqrt{\frac{P}{|h_{sr}|^2 P + N_0}}$$

3-node communication system

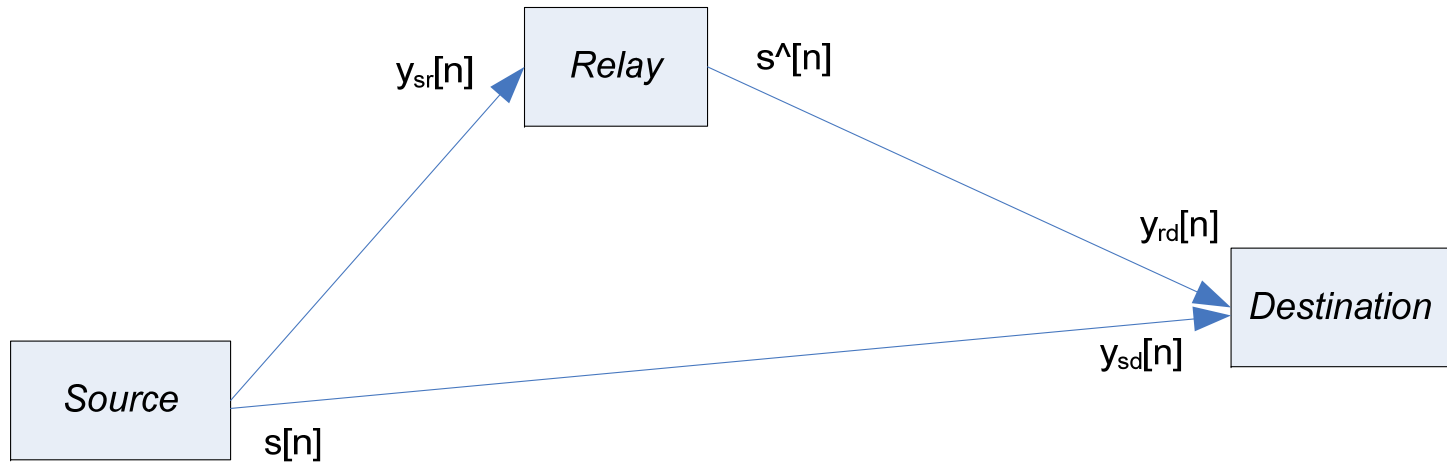
Decode-and-forward

- we decode the received signals at relay and transmit the fresh codeword using the same code to destination.



3-node communication system

Detect-and-forward



3-node communication system

- **Combining**
- *Maximum Ratio Combining (MRC)*
- MRC is used for combining in the decode-and-forward in our simulation.

3-node communication system

- **Combining**
- *Improved Maximum Ratio Combining (MRC)*
- This method is used in amplify-and-forward relay.

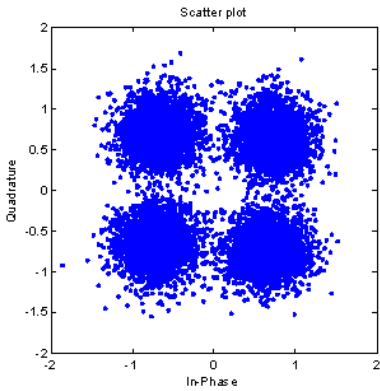
$$g_1 = h_{sd}^*, \quad g_2 = \frac{h_{rd}^* \beta^* h_{sr}^*}{|h_{rd}|^2 |\beta|^2 + 1}$$

3-node communication system

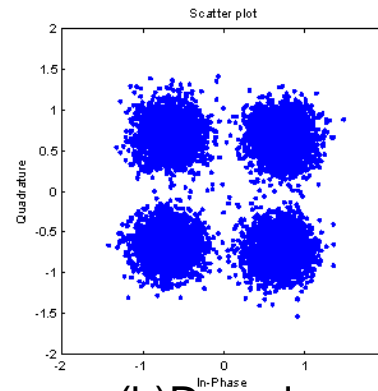
- **Combining**
- *Maximum Ratio Combining (MRC) and*
- *Sub-optimal Combining (Linear MMSE)*
- *Used for detect-and-forward*

Simulation results in Matlab

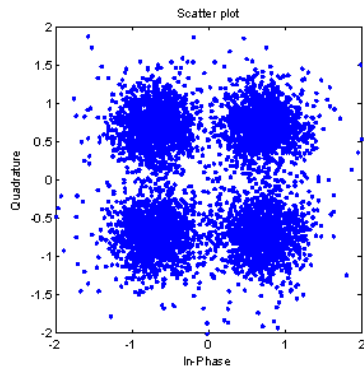
- Signal Constellation



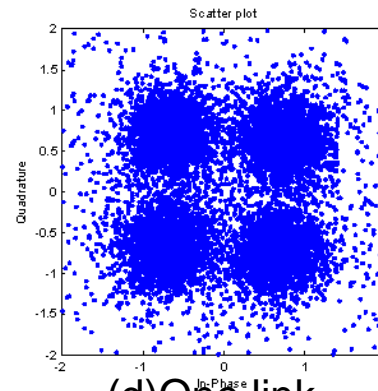
- (a) Amplify-and-forward



- (b) Decode-and-forward



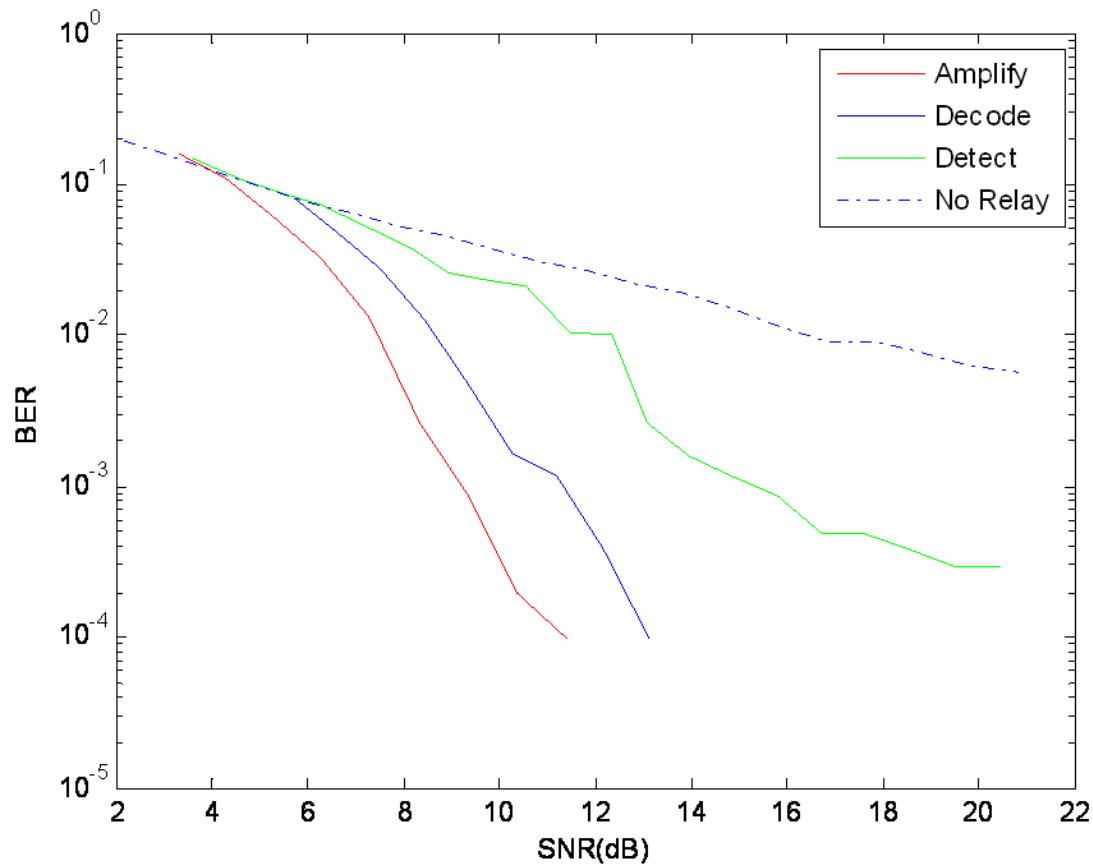
- (c) Detect-and-forward



- (d) One link

Simulation results in Matlab

- BER comparison

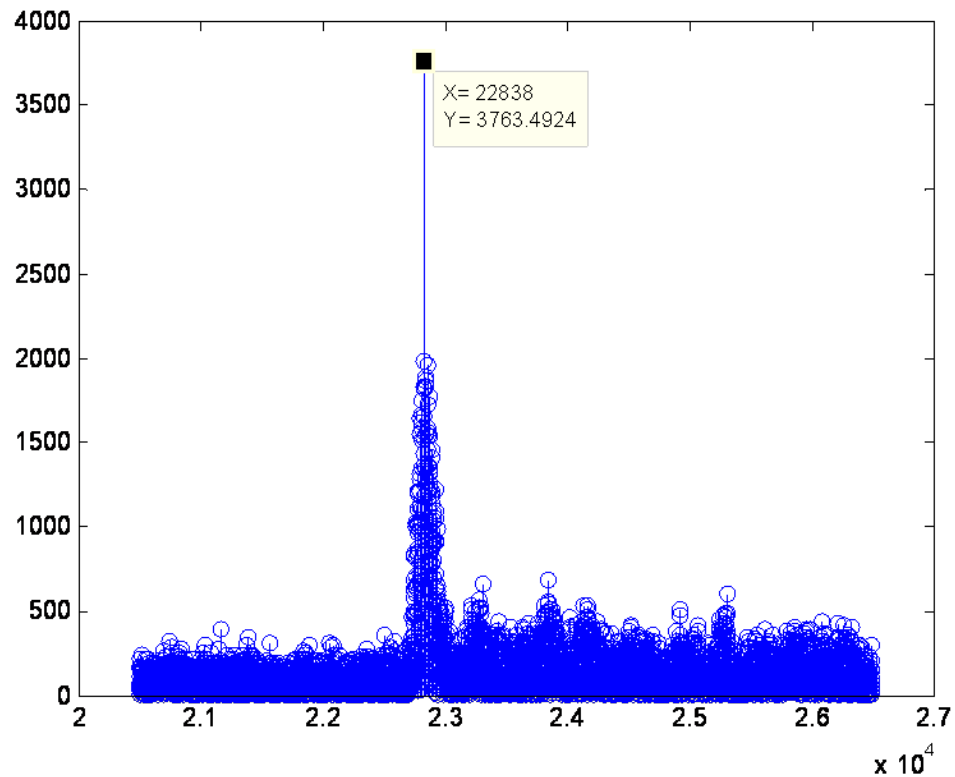


SISO Simulation on DSP Board

- We simulate the SISO system using the channel enumerated on DSP board.
- The system works in the way that the receiver always keeps listening to the incoming signal and measure the variance of it.
- The receiver will detect the signal by using the noise variance threshold.

SISO Simulation on DSP Board

- Synchronization



SISO Simulation on DSP Board

- BER performance

