GSoc' 14 : MIMO Transceiver

Personal Details:
Name: Kunal Sham Sankhe
Email: kunalsankhe.spit@gmail.com
Phone Number : +918886044070/+919028218736
College Name: IIIT Hyderabad
Open source project: GNU Radio
Potential Mentor: Michael Dickens

Title : MIMO Transceiver

Abstract :

The objective of this project is to build GNU Radio Companion(GRC) blocks of MIMO which is a key technology for modern wireless communication standards such as IEEE 802.11n (Wi-Fi), 4G, 3GPP Long Term Evolution, WiMAX and HSPA+. Various GNU Radio blocks will be developed to exploit spatial multiplexing, antenna diversity and beamforming techniques using MIMO. Channel estimator block using Least-square estimation will be used to determine channel state matrix. Spatial multiplexing will be implemented by developing GRC blocks of Spatial stream generator at transmitter and Zero Forcing detector at receiver. GRC blocks for Space Time Block Code(STBC) like Alamouti Code(transmit diversity) and Maximum Ratio Combining(receive diversity) will be implemented to improve reliability of wireless link. Signal to noise ratio (SNR) or Signal to Interference plus noise ratio (SINR) will be improved by implementing MIMO SVD(Singular Value Decomposition) beamforming block for the transmitter and receiver.

Project Details :

Channel State Information (CSI) :

Goal : To build GRC block of Least-square channel estimator.

In multiantenna systems, to achieve reliable communication with high data rates, knowledge of channel state is extremely important. Channel estimation can be done by multiple ways. But a popular approach is so called training sequence (or pilot sequence), where a known signal is transmitted and the channel matrix is estimated using the combined knowledge of the transmitted and received signal.
Let the training sequence be denoted $\mathbf{p}_1, \ldots, \mathbf{p}_N$, where the vector $\mathbf{p}_i$ is transmitted over the channel as [1]

$$\mathbf{y}_i = \mathbf{H}\mathbf{p}_i + \mathbf{n}_i .$$

By combining the received training signals $\mathbf{y}_i$ for $i = 1, \ldots, N$, the total training signalling becomes

$$\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N] = \mathbf{HP} + \mathbf{N}$$

with the training matrix $\mathbf{P}=[\mathbf{p}_1,\ldots,\mathbf{p}_N]$ and the noise matrix $\mathbf{N}=[\mathbf{n}_1,\ldots,\mathbf{n}_N]$.

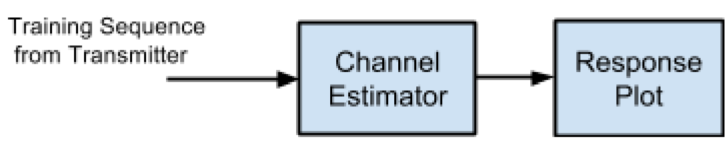
With this notation, channel estimation means that $\mathbf{H}$ should be recovered from the knowledge of $\mathbf{Y}$ and $\mathbf{P}$.

If the channel and noise distributions are unknown, then the least-square estimator (also known as the minimum-variance unbiased estimator) is

$$\mathbf{H}_{\text{LS-estimate}} = \mathbf{YP}^H(\mathbf{PP}^H)^{-1}$$

Currently there is no block for channel estimation in GNU Radio. Hence, **Least-square channel estimation** block will be implemented for GSoC 14. To implement it, QA(Quality Assurance) test file **qa_least_square_cc.py (python)** and implementation file **least_square_cc_impl.cc (c++)** will be developed**.**

For matrix manipulation, **SciPy (Python)** and **Armadillo(C++)** libraries will be used.



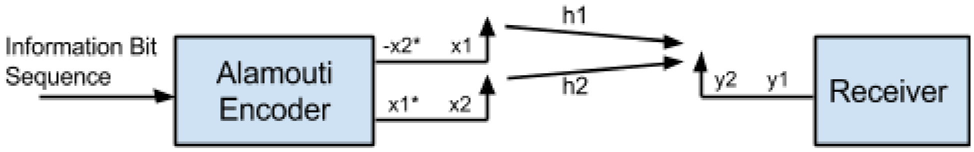
Antenna Diversity using MIMO:

Goals: To develop GRC blocks of Alamouti Encoder (Transmit diversity) and MRC decoder (Receive diversity)

Diversity techniques are used to mitigate degradation in the error performance due to unstable wireless fading channels such as Multipath Fading.

Transmit Diversity : Appropriate precoding of signals is done at the transmitter to enable coherent combining at the receiver.

Alamouti Space-Time Code:
GRC block of Alamouti Encoder will receive stream of data symbols. These data symbols will be encoded using Alamouti Code algorithm. Encoded symbols are transmitted from two transmitting antennas. This algorithm does not require channel state information(CSI)



In first time slot, symbols x_1 and x_2 are transmitted from the first and second antenna respectively. In second time slot symbols $-x_2^*$ and x_1^* are transmitted from the first and second antenna respectively. [2]

In the first time slot, the received signal is,

$$y_1 = h_1 x_1 + h_2 x_2 + n_1 = [h_1 \ \ h_2]\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + n_1 .$$

In the second time slot, the received signal is,

$$y_2 = -h_1 x_2^* + h_2 x_1^* + n_2 = [h_1 \ \ h_2]\begin{bmatrix} -x_2^* \\ x_1^* \end{bmatrix} + n_2 .$$

For convenience, the above equation can be represented in matrix notation as follows:

$$\begin{bmatrix} y_1 \\ y_2^* \end{bmatrix} = \underbrace{\begin{bmatrix} h_1 & h_2 \\ h_2^* & -h_1^* \end{bmatrix}}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2^* \end{bmatrix} .$$

The estimate of the transmitted symbol is,

$$\begin{aligned} \widehat{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}} &= (H^H H)^{-1} H^H \begin{bmatrix} y_1 \\ y_2^* \end{bmatrix} \\ &= (H^H H)^{-1} H^H \left(H \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2^* \end{bmatrix} \right) \\ &= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + (H^H H)^{-1} H^H \begin{bmatrix} n_1 \\ n_2^* \end{bmatrix} \end{aligned}$$

, where

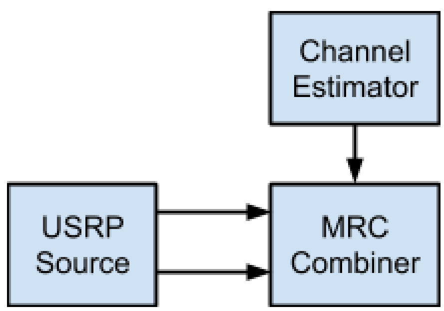
$$(H^H H)^{-1} = \begin{bmatrix} \frac{1}{|h_1|^2+|h_2|^2} & 0 \\ 0 & \frac{1}{|h_1|^2+|h_2|^2} \end{bmatrix}$$

For GSoC 14, encoder and decoder block for Alamouti Code will be developed. QA code files **qa_alamouti_tx_cc.py** and **qa_alamouti_rx_cc.py** will test desiged algorithm of encoder and decoder block. Actual algorithm will be implemented in **alamouti_tx_cc_impl.cc** and **alamouti_rx_cc_impl.cc** C++ files.

For matrix manipulation, **SciPy (Python)** and **Armadillo(C++)** libraries will be used.

Receive Diversity : The received signals in the different antennas can be combined by various techniques. These combining techniques include selection combining (SC), maximal ratio combining (MRC), and equal gain combining (EGC).

For GSoC 14, I will focus on **Maximum Ratio Combining(MRC)** in which signal from each receiving antenna is rotated and weighted according to the phase and strength of the channel, such that the signals from all antennas are combined to yield the maximal ratio between signal and noise terms. [3]



Here knowledge of channel matrix is required and hence we will use channel estimator block discussed earlier.

Consider we have M transmit antenna and one receive antenna and channel is assumed to be flat fading. Consider on the i^{th} receive antenna, the received signal is,

$y_i = h_i x + n_i$; where y_i is the received symbol , h_i is the channel on the i^{th} receive antenna, x is the transmitted symbol and n_i is the noise on i^{th} receive antenna.

Expressing it in matrix form, the received signal is,

$y = hx + n$, where $y = [y_1 y_2 \cdots y_N]^T$ is the received symbol from all the receive antenna

$h = [h_1 h_2 \cdots h_N]^T$ is the channel on all the receive antenna

x is the transmitted symbol and $n = [n_1 n_2 \cdots n_N]^T$ is the noise on all the receive antenna.

The equalized symbol is,

$$\hat{x} = \frac{h^H y}{h^H h} = \frac{h^H h x}{h^H h} + \frac{h^H n}{h^H h} = x + \frac{h^H n}{h^H h}.$$

It is intuitive to note that the term,

$h^H h = \sum_{i=1}^{N} |h_i|^2$ i.e sum of the channel powers across all the receive antennas.

To implement MRC combiner block, QA test file **qa_mrc_combiner_cc.py (python)** and implementation file **mrc_combiner_cc_impl.cc(C++)** will be created.

Spatial Multiplexing :

Goals : To build GRC blocks of Spatial stream generator at transmitter, Zero Forcing detector at receiver.

Spatial multiplexing technique uses multiple antennas at the transmitter and receiver. Capacity of MIMO system grows linearly with min {M,N} where M and N are number of transmitting and receiving antennas respectively. [4]

The transmitter block will split the data sequence into M sub-sequences that are transmitted simultaneously using the same frequency band. Here data rate is increased by a factor of M (Multiplexing Gain).

At receiver, the sub-sequences are separated by means of interference cancellation algorithms like linear **Zero Forcing** (ZF) or **Minimum Mean Squared Error** (MMSE) detector. [5]

Zero Forcing Receiver implements matrix pseudo inverse :

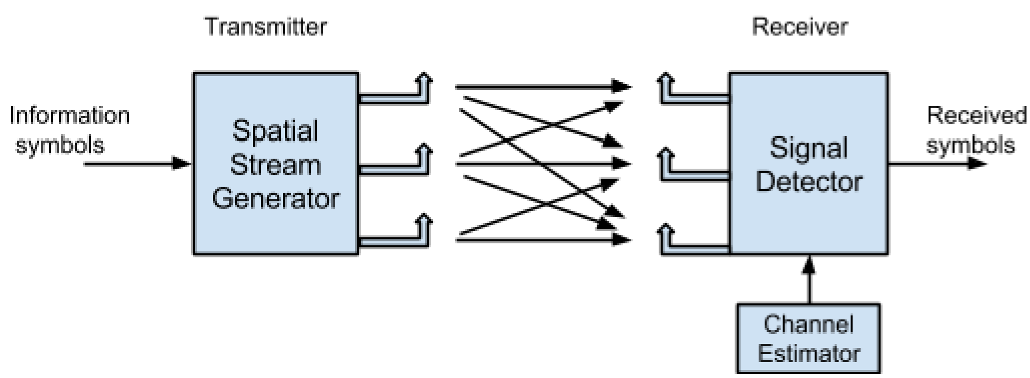
Let x = $[x_1, x_2, \ldots, x_N]^T$ be the vector representing the signals transmitted by the stations and y = $[y_1, y_2, \ldots, y_N]^T$ be the received signals, then for AWGN channel,

$$y = Hx + n$$

Detection of each symbol is given by a linear combination of the received signals by using Zero Forcing as below,

$$x\hat{} = H^{\dagger}y; \quad \text{where } H^{\dagger} = (H^H H)^{-1} H^H \text{ (pseudo–inverse)}$$

Channel estimator block will use training (pilot) data sequence to estimate channel matrix which is required by the signal detector.

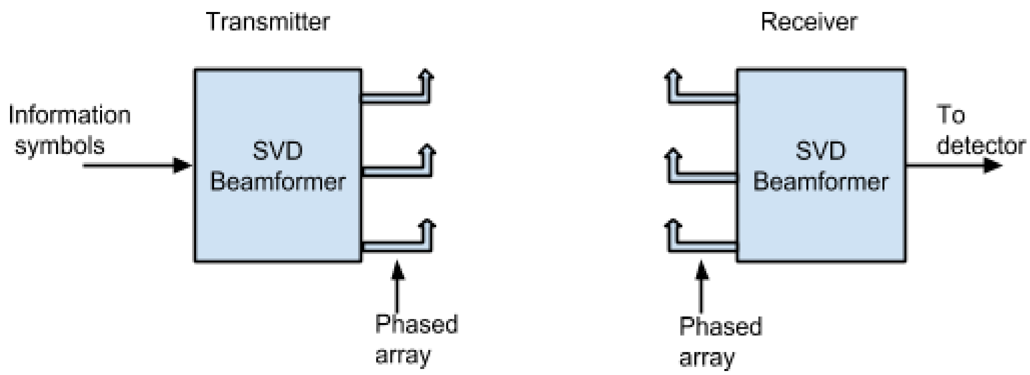


For GSoC 14, Spatial Stream Generator block and Signal Detector block will be implemented. Code for this algorithm will be implemented in C++ files **stream_generator_cc_impl.cc** and **zeroforcing_receiver_cc_impl.cc.** QA test files **qa_stream_generator_cc.py** and **qa_zeroforcing_receiver_cc.py** will validate the algorithm.

MIMO Beamforming :

Goals : To implement GRC block of SVD(Singular Value Decomposition) Beamformer at transmitter and receiver.

Beamforming can be interpreted as linear filtering in the spatial domain. It Improves SNRs or SINRs in multiuser scenarios. SVD beamforming utilizes the properties of estimated channels by performing singular value decomposition on channel matrix.The transmit precoding is performed by multiplying the input symbols with unitary matrix to produce the transmit beamforming. At the receiver, combining process is performed for receiving symbols by using maximum ratio combiner (MRC), and the receiver shaping is performed to retrieve the original input symbols by multiplying the received signal with conjugate transpose of the unitary matrix. From analysis, the SVD beamforming offers the optimal performance in comparing with other techniques. This technique requires feedback channel information. [6]



Assume that the instantaneous realization of the channel matrix H is known by using Channel Estimation block at both transmitter and receiver.

Singular Value Decomposition of H :

$$H := U\Sigma V^H ;$$

where U is m×m real or complex unitary matrix
Σ is m×n rectangular diagonal matrix
V is n×n real or complex unitary matrix

To compute SVD of channel matrix H, we will use **SciPy (Python)** and **Armadillo(C++)** libraries.

Instead of x[k], transmitter sends pre-processed vector x[k] := Vx[k]

At the receiver end, the received signal y is spatially decoupled by U^H as

$$x^{\wedge} = U^H y = U^H(U\Sigma V^H * Vx + n)$$

Two blocks, SVD beamformer at transmitter and receiver will be implemented for GSoC 14.
QA test files **qa_svd_beamformer_tx_cc.py, qa_svd_beamformer_rx_cc.py** will be developed. Implementation will be done in C++ files **svd_beamformer_tx_cc_impl.cc** and **svd_beamformer_rx_cc_impl.cc.**

Hardware Details :

- USRP1, USRP2, USRP N210
- RF Daugherbords : SBX 400-4400 MHz Rx/Tx (40 MHz), WBX 50-2200 MHz Rx/Tx (40 MHz)

- Antennas : VERT2450 Antenna, VERT900 Antenna
- OctoClock-G
- Connectors : MIMO Cable, SMA Connectors

Software Details:
- GNU Radio 3.7.2

External Libraries :
- Scipy/Numpy (Python Library)
- Armadillo (Linear Algebra C++ Library)

Benefits of this project :

MIMO is a key technology for modern wireless standards providing enhanced data rates, high reliability even under the condition of interference. Hence to support both research and real-world systems, we need to build wireless testbeds using appropriate hardware and software tools. GNU Radio is a popular software framework to develop experimental wireless testbeds among research community. Currently there is no stable gr-MIMO module present in current versions of GNU Radio. Through this project, GNU radio users will get benefit of new gr-MIMO module which will help them to build their own wireless testbed on MIMO technology and to practically analyze the features of MIMO.

Deliverables:

For GSoC 14, an out-of-tree module 'gr-MIMO' will be created using gr-modtool script. For each block, Quality Assurance(QA) test code in python will be developed. Actual algorithm will be implemented in C++ file. Phase wise deliverables (including blocks and corresponding code files that will be developed) are mentioned below.

- Channel Estimator block (**Phase 1**) :
        - qa_least_square_cc.py and least_square_cc_impl.cc

- Antenna Diversity Techniques for MIMO
    - Transmit Diversity: Alamouti Space Time Code block (**Phase 2**)
        - qa_alamouti_tx_cc.py and alamouti_tx_cc_impl.cc
        - qa_alamouti_rx_cc.py and alamouti_rx_cc_impl.cc
    - Receive Diversity: Maximum Ratio Combining block (**Phase 3**)
        - qa_mrc_combiner_cc.py and mrc_combiner_cc_impl.cc

- Spatial Multiplexing Techniques for MIMO (**Phase 4**)
    - Spatial stream generator block for transmitter
        - qa_stream_generator_cc.py and stream_generator_cc_impl.cc

    - Zero Force Receiver block for receiver
        - qa_zeroforcing_rx_cc.py and zeroforcing_rx_cc_impl.cc.

- MIMO Beamforming (**Phase 5**)
    - SVD Beamformer block for transmitter and receiver
        - qa_svd_beamformer_tx_cc.py and svd_beamformer_tx_cc_impl.cc
        - qa_svd_beamformer_rx_cc.py and svd_beamformer_rx_cc_impl.cc.

Future Plan:

Due to time constraint, some blocks will be developed in future after summer of code and will be shared to community. The below are such blocks

- MMSE Channel Estimation block
- Minimum Mean Squared Error detector block for receiver(Spatial Multiplexing)
- OSTBC generalized space time code block for transmitter(Transmit Diversity)

Milestones:

A preliminary timetable will be presented here, but the schedule is likely to change during GSoC 2014 after initial discussions with my mentors.

- May 1-May 18 : Discussion with mentor about project details.

- May 19-May 28 :Implementing channel estimation block (**Phase 1**)

- May 29-June 10: Implementing Alamouti code block (**Phase 2**)

- June 11-June 21: Implementing Maximum Ratio Combining block (**Phase 3**)

- June 22 : Documentation and test cases

- June 23 : Mid term evaluation

- June 28-July 10 : Implementing Stream generator and Zero forcing error block(**Phase 4**)

- July 11-July 31: Implementing SVD Beamformer blocks (**Phase 5**)

- August 1- August 13 : Fixing bugs and testing

- August 14 -August 17: Final documentation

- August 18 : Final Evaluation

Background Information:

I am currently pursuing MS by Research in Speech Processing and Communication Research Center (SPCRC) lab in IIIT-Hyderabad, India. It is one of the few institutes in India, working on Software Defined Radio(SDR). I worked for two years in Tata Consultancy Services, Mumbai [2011-2013], a software company after completing my Bachelor of Engineering in Electronics and Telecommunication at Sardar Patel Institute of Technology, Mumbai, India.

Two years of IT experience made me organized person which helps me to remain focussed on work and to deliver the task within scheduled time.

I am very passionate about open source and am using it for my daily activities. I am working on GNU radio after I joined MS program and did a lot of basic experiments with GNU radio. My current lab project is on Cognitive Cellular Network which is supported by Govt of India, Department of Electronics and Information Technology(DEITY). I am also working on ‘Direction Finding’ as my semester project which is being implemented in GNU Radio.

Being from communication background, I have all necessary prerequisite knowledge required for this project. Also I am well versed with all hardware components mentioned earlier.

After I started working on GNU Radio, I joined GNU Radio mailing list which helped me to resolve many of my doubts and I gained good knowledge from it. I read each discussion posted by users to understand different problems faced by them. Though I could not resolve them yet, but in near future, I am hopeful that I can definitely participate actively in the discussions.

GSoC is an excellent opportunity for me to contribute to the GNU Radio community and surely I will contribute to it even after completion of summer of code.

Software and hardware Skills :
Software : GNU Radio, C and C++ , python, MATLAB, java, Sharepoint 2010 , Visual Studio .NET, XML, HTML, CSS.

Hardware: USRP1, USRP2

Availability: I will available full time for this project as we have three month of summer vacation from 1 May to 27 July. I can spend 40Hrs per week as I don’t have any other commitments. I may visit my home town for 4-5 days, but I will make sure that it will not impact any deliverables.

Previous work done:

To contribute to the community, we started with youtube channel (SDR India) of our experiments (done along with labmate). Below is the link for the same
http://www.youtube.com/channel/UCStYdbRK-d64_hohDMvIogQ

References:

[1] M. Biguesh and A. Gershman, Training-based MIMO channel estimation: a study of estimator tradeoffs and optimal training signals, IEEE Transactions on Signal Processing, vol 54, pp. 884-893, 2006.

[2] S.M. Alamouti (October 1998). "A simple transmit diversity technique for wireless communications". IEEE Journal on Selected Areas in Communications 16 (8): 1451–1458. doi:10.1109/49.730453

[3] http://en.wikipedia.org/wiki/Maximal-ratio_combining

[4] www.ece.ubc.ca/~janm/Lectures/lecture_mimo.pdf

[5] http://en.wikipedia.org/wiki/Multiple-input_multiple-output_communications.

[6] In-Keong Choi; Seong Rag Kim; In-Keong-Choi, "Eigenbeamforming with selection diversity for MIMO-OFDM downlink," Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th , vol.3, no., pp.1806,1810 Vol. 3, 26-29 Sept. 2004**.**