

USRP を用いたチャネル推定と等化に関する研究

2012SE015 藤田光紀 2012SE107 河村裕介

指導教員：奥村康行

1 はじめに

近年、無線通信において、その技術は日々新しく開発されており、それに伴って新しいシステムが次々と導入されている。しかし、ユーザにとって新しいシステムを利用するための端末を随時用意するのは非常に難しく、また、経済的にも負担となると考えられる。開発者の観点でも新規装置を開発する場合のコストは材料費よりも人件費の割合が高くなっていることから、開発にも人的な負担が大きい。

これらの理由から、一つの無線機で様々な機能の追加やシステムの変更と更新に対応できる無線端末の開発が期待されている。そこで、ソフトウェア無線の概念がその問題を解決するための手段として挙げられる [1][3][4]。

本研究ではソフトウェア無線通信プラットフォームとして USRP2 と LabVIEW の二つを用いて、様々な通信方式の実装方法を研究するとともに、その通信方式について明らかにする。

2 研究対象の技術

本研究で用いる技術について説明する。それについて以下に述べる。

2.1 ソフトウェア無線

ソフトウェア無線とは、従来の無線通信端末においてハードウェアで行われていた信号処理のほとんどをソフトウェアで行う無線通信のことである。この技術によって、ユーザはサービスの切り替え、ソフトウェアの変更と更新、新規の通信規格など以前はハードウェアを変える必要があったものを、ハードウェア上のソフトウェアを切り替えるだけで対応できるようになった。その一つとして汎用性の高い USRP というハードウェア、動作制御に LabVIEW というソフトウェアを使用する [1]~[4]。

2.2 USRP の概要

USRP (Universal Software Radio Peripheral) は汎用信号処理ハードウェアである。研究で使用する USRP2 は USRP1 よりも高性能であり、利用可能周波数は 50MHz ~ 2.2 GHz である。PC との接続にはギガビットイーサネットを使用する。USRP2 は図 1 のように Daughter board (ドーターボード) と Mother board (マザーボード) から構成される。受信した信号はアンテナから入って、USRP2 の Daughter board で中間周波数に落とされる。Daughter board から送られてきた信号は Mother board でアナログ信号からデジタル信号変換され、条件に従って標本化される。また信号を送信する際は、Mother board から送られて来た信号を Daughter board で中間周波数に上げる [3][4]。

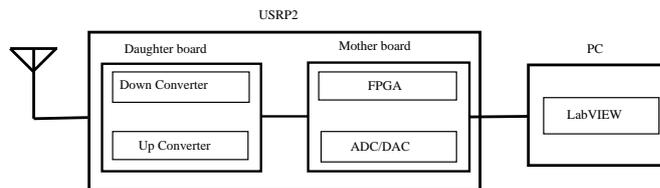


図 1 USRP2 の構成 [3][4]

2.3 LabVIEW

LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) は NATIONAL INSTRUMENTS 社が 1986 年に開発したバーチャル計測用かつグラフィカルなプログラミング言語である。グラフィカルプログラミング言語とは視覚的な操作でプログラムを作成する言語である。LabVIEW で作成されたプログラムは VI (Virtual Instrument) と呼ばれる。作成されたプログラムファイルの拡張子は vi である [3][4]。

3 実装する技術

送信信号と受信信号の誤差を小さくし、信号を正確に伝えるために、受信信号の等化を行う必要がある。等化の処理を行うためのプログラムを作成し、受信機のプログラムに実装する。そのため、2 つのプログラムの作成を行った。作成した mmse_linear.vi と make-Toeplitz.vi を受信プログラム (top_rx.vi) に実装することで、送信信号を推定する。実装したプログラムのつながり方を図 2 に示す。作成したプログラム (図 2 中の破線) を以下に示す。

3.1 mmse の実装

送信側が送る信号を、受信した信号とチャネル (伝送路) の伝達関数から復元する。送信信号とチャネルの伝達関数を畳み込みして、受信信号を導き出す。そのために、チャネルの伝達関数を知る必要がある。mmse (minimum mean square error) を用いて送信側の信号と受信側が推定した信号の二乗誤差が最小になるようにチャネルの伝達関数を推定することで、送信信号をより正確に受信信号として推定することができる。これを行うためのプログラム mmse_linear.vi を作成し、receiver.vi に組み込む。

3.2 mmse_linear.vi

mmse_linear.vi は送信信号と受信信号からチャネルの伝達関数を推定し、送信信号と推定した受信信号の差を二乗した値を計算するプログラムである。図 3 は作成した mmse_linear.vi のプログラムのブロック図である。下記に記すプログラムを繋ぎ合わせて作成されている。また、

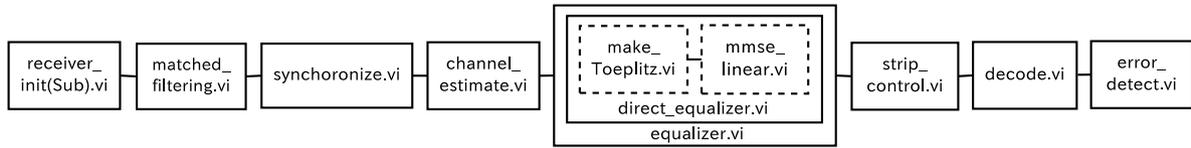


図 2 receiver.vi のブロック図

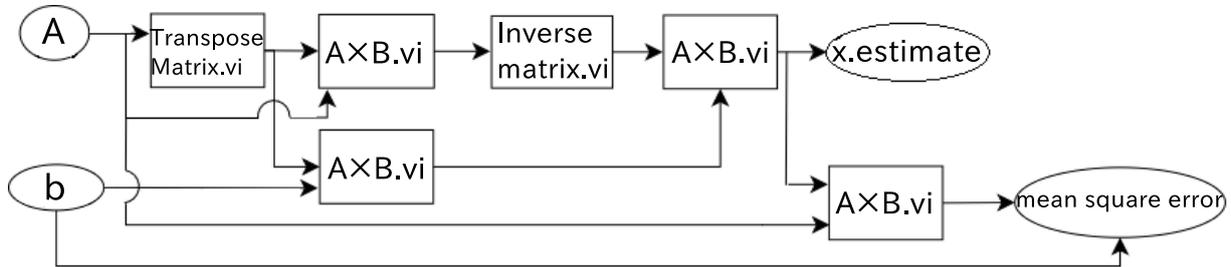


図 3 mmse_linear.vi のブロック図

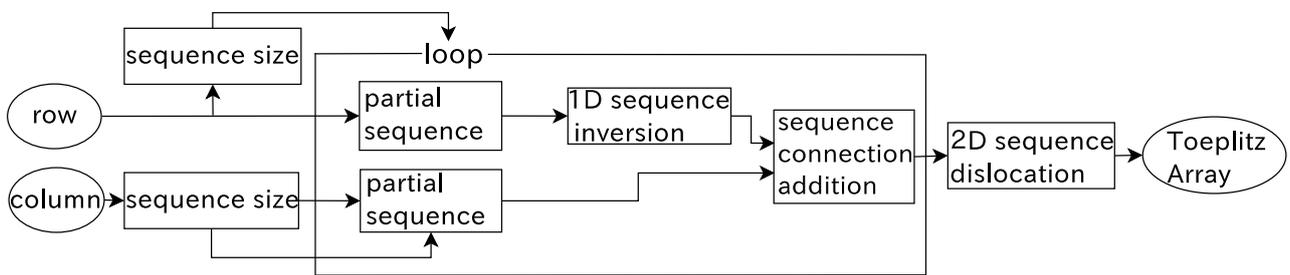


図 4 make_Toeplitz.vi のブロック図

チャンネルの伝達関数の推定と二乗誤差を算出する式は以下の第 3.5 節に記す。構成する各プログラムについて説明する。Transpose Matrix.vi は複素共役転置行列を行うプログラムである。入力された行列を転置する。入力行列が複素行列の場合、共役転置を実行する。A × B.vi は複素 A × B を行うプログラムである。2 つの入力行列、または入力行列と入力ベクトルの乗算を行う。Inverse Matrix.vi は複素逆行列を算出するプログラムである。入力された行列に逆行列が存在する場合、入力行列の逆行列を求める。

3.3 Toeplitz の実装

Toeplitz 行列とは 1 行 1 列で定義され、左から右の各下降対角線に沿って要素が一定であるような行列である。mmse_linear.vi のプログラムで畳み込みを用いる際、受信した信号ベクトルを Toeplitz 行列にして計算する。受信した信号ベクトルを Toeplitz 行列に変換するプログラムを作成し、mmse_linear.vi の直前に組み込む必要がある。Toeplitz 行列を作成するプログラムを make_Toeplitz.vi とし、その説明を以下に記す。

3.4 make_Toeplitz.vi

受け取った送信信号から Toeplitz 行列を作成するプログラムである。図 4 は作成した make_Toeplitz.vi のプログラムのブロックである。プログラムの流れは、まず、送

信信号の行ベクトルと列ベクトルを受け取る。ベクトルの要素数を把握する（配列サイズ）。受け取った信号の要素数に合った行列を作成する（部分配列、1D 配列反転、配列連結追加）。行列を転置（2D 配列転置）して Toeplitz 行列完成である。以下に make_Toeplitz.vi 内で組み立てられているブロックについて記す。

3.4.1 各ブロックの役割

「配列サイズ」は配列の各次元の要素数を調べて返すブロックである。「部分配列」は設定した指標から長さの分だけ要素数を含む配列の一部を返すブロックである。「1D 配列反転」は配列の要素の順番を逆に並び替えるブロックである。「配列連結追加」は複数の配列同士を継げたり、要素を追加するためのブロックである。「2D 配列転置」は 2D 配列 $[i, j]$ を転置させた配列 $[j, i]$ に置き換えるブロックである。

3.5 線形最小二乗法

線形最小二乗法はチャンネルを推定するため、イコライザを計算するために用いられる。A を $N \times M$ 行列とし、N が行数、M が列数を表す。行列 A に対し、エルミート転置を A^* と表現する。b を $N \times 1$ 次元のベクトルとする。A が $N = M$ の正方行列で列が線形独立ならば A は逆行列が可能である。A が逆行列可能ならば、逆行

列 A^{-1} が存在し、 $AA^{-1} = A^{-1}A = I$ を満たす。もし A が $N > M$ で、列が線形独立ならば A はフルランクである。また A が $N < M$ で、行が線形独立ならば A はフルランクである。

$$Ax = b \quad (1)$$

式 (1) のような行列形式で書かれる線形方程式で A はデータという既知の係数行列、 x は未知のベクトル、 b は観察ベクトルという既知のベクトルである。 A はフルランクであるとする。(1) 式で $x = A^{-1}b$ と表せる。

次に $N > M$ とする。この条件だとシステムは重複決定されてしまう。このようなケースの時、最小二乗法を用いて、およそその解を決定する。(1) 式を解く代わりに、 $\min \|Ax - b\|^2$ のノルムの二乗誤差を最小になるようにすればよい。 A がフルランクであることを仮定して、 x の最適値を求めると

$$x_{LS} = (A^*A)^{-1}A^*b \quad (2)$$

となる。 x_{LS} は (1) 式の最小二乗誤差である x の推定値である。求めた x の推定値のクオリティを測定するため、 x_{LS} によってできる二乗誤差を測定する。最小二乗誤差は

$$J(x_{LS}) = \|Ax_{LS} - b\|^2 = x^*A^*(Ax - b) - b^*(Ax - b) = b^*b - b^*Ax \quad (3)$$

によって得られる。上記より、チャネル推定のために最小二乗法による解法を利用する。

3.6 チャネル推定

受信機で送信側のデータを推定するには、その信号が通過したチャネルにあったフィルタを通す必要がある。フィルタを作成するためにチャネルを推定する必要がある。送信機がデータを送る際、データよりも前にトレーニング信号を付けて送信する。トレーニング信号は送信側と受信側であらかじめ、何を送るか決められている既知系列である。チャネルの伝達関数を推定するために線形最小二乗法を用いる。

$$\{\hat{h}[0], \hat{h}[1], \dots, \hat{h}[L]\} = \arg \min_{a[0], a[1], \dots, a[L]} \sum_{n=L}^{N_t-1} \left\| y[n] - \sum_{l=0}^L a[l]t[n-l] \right\|^2 \quad (4)$$

$\hat{h}[n]$ はチャネルの伝達関数の推定値、 $a[l]$ は変数、 $t[n]$ はトレーニング信号、 $y[n]$ は受信信号をそれぞれ表す。

(4) 式の \sum の二乗が最小になるような $a[l]$ の値をチャネルの伝達関数の推定値 $\hat{h}[n]$ とする。受信信号と伝達関数、トレーニング信号を行列の形で表現した式が以下の (5) 式である。

$$\begin{bmatrix} y[L] \\ y[L+1] \\ \vdots \\ y[N_t-1] \end{bmatrix} = \begin{bmatrix} t[L] & \dots & t[0] \\ t[L+1] & \ddots & \vdots \\ \vdots & & \vdots \\ t[N_t-1] & \dots & t[N_t-1-L] \end{bmatrix} \begin{bmatrix} a[0] \\ a[1] \\ \vdots \\ a[L] \end{bmatrix} \quad (5)$$

$t[n]$ の行列を T とする。 T が縦に長い行列でフルランクならば、 T^*T は二乗行列で逆行列にすることができる。二乗等化を行うには $N_t - L \geq L + 1$ 、つまり $N_t \geq 2L + 1$ であることが必要である。

4 評価と考察

実世界と近似である AWGN (additive white Gaussian noise) の空間で noise power の値を 0dB から -30dB の範囲で 5dB ずつ変化させシミュレーションと実測を行った。実測とシミュレーションの測定条件は同様とし、表 1 に示す。実測は USRP2 を 2 台用いて行った。2 台の USRP2 を同軸ケーブルと MIMO ケーブルでつなぐ。次に、PC と USRP2 の一方を LAN ケーブルでつなぎ、PC と 2 台の USRP2 を同期させる。この実測の構成図を図 5 で示す。シミュレーションは LabVIEW のソフトウェアを用いて行った。作成した mmse_linear.vi と make_Toeplitz.vi のプログラムを追加した受信プログラム (top_rx.vi) を使用し、雑音による影響をどれだけ抑えられるかを測定した。シミュレーション、実測ともに 500bit の信号を 10000 回送り、その平均の BER と二乗誤差の値を算出し、グラフにまとめた。USRP2 を用いた実測結果とシミュレーションの結果を評価、考察をする。

表 1 測定条件

software	LabVIEW
channel model	AWGN
modulation type	QPSK
Packet length[bit]	500
Carrier frequency[Hz]	2.2G
TX sample rate[Samp/sec]	20M
TX oversample factor	4
RX sample rate[Samp/sec]	10M
RX oversample factor	2
Channel estimate length	6
Equalizer length	6
Capture time[sec]	80 μ

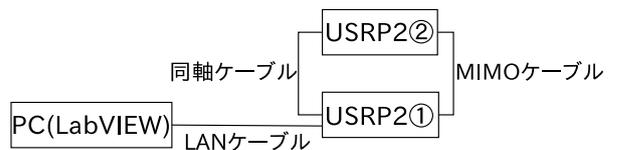


図 5 実験の構成図

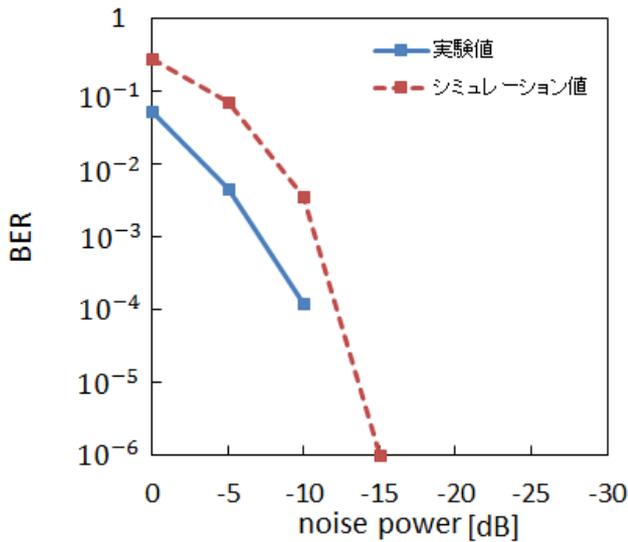


図6 BERの測定結果

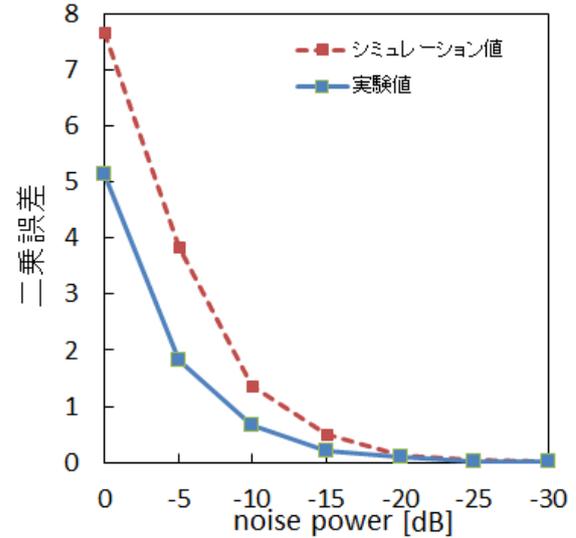


図7 二乗誤差の測定結果

4.1 追加したプログラムによる BER の評価

noise power の影響による信号配列の BER を測定した。この実験結果を図 6 に示す。これにより作成したプログラムがどれほどの雑音に対応できるかを観測することができた。測定すると、シミュレーション時では noise power が-15dB の値以降から BER が 0 になるが、実験時では noise power が-15dB の時点で BER は 0 になる。実測の結果から実測したプログラムを用いて-10dB 以下の雑音比ならば、500bit の信号は正しく送ることができると分かった。

4.2 追加したプログラムによる二乗誤差の評価

次に送信信号の信号配列と推定した受信信号の信号配列の二乗誤差を調べる。この実験結果を図 7 に示す。実測の結果、noise power が-5dB から-15dB まではシミュレーション結果よりも二乗誤差を半分に抑えることができた。さらに、noise power が-15dB 未満の場合は実験値、シミュレーション値との大きな違いは見られず、BER の値も共に 0 だった。結果から、シミュレーション、実験共に雑音の影響を緩和できていると言える。

4.3 総合評価

現実での noise power は-10dB から-20dB の数値である。BER と mmse をシミュレーションと実験の結果から作成したプログラムにより、実用できる範囲内で雑音の影響を緩和できている。シミュレーションと実験の誤差は実験で使用した同軸ケーブルによるものであると思われる。

5 おわりに

本研究では、LabVIEW のソフトウェアを用いて、受信プログラムに作成したプログラムを実装し、送信信号と受信信号の誤差を最小にすることを目指した。まず、チャンネルの伝達関数を推定するためのプログラム

mmse_linear.vi を作成した。その際、受け取った送信信号を Toeplitz 行列に直す必要があるため、それを行うプログラム make_Toeplitz.vi を作成し、実装する。これらのプログラムから送信信号と推定した受信信号の二乗誤差を確認した。また、受信信号と送信信号から BER の値を測定し、実装したプログラムの動作を確認した。

参考文献

- [1] 鈴木康夫, 荒木純道, “ソフトウェア無線機とその国内における開発の現状,” 電子情報通信学会論文誌 B, vol.J84-B, no.7, pp.1120-1131, July 2001.
- [2] 河野隆二, 春山真一郎, “ソフトウェア無線の現状と将来,” 電子情報通信学会論文誌 B, vol.J84-B, no.7, pp.1112-1119, July 2001.
- [3] 木村一也, 早井智穂, 加藤万貴, “USRP2 への物理層通信機能の実装に関する研究,” 南山大学情報理工学部システム創成工学科 2012 年度卒業論文, January 2013.
- [4] 朝倉彰洋, 北原慎平, 西田結梨香, “USRP2 への 16QAM とシンボル同期の実装に関する研究,” 南山大学情報理工学部システム創成工学科 2014 年度卒業論文, January 2015.
- [5] Robert W. Health Jr., “Digital Wireless Communication, Student Lab Manual,” pp.1-81, NATIONAL TECHNOLOGY&SCIENCE PRESS, 2012.
- [6] National Instruments, “グラフィカルプログラミング,” <http://japan.ni.com/> (accessed 2015).
- [7] 久保博嗣, “電子情報通信学会 知識ベース, 4 群 モバイル・無線, 1 編 無線通信基礎, 8 章 復調技術,” http://www.ieice-hbkb.org/files/04/04gun_01hen_08.pdf (accessed Sep.2015).