

GNU Radioに関する調査

猿渡 俊介[†] 菅沼 久浩[†]

[†] 東京大学 先端科学技術研究センター 森川研究室

〒 153-8904 東京都目黒区駒場 4-6-1

E-mail: †{saru,suganuma}@mlab.t.u-tokyo.ac.jp

森川研究室 技術研究報告書 No. 2011001

2011年6月1日

あらまし 本稿では、オープンソースのソフトウェア無線ツールキットである GNU Radio の仕組み、使い方、応用例、課題について紹介する。既存のソフトウェア無線の多くが FPGA で無線の物理層の処理を行っているのに対し、GNU Radio ではパーソナルコンピュータ上で動作する Python で無線の物理層の処理を行う。GNU Radio は C++ で提供されたライブラリと Python を密に連携させることで Python の持つ簡便性、柔軟性を最大限に生かしつつもパーソナルコンピュータの性能を引き出すことに成功している。GNU Radio の登場により多様な専門背景を持つユーザが電波を簡単に扱えるようになったことで、広い分野で多くの研究成果が生まれ始めている。

キーワード GNU Radio, USRP, ソフトウェア無線, オープンソース, 信号処理

1. はじめに

この技術は魔法のような技術であろうか。筆者が新しい技術と対峙した時に、常に自分に対して投げかける問いである。この問いはアーサー・C・クラークの「洗練された技術は魔法と区別がつかない」を由来としている。「魔法のような技術」の実現に関わることが研究者としての筆者の大きなテーマである。

「魔法のような技術」を創るためのヒントを得るために、まず最初に筆者が調べたのは HCI (Human Computer Interaction) の世界である。HCI で生まれた技術は新しい体験を我々に与えてくれる。筆者の好きな研究のひとつに Tangible Bits [1] がある。今までコンピュータの中に閉じ込められていた情報が空間に染み出していくような感覚に衝撃を受けた。それと同時に、Tangible Bits での経験は魔法のように感じるものの、波及効果という観点では個人のその場での一時的な体験に制限されるという限界を感じた。

そこで筆者は、方向性を変えて技術の歴史を追った。人類の歴史の転換点となった活版印刷、電信、飛行機、パーソナルコンピュータ、インターネットなどのさまざまな技術を調べていくうちに、筆者はひとつの答えに行きついた。波及効果の高い「魔法のような技術」とは、今までに存在しなかった全く新しい技術なのではなく、今までごく一部の人がしか享受することができなかったものを誰でも簡単に実現する技術なのではなかろうか。想像したことがいとも簡単に誰の手によっても具現化できる。先に挙げた活版印刷、電信、飛行機、パーソナルコンピュータ、インターネットによって実現されたサービス自体は既存のものであり、既存のサービスの実現コストを圧倒的に下げるといった特徴を備える。誰もが魔法を使えるようになったかのような技術こそが、波及効果の高い魔法のような技術である

と言えよう。

前置きが長くなったが、本稿のテーマは GNU Radio である。GNU Radio は、米国の Eric Blossom が始めたオープンソースのソフトウェア無線ツールキットである [2]。電波送受信装置のプロトタイピング、電波を扱う研究、無線通信の教育の3つに適しているツールである。筆者は、GNU Radio は未熟な点が多いものの「魔法のような技術」に当てはまると考えている。GNU Radio は電波送受信装置の開発を圧倒的に簡単にするからである。

GNU Radio はソフトウェア無線技術に分類される。GNU Radio が既存のソフトウェア無線技術と大きく異なる点が1点存在する。既存のソフトウェア無線技術の多くが専用ハードウェア上の FPGA で動作するのにに対し、GNU Radio ではパーソナルコンピュータ上で動作する Python で無線の物理層の処理を行う点である。Python はスクリプト言語に分類される。一般的に、スクリプト言語は柔軟性が高いものの処理の遅い言語である。電波の処理を行うデジタル信号処理は重い処理である。GNU Radio はコンパイラ言語である C++ で提供されるライブラリとスクリプト言語である Python を巧妙に連携させることでスクリプト言語の持つ簡便性、柔軟性を最大限に生かしつつもパーソナルコンピュータの性能を活かすことに成功している。

これまで電波や回路技術に精通した一部の人がしか電波送受信装置を開発できなかったのに対し、GNU Radio の登場により、開発者の裾野が大きく広がった。無線回路の専門家のみならず、多様な専門背景を持つユーザが電波を簡単に扱える環境ができたことで、これまでの電波の扱い方を根本から変えるような革新的な技術の登場が期待できる。実際に、SIGCOMM, MobiCom, INFOCOM などのトップ国際会議では GNU Radio を

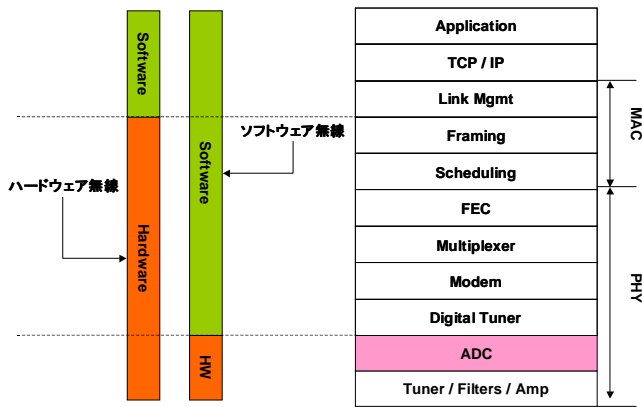


図1 ソフトウェア無線機とハードウェア無線機の比較

用いた研究が増加する傾向にある。

本稿の構成は以下の通りである。まず2.で、GNU Radioが生まれた背景を明らかにすることを目的として、ソフトウェア無線技術の概要と歴史について述べる。3.では、GNU RadioとUSRPがどのように動作するかを述べる。4.では、実際にGNU Radioを体験するために必要な機材とGNU Radioの使い方について説明する。5.では、GNU Radioを用いて行われている研究開発について述べる。6.で現在のGNU Radioの課題について述べ、最後に7.でまとめとする。

2. ソフトウェア無線技術

ソフトウェア無線・ソフトウェアラジオ (SDR: Software Defined Radio: SDR) は、従来の電波送受信装置において、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技術である。電波送受信装置上で動作するソフトウェアを切り替えるだけで、使用する変復調の方式や利用するサービスを切り替えたり、ソフトウェアの追加や更新によって新しい通信規格に対応することができる。

図1に既存のハードウェア無線機とソフトウェア無線機の比較を示す。現在の無線機の主流の構成であるハードウェア無線機では、MAC層で受信したパケットをフレームに分割するフレーミングまでをハードウェア処理するのが一般的である。それに対して、ソフトウェア無線機では、AD変換後、DA変換前のデジタル信号を全てソフトウェアで処理する。

ソフトウェア無線技術を用いることで、究極的には、普段使っている携帯電話を用いてインターネットから新しい無線通信機能をソフトウェアとしてダウンロードすると、携帯電話がFMラジオになったり、テレビ放送局になったり、無線LANで通信可能になったり、Bluetoothで通信可能になったりなどといったことが実現できる。ソフトウェア無線技術は、無線通信の概念を変える可能性を秘めた技術である。

ソフトウェア無線の起源としては、軍事目的での研究開発が1970年ごろから行われていたとされている。我々の目に見える形で現れたのは、1990年代前半に米軍が展開したPeakeasy [3]と呼ばれるプロジェクトである。当時の米軍では、各組織の軍事通信システムに互換性が無かったことや、ソフトウェア無線

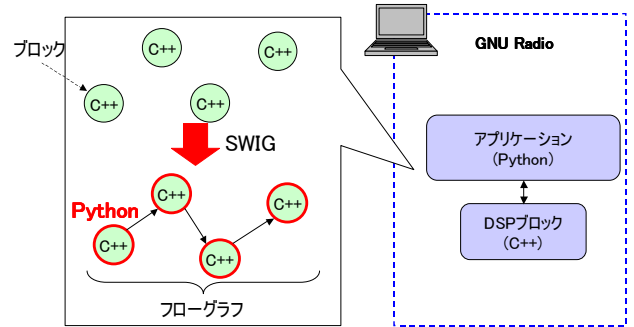


図2 GNU Radio 概要

技術を用いたマルチモード無線機によって冷戦終結に伴う軍事予算の削減を行うなどの目的で、ソフトウェア無線技術の開発を行っていた。

ソフトウェア無線の概念が研究者の間で広く知られるようになるきっかけとなったのは、1995年のIEEE Communication Magazineでのソフトウェア無線の特集 [4] である。コグニティブ無線 [5] の発案者であるとされている Joseph Mitola もソフトウェア無線の記事を執筆している [6]。その後、ソフトウェア無線に関する研究開発プロジェクトが国内外で立ち上がり、活発に研究が行われるようになった。

現在、主流のソフトウェア無線技術はFPGAを用いたものである。FPGAを用いたソフトウェア無線技術では、アンテナから入ってきた電波をAD変換し、デジタル信号をFPGAで処理する。FPGAはField Programmable Gate Arrayの略であり、回路素子がプログラマブルになったものである。回路をソフトウェア的に書き換えられる上に、汎用CPUに比べて高速でデジタル信号処理を実行することができる。

3. GNU Radio と USRP

現在主流のソフトウェア無線技術がFPGAでデジタル信号処理を行っているのに対し、全てのデジタル信号処理をパーソナルコンピュータ上のスクリプト言語でやってしまうというのがGNU Radioの基本コンセプトである。GNU RadioはUSRPと呼ばれる無線フロントエンドとセットで使用される。電波を受信する場合、USRPが電波をデジタル信号に変換してパーソナルコンピュータに渡すと、GNU Radio上で処理される。電波を送信する場合、GNU Radioで処理したデジタルデータがUSRPへと渡され、電波に変換されて放出される。

3.1 GNU Radio

GNU Radioは2001年にEric Blossomが開始したソフトウェア無線を開発するためのツールキットである [2]。図2にGNU Radioの全体像を示す。GNU Radioでは、C++で記述された信号処理ブロックをPythonで接続して無線機を実装する。C++で記述された信号処理ブロックは、それぞれ入力と出力のインタフェースを持ち、SWIGと呼ばれるラップを使ってPythonから呼び出せる形に加工される。ユーザはPythonを用いて複数のブロックの入出力を接続することで無線機を構築する。Pythonによって作られた無線機をフローグラフと呼ぶ。

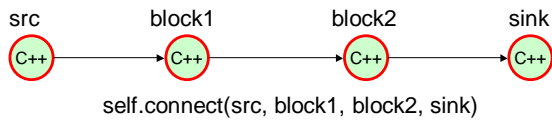


図 3 connect 関数でフローグラフを作成

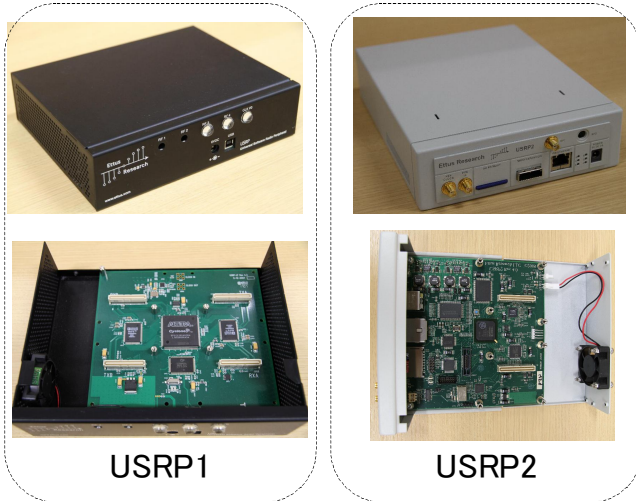


図 4 USRP: Universal Software Radio Periferal

図 3 に Python 上の `connect` 関数を用いて作成したフローグラフの例を示す。信号処理ブロックには、ソース (src)、ブロック (block1, block2)、シンク (sink) の 3 種類が存在する。ソースとは、USRP、パーソナルコンピュータのマイク、サイン波を生成するブロックなど、出力しか持たない信号処理ブロックである。シンクは、USRP、パーソナルコンピュータのスピーカなど、入力した持たない信号処理ブロックである。図 3 では、Python 上の `connect` 関数を用いて `src`, `block1`, `block2`, `sink` を直列に接続してフローグラフを作成している。例えばスピーカに対してサイン波を入力したい場合には Python で

```
src = gr.sig_source_f (sample_rate, gr.GR_SIN_WAVE,
                      350, ampl)

dst = audio.sink (sample_rate, "")
self.connect (src, dst)
```

のように記述する。プログラムを実行するとスピーカから 350Hz の音が聞こえる。

3.2 USRP

USRP は、Universal Software Radio Periferal の略であり、メインボードとドータボードから構成される。メインボードでは、ドータボードから入ってきた信号をデジタル信号として扱えるようにするための AD 変換をした後、設定された通信レートに応じて間引きを行う。全ての周波数帯の送受信で共通のメインボードを使用する。ドータボードは、アンテナから入ってきた電波を中間周波数に落としたり、メインボードから送られてきた信号を送信周波数に上げたりする。

2011 年 4 月の時点では、USRP には USRP1 と USRP2 が存在する。共に Ettus Research 社から発売されている [7]。図 4 に USRP1 と USRP2 を示す。USRP1 と USRP2 の違いは、USRP1 が使用の簡便さを追求しているのに対し、USRP2 が性

能を追求していることである。USRP1 では、パーソナルコンピュータとの接続に USB を用いているため、パーソナルコンピュータとの通信速度は 32MB/s である。ドータボードを 2 つ挿することができる。USRP2 では、パーソナルコンピュータとの接続にギガビットイーサネットを用いているため、パーソナルコンピュータとの通信速度は 100MB/s である。ドータボードは 1 つしか挿すことができない。ドータボードを同時に 2 つ使いたいなどの特殊な理由のない限り、性能の高い USRP2 を用いるのが一般的である。

ドータボードは、対象としている周波数帯や電波の送受信に応じて付け替えるオプションボードである。例えば、50MHz~870MHz の電波の受信が可能な TVRX, 800MHz~2.4GHz の電波を受信可能な DBSRX, 2.3GHz~2.9GHz の電波を送受信可能な RFX2400 などが存在する。ドータボードは随時新しいものが開発されているため、詳細は Ettus Research 社のウェブサイト参照されたい [7]。

4. ソフトウェア無線を体験する

本節では、GNU Radio を用いてソフトウェア無線を実際に体験する。本節で紹介する機能を実現するために必要なものは、ギガビットイーサネットを持ったパーソナルコンピュータ、ギガビットイーサケーブル、USRP2、ドータボードの TVRX, TVRX と USRP2 のアンテナノを接続する F-Bulkhead ケーブル、アンテナである VERT900 である。パーソナルコンピュータ以外はすべて Ettus Research 社のウェブサイトから購入できる。なお、厳密には VERT900 は TVRX 用のアンテナではないことに注意されたい。TVRX にテレビ用のアンテナを接続することも可能である。

4.1 GNU Radio のインストール

GNU Radio のインストールは Debian GNU/Linux をベースとしたディストリビューションである Ubuntu で行うのが最も簡単である。Ubuntu をパーソナルコンピュータにインストールし、`/etc/apt/source.list` に以下の 2 行を追加する。

```
deb http://deb http://gnuradio.org/ubuntu stable main
deb-src http://gnuradio.org/ubuntu stable main
```

そして、コマンドラインから

```
$ sudo apt-get update
$ sudo apt-get install gnuradio gnuradio-companion
$ sudo addgroup yourname usrp
```

を実行するとインストールが完了する。

4.2 スペクトルの表示

GNU Radio と USRP を用いてスペクトルアナライザを構築する。コマンドラインから `usrp2_fft.py` を実行すると図 5 のような画面が表示される。`usrp2_fft.py` は GNU Radio が提供している組み込みコマンドである。GNU Radio をインストールすると `/usr/bin` 以下に USRP 関連のコマンドが追加される。中心周波数をラジオの電波が存在する周波数帯に合わせるとスペクトラムのピークが表示される。図 5 では、81.3MHz の J-WAVE や、82.5MHz の NHK の電波が確認できる。

図 6 に `usrp2_fft.py` で構築するフローグラフを示す。

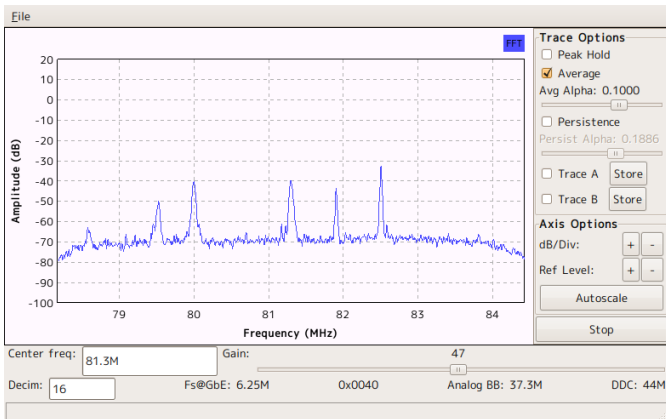


図 5 スペクトラムアナライザのスクリーンショット

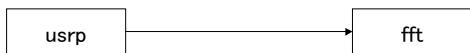


図 6 スペクトルアナライザのフローグラフ

```

src = usrp2.source_32fc()
usrp_decim = 312
src.set_decim(usrp_decim)
src.set_center_freq(81.3e6)
adc_rate = src.adc_rate()
usrp_rate = adc_rate / usrp_decim
audio_decim = 10
audio_rate = usrp_rate / audio_decim
param = optfir.low_pass (1,          # gain
                        usrp_rate, # sampling rate
                        80e3,       # passband cutoff
                        115e3,      # stopband cutoff
                        0.1,        # passband ripple
                        60)         # attenuation

blk_lp = gr.fir_filter_ccf(1, param)
blk_fm = blks2.wfm_rcv(usrp_rate, audio_decim)
sink = audio.sink(int(audio_rate), "plughw:0,0")

self.connect(src, blk_lp, blk_fm, sink)

```

図 7 FM ラジオのソースコード

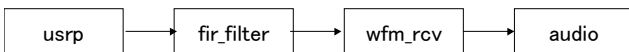


図 8 FM ラジオのフローグラフ

USRP2 のソースブロックと FFT のシンクブロックを接続するというシンプルな構成である。usrp2_fft.py を実行すると、USRP2 から送られてきた電波に対して高速フーリエ変換をかけ、パワースペクトラムとして画面に表示する。

4.3 FM ラジオの受信

スペクトルアナライザと同じハードウェアを用いて、今度は FM ラジオを構築する。図 7 に FM ラジオのソースコードを示す。FM ラジオでは、USRP2 (usrp) から入ってきた 81.3MHz 帯の信号から高周波成分をローパスフィルタ (fir.filter) で除

去し、FM 復調 (wfm_rcv) したものをスピーカ (audio) に入力している。Python 上で connect 関数が呼ばれると図 8 のようなフローグラフが実行され、スピーカから 81.3MHz 帯で放送されているラジオの音声が流れる。

4.4 その他の利用

TVRX は、50MHz~860MHz の帯域が受信可能であるため、ここで挙げた以外にもいろいろな電波受信装置が構築できる。例えば図 9 のようにテレビのアナログ放送を行っている VHF 帯の電波を受信して AM でアナログ復調するブロックを用いることでテレビの映像を表示することができる。また、図 10 のように FSK や ASK などのデジタル復調を行うブロックを用いることで 300MHz 帯の微弱無線を用いているアクティブ RFID 受信機を構築することもできる。(アクティブ RFID 受信機は筆者の研究室で開発した独自のブロックも用いている。) このように、パーソナルコンピュータ上で実行しているソフトウェアの変更だけでさまざまなサービスを構築可能である点がソフトウェア無線の醍醐味である。

5. GNU Radio を利用した研究

GNU Radio は簡単に使用できるため、さまざまな分野の研究者が利用している。GNU Radio を用いた研究は大きく分けると、GNU Radio のアプリケーション、コグニティブ無線、無線通信品質の向上の 3 つに分けられる。

5.1 アプリケーション

GNU Radio を用いることで、ユーザの用途に合わせて電波送受信装置を安価に構築することができる。一般的に、電波送受信装置は開発コストが大きい。携帯電話や無線 LAN などの大量生産されている電波送受信装置は価格が安くなっているが、既に生産が中止されたものや、特殊用途などの理由で量産されていないものはハードウェアで実現すると膨大な開発コストがかかる。

GNU Radio の効果が大きく発揮された例として、京都大学の山本らによる電離圏観測用衛星ビーコン受信機の開発がある [8]。これまでの電離圏観測用衛星ビーコン受信機は、アナログ回路を主として構築されており、価格も数百万円と高価であった。山本らはデジタル信号処理技術の進歩とパーソナルコンピュータの性能の向上に着目し、GNU Radio と USRP を用

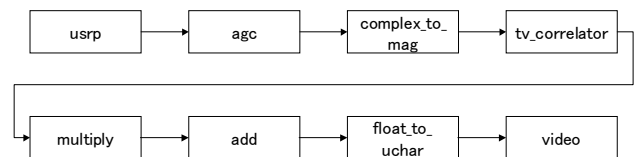


図 9 テレビのフローグラフ

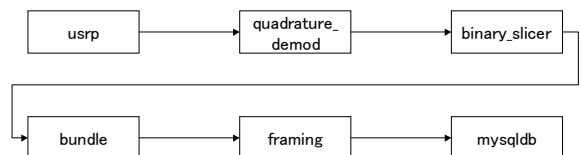


図 10 RFID 受信機のフローグラフ

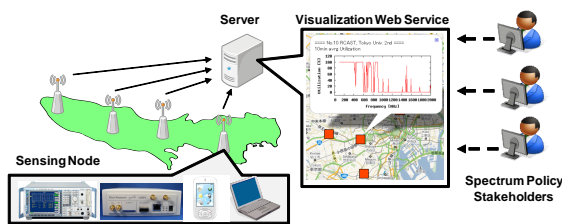


図 11 分散スペクトラムセンシング

いてオープンソースの電離圏観測用衛星ビーコン受信機を GNU Radio Beacon Receiver (GRBR) として開発した [9]。GRBR は全て Python で記述されており、C++を用いた独自信号処理ブロックの開発は行っていない。GNU Radio と USRP を用いた結果として、安価に電離圏観測用衛星ビーコン受信機を開発できただけでなく、デジタル信号処理による精度の向上も実現している。山本らによる GRBR の開発は、無線回路の専門家ではなく、ユーザが電波の受信機を開発できることのメリットが発揮された例であるといえる。

その他にも、ヨーロッパの地上デジタル放送である DVB-T [10]、無線センサネットワーク向けの IEEE 802.15.4 [11]、Texas Instruments 社の CC1000 互換通信 [11]、無線 LAN 向けの IEEE 802.11、プラズマを用いた風速計 [12]、電磁波障害 (EMI) 信号センサ [13]、電力線を使ったセンサネットワーク [14] など多くの機能が GNU Radio を用いて実現されている。ソフトウェアをダウンロードするだけでこれらの機能を利用でき、かつソースコードをスクリプト言語として読むことができれば、教育の観点からも有用であろう。

5.2 コグニティブ無線

1999 年の Joseph Mitola の論文でソフトウェア無線とコグニティブ無線が組で示されたように [5]、ソフトウェア無線とコグニティブ無線は密接に関係している。しかしながら、GNU Radio は Joseph Mitola が文献 [5] に示していたソフトウェア無線とコグニティブ無線の関係とは違った形で使われている。

現在、テレビのホワイトスペースを利用した無線通信システムという観点でのコグニティブ無線の研究が活発化している。Joseph Mitola が示したものが広義のコグニティブ無線だとすると、テレビのホワイトスペースを利用したものは狭義のコグニティブ無線だと言える。GNU Radio は、狭義のコグニティブ無線の研究開発において、研究のアイデアを実証するためのツールとして多用されている。具体的には、テストベッドやプラットフォーム [15], [16]、空き帯域や通信しているデバイスの検出 [17]~[20]、コグニティブ無線用の通信方式 [18], [21], [22]、位置情報データベース [23] などコグニティブ無線に関わる幅広いテーマで GNU Radio が用いられている。

筆者らも、分散スペクトラムセンシングシステムのプロトタイプに GNU Radio を用いている [24]。分散スペクトラムセンシングでは、ウェブサービスとスペクトラムアナライザを融合し、広域に分散された複数のスペクトラムアナライザを広域にわたる巨大な 1 つのスペクトラムアナライザとみなして電波の利用状況の見える化を行う。図 11 にシステム全体像を示

す。本システムは多数のセンシングノードとサーバによって構成される。センシングノードとしては多様な無線機や測定器を利用する。各センシングノードは電波の利用状況を測定し、測定結果をインターネットを通じてサーバへと送信する。電波の利用状況を一般に公開することで、効率的な電波政策の執行に貢献したいと考えている。

5.3 無線通信品質の向上

理論的には実現できるが、実際には実現が困難な技術というのは世の中に多く存在する。例えば理論的には $E = mc^2$ によって質量をエネルギーに変換可能であることが示されているが、工学的には質量を全てエネルギーに変換するのは非常に難しい。

同様に、無線通信においても、理論的には可能であっても工学的には実現するのが困難な技術がある。モバイル通信の分野では、キャンセレーションを中心に実際に GNU Radio を使ってアイデアを実証する研究が盛んに行われている。

GNU Radio を用いた研究で最もインパクトが大きかったのが全二重の無線通信の実証である [25]。一般的に、2 台の端末で通信を行う場合、電波では 1 つの周波数帯では片方向通信しかできない。1 つの周波数帯で端末の送信機と受信機を同時に動作させると、自端末が送信している電波が回り込んで自端末の受信機で受信し、相手端末が送信している電波を受信できないからである。自端末が自分で送信している電波は自分で知っているため、理論的にはキャンセレーションができるはずである。しかしながら、自端末で出した電波と相手端末が出した電波の差が大きいため、キャンセレーションは技術的には困難であるというのがこれまでの認識であった。文献 [25] では、アナログとデジタルのキャンセレーションを複数組み合わせることで全二重の無線通信が実現できることを GNU Radio と USRP2 を用いて実証している。

文献 [25] と同様にキャンセレーションを駆使している研究は多い。異なるタイミングで衝突した 2 つの packets を用いた ZigZag [26]、キャプチャ効果を利用したもの [27]、MIMO を利用した IAC [28]、キャンセレーションを前提とした MAC プロトコル [29] など、GNU Radio で取得した物理層の信号を用いて packets を修復する研究がなされている。

キャンセレーションのように信号レベルの情報を用いるのではなく、復調した時の情報を用いる Soft Value を用いた研究もなされている。Soft Value は、通常は 0 か 1 の 2 進数で表されたビットを、復調した時の情報を用いて 0~1 の実数としてビットを扱ったものである。Soft Value を用いて壊れた部分だけを再送する PPR [30]、複数の基地局で受信した packets を集めて Soft Value で修復するもの [31]、受信した Soft Value に応じて送信レートを変更する SoftRate [32] などの研究がなされている。

その他、1 対多の同時通信 [33]、協力通信 [34]、無線上での疑似 CSMA/CD [35]、チャンネルの状態に合わせた最適な伝送速度制御 [36]、複数の伝送速度で実現するマルチホップ通信 [37]、チャンネルの状態から packets が到達したことを予測する手法 [38]、チャンネルの情報を利用した安全な無線通信システム [39] などの研究も行われている。特筆すべきは、インターネットやモバイ

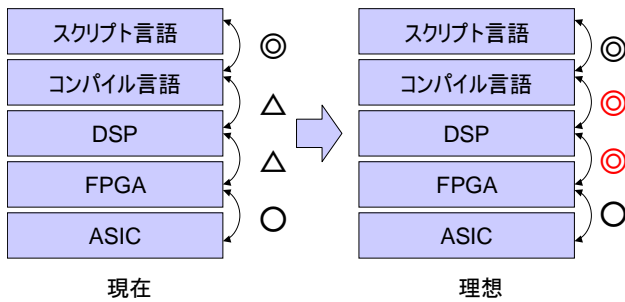


図 12 GNU Radio の課題

ルコンピューティングなどのこれまで無線の物理層をブラックボックスとして扱っていた研究コミュニティにおいて、積極的に物理層を扱っている研究が増加していることである。

6. GNU Radio の課題

GNU Radio は、研究者の間では頻りに使われるようになってきたものの、性能の低さが課題となっている。GNU Radio は使いやすさを基本コンセプトに構築されている。FPGA を用いれば性能が上がるのが明らかであるものの、FPGA を利用することで使いやすさが損なわれることは避けなければならない。

使いやすさの基本コンセプトを維持しつつ、性能を向上させるためには信号処理ハードウェアの階層化をユーザに意識させない形で実現する必要がある。図 12 に現在の GNU Radio と理想形の GNU Radio を示す。現在の GNU Radio では、スクリプト言語である Python とコンパイラ言語である C++ の密な連携に成功している。今後は、GPU (Graphical Processor Unit) 上の DSP (Digital Signal Processing) の機能や USRP が備える FPGA の機能との密な連携が必要である。パーソナルコンピュータ上でソフトウェア無線を効率的に処理する試みは他でも行われており [40]、これらの研究成果を取りこみつつ性能が向上していくことが予想される。

7. おわりに

本稿では GNU Radio について紹介した。GNU Radio は、電波送受信装置のプロトタイピング、電波を扱う研究、無線通信の教育の用途に対して非常に強力なツールである。GNU Radio はまだ発展途上であるため、現状ではまとまった情報が無いのが弱点である。GNU Radio の基本に関しては GNU Radio のウェブサイト [41] のチュートリアルで身に付く。Python 上のテクニックを身に付けたい場合にはサンプルソースコードの `benchmark_tx.py` と `benchmark_rx.py` を読み込むことで多くの情報が得られるであろう。また、静岡大学の渡辺研究室・石原研究室などの有志で情報を集めている Wiki [42] や、C++での信号処理ブロックの開発までを扱っている YRP 情報通信技術研修 [43] も参照されたい。

文 献

[1] H. Ishii and B. Ullmer, “Tangible bits: Towards seamless interfaces between people, bits and atoms,” Proceedings of the SIGCHI Conference on Human Factors in Computing Sys-

tems (CHI’97), pp.234–241, Atlanta, Georgia, March 1997.

[2] E. Blossom, “GNU Radio: Tools for exploring the radio frequency spectrum,” *Linux Journal*, vol.2004, no.122, June 2004.

[3] R.J. Lackey and D.W. Upmal, “Speakeasy: the military software radio,” *IEEE Communication Magazine*, vol.33, no.5, pp.56–61, May 1995.

[4] IEEE Communications Society, *IEEE communications magazine*, vol.33, Institute of Electrical and Electronics Engineers, Inc., May 1995.

[5] J. Mitola and G.Q. Maguire, “Cognitive radio: Making software radios more personal,” *IEEE Personal Communications*, vol.6, no.4, pp.13–18, 1999.

[6] J. Mitola, “The software radio architecture,” *IEEE Communication Magazine*, vol.33, no.5, pp.26–38, May 1995.

[7] Ettus Research LLC, <http://www.ettus.com/>.

[8] M. Yamamoto, “Digital beacon receiver for ionospheric TEC measurement developed with GNU radio,” *Earth Planets Space*, vol.60, no.11, pp.21–24, 2008.

[9] GNU Radio Beacon Receiver (GRBR), <http://www.rish.kyoto-u.ac.jp/digitalbeacon/>.

[10] V. Pellegrini, G. Bacci, and M. Luise, “Soft-DVB: A fully-software gnu radio-based etsi dvb-t modulator,” Proceedings of the Karlsruhe Workshop on Software Radios (WSR’08), pp.1–6, Karlsruhe, Germany, March 2008.

[11] T. Schmid, O. Sekkat, and M.B. Srivastava, “An experimental study of network performance impact of increased latency in software defined radios,” Proceedings of the 2nd ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization (WiNTECH’07), pp.59–66, Montreal, Canada, Sept. 2007.

[12] E.H. Matlis and T.C. Corke, “A.c. plasma anemometer for hypersonic mach number experiments,” Proceedings of the 21st International Congress on Instrumentation in Aerospace Simulation Facilities (ICIASF’05), pp.245–256, Sendai, Japan, Aug. 2005.

[13] S. Gupta, M.S. Reynolds, and S.N. Patel, “ElectriSense: Single-point sensing using EMI for electrical event detection and classification in the home,” Proceedings of the 12th ACM International Conference on Ubiquitous Computing (Ubicomp’10), pp.139–148, Copenhagen, Denmark, Sept. 2010.

[14] G. Cohn, E. Stuntebeck, J. Pandey, B. Otis, G.D. Abowd, and S.N. Patel, “SNUPI: Sensor nodes utilizing powerline infrastructure,” Proceedings of the 12th ACM international conference on Ubiquitous computing (Ubicomp’10), pp.159–168, Copenhagen, Denmark, Sept. 2010.

[15] D. Scaperroth, B. Le, T. Rondeau, D. Maldonado, and C.W. Bostian, “Cognitive radio platform development for interoperability,” Proceedings of the IEEE Military Communications Conference (MILCOM’06), pp.1–6, Washington, DC, Oct. 2006.

[16] K.E. Nolan, P.D. Sutton, L.E. Doyle, T.W. Rondeau, B. Le, and C.W. Bostian, “Dynamic spectrum access and coexistence experiences involving two independently developed cognitive radio testbeds,” Proceedings of the 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN’07), pp.270–275, Dublin, Ireland, April 2007.

[17] S. Chantaraskul and K. Moessner, “Experimental study of multi-resolution spectrum opportunity detection using wavelet analysis,” Proceedings of the IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN’10), pp.1–6, Singapore, April 2010.

[18] H. Rahul, N. Kushman, D. Katabi, C. Sodini, and F. Edalat, “Learning to share: Narrowband-friendly wideband networks,” Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication (SIGCOMM’08), pp.147–158,

- Seattle, Washington, Aug. 2008.
- [19] R. Miller, W. Xu, P. Kamat, and W. Trappe, "Service discovery and device identification in cognitive radio networks," Proceedings of the 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON'07), pp.670–677, San Diego, California, June 2007.
- [20] P. Bahl, R. Chandray, T. Moscibrodsky, R. Murty, and M. Welsh, "White space networking with Wi-Fi like connectivity," Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication (SIGCOMM'09), pp.27–38, Barcelona, Spain, Aug. 2009.
- [21] L. Yang, W. Hou, L. Cao, B.Y. Zhao, and H. Zheng, "Supporting demanding wireless applications with frequency-agile radios," Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation (NSDI'10), pp.5–5, San Jose, California, April 2010.
- [22] L. Yang, B.Y. Zhao, and H. Zheng, "The spaces between us: Setting and maintaining boundaries in wireless spectrum access," Proceedings of the 16th Annual International Conference on Mobile Computing and Networking (MobiCom'10), pp.37–48, Chicago, Illinois, Sept. 2010.
- [23] X. Feng, J. Zhang, and Q. Zhang, "Database-assisted multi-ap network on tv white spaces: Architecture, spectrum allocation and AP discovery," Proceedings of the 5th IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN'11), pp.265–276, Aachen, Germany, May 2011.
- [24] J. Naganawa, H. Kim, S. Saruwatari, H. Onaga, and H. Morikawa, "Distributed spectrum sensing utilizing heterogeneous wireless devices and measurement equipment," Proceedings of the 5th IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN'11), pp.173–184, Aachen, Germany, May 2011.
- [25] J.I. Choi, M. Jain, K. Srinivasan, P. Levis, and S. Katti, "Achieving single channel, full duplex wireless communication," Proceedings of the 16th ACM Annual International Conference on Mobile Computing and Networking (MobiCom'10), pp.1–14, Chicago, Illinois, Sept. 2010.
- [26] S. Gollakota and D. Katabi, "ZigZag decoding: Combating hidden terminals in wireless networks," Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication (SIGCOMM'08), pp.159–170, Seattle, Washington, Aug. 2008.
- [27] D. Halperin, T. Anderson, and D. Wetherall, "Taking the sting out of carrier sense: Interference cancellation for wireless LANs," Proceedings of the 14th ACM International Conference on Mobile Computing and Networking (MobiCom'08), pp.339–350, San Francisco, California, Sept. 2008.
- [28] S. Gollakota, S.D. Perli, and D. Katabi, "Interference alignment and cancellation," Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication (SIGCOMM'09), pp.159–170, Barcelona, Spain, Aug. 2009.
- [29] D. Halperin, J. Ammer, T. Anderson, and D. Wetherall, "Interference cancellation: Better receivers for a new wireless MAC," Proceedings of the 6th Workshop on Hot Topics in Networks (HotNets-VI), pp.1–6, College Park, Maryland, Nov. 2007.
- [30] K. Jamieson and H. Balakrishnan, "PPR: Partial packet recovery for wireless networks," Proceedings of the ACM SIGCOMM 2007 Conference on Data communication (SIGCOMM'07), pp.409–420, Kyoto, Japan, Aug. 2007.
- [31] G.R. Woo, P. Kheradpour, D. Shen, and D. Katabi, "Beyond the bits: Cooperative packet recovery using physical layer information," Proceedings of the 13rd ACM Annual International Conference on Mobile Computing and Networking (MobiCom'07), pp.147–158, Montreal, Canada, Sept. 2007.
- [32] M. Vutukuru, H. Balakrishnan, and K. Jamieson, "Cross-layer wireless bit rate adaptation," Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication (SIGCOMM'09), pp.3–14, Barcelona, Spain, Aug. 2009.
- [33] Z. Zhang, S. Bronson, J. Xie, and H. Wei, "Employing the one-sender-multiple-receiver technique in wireless LANs," Proceedings of the 29th IEEE Conference on Information Communications (INFOCOM'10), pp.1379–1387, San Diego, California, March 2010.
- [34] X. Zhang and K.G. Shin, "DAC: Distributed asynchronous cooperation for wireless relay networks," Proceedings of the 29th IEEE Conference on Information Communications (INFOCOM'10), pp.1064–1072, San Diego, California, March 2010.
- [35] S. Sen, R.R. Choudhury, and S. Nelakuditi, "CSMA/CN: Carrier sense multiple access with collision notification," Proceedings of the 16th Annual International Conference on Mobile Computing and Networking (MobiCom'10), pp.25–36, Chicago, Illinois, Sept. 2010.
- [36] S. Sen, N. Santhapuri, R.R. Choudhury, and S. Nelakuditi, "AccuRate: Constellation based rate estimation in wireless networks," Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation (NSDI'10), pp.12–12, San Jose, California, April 2010.
- [37] C. Yu, T. Shen, K.G. Shin, J.Y. Lee, and Y.J. Suh, "Multihop transmission opportunity in wireless multihop networks," Proceedings of the 29th IEEE Conference on Information Communications (INFOCOM'10), pp.2802–2810, San Diego, California, March 2010.
- [38] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Predictable 802.11 packet delivery from wireless channel measurements," Proceedings of the ACM SIGCOMM 2010 Conference on Data Communication (SIGCOMM'10), pp.159–170, New Delhi, India, Sept. 2010.
- [39] Z. Li, W. Xu, R. Miller, and W. Trappe, "Securing wireless systems via lower layer enforcements," Proceedings of the (WiSe'06), pp.33–42, Los Angeles, California, Sept. 2006.
- [40] K. Tan, J. Zhang, J.F.H. Liu, Y. Ye, S. Wang, Y. Zhang, H. Wu, W. Wang, and G.M. Voelker, "Sora: High performance software radio using general purpose multi-core processors," Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation (NSDI'09), pp.75–90, Boston, Massachusetts, April 2009.
- [41] GNU Radio, <http://gnuradio.org/>.
- [42] GNU Radio/USRP Wiki, <http://www.mlab.t.u-tokyo.ac.jp/~saru/usrp/>.
- [43] YRP 情報通信技術研修, http://www.yrp.co.jp/company/kenshu/top_kenshu_tsushin.html.