

分散スペクトラムセンシングにおける異種電波取得機器の統一操作方法

Common Interface for Different Measurement Equipments in the Distributed Spectrum Sensing

長縄 潤一 Juichi NAGANAWA 金 昊俊 Hojun KIM 猿渡 俊介 Shunsuke SARUWATARI 翁長 久 Hisashi ONAGA 森川 博之 Hiroyuki MORIKAWA

東京大学 先端科学技術研究センター / RCAST, The University of Tokyo

1 はじめに

周波数帯域の効率的利用に向けた技術開発や電波政策を進めるにあたっては、現在の電波利用状況を詳細に把握することが基礎となる。このような観点から、筆者らは電波利用状況を空間的に測定する分散スペクトラムセンシングの研究開発を進めている [1]。本システムでは、電波の周波数スペクトルを観測するセンシングノードを多数設置し、測定操作の指示と測定データの回収をインターネットを通じて行う。

本システムは可能な限り多くのセンシングノードを含めるために、電波取得機器の抽象化を行う無線デバイスコントローラを各センシングノードに導入することで電波取得機器の選択性を実現することを目指している。対象とする電波取得機器としては、市販の汎用スペクトラムアナライザに加えて、筆者らが開発中の超低コストスペクトラムセンサ [2] のほか、組み込みの無線 LAN カード等の既存の無線システムをスペクトル測定用に転用させたものを想定している。本稿ではこれらの多様な電波取得機器の統一の利用に向けて、電波利用状況の測定という観点から抽象化したインタフェースの定義と実装を述べる。具体的には、無線デバイスコントローラがサポートすべき共通コマンドセットを定義し、それがどのように各電波取得機器に固有の測定操作として実行されるかを、掃引式スペクトラムアナライザおよびソフトウェア無線用フロントエンドを例に示す。

2 共通コマンドセットの定義

本システムの測定対象である周波数スペクトルの時間的変化は、図 1 に示されるように周波数軸および時間軸で構成される平面上の微小領域に含まれる電力値を測定したものである。そのため周波数軸および時間軸に関して、起点、終点、および分解能を最低限の測定パラメータとして指定する必要がある。これらの測定パラメータを指定する動作をコマンドとして定義する。

まず、周波数軸上の測定パラメータを指定するコマンドとしては、周波数帯域の起点および終点を、それぞれ `FREQ_LOWER`, `FREQ_UPPER` と定める。周波数分解能は `RBW` と定める。コマンドの使用法としては、コマンド名と設定値である引数のペアとして記述する。例えば 100MHz から 1GHz までの測定を希望する際には、`FREQ_LOWER 100M` および `FREQ_UPPER 1G` のように記述する。次に時間軸上の測定パラメータを指定するコマンドを定める。測定時間の起点と終点はそれぞれ `START` および `STOP` とする。`START` および `STOP` は引数に所望の日付・時刻を指定するが、引数が指定されなかった場合はコマンドを受け取った瞬間を測定の起点あるいは終点とみなす。測定の時間間隔は `TIME_RESOLUTION` と定める。

最後に、ユーザの利便性を考えて、現在の測定パラメータの設定値を確認するコマンドとして `GET` を定義する。`GET` は引数として設定値を確認したい測定パラメータ名を指定する。また、各測定機器に詳細な指示を与えない場合をも考慮して、測定機器固有の操作を行うための `DEVICE_LIST` および `DEVICE` コマンドを用意する。`DEVICE_LIST` は、`DEVICE` コマンドで利用できる当該電波取得機器に固有のコマンドの一覧を返す。`DEVICE` の利用方法としては、アッテネータを 10dB に指定したい場合を例とすると `DEVICE ATT 10` のように引数に固有のコマンドをとる。以上に述べた共通コマンドセットの一覧を表 1 に示す。

3 各電波取得機器での実装方法

掃引式スペクトラムアナライザでの実装

図 2(a) に一般的な掃引式スペクトラムアナライザの回路構成を示す。掃引式スペクトラムアナライザは局部発振器、ミキサ、フィルタ、電力検出器によって構成されており、局部発振器の周波数を掃引しつつ無線信号を中間周波数またはベースバンドへダウコンバートして電力を検出することで周波数スペクトルを取得する。このような回路構成では、`FREQ_LOWER` および `FREQ_UPPER` は掃引の開始および終了の周波数となる。`RBW` は電力検出の前段に挿入するバンドパスフィルタの幅となる。加えて本システムでは、`RBW` に応じて掃引する際の周波数の変化させる。通常、`RBW` と掃引時の周波数の間隔 ΔF_{LO} は独立である。しかし `RBW` よりも掃引時の間隔が広い時は図 2(b) のように測定されないギャップが生じるため、本システムでは $\Delta F_{LO} \leq RBW$ を満たす ΔF_{LO} を選択する。`TIME_RESOLUTION` は 1 回の掃引に費やす時間となる。

今回の実装では、Rohde Schwarz 社の汎用スペクトラムアナライザ FSL 6 を用いた。FSL 6 ではソフトウェアからの制御を実現するためのライブ

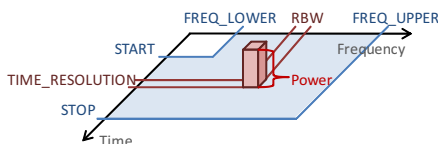


図 1 周波数-時間平面上の測定パラメータの意味

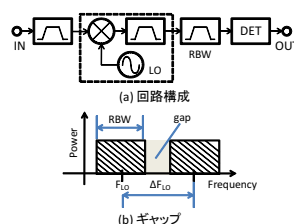


図 2 掃引式スペクトラムアナライザによる測定

表 1 共通コマンドセット

コマンド名	意味
<code>FREQ_LOWER</code>	測定の上限周波数を指定する
<code>FREQ_UPPER</code>	測定の下限周波数を指定する
<code>RBW</code>	周波数分解能を指定する
<code>TIME_RESOLUTION</code>	時間分解能を指定する
<code>GET</code>	測定パラメータの設定値問い合わせ
<code>START</code>	測定開始
<code>STOP</code>	測定中止
<code>DEVICE_LIST</code>	機器固有の操作一覧を問い合わせ
<code>DEVICE</code>	機器固有の操作を指示する

ラリである `RsSpecAn` が提供されている。`RsSpecAn` では、掃引の開始および終了の周波数を `ConfigureFrequencyStartStop` 関数で、バンドパスフィルタの幅を `COUPLING_RBW` フラグを立てた `ConfigureSweepCoupling` 関数で、1 回の掃引の時間を `COUPLING_SWEEP_TIME` フラグを立てた `ConfigureSweepCoupling` 関数でそれぞれ設定した。周波数の掃引間隔は、`RsSpecAn` では直接指定する代わりに、測定帯域幅/`RBW` で決まる掃引点数を `ConfigureSweepPoints` 関数で設定することで間接的に変更した。`DEVICE` コマンドの例としては、アッテネータの指定を `ConfigureAttenuation` 関数で行った。

ソフトウェア無線用フロントエンドでの実装

ソフトウェア無線用フロントエンドへの無線デバイスコントローラの実装にあたっては、サンプリングによって得られる信号がベースバンド信号であるため、FFT による周波数領域への変換を組み込む必要がある。ただし、FFT で得られる周波数スペクトルの帯域幅はベースバンド信号のサンプリング周波数 F_S で制限されるため、図 3(a) に示されるようにフロントエンドの中心周波数 F_C の掃引と FFT を繰り返して広帯域にわたる周波数スペクトルを取得する。FFT を用いる場合、`FREQ_LOWER` (F_{LOWER}) および `FREQ_UPPER` (F_{UPPER}) はフロントエンドの中心周波数を変化させる際の下限 F_{CL} と上限 F_{CU} を決定づけ、それぞれ次式となる。

$$F_{CL} = F_{LOWER} + F_S/2 \quad (1)$$

$$F_{CU} = [(F_{UPPER} - F_{LOWER})/F_S] \cdot F_S - F_S/2 + F_{LOWER} \quad (2)$$

`RBW` は F_S/RBW で決まる FFT 点数となる。`TIME_RESOLUTION` は、分割された各周波数区間あたりの測定時間を決定づけ、各区間あたり $\frac{TIME_RESOLUTION}{[(F_{UPPER}-F_{LOWER})/F_S]}$ となる。

今回の実装にあたっては、ソフトウェアに GNU Radio, フロントエンドに USRP2 および DBSRX を利用した。フロントエンドの中心周波数は、GNU Radio に標準で付属する信号処理ブロックである `usrp2_source_32fc` の持つ `set_center_freq` 関数で設定した。FFT 点数は、FFT 計算を実行するブロックである `fft_vcc` の初期化時のパラメータとして設定した。さらに、USRP2/DBSRX は FFT 結果が図 3(b) に示されるように歪む特性があるため、歪み補正を実現する信号処理ブロックを作成した。具体的には、25MHz の帯域を 5MHz, 15MHz, 5MHz の区間に分け、それぞれの区間での歪みを近似する二次関数をあらかじめ求めておき、FFT 結果から歪み分を減らすことで補正を実現した。

4 おわりに

本稿では、分散スペクトラムセンシングシステムにおいて、電波取得機器の種類に依らない共通コマンドセットを定義し、スペクトラムアナライザ、ソフトウェア無線用フロントエンドでの実装を示した。

謝辞 本研究は総務省からの委託研究「ユビキタスサービスプラットフォーム技術の研究開発」の成果である。

参考文献 [1] 長縄他, “広範囲・高密度に展開可能な分散スペクトラムセンシングの設計と実装,” 信学技報 RCS2009-342, 2009. [2] 金他, “分散スペクトラムセンシングに向けたセンサノードの初期的検討,” 信学ソ大, 2009.

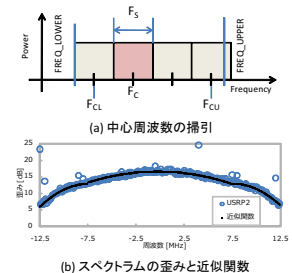


図 3 ソフトウェア無線用フロントエンドによる測定