

目次

目次.....	1
第1章 序論.....	2
第2章 理論.....	3
2.1 アンテナ.....	3
2.2 放射パターン.....	4
2.3 アイソトロピックとダイポールアンテナ.....	7
2.4 電界領域.....	7
2.5 バンド幅.....	9
2.6 偏波.....	9
2.7 測定するアンテナの特製.....	10
第3章 円偏波合成開口レーダの測定システム.....	13
第4章 プログラムの説明	
第5章 実験.....	25
第6章 シミュレーションデータとの比較.....	30
第7章 まとめと今後の課題.....	32
参考文献.....	33

謝辞

付録

A ソースコード

1 序論

宇宙からの地球観測は現在ではごく普通のことになってきている。衛星からの地球観測では光学センサー，赤外センサーなどさまざまなセンサーの種類が人工衛星に搭載されてきたが，特に近代，その高解像度やコヒーレンス特性で注目になってきているのは合成開口レーダ (Synthetic Aperture Radar, SAR) である。

当研究室では無人航空機(Unmanned Aerial Vehicle, UAV) および衛星搭載用の次世代円偏波合成開口レーダ(Circularly Polarized Synthetic Aperture Radar, CP-SAR)システムを開発している。レーダや通信分野において，アンテナは欠かせないものである。アンテナの製造はシミュレーションデータに基づいて行われるが，実際に製造されたアンテナの特性がシミュレーション通りになるかどうかを確認するために測定を行う必要がある。しかし，アンテナの放射パターンを測定するにはあらゆる角度からの測定を行わなければならないが，角度を少しずつ回転して手動で測定するという方法で行うのは非常に手間がかかる上に，ずれや人間の誤作動により精度が低くなる。そのため，これらの測定は精密機械で回転させ，自動的に測定したデータを記録させることで効率良く行うことができる。

本研究で開発されたアンテナをはじめ SAR の測定システムは，回転台およびそのコントローラ，標準ダイポールアンテナ，円偏波信号を放射する左旋と右旋のコニカルログスパイラルアンテナ，測定装置であるネットワークアナライザ，および全体操作を制御し測定結果を記録する計算機から成り立ち，これらの組み合わせで対象アンテナの円偏波特性を測定する。主な測定対象としては利得と軸比であり，これらの結果から放射パターンをグラフ化することもでき，またアンテナの性質は良いかどうかを知ることができる。

このシステムの一番の利点は，従来手動で行った回転させて測定するという操作を自動化し，各 1 度ずつの測定データを自動的に測定し記録するところである。これによって 1 回の測定時間は 6 分以内で納められ，また測定結果を Excel などですぐ加工できる csv の形として保存されるため結果を短時間で見るることができる。また，GUI (Graphical User Interface) も備えてあるので，命令コードなどを覚える必要がなく，コンピュータにそれほど詳しくない人にも比較的短時間の訓練で操作できるようになる。

このシステムは将来，円偏波合成開口レーダアンテナの測定に利用する予定である。また，近傍界から遠方界変換を導入することにより寸法制限のある電波無響室内でも低周波数アンテナ，または大型およびアレーアンテナの特製を測定することも可能である。

2 理論

2.1 アンテナ

アンテナは電波を発生，または受信するための媒体であり，通常は金属の細い棒またはワイヤーの形として見られる．マイクロストリップアンテナなどのように金属の板として作られているアンテナもある．

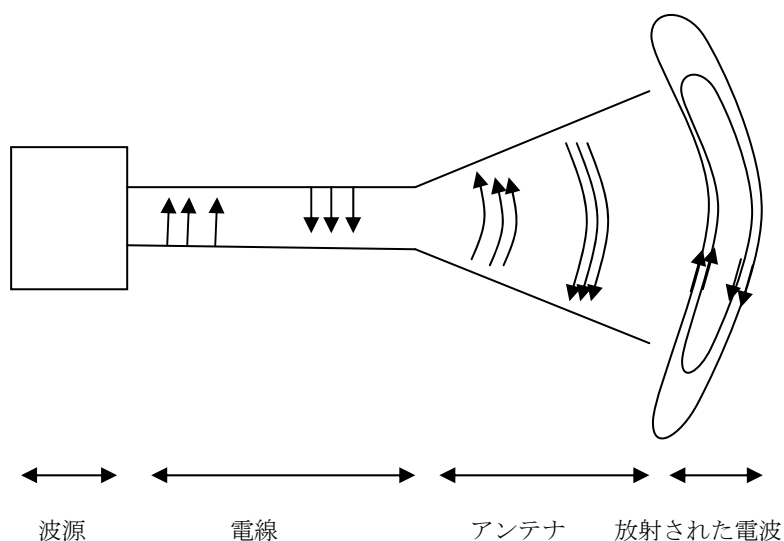


図 2.1 基本的なアンテナの例

アンテナから電波が放射される条件としては，アンテナ上で流れる電流が時間に対して変化する必要がある，つまり，電荷の速度変化がなければ電波が放射されない．電荷の速度変化を作る条件として電線が曲がったり，折れたり，不連続だったり，切られたりする必要がある．また，電線内の電荷が調波振動すれば電荷の加速度も生じる．交流の電圧を用いて電荷を調波振動させることができる．

つまり，

- 電荷が移動しなければ，電流が存在しなくなり，電波は放射されない
- 電荷が一定速度で移動すれば，
 - 電線が直線状態かつ無限に長ければ電波は放射されない
 - 電線が曲がったり，折れたり，不連続だったり，グラウンドされたり，切られたりしたら電波が放射される
- 電荷が調波振動すれば直線の電線上でも電波が放射される

アンテナの放射原因を定性的に説明すれば，図 2.2(d)のようなあるグラ

ウインドされた電線にパルス電源を接続することで想像することができる。電線が最初に電源からエネルギーをもらった時、電線内の電荷（自由電子）は電源のエネルギーにより移動し始める。電荷が電源により加速するとき、または端からの反射により減速するとき、電界が両端と電線の間から放射される。パルスが短ければ、または時間幅が少なければより周波数幅が広い強い放射が起こる。また、連続的に調波振動する電荷は理想的に単周波数の放射を発生し、その周波数は振動の周波数によって決まる。

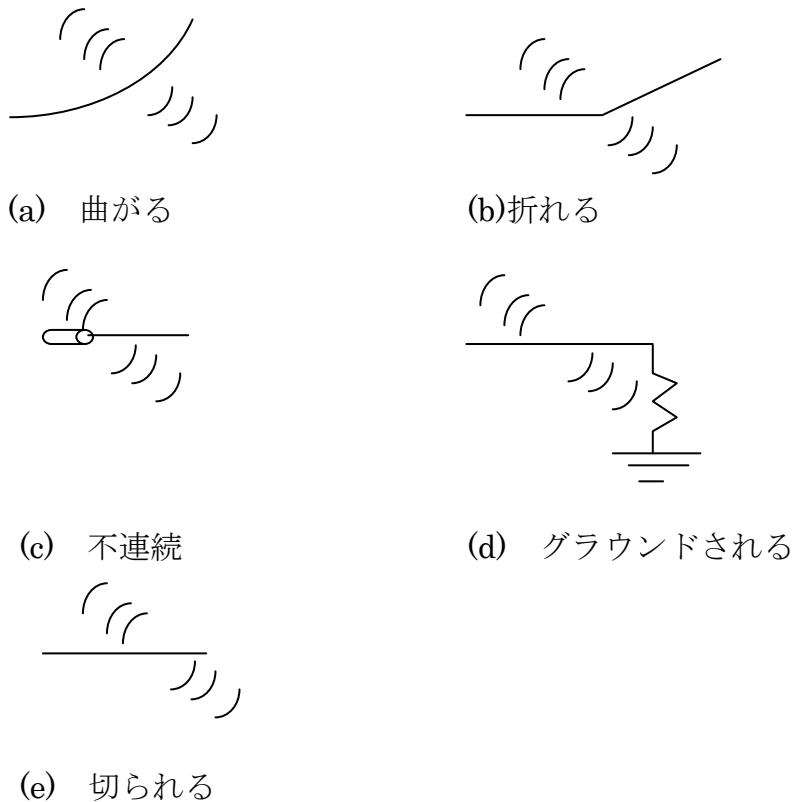


図 2.2 電波を放射する電線

2.2 放射パターン

アンテナの放射パターン、またはアンテナパターンはアンテナの放射特性を空間座標の関数として表す数式または図形である。基本的に放射パターンは遠方界領域の放射パターンとして定義されていて、方向座標の関数として表される。放射の特性は電力束密度、放射強度、電界強度、方向性、位相及び偏波を含む。特に重要な特性は 2 次元または 3 次元空間での放射エネルギー分布であり、任意の一定半径の表面に沿って観察者の位置関数としてグ

ラフ化される。任意の一定半径での受信電界（磁界）の図は電界強度パターンといわれる。一方、ある一定半径の電力分布は電力強度パターンといわれる。

多くの場合、電界と電力パターンは最大値に対して正規化される。また、電力パターンは通常、対数目盛またはデシベル（dB）単位で図示される。この目盛の特徴は非常に小さい値を持つマイナーローブを強調することができることである。

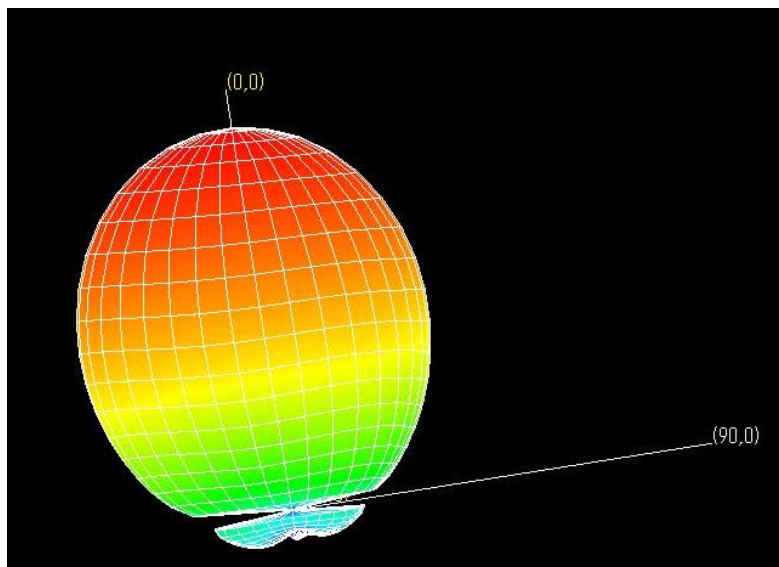


図 2.3 放射パターンの例

パターンが最大電力の半分になる点を見つけるために、パターンの最大値に対して、以下の点を決める。

- 電界パターンでは最大値の 0.707 倍となる点
- 電力パターン（線形目盛）では最大値の 0.5 倍となる点
- 電力パターン（対数目盛）では最大値の -3dB となる点

二つの最大電力の半分となるところの間は HPBW（Half Power Beamwidth, 電力半値ビーム幅）といわれる。

実際、3次元のパターンは2次元パターンの列として測定され記録されるが、多くの場合、特定の ϕ 値に対していくつかの θ 関数のパターン及び特定の θ 値に対していくつかの ϕ 関数のパターンは最も必要であり重要な情報を提供する。

放射パターンの部分はローブといわれ、メインまたは主ローブ及び副ローブとして分類される。ローブは放射パターンの中にある放射強度の強い部分であり比較的弱い放射強度の領域に囲まれる領域である。主ローブは最大放射の方向にあるローブである。分割ビームアンテナのようなアンテナには

1つ以上の主ローブが存在する可能性がある。副ローブは主ローブ以外のローブであり、サイドローブとバックローブに分類されている。サイドローブは主ローブの方向以外のローブを言う。一方、バックローブは主ローブに対して180度の方向を持つローブ、つまり反対方向にある副ローブを言う。

副ローブは通常望ましくない方向にある放射を表すためできる限り抑えられることが多い。特にサイドローブは副ローブの中で最も強い放射であるため、通常-20dB以上のサイドローブは望ましくない。

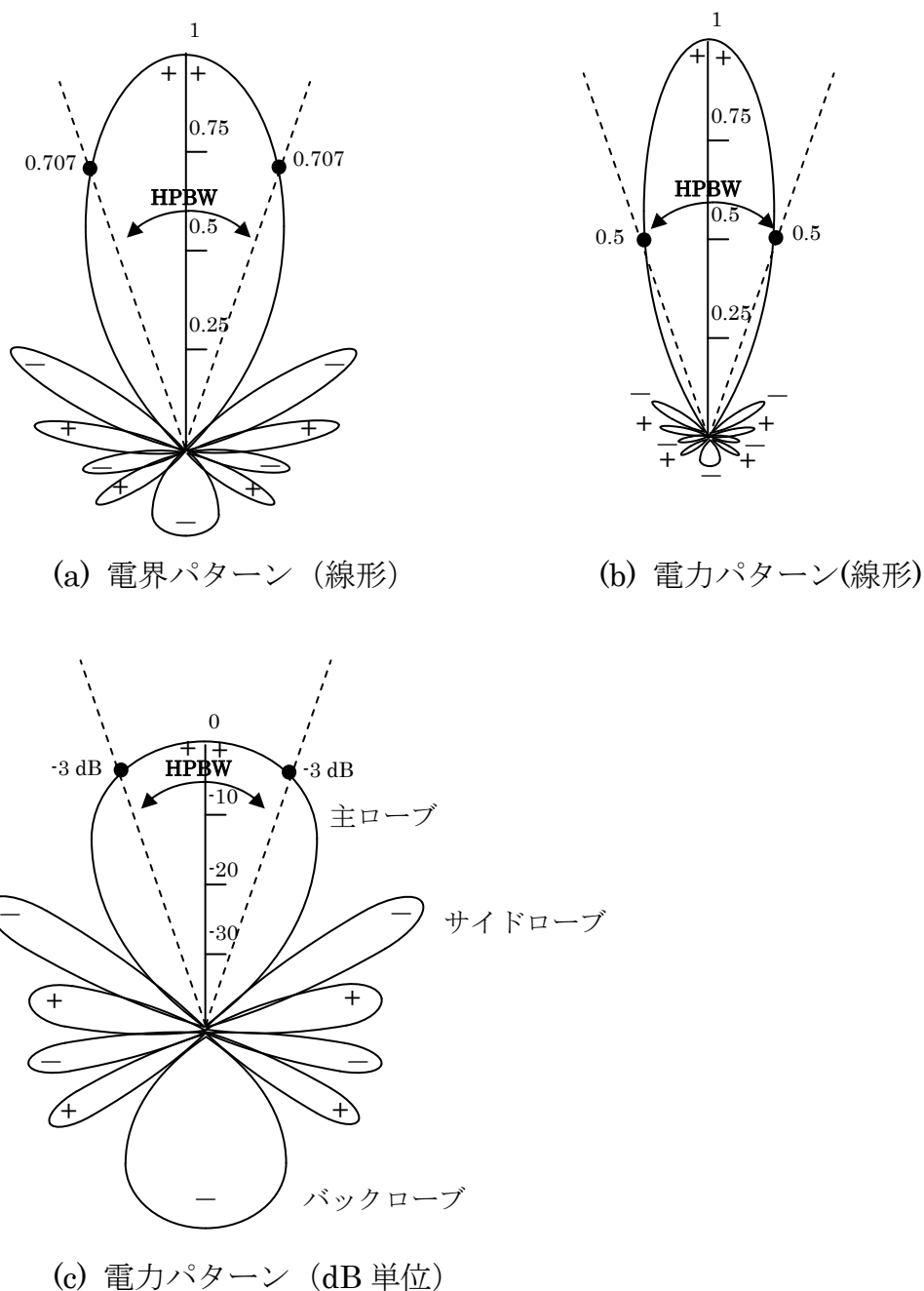


図 2.4 $d=0.25\lambda$ の間隔をもつ 10 要素線形アンテナから図形された二次元正規化電界

パターン（線形）、電力パターン（線形）及び電力パターン（dB 単位）

2.3 アイソトロピックとダイポールアンテナ

アイソトロピックアンテナは全方向に対して同じ放射強度を持つ理想のロスレスアンテナとして定義される。実際に存在することがあり得ないが、測定されるアンテナの方向性を表すための基準として使われることが多い。指向性アンテナは、ある方向に対してほかの方向より効率的に電波を放射または受信する特性を持つアンテナとして定義される。

ダイポールアンテナは理論的に最も単純なアンテナであり、2本のワイヤから作成することができる。電流の強度が真中で最大の値から一意的に両端でゼロになる。ダイポールアンテナは標準アンテナとして使われることが多い。半波長ダイポールアンテナはアイソトロピックアンテナに対して1.64倍、つまり2.15dBの利得を持つ。

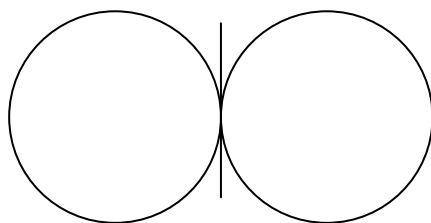


図 2.5 ダイポールアンテナ（中心にある棒）の放射パターン

2.4 電界領域

アンテナ周りがある領域は通常リアクタンス近傍界(reactance near field)、放射近傍界(radiance near field)またはフレネル近傍界及び遠方界(far field)またはフラウンホーファー領域という3つの領域に区分される。これらの領域の間に放射特性が段階的に変わることがないが、一つの領域内とほかの領域内での放射特性は異なる。これらの領域の境界は一意ではないが、さまざまな条件が確立されてきていてこれらの領域を区別するために一般に使用される。

リアクタンス近傍界領域は、アンテナのすぐ近くにあるリアクタンス性の電界が優位をしめる近傍界領域として定義される。通常この領域の外部境界はアンテナ表面から

$$R < \frac{0.62\sqrt{D^3}}{\lambda} \quad (2.1)$$

の距離にあるとされる。ここで、 λ は波長を、 D はアンテナの最長の寸法を

表す. 非常に短いダイポールまたはそれと等価する放射物の場合, 外部境界は通常アンテナ表面から $\lambda/2\pi$ の距離にあるとされる.

放射近傍界 (またはフレネル) 領域はリアクタンス近傍界と遠方界領域の間にある領域であり, 放射電界が優位をしめる及び角電界分布はアンテナからの距離に依存する領域として定義される. アンテナの最大寸法が波長に対して大きくない場合, この領域は存在しない可能性がある. この領域の境界は,

$$0.62\sqrt{\frac{D^3}{\lambda}} \leq R < \frac{2D^2}{\lambda} \quad (2.2)$$

の距離内にある. ここで, λ は波長を, D はアンテナの最長の寸法を表す. この条件は最大位相差が $\pi/8$ に基づく.

遠方界 (またはフラウンホーファー) 領域は角電界分布がアンテナからの距離にほとんど依存しないアンテナの電界領域として定義される. この領域は, アンテナから

$$R > 2D^2/\lambda \quad (2.3)$$

の距離に存在するとされる. この領域内では送信される電波が平面波として扱われる. また, 経験的にアンテナの寸法が小さいときに 10λ 以上の距離を置ければ遠方界領域になる.

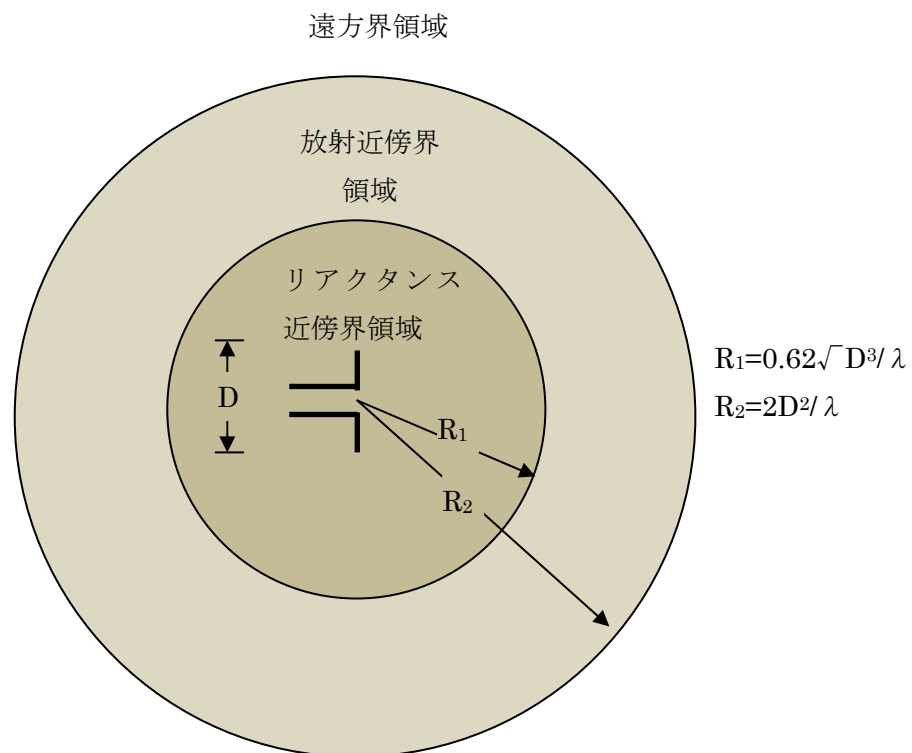


図 2.6 アンテナ周りの電界領域

2.5 バンド幅

アンテナのバンド幅（帯域幅）は、アンテナの性能が、決められた特性において、特定された標準を満たしている周波数帯域の幅をいう。つまり、中心周波数（ダイポールにおいては通常共振周波数）から両端までの周波数範囲であり、その周波数範囲内においてアンテナ特製（入力インピーダンス、パターン、ビーム幅、偏波、サイドローブ利得、ビーム方向、放射効率など）が中心周波数のアンテナ特製から許容範囲内の値を持つ。

この特性は、アンテナの受信周波数領域に関係している。特に合成開口レーダの分野では、線形的に周波数が増加するチャープパルスを使用するため、特性条件上で広いバンド幅を持つアンテナが望ましい。

2.6 偏波

ある方向のアンテナ偏波はそのアンテナによって送信された（放射された）電波の偏波として定義される。方向が特に記述されなければ、一般に偏波は最大利得の方向の偏波とされる。実際は、放射されたエネルギーの偏波はアンテナの中心からの方向によって変わり、パターンのあらゆる所に異なる偏波を持つ可能性がある。

放射された電波の偏波は、時間的に変わる電界ベクトルの方向と相対大きさを表す電波の特性として定義されて、特に、進行方向から見た固定された位置にあるベクトルの極値を時間の関数として図示され表される。このとき、偏波は瞬時電界を表すベクトルの先端で描かれた曲線となる。[1]

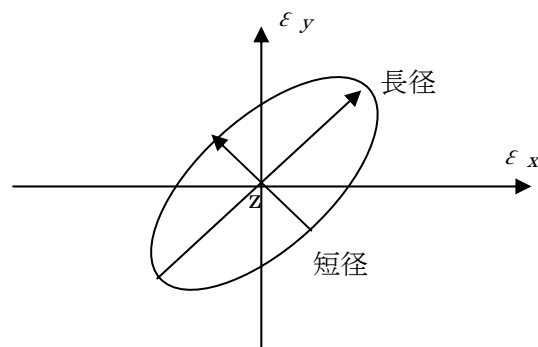


図 2.7 楕円の偏波

偏波は直線、円及び楕円として分類することができる。ある点での電界を表すベクトルが時間関数として常に直線上にあるならその電界は直線偏

波といわれる。一般には電界が描いた図形は楕円であり、そしてその電界は楕円偏波といわれる。直線と円偏波は楕円偏波の特例であり、楕円が直線及び円になるときに得られる。また、電界の図形は時計回り (Clockwise, CW) か反時計回り (Counter-clockwise, CCW) に描かれるが、時計回りの回転は右旋偏波 (Right-Handed Circular Polarization, RHCP) として、反時計回りの回転は左旋偏波 (Left-Handed Circular Polarization, LHCP) として決められる。

2.7 測定するアンテナの特製

本研究で測定されるアンテナの特性はリターンロス (反射損失) または S パラメータ, 入力インピーダンス (input impedance), VSWR (Voltage Standing Wave Ratio, 電圧定在波比), 利得 (gain) 及び円偏波アンテナの特性である軸比 (axial ratio) である。

リターンロス (return loss) は高周波回路の 1 つのポートにおいて、入力電力対反射電力の割合である。通常 dB の単位で表わされる。性能の良いアンテナは使用周波数およびそのバンド幅の間において非常に小さいリターンロスを持つ。通常 -10dB 以下の値が求められる。

ポートに入力された電力の平方根 (入射波, incident power wave) を a_1 とし、そのポートから反射される電力の平方根 (反射波, reflected power wave) を b_1 とすると、リターンロス RL は

$$RL = -20 \log_{10} \left| \frac{b_1}{a_1} \right| \quad (2.3)$$

となる。

入力インピーダンス (input impedance) はある電気回路または電気機器の入力側から見た等価インピーダンスである。この回路をほかの回路またはケーブルに接続するとき、等価インピーダンスが異なるとき電力が完全に伝わらなくなり、ロスが生じる。通常アンテナの分野では使用周波数において入力インピーダンス 50Ω が望ましい。

VSWR (Voltage Standing Wave Ratio) または電圧定在波比は交流の伝送路における進行波と反射波の関係を示す数値であり、この値が 1 の場合に給電線からアンテナに供給された電力がすべて放射され、それ以外の場合には電力の一部が給電線に反射波として分布し、有効に放射されなくなる。一般に VSWR は 1.5 以下が理想であり、3 以下が実用上の限界とされている。VSWR 1.5 でアンテナの電力効率は 96% であり、3 で効率は 75% である。今回測定されたアンテナは使用周波数において VSWR を 2 以下に抑えられ

た.

利得 (gain) は, アンテナが放射した表面の電力とアイソトロピックアンテナが放射した表面の電力の割合を言う. 数式では,

$$G = \frac{\left(\frac{P}{S}\right)_{ant}}{\left(\frac{P}{S}\right)_{iso}} \quad (2.4)$$

と利得を求めることができる. ここで, P は受信電力であり, S は表面積である. ここで放射するアンテナが同じものなので表面も同じになり, またアイソトロピックアンテナの変わりに半波長ダイポールアンテナを使用するので, 利得は

$$G = 1.64 \frac{P_{ant}}{P_{\lambda/2dipole}} \quad (2.5)$$

となる. ここで P_{ant} は対象アンテナの受信電力であり, $P_{\lambda/2dipole}$ は半波長ダイポールアンテナの受信電力である. さらに, dB 単位で計算するとしたら,

$$G(dB) = 10 \log |G| = 2.15 + 10 \log \frac{P_{ant}}{P_{\lambda/2dipole}} \quad (2.6)$$

となる.

軸比 (axial Ratio) は図 2.7 に示した楕円偏波の長径と短径の割合である. 直線偏波において, 偏波の短径が 0 になり, 軸比が無限になる. 一方, 円偏波において, 偏波の短径と長径は同じ長さを持つため, 軸比は 1 となる. つまり, 軸比は,

$$|R| = \frac{\text{長径}}{\text{短径}} \geq 1 \quad (2.7)$$

になる. 測定するとき, 楕円の長径は対象アンテナと同じ偏波をこの数式を書きかえると,

$$|R| = \frac{E_{max}}{E_{min}} \quad (2.8)$$

で計算することができる。ここで、 E_{max} は最大電界であり、 E_{min} は最小電界である。

測定するときに、対象アンテナを LHCP と RHCP のスパイラルログコニカルアンテナから電波を放射され、 E_{max} と E_{min} を

$$E_{max} = P_L + P_R \quad (2.9)$$

$$E_{min} = P_R - P_L \quad (2.10)$$

として計算される。この定義を使用することにより、軸比の値は、

$$R = \frac{P_L + P_R}{P_R - P_L} \quad (2.11)$$

として求めることができる。この定義では R が負の値になり得る。 R が負の場合、対象アンテナは LHCP であり、 R が正の場合、対象アンテナは RHCP である。

また、dB 単位で計算するとしたら、

$$R(\text{dB}) = 20 \log |R| \geq 0 \quad (2.9)$$

として計算することができる。

3 円偏波合成開口レーダの測定システム

今回構築したアンテナ自動測定システムは、航空機および衛星搭載用の円偏波合成開口レーダアンテナの特製を測定することが構築の目的とする。そのため、通常のアンテナの特性であるリターンロス、入力インピーダンス、VSWR、利得などを測定するのみならず、円偏波アンテナの特有性質である軸比も高精度及び高速で測定しなければならない。

構築したシステムは、電波無響室内にある回転台およびその制御用コントローラ、アンテナの受信電力を測定するネットワークアナライザ及び測定データを記録する計算機を連携させる。

アンテナ自動測定システムの概略図は図 3.1 に示す。

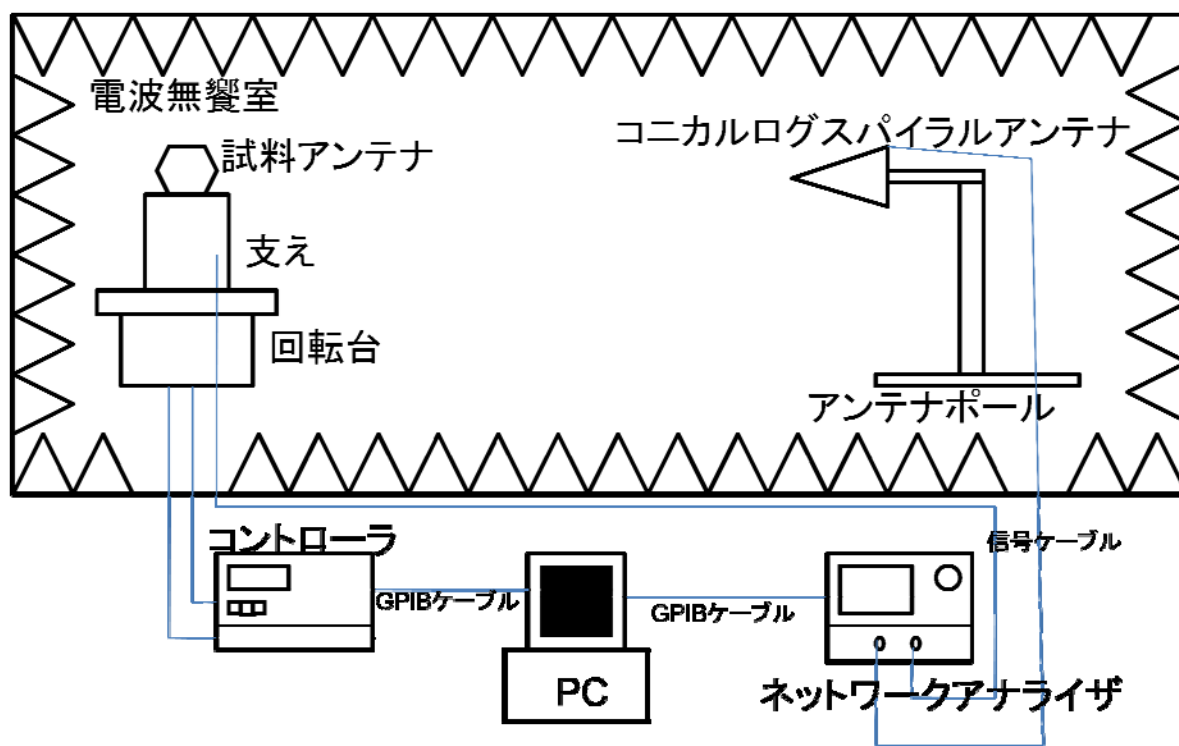


図 3.1 システムの概略図

回転台は対象アンテナ（AUT、Antenna Under Test）を乗せるところであり、あらゆる角度で測定を行うために自由に回転することができる。この回転を制御するには、回転台コントローラを用いる。このコントローラの精度は 0.1 度まで精密に回転することができる。対象アンテナと床から距離を置くために支えを回転台の上に乗せる。この支えはアクリル製の円柱であり、その中には発砲スチロールの円柱がある。アクリルは固定されるが、発砲スチロールの部分は自由に上下させることができ、アンテナの高さを調整することもできる。

しかし、この発砲スチロール部を上に移動させるとぶれやすくなり、回転台が回転するとき振動が大きくなるためこの部は最低の高さに調整した。支えの上に対象アンテナを置きやすくするため円盤を置いてその上に対象アンテナを置く。またこの円盤にもダイポールを入れるための溝を作成しダイポールのホルダーにもした。

対象アンテナに電波を送るのはコニカルログスパイラルアンテナである。このコニカルログスパイラルアンテナは、円錐型アンテナであり、円偏波の電波を出すことができる。左旋(LHCP)と右旋(RHCP)用のアンテナがあり、測定するときに適切なアンテナを利用しなければならない。このアンテナはネットワークアナライザと接続され、送信電波はネットワークアナライザからコニカルアンテナへ伝わり放射される。

正しい結果を得るために対象アンテナはコニカルアンテナの真正面に置かなければならない。コニカルアンテナの高さと横位置を調整するためにアンテナポールを利用し、コニカルアンテナの中心がアンテナの重心の真正面にあることを確認するためにレーザレベルと距離測定装置を用いた。レーザレベルはコニカルアンテナの後ろに置き、レーザレベルの垂直レーザを見ることによりコニカルアンテナの横位置は対象アンテナの横位置と一致するかがわかる。距離測定装置はアンテナポールの高さと対象アンテナの高さを確認するために利用した。また、回転台、アンテナポールが傾いているかどうかを確認するためにライトレベルを用いた。

今回の測定目的は遠方界 (far field) のアンテナ特性を測定することなので、対象アンテナとコニカルアンテナの間に 10λ 以上の距離を置かなければならない。しかし、距離が遠すぎると逆に受信電力が落ちて、ノイズ成分が強くなってしまう。対象アンテナの使用周波数が 2.5GHz あたりにあるので λ は 0.12m になり、最低距離は 1.2m である。実験したとき対象アンテナとコニカルアンテナの距離は 2m であった。

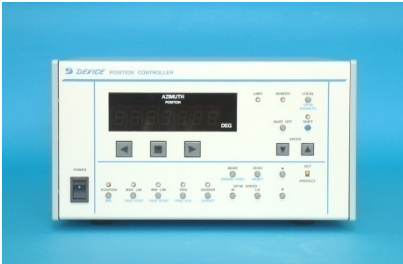
標準アンテナとして $1/4$ 波長ダイポールアンテナを用いる。このダイポールの利得はアイソトロピックアンテナに対して 2.15dB の利得を持つ。

システムに利用されたソフトウェアは表 3.1 に示す。システムに利用された機械などのハードウェアは表 3.2 に示す。

表 3.1 使用ソフトウェア

OS	Microsoft Windows XP Professional Edition
開発ツール	Microsoft Visual C# 2005
ライブラリ	LabView 8.0
GPIB ドライバー	National Instrument ドライバー

表 3.2 仕様ハードウェア

<p>回転台</p>	<p>デバイス社 DM33121AV1 アジ マステーブル</p>	
<p>回転台コン トローラ</p>	<p>デバイス社 DM3152HV1 ポジ ションコントローラ 精度 0.1 度</p>	
<p>アンテナポ ール</p>	<p>デバイス社 D2221C アンテナ ポール</p>	

アンテナアダプタ	デバイス社 ADP0017 ログスパイラル用アンテナアダプタ	
送信アンテナ	ETSL 社 コニカルログスパイラルアンテナ 3102(LHCP) と 3102L(RHCP)	
受信アンテナ (対象アンテナ)	マイクロストリップパッチアンテナ 使用周波数 2.5025GHz 波長 0.12m	
標準アンテナ	Anritsu 社 MA5612 シリーズ ダイポールアンテナ Anritsu 社 MP651 シリーズ ダイポールアンテナ	
支え	アクリル円柱 (中心には発砲スチロール円柱) の支え	

アンテナケーブル	発砲スチロール円盤	
測定装置	Agilent 社 E5062A ネットワークアナライザ	
傾向測定装置	KOD 社 L-120Q ライトレベル(液体)	
キャリブレーションキット	Agilent 社 85033E 3.5mm キャリブレーションキット	
レーザーレベル	レーザーテクノ 高精度レーザーレベル LTK-3001	

<p>距離測定装置</p>	<p>Leica Geosystems 社 Leica DISTO D3 精度 1mm</p>	
<p>データケーブル</p>	<p> GPIB ケーブル (2本)</p>	
<p>信号ケーブル</p>	<p> N型ケーブル (2本)</p>	
<p>アンテナホルダー</p>	<p> アクリル製スタンドを加工する</p>	
<p>記録用計算機</p>	<p>DELL 2.4GHz Intel Pentium 4 メモリ 3GB ハードディスク 500GB</p>	

このシステムを利用して、測定できるアンテナの特製は、

- アンテナのリターンロス(Sパラメータ)
- アンテナの VSWR (Voltage Standing Wave Ratio)
- アンテナの入力インピーダンス
- 周波数に対しての利得
- 周波数に対しての軸比
- 角度に対しての利得
- 角度に対しての軸比
- バンド幅, 放射パターン等

リターンロス, 入力インピーダンスを測定するときに, ネットワークアナライザをあらかじめ較正しなければならない. 較正に利用される道具は Agilent 社の 85033E キャリブレーションキットである. 較正する対象はオープン, ショート, 及びロード接続という 3 通りの状態をキャリブレーションキットに合わせることにより装置の誤差を減少させる.

利得を測定するには, 対象アンテナと同じ偏波 (左旋か右旋) のスパイラルアンテナで放射し受信電力を測定する. その後, 対象アンテナを標準アンテナ (ダイポール) に変えて再び受信電力を測定する. 対象アンテナの受信電力とダイポールアンテナの受信電力の割合 (dB 単位であれば引き算するだけでよい) を求めることによりアンテナの利得を求めることができる. このとき, 対象アンテナとダイポールアンテナは同じ条件で測定するので較正する必要がない. 通常測定された利得は, アンテナの使用偏波の利得を測定する. 左旋のアンテナは左旋の利得, 右旋のアンテナは右旋の利得を測定する.

また, 軸比を測定するには, 同じ対象アンテナを左旋と右旋で測定し左旋と右旋の割合を求めて軸比を計算する.

周波数に対しての利得または軸比を測定するには, アンテナをある角度に固定され, ネットワークアナライザから出た信号を決められた最低周波数値から最大周波数値まで決められた間隔で変化させ (スイープ), 各周波数においての受信電力を記録する.

角度に対しての利得または軸比を測定するには, 周波数を固定させてアンテナを 0 度から 359 度まで回転させ, 決められた角度の間隔で測定を行う.

このシステムは,

- 決められた間隔ごとに回転台を回転・停止させてから測定を行う方式
- 連続的に回転台を回転しコントローラのトリガーにより決められた間隔ごとにネットワークアナライザが信号を出力し測定する方式

という 2 通りの測定方式を行うことができる。

前者の方式は確実に決められた角度で測定することができるが、回転・停止の繰り返しを行うために測定時間が長くなり、また不安定の支えで測定を行えば支えが激しく振動し、正しい測定結果が得られない。この測定方式で測定を行うと、10 分以上の測定時間がかかってしまう。

後者の方式は測定された角度にずれが生じてしまう可能性があるが、測定時間が早く連続的に回転するので支えの振動を抑えることができる。この測定方式で測定を行うと、6 分以内の測定時間で測定が終了した。実験の時は後者の方式で測定を行った。

これらの方式の流れを表す流れ図は以下の図 3.1 に示す。

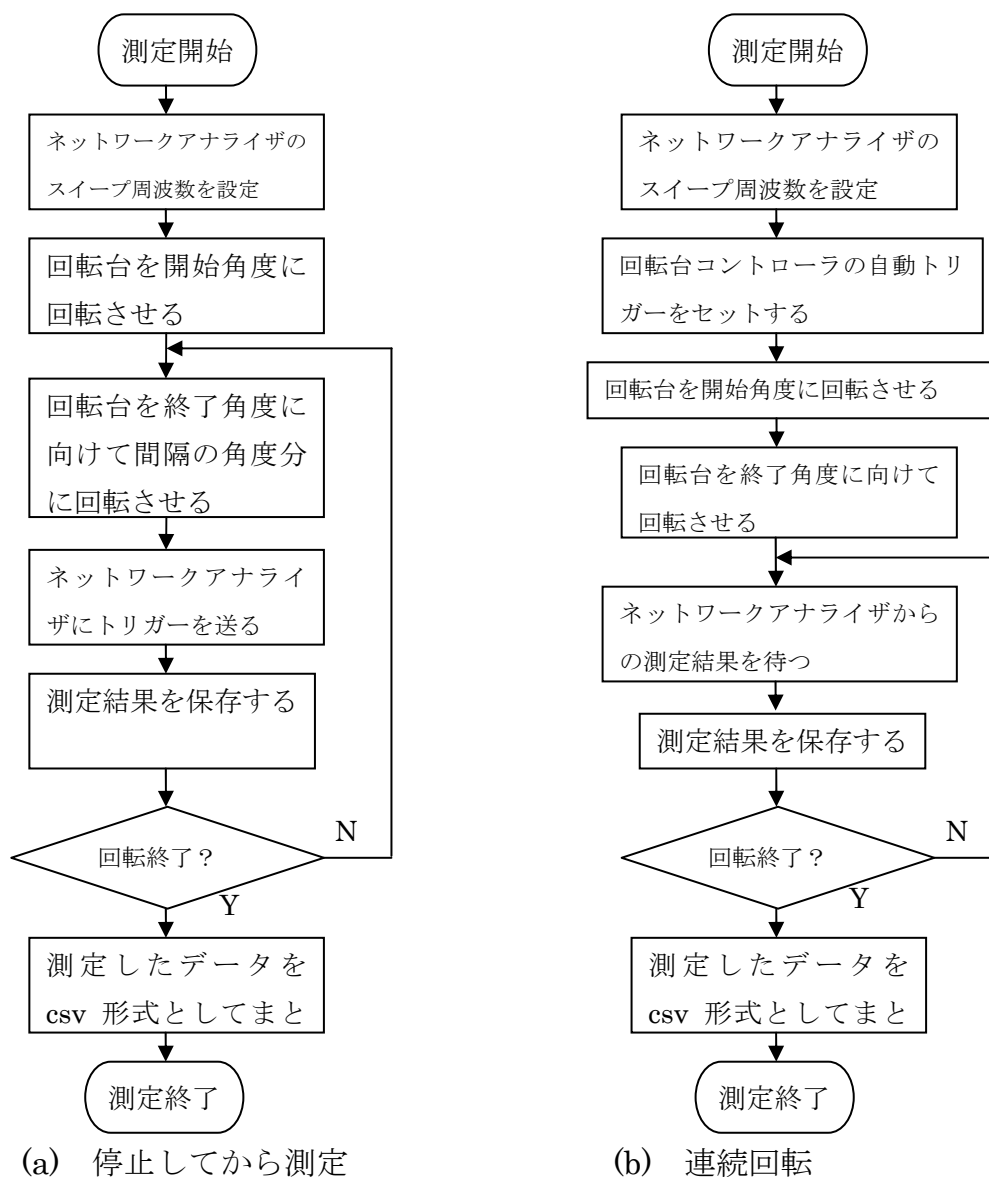


図 3.1 2 方式の測定の流れ図

また、標準アンテナであるダイポールを測定するときに、3dB ビーム幅を測定するため、周波数スイープを行いながら -30° から 30° まで回転台を回転させ測定を行い、各周波数での平均値を求める。

アンテナ測定システムのプログラム仕様を以下に紹介する。プログラムの GUI (Graphical User Interface) は図 3.2 に示す。

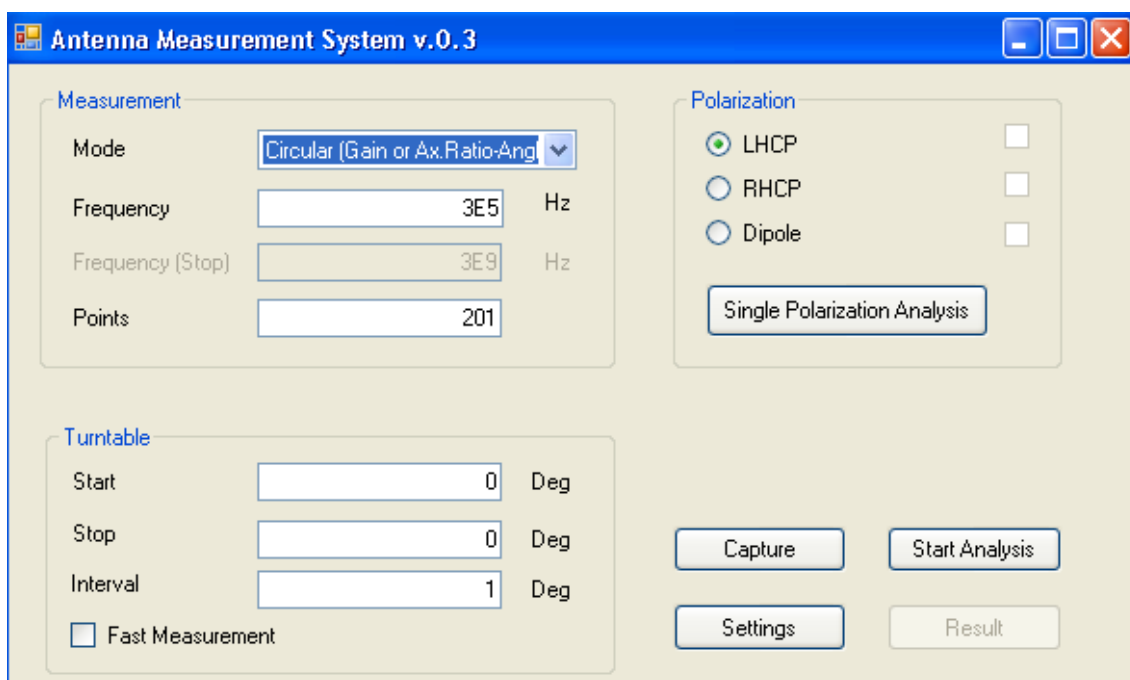


図 3.2 測定システムの GUI 表示

左上にある **Measurement** グループボックスは測定方法にかかわるものを設定するところである。Mode コンボボックスでは測定モードを選択することができる。現在この測定システムには直線偏波（周波数対利得）、直線偏波（角度対利得）、円偏波（周波数対利得、軸比）及び円偏波（角度対利得、軸比）という 4 つのモードがある。その下にある **Frequency**（周波数）のテキストボックスは測定周波数を指定するところである。周波数対利得などの測定では周波数範囲を指定することができる。また、そのさらに下にある **Point** のテキストボックスは測定する点の数を指定するところである。この値が大きくなればなるほど 1 回の測定に時間がかかるが、精密度の高い測定ができるようになる。

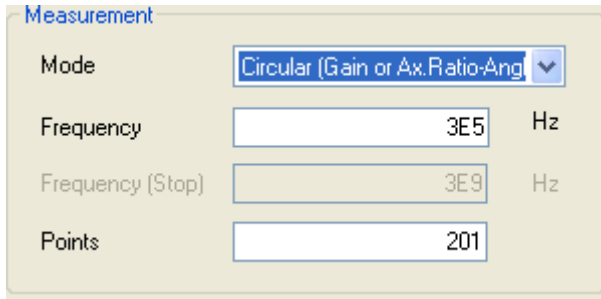


図 3.3 測定方法グループ

その右にある **Polarization** グループボックスは偏波を指定するところである。直線偏波を測定する場合このグループは無効となる。ここにあるラジオボタンを選択し現在測定される送信アンテナの偏波が左旋，右旋，またはダイポール測定を指定することができる。この偏波を指定することにより出力された表計算データがどの列に出力されるか決定される。なお，下にある **Single Polarization Analysis** ボタンを押せば一つの偏波に対して測定を行うことができる。

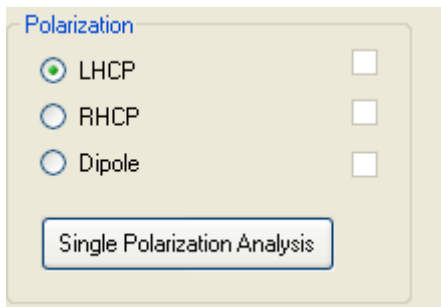


図 3.4 偏波選択グループ

Measurement グループボックスのしたにある **Turntable** グループボックスは回転台の回転方法を設定するところである。角度の測定を行わない場合，このグループは無効となる。ここで開始角度 (**Start**)，停止角度 (**Stop**) と測定角度間隔 (**Interval**) を指定することができる。下にある **Fast Measurement** チェックボックスをチェックすれば連続回転の測定方法で測定を行い，チェックしなければ回転・停止の測定方法で測定を行う。

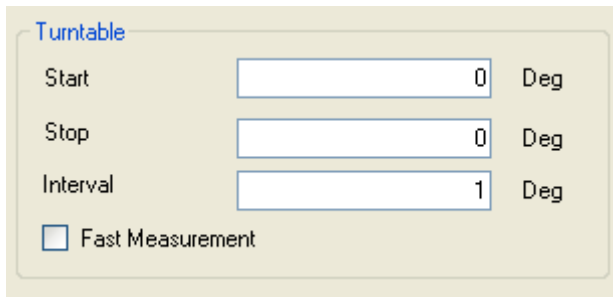


図 3.5 回転台設定グループ

そして右下にあるボタンはコントロールボタンである。Capture ボタンを押すことにより画面にある測定データを記録し、表計算データとして計算機上で保存される。Start Analysis ボタンを押すことにより測定が開始され、すべての偏波を測定し記録する。

Settings ボタンを押すことにより図 3.7 に示す設定画面が表示される。ここで、現在の角度を 0 度にする Set as 0 degree ボタン、特定の角度に回転台を回転させる Move ボタン、回転台を決められた角度で時計回りまたは反時計回りに回転させる CW および CCW ボタンがある。さらに、その下にある Power スライドバーで送信信号の電力を調整し、Rotation Speed スライドバーで回転台の回転速度を調整することができる。注意しなければならないことは、連続回転方法で測定する場合、回転速度が速すぎると測定が追い付けなくなり、システムが停止してしまうことがある。目安として、20%以下の速度に設定すべきである。その下にある 1-port calibration ボタンはネットワークアナライザの送信ポートを校正させるが、現在はまだ完成されていなく、校正は直接ネットワークアナライザから行われた。

Result ボタンは測定が終了したら有効になり、このボタンを押すことにより測定されたデータを計算機上で表計算データとして記録することができる。

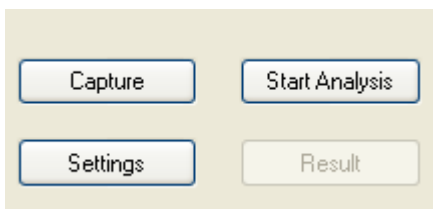


図 3.6 コントロールボタン

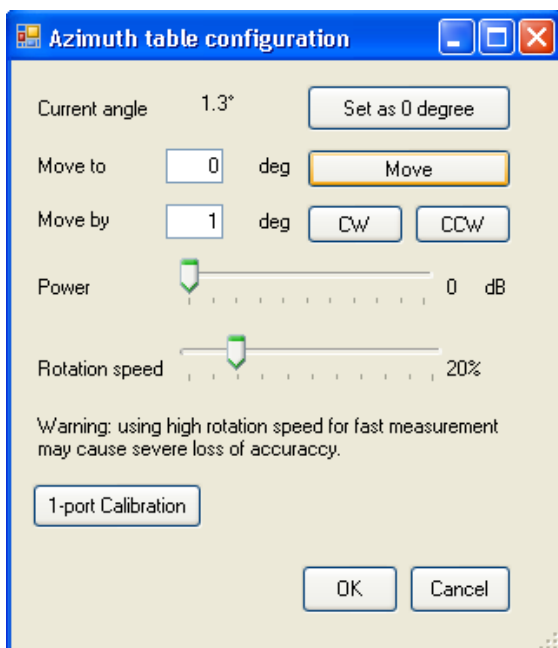


図 3.7 設定画面

この GUI では測定を行うことができるが、実際に測定をしてみると、かえって使いにくいところがある。まず、アンテナを変えるたびに電波無響室の準備を行う必要があるが、一回の準備で最低 30 分かかり、結局 **Start Analysis** のすべての偏波測定が使用できなくなった。また、直線偏波と円偏波に分けても結局あまり差がなく、これらを一つのモードに統合されてもよい。その代わりに、表計算データを読み込んで、2つの偏波データとダイポールデータを統合させて新しい表計算データに書き込む機能があるとより速く記録データをまとめることができる。

実際アンテナを測定すると、シミュレーション上では特性が条件を満たしても実験では特性が条件を満たさなくなる場合がある。その時はやむ得ずアンテナをもう一度作成しなおす場合もある。

測定の流れ図は以下の図 3.7 に示す。

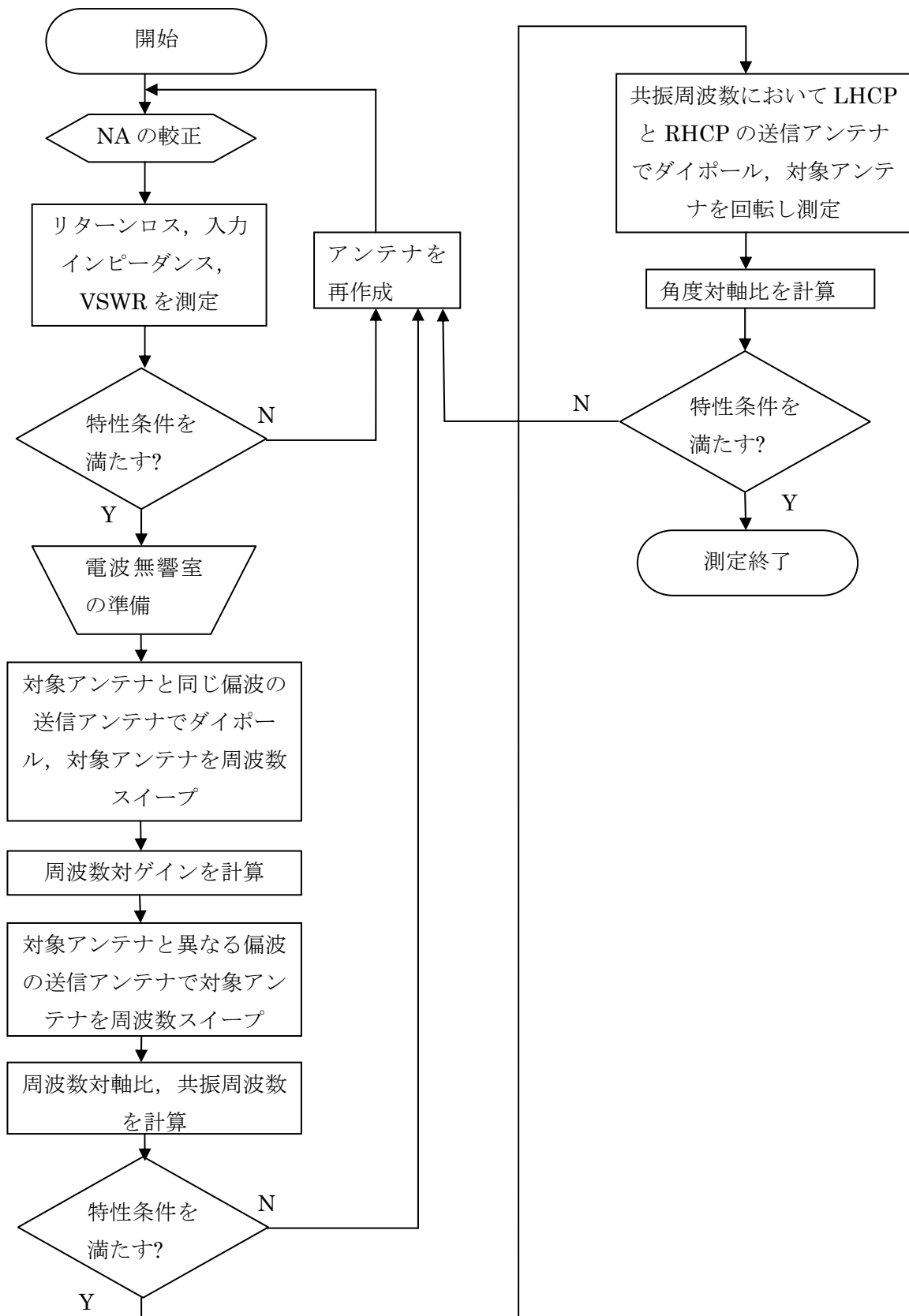


図 3.7 測定の流れ図

4 プログラムの説明

制御用のプログラムは LabView のライブラリを用いて、Microsoft Visual C# で作成した。Visual C# のプログラムの作り方は Visual C++ と異なり、すでに準備されているコントロールを用いてインターフェースを築く。このやり方は Visual Basic でのプログラムの作り方と似ている。ソースコードは C++ 言語を用いて、オブジェクト指向プログラミング(OOP)で作成した。そのため、プログラムはクラスごとに分けて説明できる。以下に、クラスごとにプログラムの説明をする。

CazController クラスは回転台またはアジマステーブルのコントローラを制御するクラスである。このクラスでは回転台を制御する命令コードが登録されていて、このクラスの関数を呼び出すだけで回転台を制御することができる。このクラスの関数は順番に説明する。

初期化するための関数は **Initialize** である。この関数は回転台に初期化の命令を送る。

```
public void Initialize0
{
    Write("INI");
}
```

SetDelimiter は回転台の命令コードの改行に使われる文字コードを決定する関数である。Windows の OS では一般的に CR+LF を開業コードとして利用される。

```
//set the delimiter type
//DL_CRLFEOI - CR+LF, with EOI
//DL_LF - LF only
//DL_EOI - EOI only
//DL_CRLF - CR+LF only
//DL_LFEOI - LF, with EOI
public void SetDelimiter(int Type)
{
    const int DL_CRLFEOI = 0;
    const int DL_LF = 1;
    const int DL_EOI = 2;
    const int DL_CRLF = 3;
    const int DL_LFEOI = 4;

    switch (Type)
    {
        case DL_CRLFEOI:
            Write("DL0");
            break;
        case DL_LF:
            Write("DL1");
            break;
        case DL_EOI:
            Write("DL2");
    }
}
```

```

        break;
    case DL_CRLF:
        Write("DL3");
        break;
    case DL_LFEOI:
        Write("DL4");
        break;
    }
}

```

`EnableServiceRequest` は回転台の `Service Request`、つまりサービス要求を受け入れるかどうかを設定する関数である。サービス要求は回転台の状態を知るために必要なので、測定する前に必ず `ON` にする。この関数に `True` のパラメータを送ればサービス要求が可能になる。

```

public void EnableServiceRequest(bool Enable)
{
    if (Enable)
    {
        Write("S0");
    }
    else
    {
        Write("S1");
    }
}

```

`GetAngle` は現在の回転台の角度を `int` 型変数として戻す関数である。

```

public int GetAngle() //get current angle
{
    string message;
    int pos;
    message = Query("CP?");
    message.Trim();
    message = message.Substring(2);
    message += '\0';

    pos =.ToInt(message.ToCharArray());
    return pos;
}

```

`Stop` は回転台を停止させる命令を送る関数である。

```

public void Stop() //stop rotation
{
    Write("ST");
}

```

`TurnClockwise` は回転台を時計回りに回転する命令を送る関数である。角度パラメータがなければ、この関数は停止させるまで時計回りに回転する命令を送る関数である。

```

public void TurnClockwise(){ //turn clockwise until stopped
    Write("CW");
}

public void TurnClockwise(int angle){
    //turn clockwise for specific angle
    int pos, newpos;

    pos = GetAngle();
    newpos = pos + angle;

    if (newpos > 3600) newpos -= 3600;
    Write("LD" + newpos + "DGCW");
}

```

TurnCntClockwise は TurnClockwise の反対であり、回転台を反時計回りに回転させる関数である。

```

public void TurnCntClockwise()
//turn counter-clockwise until stopped
{
    Write("ST");
    Write("CC");
}

public void TurnCntClockwise(int angle)
//turn counter-clockwise for specific angle
{
    int pos, newpos;

    pos = GetAngle();
    newpos = pos - angle;
    if (newpos < 0) newpos += 3600;

    Write("LD" + newpos + "DGCC");
}

```

MoveTo, MoveClockwise、 MoveCntClockwise は特定の角度に回転させる命令を送る関数である。MoveClockwise は必ず時計回りで、MoveCntClockwise は必ず反時計回りで特定の角度まで回転させる。

```

public void MoveTo(int angle)
//move to specific angle
{
    if (angle < 0) angle += 3600;
    if (angle > 3600) angle -= 3600;
    Write("LD" + angle + "DGMV");
}

public void MoveClockwise(int angle)
//move to specific angle, clockwise
{
    if (angle < 0) angle += 3600;
    if (angle > 3600) angle -= 3600;
    Write("LD" + angle + "DGCW");
}

public void MoveCntClockwise(int angle)
//move to specific angle, counter-clockwise
{

```

```

    if (angle < 0) angle += 3600;
    if (angle > 3600) angle -= 3600;
    Write("LD" + angle + "DGCC");
}

```

SetAngle は回転台の現在位置での角度を再定義する命令を送る関数である。

```

public void SetAngle(int angle)
    //set current angle
    {
        if (angle < 0) angle += 3600;
        if (angle > 3600) angle -= 3600;
        Write("CP" + angle);
    }

```

SetSpeed は回転台の回転速度を調整する命令を送る関数である。

```

public void SetSpeed(int sp)
    {
        //set speed
        Write("SP" + sp);
    }

```

WaitForFinish は回転台が止まるまでプログラムを一時的に停止させる関数である。**NetworkAnalyzer** との同期をとるために必要となる。

```

public void WaitForFinish()
    {
        //wait until the rotation finishes
        m_MBSSession.WaitOnEvent(MessageBasedSessionEventType.ServiceRequest);
        try
        {
            StatusByteFlags b = m_MBSSession.ReadStatusByte();
        }
        catch (Exception e)
        {
            MessageBox.Show(e.Message);
        }
    }

```

EnableTrigger はトリガーを使うかどうかを設定する命令を送る関数である。**True** であれば **ON**, **False** であれば **OFF** になる。

```

//enables/disable ext trigger out
//enable=true ->enable trigger
//enable=false ->disable trigger
public void EnableTrigger(bool Enable)
    {
        if (Enable)
        {
            Write("SG1");
        }
        else
        {
            Write("SG0");
        }
    }

```

```

        Write("SG2"); //clear
    }

```

SetTriggerStartAngle, SetTriggerStopAngle と SetTriggerInterval はそれぞれトリガーの開始角度、終了角度と間隔を設定する命令を送る関数である。

```

    public void SetTriggerStartAngle(int Start)
    {
        Write("TS" + Start);
    }

    public void SetTriggerStopAngle(int Stop)
    {
        Write("TE" + Stop);
    }

    public void SetTriggerInterval(int Inter)
    {
        Write("TC" + Inter);
    }
}

```

CGPIBDevice のクラスは GPIB デバイスを制御するためのクラスである。このクラスは CAzController、CNetAnalyzer にオーバーロードされる。

CNetAnalyzer はネットワークアナライザを計算機から制御するためのクラスである。このクラスにはネットワークアナライザに利用される命令コードがクラス関数の中に登録されていて、クラスの関数を呼び出すだけでネットワークアナライザを制御することができる。以下には順番にクラスの関数を説明する。

EnableServiceRequest は回転台の関数と同じ役割を持ち、ネットワークアナライザのサービス要求の状態を制御する命令を送る関数である。

```

//enable service request
public void EnableServiceRequest(bool Enable)
{
    if (Enable)
    {
        //enable the 7th bit (128) of Service Request Enable Register
        //This bit enables Operation Status Register Summary
        //which includes measurement event
        Write("*SRE 128¥n");

        //we don't need to check for the start of measurement
        //so we set the positive filter to 0
        Write(".:STAT:OPER:PTR 0¥n");

        //we need to check for the end of measurement
        //measurement event is set on the fourth bit of operation register
        //so we set the negative filter to respond to the fourth bit
        Write(".:STAT:OPER:NTR 16¥n");
    }
}

```

```

        //lastly we need to enable the service request to detect event
        //from the fourth bit of operation register
        Write(":STAT:OPER:ENAB 16¥n");
    }
    else
    {
        //disable, just turn the bits 0
        Write("*SRE 0¥n");
        Write(":STAT:OPER:PTR 0¥n");
        Write(":STAT:OPER:NTR 0¥n");
        Write(":STAT:OPER:ENAB 0¥n");
    }
}
}

```

Trigger 関数を呼び出すとネットワークアナライザにトリガーを計算機から送る。回転・停止の測定モードで利用される。**WaitForSingleTriggerFinish**はそのトリガーによる測定が終わるまでプログラムを停止させる関数である。

```

//send a single trigger
public void Trigger()
{
    Write(":TRIG:SING¥n");
}

//wait for a single trigger to finish
//this will only wait for trigger created from Trigger() function
//will not wait for other trigger
public void WaitForSingleTriggerFinish()
{
    try
    {
        Query("*OPC?");
    }
    catch (Exception e)
    {
        MessageBox.Show(e.Message);
    }
}
}

```

WaitForMeasurementFinish はネットワークアナライザの測定が完了するまでプログラムを一時停止させる関数である。この関数は同期するために必要である。

```

//this will wait for measurement finish
//dont forget to enable the service request
public void WaitForMeasurementFinish()
{
    Write("*CLS¥n");
    Query("*OPC?¥n");

    m_MBSSession.WaitOnEvent(MessageBasedSessionEventType.ServiceRequest);
    try
    {
        StatusByteFlags b = m_MBSSession.ReadStatusByte();
    }
    catch (Exception e)
    {
        MessageBox.Show(e.Message);
    }
}
}

```

```
}
```

Echo はネットワークアナライザのメッセージウインドウにメッセージを書くための関数である。そして **ClearScreen** 関数を呼び出すと画面をクリアすることができる。

```
public void Echo(string message) //display message on the screen
{
    Write("DISP:ECHO ¥"+message+"¥\n");
}

public void ClearScreen()
{
    Write("*CLS¥\n");
}
```

LockPanel はネットワークアナライザのボタンをロックする関数である。もし誤作動の可能性があるならばこの関数を呼び出し、ユーザがネットワークアナライザに触れさせないことができる。**UnlockPanel** を呼び出すとロックが開場される。現在のロック状態を知りたいときは **GetLockStatus** を呼び出すことにより知ることができる。

```
//lock, unlock
public void LockPanel()
{
    Write("SYST:KLOC:KBD ON¥\n");
    bLockStatus=true;
}

public void UnlockPanel()
{
    Write("SYST:KLOC:KBD OFF¥\n");
    bLockStatus=false;
}

public bool GetLockStatus()
{
    return bLockStatus;
}
```

SetMeasurementParameters はネットワークアナライザの測定パラメータを設定する命令を送る関数である。**S11,S12,S21,S22** の中から一つのモードを選択し設定する。現在の測定パラメータを知りたいときには **GetMeasurementParameters** を呼び出すことにより知ることができる。

```
//set measurement parameters
//measurement parameters can only be "S11", "S12", "S21" or "S22"
public void SetMeasurementParameters(int Channel, int Trace, string mp)
{
    Write("CALC"+Channel+":PAR"+Trace+":DEF "+mp+"¥\n");
}

public void SetMeasurementParameters(string mp)
```

```

    {
        SetMeasurementParameters(ActiveChannel, ActiveTrace[ActiveChannel], mp);
    }

    public string GetMeasurementParameters()
    {
        return GetMeasurementParameters(ActiveChannel, ActiveTrace[ActiveChannel]);
    }

    public string GetMeasurementParameters(int Channel, int Trace)
    {
        return Query("CALC"+Channel+":PAR"+Trace+":DEF?¥n");
    }

```

SetSweepType はスイープ種類を設定する関数である。LIN,LOG,SEGM, POWの中から選ぶことができる。この測定システムでは使われるモードはリアモードなので、LINを選択する。

```

//set sweep method
//LIN linear
//LOG log
//SEGM segment
//POW power
//for now, use string
public void SetSweepType(string SweepMethod)
{
    SetSweepType(ActiveChannel, SweepMethod);
}

public void SetSweepType(int Channel, string SweepMethod)
{
    Write("SENS" + Channel + "SWE:TYPE " + SweepMethod + "¥n");
}

```

LinearConfigStartStop はリアスイープを行うために開始周波数と終了周波数を設定する関数である。

```

//set stimulus configuration
//start-stop or center-span
//frequency range from 3E5 to 3E9
public void LinearConfigStartStop(string Start, string Stop)
{
    LinearConfigStartStop(ActiveChannel, Start, Stop);
}

public void LinearConfigStartStop(int Channel, string Start, string Stop)
{
    Write("SENS" + Channel + "FREQ:STAR " + Start + "¥n");
    Write("SENS" + Channel + "FREQ:STOP " + Stop + "¥n");
}

```

GetStatistics はネットワークアナライザから統計データを得る関数である。幅、平均、標準偏差、ピークツーピークの情報が得られる。

```

//get statistics data

```

```

//0 - span
//1 - average
//2 - standard deviation
//3 - peak to peak
public double GetStatistics(int Type)
{
    Write("::CALC1:MARK:MATH:STAT:DATA?¥n");
    double[] data = new double[4];

    string msg;
    msg = Read(19);
    data[0] = Convert.ToDouble(msg);
    Read(1);
    msg = Read(19);
    data[1] = Convert.ToDouble(msg);
    Read(1);
    msg = Read(19);
    data[2] = Convert.ToDouble(msg);
    Read(1);
    msg = Read(19);
    data[3] = Convert.ToDouble(msg);
    Read(1);

    return data[Type];
}

```

AutoScale

```

//auto scale the screen to find trace
public void AutoScale()
{
    Write("::DISP:WIND"+ActiveChannel+":TRAC"+ActiveTrace[ActiveChannel]+"::Y:AUTO¥n");
}

//trigger function

//trigger from bus
public void SetTriggerFromBus()
{
    Write("::TRIG:SOUR BUS¥n");
}

//from external source
public void SetTriggerFromExt()
{
    Write("::TRIG:SOUR EXT¥n");
}

public void SetNumberOfMeasurementPoints(int num)
{
    Write("::SENS" + ActiveChannel + "::SWE:POIN " + num + "¥n");
}

/*****
* private members
* *****/

//enable/disable signal output
private void EnableOutput(bool flag)
{

```

```

    if (flag)
    {
        Write(":OUTP 1¥n");
    }
    else
    {
        Write(":OUTP 0¥n");
    }
}

private void InitializeMeasurement()
{
    Write(":INIT1:CONT ON");
    for (int i = 2; i <= 4; i++)
    {
        Write(":INIT" + i + ":CONT OFF");
    }

    //SetTriggerFromBus();

    Write(":DISP:TABL:TYPE ECHO¥n");
    Write(":DISP:TABL ON¥n");
}
}
}

```

5 実験

実験に使用された対象アンテナ (AUT, Antenna Under Test) は GPS 2.5GHz マイクロストリップ LHCP アンテナである。

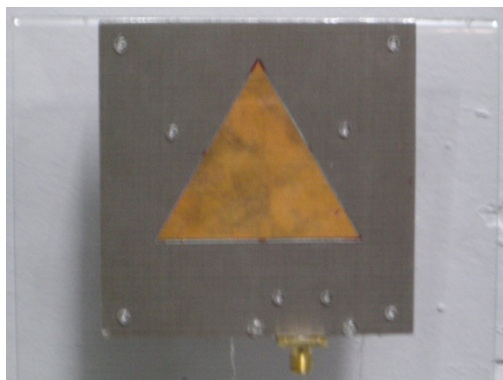


図 4.1 対象アンテナ



図 4.2 キャリブレーションキット

実験は次のように行った。まず、アンテナのリターンロス、SWR および入力インピーダンスを測定した。測定を行う前に、入力ポートを校正した。手順は、キャリブレーションキットをオープン、ショート、ロードという順番でネットワークアナライザに接続し校正した。

次に、アンテナの使用周波数を測定する。使用周波数を求めるには、まずアンテナの最大利得および最低軸比を測定しなければならない。最大利得および最低軸比が同時に得られる周波数は共振周波数となり、その共振周波数は使用周波数となる。

使用周波数が得られた後、その使用周波数にて回転台を回転させ角度ごとに利得および軸比を測定する。また、対象アンテナを 90 度回転させその特性も測定する。

実験結果は以下の図 4.3～図 4.8 に示す。

図 4.3 からは仕様周波数の近傍にはリターンロスが -10dB 以下が明らかになる。図 4.4 には VSWR が 1 に近いのでロスがとても小さいことが分かる。また、図 4.5 から入力インピーダンスは 42Ω になる。理想の 50Ω ではないのでここで少しロスが生じることがわかった。

このアンテナは 2.5GHz を使用周波数として設計されていた。しかし、図 4.6 をみると、共振周波数が 2.5GHz より低い周波数にあることが明らかである。測定値のデータを見ると、共振周波数の値は 2.4945GHz にあった。この結果で次の実験、角度に対しての軸比と利得を測定するには使用周波数を 2.4945GHz にした。ちなみに、最低軸比のところ測定された E パラメータは負の値になるので対象アンテナの偏波は LHCP であることがあきらかになる。

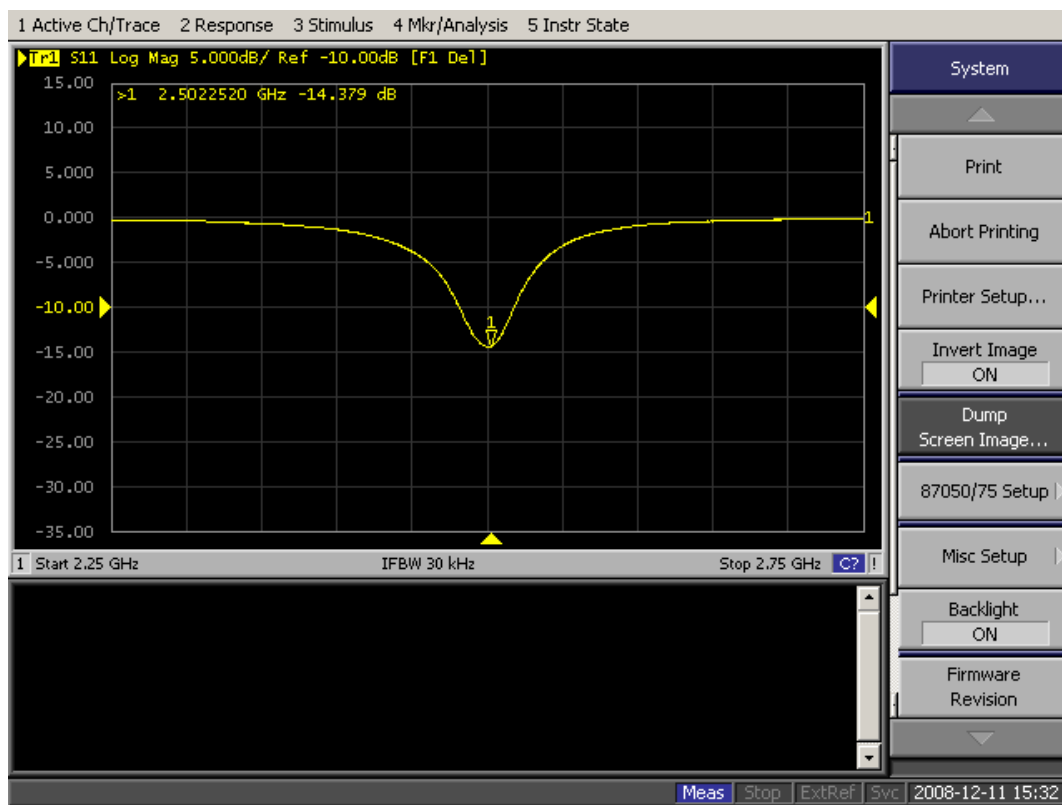


図 4.3 リターンロス

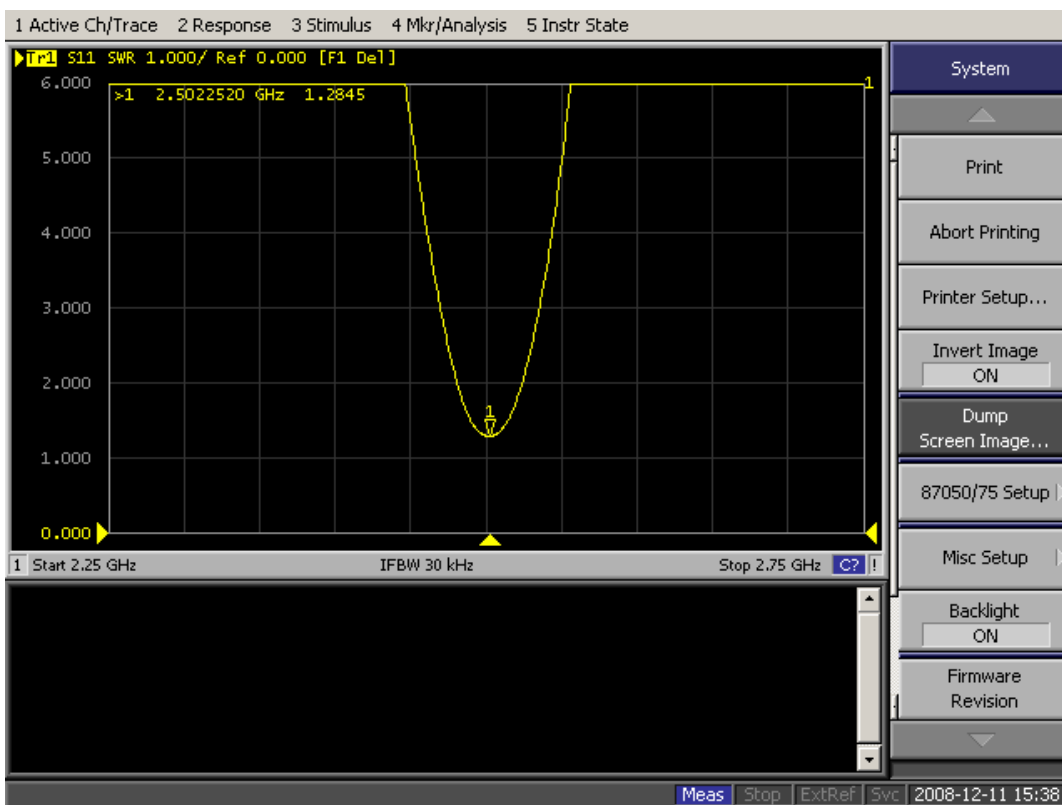


図 4.4 VSWR

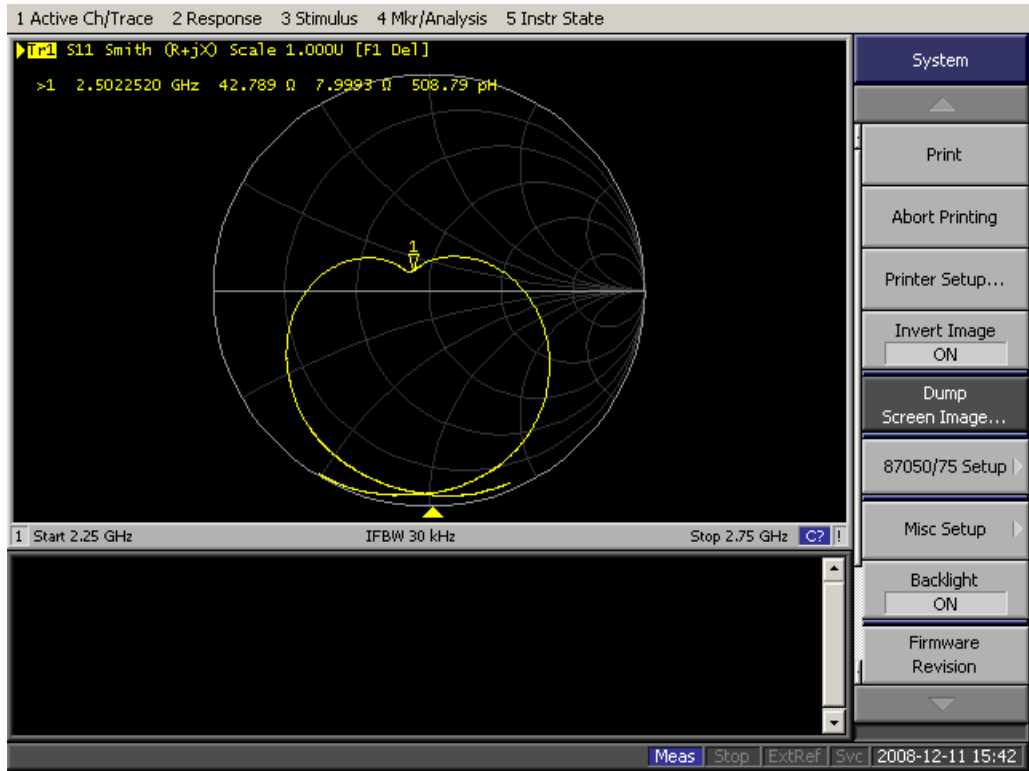


図 4.5 入力インピーダンス (スミス図にて)

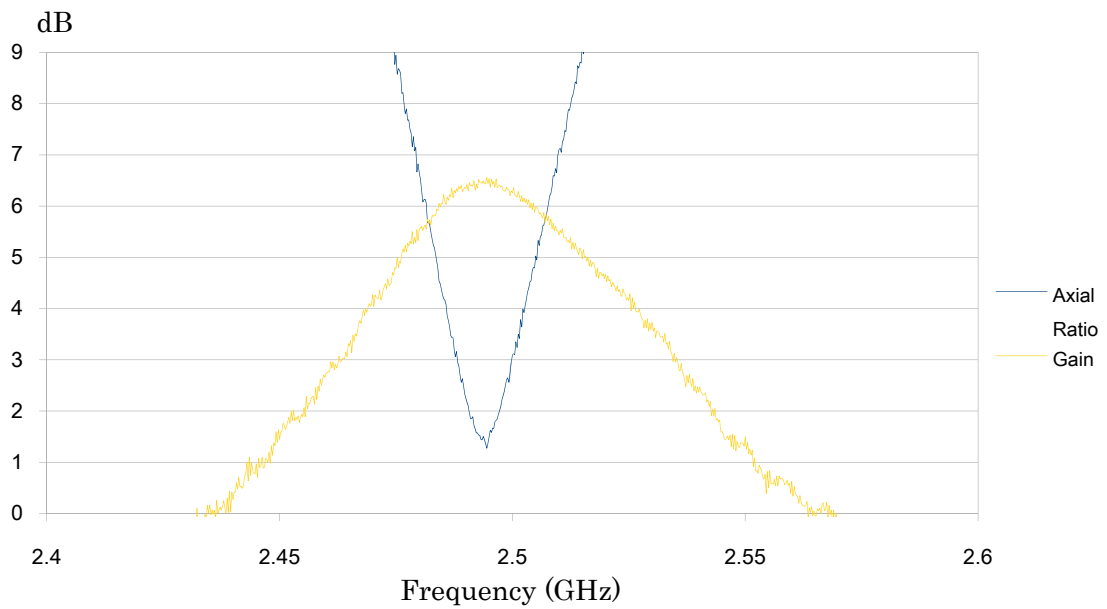


図 4.6 周波数に対しての軸比と利得

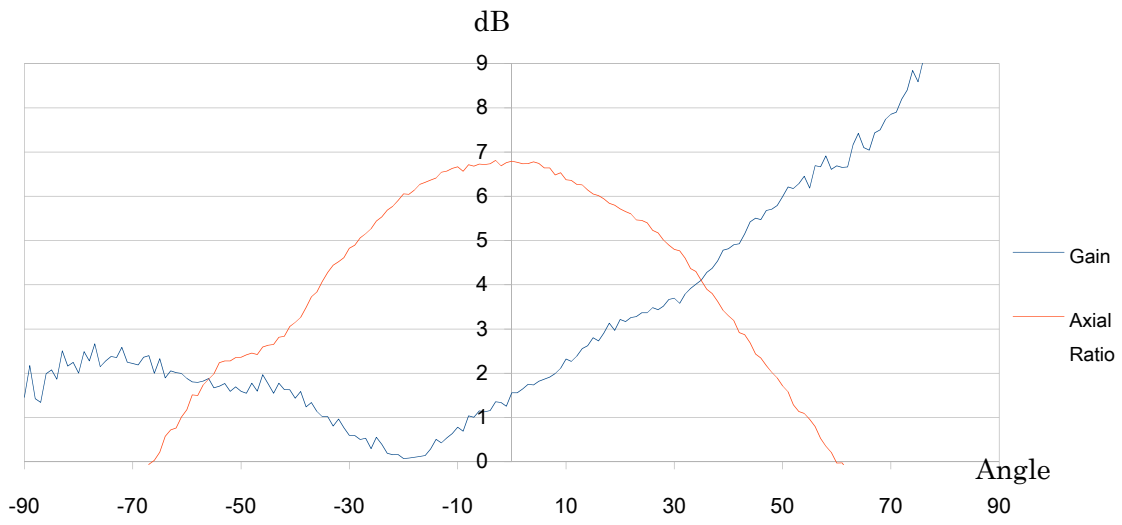


図 4.7 $Az=90$ の時の角度に対しての利得と軸比

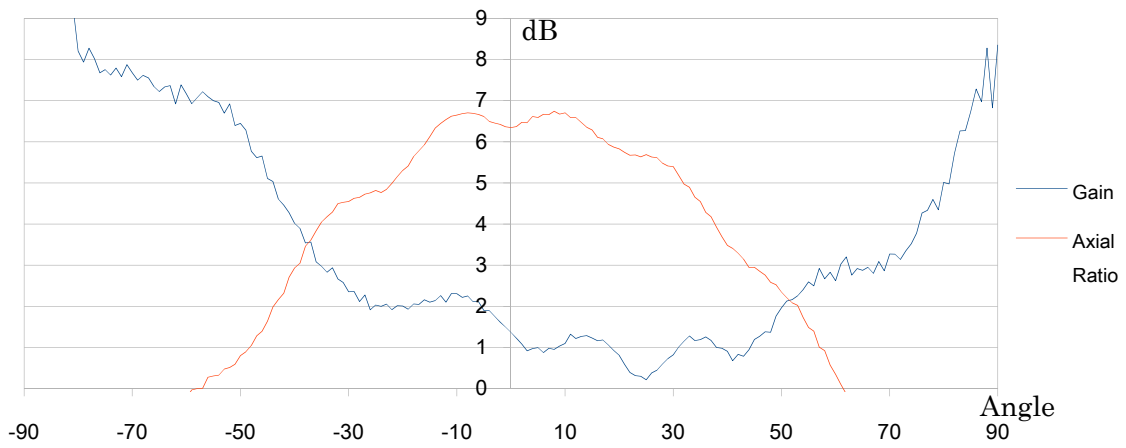


図 4.8 $Az=0$ の時の角度に対しての利得と軸比

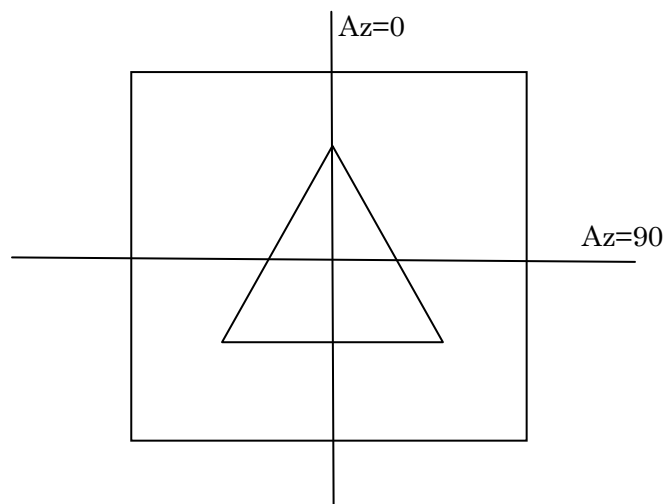
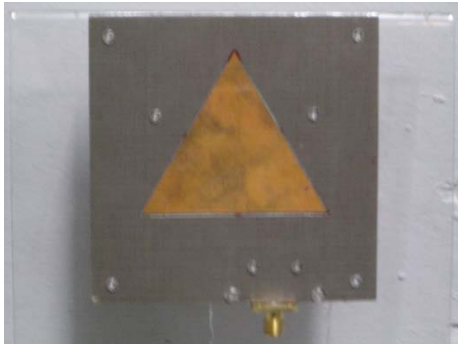
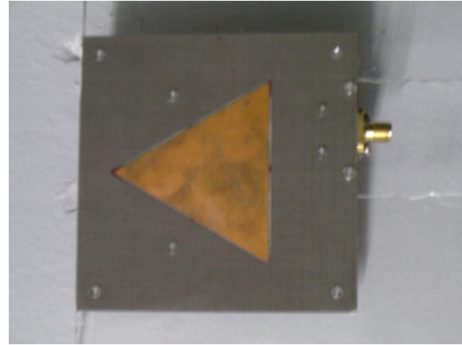


図 4.9 アンテナの方向



(a) $Az=0$



(b) $Az=90$

図 4.10 実際にアンテナを置くときの方向

6 シミュレーションデータとの比較

アンテナのシミュレーション結果では，使用周波数は 2.5025GHz である．しかし，図 5.1 に示すように実際に実験室で測定すると，使用周波数がシフトし，2.4945GHz になった．図 5.2 には使用周波数 2.5025GHz でのシミュレーション結果を示す．利得は dBi(アイソトロピックに対しての利得)単位で表示する．

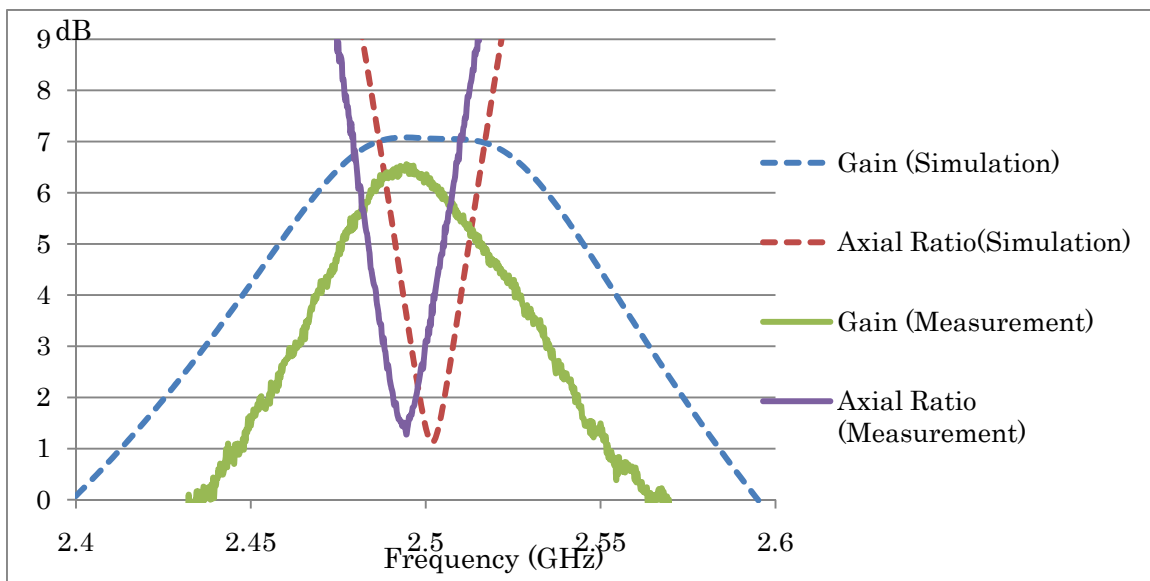
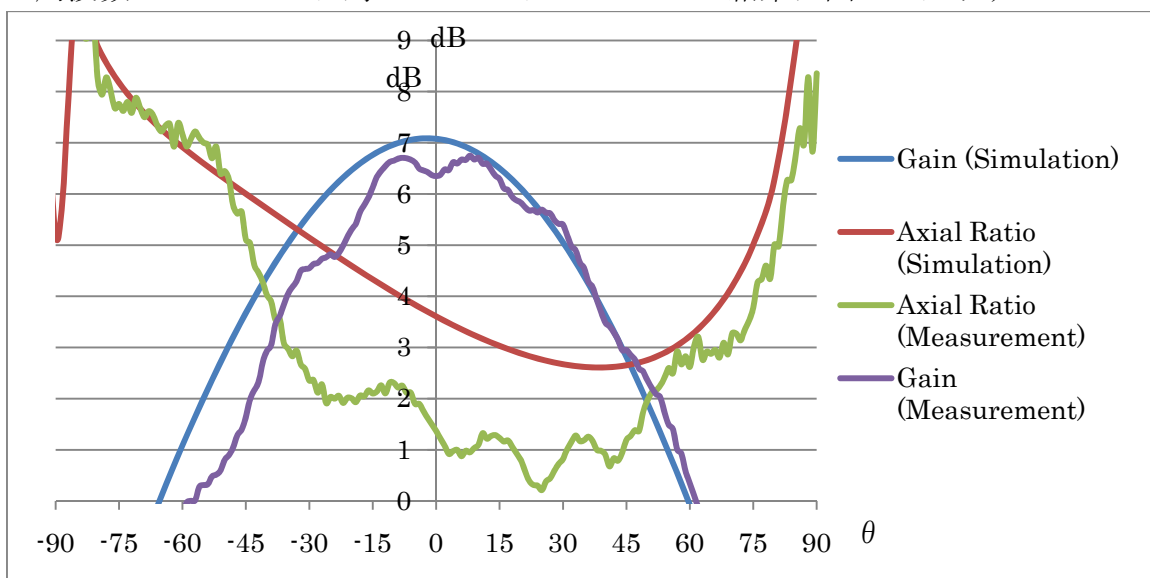
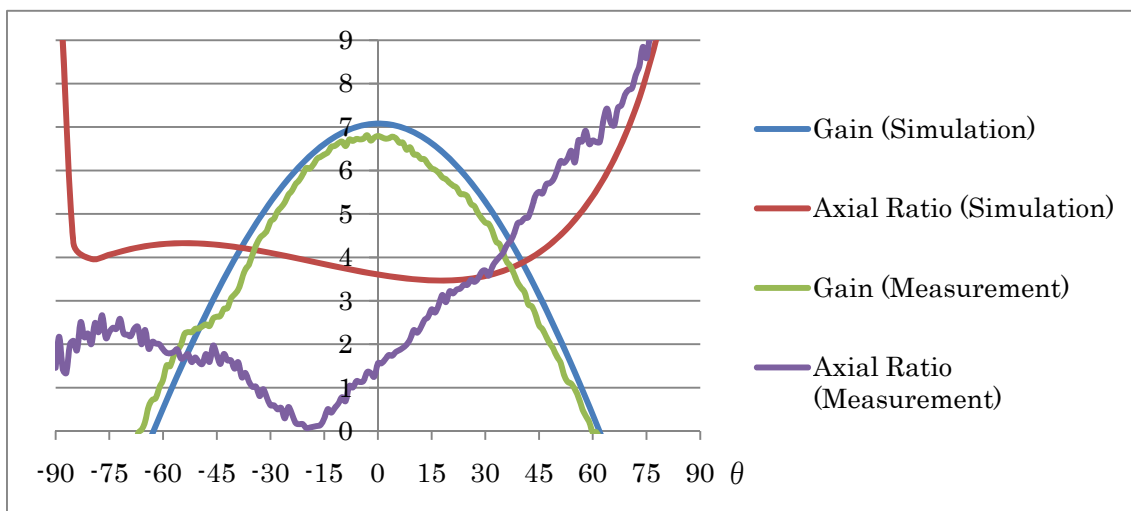


図 5.1 周波数に対しての利得と軸比

周波数 2.4945GHz に対してのシミュレーション結果は図 5.2 に示す．



(a) $Az=0$



(b) $Az=90$

図 5.2 周波数 2.4945GHz, (a) $Az=0$, (b) $Az=90$ で角度に対しての利得と軸比

図 5.2 に示す実験結果とシミュレーション結果を見れば、利得のほうはパターンが似ているが、軸比は実験のほうが良い結果が得られた。このことは、実験に使われたアンテナの使用周波数は 2.4945GHz にシフトしたと一致している。もしアンテナの使用周波数は 2.5025GHz のままであればよい軸比が得られず、シミュレーションに近い結果が得られると考えられる。

7 まとめと今後の研究

本研究では航空機・小型衛星搭載用の円偏波合成開口レーダの測定システムを構築した。このシステムは、電波無響室内の回転台およびその制御用コントローラ、アンテナの受信電力を測定するネットワークアナライザ、および測定データを記録する計算機を連携させることによって、アンテナのゲインや円偏波アンテナ特有の性質である軸比を手動測定の場合に比較してはるかに高精度かつ短時間で測定することができる。角度ごとに回転台を回転・停止させてから測定する方式と台を連続的に回転しながら測定を行う方式の2通りの測定方法があるが、前者の方式は測定時間が長く、停止・回転の繰り返しでアンテナポールがぶれるため後者の方式に比較すると不利である。円偏波アンテナを測定する実験によりシステムの正常動作が確認されたが、測定するときに回転台を高速で回転させるとネットワークアナライザの測定が遅れ、システムが停止する問題が明らかになった。

また、計算機に使われたインターフェースにはGUIを利用したが、システムを構築したときに想像した使用方法と異なって、実際に測定したときに使いにくいところがある。そのため、このインターフェースを改良する必要がある。

今回行われた実験ではやはり軸比がシミュレーション結果に比較すると低くなった。しかし、アンテナの使用周波数において軸比が低くならないためには、この測定結果は正しいと考えられる。この現象は周波数シフトによるものと思われる。この仮定が正しければ、システムの正常動作が確認できるが、まだ不確定要素が多すぎるためさらに実験を行う必要がある。

今後の研究課題として、これらの問題を解決するとともに、このシステムを拡張して、ノイズなどの効果を削減するために補間法を導入し、近傍界の測定と遠方界の変換方法の開発を行う予定である。これによって、近い将来寸法制限のある電波無響室での合成開口レーダ（アンテナを含む）性能試験を実施可能になると期待される。

参考文献

- [1] Constantine A. Balanis, “Antenna Theory: Analysis Design, Third Edition,” John Wiley & Sons, Inc., 2005.
- [2] Warren L. Stutzman, “Polarization in Electromagnetic Systems”, Artech House, Inc., 1992.
- [3] http://en.wikipedia.org/wiki/Dipole_antenna
- [4] Albert D. Helfrick, “Electrical Spectrum and Network Analyzers A Practical Approach”, Academic Press, Inc., 1991.