大角度スケールのCMB偏光測定において光学系内の反射が及ぼす系統的効果の研究

Study of systematic effects due to reflections in an optical system for the large angular scale CMB polarization measurements.

髙瀬 祐介，石野 宏和，長野 佑哉（岡山大学）

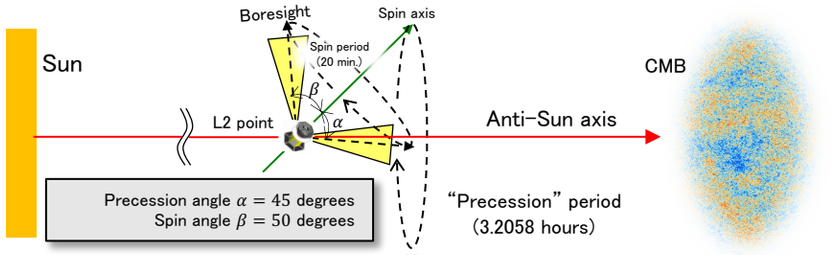
2021/01/6-7 宇宙科学シンポジウム

1. 概要

近年，大角度スケールのCMBのBモード観測が地上や衛星で実施，もしくは計画されている．本ポスターでは科学衛星によるCMB全天観測を仮定し，その光学系の内部での反射がBモードの測定に与える系統的効果について述べる．

2. 前提条件

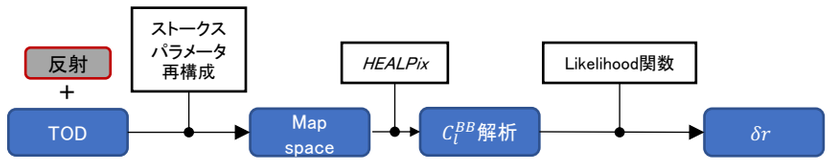
科学衛星による観測のモデルとして現在進行中のLiteBIRD計画を採用する．LiteBIRDは3年にわたって宇宙全天を以下の軌道で観測する[1]．



[1] M. Hazumi et al. , "LiteBIRD satellite: JAXA's new strategic L-class mission for all-sky surveys of cosmic microwave background polarization", *SPIE* 21 December, 2020, https://doi.org/10.1117/12.2563050

シミュレーション全体の流れ

1. LiteBIRDの掃引手法でCMBを全天観測し，TOD(Time Ordered Data)を作成する
   - ➢ TODには光学系内での反射による影響を加える
2. ストークスパラメータを再構成し，TODを全天に焼き直す
3. Likelihood関数を用いてC_l^{BB}からテンソル・スカラー比に対する系統誤差δrを算出する



3. 反射のモデル

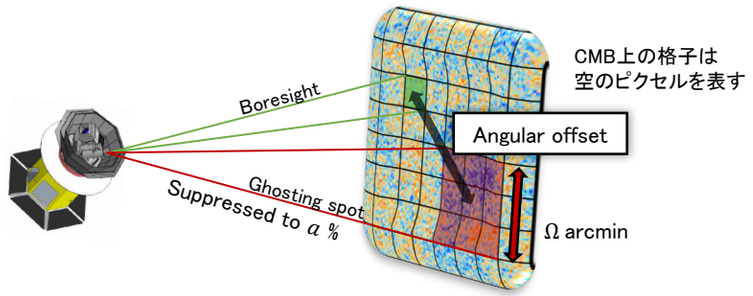
光学系内で反射が起こると，本来見ている方向(Boresight)とは違う方向のピクセルからの光が漏れ込んでくる．下の図はその効果を視覚的に表したものである．

- ある瞬間，衛星は緑のピクセル(Boresight)を観測している．
- 光学系内で起こった反射はBoresightとは反射角の分だけずれてしまい，あたかも異なる方向からやってきた光のように観測されてしまう．
  - ➢ 今回は，この反射光が飛来する角度は常にBoresightからAngular offsetの分だけ離れているような一般的なモデルを考える．
- このとき，反射光はΩ arcminの大きさを持ってやってくると仮定する．下の図ではAngular offsetだけずれた位置のピクセルを中心とした赤い領域(Ghosting spot)で表している．
- 反射光の強度はa % に抑制してBoresightの観測データに加える．

従って，観測されるTOD $P_{observe}$は以下の式で表される．

$$P_{observe} = P_{boresight} + aP_{ghost}$$

$P_{boresight}$ は緑のピクセルで検出されるパワーで，P_{ghost}は赤い領域のピクセルで検出されるパワーの総和を表す．



4. ストークスパラメータの再構成

ここでは，得られたTODを全天のピクセルに対応するストークスパラメータに再構成するための方法を示す．まず，ある空のピクセルから検出器が受け取るパワーを次のように定義する．

$$p_i = \boldsymbol{w}_i^t H_i \boldsymbol{s} + a p_{g;i}$$

ここで，iはi番目に測定されたデータであることを示し，$\boldsymbol{w}_i^t, H_i, \boldsymbol{s}$はそれぞれ次の式で定義される．

$$\boldsymbol{w}_i = \frac{1}{2}\begin{pmatrix} 1 \\ \sin 2\psi_i \\ 0 \\ \cos 2\psi_i \end{pmatrix},\quad \boldsymbol{s} = \begin{pmatrix} I \\ U \\ V \\ Q \end{pmatrix},\quad H_i = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -\cos 4\phi_i & 0 & \sin 4\phi_i \\ 0 & 0 & -1 & 0 \\ 0 & \sin 4\phi_i & 0 & \cos 4\phi_i \end{pmatrix}$$

ψ_i：検出器が感度をもつ偏光方向と空の経線とのなす角
$\boldsymbol{s}$：ストークスパラメータ
H_i：HWPのミューラーマトリックス

また，反射の効果は$p_{g;i}$によって与えられる．
今，反射光が飛来してくる方向を(θ_g, φ_g)として，$p_{g;i}$を次のように書く．

$$p_{g;i} = \sum_{k=0}^{spot\ size} \boldsymbol{w}_i^t H_i \boldsymbol{s}_{g;i}^{(k)}$$

$\boldsymbol{s}_{g;i}^{(0)}$は$(\theta_g, \varphi_g)$方向に存在するピクセルのストークスパラメータであり，kが増加するに従って，周りのピクセルの値の和がとられていくように定義する．
つまり，$spot\ size = 25$とすると(θ_g, φ_g)を中心として5×5ピクセルの値の和がとられる．
また，係数aは反射光の強度を調整するためのパラメータである．
最終的に，観測によって得られるストークスパラメータ$\tilde{\boldsymbol{s}}$は以下の式で与えられる．
なお，反射の成分については偏光が回転していないと仮定している．

$$\tilde{\boldsymbol{s}} = \left(\sum_i H_i^t \boldsymbol{w}_i \boldsymbol{w}_i^t H_i\right)^{-1} \left(\sum_i H_i^t \boldsymbol{w}_i p_i\right)$$

5. 入力パラメータ

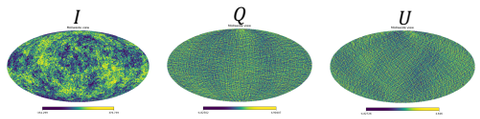
反射のモデル

- 反射光の飛来角(Angular offset)が変化したとき，その影響がδrにどのような系統的効果を及ぼすか確認する
- Angular offsetを$0 - 900$ arcminまで変化させてδrの依存性を確認する
- $\Omega = 30$ arcmin とする
  - ➢ ピクセルサイズが約6 arcminなので，$spot\ size = 5\times5 = 25$ ピクセルと定義すると，30 arcmin 程度のスポットとなる
- 反射の強度を$a = 0.05$ %に抑制

Input map

CAMB により以下のレシピでCMBマップを作成

- $N_{side} = 512$
- $r = 0$
- $lensing = False$（$E\rightarrow B$を見たい）
- Smoothing : $FWHM = 30$ arcmin
  （LiteBIRDの典型的なビーム幅を仮定）



6. δrの計算方法

Likelihood関数を次のように定義する．（C_lは全てC_l^{BB}を指す）

$$\ln L(r) = \sum_l \ln P_l(r), \qquad \ln P_l(r) = -\frac{1}{2}(2l+1)\left[\frac{\hat{C}_l}{C_l} + \ln C_l - \frac{2l-1}{2l+1}\ln \hat{C}_l\right]$$

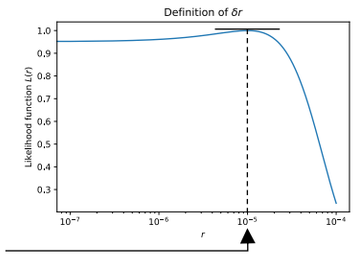
$$\hat{C}_l = C_l^{lens} + C_l^{obs}$$
$$C_l = rC_l^{tens} + C_l^{lens}$$

- C_l^{tens}は$r = 1$としたときのテンソルモード
- C_l^{obs}は観測されたC_l^{BB}

$L(r)$をプロットすると右図のようになる．
このグラフの最大値をとるrの値，すなわち

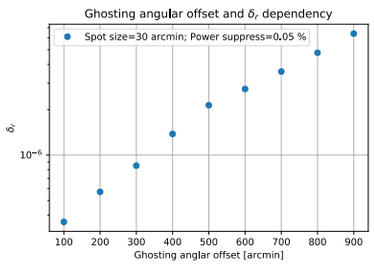
$$\left.\frac{dL(r)}{dr}\right|_{r=\delta r} = 0$$ となる点をδrとした．

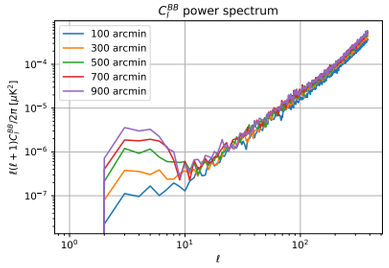
Ex.) $\delta r = 1.0\times10^{-5}$



7. 計算結果

Angular offset とδr，C_l^{BB}の関係は以下のようになった．



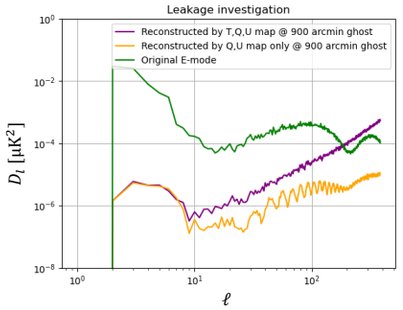


例えば$\delta r < 0.001$を求めるLiteBIRDにおいて，観測に付随する系統誤差は約100種類あると言われている．このことから，LiteBIRDは目標とする精度を実現するために一つひとつの系統誤差上限を5.6×10^{-6}以下に設定している．
従って，"反射がδrへ及ぼす系統的効果は強度が0.05 %以下の時，入射角度800 arcmin以下"までなら許容できることになる．ただし，今回はCMBのみの結果であり，前景放射を入れると要求はさらに厳しくなる可能性がある．
パワースペクトルではAngular offsetの増加とともに，ℓの小さい領域にバンプができている．また，$\ell > 10^1$の領域では直線的な増加が見られる．これらの考察を以下で行う．

8. 漏れこみの原因探索

紫：900 arcmin からの反射
橙：T mapをゼロにして計算した（P のみ）．
緑：E-mode

- 橙はT からの漏れこみは無いから$E\rightarrow B$を見ていることになる．
- グラフの形状から紫($\ell > 10^1$)は$E\rightarrow B$よりも$T\rightarrow B$が支配的
- 橙($\ell < 10^1$の領域)は$E\rightarrow B$が支配的
  - ➢ 反射によって大角度構造が生成
- HWPがあってもピクセル間の温度勾配が大きな領域ではTからの漏れこみが大きくなる．



9. Julia言語

本研究は近年，数値計算業界を賑わせているJuliaを用いて行なった．
文法はPythonと似ているが，C言語並みに高速なループ計算が行える．
Julia言語の利点を列挙すると，

- 動的型付け言語であるが，JITコンパイルにより高速化される
- Jupyter環境で使用できる
- スレッド並列計算の実装が非常に簡単
- GPU計算のハードルも低い
- インストールが簡単
- *HEALPix* が実装されている
- Python, R, C++のライブラリを呼べる

コーディングの例

Numpyの呼び出し

```
using PyCall

np = pyimport("numpy")
x = np.arange(0, 10)
```

forループのスレッド並列化

```
@threads for i in 1:100
    x += i
end
```

便利なライブラリとマクロ

- *BenchmarkTools*
  - ➢ 自作関数の実行時間，メモリ使用量を確認できる
- *@code_warntype*（マクロ）
  - ➢ 自作関数の前に置くことで関数内の変数が型安定になっているか確認できる．

julia vs Python

同じ内容の計算で速度を比べてみた．
PythonはCythonを用いて高速化したためコーディングに時間がかかった一方，Juliaは最適化に気を配らなくてもPythonの70倍の速度が実現できた．

	Run time [sec]
Python (Cython)	900
Julia	12