

ランダム変形法とモーメント法によるアンテナの最適設計

株式会社EEM 大賀明夫

2021年6月17日

電子情報通信学会 アンテナ・伝播研究会

目次

1. 研究の目的と背景
2. 計算法
 - 2.1 組み合わせ最適化
 - 2.2 計算パラメーターの検討
 - 2.3 アンテナの高さ
 - 2.4 MPIによる並列計算
3. 計算例と計算時間
 - 3.1 計算例
 - 3.2 計算時間
4. FDTD法による最適設計
5. まとめ

1. 研究の目的と背景

●研究の目的

- ・アンテナの最適設計のための汎用的な手法を求める
- ・多様なアンテナ特性をカバーすること
- ・放射パターンの制御、周波数特性の制御、偏波特性の制御等

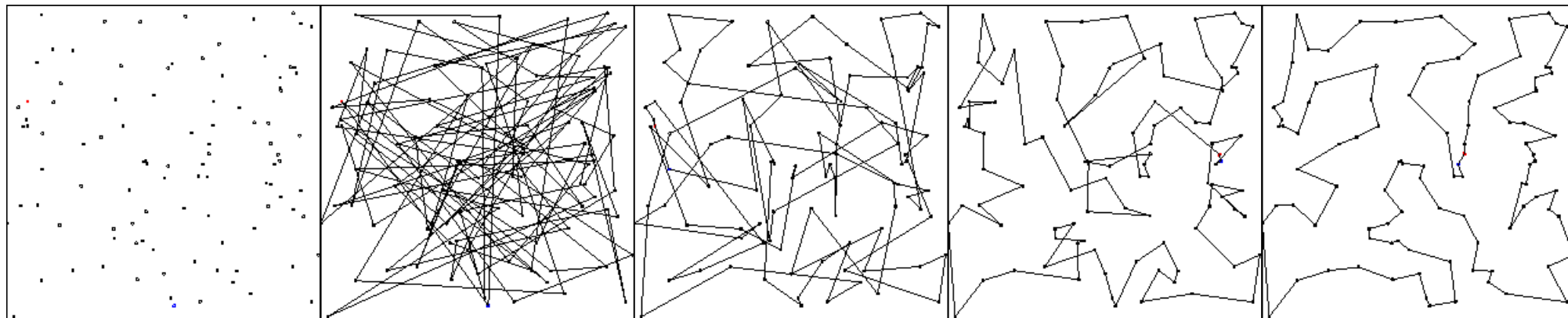
●研究の背景と考え方

- ・面倒な作業は全部コンピュータにやらせる
- ・人間はあまり頭を使わない
- ・ソルバーとしては小規模問題を極めて短時間に計算できるモーメント法を考える
- ・アンテナ形状を離散化し組み合わせ最適化問題に変換する
- ・力まかせ探索 (**Brute-force search**)
- ・狭義の力まかせ探索は全通り調べることだが本ケースでは $2^{220}=10^{66}$ 通りある (組み合わせ爆発)
- ・実用的な計算時間でそこそこの解は得られないか？

2. 計算法

参考：巡回セールスマン問題（本手法のヒントとして）

※レイトレーシング法に転用できないか



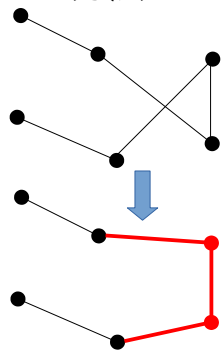
都市配置(100点)

初期経路(ランダム)

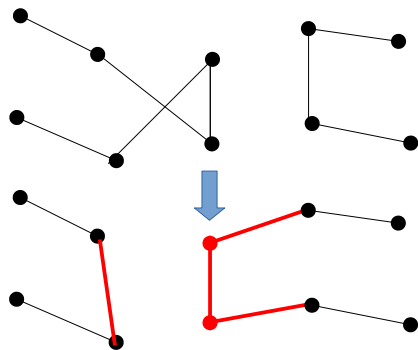


(近似)最短経路

●Linの方法



(1)反転



(2)切り取り、挿入

● アニーリング法（焼きなまし法）

- ・ 目的関数が良くなる → 採用
- ・ 目的関数が悪くなる → 確率 $\exp(-\Delta E/kT)$ で採用

■ 温度Tを徐々に下げる（焼きなまし）

=悪くても採用される確率が徐々に下がる

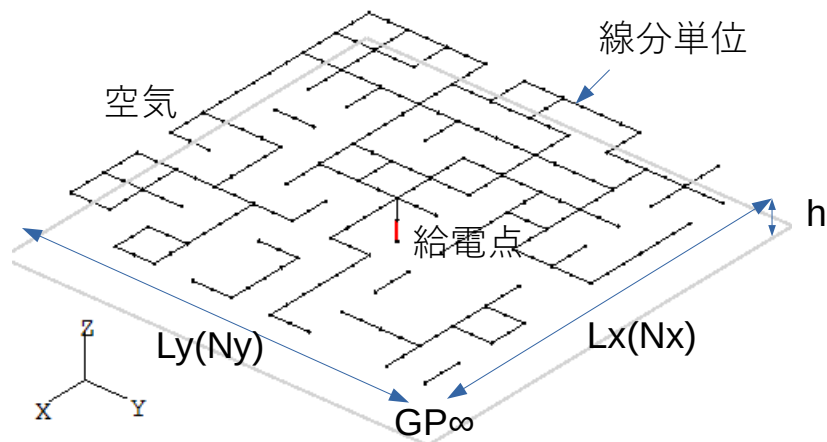
■ ランダムに多数回変形すると時々よくなることもある
これを繰り返すことによって最適解に近づく
→ 効率は悪いが汎用的で強力

2.1 組み合わせ最適化

組み合わせ最適化の4ステップ

| ステップ | 巡回セールスマン問題 | 本方法 |
|------------------|---------------------|--------------------------------|
| 1. 系の可能な配置の記述 | 都市の配置（所与であり固定） | メッシュの各線分のON/OFF |
| 2. 配置のランダムな変形法 | Linの方法 | ランダムに一つの線分をとりON/OFFを反転する |
| 3. 最小化すべき目的関数の設計 | 最短経路を求める（経路長を最小化する） | 目的とするアンテナ特性から決める |
| 4. 反復計算の制御法 | アニーリング法 | 単純ON/OFF法 良くなる時は採用、悪くなる時は棄却 |

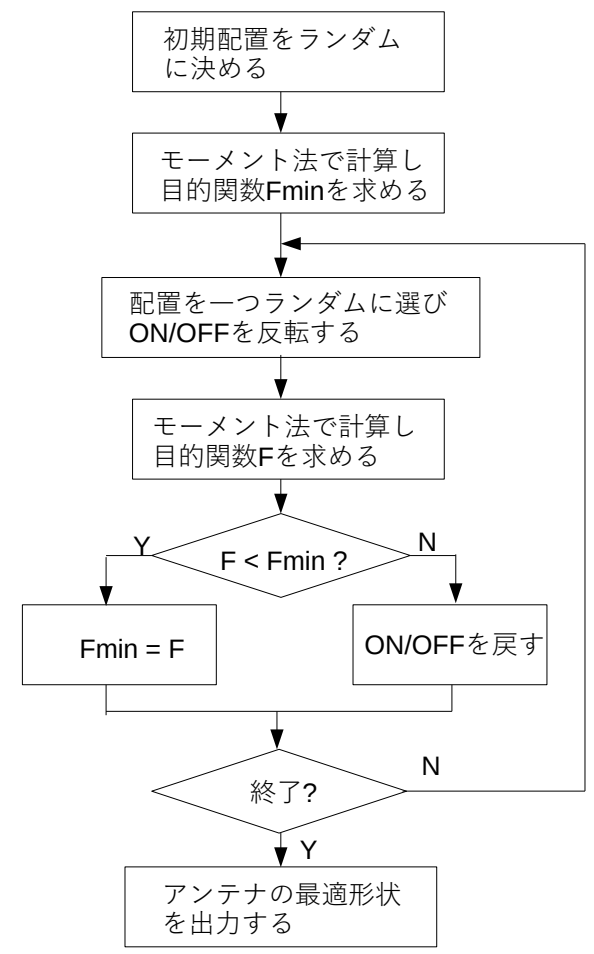
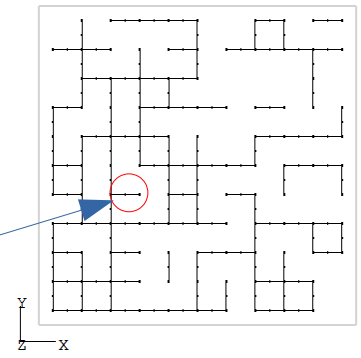
アンテナ形状と計算手順



アンテナ形状

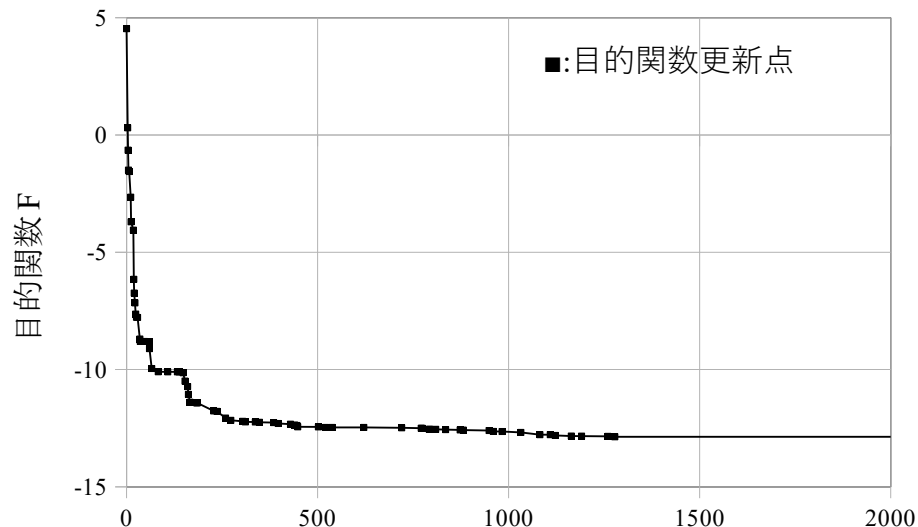
- ・モーメント法モデル
- ・無限大グラウンド板の上方に長方形をとり、それをメッシュ分割し、各線分のON/OFFを考える

ランダムに1個選びONならOFFにし、OFFならONにして目的関数を調べる



計算手順

2.2 計算パラメーターの検討：反復回数、試行回数、初期占有率



反復回数

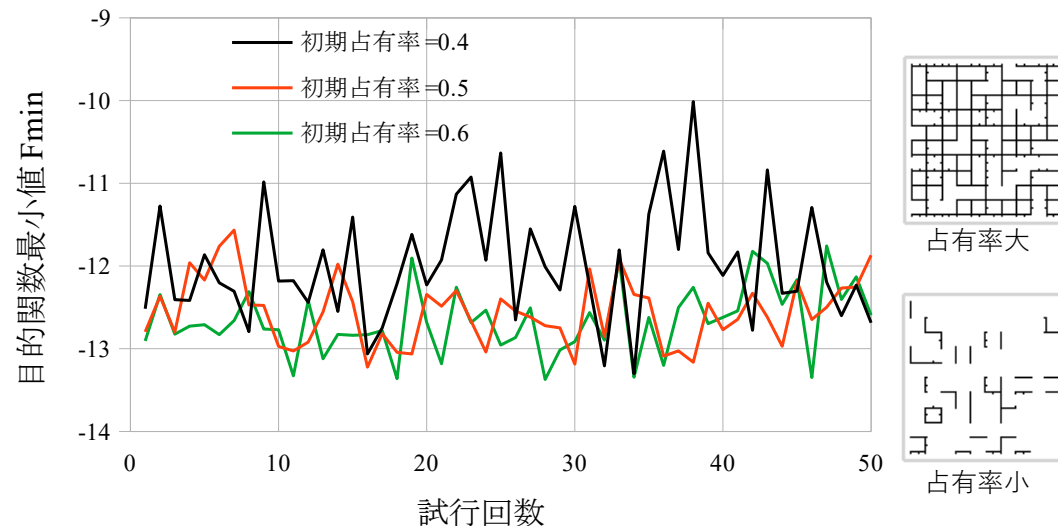
収束状況

$L_x=L_y=100\text{mm}$, $N_x=N_y=10$, $h=10\text{mm}$, 3GHz

$w_1=-1$, $w_2=0.05$, 初期占有率=0.5, $Z_0=50\Omega$

目的関数: $F=w_1G(\theta=0)[\text{dB}]+w_2|Z_{\text{in}}-Z_0|$

■反復回数は1000回を超えるとほぼ収束する

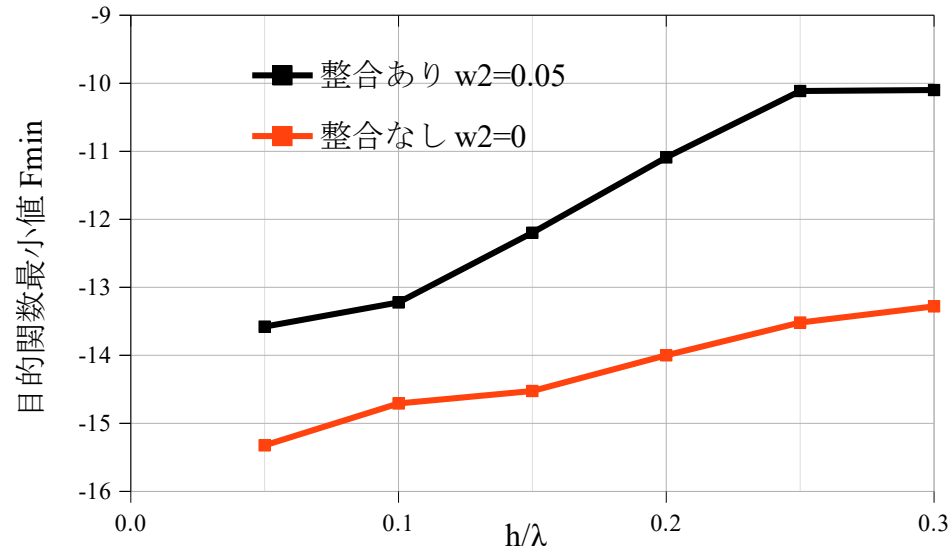


- ・一度の反復計算では局所解に陥っている可能性があるため初期配置をランダムに変えて繰り返し計算する、これを試行回数と呼ぶ
- ・占有率=ON線分数/全線分配数 N_s

■試行回数は数十回でよい解が得られる
■初期占有率は0.5~0.6がよい

■3個の計算パラメータ（ハイパーパラメーター）は最適値を見つけやすい→1回の計算で済む

2.3 アンテナの高さ



アンテナ高さ と 目的関数最小値の関係

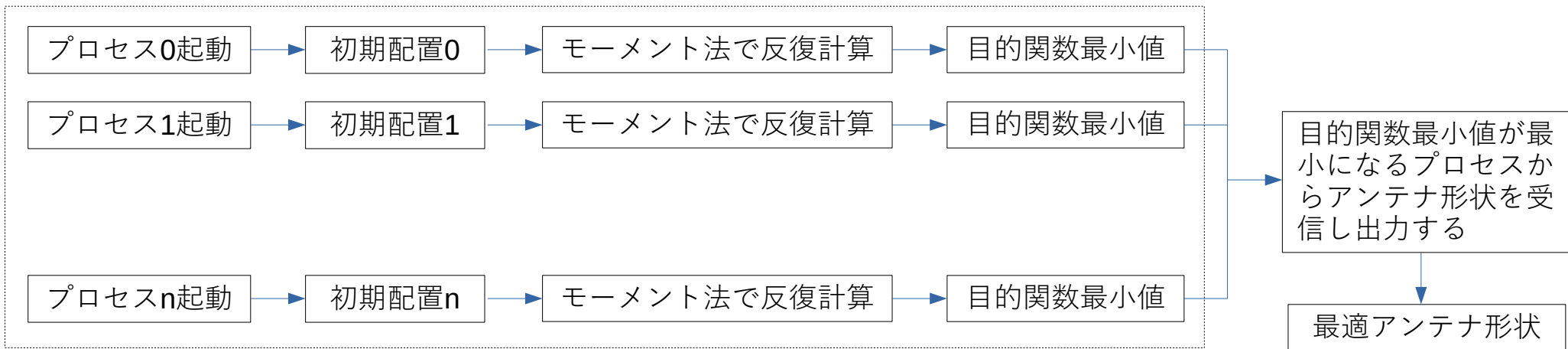
$L_x=L_y=100\text{mm}$, $N_x=N_y=10$, 3GHz,
 $w_1=-1$, 初期占有率=0.5, 反復回数=3000, 試行回数=30

■ アンテナの高さ h が低いほど 目的関数最小値が小さい

2.4 MPIによる並列計算

MPI(Message Passing Interface)による並列計算

通信不要



- 試行回数について並列計算する
- 各プロセスは完全に独立しているので分散メモリー環境での実装が容易
- 計算の途中では通信不要
- プロセスの負荷はほぼ均一

■MPIによる並列計算に非常に適している

※OpenMOM自体はMPI並列化されていない

3.1 計算例

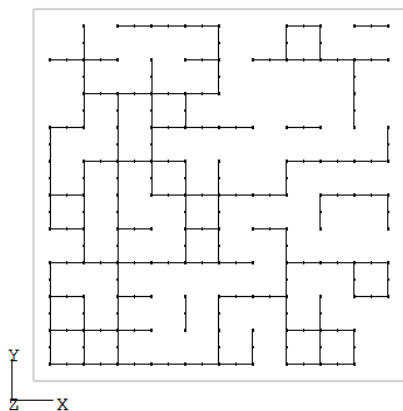
共通計算条件

| パラメーター | 値 |
|---------------------|---------------|
| $Lx \times Ly$ | 100 × 100 mm |
| $Nx \times Ny$ (Ns) | 10 × 10 (220) |
| 線分分割数 | 2 |
| アンテナ高さh | 10 mm |
| 導線半径 | 0.5 mm |
| 特性インピーダンス Z_0 | 50 Ω |
| 初期占有率 | 0.5 |
| 反復回数 | 3000 |
| 試行回数 | 32 |

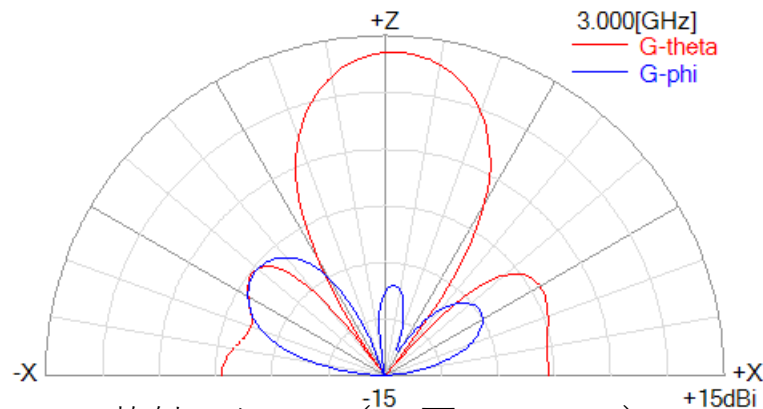
計算例(1) 1周波数、直線偏波+Z方向利得最大化

目的関数：

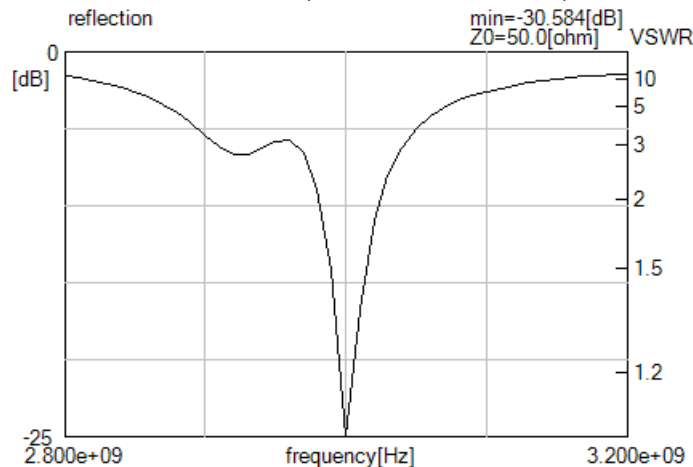
$$F = -G_{\theta}(\theta=0, \phi=0) + 0.03 |Z_{in} - Z_0|$$



アンテナ形状

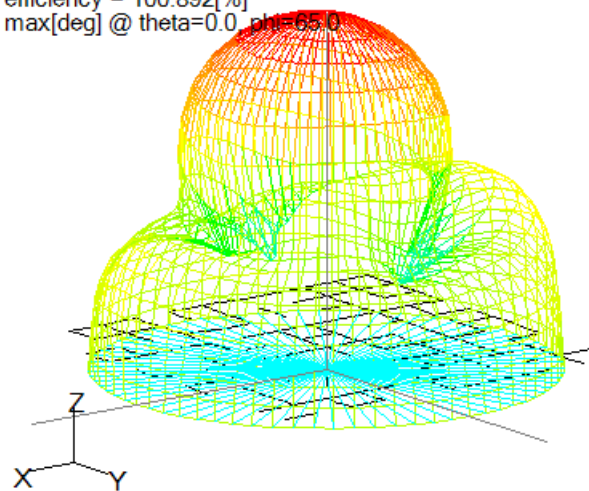


放射パターン (XZ面、3.0GHz)



反射係数 (2.8~3.2GHz)

E-abs[dB] f=3.000e+09[Hz]
directive gain = 13.510[dBi]
efficiency = 100.892[%]
max[deg] @ theta=0.0, phi=65.0

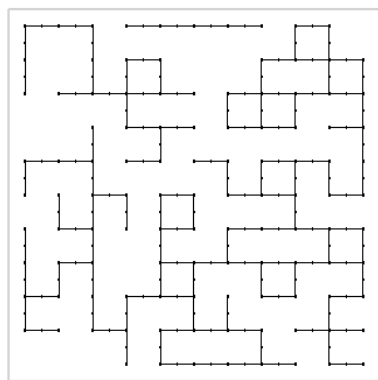


全方向放射パターン (3.0GHz)

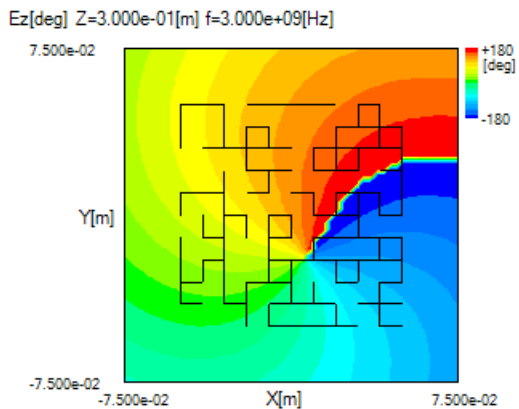
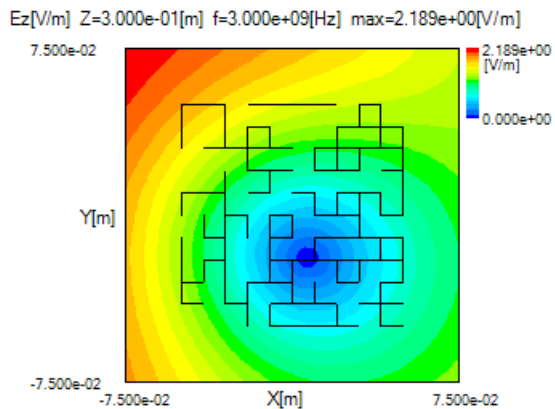
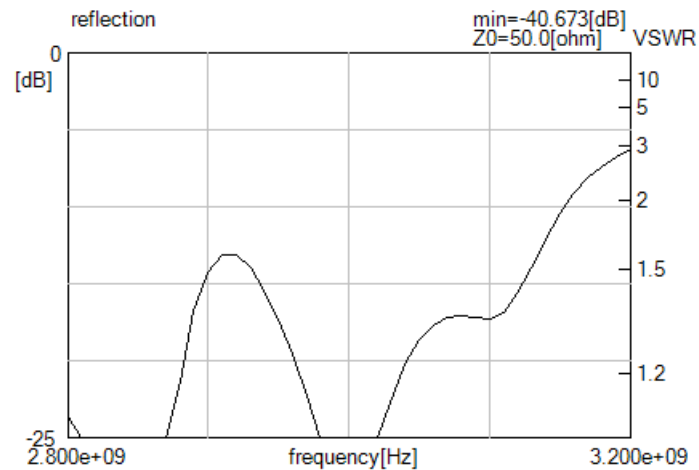
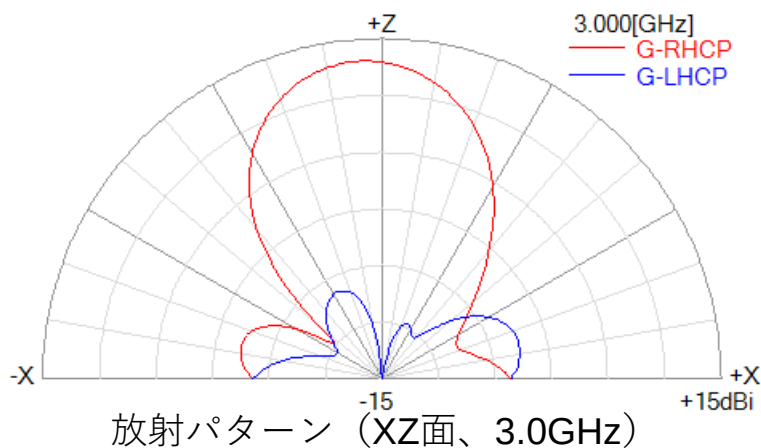
計算例(2) 1周波数、円偏波+Z利得最大化

目的関数：

$$F = -G_{RHCP}(\theta=0, \phi=0) + 0.03 |Z_{in} - Z_0|$$



アンテナ形状

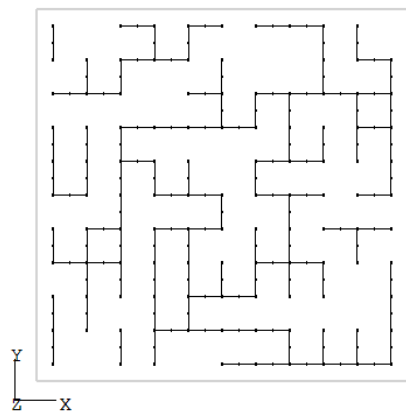


Ez分布 (左：振幅[V/m]、右：位相[度]、Z=300mm面、3.0GHz)

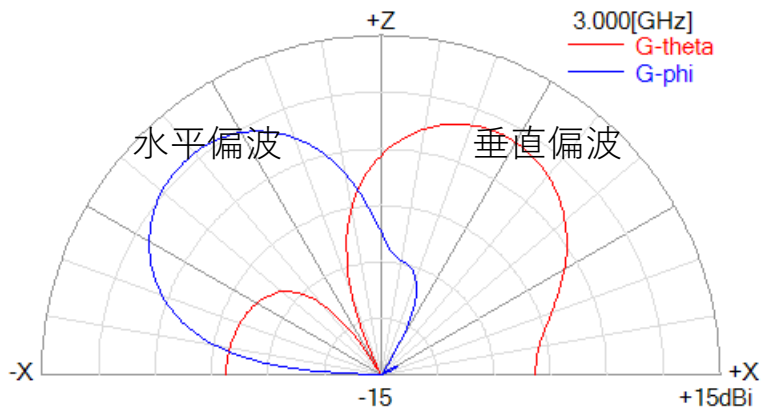
計算例(3) 1周波数、垂直水平偏波2ビーム

目的関数：

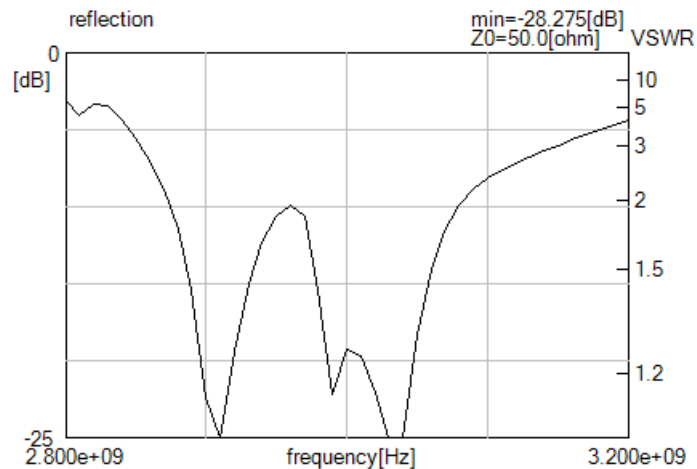
$$F = -\frac{1}{2} [G_{\theta}(45, 0) + G_{\phi}(45, 180)] + 0.02 |Z_{in} - Z_0|$$



アンテナ形状



放射パターン (XZ面、3.0GHz)

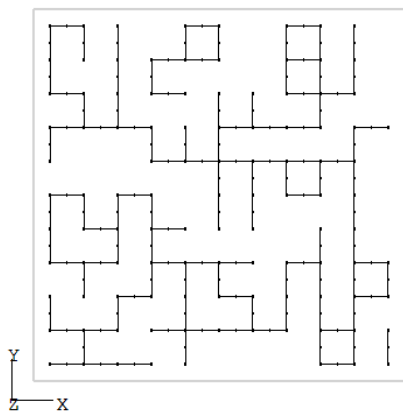


反射係数 (2.8~3.2GHz)

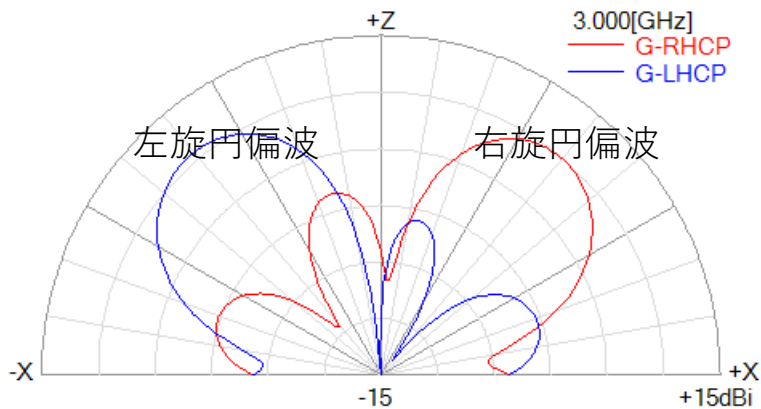
計算例(4) 1周波数、左右円偏波2ビーム

目的関数：

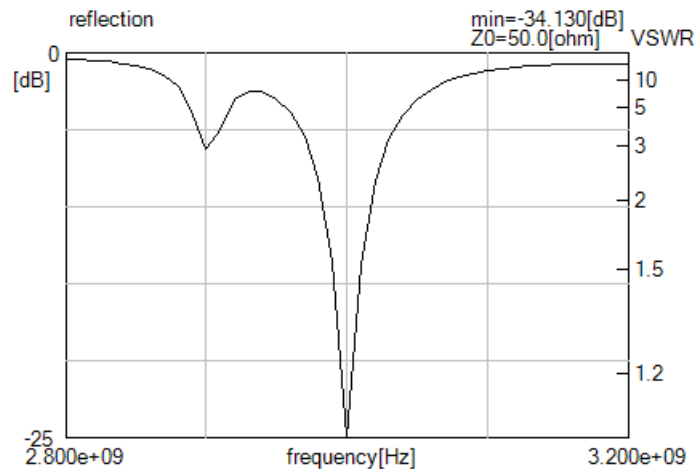
$$F = -\frac{1}{2} \{G_{RHCP}(45, 0) + G_{LHCP}(45, 180)\} + 0.02 |Z_{in} - Z_0|$$



アンテナ形状



放射パターン (XZ面、3.0GHz)

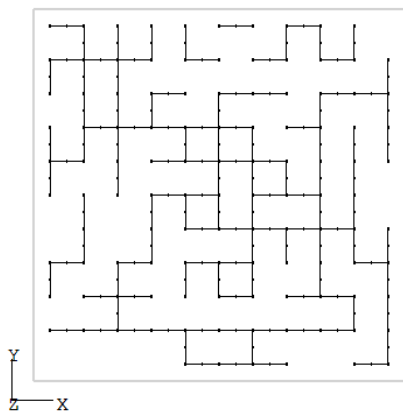


反射係数 (2.8~3.2GHz)

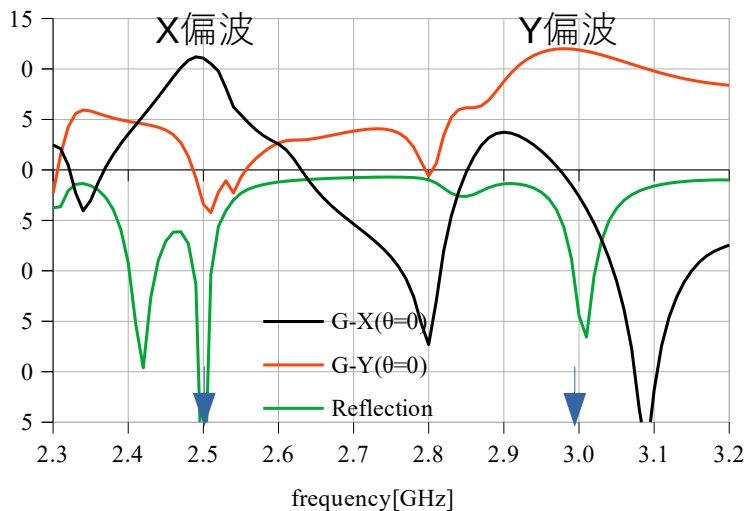
計算例(5) 2周波数、垂直水平偏波

目的関数：

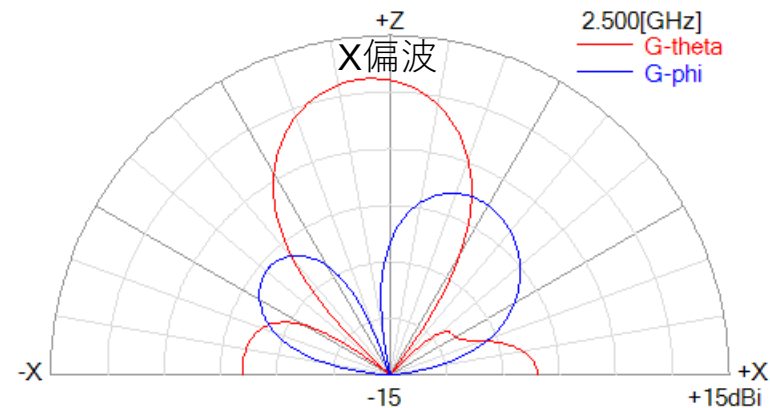
$$F = -\{G_\theta(f_1) + G_\phi(f_2)\} + 0.03\{|Z_{in}(f_1) - Z_0| + |Z_{in}(f_2) - Z_0|\}$$



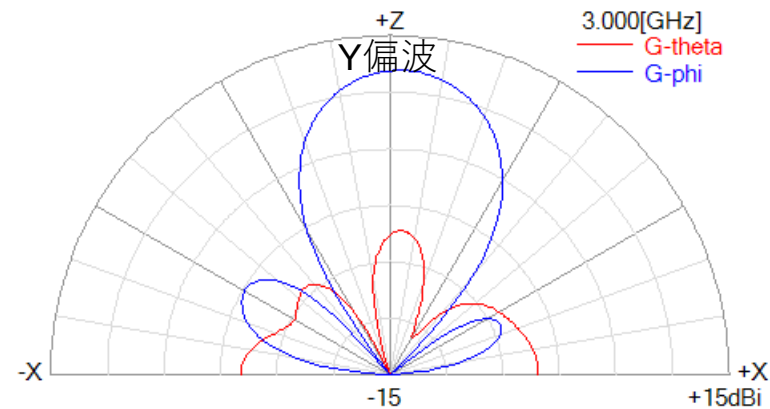
アンテナ形状



利得と反射係数の周波数特性



2.5GHz放射パターン (XZ面)

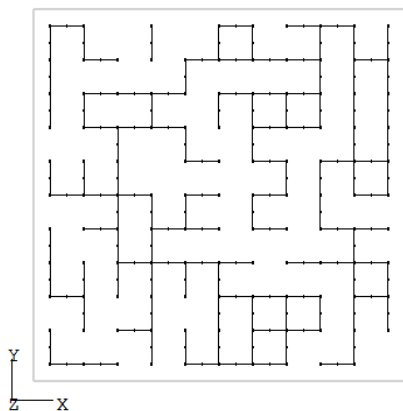


3.0GHz放射パターン (XZ面)

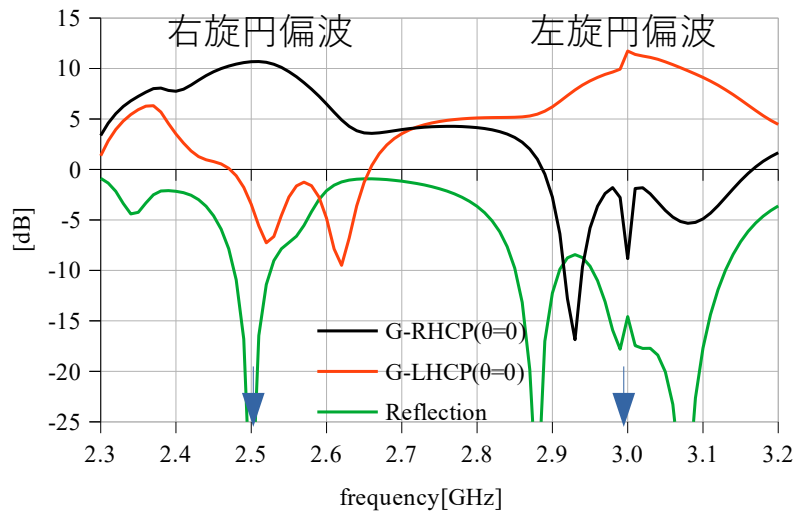
計算例(6) 2周波数、左右円偏波

目的関数：

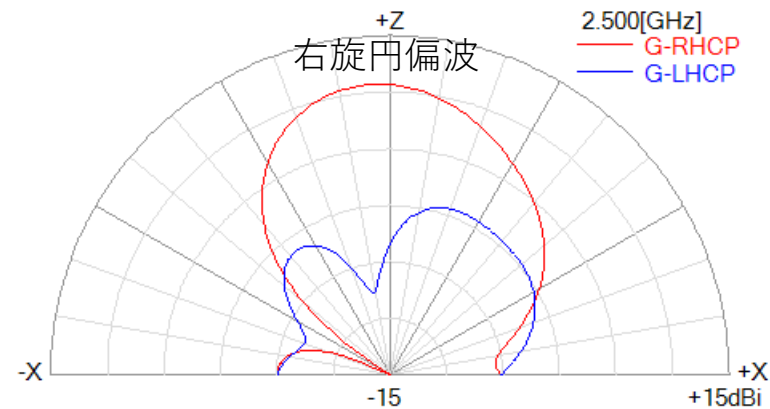
$$F = -\{G_{RHCP}(f_1) + G_{LHCP}(f_2)\} + 0.03\{|Z_{in}(f_1) - Z_0| + |Z_{in}(f_2) - Z_0|\}$$



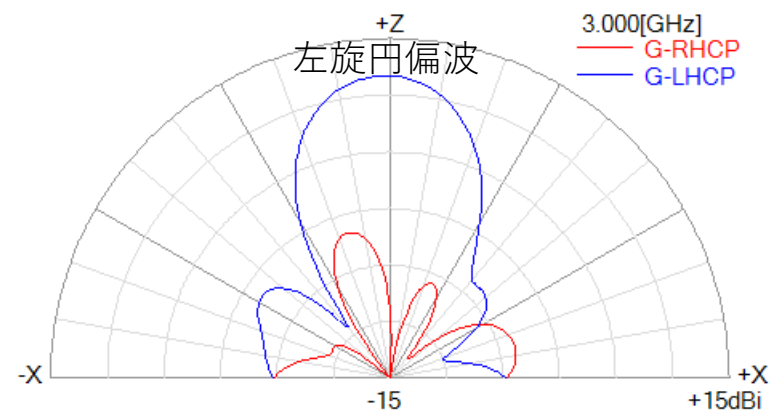
アンテナ形状



利得と反射係数の周波数特性



2.5GHz放射パターン (XZ面)

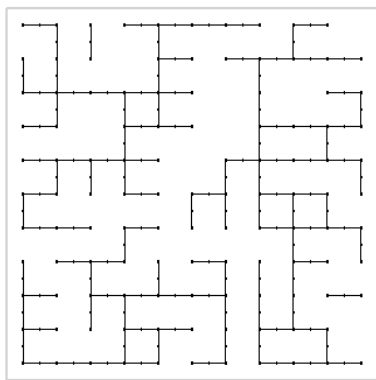


3.0GHz放射パターン (XZ面)

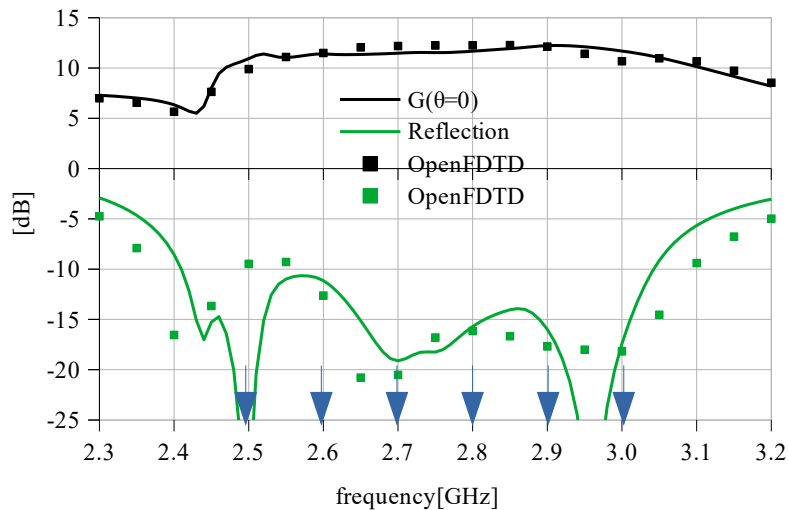
計算例(7) 広帯域、+Z高利得

目的関数：

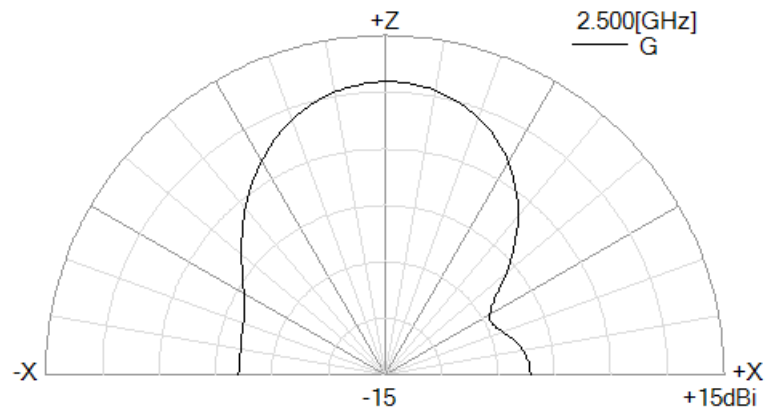
$$F = \sum_i \{-G(\theta=0, f_i) + 0.04 |Z_{in}(f_i) - Z_0|\}$$



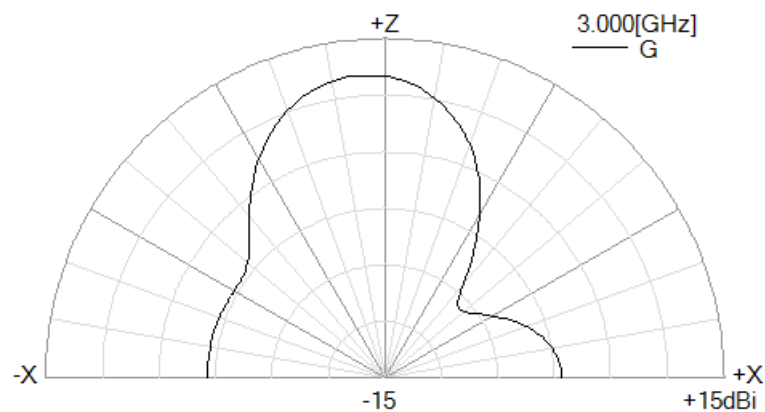
アンテナ形状



利得と反射係数の周波数特性



2.5GHz放射パターン (XZ面)

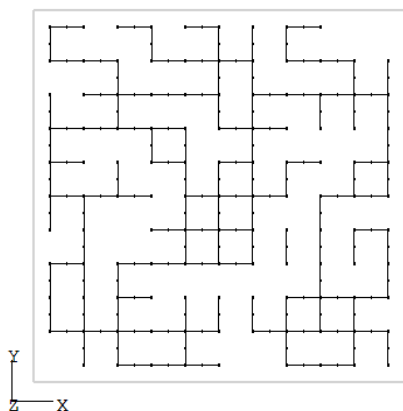


3.0GHz放射パターン (XZ面)

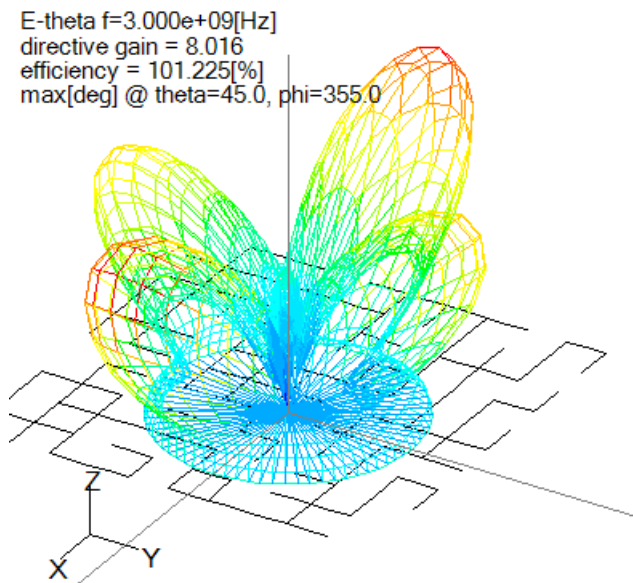
計算例(8) 1周波数、4ビーム、垂直偏波

目的関数：

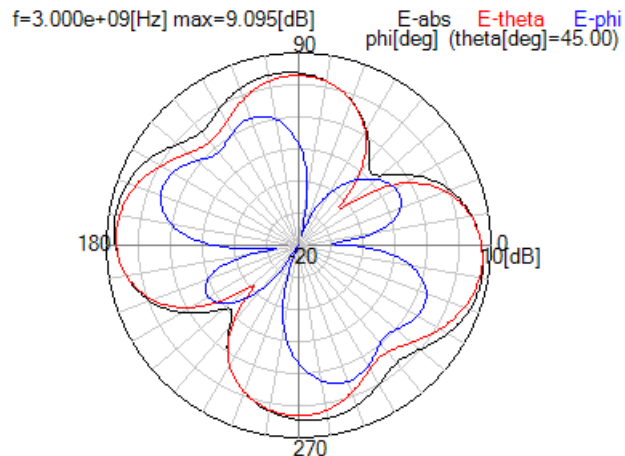
$$F = -\frac{1}{4} \{G_{\theta}(45,0) + G_{\theta}(45,90) + G_{\theta}(45,180) + G_{\theta}(45,270)\} + 0.02 |Z_{in} - Z_0|$$



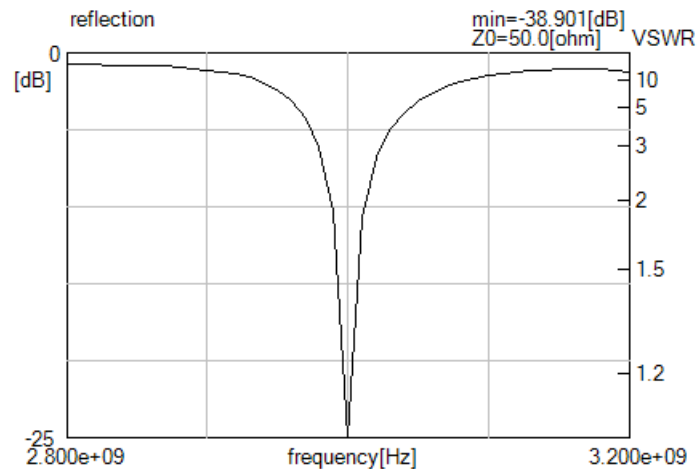
アンテナ形状



全方向放射パターン



θ=45度面放射パターン



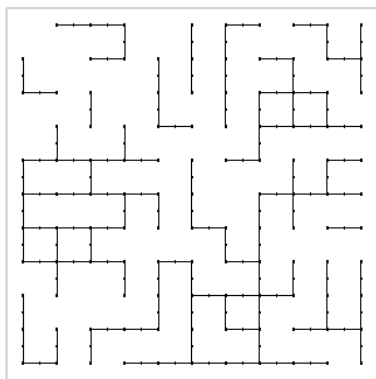
反射係数 (2.8~3.2GHz)

計算例(9) 1周波数、コニカルビーム、水平偏波

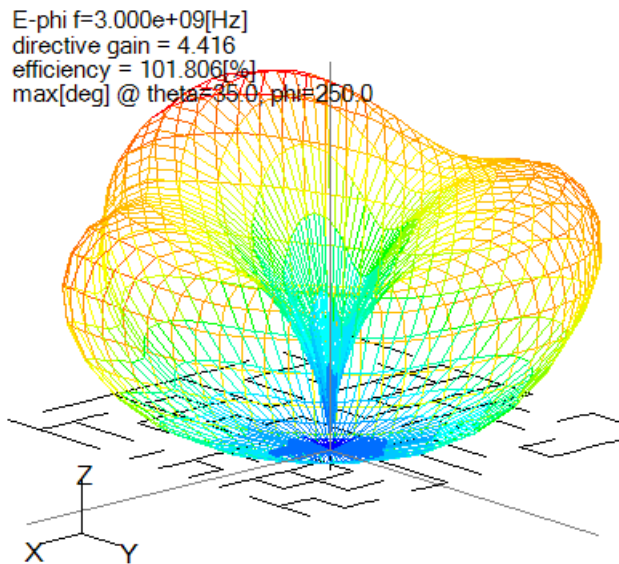
目的関数：

$$F = -\frac{1}{N_\phi} \sum_i G(45, \phi_i) + 0.3 |G_{max} - G_{min}| + 0.02 |Z_{in} - Z_0|$$

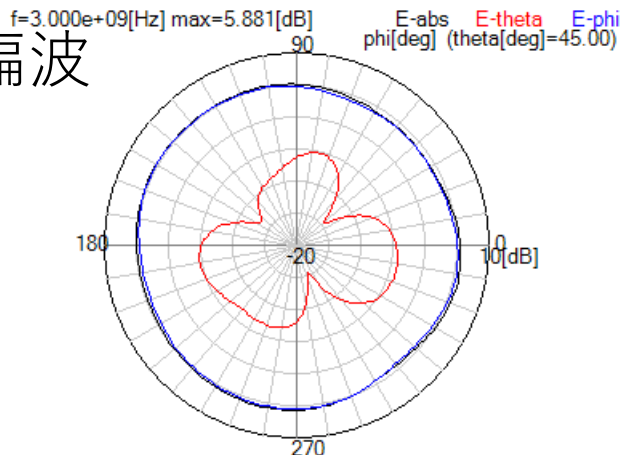
$$G_{max} \equiv \max_i G(45, \phi_i), \quad G_{min} \equiv \min_i G(45, \phi_i), \quad \phi_i = 360 \frac{i}{N_\phi}, \quad N_\phi = 36$$



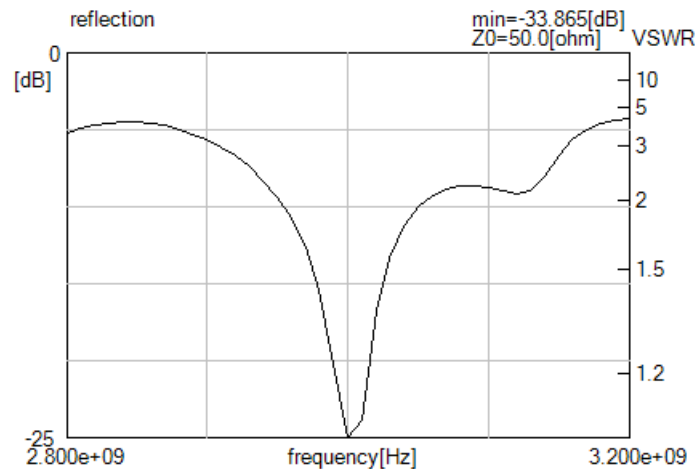
アンテナ形状



全方向放射パターン



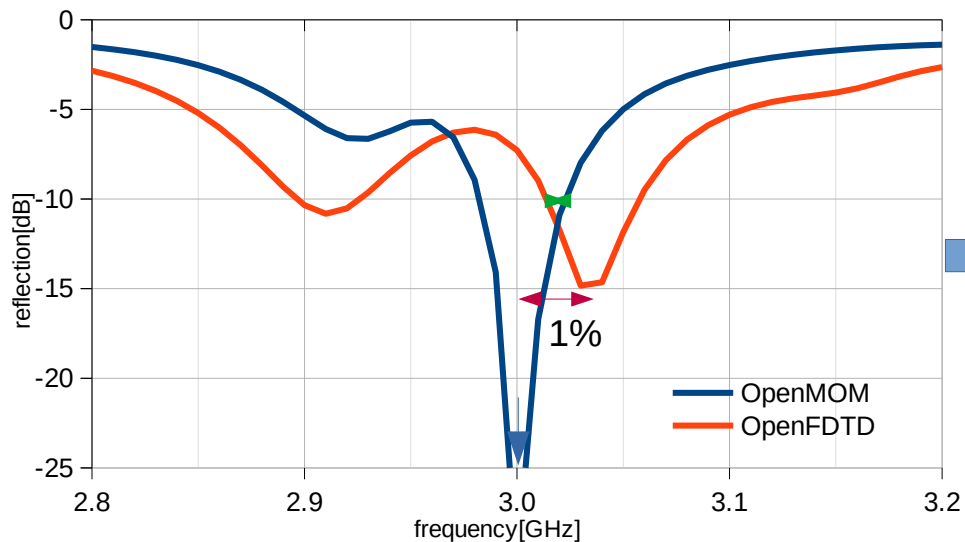
θ=45度面放射パターン



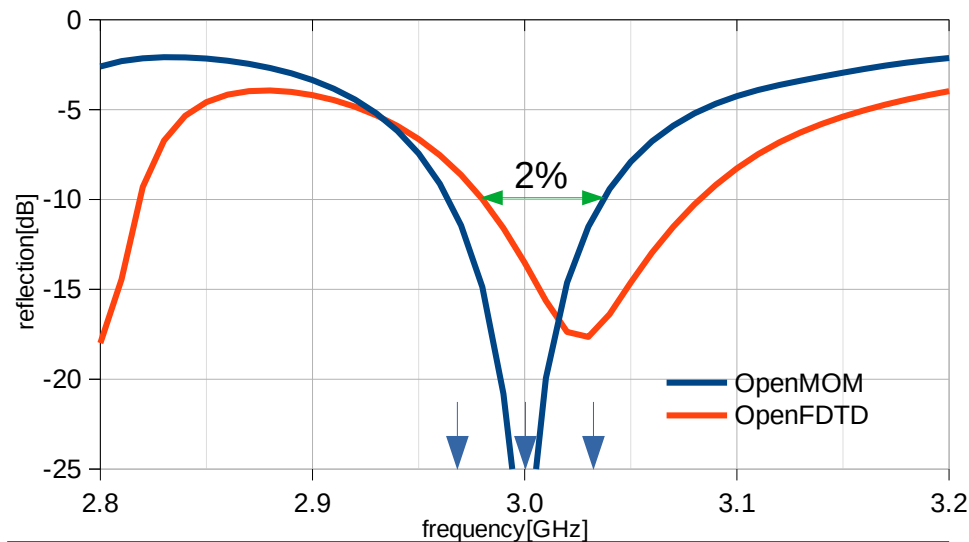
反射係数 (2.8~3.2GHz)

計算例(10) 整合特性の改善

目的関数： $F = -G_{\theta}(\theta=0, \phi=0) + 0.03 |Z_{in} - Z_0|$



目的関数： $F = \sum_i \{-G_{\theta}(\theta=0, \phi=0, f_i) + 0.03 |Z_{in}(f_i) - Z_0|\}$



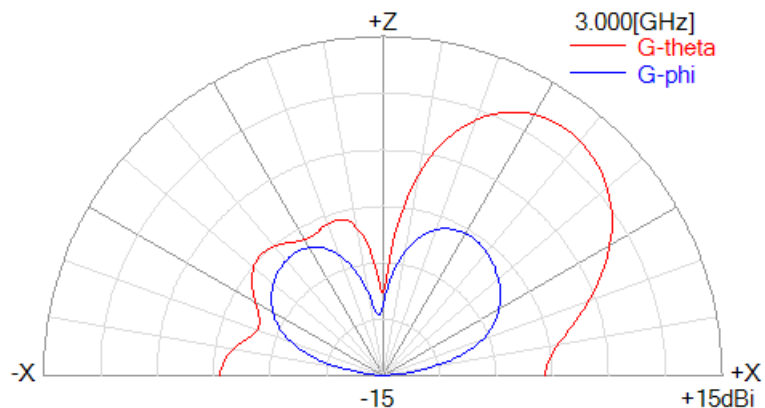
- ・ 出力されたアンテナをOpenMOMとOpenFDTDで計算した結果
- ・ 共振周波数に1%の誤差がある
- ・ 反射係数が-10dB以下になる共通部分がない
- ・ 本手法はアンテナ形状の微調整に不向きであり、どこを調整すればよいかも不明

- ・ 3つの周波数で最適化したアンテナの特性
- ・ 両者の帯域幅が広いので、反射係数が-10dB以下になる共通帯域幅が2%ある
- 共振周波数の誤差を見越してある程度の帯域を持たせて最適設計したほうがよい

計算例(11) サイドローブの抑制

目的関数：

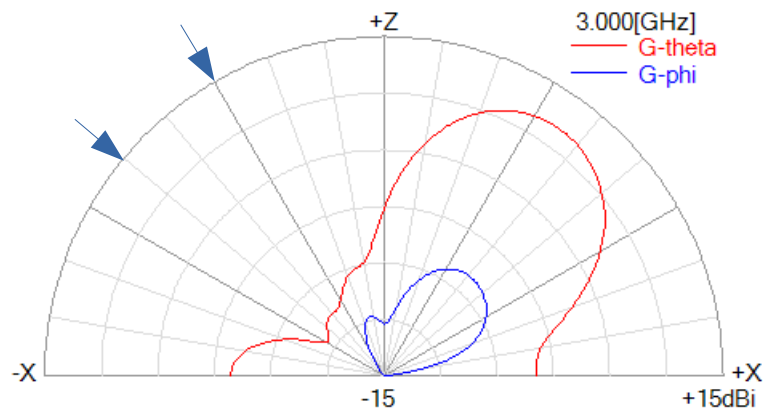
$$F = -G_{\theta}(45, 0) + 0.01 |Z_{in} - Z_0|$$



- 右45度の利得を最大化したときの放射パターン
- 反対方向にサイドローブがしやすい

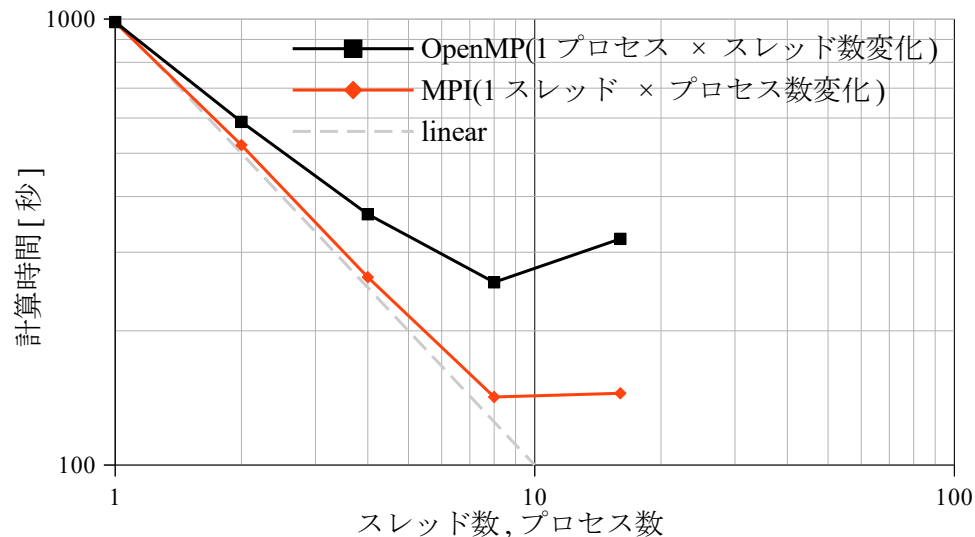
目的関数：

$$F = -G_{\theta}(45, 0) + 0.02 \{G(30, 180) + G(50, 180)\} + 0.01 |Z_{in} - Z_0|$$

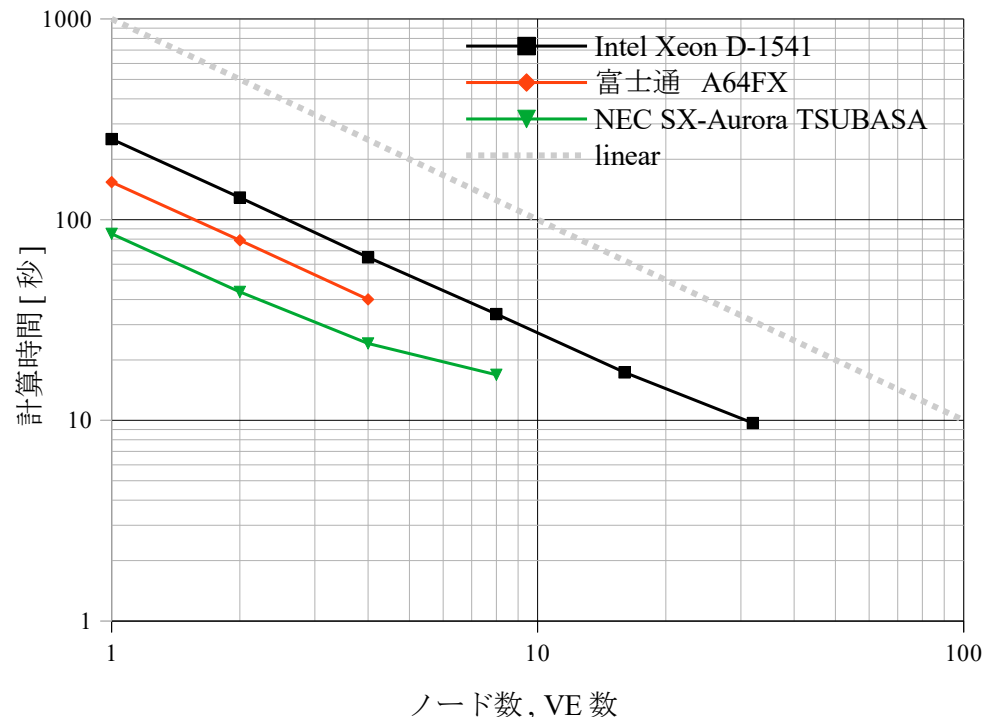


- サイドローブ周辺の利得を小さくしたときの放射パターン
 - サイドローブが小さくなることが確認できる
- ※利得を小さくするときは単位はlinearとし重みの符号を反対にする
 ※両偏波成分の利得を小さくするために合成利得Gを考えている
 ※目的関数の項数を増やしても計算時間は変わらない

3.2 計算時間

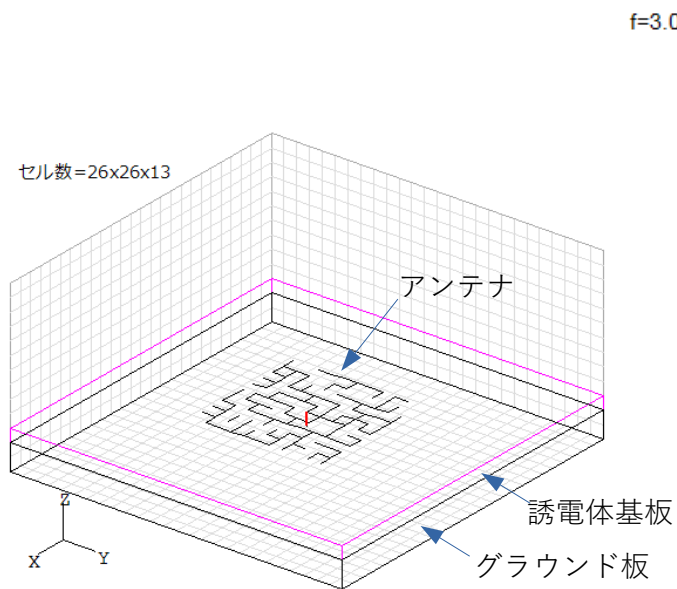


スレッド数, プロセス数と計算時間の関係
 Intel Xeon D-1541 (8コア16スレッド×1CPU)
■リソースが限られるときはMPIで試行回数を分割したほうがよい



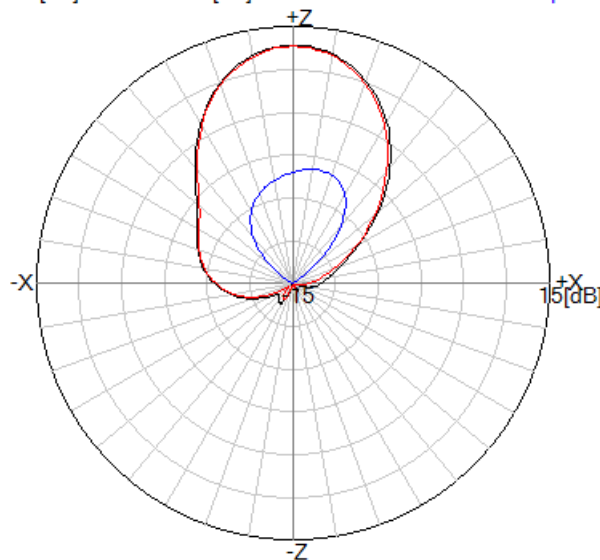
ノード数と計算時間の関係
 Intel : 8スレッド×1プロセス/ノード
 富士通: 12スレッド×4プロセス/ノード
 NEC : 1スレッド×8プロセス/VE (8VEのときは2×4)
■MPIにより高い並列性能が得られる

4. FDTD法による最適設計

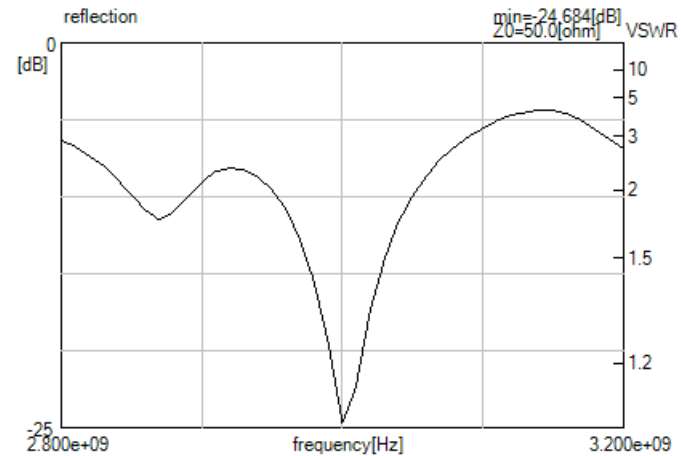


アンテナ形状

f=3.000e+09[Hz] max=12.922[dB] E-abs E-theta E-phi



放射パターン (XZ面、3.0GHz)
利得=12.9dBi



反射係数 (2.8~3.2GHz)

- FDTD法(OpenFDTD)を用いた最適設計も実装して公開している
- OpenFDTDでは誘電体も扱える
- 同じ問題ではモーメント法(OpenMOM)と比べて2桁計算時間がかかるので実用性に乏しい

5. まとめ

- ・アンテナ形状をランダムに変形し目的関数が小さくなる変形を採用する操作を繰り返すことによって最適形状を得る方法について述べた.
- ・初期配置をランダムに変えて多数回試行することによって局所解を避けることができる.
- ・本アンテナモデルはアンテナの各種の特性を最適化するだけの表現能力を持っている.
- ・計算にはモーメント法を用いた.
- ・最適形状を得るためには**10万回**のオーダーの計算が必要であるがクラスタ環境において**MPI**を用いて並列計算すると計算時間は**10秒**のオーダーにすることができる(**Intel CPU 256コア時**).
- ・本プログラムのソースコードは**OpenMOM**の付属ツールとして公開している.
- ・本ケースと異なるアンテナモデルについても少しの変更で対応することができる.
- ・本方法はシミュレーターと計算モデルを取り替えることによって各種の問題に応用することができる

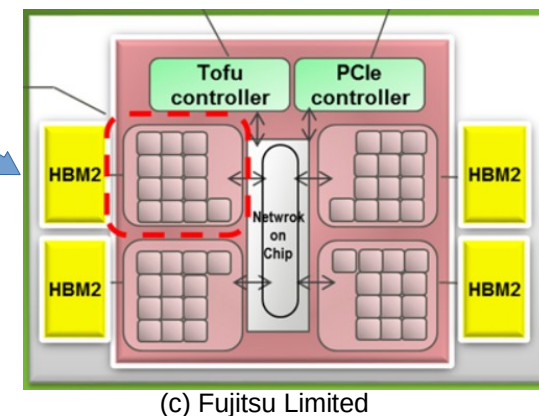
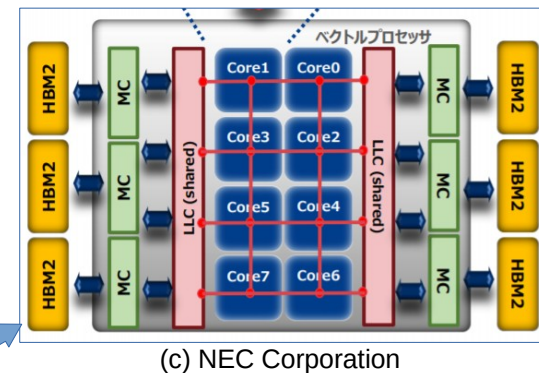
[3] OpenMOM : <http://www.e-em.co.jp/OpenMOM/>

[4] OpenFDTD : <http://www.e-em.co.jp/OpenFDTD/>

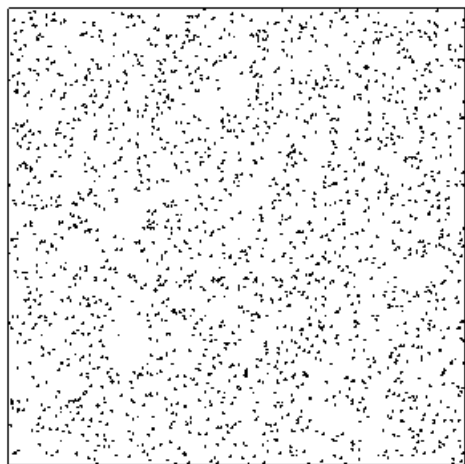
参考(1) 高速化プログラミング

| 機種 (ハードウェア) | ベクトル化 | 共有メモリー並列 | 分散メモリー並列 |
|--|---------------------|-----------------|----------|
| Intel/AMD (x86-64) | SIMD 組み込み関数 | OpenMP マルチコア | MPI |
| NEC SX-Aurora TSUBASA (Vector Engine) | 自動ベクトル化 ベクトル長256 | OpenMP 8コア | MPI |
| Fujitsu Supercomputer PRIMEHPC (A64FX) | 自動ベクトル化 ベクトル長16 | OpenMP 48コア | MPI |
| NVIDIA (GPU) | CUDA | | MPI |

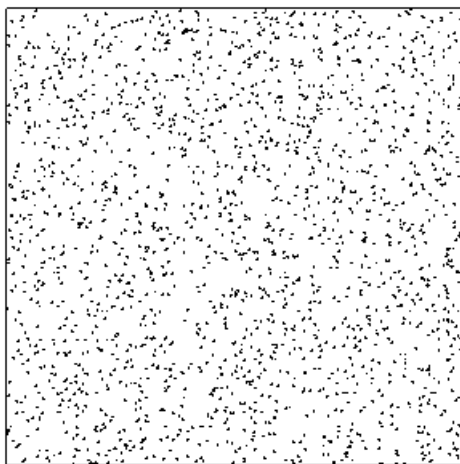
本方法で使した部分



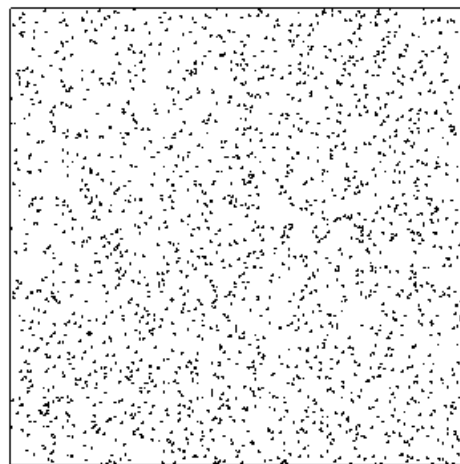
参考(2) 乱数について



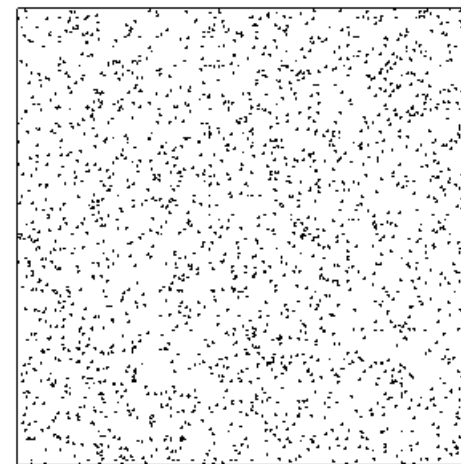
乱数1



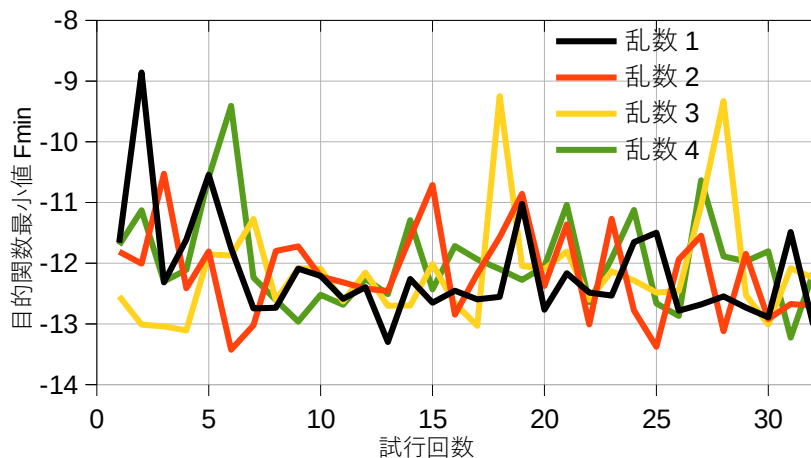
乱数2



乱数3



乱数4



乱数を4000個発生させ順にX,Y座標として2000点をプロットしたもの

(1)乱数1: コンパイラVC++付属のrand関数

(2)乱数2: [1]Park and Miller (線形合同法)

(3)乱数3: [5]森正武 (線形合同法)

(4)乱数4: [1]Knuth

試行回数32回で比べるとほぼ同程度と言える

■本方法では乱数は何でもよい

参考(3) シミュレーターを取り替えるための作業

本方法は比較的容易な作業でシミュレーターを取り替えることができる

作業内容と注意点は以下の通り

- (1) 組み合わせ最適化のパラメーターをシミュレーターの入力データに変換する
- (2) シミュレーターの入力部、出力部（図形出力等）を削除する
- (3) シミュレーターの余分な機能はすべて削除する
- (4) `malloc`はコストがかかるので最初に一度行い計算の途中では行わない
- (5) `OpenMP`等のスレッド起動はコストがかかるので回数をなるべく減らす
- (6) 計算の途中では`MPI`の通信を行わない