

Panoramic v7 TRIG & TEMPESTpr2 スーパーグリッド

高精度且つ高速な
DUVリソグラフィ
シミュレーション

課題：大規模な2Dパターンのシミュレーションを、エッジの位置を細かく制御して行いたい（小さなグリッドサイズ）

HyperLith v6ではこれは困難でした：

- RCWA はグリッドサイズに合わせてサイズ調整をすることはできますが、領域サイズには上手く対応できません。
- FDTDは領域サイズには対応できますが、グリッドサイズには上手く対応できません。
- 領域サイズ、グリッドサイズの両方に対応できる厳密なマックスウェルソルバーが必要とされています。

HyperLith v7での2つのソリューション：

1. TRIG: 新たに導入された厳密なシミュレータ
 - a. 厳密、正確、高速
 - b. 領域サイズについてはFDTDのようにサイズ調整が可能（良好）
 - c. グリッドサイズについてはRCWAのようにサイズ調整が可能（良好）
2. FDTD用のサブグリッド分解能機能
 - a. 細かいグリッド上に形状（パターン）を定義
 - b. 粗いグリッド上でFDTDを実行
 - c. 領域サイズ、グリッドサイズの両方について上手くサイズ調整を行うことができる

TRIG

Panoramic Technology 社が新たに提供する、**厳密且つ高速な3D**
マックスウェルシミュレータ

TRIG シミュレータ

- 厳密で高精度な3Dマックスウェルソルバー
- 厳密なEUVについて超高速化を実現（20～30倍）
- DUVについてはTEMPESTTpr2より高速（2～5倍）
- グリッドサイズの縮小に対応
- ドメインサイズの拡大に対応
- FDTDほどメモリーを使わない
- （将来の）NVidia GPU（HSS）バージョンでは超高速化を実現
- バージョン7の別途オプションとして使用可能

TEMPESTpr2

スーパーグリッド

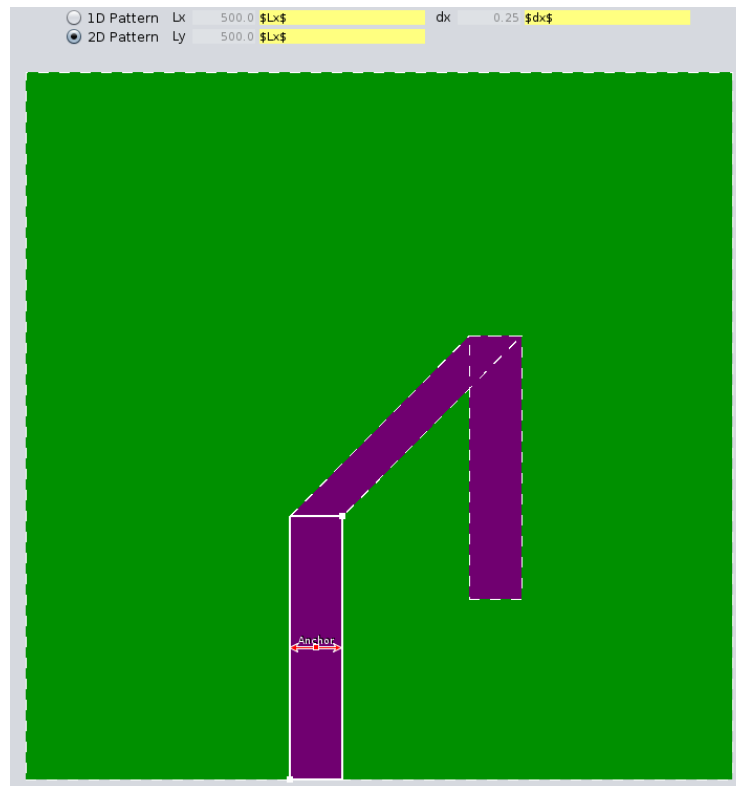
FDTD/TEMPESTpr2用
サブグリッド分解能

TEMPESTpr2/FDTD スーパーグリッド

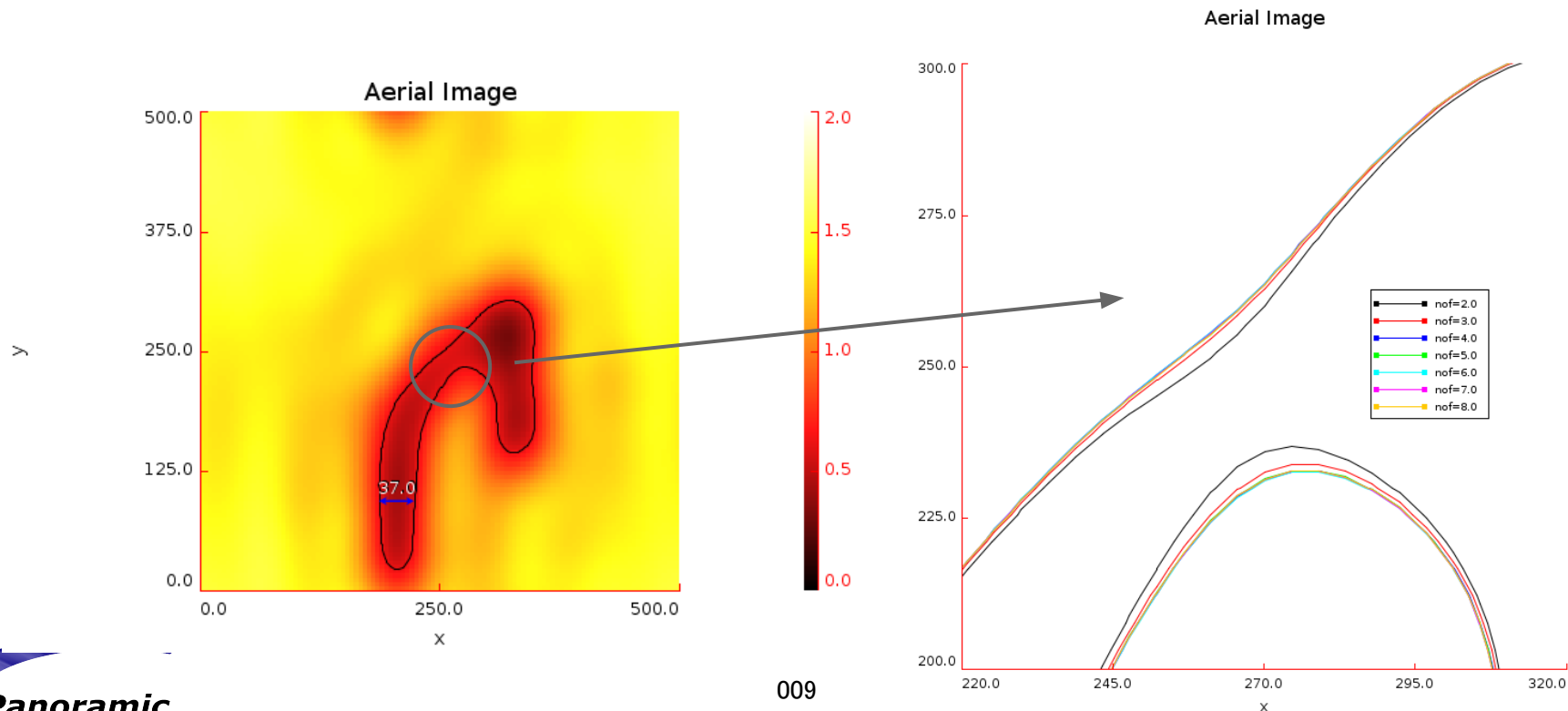
- 形状を細かいグリッド上に定義可能。FDTDは粗いグリッド上で実行
- グリッド割りに問題のあるDUVに有用
- 時間とメモリの節約になる（4～16倍の節減）

500nm x 500nm のベンチマーク

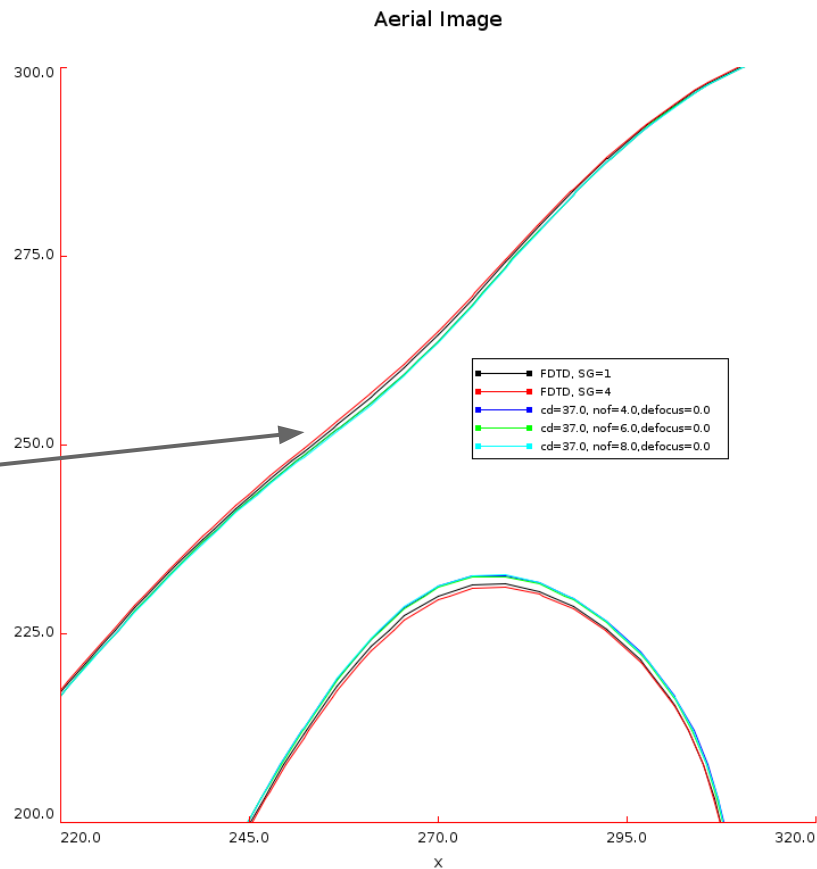
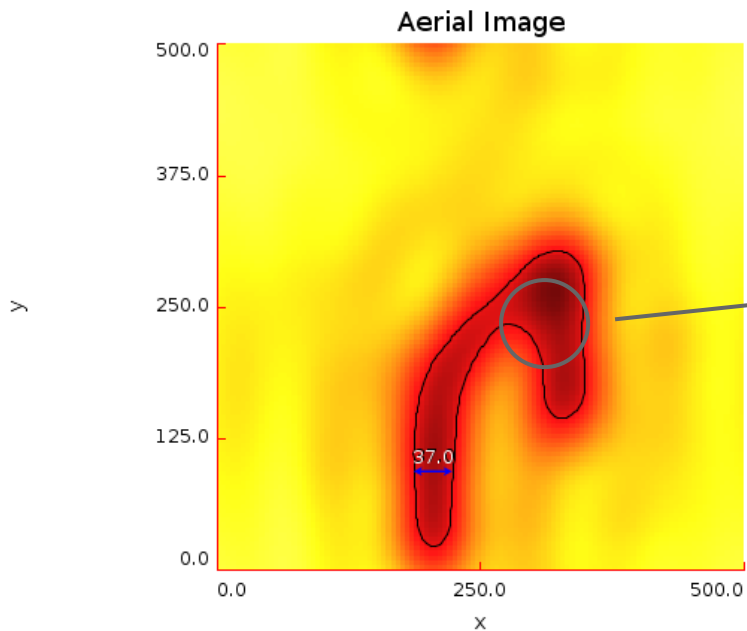
$dx=0.25\text{nm}$ - gives 0.25nm のエッジ配置の精度 (ウエハーユニット)



TRIG の収束 vs. NOF



TRIG vs FDTD



010

タイミングの結果

v7 TRIG, NOF=4.0, dz=3.7778nm: 163s

v7 TRIG, NOF=6.0, dz=3.7778nm: 193s

v7 FDTD, SG=4, dz=3.7778nm: 238s

v7 FDTD, SG=1, dz=3.7778nm: 11,300s (v6ではもっと遅くなります)

v7 RCWA, NOF=4 (大きすぎます!)

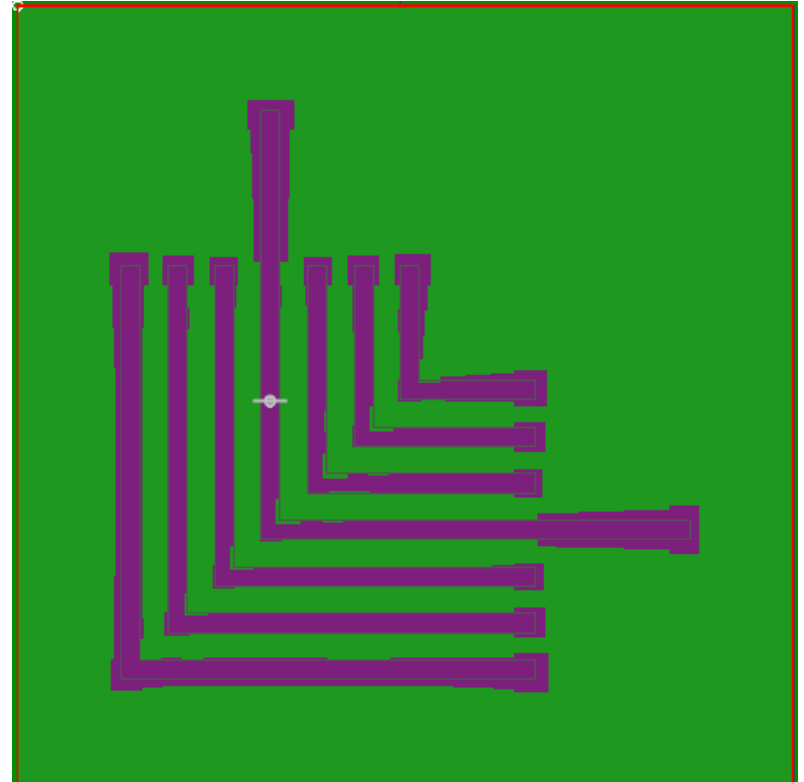
v7 RCWA, NOF=2: 2038s (v6ではもっと遅くなります)

500x500のベンチマークの結論

- NOF=4 (163s)のTRIGは正確
- SG=4 (238s) のFDTDは、SG=1 (11,300s) のFDTDとほぼ同じ解に
- TRIGもSGのFDTDも共に、高いエッジ配置精度で500x500の領域のシミュレーションを行うことが可能
- NOF=4以上の場合にもTRIGは良好に収束すること、また、ここには記されていませんが、その他の要因を考慮すると、TRIGの方がFDTDより精度が高いと考えられます。

等密度エルボーのベンチマーク

- 1500nm x 1500nm (ウエハー)
- 37nm nominal feature size
- dx=0.5nm (ウエハー)
- Att. PSM (68nmの吸収材)
- パターンはPanOPCで作成
- アンカーポイントは真ん中のライン上に表示されています
- これは大規模なシミュレーションになります!

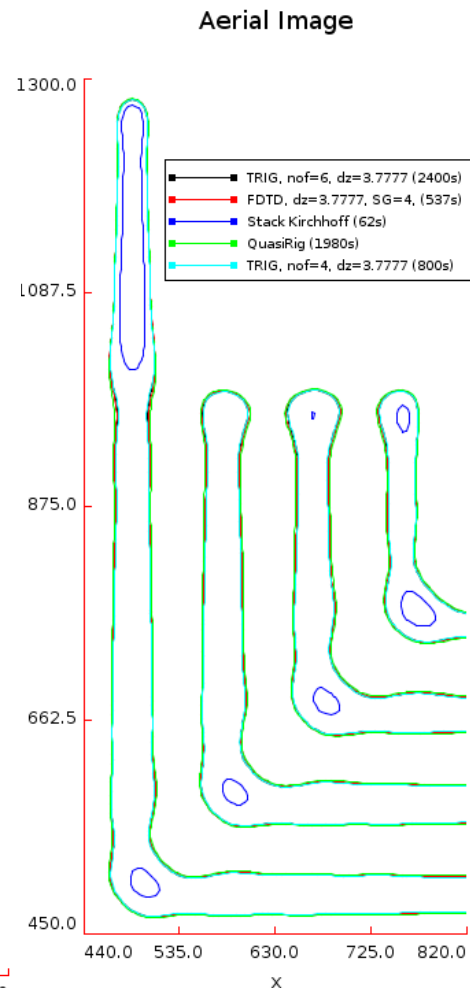
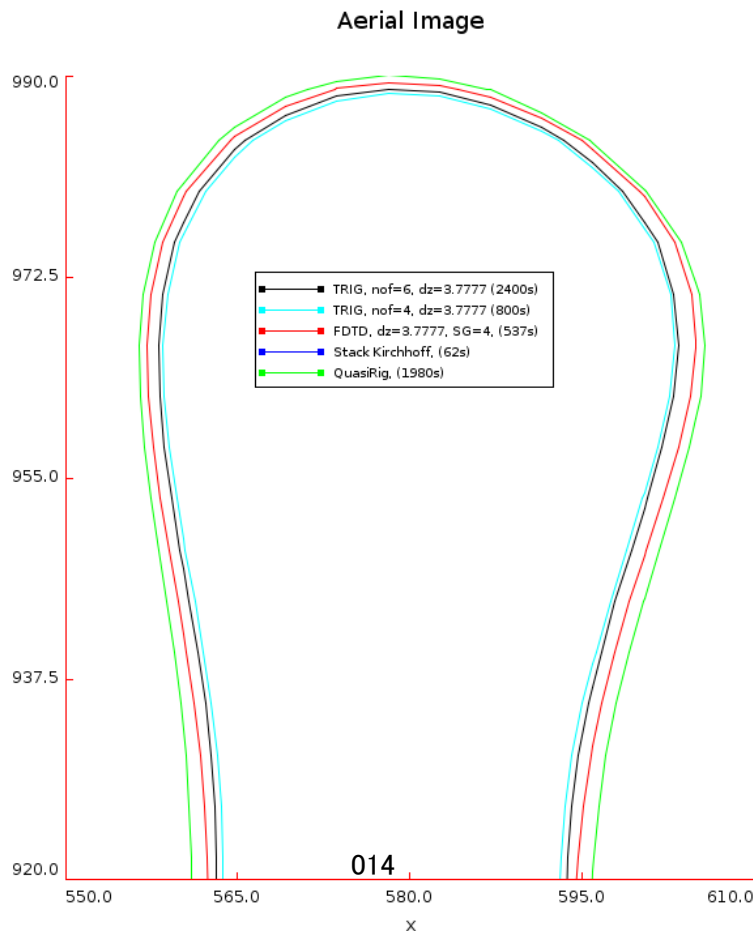


空中像が固定されていない場合

*適切な解は、
黒と赤（最良のTRIG
と最良のFDTD）の間
にあると考えられま
す。

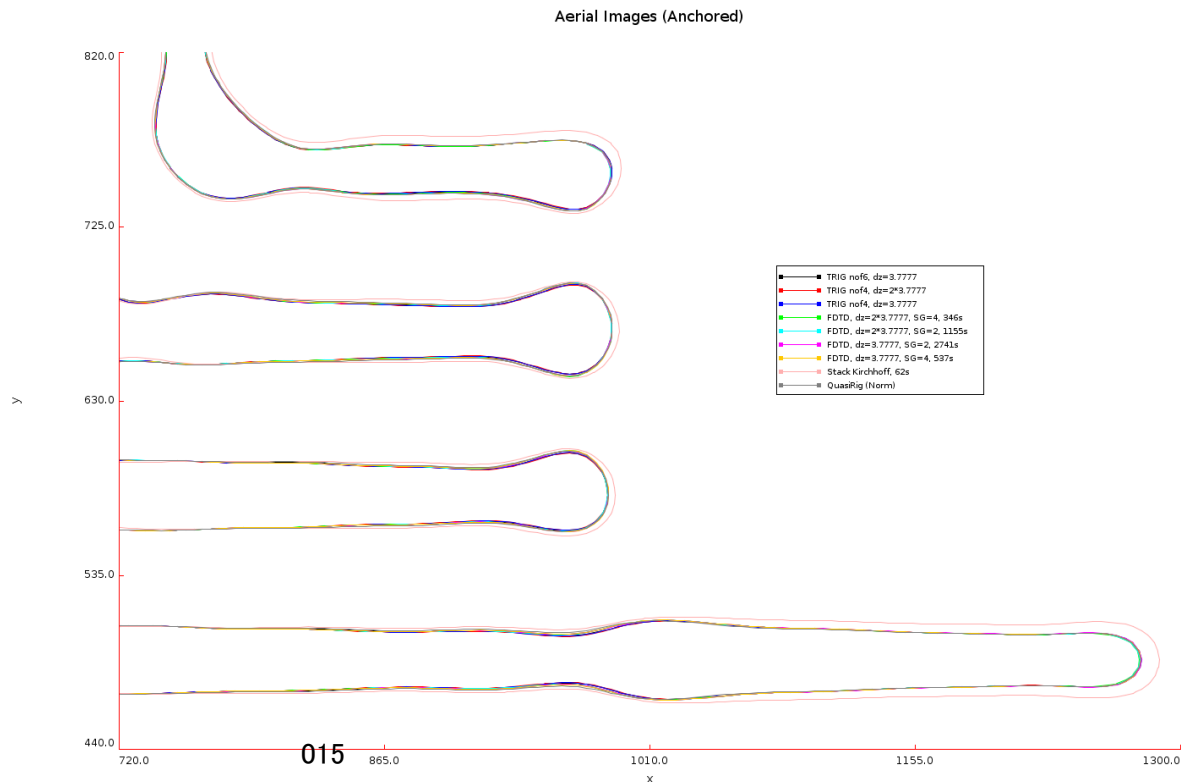
*Stack キルヒホッフ -
よくありません。 .

*QuasiRig - OK で
すが、TRIG (nof=4)
やFDTD (スーパーグ
リッド=4) と比べると
遅いです。



空中像が固定されている場合（真ん中のラインの幅に固定）

- Stack-キルヒホッフ以外、（固定されている）像はすべて一致しています。



等密度エルボアのベンチマークの結論とコメント

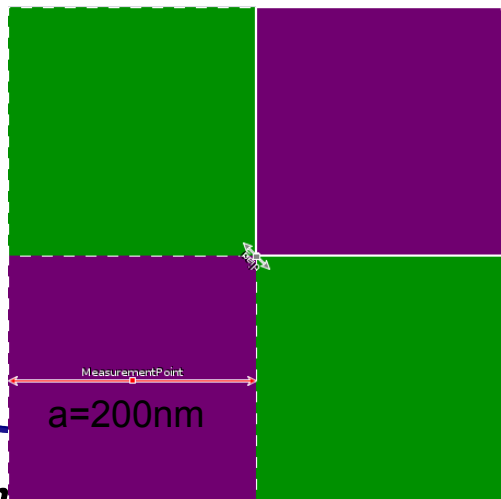
- Stack-キルヒホッフは精度が低い（ただしスピードは速い）
- TRIGとTEMPESTpr2スーパーグリッドは良好な解を導き出すことができる
- QuasiRigは精度があまり高くなく、速度に関してもTRIGやSGのFDTDより劣る
- Stack-キルヒホッフを除き、いずれの方法を使っても、空中像が固定されている場合には良好な結果が導き出される。

45度のラインのあるDUV

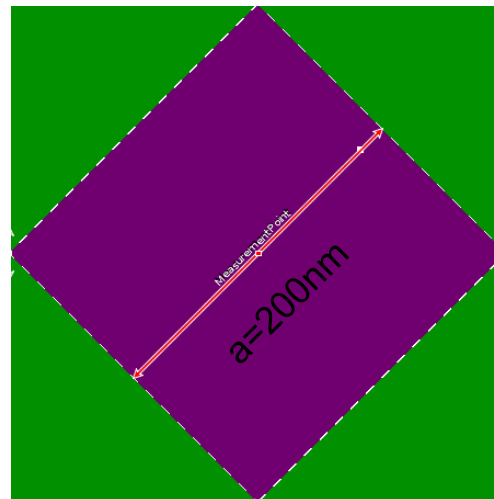
一部の方法では、45度のラインにうまく対応できません。

それを確かめるには、以下の2つの「チェックボード」パターンにより作成される空中像を比較するとよいでしょう:

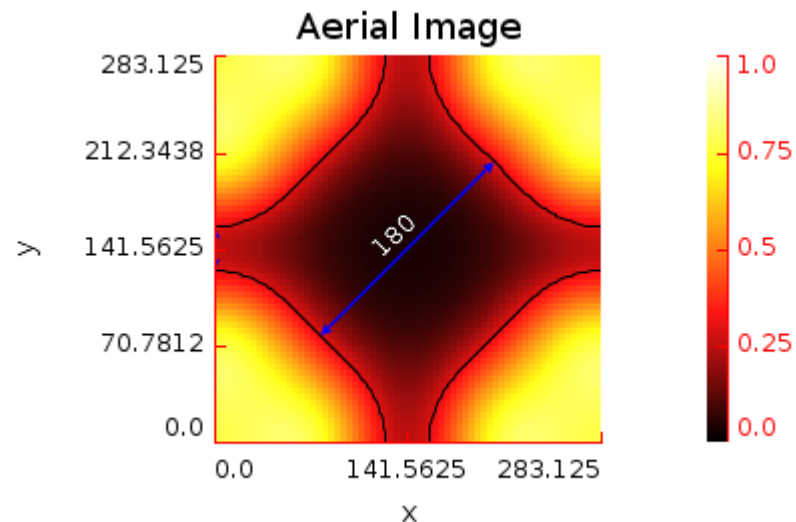
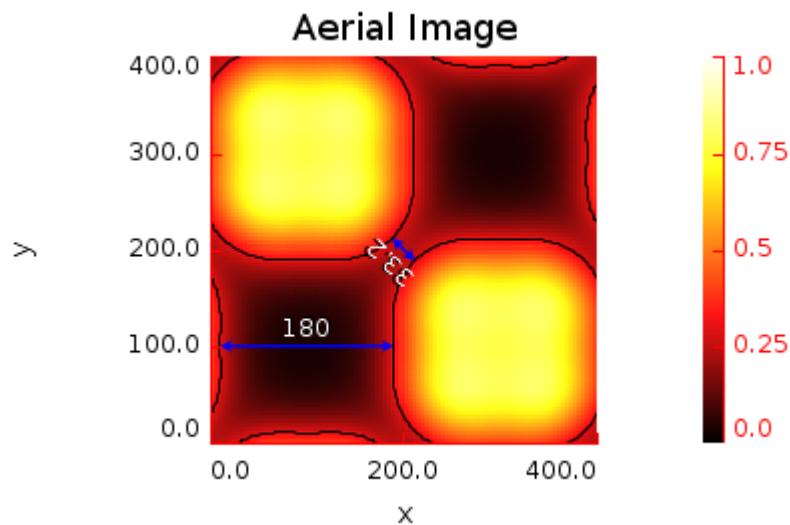
Case 0



Case 1



- 空中像は回転させるため、以下を比較するのが賢明です：
 - 最大強度
 - 最小強度
 - ラインの幅のCD
 - 角から角までのCD（ここではILSが低いため、あまり一致しないでしょう）



詳細...

- NA=1.35,
- 環状照明 (sig_in=0.7, sig_out=0.9), nsource_samples=200 (多く必要です！)
- 標準の “Att. PSM” マスク技術 (68nm n=2.343, k=0.586)
- Case 0の方がCase 1より大きい(ユニットセルの面積が2倍であるため)
- 6-スレッド, Xeon E5-2650
- v7 FDTDの定式化、しかしスーパーグリッドの関数=1 (サブグリッドを使用しない)


高精度のケース

- $dx=dy=0.625$ (ウエハーユニット),
- $dz=68/36\text{nm}=1.8888\text{nm}$ (マスクユニット)
- TRIG: $\text{NOF}=6.0$ (RCWAのようなNumber of Orders Factor)

	Case 0 TRIG	Case 1 TRIG	Case 0 FDTD	Case 1 FDTD
最大強度	0.8162	0.8149	0.8125	0.8105
最小強度	0.0356	0.0352	0.0356	0.0353
CD 幅 (nm)	179.67	180.00	180.07	180.52
CD 斜め (nm)	33.2	32.6	34.06	33.52
時間 (秒)	123	56	2500	1500
メモリ	0.3 GB	0.15 GB ₀₂₀	1 GB	0.5 GB

中精度のケース

- $dx=dy=0.625$ (ウエハーユニット) TRIG用
- $dx=dy=1.25$ (ウエハーユニット) FDTD用
- $dz=68/18\text{nm}=3.7777\text{nm}$ (マスクユニット) (TRIG および FDTD用)
- TRIG: NOF=4.0

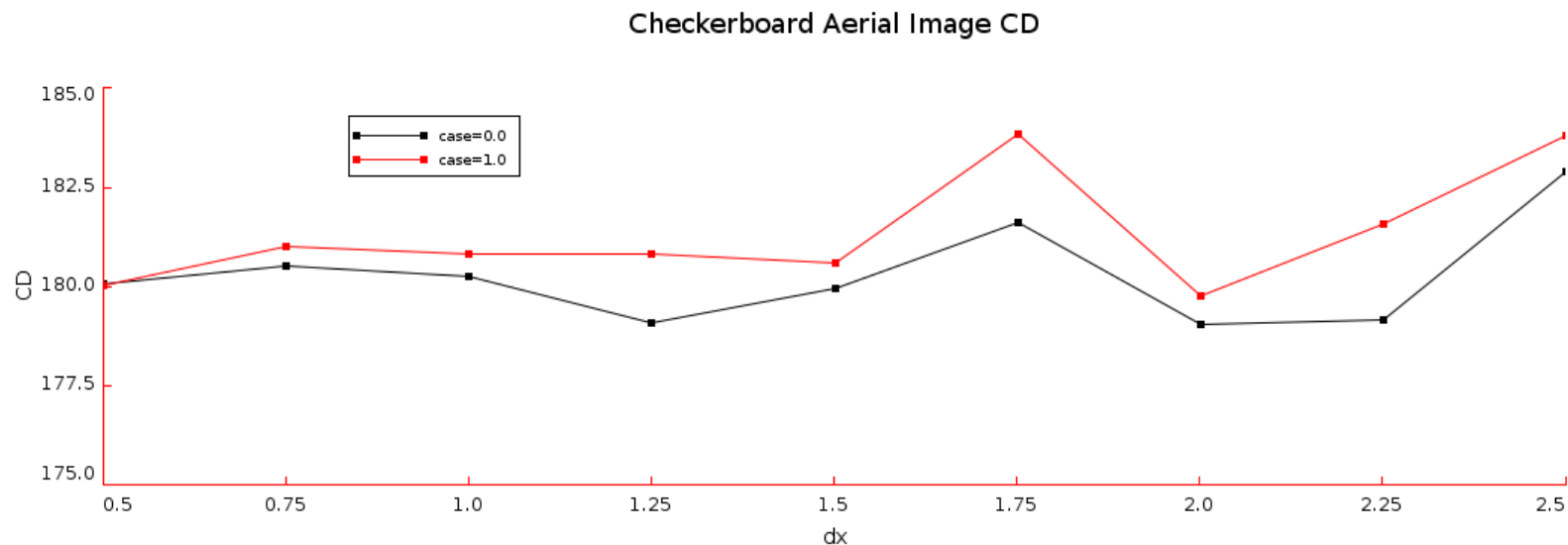
	Case 0 TRIG	Case 1 TRIG	Case 0 FDTD	Case 1 FDTD
最大強度	0.8177	0.8128	0.8109	0.8046
最小強度	0.0354	0.0350	0.0351	0.0348
CD 幅 (nm)	179.52	179.94	180.28	182.03
CD 斜め (nm)	32.85	34.60	34.31	36.32
時間 (秒)	43	20	210	92
 Anaplasmic Technology		021		

低精度

- $dx=dy=0.625$ (ウエハーユニット) for TRIG
- $dx=dy=2.5$ (ウエハーユニット) for FDTD
- $dz=68/9\text{nm}=7.5556\text{nm}$ (マスクユニット) (TRIG および FDTD)
- TRIG: NOF=2.0 (RCWAのようなNumber of Orders Factor)

	Case 0 TRIG	Case 1 TRIG	Case 0 FDTD	Case 1 FDTD
最大強度	0.8239	0.8207	0.8035	0.7989
最小強度	0.0349	0.0343	0.0335	0.0332
CD 幅 (nm)	178.59	179.53	181.18	182.08
CD 斜め (nm)	30.29	31.05	35.27	35.48
時間 (秒)	8	4	14	6
		022		

FDTD の収束 vs. dx

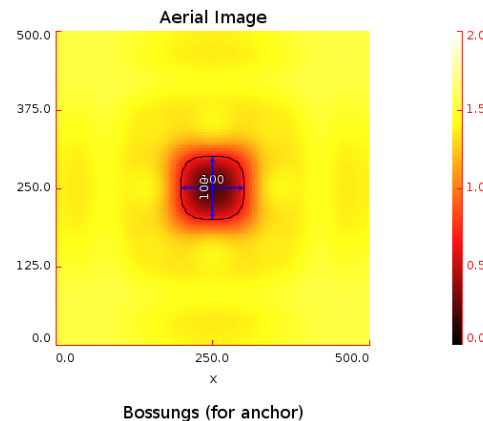
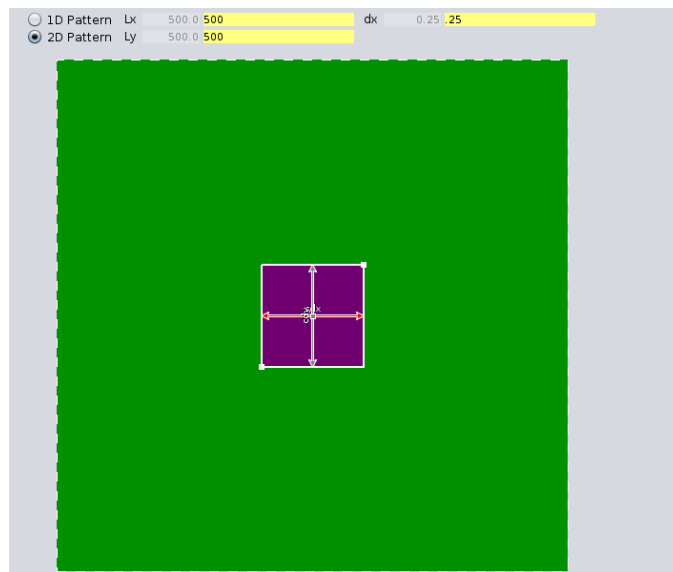


チェックボードパターンの結論：

- TRIGの方がFDTD より高速（高精度のケースでは10倍以上高速）
 - FDTD SG=1 (サブグリッド不使用)
- 「高精度」のケースでは、TRIG と FDTD はかなり一致
- TRIG と FDTD も45度のエッジにきちんと対応可能
- FDTDの場合 – ある程度のFDTDの精度を得るためには、 $dx=1\text{nm}$ （ウエハー）が必要（サブグリッドを使用すれば、幾何学の dx はこれより小さくてもよい）

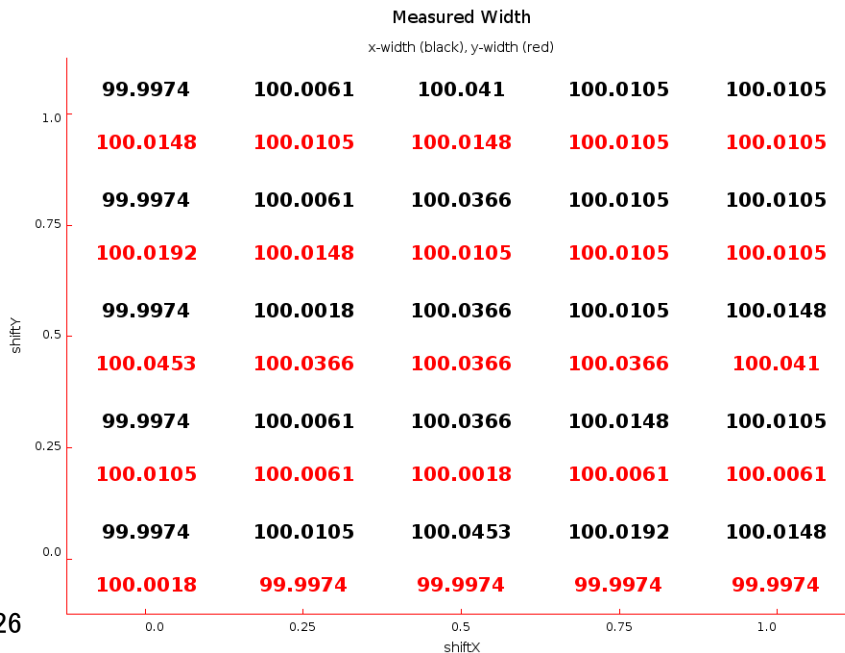
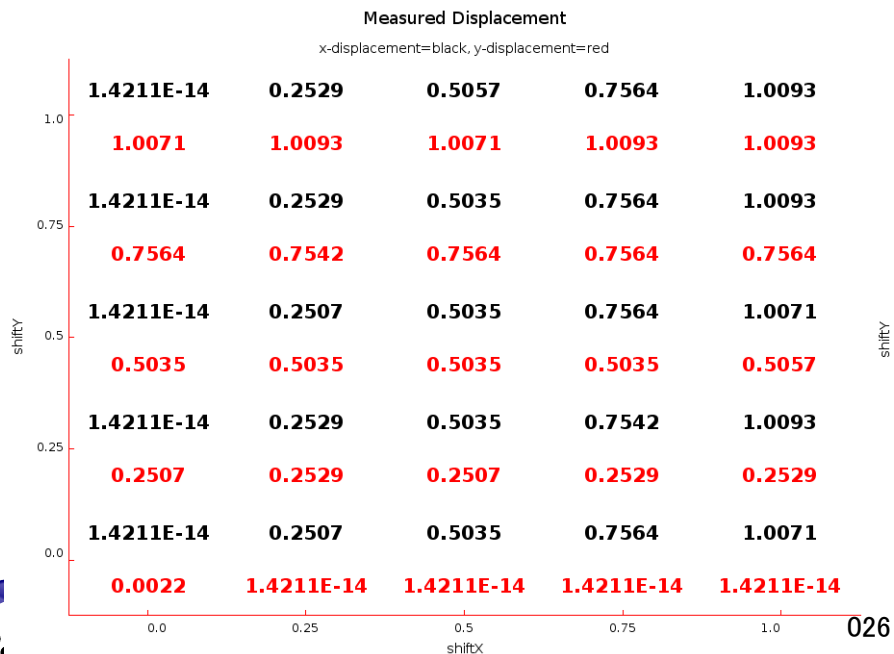
FDTD スーパーグリッドのテスト

- 四角い形状でシミュレーションを実施。四角をx方向、y方向に移動
- 細かい（形状）グリッドは $dx=0.25\text{nm}$ 、粗い（FDTD）グリッドは $dx=1.0\text{nm}$ （ウエハーの寸法）
- $dz=4\text{nm}$ （マスクの寸法）



結果....

- サブグリッドがよく機能している
- 位置に置いて優れた直線性
- CD はほぼ一定 (予想通り)



v7のDUV用新機能: TRIG、FDTD +スーパーグリッド ディスクッション...結論....

- 大規模な2Dパターンでは、TRIGはFDTD（スーパーグリッドなし）やRCWAより高速
- 同じような精度設定であれば、TRIGの方がFDTD（スーパーグリッドなし）より使用メモリが少ない。
- TRIGは、細かいエッジを配置した大きな領域をシミュレーションすることができる
- スーパーグリッドを使ったFDTDは、セルサイズの縮小に対応可能
- FDTDおよびTRIGは、45度軸に配置されたエッジのモデリングを正確に行うことができる
- EUV向けTRIGのプレゼンテーションもご参照ください[TRIG for EUV presentation](#)