



干渉波の影響を考慮した 無線システムのモデル化

- 規格に準拠した波形の生成
- 伝播環境とフェーズドアレイのモデル化
- 到来方向推定とビームフォーミングの実行
- パフォーマンスの評価と可視化

今日の無線通信システムは、幅広い信号源からの干渉にさらされる環境で運用しなければなりません。信号源が意図的であるかどうかに関わらず、システム設計には、干渉が存在する環境で問題なく運用できる信頼性の高さが重要です。

この例では、MATLAB®を使用して、エンドツーエンドの通信システムをモデル化する方法を解説します。干渉波が存在する環境で、通信リンクを保護するための有効なテクニックを取り入れます。この通信リンクは、モバイルユニットとLTE基地局間の通信に基づいています。

図1のシナリオでは、LTEダウンリンク信号が、2番目の干渉基地局から同一チャンネル干渉の影響を受けています。モバイルユニットの受信機には、送信された信号を適切に復旧し、実際の信号源との信頼性の高いリンクを維持するために、ビームフォーミングなどの干渉を軽減する技術が必要です。このような場合、システムモデルを使用することで、干渉を軽減する技術の有効性を評価し、システム全体にとって最適なパラメータを特定することができます。

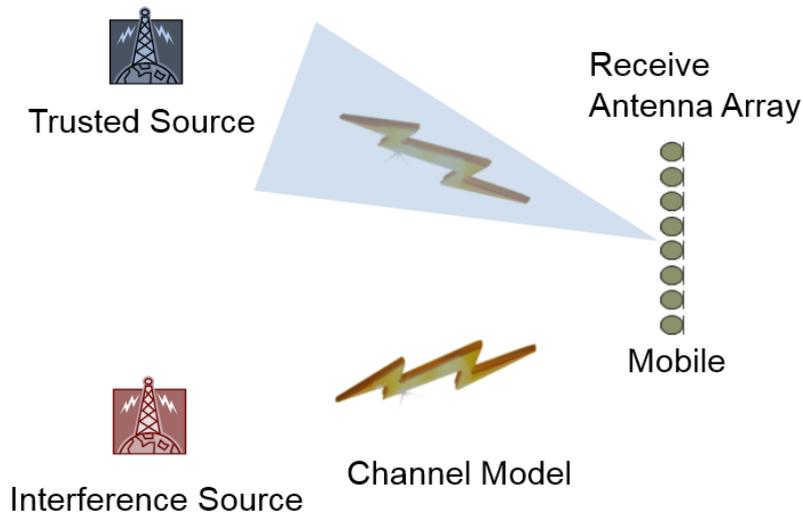


図1. 2番目の基地局からの同一チャンネル干渉が存在する環境において、モバイルユニットが実際の信号源からの送信信号を復旧するシナリオ

波形の生成

対象信号と干渉信号はLTEダウンリンク波形です。LTE System Toolbox™を使用して、規格に準拠したこれらのLTE波形を生成します。このツールボックスには、RMC (基準測定チャンネル)、FRC (固定参照チャンネル)、E-TM (E-UTRAテストモデル) などの基準信号を対話形式で生成するためのアプリが含まれています。以下の例では、アプリを使用して、R.7 RMC構成の波形を生成します。この構成には50個のダウンリンクリソースブロックがあり、64QAM変調を使用します(図2)。

図 2. RMC波形を生成するオプション

MATLABコードを使用して、規格に準拠した基準信号を生成することもできます。下のコードでは、2つの同一RMC構成を定義し、干渉信号が対象信号を模倣するというシナリオを作成しています。各信号に対して、1つのダウンリンクフレーム(サブフレームは10個)を含む波形を生成します。ランダムビットがPDSCH(物理ダウンリンク共有チャンネル)のリソースブロックにマッピングされません。

```
rmc1 = lteRMCDL('R.7'); % Use downlink reference measurement channel configuration
rmc1.TotSubframes = 10;
```

```

rmc1.NCellID = 65;
rmc2 = rmc1;

% Generate random bits to transmit
data1 = randi([0 1], sum(rmc1.PDSCH.TrBlkSizes), 1);
data2 = randi([0 1], sum(rmc2.PDSCH.TrBlkSizes), 1);

% Generate eNodeB settings structure and initial waveforms to
transmit
[txWaveform1, ~, enb1] = lteRMCDLTool(rmc1, data1);
[txWaveform2, ~, enb2] = lteRMCDLTool(rmc2, data2);

```

フェージングチャネルのモデル化

伝播環境の2つの要素であるフェージングチャネルとフリースペースパス損失をモデル化します。LTE System Toolboxは、LTE規格によって定義される3つの遅延プロファイル(EPA(拡張歩行者A)、EVA(拡張車両A)、ETU(拡張標準都市))に基づいて一連のチャネルモデルを提供します。システム設定用の構造体により、遅延プロファイル、サンプリングレート、ドップラー周波数、MIMO関連など、モデルによって使用されるパラメーターを指定できます。ここで選択したプロファイル(EVA)は、車両移動に典型的な中程度の遅延スプレッドを表しています。

Communications System Toolbox™ は、AWGN (加法性ホワイトガウスノイズ)、レイリー、ライスを含む追加のチャネルモデルを提供します。また、MATLABコードを使用すると、独自のチャネルモデルを実装することも可能です。

```

channel1 = struct; % Channel config structure
channel1.NRxAnts = 1; % Number of receive antennas
channel1.DopplerFreq = 5.0; % Doppler frequency in Hz
channel1.SamplingRate = ...
    enb1.SamplingRate; % Input signal sample rate
channel1.MIMOCorrelation = 'Low'; % Low MIMO correlation
channel1.Seed = 513; % Random number generator seed
channel1.InitTime = 1; % Fading process time offset
channel1.DelayProfile = 'EVA'; % Channel delay profile
channel2 = channel1;
channel2.Seed = 937;

rxWaveform1 = lteFadingChannel(channel1, txWaveform1);
rxWaveform2 = lteFadingChannel(channel2, txWaveform2);

```

伝播環境におけるフリースペースパス損失のモデル化

Phased Array System Toolbox™のフリースペースパス損失モデルを使用して、伝播環境モデルを補強します。このツールボックスには、波形伝播、クラッターと干渉、アレイとターゲットの動作、ターゲットの反射断面積のモデルが用意されています。これらのモデルは、レーダーシナリオおよび通信シナリオの両方に適用できます。

また、モバイルユニット、基地局、干渉の位置はXYZ座標で指定されます。これらの位置は、基地局がモバイルユニットに対して45度の角度、干渉は-45度の角度になるように定義されています。フリースペースパス損失は移動距離に基づいて計算されます。

```
fc = 2.14e9;
```

```
FSPL = phased.FreeSpace('OperatingFrequency', fc,'SampleRate', ...
    enbl.SamplingRate);
rxWaveform1 = step(FSPL, rxWaveform1, [1000;1000;0], [0;0;0], ...
    [0;0;0], [0;0;0]);
rxWaveform2 = step(FSPL, rxWaveform2, [1000;-1000;0], [0;0;0], ...
    [0;0;0], [0;0;0]);
```

フェーズドアレイと信号収集のモデル化

フェーズドアレイとは、規則的に配置されたアンテナ素子の集合です。アレイは、信号をターゲットに送信される放射エネルギーに変換、または他所から放射されたエネルギーを収集します。

フェーズドアレイは単一アンテナのシステムに対して、優れた空間分解能と干渉抑制を備えています。また、電子ビームステアリングの使用もフェーズドアレイにより可能になります。

Phased Array System Toolboxには、フェーズドアレイの設計、シミュレーション、解析のためのアルゴリズムとアプリが用意されています。例えば、Sensor Array Analyzerアプリを使用すると、形状や指向性といったフェーズドアレイの重要なプロパティを調べることができます (図 3)。これにより、アレイを対話形式で設計し、そのアレイをシミュレーションで使用するためのMATLABコードを自動生成することができます。

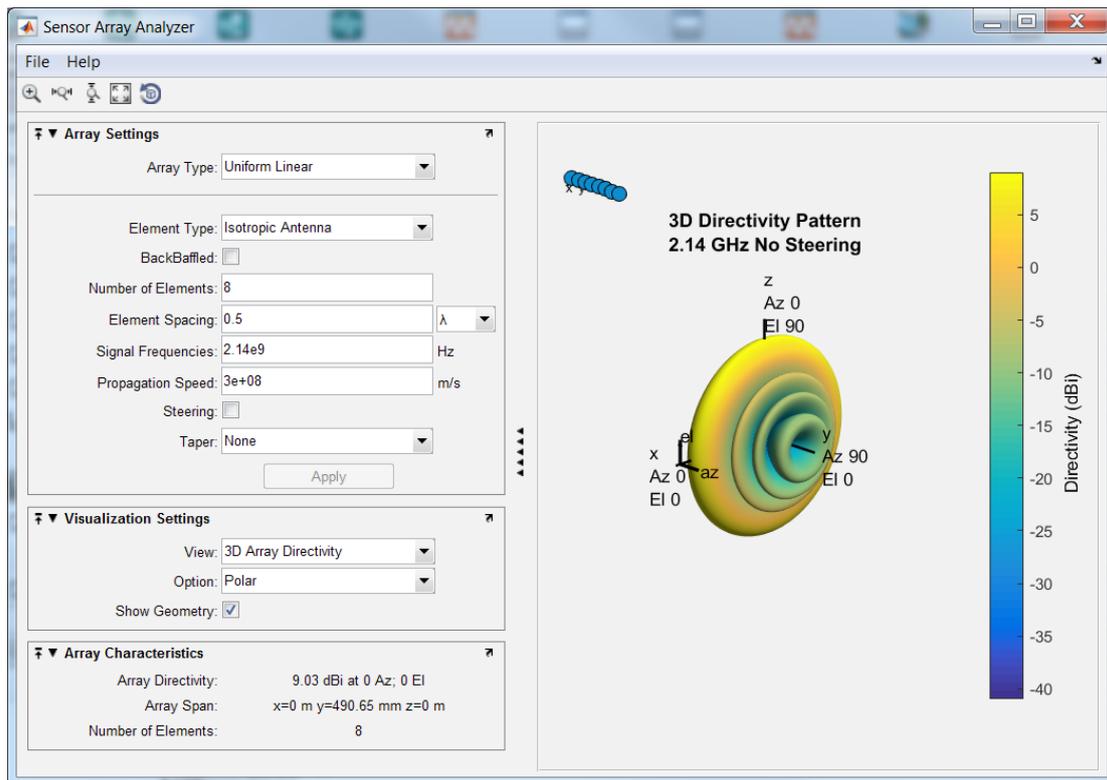


図3. Phased Array System ToolboxのSensor Array Analyzerアプリ

下記の例では、8個の素子と標準的な半波長間隔を使用してULA（等間隔直線アレイ）をモデル化します。単一のコレクターが対象信号と、異なる角度から配列に到達する干渉信号を収集します。

```
lambda = physconst('LightSpeed')/fc;
ULA = phased.ULA(8, 'ElementSpacing', lambda/2);
Collector = phased.Collector('Sensor', ULA,...
    'OperatingFrequency', fc);
receivedSignal = step(Collector, [rxWaveform1 rxWaveform2], ...
    [45 -45; 0 0]);
```

信号検知と到来方向推定

フェーズドアレイを使用すると、ビームフォーミングとDOA（到来方向）の推定が可能です。ビームフォーミングを使用した場合、アレイ素子に到達する信号が組み合わせられ、特定の角度からの信号に対しては強め合う干渉信号を与え、その他の信号は弱め合う干渉信号を与えます。また、この機能をアルゴリズムや統計手法と組み合わせることで、DOA推定を実行することができます。Phased Array System Toolboxは、多数のビームフォーミングアルゴリズムとDOA推定アルゴリズム（MUSICとESPRITを含む）を提供しています。

この例では、統計テスト（赤池情報量基準）を使用して、アレイ素子に影響を与える信号の数を推定します。その後、ビームスキャン推定アルゴリズムにより、さまざまな角度でアレイビームをスキャンし、受信信号の電力が最も大きくなる角度を特定します。これらの角度は、対象信号および干渉信号が到来すると推定される方向を表しています。

```
numSignals = aictest(awgn(receivedSignal*10000, 40));
Beamscan = phased.BeamscanEstimator('SensorArray', ULA, ...
    'OperatingFrequency', fc, 'DOAOutputPort', true, ...
    'NumSignals', numSignals, 'ScanAngles', -90:90);
[~, doas] = step(Beamscan, receivedSignal)
doas =
    45    -45
```

ビームフォーミングとセル検索の実行

到来角度が特定されたら、ビームフォーミングを使用して各信号を空間的に分離します。特定された各角度にアレイビームを誘導し、信号を受信し、受信機での信号処理を実行します。

ビームフォーミングは、各アレイ素子で受信された信号に掛け合わせる重みを計算することで実行されます。受信された信号と重みを掛け合わせることで、望ましい角度から受信された信号を強調し、望ましくない角度から受信された信号を最小化します (nullステアリング)。

LTE System Toolboxの関数`lteCellSearch`は、受信された信号に対してセル検索を実行し、プライマリ同期信号およびセカンダリ同期信号との相関のピーク振幅に基づいて、存在するセルを特定します。この場合、セルのRSRP（基準信号受信電力）に基づいて、各角度から受信された最も強力なセルを特定します。RSRPは、通信対象のセルを選択するためにモバイルユニットで使用される指標の1つです。関数`demodulateLTESignal`はLTEフレームの始まりを特定し、RSRPを計算し、チャンネル応答を推定します。また、等化を実行し、受信データに対してデブリコーディング、レイヤーデマッピング、復調、スクランブリング解除を実行します。これらの最終結果として、受信ビットの推定が生成されます。

```
for ii = 1:length(doas)
    % Use beamforming to steer beam to angle of arriving signal
    w(:,ii) = sidelobeCanceller(ULA.getElementPosition/lambda, ...
        doas(ii), setdiff(doas, doas(ii)));
    ybf(:,ii) = receivedSignal * conj(w(:,ii));
    % Identify the most powerful cell received from this angle
    searchalg.MaxCellCount = 1;
    searchalg.SSSDetection = 'PostFFT';
    [cellIDs(ii), ~] = lteCellSearch(enb1, ybf(:,ii), searchalg);
```

```

    % Perform LTE demodulation and RSRP measurement on beam-
    formed signal
    try [rxData(:,ii) rxRSRPs(ii), rxSymbols(:,ii)] = ...
        demodulateLTESignal(enb1, ybf(:,ii));
catch, end
end

% Compute result without beamforming (for comparison later)
y = sum(receivedSignal,2);
rxNoBF = demodulateLTESignal(enb1, y);

```

対象信号の選択

存在する信号とその到来角度を特定したら、その角度に基づいて対象信号を選択できます。ここでは、単純に目的の角度である45度に最も近い角度を選択します。極座標プロットを使用して、アレイパターンと特定した信号の到来角度を表示します(図4)。

```

[~, signalSelect] = min(abs(doas - 45));
rxData = rxData(:,signalSelect);

```

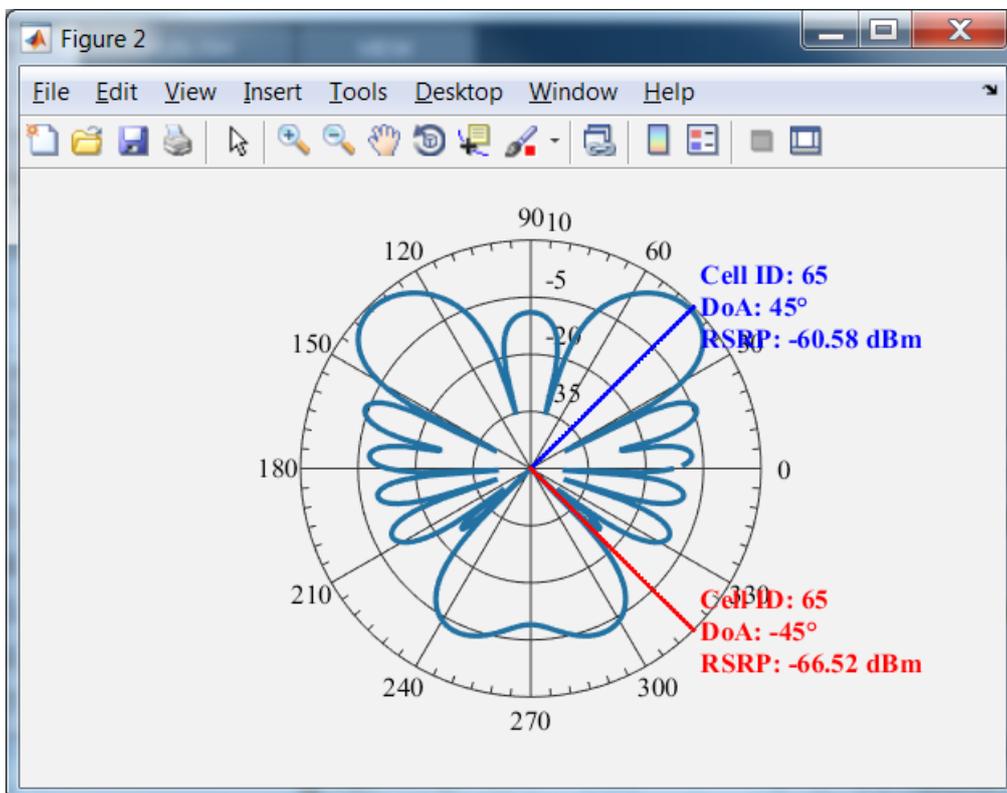


図4. アレイパターンと、到来角度を備えた特定済みの信号を示す極座標プロット

信号の可視化

Communications System ToolboxとDSP System Toolbox™によって提供されたスコープを使用して、ビームフォーミング後の対象信号の受信シンボルコンスタレーション (図 5) と、各信号の受信スペクトル (図 6) をプロットします。

```
Constellation = ...
    comm.ConstellationDiagram('ShowReferenceConstellation', ...
        false, 'ShowGrid', true);

step(Constellation, rxSymbols(:,signalSelect));

SpectrumAnalyzer = dsp.SpectrumAnalyzer('SampleRate', 15360000,
    'SpectralAverages', 1, ...
    'PowerUnits', 'dBW', 'PlotAsTwoSidedSpectrum', true, 'Name', ...
    'Received Signals at UE', 'ChannelNames', {'SOI', ...
    'Interferer'}, 'ShowLegend', true);

step(SpectrumAnalyzer, [rxWaveform1 rxWaveform2]);
```

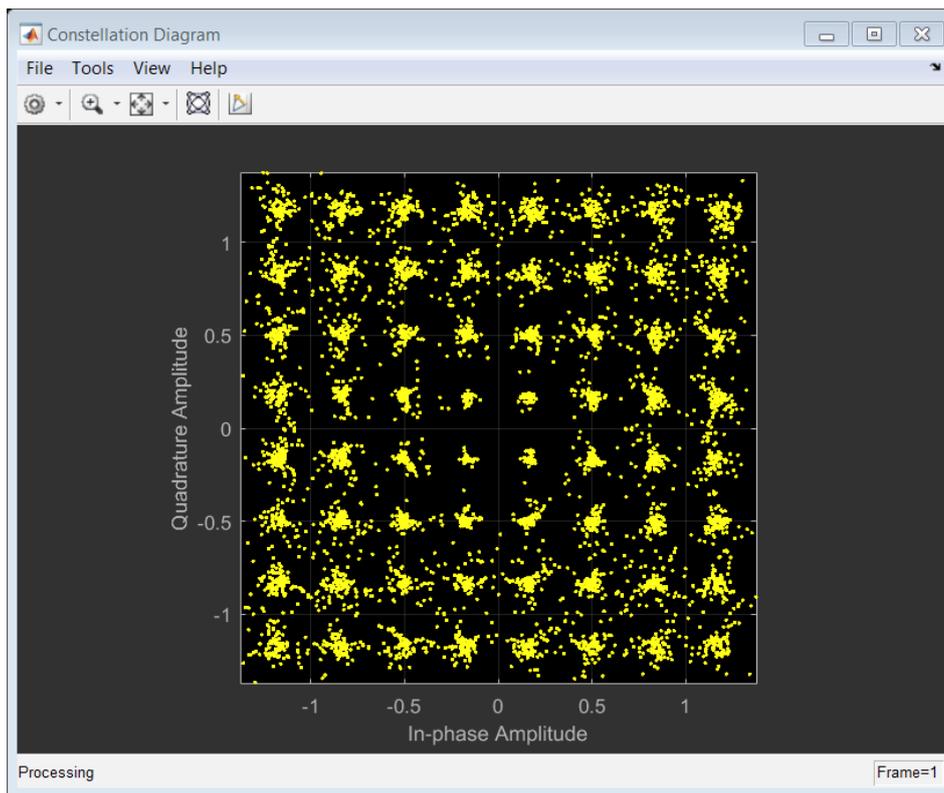


図5. 対象信号のコンスタレーションダイアグラム

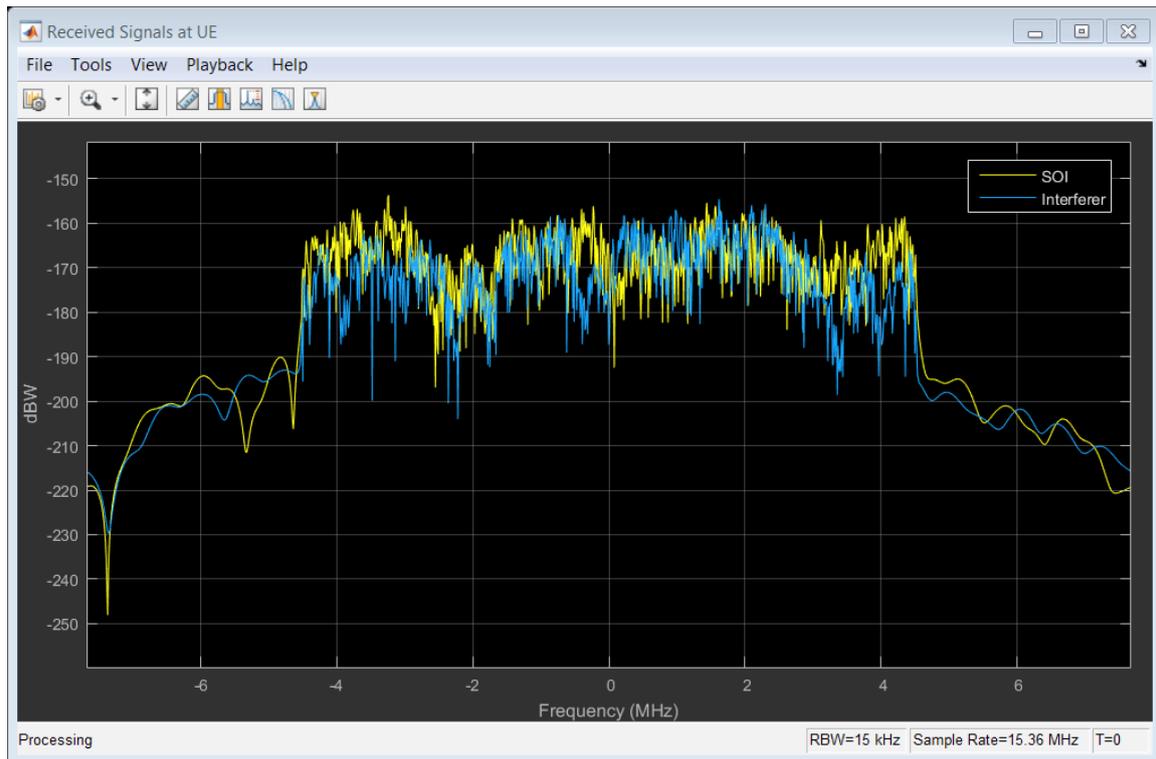


図 6. 対象信号および干渉の受信スペクトル。

システムパフォーマンスの評価

デジタル通信システムの主要なパフォーマンス指標はBER（ビット誤り率）、つまりエラーの状態を受信されたビットの割合です。Communications System ToolboxのSystem objectを使用してBERを計算します。そのBERメトリックを使用して、システムパフォーマンスの改善におけるビームフォーミングの効果を定量化します。

```
fprintf('\nWithout beamforming:\n-----\n');
BER1=comm.ErrorRate;
errors1 = step(BER1, double(rxNoBF), data1(1:length(rxNoBF)));
fprintf(['Number of bits transmitted = %d' ...
        '\nNumber of bit errors = %d' ...
        '\nBit error rate = %f\n'], errors1(3), errors1(2), errors1(1));

fprintf('\nWith beamforming:\n-----\n');
BER2=comm.ErrorRate;
errors2 = step(BER2, double(rxData), data1(1:length(rxData)));
```

```
fprintf(['Number of bits transmitted = %d' ...  
        '\nNumber of bit errors = %d' ...  
        '\nBit error rate = %f\n'], errors2(3), errors2(2), errors2(1));
```

Without beamforming:

Number of bits transmitted = 242368

Number of bit errors = 81091

Bit error rate = 0.334578

With beamforming:

Number of bits transmitted = 242368

Number of bit errors = 0

Bit error rate = 0.000000

まとめ

MATLABを使用して、干渉波の影響を考慮した無線システムをモデル化する方法をご紹介します。LTE System Toolboxの波形生成および信号処理、Phased Array System Toolboxのフェーズドアレイおよび伝播モデリング、Communications System Toolboxのパフォーマンス評価と可視化の機能を組み合わせて使用することで、現実的なシナリオのエンドツーエンド通信システムをモデル化することができます。さらに、このモデルを使用することにより、システムに最適なパラメーターと仕様を導き出すことができます。

使用製品

MATLAB

Communications System Toolbox

LTE System Toolbox

Phased Array System Toolbox