Simulating Rayleigh Fading MIMO Channel Capacity*
Project #5

Intelligent Communication Systems (ICS) Lab.
노용재

Winter Intern Seminar (2023-1)

Contents

---

1 Theory

1.1 Channel Unknown to the Transmitter

특이값 분해(Singular Value Decomposition;SVD)에 따르면 모든 행렬은 $U\Sigma V^H$로 분해할 수 있다. 이때, Σ는 대각 성분이 음수가 아닌 실수로만 이루어진 대각행렬이며 U와 V는 직교행렬이다.
임의의 행렬 H에 대해 다음이 성립한다.

$$H = U\Sigma V^H \tag{1}$$

$$H^H = V\Sigma^H U^H \tag{2}$$

$$\begin{aligned} HH^H &= (U\Sigma V^H)(V\Sigma^H U^H) \\ &= U\Sigma\Sigma^H U^H \\ &= U\Sigma^2 U^H (\because \Sigma = \Sigma^H) \end{aligned} \tag{3}$$

$$\therefore HH^H = U\Sigma U^H$$

*Simulation code uploaded on https://github.com/lightwick/ICS_project/tree/main/Spatial_Diversity

직교행렬 Q에 의해 $HH^H = Q\Sigma Q^H$와 같이 나타낼 수 있다. 이 특성을 이용하여 $Capacity$, C를 다음과 같이 나타낼 수 있다.

$$\begin{aligned} C &= \log_2 \det(I_{N_R} + \frac{E_S}{N_T N_0} HH^H) \\ &= \log_2 \det(I_{N_R} + \frac{E_S}{N_T N_0} Q\Lambda Q^H) \\ &= \log_2 \det(I_{N_R} + \frac{E_S}{N_T N_0} Q^H Q\Lambda) \ (\because \det(I+AB) = \det(I+BA)) \\ &= \log_2 \det(I_{N_R} + \frac{E_S}{N_T N_0} \Lambda) \ (\because Q^H Q = I) \\ &= \sum_{i=1}^{r} \log_2(1 + \frac{E_S}{N_T N_0}\lambda_i) \end{aligned} \tag{4}$$

1.2 Channel Known to the Transmitter; Waterpouring Algorithm

r개의 sub-channel이 있을 때, $\sum_{i=1}^{r} \gamma_i = N_T$라 하자.

$$C = \sum_{i=1}^{r} \log_2(1 + \frac{E_S \gamma_i}{N_T N_0}\lambda_i) \tag{5}$$

r개의 sub-channel에 따라 전력을 적절히 분배하여 Capacity를 최대화하려 한다.

Lagrangian method를 사용하여 $\sum_{i=1}^{r} \gamma_i = N_T$일 때, $C = \sum_{i=1}^{r} \log_2(1 + \frac{E_S \gamma_i}{N_T N_0}\lambda_i)$의 값을 최대화하자.

Lagrangian Method

$$\mathcal{L}(\gamma_1, ..., \gamma_r, \theta) = \sum_{i=1}^{r} \log_2(1 + \frac{E_S \gamma_i}{N_T N_0}\lambda_i) - \theta(\sum_{i=1}^{r} \gamma_i - N_T) \tag{6}$$

Lagrangian이 식(6)과 같을 때 $\nabla\mathcal{L} = \vec{0}$을 만족하여야 한다. ($\rho = \frac{E_S}{N_0}$)

$$\nabla\mathcal{L} = \begin{bmatrix} \frac{\partial\mathcal{L}}{\partial\gamma_1} \\ \vdots \\ \frac{\partial\mathcal{L}}{\partial\gamma_r} \\ \frac{\partial\mathcal{L}}{\partial\theta} \end{bmatrix} = \begin{bmatrix} \frac{1}{(1+\frac{\gamma_1}{N_T}\rho\lambda_1)\ln 2}\frac{\rho\lambda_1}{N_T} - \theta \\ \vdots \\ \frac{1}{(1+\frac{\gamma_r}{N_T}\rho\lambda_r)\ln 2}\frac{\rho\lambda_r}{N_T} - \theta \\ \sum_{i=1}^{r}\gamma_i - N_T \end{bmatrix} = \vec{0} \tag{7}$$

즉, 모든 i $(i = 1, 2.., r)$에 대해 다음을 만족한다.

$$\frac{1}{(1 + \frac{\gamma_i}{N_T}\rho\lambda_i)\ln 2}\frac{\rho\lambda_i}{N_T} - \theta = 0 \tag{8}$$

이를 정리하면,

$$\gamma_i = \frac{1}{\theta \ln 2} - \frac{N_T}{\rho\lambda_i} \tag{9}$$

$$
\begin{aligned}
\sum_{i=1}^{r}\gamma_i - N_T &= \sum_{i=1}^{r}(\frac{1}{\theta\ln 2} - \frac{N_T}{\rho\lambda_i}) - N_T \\
&= \frac{r}{\theta\ln 2} - \frac{N_T}{\rho}\sum_{i=1}^{r}\frac{1}{\lambda_i} - N_T
\end{aligned} \tag{10}
$$

$$
\frac{r}{\theta\ln 2} - \frac{N_T}{\rho}\sum_{i=1}^{r}\frac{1}{\lambda_i} - N_T = 0 \tag{11}
$$

$$
\tag{12}
$$

$$
\frac{1}{\theta\ln 2} = \frac{N_T}{r}(1 + \frac{1}{\rho}\sum_{i=1}^{r}\frac{1}{\lambda_i}) \tag{13}
$$

이를 식(9)에 대입하자.

$$
\gamma_i = \frac{N_T}{r}(1 + \frac{1}{\rho}\sum_{j=1}^{r}\frac{1}{\lambda_j}) - \frac{N_T}{\rho\lambda_i} \tag{14}
$$

2 Implementation

2.1 Channel Unknown to the Transmitter

Capacity를 SNR에 따른 column vector로 정리하여 나타낸다.

```
function [EntireCapacity, ErgodicCapacity, OutageCapacity] = getCapcity_CU(Nr, Nt,
  EsN0_dB, iTotal)
    % EntireCapacity is a 'Column Vector'
    EntireCapacity = zeros(iTotal, length(EsN0_dB));
    ErgodicCapacity = zeros(length(EsN0_dB), 1);
    OutageCapacity = zeros(length(EsN0_dB), 1);

    EsN0 = db2pow(EsN0_dB);

    for EsN0_idx=1:length(EsN0_dB)
        %% Get Capacity array
        for iteration=1:iTotal
            H = (randn(Nr, Nt) + 1j * randn(Nr, Nt)) / sqrt(2); % Receiver x Transmitter
            Capacity = log2(det(eye(Nr)+EsN0(EsN0_idx)/Nt*H*H'));
            EntireCapacity(iteration, EsN0_idx) = Capacity;
        end

        %% Outage Capcity
        [cdf,x] = ecdf(EntireCapacity(:, EsN0_idx));
        [~, Outage_idx] = min(abs(cdf-0.1));
        OutageCapacity(EsN0_idx) = x(Outage_idx);

    end
    %% Ergodic Capcity
    ErgodicCapacity = real(mean(EntireCapacity))';
end
```

2.2 Channel Known to the Transmitter

Capacity를 SNR에 따른 column vector로 정리하여 나타낸다.

```
function [EntireCapacity, ErgodicCapacity, OutageCapacity] = getCapcity_CK(Nr, Nt, EsN0_dB, iTotal)
    % EntireCapacity is a 'Column Vector'
    EntireCapacity = zeros(iTotal, length(EsN0_dB));
    ErgodicCapacity = zeros(length(EsN0_dB), 1);
    OutageCapacity = zeros(length(EsN0_dB), 1);

    EsN0 = db2pow(EsN0_dB);

    for EsN0_idx=1:length(EsN0_dB)
        %% Get Capacity array
        for iteration=1:iTotal
            H = (randn(Nr, Nt) + 1j * randn(Nr, Nt)) / sqrt(2); % Receiver x Transmitter

            p = 1;
            [~,S,~] = svd(H);

            if min(size(S))==1
                EigenValues = S(1,1)^2;
            else
                EigenValues = diag(S).^2;
            end

            while true
                r = rank(H);
                PowerDistribution = Nt/(r-p+1)*(1+1/EsN0(EsN0_idx)*sum(1./EigenValues, 'all')) - Nt/EsN0(EsN0_idx)*(1./EigenValues);
                if PowerDistribution(r-p+1)>=0
                    break;
                else
                    p = p+1;
                    EigenValues(length(EigenValues))=[];
                end
            end
            Capacity = sum(log2(1+EsN0(EsN0_idx)/Nt * PowerDistribution.*EigenValues), 'all');
            EntireCapacity(iteration, EsN0_idx) = Capacity;
        end

        %% Outage Capcity
        [cdf,x] = ecdf(EntireCapacity(:, EsN0_idx));
        [~, Outage_idx] = min(abs(cdf-0.1));
        OutageCapacity(EsN0_idx) = x(Outage_idx);
    end
    %% Ergodic Capcity
    ErgodicCapacity = real(mean(EntireCapacity))';
end
```

3 결과 및 분석

3.1 Simulation Result

3.1.1 CDF of Capacity

2x2 MIMO: E_S/N_0=10dB

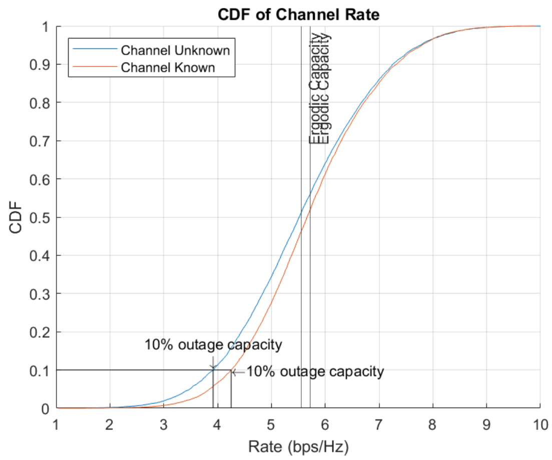
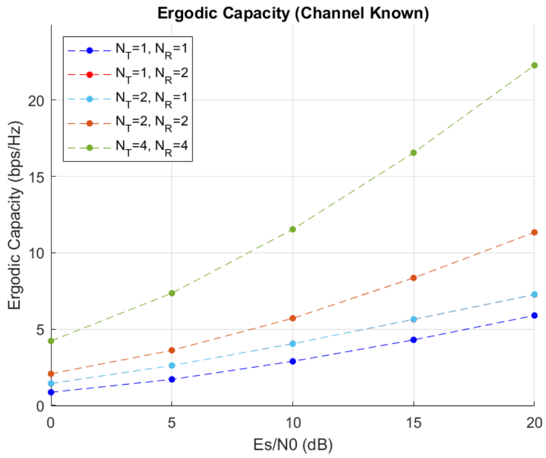
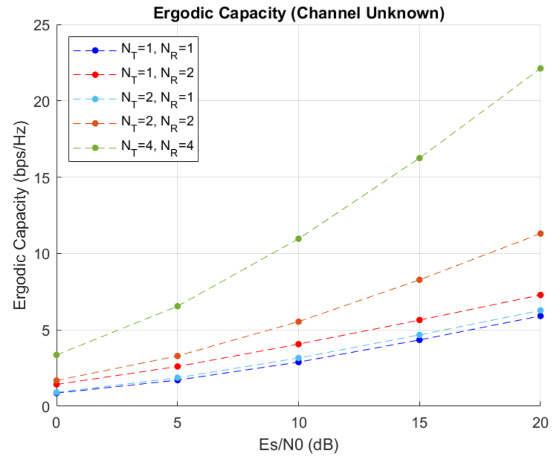


Figure 1

3.1.2 Capacity by Antenna Number

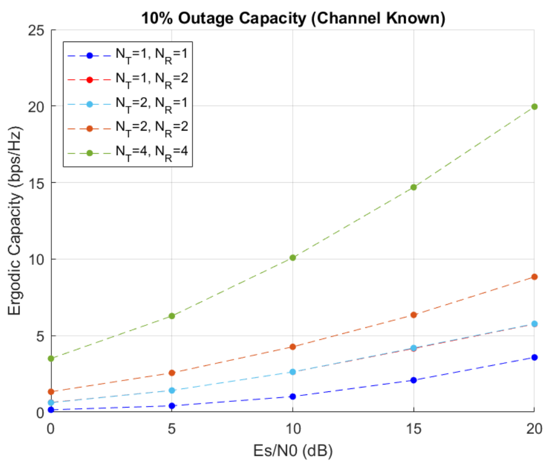


(a) Channel Known to Transmitter

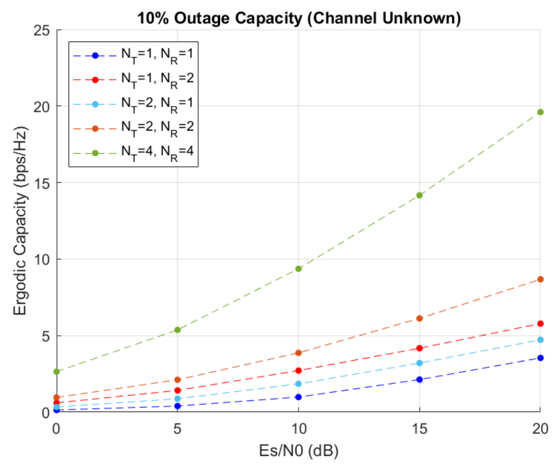


(b) Channel Unknown to Transmitter

Figure 2: Ergodic Capacity



(a) Channel Known to Transmitter

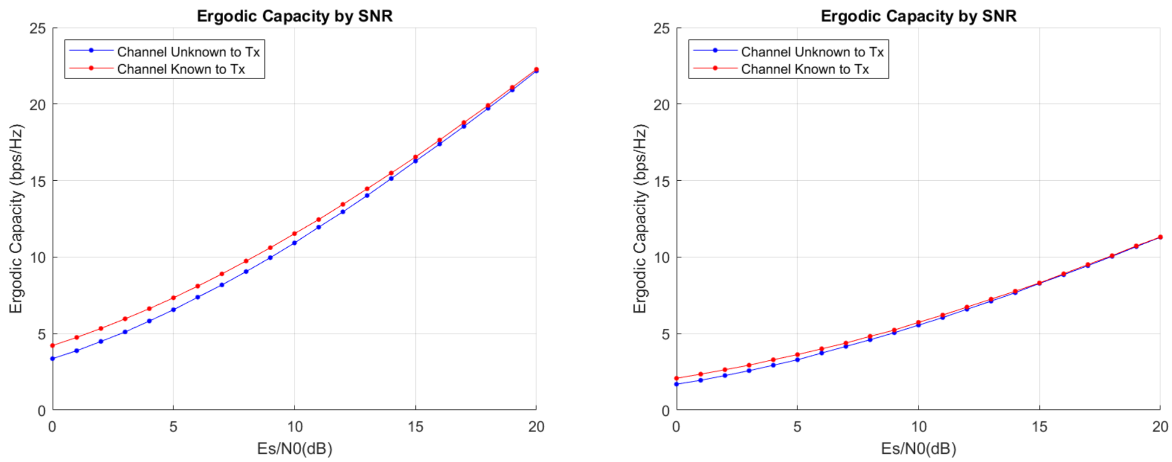


(b) Channel Unknown to Transmitter

Figure 3: 10% Outage Capacity

Channel Known의 경우, $N_T > N_R (N_T = \alpha, N_R = \beta)$일 때의 *capacity*는 $N_T = \beta, N_R = \alpha$일 때의 *capacity*와 거의 일치하는 모습을 보였다.

3.1.3 Ergodic Capacity by Channel Knowledge



(a) 4x4 MIMO (b) 2x2 MIMO

Figure 4

3.2 CK, CU간 Capacity 차이 분석

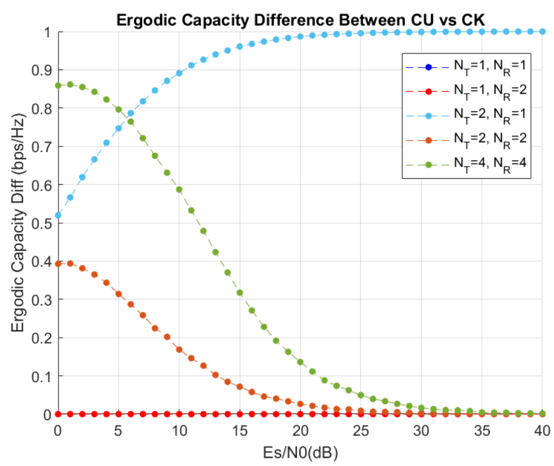


Figure 5

Figure 5는 transmitter가 channel에 대한 정보를 아는 경우와 모르는 경우 간의 capacity 차이를 나타낸 것이다. $N_T = 1, N_R = 1$와 $N_T = 1, N_R = 2$일 때, 그 차이는 0이다. 그 외의 경우에는 모두 특정 값으로 수렴한다. 이것에 대해 분석해 보겠다.
이후 Channel에 대한 정보가 transmitter에게 있는 경우의 capacity는 C_{CK}, 없는 경우의 capacity는 C_{CU}이라 하겠다.

3.2.1 SIMO에서의 Capacity; $C_{CK} - C_{CU} = 0$

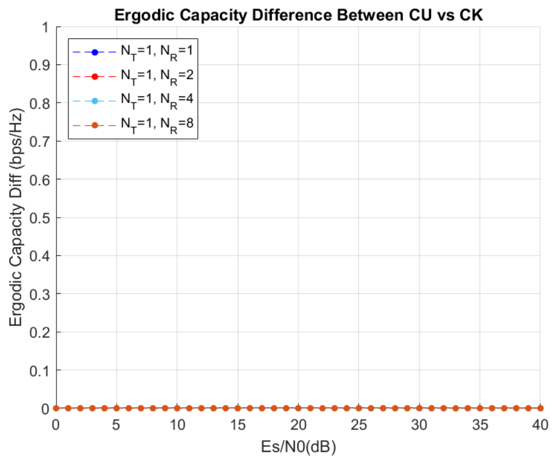


Figure 6

Figure 6를 보면 SISO와 SIMO의 경우에 $C_{CK} - C_{CU} = 0$인 것을 확인할 수 있다. 즉, channel knowledge가 capacity에 영향을 미치지 않는다는 것을 알 수 있다.
*Channel Known case*인 Waterpouring algorithm의 식 (5)를 살펴보자. SIMO에서 $r = rank(H) = 1$이다.이며 $\gamma_1 = 1$이다.

$$\begin{aligned} C_{CK} &= \sum_{i=1}^{r} \log_2(1 + \frac{E_S \gamma_i}{N_T N_0} \lambda_i) \\ &= \sum_{i=1}^{r} \log_2(1 + \frac{E_S}{N_T N_0} \lambda_i) \\ &= C_{CU} \end{aligned} \tag{15}$$

즉, $C_{CK} = C_{CU}$이다.

3.2.2 MISO에서 Capacity 차이 수렴; $C_{CK} - C_{CU} = \log_2 N_T$

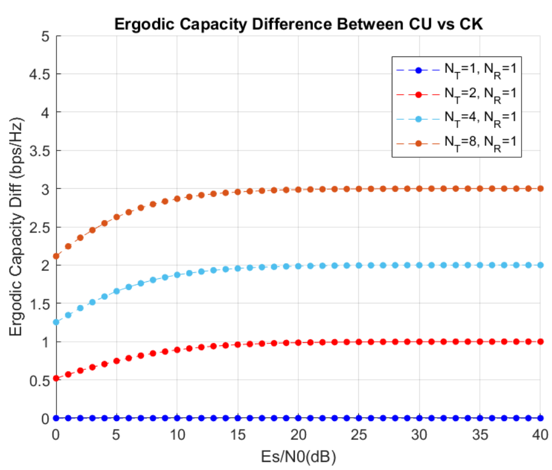


Figure 7

Figure 7를 보면, $C_{CK} - C_{CU} = \log_2 N_T$로 수렴하는 것을 볼 수 있다. MISO에서 $r = rank(H) = 1$이다.

$$\begin{aligned}
\lim_{\rho\to\infty}(C_{CK} - C_{CU}) &= \lim_{\rho\to\infty}(\log_2(1 + \frac{\gamma}{N_T}\rho\lambda_1) - \log_2(1 + \frac{\rho}{N_T}\lambda_1)) \\
&= \lim_{\rho\to\infty}\log_2\frac{1 + \frac{\gamma_1}{N_T}\rho\lambda_1}{1 + \frac{1}{N_T}\rho\lambda_1} \\
&= \log_2 \gamma_1 \\
&= \log_2 N_T
\end{aligned} \tag{16}$$