Roja Atashkar
QCSS: Prof. Salehi Quantum Communication of Signals and Systems 2
June 27, 2024

Final Exam (Take Home)

Question 1.

Question 2. Plot the error probability of 4-PPM modulation with equiprobable symbols using the SRM method (figure 8.23). Consider Ns = 0.5 0.5 8 and N = 0, 0.1, 0.2. Choose suitable matrix dimensions. Plot the proposed bound in problem 1 and compare them with the SRM method.

Recall of SRM Decision with Mixed States With Gram matrix Approach

(1)From the state matrix $\Gamma_{n\times H} = [\beta_0, \beta_1, \ldots, \beta_{K-1}]$ we evaluate the Gram matrix:

$$\underset{H\times H}{G} = \underset{H\times n}{\Gamma^*_n}\,\underset{n\times H}{\Gamma} = [\beta_i^*\beta_j], \quad H = h_0 + h_1 + \cdots + h_{K-1}$$

where $\beta_i = [\beta_{i0}, \beta_{i1}, \ldots, \beta_{i,h_i}]$.

The dimension H × H of the Gram matrix is determined by the "virtual" rank of the density operators. Without restriction, we may suppose that the ranks are equal, i = 0, and hence the size of the matrix G, consisting of K × K blocks of dimensions 0 × 0, results in H = K 0.

(2) Evaluate the reduced EID $G = V_r\Sigma_r^2 V_r^*$.

(3) Find the inverse square root $G^{-1/2} = V_r\Sigma_r^{-1}V_r^*$.

(4) Subdivide the matrices $G^{\pm\frac{1}{2}}$ into blocks $\left(G^{\pm\frac{1}{2}}\right)_{ij}$ of dimensions $h_i \times h_j$.

(5) Evaluate the measurement matrix as $M = \Gamma G^{-1/2}$.

(6) Considering that $B = G^{-1/2}$, evaluate the transition probabilities from $p_c(j \mid i) = \mathrm{Tr}\left[b_{ji}^* b_{ji}\right], P_c = \frac{1}{K}\sum_{i=0}^{K-1}\mathrm{Tr}\left[b_{ii}^* b_{ii}\right]$, with b_{ij} given by $\left(G^{-1/2}\right)_{ij}$.

The computational complexity of the whole procedure is concentrated in the EID of the $H \times H$ Gram matrix.

Gram Matrix Approach with GUS

$$\rho_i = S^i\rho_0 S^{-i}, \quad i = 0, 1, \ldots, K-1$$

$$\beta_i = S^i\beta_0, \quad \mu_i = S^i\mu_0, \quad i = 0, 1, \ldots, K-1$$

Where S is the symmetry operator.

(1) Evaluate the matrices of dimension $h_0 \times h_0$

$$D_k = \sum_{i=0}^{K-1}\beta_0^*\beta_i W_K^{-ki}, \quad \beta_i = S^i\beta_0$$

(2) Evaluate by EID the square roots $D_k^{1/2}$. (3) Evaluate the (i, j) block of $G^{1/2}$ as

$$\left(G^{1/2}\right)_{ij} = \frac{1}{\sqrt{K}}\sum_{k=0}^{K-1} W_K^{k(j-i)} D_k^{1/2}$$

(4) Evaluate the transition probabilities as $p_c(j \mid i) = \mathrm{Tr}\left[\left(G^{1/2}\right)_{ji}^* \left(G^{1/2}\right)_{ji}\right]$. (5) The correct decision probability is given by

$$P_c = \mathrm{Tr}\left(\frac{1}{K}\sum_{k=0}^{K-1} D_k^{1/2}\right)^2$$

- The computational complexity is confined to the evaluation of the square roots $D_k^{1/2}$. - Remark With mixed states, also in the presence of GUS, the decision obtained with the SRM in general is not optimal.

$$W_{[K]} = \frac{1}{\sqrt{K}} \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W_K^{-1} & W_K^{-2} & \dots & W_K^{-(K-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & W_K^{-(K-1)} & W_K^{-2(K-1)} & \dots & W_K^{-(K-1)(K-1)} \end{bmatrix}, \quad W_K = e^{i\frac{2\pi}{K}}$$

Signal in the Presence of Thermal Noise

$$N_\gamma := |\gamma|^2 = \text{ number of signal photons}$$

while $\mathcal{N}$ preserves the meaning of the number of thermal photons. - As in the case of pure noise, the operator $\rho(\gamma)$ can be expressed in terms of the eigenstates $|n\rangle$ of the number operator.

$$\rho(\gamma) = \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} R_{mn}(\gamma) |m\rangle\langle n|$$

where the coefficients have the expression: - for $m \leq n$

$$R_{mn}(\gamma) = \frac{\mathcal{N}^n}{(\mathcal{N}+1)^{n+1}} \sqrt{\frac{m!}{n!}} \left(\frac{\gamma^*}{\mathcal{N}}\right)^{n-m} \exp\left(-\frac{|\gamma|^2}{\mathcal{N}+1}\right) L_m^{(n-m)}\left(-\frac{|\gamma|^2}{\mathcal{N}(\mathcal{N}+1)}\right)$$

while for $m > n$ one uses the Hermitian symmetry $R_{nm}(\gamma) = R_{mn}^*(\gamma)$. - $L_m^{(n-m)}(x)$ is the generalized Laguerre polynomial of degree m and of parameter $n-m$, which is given by

$$L_m^{(n-m)}(x) = \sum_{k=0}^{m} (-1)^k \binom{n}{m-k} \frac{x^k}{k!} = \sum_{k=0}^{m} (-1)^k \frac{n!}{(n-m+k)!(m-k)!k!} x^k$$

Discretization of Density Operators - In a quantum transmission with coherent states in the presence of thermal noise the constellation

$$\rho_i = \rho(\gamma_i), i = 0, 1, \dots, K-1$$

is formed by Glauber density operators. - To proceed, one must approximate this expression, which gives each ρ_i as a linear combination of a continuum of coherent states, with a finite expression, given by the matrix representation but limited to a finite number n of terms, that is,

$$\rho(\gamma) \simeq \sum_{h=0}^{n-1} \sum_{k=0}^{n-1} R_{hk}(\gamma) |h\rangle\langle k| := R(\gamma)$$

- In such a way, the infinite-dimensional density operator is approximated by a square matrix $R(\gamma) = [R_{hk}(\gamma)]$ of finite dimension $n \times n$

Spectral Decomposition (EID) and Factorization - The finite representation can be elaborated through the EID. - Considering that ρ is Hermitian and positive semidefinite and that these properties also hold for its approximation, the EID of R can be expressed in the forms

$$R = Z\Lambda_\rho Z^* = \sum_{i=1}^{h} d_i^2 |z_i\rangle \langle z_i| = Z_h D_h Z_h^* = \underbrace{Z_h\sqrt{D_h}}_{\beta} \underbrace{\sqrt{D_h} Z_h^*}_{\beta^*} = \beta\beta^*$$

Where h is the virtual rank of R.

Approximation Criteria

In the EID and in the previous factorizations we have two approximations:

1. the approximation R of the density operator ρ by a finite number n of terms,
2. the approximation of the true rank n of the matrix R by the virtual rank h, which allows us to find the approximate factor $\tilde{\beta}$ of ρ of dimensions $n \times h$.

For both approximations, we need to establish a criterion that guarantees a given accuracy.

- For the first approximation the trace criterion seems to be more appropriate. We recall that a density operator has always unitary trace and that $\mathrm{Tr}[\rho] = 1$ represents also the normalization condition of the Laguerre distribution $p_n(k \mid \gamma)$, which gives the meaning of the diagonal elements.

Hence, only if $\mathrm{Tr}[R]$ gives a value close to one, the approximation is satisfactory, and, having fixed an accuracy ε, the number of terms $n = n_\varepsilon$ must be calculated according to the condition

$$\mathrm{Tr}[R] = \sum_{k=0}^{n_\varepsilon - 1} R_{kk} = \sum_{k=0}^{n_\varepsilon - 1} p_n(k \mid \gamma) \geq 1 - \varepsilon$$

- For the second approximation one can use the reconstruction criterion, based on the square error between the matrix R and its "reconstruction" $\beta\beta^*$, where both R and $\beta\beta^*$ are $n \times n$ matrices, with n previously evaluated.
Hence, having fixed the accuracy v, we evaluate the virtual rank $n_v = h$ such that

$$\mathrm{mse}\left(R - \beta\beta^*\right) \leq v$$

4-PPM without Thermal noise

$$|X|^2 = \mathrm{e}^{-N_s}$$

The error probability is computed from $P_e = 1 - \frac{1}{K}\sum_{i=0}^{K-1} \left|\left(G^{\frac{1}{2}}\right)_{ii}\right|^2$ and becomes

$$P_e = 1 - \frac{1}{K^2}\left(\sqrt{1 + (K-1)|X|^2} + (K-1)\sqrt{1 - |X|^2}\right)^2$$

4-PPM modulation with equiprobable symbols using the SRM method

$$|\beta_i\rangle = |\beta_{i,K-1}\rangle \otimes \cdots \otimes |\beta_{i,1}\rangle \otimes |\beta_{i,0}\rangle\,, \quad i = 0, 1, \ldots K-1$$

with $|\beta_{ij}\rangle = |\Delta\rangle$ for $i = j$ and $|\beta_{ij}\rangle = |0\rangle$ for $i \neq j$, where $|\Delta\rangle$ is the coherent state with real parameter Δ, and $|0\rangle$ is the "ground state".
For example, for $K = 4$ we have the four states listed as $|\gamma_0\rangle = |0\rangle \otimes |0\rangle \otimes |0\rangle \otimes |\Delta\rangle, |\gamma_1\rangle = |0\rangle \otimes |0\rangle \otimes |\Delta\rangle \otimes |0\rangle, |\gamma_2\rangle = |0\rangle \otimes |\Delta\rangle \otimes |0\rangle \otimes |0\rangle, |\gamma_3\rangle = |\Delta\rangle \otimes |0\rangle \otimes |0\rangle \otimes |0\rangle$.
We recall that all the states have the same number of signal photons per symbol, given by

$$N_s = \Delta^2$$

In the presence of thermal noise, the representation must be done in terms of density operators so that to the symbol i we associate the operator

$$\rho_i = \rho_{i,K-1} \otimes \cdots \otimes \rho_{i,1} \otimes \rho_{i,0}, \quad i = 0, 1, \ldots K-1$$

With

$$\rho_{ij} = \begin{cases} \rho(\Delta) & i = j \\ \rho(0) & i \neq j \end{cases}$$

From Proposition 8.3 the matrix representation of the ρ_i results in

$$R_i = R_{i,K-1} \otimes \cdots \otimes R_{i,1} \otimes R_{i,0}, \quad i = 0, 1, \ldots K-1$$
$$\underbrace{n \times n \quad n \times n}_{K \times n \text{ times}} \to n^K \times n^K = N \times N$$

With

$$R_{ij} = \begin{cases} R(\Delta) & \text{for } i = j \\ R(0) & \text{for } i \neq j \end{cases}$$

Let

$$\rho^0 = \beta^0 \left(\beta^0\right)^*, \quad \rho^1 = \beta^1 \left(\beta^1\right)^*$$

be the factorizations of the component density operators.
Then the factorizations of the composite operators, $\rho_i = \beta_i \beta_i^*$, turn out to be given by the factors

$$\beta_i = \beta_{i,K-1} \otimes \cdots \otimes \beta_{i,1} \otimes \beta_{i,0}, \quad i = 0, 1, \ldots K-1$$

Notice the perfect symmetry of expressions $|\beta_i\rangle = |\beta_{i,K-1}\rangle \otimes \cdots \otimes |\beta_{i,1}\rangle \otimes |\beta_{i,0}\rangle$, $\rho_i = \rho_{i,K-1} \otimes \cdots \otimes \rho_{i,1} \otimes \rho_{i,0}$ and $\beta_i = \beta_{i,K-1} \otimes \cdots \otimes \beta_{i,1} \otimes \beta_{i,0}$.

Matlab

Following the book, I set n = 35 and h = 4 for 4-PPM Modulation to ensure the accuracy $v = pe = 10^{-5}$

```
function [R] = density_operator(n, a , N)
% Pre-allocate the matrix with zeros
R = zeros(n, n);
if a==0
    for j=1:n
        R(j, j) = N^(j-1)/((N+1)^j);
    end
else

% Fill the matrix using the provided function f(m, n)
for i = 1:n
  for j = 1:n
    R(i, j) = N^(j-1)/((N+1)^j) * sqrt(factorial(i-1)/factorial(j-1)) * (conj(a)/N)^(j-
   i)  * exp(-(abs(a))^2/(N+1)) *laguerreL(i-1, j-i, (-(abs(a))^2/N/(N+1)) );
  end
end

end

end
```

LISTING 1. Matlab implementing density operator with Thermal noise

```
function [P_error] = PPM4(n, h, N , Ns)
K = 4;
if N==0
    X = exp(-Ns);
    P_error= 1-1/(K^2)*(sqrt(1 + (K-1)*X) + (K-1)* sqrt(1-X))^2;
else
delta = sqrt(Ns);
rho_delta = density_operator(n, delta, N);
rho_zero = density_operator(n, 0, N);
[Z1,D1] = eig(rho_delta);
Z_h1  = Z1(:, 1:h);
D_h1 = D1(1:h, 1:h);
```

```
beta_1 = Z_h1 * (D_h1 ^(1/2));

Z0= eye(n);
Z_h0  = Z0(:, 1:h);
D_h0 = rho_zero(1:h, 1:h);
beta_0 = Z_h0 * (D_h0 ^(1/2));

B_0=kron(beta_0, kron(beta_0, kron(beta_0, beta_1)));
B_1=kron(beta_0, kron(beta_0, kron(beta_1, beta_0)));
B_2=kron(beta_0, kron(beta_1, kron(beta_0, beta_0)));
B_3 =kron(beta_1, kron(beta_0, kron(beta_0, beta_0)));


W_k = exp(2* pi* 1i/K);
sum = zeros(h^K, h^K);
for i = 1:K
    D = B_0' * B_0  + B_0.'* B_1 * W_k^(-(i-1)) + B_0.' * B_2 * W_k^(-2*(i-1)) + B_0.'
   * B_3*W_k^(-3*(i-1));
    sum = sum + D^(1/2);
end

P_correct = real(trace(sum^2)/(K*K));
P_error = 1-P_correct;


end
end
```

LISTING 2. Matlab implementing PPM4 with SRM

I am plotting the error for 4-PPM for N=0:0.1:0.2 and Ns=0.5:0.5:8.

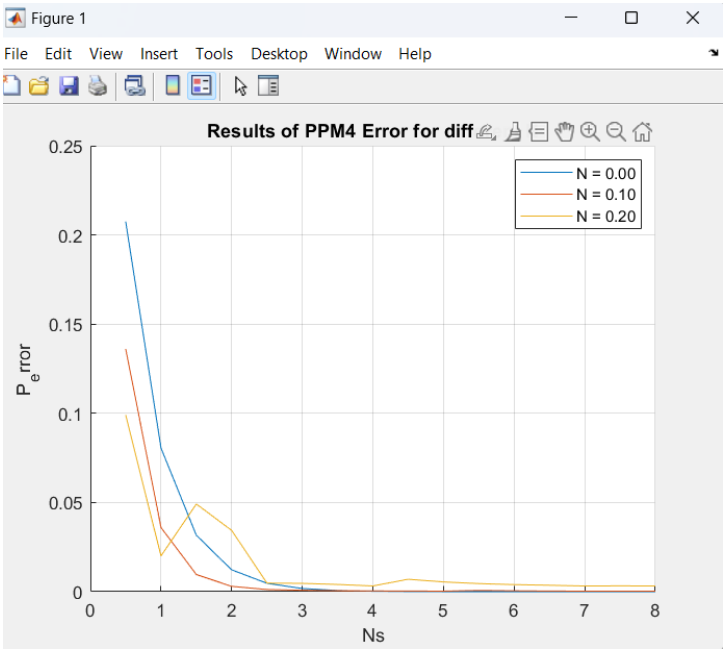

FIGURE 1. 4-PPM error

Upper bound from question 1

$$A^{+}=\sum_{\lambda>0}\lambda|\lambda\rangle\left\langle\lambda\right|,\quad A^{-}=\sum_{\lambda<0}\lambda\left|\lambda\right\rangle\langle\lambda|$$

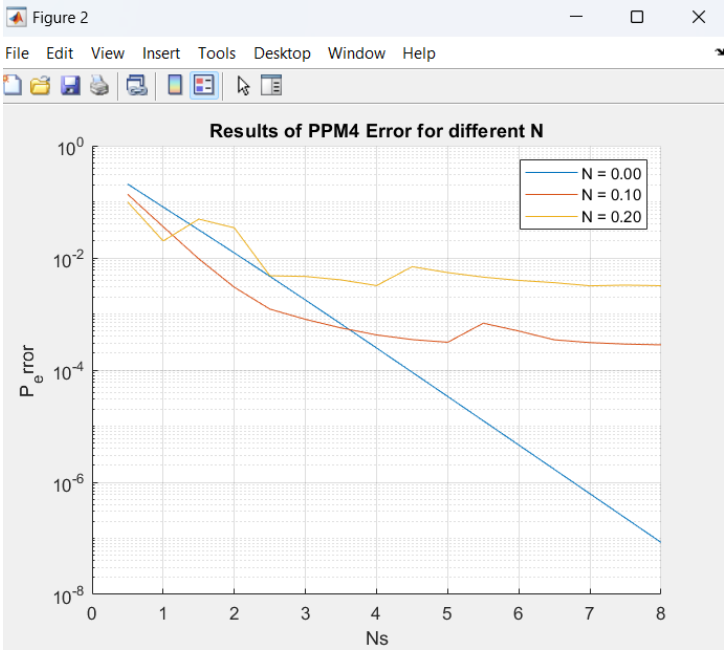

FIGURE 2. 4-PPm error semilogy

Define $F_i(i = 2, 3, \ldots, K)$ as

$$F_i = \rho_i - \rho_1 - \sum_{j=2}^{i-1} F_j^+$$

The optimum correct probability is bounded as

$$P_c \leq \mathrm{Tr}\left[\rho_1 + \sum_{i=2}^{K} F_i^+\right]$$

```
function [A_positive] = PositiveMatrix(A)
  % Find all eigenvalues and eigenvectors
  [V, D] = eig(A);

  % Find indices of positive eigenvalues
  positive_idx = D > 0;

  % Extract positive eigenvalues and corresponding eigenvectors
  positive_eigenvalues = diag(D(positive_idx, positive_idx));
  positive_eigenvectors = V(:, positive_idx);
  A_positive = positive_eigenvectors * positive_eigenvalues * positive_eigenvectors.';
end
```

LISTING 3. Matlab implementing positive matrix

```
function [bound] = upper_bound(K, n, h, N, Ns)
delta = sqrt(Ns);
rho_delta = density_operator(n, delta, N);
rho_zero = density_operator(n, 0, N);
[Z1,D1] = eig(rho_delta);
Z_h1  = Z1(:, 1:h);
D_h1 = D1(1:h, 1:h);
beta_1 = Z_h1 * (D_h1 ^(1/2));
```

```
9
10  Z0= eye(n);
11  Z_h0  = Z0(:, 1:h);
12  D_h0 = rho_zero(1:h, 1:h);
13  beta_0 = Z_h0 * (D_h0 ^(1/2));
14
15  B_0=kron(beta_0, kron(beta_0, kron(beta_0, beta_1)));
16  B_1=kron(beta_0, kron(beta_0, kron(beta_1, beta_0)));
17  B_2=kron(beta_0, kron(beta_1, kron(beta_0, beta_0)));
18  B_3 =kron(beta_1, kron(beta_0, kron(beta_0, beta_0)));
19  rho_0 = B_0 * B_0.';
20  rho_1 = B_1 * B_1.';
21  rho_2 = B_2 * B_2.';
22  rho_3 = B_3 * B_3.';
23
24  rho = {rho_0, rho_1, rho_2, rho_3};
25  F = {};
26  F{end+1} = rho{2} - rho{1};
27  S = PositiveMatrix(F{1});
28  for i=1:K-2
29      F{end + 1} = rho{i+2} - rho{1} - S;
30      S = S + PositiveMatrix(F{end});
31  end
32  bound = trace(rho{1} + S);
```

LISTING 4. Matlab implementing upper bound

This operation is out of memory!

Question 3. Plot and compare the error probability of 4PSK, 4QAM and 4PPM modulation formats. Consider equiprobable symbols in the absence of thermal noise, and choose NR = 0: 10.
The note and code for 4PPM are given in Question 2.

4QAM

We start from the construction of Gram's matrix G, whose elements are the inner products

$$\langle \gamma_{uv} \mid \gamma_{u'v'} \rangle = \langle \Delta(u + \mathrm{i}v) \mid \Delta\,(u' + \mathrm{i}v') \rangle$$

Remembering $\langle \alpha \mid \beta \rangle = \mathrm{e}^{-\frac{1}{2}\left(|\alpha|^2 + |\beta|^2 - 2\alpha^*\beta\right)}$, we get

$$\langle \gamma_{uv} \mid \gamma_{u'v'} \rangle = \exp\left\{-\frac{1}{2}\Delta^2\left[(u'-u)^2 + (v'-v)^2 - 2\mathrm{i}\,(u'v - v'u)\right]\right\}, \quad u, v, u', v' \in \mathcal{A}_L$$

$$P_e = 1 - \frac{1}{K}\sum_{i=0}^{K-1}\left[\left(G^{\frac{1}{2}}\right)_{ii}\right]^2$$

$$N_S = \frac{2}{3}\left(L^2 - 1\right)\Delta^2 = \frac{2}{3}(K-1)\Delta^2$$

Finally, from N_s, we get the number of signal photons per bit as

$$N_R = N_s / \log_2 K$$

4PSK

The generic element p, q of Gram's matrix $G = [G_{pq}]$ is the inner product $G_{pq} = \langle \gamma_p \mid \gamma_q \rangle$ obtained from

$\langle \alpha \mid \beta \rangle = \mathrm{e}^{-\frac{1}{2}\left(|\alpha|^2+|\beta|^2-2\alpha^*\beta\right)}$ with $\alpha = \Delta W_K^p$ and $\beta = \Delta W_K^q$, namely,

$$G_{pq} = \langle \gamma_p \mid \gamma_q \rangle = \exp\left[-\Delta^2\left(1 - W_K^{q-p}\right)\right], \quad p, q = 0, 1, \ldots, K-1$$

As predicted (by the GUS), the element p, q depends only on the difference $q - p$. And therefore Gram's matrix becomes circulant. The eigenvalues are obtained by computing the DFT of the first row of Gram's matrix, that is,

$$\lambda_i = \sum_{k=0}^{K-1} G_{0k} W_K^{-ki}, \quad \text{for } i = 0, \quad \lambda_0 = \sum_{k=0}^{K-1} G_{0k} \quad \left(\begin{array}{c}\text{The sum of the}\\ \text{first row of } G\end{array}\right)$$

and the corresponding eigenvectors are given by the columns of the DFT matrix.

$$|w_i\rangle = \frac{1}{\sqrt{K}}\left[1, W_K^{-i}, W_K^{-2i}, \ldots, W_K^{-(K-1)i}\right]^{\mathrm{T}}$$

Thus the matrices $G^{\pm\frac{1}{2}}$ are obtained from $\left(G^{\pm\frac{1}{2}}\right)_{ij} = \frac{1}{K}\sum_{p=0}^{K-1} \lambda_p^{\pm\frac{1}{2}} W_K^{-p(j-i)}$, where the element i, j is given by

$$\left(G^{\pm\frac{1}{2}}\right)_{ij} = \frac{1}{K}\sum_{p=0}^{K-1} \lambda_p^{\pm\frac{1}{2}} W_K^{(j-i)p}$$

The measurement vectors are computed as a linear combination of the states according to $|\mu_i\rangle = \sum_{j=0}^{K-1}\left(G^{-\frac{1}{2}}\right)_{ji}|\gamma_j\rangle$, i.e.,

$$|\mu_i\rangle = \sum_{j=0}^{K-1}\left(G^{-\frac{1}{2}}\right)_{ji}|\gamma_j\rangle$$

Finally, the error probability with equiprobable symbols is simply

$$P_e = 1 - \frac{1}{K^2}\left(\sum_{i=0}^{K-1}\sqrt{\lambda_i}\right)^2$$

Therefore, to calculate P_e it suffices to evaluate the eigenvalues according to $\lambda_i = \sum_{k=0}^{K-1} G_{0k} W_K^{-ki}$ and to apply the relation above.

Matlab

```
function [P_error] = QAM(Ns, K)
L=sqrt(K);
G= zeros(K, K);
delta = sqrt(3/2/(K-1)*Ns);
row_count= 1;
col_count= 1;
for i=1:L
    for j=1:L
       u = -(L-1) + 2*(i-1);
       v = -(L-1) + 2*(j-1);
        for k=1:L
            for p=1:L
                 u_prime = -(L-1) + 2*(k-1);
                 v_prime = -(L-1) + 2*(p-1);
                 alpha = u + 1i*v;
                 beta = u_prime + 1i*v_prime;
```

```
                    G(row_count, col_count) = exp(-delta^2 *(abs(alpha)^2 + abs(beta)^2 -
    2*conj(alpha)*beta));
                    col_count = col_count+1;
                end
            end
            col_count = 1;
            row_count = row_count +1;
        end
    end

    X = sqrtm(G);
    P_correct = 0;
    for i=1:K
        P_correct = P_correct + X(i, i)*conj(X(i, i));
    end
    P_correct = P_correct/K;
    P_error = 1-P_correct;
```

LISTING 5. Matlab implementing QAM

```
function [P_error] = PSK4(Ns)
K=4;
delta = sqrt(Ns);
sum = 0;
W_k = exp(2*pi*1i/K);
for i=1:K
    lamda=0;
    for j=1:K
        lamda = lamda + exp(-delta^2*(1-W_k^(j-1)))*W_k^(-(i-1)*(j-1));
    end
    sum = sum + sqrt(lamda);
end
P_error = real(1-sum^2/K/K);
```

LISTING 6. Matlab implementing PSK4

I am plotting the error for 4-PPM for NR = 0:0.1:10;

Compare

PSK4 and QAM4 have the same spectral efficiency (Ns) for a fixed symbol rate (Ns). However, for a fixed bit rate (NR), PSK4 requires half the number of symbols compared to QAM4. Additionally, PPM4 did way better than both QAM4 and PSK4.

Question 4. Find the values of n_ϵ and n_v in section 8.4.2 of the book for $N = 0.1$, $N_\gamma = 5$ and $\epsilon = \nu = 10^{-5}$. Compare your results with table 8.1 and explain your points of view.
From the table, it can be concluded that to satisfy the given condition, $n_\epsilon = 22$ and $n_v = 4$ are needed.
In the EID we have two approximations:
1. the approximation R of the density operator ρ by a finite number n of terms,
2. the approximation of the true rank n of the matrix R by the virtual rank h, which allows us to find the approximate factor $\tilde{\beta}$ of ρ of dimensions $n \times h$.
For both approximations, we need to establish a criterion that guarantees a given accuracy.
For the first approximation, the trace criterion seems to be more appropriate. We recall that a density operator has always unitary trace and that $\mathrm{Tr}[\rho] = 1$ also represents the normalization condition of the Laguerre distribution $p_n(k \mid \gamma)$, which gives the meaning of the diagonal elements.
Hence, only if $\mathrm{Tr}[R]$ gives a value close to one, the approximation is satisfactory, and, having fixed an

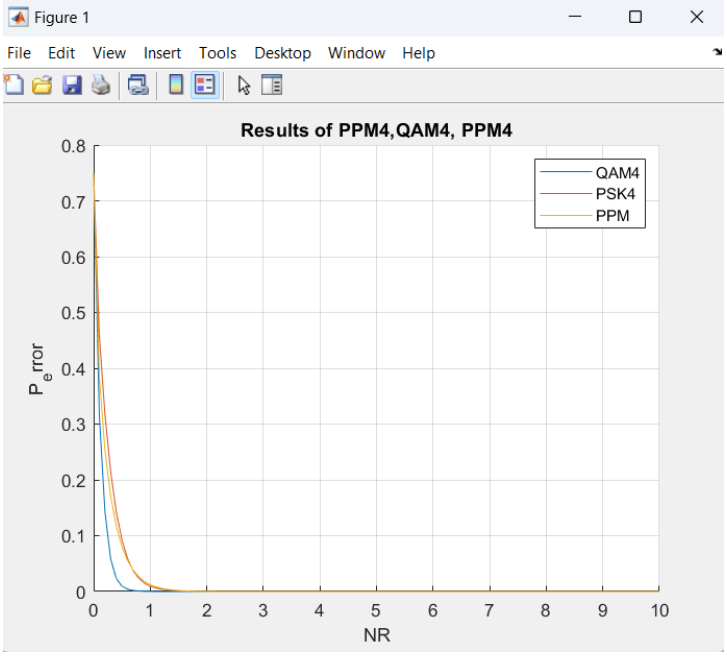

FIGURE 3. QAM4,PSK4,PPM4 error

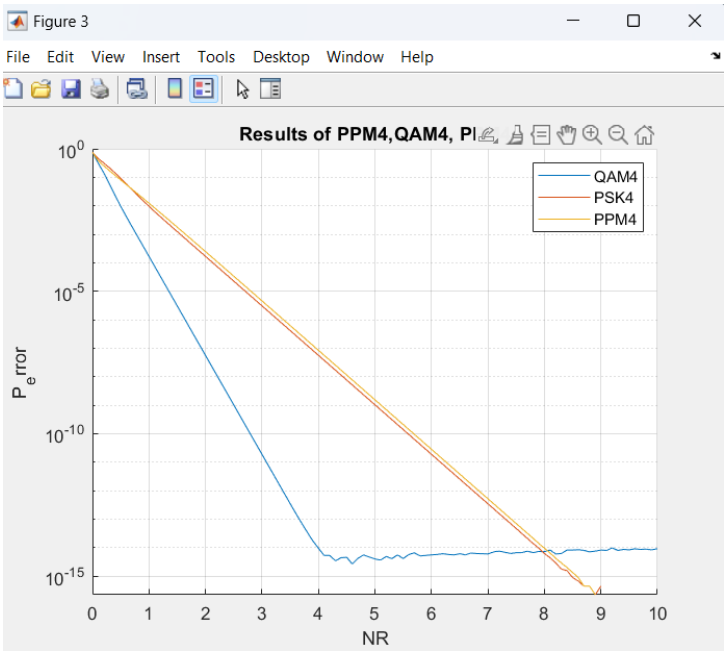

FIGURE 4. QAM4,PSK4,PPM4 error semilogy

accuracy ε, the number of terms $n = n_\varepsilon$ must be calculated according to the condition

$$\mathrm{Tr}[R] = \sum_{k=0}^{n_\varepsilon - 1} R_{kk} = \sum_{k=0}^{n_\varepsilon - 1} p_n(k \mid \gamma) \geq 1 - \varepsilon$$

For the second approximation, one can use the reconstruction criterion, based on the square error between the matrix R and its "reconstruction" $\beta\beta^*$, where both R and $\beta\beta^*$ are $n \times n$ matrices, with n previously evaluated.
Hence, having fixed the accuracy v, we evaluate the virtual rank $n_v = h$ such that

$$\mathrm{mse}\,(R - \beta\beta^*) \leq v$$

Matlab I used the function density operator which is explained in question 1, to construct the density operator here.
1. Check the first condition. I constructed the density operator for $n = 25$ and calculated the error, which is the difference of trance from value 1, for $n = 15$ to $n = 25$.

$$p_e = 1 - trace(\rho)$$

It can be seen from Table 1, that for $n_\epsilon \geq 20$, we have $p_e < 10^{-5}$

n	15	16	17	18	19	20	21	22	23	24	25
p_e	0.0011	4.30e0-4	1.69e-4	6.41e-5	2.36e-5	8.43e-6	2.93e-6	9.94e-7	3.29e-7	1.06e-7	3.36e-8

Table 1. Error for different n_ϵ

2. Check the second condition. I constructed the density operator for $n_\epsilon = 20, 21, 22, 23$ and calculated the error, which is the reconstruction error for each n_ϵ, for $n_v = 2, 3, 4, 5, 6$.

$n_\epsilon \vert n_v$	2	3	4	5	6
20	1.4220e − 07	1.1675e − 09	9.2810e − 12	6.6425e − 14	3.8599e − 16
21	1.2904e − 07	1.0639e − 09	8.6449e − 12	6.6222e − 14	4.3823e − 16
22	1.1759e − 07	9.7105e − 10	7.9711e − 12	6.3522e − 14	4.6173e − 16
23	1.0759e − 07	8.8897e − 10	7.3287e − 12	5.9611e − 14	4.6052e − 16

Table 2. Error for different n_ϵ

We can see that for all of the suggested n_ϵ and n_v, the error pe and v are less than 10^{-5}!

Question 5. Obtain the following results from the book.
a) Page 390: Fig. 8.14
b) Pages 394: Example with Fig. 8.17
a) Error probability in OOK system as a function of Ns for some values of N.

Binary Systems in the Presence of Thermal Noise
(1) We start from the EID of the decision operator

$$D = \frac{1}{2}\left[\rho\left(\beta_1\right) - \rho\left(\beta_0\right)\right] = \sum_k \eta_k \left|\eta_k\right\rangle \left\langle\eta_k\right|$$

where η_k are the eigenvalues and $|\eta_k\rangle$ the corresponding eigenvectors,
(2) The two optimal measurement operators are given by the sum of the elementary projectors $|\eta_k\rangle \langle\eta_k|$, and specifically

$$Q_0 = \sum_{\eta_k<0} \left|\eta_k\right\rangle \left\langle\eta_k\right|, \quad Q_1 = \sum_{\eta_k>0} \left|\eta_k\right\rangle \left\langle\eta_k\right|$$

(3) The maximum correct decision probability is obtained as the sum of the positive eigenvalues according to

$$P_c = \frac{1}{2} + \sum_{\eta_k>0} \eta_k \quad \rightarrow \quad P_e = \frac{1}{2} - \sum_{\eta_k>0} \eta_k$$

Using the expansion of $\rho(\gamma)$ for the decision operator D one gets the following matrix representation $[D_{mn}]$

of D

$$D_{mn} = \frac{1}{2}\left[R_{mn}\left(\gamma_1\right) - R_{mn}\left(\gamma_0\right)\right], \quad m, n = 0, 1, 2, \ldots$$

To proceed we need to approximate this matrix to finite dimensions $n \times n$ according to the criterion explained in question 4.

Application of OOK Modulation

The constellation of scaled symbols is

$$\gamma_0 = 0 \quad \gamma_1 = \alpha$$

for equally likely symbols (q0 = q1 = 1/2):

$$N_s = N_a/2$$

Without Thermal noise, we have:
From $P_e = \frac{1}{2}\left[1 - \sqrt{1 - |X|^2}\right]$, we obtain that the error probability of the OOK quantum system with equiprobable symbols becomes

$$P_e = \frac{1}{2}\left[1 - \sqrt{1 - \mathrm{e}^{-2N_R}}\right]$$

Matlab
I used the function density operator explained in question 1 for constructing the density operators.

```
function [P_error] = OOK(n, Ns, N)
if N==0
    P_error= 1/2 * (1-sqrt(1-exp(-2*Ns)));
else
a =sqrt(2*Ns);
Des = 1/2 * (density_operator(n,a, N) - density_operator(n, 0, N));
[V, D] = eig(Des);
sum=0;
for i=1:n
    if D(i, i)>0
        sum = sum + D(i, i);
    end
end
P_error = 1/2 -sum;
end
```

Listing 7. Matlab implementing OOK

I am plotting the error of OOK for N = 0, 0.05, 0.1, 0.2 and N_s = 05:0.5:8. figure 5.

b) Comparison of error probability in 16-QAM versus Ns for some values of N. Black lines refer to CVX evaluation and red lines to SRM evaluation.

QAM in the presence of Thermal noise
In the presence of thermal noise we have to consider a constellation of $K = L^2$ Glauber density operators

$$\rho_{uv} = \rho(\Delta(u + iv)), \quad u, v \in \mathcal{A}_L$$

(1) SRM Method To evaluate the performance in the absence of thermal noise, we started from the Gram matrix G, whose elements are the inner products $\langle \gamma_{uv} \mid \gamma_{u'v'} \rangle, u, v, u', v' \in \mathcal{A}_L$.
Having computed the eigenvalues and eigenvectors of G, we calculated the square root $G^{\frac{1}{2}}$, and hence the

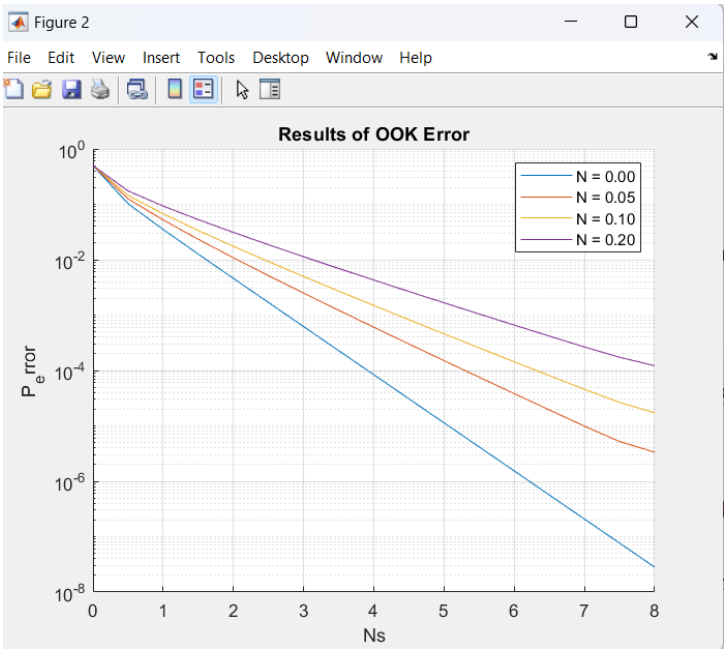

FIGURE 5. OOk error

error probability according to

$$P_{e,0} = 1 - \frac{1}{K} \sum_{i=0}^{K-1} \left[\left(G^{\frac{1}{2}} \right)_{ii} \right]^2,$$

(absence of thermal noise)
The performance was evaluated as a function of the number of signal photons per symbol, which for the QAM with equiprobable symbols results in

$$N_s = \frac{2}{3} \left(L^2 - 1 \right) \Delta^2 = \frac{2}{3} (K-1) \Delta^2$$

```
function [P_error] = QAM_SRM(n, h, Ns, N, K)
if N==0
    P_error = QAM(Ns, K);
else
L=sqrt(K);
Gamma = [];
delta = sqrt(3/2/(K-1)*Ns);
for i=1:L
    for j=1:L
       u = -(L-1) + 2*(i-1);
       v = -(L-1) + 2*(j-1);
       a = (u + 1i*v)* delta;
       rho = density_operator(n,a, N);
       [Z,D] = eig(rho);
       Z_h  = Z(:, 1:h);
       D_h = D(1:h, 1:h);
       beta = Z_h * (D_h ^(1/2));
       Gamma = [Gamma, beta];

    end
end
G = Gamma.' * Gamma;
X = sqrtm(G);
```

```
P_correct = 0;
for i=1:K
    P_correct = P_correct + X(i, i)*conj(X(i, i));
end
P_correct = P_correct/K;
P_error = 1-P_correct;
end
```

LISTING 8. Matlab implementing QAM with SRM method

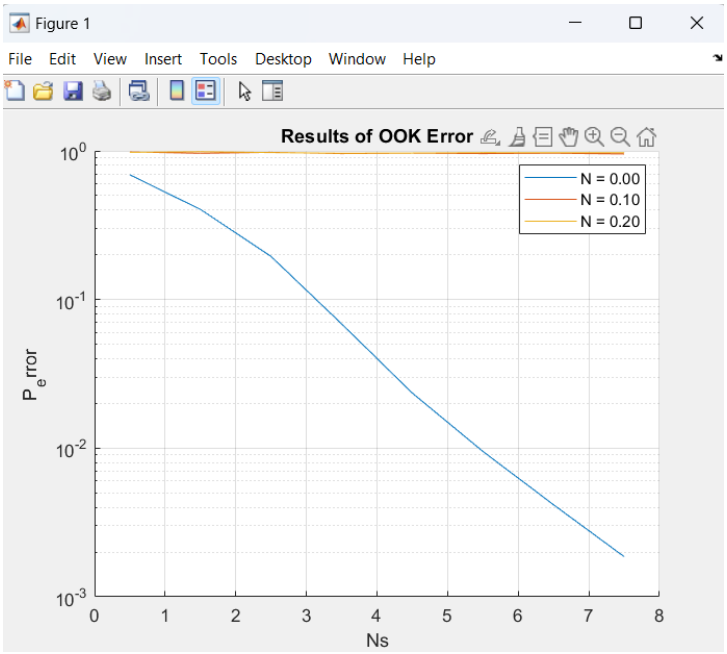

FIGURE 6. 16 QAM error

(2) CSP Method

n=150 (!) is needed, Which due to the shortage of time the simulation was not completed before the deadline.

```
function [P_error] = QAM_CSP(n, N, Ns)
K=16;
L=sqrt(K);
rho = {};
delta = sqrt(3/2/(K-1)*Ns);
for i=1:L
    for j=1:L
       u = -(L-1) + 2*(i-1);
       v = -(L-1) + 2*(j-1);
       a = (u + 1i *v) * delta
       rho{end +1} =density_operator(n,a, N);

    end
end
dim = n;

cvx_begin SDP
variable X(dim, dim) hermitian
minimize(trace(X))
subject to
X> rho{1};
X> rho{2};
```

```
X> rho{3};
X> rho{4};
X> rho{5};
X> rho{6};
X> rho{7};
X> rho{8};
X> rho{9};
X> rho{10};
X> rho{11};
X> rho{12};
X> rho{13};
X> rho{14};
X> rho{15};
X> rho{16};
cvx_end
copt=cvx_optval;
t=(copt)
t=trace(X)
P_error=1.0-t;
```

LISTING 9. Matlab implementing QAM with CSP method