

Rareș Călin BUTA
Cristian CODĂU
Andra PASTRAV
Emanuel PUȘCHIȚĂ

RADIOCOMUNICAȚII CELULARE

Îndrumător de laborator



U.T.PRESS
Cluj-Napoca, 2025
ISBN 978-606-737-816-0

Rareș-Călin BUTA
Cristian CODĂU
Andra PASTRAV
Emanuel PUȘCHIȚĂ

RADIOCOMUNICAȚII CELULARE

Îndrumător de laborator



U.T.PRESS
Cluj-Napoca, 2025
ISBN 978-606-737-816-0



Editura U.T.PRESS
Str. Observatorului nr. 34
400775 Cluj-Napoca
Tel.: 0264-401.999
e-mail: utpress@biblio.utcluj.ro
www.utcluj.ro/editura

Recenzia: Prof.dr.ing. Tudor Palade
 Prof.dr.ing. Marina Țopa

Pregătire format electronic on-line: Gabriela Groza

Copyright © 2025 Editura U.T.PRESS
Reproducerea integrală sau parțială a textului sau ilustrațiilor din această carte este posibilă numai cu acordul prealabil scris al editurii U.T.PRESS.

ISBN 978-606-737-816-0

Prefață

Acest îndrumător de laborator cuprinde 12 lucrări gândite pentru a sprijini activitatea de laborator a studenților de la Facultatea de Electronică, Telecomunicații și Tehnologia Informației din cadrul Universității Tehnice din Cluj-Napoca, care studiază sistemele de radiocomunicații celulare.

Lucrările de laborator care compun acest material pot fi grupate în două tematici majore, care își propun să urmărească, consolideze și completeze traseul logic al subiectelor prezentate în cadrul cursului de Radiocomunicații Celulare.

Primele șase lucrări de laborator sunt dedicate simulării caracteristicilor fundamentale ale canalului radio, precum și fenomenelor care influențează propagarea undelor electromagnetice. Aceste lucrări practice pun bazele înțelegerii mai multor modele de propagare radio în context fix și mobil, a proceselor stocastice care influențează transmisia semnalului, precum și a tehnicilor moderne de simulare tridimensională a mediului de propagare.

Ultimele șase lucrări se concentrează pe planificarea, dimensionarea și analiza rețelelor celulare. Aici sunt abordate noțiuni și parametri de bază asociați sistemelor mobile, aspecte legate de planificarea rețelelor celulare, elemente de gestionarea resurselor radio și a mobilității utilizatorilor, precum și tehnologii și metrici specifice standardelor actuale.

Studenții sunt ghidați pe parcursul fiecărei lucrări de laborator prin concepte și elemente de teorie fundamentale în domeniul radiocomunicațiilor celulare, urmate de o validare aplicativă a acestora prin simulări și analize MATLAB. Pe măsură ce parcurg materialul, studenții completează *Fișa de lucru* aferentă fiecărui laborator, care include ajustări/corecții de parametri sau script, analiza rezultatelor obținute și interpretarea reprezentărilor grafice generate.

Toate aceste elemente sunt concepute pentru a le consolida studenților înțelegerea cunoștințelor teoretice dobândite și pentru a le dezvolta acestora competențe profesionale relevante în domeniul radiocomunicațiilor celulare.

Autorii
Cluj-Napoca, 2025

Cuprins

Lucrarea 1 – Modelul Friis. Impactul condițiilor meteo asupra propagării radio.....	1
1.1 Ecuația de transmisie Friis	1
1.2 Bugetul legăturii radio.....	1
1.3 Sensibilitatea receptorului	2
1.4 Impactul condițiilor meteo asupra propagării radio	2
Script-uri Matlab	4
Fișă de lucru	8
Lucrarea 2 – Modelul cu două raze. Modelul Okumura-Hata.....	11
2.1 Modelul cu două raze	11
2.2 Propagarea în zone locuite	12
Script-uri Matlab	15
Fișă de lucru	19
Lucrarea 3 – Difracția. Modelul log-normal. Zona Fresnel.....	21
3.1 Fenomenul de umbră	21
3.2 Modelul log-normal de propagare.....	22
3.3 Zona Fresnel.....	23
3.4 Modelarea pierderilor de propagare datorate difracției.....	24
Script-uri Matlab	26
Fișă de lucru	31
Lucrarea 4 – Efectul Doppler	35
4.1 Descrierea fenomenului.....	35
4.2 Modelul cu două raze în scenariu mobil	36
4.3 Dispersia semnalului	37
Script-uri Matlab	40
Fișă de lucru	47
Lucrarea 5 – Analiza propagării multicanale.....	53
5.1 Profilul de putere al întârzierilor	53
5.2 Fading.....	54
Script-uri Matlab	57
Fișă de lucru	61
Lucrarea 6 – Analiza legăturii radio în medii 3D folosind Ray Tracing	63
6.1 Modelul Ray Tracing	63
6.2 Pași de Implementare	63
Script-uri Matlab	66
Fișă de lucru	70
Lucrarea 7 – Conceptul celular. Reutilizarea frecvențelor. Interferențe specifice sistemelor celulare.....	75
7.1 Principiile conceptului celular.....	75
7.2 Reutilizarea frecvențelor	75
7.3 Interferența co-canal.....	76
7.4 Interferența canalului adiacent	77
Script-uri Matlab	79
Fișă de lucru	87
Lucrarea 8 – Divizarea celulei. Sectorizarea celulei. Procedura de handover.....	91
8.1 Divizarea celulei.....	91
8.2 Sectorizarea celulei	91
8.3 Procedura de handover	93
Script-uri Matlab	95
Fișă de lucru	107

Lucrarea 9 – Traficul celular	111
9.1 Traficul celular	111
9.2 Tipuri de sisteme trunked	112
Script-uri Matlab	114
Fișă de lucru	120
Lucrarea 10 – Organizarea resurselor radio în GSM	123
10.1 Schemele de acces multiplu în GSM	123
10.2 Salvele de comunicație	124
Script-uri Matlab	129
Fișă de lucru	134
Lucrarea 11 – Tehnici de creștere a eficienței spectrale în sistemele celulare	137
11.1 Introducere	137
11.2 Diversitatea la recepție	137
11.3 Diversitatea la transmisie	139
Script-uri Matlab	142
Fișă de lucru	149
Lucrarea 12 – Măsurarea și analiza parametrilor radio în rețele celulare folosind terminale comerciale.....	153
12.1 Parametrii investigați	153
12.2 Resurse necesare	154
12.3 Activitate de teren	154
12.4 Procesarea datelor în Matlab	155
12.5 Analiză măsurători și simulare	155
Script-uri Matlab	157
Fișă de lucru	163

Lucrarea 1 – Modelul Friis. Impactul condițiilor meteo asupra propagării radio

Un sistem de comunicații cuprinde trei elemente cheie: emițătorul, receptorul și mediul de propagare. Lucrarea de față urmărește prezentarea parametrilor ce pot fi configurați în scripturile Matlab la nivelul stațiilor emițătoare, receptoare și respectiv al condițiilor de mediu, precum și efectul acestora asupra sistemului de comunicații.

Obiective

- O1. Prezentarea ecuației Friis.
- O2. Calcularea bugetului legăturii și a razei maxime de transmisie.
- O3. Efectul parametrilor de canal asupra atenuării semnalului.
- O4. Elaborarea unui program Matlab care să ilustreze gradul de acoperire al unui punct de acces WiFi.

Durata medie de studiu individual: 30 minute.

1.1 ECUAȚIA DE TRANSMISIE FRIIS

Figura 1.1 ilustrează un sistem simplificat de comunicații. Emițătorul are o putere de emisie P_{tx} și un câștig al antenei G_{tx} . Semnalul este recepționat de o antenă cu un câștig G_{rx} . Semnalul recepționat la distanța R are puterea P_{rx} .

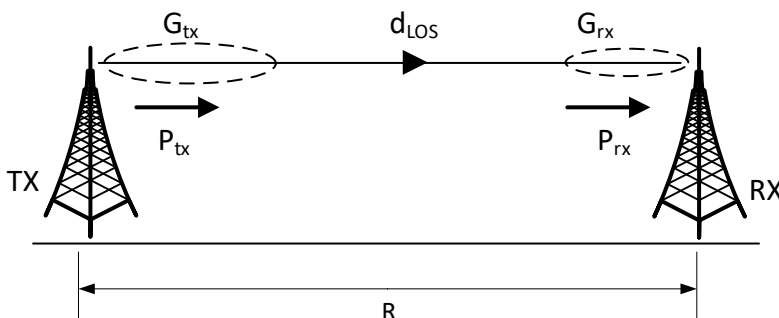


Figura 1.1 Sistem de comunicații radio simplificat.

Dacă se consideră că nu există pierderi atmosferice, nu sunt dezadaptări pe feeder-ele antenelor, antenele sunt în vizibilitate directă și sunt polarizate corespunzător, pentru sistemul simplificat din Figura 1.1, ecuația Friis [1] se poate scrie ca (1.1).

$$P_{rx} = P_{tx} \frac{G_{tx} G_{rx} \lambda_0^2}{(4\pi R)^2} \quad (1.1)$$

Unde:

- P_{rx} - puterea de recepție [W];
- P_{tx} - puterea de transmisie [W];
- G_{tx} - câștigul antenei de transmisie (adimensional);
- G_{rx} - câștigul antenei de recepție (adimensional);
- λ_0 - lungimea de undă în spațiu liber [m];
- R - distanța dintre emițător și receptor [m].

Prin urmare, puterea recepționată este proporțională cu câștigul antenei și invers proporțională cu pătratul distanței dintre antene.

1.2 BUGETUL LEGĂTURII RADIO

Dacă pentru sistemul din Figura 1.1 se iau în considerare și diferite pierderi în sistem, notate cu L_{sis} , atunci puterea la recepție se calculează conform (1.2).

$$P_{rx} = P_{tx} G_{tx} G_{rx} \left(\frac{\lambda_0}{4\pi R} \right)^2 \frac{1}{L_{sis}} \quad (1.2)$$

Unde L_{sis} reprezintă pierderile sistemului (datorate feed-erelor antenei, polarizării incorecte, fenomenelor atmosferice, etc.).

Ecuția (1.2), poartă numele de bugetul legăturii și poate fi exprimată în dB astfel [1]:

$$P_{rx \text{ dB}} = P_{tx \text{ dB}} + G_{tx \text{ dB}} + G_{rx \text{ dB}} + 20 \lg \left(\frac{\lambda_0}{4\pi R} \right) - L_{sis \text{ dB}} \quad (1.3)$$

Pierderile de propagare în spațiu liber sunt pierderile suferite de semnal în timpul propagării:

$$P_L(\text{dB}) = 20 \log \left(\frac{4\pi R}{\lambda_0} \right) \quad (1.4)$$

1.3 SENSIBILITATEA RECEPTORULUI

Dacă la recepție se detectează semnalul minim de recepție al sistemului ($S_{\min}$), atunci distanța maximă de transmisie dintre emițător și receptor se poate scrie ca [2]:

$$R_{max} = \sqrt{\frac{P_{tx} G_{tx} G_{rx} \lambda_0^2}{(4\pi)^2 S_{\min} L_{sis}}} \quad (1.5)$$

Sarcina 1

Se consideră o comunicație cu vizibilitate directă între un emițător și un receptor situate la distanța de 100 m unul față de celălalt, frecvența de operare fiind de 900 MHz. Presupunem că puterea de transmisie este de 0.1 W și atât emițătorul, cât și receptorul au un câștig al antenei de 5 dB. Dacă sensibilitatea receptorului este de -85 dBm, care este marginea de zgomot permisă legăturii de comunicații? (*Rezolvați aplicația în [Fișa de lucru](#)*).

1.4 IMPACTUL CONDIȚIILOR METEO ASUPRA PROPAGĂRII RADIO

Propagarea în spațiu liber cu vizibilitate directă (Line-of-Sight - LOS) descrie doar un aspect particular referitor la atenuarea semnalului. Apropiindu-ne de cazurile reale de propagare, trebuie să menționăm că semnalele interacționează cu particulele din aer și pierd energie de-a lungul căii lor de propagare. Pierderile variază în funcție de diferiți factori, cum ar fi presiunea atmosferică, temperatura sau densitatea vaporilor de apă din atmosferă.

Ploaia poate fi un factor major în ce privește atenuarea semnalului în cazul sistemelor de comunicații, mai ales atunci când acestea lucrează la frecvențe de peste 5 GHz. În modelul ITU din [3], ploaia este caracterizată de debitul precipitațiilor (în mm / h). Conform [4], precipitațiile pot varia de la mai puțin de 0,25 mm / h pentru ploi foarte slabe la peste 50 mm/h pentru averse puternice.

În plus, datorită formei picăturii de ploaie și a dimensiunii sale relative la lungimea de undă a semnalului de radiofrecvență, pierderea de propagare datorată ploii este, de asemenea, o funcție de polarizare a semnalului.

Sarcina 2

Rulați scriptul [L1 Propagare LOS.m](#) în mediul de programare Matlab pentru a evalua variația pierderilor de propagare în spațiu liber. Modificați parametrii relevanți astfel încât să efectuați teste pentru diferite distanțe între stația de bază și terminalul mobil, precum și pentru diferite frecvențe de operare. Cu rezultatele obținute completați tabelul din [Fișa de lucru](#). Mai întâi, asigurați-vă că ați corectat funcția [LOS.m](#).

Reprezentați grafic pierderile de propagare în spațiu liber pentru intervalul de frecvențe 900 MHz-5 GHz și pentru distanțele 100 m, 1 km, 10 km. Adăugați reprezentarea grafică în [Fișa de lucru](#). Ce se poate observa odată cu creșterea distanței dintre emițător și receptor? Dar odată cu creșterea frecvenței de lucru? Notați în secțiunea de observații răspunsurile.

Sarcina 3

Considerăm un semnal polarizat orizontal, deci unghiul de înclinație al antenei de emisie în raport cu solul este 0° . În general, polarizarea orizontală reprezintă cazul cel mai defavorabil pentru propagarea în condiții de ploaie [5].

- Rulați scriptul [L1_Propagare_Precipitatii.m](#) în mediul de programare Matlab. Calculați atenuarea suplimentară a semnalului datorată ploii pentru o frecvență din banda Ku (12-18 GHz) pentru distanța de 550 km în cazul condițiilor de aversă (50 mm/h), ploaie moderată (25 mm/h), respectiv ploaie ușoară (5 mm/h). Comparați rezultatele și notați observațiile în [Fișa de lucru](#).
- Reprezentați grafic atenuarea semnalului datorată mai multor tipuri de debite de precipitații (1 mm/h, 4 mm/h, 16 mm/h, 50 mm/h) pentru semnale situate în intervalul de frecvențe [10 GHz, 100 GHz]. Distanța dintre emițător și receptor este de 10 km. Notați observațiile în [Fișa de lucru](#).

Ceața și norii sunt formați din picături de apă, care sunt mult mai mici în comparație cu picăturile de ploaie. Dimensiunea picăturilor de ceață este în general mai mică de 0,01 cm. Ceața este adesea caracterizată prin densitatea vaporilor de apă în aer. O ceață medie cu o vizibilitate de aproximativ 300 m, are o densitate de $0,05 \text{ g / m}^3$ [4].

- Rulați [L1_Propagare_Conditii_Meteo.m](#) și comparați pentru distanța de 550 km atenuările semnalului relativ la diferite condiții meteo (ploaie puternică – 50 mm/h, ceață densă – 0.5 g / m^3), respectiv atenuarea corespunzătoare propagării în condiții de vizibilitate directă. Utilizați ca frecvență de lucru o valoare din banda Ka (27 GHz – 40 GHz). Adăugați graficul obținut în [Fișa de lucru](#).

Sarcina 4

Analizați pierderile de propagare în cazul unui punct de acces WiFi care poate opera atât în banda de 2.4 GHz cât și în banda de 5 GHz. Puterea de emisie este de 3 dBm, câștigul antenelor de emisie și recepție este de 1 dBi. Pierderile sistemului se consideră a fi $L_{\text{sis}} = 1 \text{ dB}$. Folosind ecuația lui Friis, reprezentați grafic puterea recepționată în dBm pentru un interval de la 1 m până la 32 m. Porniți de la macheta furnizată în [L1_Acoperire_WiFi.m](#) Treceți atât programul elaborat cât și graficul rezultat de pe urma rulării lui în [Fișa de lucru](#) la secțiunea corespunzătoare acestei cerințe. Care este atenuarea semnalului cu fiecare dublare a distanței?

Bibliografie

- [1] E. Puschita, s.a., *Radiocomunicatii Celulare - canalul radio - antene - proiectarea sistemelor – Manual de laborator*, U.T. PRESS, ISBN 978-973-662-496-4, 170 pag., 2009.
- [2] John S. Seybold, Ph.D., *Introduction to RF Propagation*, John Wiley & Sons, Inc., Hoboken, New Jersey, 2005, pp. 66.
- [3] Recommendation ITU-R P.838-3, 2005
- [4] Recommendation ITU-R P.840-3, 2013
- [5] „Signal Attenuation Due to Rainfall,” Mathworks.com, 2017, https://se.mathworks.com/help/phased/ref/rainpl.html#bu6e5ui_sep_bu6e5ui-elev.

SCRIPT-URI MATLAB

L1 Propagare LOS.m

```

clc;
clear;
%% =====
%   Propagare LOS (Line-of-Sight) - Calcul tabelar
%   =====

c0 = physconst('LightSpeed'); % viteza luminii [m/s]
R0 = ?; % distanța între Tx și Rx [m]
f0 = ?; % frecvența semnalului transmis [Hz]

% Calculul pierderii de propagare în condiții LOS
PL_LOS = LOS(?, ?); % completează parametrii funcției corespunzător
fprintf('Atenuarea LOS conform Friis la %.1f GHz pe o distanță de %.1f km
este: %.2f dB\n', ...
        f0/1e9, R0/1e3, PL_LOS);

%% =====
%   Reprezentare grafică - Pierderi pe calea de propagare în spațiu liber
%   =====

R0 = [?; ?; ?]; % distanțe [m]
f0 = (?:?):.' * 1e6; % interval de frecvențe [Hz]

% Calcul pierdere de propagare în spațiu liber
PL_graph = LOS(R0, f0);

% Reprezentare log-log
plot(f0/1e9, PL_graph);
grid on;
legend('Range: 100 m', 'Range: 1 km', 'Range: 10 km');
xlabel('Frecvență (GHz)');
ylabel('Pierdere de propagare (dB)');
title('Free Space Path Loss');

```

LOS.m

```

function PL_dB = LOS(d, f)
% Intrări:
%   d - distanța Tx-Rx [m] (scalar sau vector)
%   f - frecvența purtătoare [Hz] (scalar sau vector compatibil)
%
% ieșiri:
%   PL_dB - pierderea de propagare în spațiu liber [dB]

c = 3e8; % viteza luminii [m/s]

% Lungimea de undă [m]
lambda = c ./ f;

% Pierderea de propagare în spațiu liber [dB]
PL_dB = 20*log10(4*pi.*d) - 20*log10(lambda);

end

```

L1 Propagare Precipitatii.m

```

clc;
clear all;
close all;

%% =====
%   PIERDERI DATORATE PLOII
%   =====

% Constante fizice
c = physconst('lightspeed');    % Viteza luminii [m/s]

% Parametrii de intrare
R0 = ?;                          % Distanța totală între Tx și Rx [m]
freq = ?;                        % Frecvența de transmisie [Hz]
rainrate = ?;                    % Intensitatea ploii [mm/h] (ploaie puternică)
el = 0;                          % Unghi de elevație [°]
tau = 0;                         % Polarizare orizontală [°]

% Calculul pierderilor datorate ploii
rainloss = rainpl(R0, freq, rainrate, el, tau);

% Afișarea rezultatului numeric
fprintf('Atenuarea datorată ploii la %.1f GHz pe o distanță de %.1f km, cu o  
ploaie de %.1f mm/h este: %.2f dB\n', ...
        freq/1e9, R0/1e3, rainrate, rainloss);

%% =====
%   PIERDERI DE PROPAGARE DATORATE PLOII
%   =====

% Domeniul de frecvență pentru analiză
freq = (10:1:100).' * 1e9;        % De la 10 GHz la 100 GHz

% Parametrii fixați
R0 = ?;                          % Distanța
rainrate = [? ? ? ?];            % Intensități: ploaie ușoară → extremă
el = 0;                          % Unghi de elevație [°]
tau = 0;                         % Polarizare orizontală

% Inițializarea matricei de rezultate
rainloss = zeros(length(freq), length(rainrate));

% Calculul atenuării pentru fiecare rată de ploaie
for m = 1:numel(rainrate)
    rainloss(:,m) = rainpl(R0, freq, rainrate(m), el, tau)';
end

% Reprezentare grafică logaritmică
loglog(freq/1e9, rainloss);
grid on;
legend('Ploaie ușoară (1 mm/h)', 'Ploaie moderată (4 mm/h)', ...
        'Ploaie puternică (16 mm/h)', 'Ploaie extremă (50 mm/h)', ...
        'Location', 'SouthEast');
xlabel('Frecvența (GHz)');
ylabel('Atenuare datorată ploii (dB)');
title('Atenuarea semnalului în funcție de frecvență și intensitatea ploii');

```

L1 Propagare Conditii Meteo.m

```

clc;
clear all;
close all;

%% =====
%   PIERDERI DE PROPAGARE
%   =====

% Constante fizice
c = physconst('lightspeed');    % Viteza luminii [m/s]

% Condiții de mediu și parametri de simulare
T = 15;                          % Temperatura ambiantă [°C]
R = (1:20e3).';                  % Distanțe de la 1 m la 20 km
fc = ?;                          % Frecvența purtătoare [Hz]
el = 0;                          % Unghi de elevație [°]
tau = 0;                         % Polarizare orizontală [°]

%% =====
%   1. PIERDERI ÎN SPAȚIU LIBER
%   =====
apathloss = fspl(?, ?);          % Model Friis (Free Space Path Loss)

%% =====
%   2. PIERDERI DATORATE PLOII
%   =====
rr = ?;                          % Rata ploii [mm/h] - ploaie puternică
arainloss = rainpl(R, fc, rr, el, tau);

%% =====
%   3. PIERDERI DATORATE CEȚEI
%   =====
M = ?;                           % Densitatea ceții [g/m³] - ceață densă
afogloss = fogpl(R, fc, T, M);

%% =====
%   REPREZENTARE GRAFICĂ
%   =====
semilogy(R, [apathloss arainloss afogloss]);
grid on;
xlabel('Distanța de propagare (m)');
ylabel('Atenuare totală (dB)');
legend('Spațiu liber', 'Ploaie', 'Ceață', 'Location', 'SouthEast');
title('Comparația pierderilor de propagare în diverse condiții atmosferice');

```

L1 Acoperire WiFi.m

```

clc;
clear all;
close all;

%% =====
%   ANALIZA PIERDERILOR DE PROPAGARE WiFi
%   =====

% === Date de intrare ===
Pt_dBm = ?;      % Puterea emisă [dBm]
Gt_dBi = ?;      % Câștigul antenei de emisie [dBi]
Gr_dBi = ?;      % Câștigul antenei de recepție [dBi]
L_dB    = ?;      % Pierderi de sistem [dB]

% === Domeniul de distanțe ===
d = 1:32;        % Intervalul de distanțe [m] (ex. 1:1:32)

% === Frecvențele de analizat ===
f1 = ?;          % Frecvența pentru banda de 2.4 GHz
f2 = ?;          % Frecvența pentru banda de 5 GHz

% === Calculul lungimii de undă ===
c = 3e8;
lambda1 = ?;     % Lungimea de undă pentru 2.4 GHz
lambda2 = ?;     % Lungimea de undă pentru 5 GHz

% === Calculul pierderii de propagare în spațiu liber (Friis) ===
% Formula generală:
%  $Pr(dBm) = Pt(dBm) + Gt(dBi) + Gr(dBi) - L(dB) - 20 \cdot \log_{10}(4 \cdot \pi \cdot d / \lambda)$ 
Pr_2_4 = ?;      % Puterea recepționată pentru 2.4 GHz
Pr_5    = ?;      % Puterea recepționată pentru 5 GHz

%% === Reprezentare grafică ===
figure;
plot(?, ?, 'b', 'LineWidth', 1.5); hold on;
plot(?, ?, 'r', 'LineWidth', 1.5); grid on;

xlabel('Distanța (m)');
ylabel('Puterea recepționată P_r (dBm)');
legend('2.4 GHz', '5 GHz', 'Location', 'SouthWest');
title('Puterea recepționată în funcție de distanță pentru un punct de acces WiFi');

```

FIȘĂ DE LUCRU**Sarcina 1**

--

Sarcina 2

Frecvență Distanță	900 MHz	2100 MHz	2400 MHz	5000 MHz
100 m				
200 m				
400 m				
800 m				
1000 m				

Reprezentare grafică:

--

Observații:

--

Sarcina 3

Debite (mm/h)	Atenuarea suplimentară a semnalului (dB)
5 mm/h	
25 mm/h	
50 mm/h	

Reprezentare grafică:

Reprezentare grafică cu plasarea punctelor de măsură:

Sarcina 4

Script funcțional:

Reprezentare grafică:

Lucrarea 2 – Modelul cu două raze. Modelul Okumura-Hata

Lucrarea de față urmărește familiarizarea studenților cu principalele modele de propagare utilizate în rețelele celulare. Se vor analiza modelul cu două raze și modelele empirice Okumura-Hata și COST231 pentru medii construite, evidențiind modul în care acești parametri influențează atenuarea semnalului.

Obiective

- O1.Prezentarea modelului cu două raze (two-ray model).
- O2.Prezentarea modelelor empirice Okumura-Hata și COST231.
- O3.Efectuarea unor sarcini de lucru în Matlab privitoare la modelele de propagare prezentate.
- O4.Studierea efectului parametrilor de canal asupra atenuării semnalului.

Durata medie de studiu individual: 30 minute.

2.1 MODELUL CU DOUĂ RAZE

Modelul de propagare Friis ia în considerare doar calea de propagare directă dintre emițător și receptor. Pasul următor în modelarea propagării radio presupune luarea în considerare a reflexiilor semnalului cu solul. În plus față de calea de vizibilitate directă, se adaugă o singură cale reflectată în modelul two-ray, așa cum este ilustrat în Figura 2.1. Acest model ia în considerare fenomenul de reflexie de la sol și înălțimile antenelor deasupra solului. Suprafața solului este caracterizată de coeficientul de reflexie - R, care depinde de proprietățile materiale ale suprafeței și de tipul de polarizare a undelor. Antenele emițătorului și receptorului au înălțimile h_t și, respectiv, h_r și sunt separate de distanța de d metri [1].

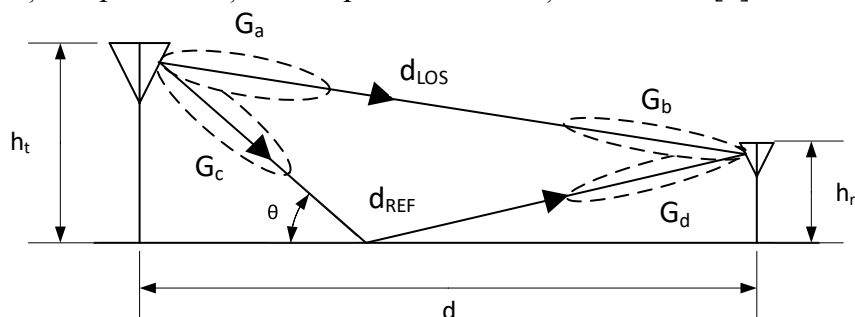


Figura 2.1 Modelul de propagare cu undă directă și undă reflectată.

Semnalul recepționat este format din două componente: unda directă și unda reflectată de la suprafața solului. Distanțele parcurse de unda directă și de cea reflectată sunt date de:

$$d_{los} = \sqrt{d^2 + (h_t - h_r)^2} \quad (2.1)$$

$$d_{ref} = \sqrt{d^2 + (h_t + h_r)^2} \quad (2.2)$$

În funcție de diferența de fază (φ) dintre unda directă și unda reflectată, semnalul recepționat poate suferi interferențe constructive sau distructive. Prin urmare, acest model este denumit și modelul de interferență cu două raze. Defazajul dintre cele două căi de propagare este:

$$\varphi = \frac{2\pi(d_{ref} - d_{los})}{\lambda} \quad (2.3)$$

unde λ este lungimea de undă a unei incidente, care poate fi dedusă din frecvența semnalului transmis. Puterea semnalului recepționat poate fi exprimată astfel:

$$P_r = P_t \left[\frac{\lambda}{4\pi} \right]^2 \left| \frac{\sqrt{G_{los}}}{d_{los}} + R \frac{\sqrt{G_{ref}} e^{-i\varphi}}{d_{ref}} \right|^2 \quad (2.4)$$

unde $\sqrt{G_{los}} = \sqrt{G_a G_b}$ este produsul câștigului antenelor de emisie/recepție pe direcția LOS, iar $\sqrt{G_{ref}} = \sqrt{G_c G_d}$ produsul câștigului antenelor de emisie/recepție pe direcția căii reflectate [2].

Așadar, după cum am văzut, semnalele nu se propagă întotdeauna pe calea directă (LOS). Unele semnale pot ajunge la destinație pe căi diferite prin reflexii și pot interfera fie în mod constructiv, fie distructiv. Acest efect de propagare multicală poate provoca fluctuații semnificative în semnalul recepționat.

2.2 PROPAGAREA ÎN ZONE LOCUITE

Propagarea undelor electromagnetice în zone aglomerate pornind de la locații de tip suburban la dens-urban reprezintă un subiect de interes pentru telefonie mobilă. Propagarea undelor electromagnetice în zonele locuite depinde de frecvență, polarizare, geometria clădirii, structura materialului clădirii, orientare, înălțime și densitate.

Modelele de propagare în care apar obstacole implică estimarea pierderilor de propagare pe terenuri neregulate cu diferite tipuri de obstacole precum copaci și clădiri. Aceste modele variază în ceea ce privește abordarea, precizia și complexitatea. Modelul Okumura-Hata este unul dintre aceste modele. Bazele modelului Okumura-Hata au fost puse în 1968 de Yoshihisa Okumura care a călătorit în jurul orașului Tokyo și a efectuat măsurători pentru a evalua atenuarea semnalului de la stația de bază la stația mobilă. El a realizat un set de grafice care determină atenuarea mediană în raport cu pierderea pe calea de propagare în spațiu liber. Okumura a obținut trei seturi de date pentru trei scenarii diferite: zonă deschisă, zonă suburbană și zonă urbană. Deoarece acest model se bazează pe studii empirice, valabilitatea parametrilor este una limitată. Valorile parametrilor din afara intervalului pot fi obținute prin extrapolarea curbelor obținute [3].

Modelul Hata este o versiune a modelului Okumura dezvoltat pentru a fi utilizat în instrumente de calcul computerizat a acoperirii cu semnal radio. Hata a dat o expresie formal-matematică modelului Okumura prin ajustarea curbelor empirice furnizate de acesta.

Pierderile de propagare (dB) sunt date de următoarea relație [3]:

$$L(\text{dB}) = [69.55 + 26.16 \log(f_c) - 13.82 \log(h_b) - a(h_m) + (44.9 - 6.55 \log h_b) \log d] + C \quad (2.5)$$

Tabelul 2.1 Parametrii modelului Okumura-Hata.

Tip de mediu	$a(h_m)$	C
Rural	$[1.1 \log f_c - 0.7] h_m - [1.56 \log f_c - 0.8]$	$-4.78 [\log(f_c)]^2 + 18.33 \log(f_c) - 40.98$
Suburban		$-2 [\log(\frac{f_c}{28})]^2 - 5.4$
Oraș mediu/mic		0
Metropolă ($f_c < 200$ MHz)	$8.29 [\log(1.54 h_m)]^2 - 1.1$	0
Metropolă ($f_c > 200$ MHz)	$3.2 [\log(11.75 h_m)]^2 - 4.92$	0

- f_c – frecvența de operare (150 MHz-1500 MHz);
- h_b – înălțimea antenei de emisie (30 m-100 m);
- h_m – înălțimea antenei de recepție (1 m-10 m);
- $a(h_m)$ – factorul de corecție al înălțimii antenei;

- C – factor de corecție pentru zonele suburbane și deschise.

Modelul COST231, numit uneori extensia PCS a modelului Okumura-Hata, este o versiune îmbunătățită a modelului Hata și este valabil pentru frecvențe între 1500 MHz și 2000 MHz.

Astfel că, intervalele de lucru pentru modelul COST231 sunt:

- Frecvență: 1500 MHz–2000 MHz
- Înălțimea efectivă a antenei emițătorului (stația de bază), h_{te} : 30 m–200 m
- Înălțimea efectivă a antenei receptorului (mobil), h_{re} : 1–10 m
- Distanța dintre antena de emisie și cea de recepție, d : 1–20 km.

Modelul COST 231 are următoarea formulă:

$$L(\text{dB}) = 46.3 + 33.9 \log(f_c) - 13.82 \log(h_b) - a(h_m) + (44.9 - 6.55 \log h_b) \log d + C \quad (2.6)$$

unde $a(h_m)$ are aceeași expresie ca în modelul Okumura-Hata, iar C este egal cu 0 dB pentru orașele de mărime medie, respectiv zonele suburbane și 3 dB pentru centrele metropolitane [3].

Sarcina 1

Corecțați și rulați script-ul [L2_Model_TwoRay.m](#). Acesta implementează ecuația (2.4) și reprezintă grafic puterea recepționată (P_{rx}) în funcție de distanța dintre emițător și receptor. Stabiliți $f=900$ MHz, $R=-1$, $h_t=50$ m, $h_r=2$ m, $G_{los}=G_{ref}=1$. Puterea transmițătorului este normalizată la 0 dBm. Graficul conține aproximări ale puterii recepționate în trei regiuni diferite, conform formulelor din Tabelul 2.2. Prin urmare, corecțați scriptul dat astfel încât acesta să corespundă cu aproximările din Tabelul 2.2. Cu d_c este notată distanța critică, care este definită ca $d_c = 4h_t h_r / \lambda$. Adăugați în [Fișa de lucru](#) script-ul funcțional și graficul obținut.

Tabelul 2.1 Expresii aproximative ale pierderilor de propagare pentru diferite distanțe dintre emițător și receptor.

	$d < h_t$	$h_t \leq d \leq d_c$	$d > d_c$
Atenuarea puterii	plat	-20 dB/decadă	-40 dB/decadă
Aproximare P_r	$\frac{P_{tx} K}{(d^2 + h_t^2)}$	$\frac{P_{tx} K}{d^2}$	$\frac{P_{tx} K}{d^4}$
Factorul K (L=pierderi de sistem)	$K = G_{los} G_{ref} \frac{\lambda^2}{(4\pi)^2 L}$	$K = G_{los} G_{ref} \frac{\lambda^2}{(4\pi)^2 L}$	$K = G_{los} G_{ref} \frac{h_t^2 h_r^2}{(4\pi)^2 L}$

Sarcina 2

Rulați fișierul [L2_TwoRay vs LOS.m](#). Luați în considerare o frecvență de lucru în banda de operare LTE 1900 MHz. Presupuneți că stația mobilă și stația de bază sunt situate la o distanță de 1 km una de cealaltă. Completați tabelul din [Fișa de lucru](#) cu atenuările semnalului pentru propagarea LOS și propagarea în cazul modelului two-ray. Comparați atenuările obținute în cazul celor două modele de propagare și notați observațiile în [Fișa de lucru](#).

Sarcina 3

- Corecțați funcția [Model_Hata.m](#) și folosindu-vă de scriptul [L2_Analiza_Model_Hata.m](#) apelați această funcție astfel încât să obțineți câte un grafic pentru trei tipuri de medii de propagare definite de către acest model (deschis, suburban și metropolitan). Stabiliți $d=1 \div 100$ km, $f_c=1500$ MHz, $h_b=70$ m și $h_m=1.5$ m.

- b) Pornind de la funcția [Model_Hata.m](#) construiți o funcție *Model_COST231.m* care să calculeze atenuarea de propagare conform modelului COST231 descris în expresia (2.6). Folosindu-vă de scriptul [L2_Comparatie_COST231_Hata.m](#) completați tabelul din [Fișa de lucru](#) cu atenuările aferente celor două modele studiate (Hata și COST231). Considerați: $h_m=3$ m, $h_b=68$ m, $f_{Okumura-Hata}=870$ MHz, $f_{COST231}=1900$ MHz, $d=3.7$ km.

Bibliografie

- [1] John S. Seybold, Ph.D., *Introduction to RF Propagation*, John Wiley & Sons, Inc., Hoboken, New Jersey, 2005, pp. 66.
- [2] Mathuranathan Viswanathan, *Wireless Communication Systems in Matlab: Second Edition*, Independently Published, ISBN 979-864-852-321-0, 2020
- [3] F. Perez Fontan, P. Marino Espinera, *Modelling the Wireless Propagation Channel*, John Wiley & Sons, Inc., 2008.

SCRIPT-URI MATLAB

L2 Model TwoRay.m

```

clear; clc; close all;
%% === Parametri inițiali ===

f = ?;          % Frecvența transmisiei [Hz]
R = ?;          % Coeficient de reflexie
Pt = ?;         % Puterea transmisă [mW] (transformați 0 dBm în mW)
Glos = ?;       % Produsul câștigului antenelor Tx/Rx pe calea directă (LOS)
Gref = ?;       % Produsul câștigului antenelor Tx/Rx pe calea reflectată
htx = ?;        % Înălțimea antenei Tx [m]
hrx = ?;        % Înălțimea antenei Rx [m]
d = 1:0.1:1e5; % Distanța Tx-Rx [m]
L = 1;          % Pierderi suplimentare (L=1 => fără pierderi)

%% === Modelul two-ray ===
d_los = sqrt(d.^2+(?-?)^2); % Distanța pe calea directă LOS
d_ref = sqrt((? + ?)^2 + ?.^2); % Distanța pe calea reflectată
lambda = 3e8 / f;          % Lungimea de undă [m]
phi = 2*pi*(?-d_los)/?;    % Diferența de fază dintre cele două căi
s = lambda / (4 * pi) * ( sqrt(?) ./ d_los + R * sqrt(?) ./ d_ref .* exp(1i *
?) );

Pr = Pt * abs(?).^2; % Puterea recepționată
Pr_norm = Pr / Pr(1); % Normalizare la valoarea inițială

Pr_LOS = Pt*?(?./(4*pi*?).^2); % Puterea recepționată pentru cazul LOS

%% === Reprezentare grafică ===
semilogx(d, 10 * log10(Pr)); hold on; ylim([-160 -55]);
semilogx(d, 10*log10(Pr_LOS));
title('Two-Ray Ground Reflection Model');
xlabel('log_{10}(d) [m]');
ylabel('Received Power [dB]');
legend('Two-ray model', 'Line-of-Sight');
%% === Modele aproximative în 3 regiuni ===
dc = 4*htx*hrx/lambda; % Distanța critică
d1 = 1:0.1:htx; % Regiunea 1: d <= htx
d2 = htx:0.1:dc; % Regiunea 2: htx <= d <= dc
d3 = dc:0.1:1e5; % Regiunea 3: d >= dc

K_1 = Glos * ? * ?^2 / ((4 * pi)^2 * L);
K_2 = ? * Gref * ?^2 * hrx^2 / ((4 * pi)^2 * L);

Pr1 = Pt * K_1 ./ (d1.^2 + htx^2); % Putere regiunea 1
Pr2 = Pt * K_1 ./ d2.^2;          % Putere regiunea 2
Pr3 = Pt * K_2 ./ d3.^4;          % Putere regiunea 3

semilogx(d1, 10 * log10(Pr1), 'k-.'); % Regiunea 1
semilogx(d2, 10 * log10(Pr2), 'r-.'); % Regiunea 2
semilogx(d3, 10 * log10(Pr3), 'g-.'); % Regiunea 3

%% === Marcaje pe grafic (htx și dc) ===
h = line([htx htx], [-160 -55]); set(h, 'Color', 'm');
h = line([dc dc], [-160 -55]); set(h, 'Color', 'm');

```

L2 TwoRay vs LOS.m

```

clear; clc;

%% === Constante fizice ===
c0 = 3e8;           % Viteza luminii [m/s]
Gamma = -1;         % Coeficient de reflexie
d = 1000;           % Distanța fixă între Tx și Rx [m]
Pt_mW = 100;        % Puterea de emisie [mW]

disp('Introduceți frecvența și înălțimile antenelor pentru fiecare caz.')
disp('Notați valorile pierderilor de propagare în tabelul din îndrumător.')
fc = input('Introduceți frecvența [Hz]: ');
lambda = c0 / fc; % Lungimea de undă
keepGoing = true;

while keepGoing
    %% === Introducere date ===
    h_tx_m = input('Introduceți înălțimea antenei Tx [m]: ');
    h_rx_m = input('Introduceți înălțimea antenei Rx [m]: ');

    %% === Pierdere de propagare LOS (Friis) ===
    Pr_LOS = Pt_mW * (lambda/(4*pi*d))^2; % Putere recepționată
    PL_LOS_dB = -10*log10(Pr_LOS/Pt_mW); % Pierderi de propagare în dB

    %% === Pierdere de propagare Two-Ray ===
    d_LOS = sqrt((h_tx_m - h_rx_m)^2 + d^2); % Distanța pe calea directă
    d_REF = sqrt((h_tx_m + h_rx_m)^2 + d^2); % Distanța pe calea reflectată
    phi = 2*pi*(d_REF - d_LOS) / lambda; % Diferența de fază

    E_LOS = 1 / d_LOS;
    E_REF = Gamma / d_REF * exp(1i*phi);
    E_tot = (lambda/(4*pi)) * (E_LOS + E_REF);

    Pr_2ray = Pt_mW * abs(E_tot)^2; % Putere recepționată
    PL_2ray_dB = -10*log10(Pr_2ray/Pt_mW); % Pierderi de propagare în dB

    %% === Afișare rezultate ===
    fprintf('\n=== Rezultate ===\n');
    fprintf('Distanță: %.2f m | Frecvență: %.2f MHz\n', d, fc/1e6);
    fprintf('hTx: %.2f m | hRx: %.2f m\n', h_tx_m, h_rx_m);
    fprintf('Pierdere de propagare LOS: %.2f dB\n', PL_LOS_dB);
    fprintf('Pierdere de propagare Two-Ray: %.2f dB\n', PL_2ray_dB);
    fprintf('-----\n');
    fprintf('Notați aceste valori în foaia de laborator.\n\n');

    %% === Întrebare pentru continuare ===
    keepGoing = strcmpi(input('Doriți să calculați pentru alt set de
    înălțimi/frecvențe? (y/n): ', 's'), 'y');
end

disp('Sesiunea s-a încheiat. Verificați rezultatele notate.')

```

Model Hata.m

```

function PL = Model_Hata(fc,d,hb,hm,envType)

envType = lower(envType);
switch envType
    case 'metro'
        C = 0;
        if fc <= 200
            aHm = 9.29*(log10(1.54*?))^2 - 1.1;
        else
            aHm = 3.2*(log10(11.75*hm))^2 - 4.92;
        end
    case 'smallcity'
        C = 0;
        aHm = (1.1*log10(?) - 0.7)*? - (1.56*log10(fc) - 0.8);
    case 'suburban'
        aHm = (1.1*log10(fc) - 0.7)*? - (1.56*log10(fc) - 0.8);
        C = -2*(log10(fc/28)).^2 - 5.4;
    case 'open'
        aHm = (1.1*log10(fc) - 0.7)*hm - (1.56*log10(fc) - 0.8);
        C = -4.78*(log10(?)).^2 + 18.33*log10(fc) - 40.98;
    otherwise
        error('Tip de model invalid');
end

A = 69.55 + 26.16*log10(?) - 13.82*log10(?) - aHm;
B = 44.9 - 6.55*log10(?);
PL = ? + B*log10(?) + ?;
end

```

L2 Analiza Model Hata.m

```

%----- Parametri de intrare -----%
fc = ?;          % Frecvența purtătoare [MHz]
hb = ?;          % Înălțimea antenei stației de bază [m]
hm = ?;          % Înălțimea antenei terminalului mobil [m]
d = 0:0.001:1;   % Distanța dintre Tx și Rx [km]

%----- Calculul pierderilor -----%
PL_open = Model_Hata(?, ?, ?, ?, 'open');      % Mediu deschis
PL_suburban = Model_Hata(?, ?, ?, ?, 'suburban'); % Mediu suburban
PL_urban = Model_Hata(?, ?, ?, ?, 'metro');     % Mediu metropolitan

%----- Reprezentare grafică -----%
figure;
plot(d, PL_open, 'b', 'LineWidth', 1.5); hold on;
plot(d, PL_suburban, 'g', 'LineWidth', 1.5);
plot(d, PL_urban, 'r', 'LineWidth', 1.5);
grid on;

title('Compararea pierderilor de propagare pentru diferite medii (Modelul Hata)');
xlabel('Distanța [km]');
ylabel('Pierdere de propagare [dB]');
legend('Mediu deschis', 'Mediu suburban', 'Mediu metropolitan', 'Location',
    ..

```

L2 Comparatie COST231 Hata.m

```
clearvars; clc;
% Script pentru compararea atenuărilor calculate cu modelele Hata și COST231

%----- Parametri de intrare -----%
h_MS = ?;           % Înălțimea antenei terminalului mobil [m]
h_BS = ?;           % Înălțimea antenei stației de bază [m]
f_Hata = ?;         % Frecvența pentru modelul Okumura-Hata [MHz]
f_COST = ?;         % Frecvența pentru modelul COST231 [MHz]
d = ?;             % Distanța dintre Tx și Rx [km]

%----- Calculul pierderilor -----%
% Model Okumura-Hata pentru mediu urban
PL_Hata_urban = Model_Hata (?, ?, ?, ?, 'open');

% Model COST231-Hata pentru mediu urban
PL_COST231_urban = Model_COST231 (?, ?, ?, ?, 'open');

%----- Afișarea rezultatelor -----%
fprintf('Rezultatele simulării:\n');
fprintf('-----\n');
fprintf('Modelul Okumura-Hata (f = %.1f MHz): %.2f dB\n', f_Hata,
PL_Hata_urban);
fprintf('Modelul COST231 (f = %.1f MHz): %.2f dB\n', f_COST,
PL_COST231_urban);
fprintf('-----\n');
```

FIȘĂ DE LUCRU

Sarcina 1

i) Adăugați aici scriptul funcțional:

ii) Reprezentare grafică:

Sarcina 2

Înălțimea antenei TX	Înălțimea antenei RX	Atenuarea în cazul propagării two-ray	Atenuarea în cazul propagării LOS
100 m	1.6 m		
100 m	2 m		
100 m	2.3 m		
120 m	2.3 m		
140 m	2.3 m		

Observații:

Sarcina 3

a) Reprezentare grafică

b) Completați tabelul de mai jos:

	Okumura-Hata	COST231
Oraș de dimensiuni mici/ medii		
Oraș de dimensiuni mari		
Zonă suburbană		
Zonă deschisă		

Lucrarea 3 – Difrakția. Modelul log-normal. Zona Fresnel

Lucrarea de față urmărește atât consolidarea cunoștințelor teoretice privind efectele de umbrire (shadowing), difracție și zona Fresnel, cât și aplicarea modelului log-normal pentru evaluarea atenuărilor de semnal în medii reale de propagare, cu ajutorul simulărilor efectuate în Matlab.

Obiective

- O1.Prezentarea conceptului de umbrire în cazul propagării semnalelor RF.
- O2.Prezentarea modelului log-normal de calcul al atenuărilor de propagare.
- O3.Prezentarea unor noțiuni legate de difracție. Modelul knife-edge.
- O4.Prezentarea zonelor Fresnel.
- O5.Efectuarea unor sarcini de lucru în Matlab privitoare la simularea modelelor matematice ale fenomenelor sus-amintite.

Durata medie de studiu individual: 30 minute.

3.1 FENOMENUL DE UMBRIRE

Pierderile de propagare apar nu doar datorită distanței dintre antena de emisie și cea de recepție, ci și datorită diferitelor obstacole de pe traseul de propagare.

Dacă anumite obiecte obstruzionează traseul de propagare al undei (de exemplu, copaci, clădiri etc.), o parte din semnalul transmis va fi pierdut prin absorbții, reflexii, difracții sau datorită fenomenului de dispersie a undei.

Difrakția reprezintă fenomenul fizic prin care o undă electromagnetică se poate propaga peste sau în jurul unor obstacole care blochează vizibilitatea directă dintre un emițător și un receptor. Efectul difracției este acela de a „umple” regiunile de umbră, astfel încât o anumită cantitate de energie electromagnetică ajunge și în zonele aflate în spatele obstacolului.

Cea mai intuitivă modalitate de a înțelege acest fenomen este prin principiul lui Huygens-Fresnel, care afirmă că fiecare punct al unui front de undă se comportă ca o sursă de „unde secundare”, iar suprapunerea acestora generează un nou front de undă în direcția de propagare, după cum se poate observa și în Figura 3.1. În acest fel ne putem explica de ce întâlnim o energie nenulă măsurată la umbra unui obstacol [1]. Un exemplu interesant legat de fenomenul de umbrire apare atunci când sursa semnalului se află dincolo de orizontul terestru. Deși linia de vizibilitate directă este blocată, o mică parte din energia RF este difractată de suprafața Pământului și poate ajunge astfel la receptor.

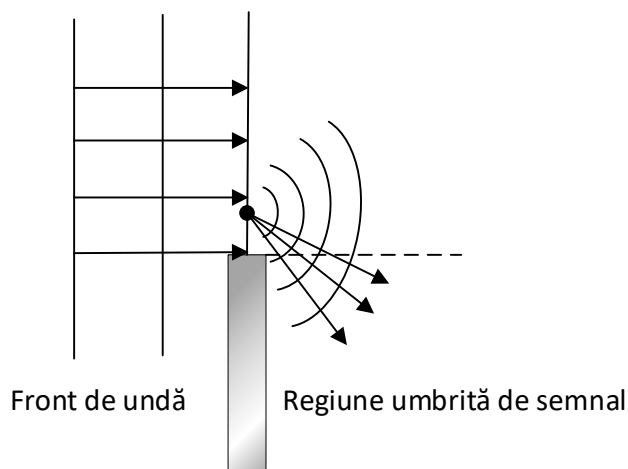


Figura 3.1 Principiul Huygens-Fresnel.

Difracția apare indiferent dacă obstacolul este conductor sau neconductor, cu condiția ca materialul neconductor să nu permită trecerea energiei RF prin el (adică să nu fie transparent pentru undele radio). Altfel spus, referindu-ne la un exemplu mai concret, dacă antena stației de bază ar fi considerată o sursă de lumină, clădirea reprezentată în mijlocul figurii ar acționa ca o umbră asupra antenei aflată la stația mobilă (Figura 3.2), de aici și numele acestui fenomen de umbră a semnalului.

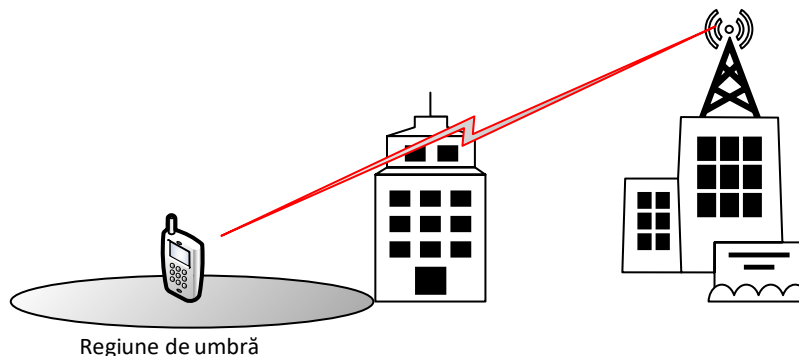


Figura 3.2 Fenomenul de umbră a semnalului.

3.2 MODELUL LOG-NORMAL DE PROPAGARE

Primul pas în estimarea pierderilor pe calea de propagare este determinarea pierderii mediane. Mai multe metode pentru estimarea acestora au fost prezentate în lucrările anterioare. Al doilea pas constă în determinarea atenuării pe scară largă (large-scale fading) față de valoarea medie a pierderii pe calea de propagare. Această variație pe scară largă este atribuită, de obicei, difracției, fenomen cunoscut și sub numele de umbră (shadowing) [2].

Atunci când semnalul este obstrucționat de către un anumit obstacol, acesta poate ajunge la receptor pe mai multe căi de difracție sau reflexie. Dacă numărul de puncte de difracție sau de reflexie este suficient de mare, se pot modela pierderile pe calea de propagare printr-o variabilă aleatoare Gaussiană.

Această abordare este utilizată frecvent în sistemele de telefonie mobilă, unde fenomenul este cunoscut sub denumirea de fading log-normal, deoarece atenuarea urmează o distribuție normală în domeniul logaritmic. În acest caz, valoarea medie a atenuărilor de propagare este egală cu valoarea mediană. Prin urmare, pentru caracterizarea completă a fadingului log-normal, este necesară doar varianța (dispersia).

Modelul log-normal de pierderi pe calea de propagare este o extensie a modelului de pierderi teoretizat de ecuația lui Friis. Acest model este folosit pentru a estima pierderile de propagare pentru o gamă largă de medii, în timp ce modelul lui Friis este limitat doar la propagarea în spațiu liber, fără obstacole între emițător și receptor [2].

Modelul log-normal include în estimările sale și efectele aleatoare cauzate de fenomenul de umbră a semnalului. În regiunea de câmp îndepărtat, dacă $P_L(d_0)$ reprezintă pierderea de propagare pentru distanța de referință d_0 față de emițător, atunci pierderea de propagare la o distanță arbitrară $d > d_0$ este dată de expresia:

$$[P_L(d)]_{ab} = [P_L(d_0)]_{ab} + 10n \log_{10} \left(\frac{d}{d_0} \right) + \chi, d_f \leq d_0 \leq d \quad (3.1)$$

unde, $P_L(d)$ reprezintă pierderea de propagare la distanța d , iar n este exponentul pierderilor de propagare. Câteva valori de referință pentru acest exponent n sunt date în Tabelul 3.1 pentru diferite medii de propagare [3].

În continuare, χ este o variabilă ce urmărește o distribuție normală de medie $\mu=0$ și deviație standard σ , exprimată în dB, folosită doar atunci când se iau în calcul pierderile de propagare și efectul de umbră.

Tabelul 3.1 Exponentul de pierderi pe calea de propagare pentru diferite medii.

Mediu	Exponent al pierderilor pe calea de propagare (n)
Spațiu liber	2
Celulă radio în mediu urban	2.7÷3.5
Celulă radio în mediu urban cu umbrire	3÷5
Interior de clădire – propagare în spațiu liber	1.6÷1.8
Propagare cu obstacole în interiorul clădirilor	4÷6
Propagare cu obstacole în interiorul unei fabrici	2÷3

Pierdere pe calea de propagare la distanța d_0 față de emițător, $P_L(d_0)$, este obținută din ecuația lui Friis generalizată pentru diferite medii de propagare. Expresia matematică este redată mai jos. În mod tipic, $d_0 = 1 \div 10 \text{ m}$ pentru micro-celule și $d_0 = 1 \text{ km}$ pentru celule de dimensiuni mari.

$$[P_L(d_0)]_{dB} = -10 \log \frac{\lambda^2}{(4\pi)^2 d_0^n} \quad (3.2)$$

3.3 ZONA FRESNEL

O considerație importantă pentru modelele de propagare este existența obstacolelor în prima zonă Fresnel.

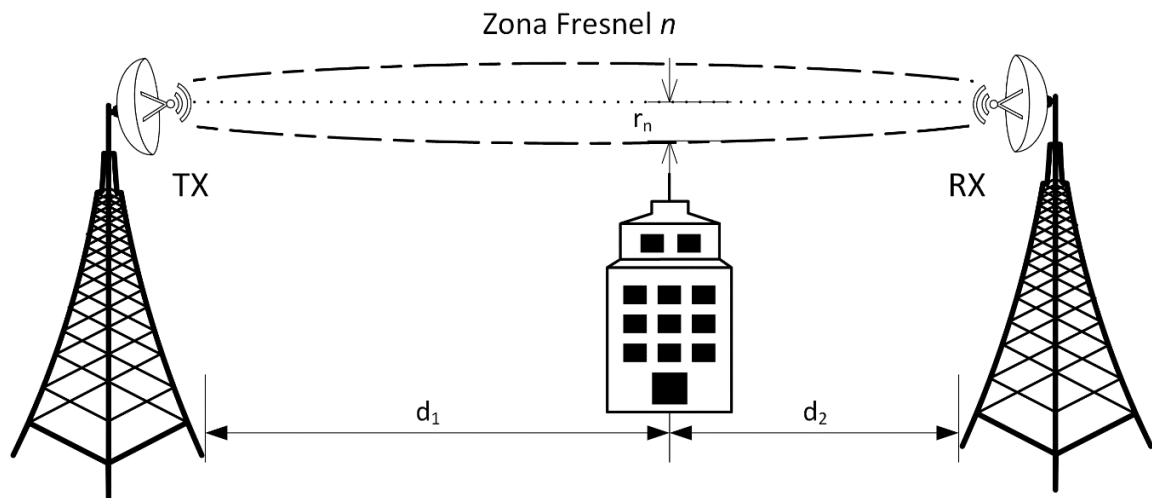


Figura 3.3 Zona Fresnel.

Zona Fresnel sunt elipsoide care au drept focare emițătorul și receptorul (Figura 3.3), iar distanța dintre calea de propagare directă și căile alternative de propagare sunt multipli de $\lambda/2$. Undele electromagnetice emise din zona Fresnel cu număr impar provoacă interferențe distructive, iar undele electromagnetice emise din zona Fresnel cu număr par provoacă interferențe constructive [3].

Pentru geometrii de propagare la scală largă, diferența de drum dintre calea directă (LOS) și undele secundare este:

$$\Delta = \frac{h^2}{2} \left(\frac{1}{d_1} + \frac{1}{d_2} \right) \quad (3.3)$$

Raza celei de-a n -a zonă Fresnel poate fi aproximată ca:

$$r_n = \sqrt{\frac{n\lambda d_1 d_2}{(d_1 + d_2)}} \quad (3.4)$$

Data fiind înălțimea obstacolului h , putem determina indicele n al zonei Fresnel care este afectată de prezența acestui obstacol. Stabilind $r_n = h$ în ultima relație, obținem:

$$n = \frac{2\Delta}{\lambda} = \frac{h^2}{\lambda} \left(\frac{1}{d_1} + \frac{1}{d_2} \right) \quad (3.5)$$

Ca regulă generală pentru comunicațiile punct-la-punct, dacă cel puțin 60% din prima zonă Fresnel nu conține obstacole, pierderile datorate difracției sunt neglijabile. Este recomandat ca procentajul să ajungă la aproximativ 80%. Orice degajare suplimentară a zonei Fresnel nu va modifica semnificativ pierderile datorate difracției.

3.4 MODELAREA PIERDERILOR DE PROPAGARE DATORATE DIFRAȚIEI

În mediul de propagare radio pot apărea obstacole care să împiedice transmisia radio dintre emițător și receptor. În acest sens, sunt disponibile modele pentru estimarea pierderilor de semnal asociate cu fenomenul de difracție provocat de către astfel de obstacole.

Modelul prezentat în Figura 3.4 ia în considerare două cazuri idealizate în care un obstacol ascuțit este plasat între emițător și receptor. Folosind toți parametrii geometrici indicați în figură, pierderile de propagare provocate de difracție pot fi estimate cu ajutorul unei mărimi adimensionale, v , numită parametru de difracție Fresnel-Kirchhoff.

Pe baza informațiilor disponibile, oricare dintre următoarele ecuații poate fi utilizată pentru a calcula acest parametru [3].

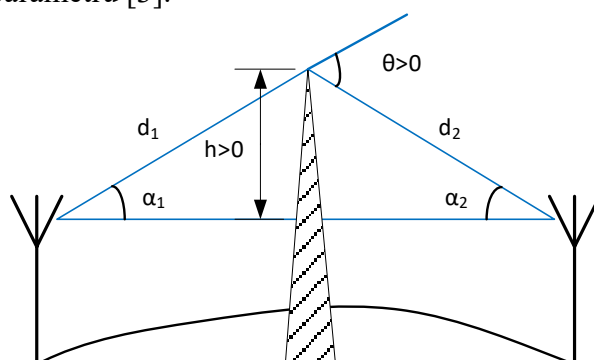


Figura 3.4 Difracția după modelul cu obstacole de tip knife-edge pentru $h > 0$.

$$v = h \sqrt{\frac{\lambda}{2} \left(\frac{1}{d_1} + \frac{1}{d_2} \right)} = \theta \cdot \sqrt{\frac{2}{\lambda \left(\frac{1}{d_1} + \frac{1}{d_2} \right)}} = \sqrt{\frac{2h\theta}{\lambda}} = \sqrt{\frac{2d}{\lambda} \alpha_1 \alpha_2} \quad (3.6)$$

După calcularea parametrului de difracție Fresnel-Kirchhoff, nivelul semnalului obținut în urma difracției knife-edge poate fi determinat prin metode numerice complexe. Totuși, pentru $v > -0.7$, poate avea loc următoarea aproximație a pierderilor datorate difracției:

$$G(v) = 6.9 + 20 \log \left(\sqrt{(v - 0.1)^2 + 1} + v - 0.1 \right) \text{ dB} \quad (3.7)$$

Sarcina 1

Folosind funcțiile: [Model_LogNormal.m](#) și [Model_Friis.m](#) ce implementează modelul de umbrire log-normal, rulați scriptul [L3_Test_Model_LogNormal.m](#), reprezentați grafic

puterea semnalului recepționat atât pentru cazul în care luăm în considerare efectul de umbră, cât și pentru cazul în care acesta este omis. Generați și comparați seturi de câte 2 grafice per figură utilizând următorii parametri:

- $f_1=2.4\text{GHz}$, $f_2=5.8\text{GHz}$ iar ca valori comune pentru parametri de simulare: $n=2$, $d_0=1\text{m}$, $\sigma=2\text{dB}$, $P_t=0\text{ dBm}$, $G_t=G_r=1\text{ dBi}$.
- n_1 pentru interior de clădire fără obstacole, n_2 pentru interior de clădire cu obstacole, iar ca valori comune pentru parametri de simulare: $f=2.4\text{GHz}$, $d_0=1\text{m}$, $\sigma=2\text{dB}$, $P_t=0\text{ dBm}$, $G_t=G_r=1\text{ dBi}$.
- $\sigma_1=1\text{dB}$, $\sigma_2=4\text{dB}$, $f=2.4\text{GHz}$, iar ca valori comune pentru parametri de simulare: $n=2$, $d_0=1\text{m}$, $P_t=0\text{ dBm}$, $G_t=G_r=1\text{ dBi}$.

Sarcina 2

Corectați și rulați scriptul [L3_Test_ZonaFresnel.m](#) și, respectiv, funcția pe care acesta o apelează [ZonaFresnel.m](#). Măsurați raza primei zone Fresnel, pentru o comunicație punct-la-punct între un emițător și un receptor care sunt situați la o distanță de $d = 25\text{ km}$ și care comunică pe frecvența $f = 12\text{ GHz}$. Care este raza minimă pentru o legătură radio în care pierderile datorate difracției să poată fi neglijate (degajare 60%)? Repetați experimentul pentru $f = 2\text{GHz}$ și $d = 25\text{ km}$. Treceți rezultatele în [Fișa de lucru](#).

Sarcina 3

Determinați atenuarea datorată difracției în cazul unei comunicații punct la punct care operează pe frecvența $f = 10\text{ GHz}$ și pe o distanță de 1 km . Între emițător și receptor se află un deal abrupt situat la 300 m de emițător și 700 m de receptor și care se ridică la 3 m deasupra liniei de vizibilitate directă dintre cele două sisteme. De asemenea, determinați zona Fresnel (n) în care se situează obstacolul dat. Ce parametri trebuie modificați și în ce fel pentru a aduce obstacolul în prima zonă Fresnel?

Folositi-vă de scriptul [L3_Test_SingleKnifeEdge.m](#) și de funcția [Model_SingleKnifeEdge.m](#) pe care va trebui să le corectați mai întâi. Treceți rezultatele în [Fișa de lucru](#).

Sarcina 4

Folosindu-vă de site-urile [radiofresnel.com](#) poziționați un emițător pe clădirea de Laboratoare UTCN de pe str. Observatorului (Longitudine 23.596524, Latitudine 46.757131, înălțime 455 m) și un receptor pe clădirea UTCN de pe str. Barițiu 26-28 (Longitudine 23.585695, Latitudine 46.772501, înălțime 365 m). Generați un fișier *.kml și importați-l într-un proiect nou pe [earth.google.com](#). Vizualizați elipsoidul Fresnel obținut. Reglați înălțimea antenelor de emisie și recepție astfel încât să vă asigurați că aveți peste 80% degajare pentru zona Fresnel 1. Adăugați în [Fișa de lucru](#) capturi de ecran pentru scenariile de lucru la frecvențele de 90 MHz și 2500 MHz pentru zona Fresnel 1 și, respectiv 2.

Bibliografie

- [1] F. Perez Fontan, P. Marino Espineira, *Modeling the wireless propagation channel: a simulation approach with Matlab*, John Wiley & Sons Ltd, , ISBN 978-0-470-72785-0
- [2] Theodore Rapaport, *Wireless Communications. Principles and Practice*, Prentice Hall, 2002.
- [3] Mathuranathan Viswanathan, *Wireless Communication Systems in Matlab: Second Edition*, Independently Published, ISBN 979-864-852-321-0, 2020.

SCRIPT-URI MATLAB

L3 Test Model LogNormal.m

```

clear; close all; clc;

%% === Parametri comuni =====
P_tx_dBm = 0;      % Puterea emisă [dBm]
G_tx_dBi = 1;      % Câștig antenă TX [dBi]
G_rx_dBi = 1;      % Câștig antenă RX [dBi]
d_ref = 1;         % Distanța de referință [m]
L_sys = 1;         % Pierderi de sistem
d_vec = 100*(1:0.2:100); % Vectorul distanțelor simulate [m]

%% =====
% (a) f1=2.4GHz, f2=5.8GHz; n=2; σ=2dB
% =====
f1 = 2.4e9; f2 = 5.8e9; n = 2; sigma_dB = 2;

[P_rx_shadow_f1, ~] = Model_LogNormal (P_tx_dBm, G_tx_dBi, G_rx_dBi, f1,
d_ref, d_vec, L_sys, sigma_dB, n);
[P_rx_friis_f1, ~] = Model_Friis (P_tx_dBm, G_tx_dBi, G_rx_dBi, f1, d_vec,
L_sys, n);

[P_rx_shadow_f2, ~] = Model_LogNormal (P_tx_dBm, G_tx_dBi, G_rx_dBi, f2,
d_ref, d_vec, L_sys, sigma_dB, n);
[P_rx_friis_f2, ~] = Model_Friis (P_tx_dBm, G_tx_dBi, G_rx_dBi, f2, d_vec,
L_sys, n);

figure;
plot(d_vec, P_rx_friis_f1, '--b', 'LineWidth', 1.2); hold on;
plot(d_vec, P_rx_shadow_f1, 'b', 'LineWidth', 1.5);
plot(d_vec, P_rx_friis_f2, '--r', 'LineWidth', 1.2);
plot(d_vec, P_rx_shadow_f2, 'r', 'LineWidth', 1.5);
grid on;
xlabel('Distanța [m]');
ylabel('P_r [dBm]');
title('(a) Comparație pentru f1=2.4GHz și f2=5.8GHz, n=2, \sigma=2dB');
legend('Friis 2.4GHz', 'Log-Normal 2.4GHz', ...
'Friis 5.8GHz', 'Log-Normal 5.8GHz', ...
'Location', 'southwest');

%% =====
% (b) n1 și n2 diferite; f=2.4GHz; σ=2dB
% =====
f = 2.4e9; sigma_dB = 2;
n1 = 1.6; % interior fără obstacole
n2 = 3.0; % interior cu obstacole

[P_rx_shadow_n1, ~] = Model_LogNormal (P_tx_dBm, G_tx_dBi, G_rx_dBi, f, d_ref,
d_vec, L_sys, sigma_dB, n1);
[P_rx_friis_n1, ~] = Model_Friis (P_tx_dBm, G_tx_dBi, G_rx_dBi, f, d_vec,
L_sys, n1);

[P_rx_shadow_n2, ~] = Model_LogNormal(P_tx_dBm, G_tx_dBi, G_rx_dBi, f, d_ref,
d_vec, L_sys, sigma_dB, n2);
[P_rx_friis_n2, ~] = Model_Friis (P_tx_dBm, G_tx_dBi, G_rx_dBi, f, d_vec,
L_sys, n2);

```

```

figure;
plot(d_vec, P_rx_friis_n1, '--b', 'LineWidth', 1.2); hold on;
plot(d_vec, P_rx_shadow_n1, 'b', 'LineWidth', 1.5);
plot(d_vec, P_rx_friis_n2, '--r', 'LineWidth', 1.2);
plot(d_vec, P_rx_shadow_n2, 'r', 'LineWidth', 1.5);
grid on;
xlabel('Distanța [m]');
ylabel('P_r [dBm]');
title('(b) Comparație pentru n1=1.6 și n2=3.0, f=2.4GHz, \sigma=2dB');
legend('Friis n1', 'Log-Normal n1', ...
       'Friis n2', 'Log-Normal n2', ...
       'Location', 'southwest');

%% =====
% (c)  $\sigma_1=1\text{dB}$ ,  $\sigma_2=4\text{dB}$ ;  $f_1=2.4\text{GHz}$ ,  $f_2=5.8\text{GHz}$ ;  $n=2$ 
% =====
sigma1 = 1; sigma2 = 4; n = 2;
f1 = 2.4e9; f2 = 5.8e9;

[P_rx_shadow_s1, ~] = Model_LogNormal (P_tx_dBm, G_tx_dBi, G_rx_dBi, f1,
d_ref, d_vec, L_sys, sigma1, n);
[P_rx_friis_s1, ~] = Model_Friis (P_tx_dBm, G_tx_dBi, G_rx_dBi, f1, d_vec,
L_sys, n);

[P_rx_shadow_s2, ~] = Model_LogNormal (P_tx_dBm, G_tx_dBi, G_rx_dBi, f2,
d_ref, d_vec, L_sys, sigma2, n);
[P_rx_friis_s2, ~] = Model_Friis (P_tx_dBm, G_tx_dBi, G_rx_dBi, f2, d_vec,
L_sys, n);

figure;
plot(d_vec, P_rx_friis_s1, '--b', 'LineWidth', 1.2); hold on;
plot(d_vec, P_rx_shadow_s1, 'b', 'LineWidth', 1.5);
plot(d_vec, P_rx_friis_s2, '--r', 'LineWidth', 1.2);
plot(d_vec, P_rx_shadow_s2, 'r', 'LineWidth', 1.5);
grid on;
xlabel('Distanța [m]');
ylabel('P_r [dBm]');
title('(c) Comparație pentru  $\sigma_1=1\text{dB}$ ,  $\sigma_2=4\text{dB}$ ,  $f_1=2.4\text{GHz}$ ,  $f_2=5.8\text{GHz}$ ,  $n=2$ ');
legend('Friis  $\sigma_1, f_1$ ', 'Log-Normal  $\sigma_1, f_1$ ', ...
       'Friis  $\sigma_2, f_2$ ', 'Log-Normal  $\sigma_2, f_2$ ', ...
       'Location', 'southwest');

```

Model_LogNormal.m

```

function [P_rx_dBm, L_path_dB] = Model_LogNormal (P_tx_dBm, G_tx_dBi,
G_rx_dBi, f0, d0, d, L_sys, sigma_dB, n)
% P_tx_dBm - Puterea emisă [dBm]
% G_tx_dBi - Câștigul antenei de emisie [dBi]
% G_rx_dBi - Câștigul antenei de recepție [dBi]
% f0 - Frecvența purtătoare [Hz]
% d_ref - Distanța de referință [m]
% d_vec - Vectorul distanțelor la care se calculează pierderea [m]
% L_sys - Pierderi adiționale de sistem (L=1 pentru lipsă pierderi)
% sigma_dB - Deviația standard a umbririi log-normale [dB]
% n - Exponentul pierderilor pe calea de propagare
%
% L_path_dB - Pierderea totală de propagare (cu umbrire log-normală) [dB]
% P_rx_dBm - Puterea recepționată la distanțele d_vec [dBm]

% Calculul lungimii de undă
lambda = 3e8 / f0;

% Calculul factorului de pierdere pe calea de propagare
K_dB = 20*log10(lambda/(4*pi)) - 10*n*log10(d/d0) - 10*log10(L_sys);

% Termen aleator pentru umbrire log-normală
X_sigma = sigma_dB * randn(1, numel(d));

% Pierderea totală de propagare (inclusiv câștigurile antenelor)
L_path_dB = P_tx_dBm + G_tx_dBi - G_rx_dBi + K_dB - X_sigma;

% Puterea recepționată
P_rx_dBm = P_tx_dBm + L_path_dB;
end

```

Model_Friis.m

```

function [P_rx_dBm, L_path_dB] = Model_Friis(P_tx_dBm, G_tx_dBi, G_rx_dBi, f,
d, L_sys, n)
% P_tx_dBm - Puterea emisă [dBm]
% G_tx_dBi - Câștigul antenei de emisie [dBi]
% G_rx_dBi - Câștigul antenei de recepție [dBi]
% f0 - Frecvența purtătoare [Hz]
% d_vec - Vectorul distanțelor pentru care se calculează pierderea [m]
% L_sys - Pierderi de sistem (L=1 pentru cazul ideal)
% n - Exponentul pierderilor pe calea de propagare
%
% P_rx_dBm - Puterea recepționată [dBm]
% L_path_dB - Pierderea de propagare [dB] (inclusiv câștigurile antenelor)

% Lungimea de undă
lambda = 3e8 / f;

% Pierderea de propagare [dB] (modelul Friis generalizat)
L_path_dB = P_tx_dBm + G_tx_dBi - G_rx_dBi + 20*log10(lambda/(4*pi)) - 10*n*log10(d/d0) - 10*log10(L_sys);

% Puterea recepționată
P_rx_dBm = P_tx_dBm + L_path_dB;
end

```

L3 Test Zona Fresnel.m

```

clear; close all; clc;

d = ?;          % [m] distanța totală între emițător (Tx) și receptor (Rx)
f0 = ?;         % [Hz] frecvența semnalului transmis
n = ?;          % numărul zonei Fresnel (influențează doar r_n)

% Punctul de măsurare se află la mijlocul distanței
d1 = d / 2;
d2 = d / 2;

% r_n      = raza pentru zona Fresnel n
% r_clear  = degajarea necesară pentru prima zonă Fresnel (~60% din r_1)
[r_n, r_clear] = ZonaFresnel(d1, d2, f0, n)

```

ZonaFresnel.m

```

function [r_n, r_clear] = ZonaFresnel(d1, d2, f0, n)
% INPUT:
%   d1 - [m] distanța de la emițător până la punctul de măsurare
%   d2 - [m] distanța de la receptor până la punctul de măsurare
%   f0 - [Hz] frecvența undei transmise
%   n   - numărul zonei Fresnel
%
% OUTPUT:
%   r_n      - [m] raza celei de-a n-a zone Fresnel la punctul de măsurare
%   r_clear  - [m] degajarea necesară pentru prima zonă (≈ 60% din r_1)
%=====

c0 = 3e8;          % viteza luminii [m/s]
lambda = c0 / ?;    % lungimea de undă [m]

% === Raza celei de-a n-a zone Fresnel ===
r_n = sqrt(n * lambda * ? * ? ./ (d1 + d2));

% === Degajarea necesară în prima zonă (60% din r_1) ===
r_clear = ? * sqrt(1 * ? * d1 * d2 ./ (d1 + d2));

end

```

L3 Test SingleKnifeEdge.m

```

%% == PARAMETRI DE INTRARE ==
h_obstacol = ?; % [m] Înălțimea obstacolului peste linia de vizibilitate
frecventa = ?; % Frecvența purtătoare (10 GHz) [Hz]
dist_tx = ?; % Distanța de la emițător la obstacol [m]
dist_rx = ?; % Distanța de la obstacol la receptor [m]
d_total = dist_tx + dist_rx; % Distanța totală între emițător și receptor [m]

%% == CALCUL ==
[atenuare_dB, zona] = Model_SingleKnifeEdge(?, ?, dist_tx, dist_rx);

%% == AFIȘARE REZULTATE ==
fprintf('--- Rezultate ---\n');
fprintf('Frecvență: %.2f GHz\n', frecventa/1e9);
fprintf('Distanță totală: %.0f m (Tx-Obstacol: %.0f m, Obstacol-Rx: %.0f m)\n', d_total, dist_tx, dist_rx);
fprintf('Înălțime obstacol: %.2f m peste linia de vizibilitate\n', h_obstacol);
fprintf('Atenuare prin difracție: %.2f dB\n', atenuare_dB);
fprintf('Obstacolul se află în zona Fresnel: %.1f\n', zona);

```

Model SingleKnifeEdge.m

```

function [Gv_dB, n_zone] = Model_SingleKnifeEdge(h_obst, f0, d_tx, d_rx)
% h_obst - înălțimea obstacolului(m)
% f0 - frecvența semnalului transmis [Hz]
% d_tx - distanța dintre emițător și obstacol [m]
% d_rx - distanța dintre receptor și obstacol [m]
%-----
% Returnează:
% Gv_dB - pierderea datorată difracției knife-edge [dB]
% n_zone - numărul zonei Fresnel intersectate de obstacol
c0 = 3e8; % viteza luminii [m/s]
lambda = c0 / ?; % lungimea de undă [m]

% Parametru Fresnel-Kirchhoff
v_param = ? * sqrt((?/2) * (1/d_tx + 1/d_rx));

% Pierdere prin difracție aproximată
Gv_dB = Pierderi_Difracție(v_param);

% Diferența de drum între raza LOS și cea difractată
delta = (? / 2) * (1/? + 1/d_rx);

% Numărul zonei Fresnel blocate
n_zone = 2 * ? / lambda;
end

function [Gv] = Pierderi_Difracție(v)
% Calculează G(v) [dB] pentru parametrul Fresnel-Kirchhoff (v)
Gv = zeros(1, length(v));
idx = v > -0.7; % indicii unde v > -0.7
Gv(idx) = 6.9 + 20 * log10(sqrt((v(idx) - 0.1).^2 + 1) + v(idx) - 0.1);
Gv(~idx) = 0; % pentru v <= -0.7
end

```

FIȘĂ DE LUCRU

Sarcina 1

a) Reprezentare grafică:

b) Reprezentare grafică:

c) Reprezentare grafică:

Sarcina 2

a) Scripturi corectate:

b) Rezultate obținute:

Sarcina 3

Rezultate obținute:

Parametri modificați:

Sarcina 4

Capturi de ecran cu simulările realizate:

$f=90$ MHz, Zona Fresnel 1

$f=90$ MHz, Zona Fresnel 2

$f=2500$ MHz, Zona Fresnel 1

$f=2500$ MHz, Zona Fresnel 2

Lucrarea 4 – Efectul Doppler

Lucrarea de față urmărește evidențierea efectului Doppler în comunicațiile mobile și analiza consecințelor acestuia asupra calității semnalului recepționat.

Obiective

- O1. Prezentarea efectului Doppler și a consecințelor acestuia în comunicațiile mobile.
- O2. Prezentarea unor consecințe ale fadingului datorat propagării multicale pentru receptori în mișcare.
- O3. Efectuarea unor sarcini de lucru în Matlab privitoare la simularea modelelor matematice ale fenomenelor sus-amintite.

Durata medie de studiu individual: 30 minute.

4.1 DESCRIEREA FENOMENULUI

Fie semnalul modulat:

$$y(t) = a(t)\cos[2\pi f_c t + \theta(t)] \quad (4.1)$$

unde $a(t)$ reprezintă anvelopa lui $y(t)$ iar $\theta(t)$ reprezintă faza acestuia. Întrucât întreaga informație transmisă este modulată atât în anvelopa semnalului, cât și în faza acestuia, în continuare, semnalul poate fi analizat în varianta sa din banda de bază, numită și anvelopă complexă, astfel:

$$r(t) = a(t) \exp(j\theta(t)) = I(t) + jQ(t) \quad (4.2)$$

unde egalitatea din urmă marchează faptul că anvelopa complexă a semnalului poate fi reprezentată atât în termeni de amplitudine și fază, cât și prin componentele în fază și cuadratură [2].

Recuperarea lui $y(t)$ din $r(t)$ se poate realiza prin simpla multiplicare a celui din urmă cu fazorul purtătoarei, mai apoi luând partea reală a expresiei:

$$y(t) = \operatorname{Re}\{r(t) \exp(j2\pi f_c t)\} = \operatorname{Re}\{a(t) \exp[j(2\pi f_c t + \theta(t))]\} \quad (4.3)$$

Propagarea multicale joacă un rol semnificativ în comunicațiile mobile, deoarece antena unei stații mobile (MS) este de regulă situată la înălțimi joase și înconjurată de obstacole. În general, la recepție se folosesc diagrame de radiație omnidirecționale, preluând astfel un număr mare de ecouri ale semnalului transmis de la stația de bază (BS). În schimb, legăturile radio fixe de microunde sunt amplasate deasupra obstacolelor și utilizează antene directive care reușesc să deosebească între multiplele căi de propagare ale semnalului util [1].

Semnalul RF modulat poate fi reprezentat într-o formă complexă astfel:

$$y(t, z) = a(t)e^{j(\omega_c t - k_c z + \varphi(t_0))} \quad (4.4)$$

unde a_0 reprezintă amplitudinea semnalului direct, iar $\varphi(t_0)$ reprezintă faza inițială. k_c reprezintă constanta de propagare pentru frecvența f_c cu $k_c = 2\pi/\lambda_c$. Forma din banda de bază a semnalului este:

$$r(t, z) = a(t)e^{j(-k_c z + \varphi(t_0))} \quad (4.5)$$

Pentru canalele mobile în care terminalul este în mișcare, trebuie luată în considerare și viteza mobilului, V . În continuare, efectul Doppler va fi explicat din punct de vedere matematic [2], pornind de la scenariul ilustrat în Figura 4.1.

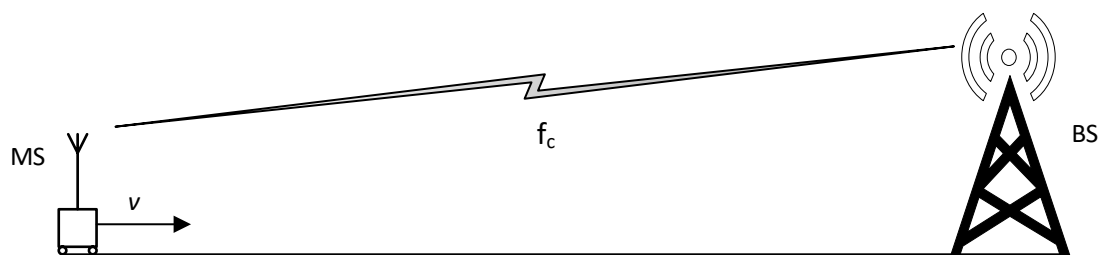


Figura 4.1 Scenariu cu utilizator mobil și o singură cale de propagare.

Dacă MS se apropie de BS cu o viteză V , anvelopa complexă a semnalului recepționat (dacă se omite faza inițială a semnalului), $r(t)$, va fi:

$$r(t) = a_0 \exp(-jk_c d(t)) = a_0 \exp\left(-j \frac{2\pi}{\lambda_c} d(t)\right) \quad (4.6)$$

Termenul $d(t)$ reprezintă distanța variabilă dintre MS-BS:

$$d(t) = d_0 - Vt \quad (4.7)$$

Putem în continuare omite termenul d_0 , iar astfel vom avea:

$$r(t) = a_0 \exp\left(j \frac{2\pi}{\lambda_c} Vt\right) = a_0 \exp(j2\pi f_D t) \quad (4.8)$$

Unde f_D reprezintă variația de frecvență indusă de efectul Doppler. Dacă reintroducem în ecuație și termenul f_c , semnalul RF recepționat va fi de forma:

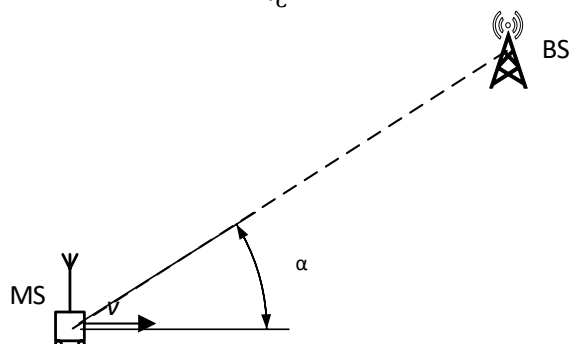
$$y(t) = a_0 \exp[j2\pi(f_c + f_D)t], \quad (4.9)$$

$$f_D = \frac{V}{\lambda_c} \quad (4.10)$$

Se poate observa cum semnalul RF nu mai este recepționat la frecvența f_c ci la $f_c + f_D$. Dacă MS s-ar îndepărta de BS, variația de frecvență Doppler va fi de $f_c - f_D$.

În plus, se poate demonstra că atunci când unghiul de deplasare al MS față de BS este α , ca în Figura 4.2, variația Doppler are forma:

$$f_D(\alpha) = \frac{V}{\lambda_c} \cos(\alpha) \quad (4.11)$$

Figura 4.2 Scenariu cu utilizator mobil și o singură cale de propagare sub unghiul de sosire α .

4.2 MODELUL CU DOUĂ RAZE ÎN SCENARIU MOBIL

În cazul scenariului ilustrat în Figura 4.3, MS se deplasează cu o viteză constantă de-a lungul axei Ox înspre un obstacol, în timp ce se îndepărtează de BS-ul la care este conectată. În această situație, MS va recepționa atât semnalul direct de la stația de bază cât și cel reflectat de la obstacolul de tip clădire. În spectrul semnalului recepționat vor apărea două vârfuri Doppler: unul corespunzător semnalului reflectat și altul corespunzător semnalului direct.

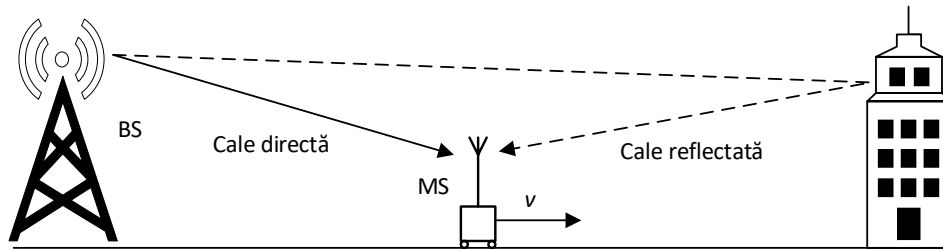


Figura 4.3 Scenariu mobil cu undă directă și undă reflectată.

Distanța parcursă de unda directă crește pe măsură ce MS se îndepărtează de BS, în timp ce distanța parcursă de unda reflectată scade pe măsură ce MS se apropie de reflector. Anvelopa semnalului rezultat reprezintă suma celor doi faze ai undelor directă și respectiv reflectată. Conform demonstrației din [2], anvelopa semnalului recepționat este de forma:

$$|m(t)| = |2\sin(2\pi f_D t)| \quad (4.12)$$

Astfel, se poate observa un model de fading de forma unei oscilații sinusoidale în anvelopa semnalului recepționat. Este interesant, de asemenea, de remarcat că anvelopa semnalului afectat de fading va avea minime și maxime și în domeniul spațial [2]:

$$|m(x)| = |2\sin\left(\frac{2\pi}{\lambda_c} x\right)| \quad (4.13)$$

4.3 DISPERSIA SEMNALULUI

Reflexia nu este singurul mecanism care dă naștere la propagarea multicală. Dispersia (scattering) este cea mai comună sursă a acestui fenomen de propagare. Dispersia semnalului este generată de obiecte mici, de exemplu, stâlpi pentru iluminatul public, semne de circulație, mașini, copaci sau diverse suprafețe neregulate care dau naștere unor ecouri mai slabe ale semnalului principal decât cele produse pe suprafețe mari, netede.

Pentru analiză, luăm în calcul două surse punctiforme de dispersie a semnalului ca singurele surse prin care semnalul transmis ajunge la receptor, în timp ce semnalul direct este blocat complet. Suntem interesați să legăm două aspecte ale fenomenului de propagare multicală: rata de fading și efectul Doppler.

Asemănător cazului precedent, expresia generală a ratei de fading pentru surse de dispersie plasate sub unghiurile α_1 și, respectiv, α_2 , față de direcția de deplasare a MS (Figura 4.4), este determinată prin însumarea fazorilor corespunzători celor două semnale dispersate, și ea este de forma [2]:

$$|s_r| = \left| 2\sin\left\{2\pi\left(\frac{f_D}{2}\right)[\cos(\alpha_1) - \cos(\alpha_2)]t\right\} \right| \quad (4.14)$$

Din această formulă deducem că odată cu scăderea dispersiei unghiulare, i.e. scăderea unghiului dintre α_1 și α_2 , rata de fading este redusă.

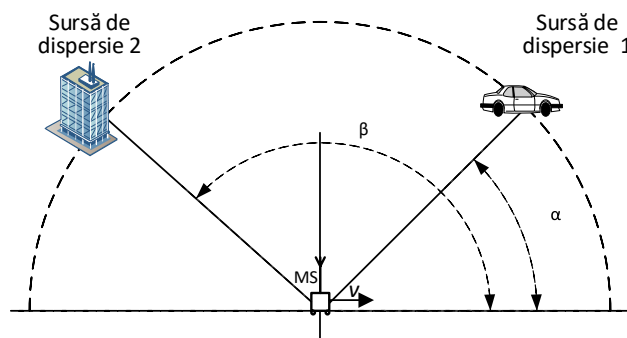


Figura 4.4 Scenariu mobil cu două surse de dispersie.

Sarcina 1

În [L4_Doppler.m](#), va fi simulat o legătură radio între o stație de bază și un receptor mobil. Să presupunem că este recepționat doar semnalul direct, neintroducându-se reflexii suplimentare. Figura 4.1 ilustrează situația analizată, unde BS este situată la o distanță de 1 km față de MS. Stația mobilă se deplasează cu viteza V înspre stația de bază. Modificați viteza de deplasare a MS la diferite valori conform tabelului din [Fișa de lucru](#). Notați în tabel frecvențele la care se găsesc vârfurile spectrale din graficul rezultat.

Sarcina 2

În [L4_Doppler_alfa.m](#) unghiul de deplasare al MS față de BS este de α , după cum este ilustrat în Figura 4.2. Modificați unghiul α și adăugați în [Fișa de lucru](#) graficele obținute. Comparați rezultatele obținute prin calcul cu cele obținute prin simulare. Notați frecvențele f_D la care se găsesc vârfurile spectrale din graficele rezultate.

Sarcina 3

Pornind de la scenariul ilustrat în Figura 4.3, rulați scriptul [L4_Doppler_direct_reflectat.m](#) și modificați viteza de deplasare a MS la diferite valori conform tabelului din [Fișa de lucru](#) și adăugați în [Fișa de lucru](#) graficele obținute. Notați în tabel frecvențele la care se găsesc vârfurile spectrale din graficele rezultate.

Stabiliți coeficientul de reflexie R la valoarea de -0.5. Ce modificări apar în spectru? Notați nivelul de putere al vârfurilor spectrale în [Fișa de lucru](#).

Sarcina 4

În Figura 4.4 este reprezentat schematic scenariul pe care îl simulăm în această sarcină. Rulați scriptul [L4_Doppler_2_surse_dispersie.m](#) și modificați unghiul dintre cele două surse de dispersie după valorile din tabelul din [Fișa de lucru](#). Observați variația Doppler asociată celor două surse de dispersie. În plus, remarcați variația formei anvelopei rezultate în urma interferenței undelor provenite de la cele două surse de dispersie. Adăugați în [Fișa de lucru](#) graficele obținute.

Sarcina 5

În această sarcină se va urmări propagarea undelor radio între un emițător și un receptor cu ajutorul metodei Ray Tracing pornind de la script-ul [L4_Ray_Tracing.m](#). Acesta este un model de propagare care calculează traiectoriile de propagare între un emițător și un receptor folosind geometria 3-D a mediului înconjurător. Astfel, se va personaliza un scenariu de propagare 3D la alegere și clădirile aferente acestui scenariu, stocate într-un fișier OpenStreetMap®, după care se va efectua analiza Ray Tracing în scenariul creat.

Pașii pentru efectuarea analizei Ray Tracing pe clădirile personalizate includ:

1. Citirea modelelor clădirilor dintr-un fișier OpenStreetMap într-un tabel geospațial.
2. Extragerea clădirilor din cadrul unei regiuni de interes.
3. Personalizarea clădirilor prin editarea tabelului geospațial.
4. Importarea clădirilor în Site Viewer.
5. Plasarea unui emițător pe o clădire și a unui receptor la sol, cu următorii parametri: $P = 1\text{ W}$, $f = 1900\text{ MHz}$, $h_{tx} = 10\text{ m}$, $h_{rx} = 2\text{ m}$.
6. Efectuarea analizei Ray Tracing utilizând clădirile și materialele selectate.

Adăugați în [Fișa de lucru](#) scenariul simulat în Site Viewer. De asemenea, adăugați caracteristicile a patru căi de propagare în tabelul din [Fișa de lucru](#).

Bibliografie

- [1] E. Puschita, s.a., *Radiocomunicatii Celulare - canalul radio - antene - proiectarea sistemelor – Manual de laborator*, U.T. PRESS, ISBN 978-973-662-496-4, 170 pag., 2009.
- [2] F. Perez Fontan, P. Marino Espineira, *Modeling the wireless propagation channel: a simulation approach with Matlab*, John Wiley & Sons Ltd, , ISBN 978-0-470-72785-0.

SCRIPT-URI MATLAB

L4 Doppler.m

```

clear
close all
clc

%% === Parametri de bază =====
fc = 2e9;           % Frecvența purtătoare (ex. 2000 MHz) [Hz]
F = 16;            % Rata de eșantionare: fracțiune din lungimea de undă
v_ms = 20;         % Viteza MS [m/s]
NFFT = 128;        % Număr de puncte FFT
N = 100;           % Număr de eșantioane

%% === Geometria scenariului =====
d_BS = 1000;        % Distanța BS față de origine [m]
alpha = 0;          % Unghiul BS-MS față de traseul MS [grade]

% Coordonatele BS
x_BS = d_BS * cosd(alpha);
y_BS = d_BS * sind(alpha);

%% === Parametri derivați =====
lambda = physconst('LightSpeed') / fc; % lungimea de undă [m]
Dx = lambda / F;    % pasul de eșantionare în spațiu [m]
Ts = Dx / v_ms;     % pasul de eșantionare în timp [s]
fs = 1 / Ts;        % frecvența de eșantionare [Hz]
k = 2*pi / lambda;  % constanta de propagare [rad/m]

% Axe pentru timp și spațiu
t_axis = Ts * (0:N); % axa timpului [s]
d_axis = Dx * (0:N); % distanța parcursă [m]

% Poziția MS pe traiectorie
x_MS = v_ms .* t_axis; % coordonata x a MS [m]
y_MS = 0;              % MS se deplasează pe axa x [m]

%% === Calcul distanță BS-MS =====
d_BSMS = sqrt((x_BS - x_MS).^2 + (y_BS - y_MS).^2);

%% === Anvelopa complexă (amplitudine + fază) =====
r = exp(-1i * k .* d_BSMS);

%% === Spectrul semnalului recepționat =====
R_f = fftshift(abs(fft(r, NFFT)).^2); % spectrul energiei
f_axis = (0:NFFT-1) * fs/NFFT - fs/2; % axa frecvențelor

figure, plot(f_axis, 10*log10(R_f) - max(10*log10(R_f)))
xlabel('Doppler shift [Hz]')
ylabel('Răspuns spectral normalizat [dB]')
title('Spectrul Doppler')
grid on

```

L4 Doppler alfa.m

```

clearvars; close all; clc

%% === Parametri de intrare =====
f0 = 2000;          % frecvența purtătoare [MHz]
M = 16;             % rata de eșantionare: fracțiune din lungimea de undă
v = 10;             % viteza MS [m/s]
Nfft = 128;        % număr de puncte FFT
Ns = 100;           % număr de eșantioane

%% === Configurația geometrică =====
d = 1000;           % distanța BS față de origine [m]
phi = 180;          % unghiul BS-MS față de traseul mobilului [grade]

% Poziția stației de bază (BS)
xBS = d * cosd(phi);
yBS = d * sind(phi);

%% === Parametri derivați =====
lambda = 300 / f0;  % lungimea de undă [m]
dx = lambda / M;    % pas de eșantionare spațial [m]
Ts = dx / v;        % interval de eșantionare temporal [s]
Fs = 1 / Ts;        % frecvența de eșantionare [Hz]
beta = 2*pi / lambda; % constanta de propagare [rad/m]

% Axe de timp și distanță
t = (0:Ns) * Ts;    % axa timpului [s]
x = (0:Ns) * dx;    % axa distanței [m]

% Poziția mobilului de-a lungul traseului
xMS = v .* t;

%% === Lungimea căii radio =====
dBS_MS = sqrt((xBS - xMS).^2 + (yBS).^2);

%% === Anvelopa complexă: amplitudine și fază =====
r = exp(-1j * beta .* dBs_MS);

%% === Spectrul anvelopei complexe =====
S = fftshift(abs(fft(r, Nfft)).^2);
f = (0:Nfft-1) * Fs/Nfft - Fs/2;

figure, plot(f, 10*log10(S) - max(10*log10(S)))
xlabel('Deviație Doppler (Hz)')
ylabel('Răspuns în frecvență normalizat (dB)')
title('Spectrul semnalului cu efect Doppler')
grid on

```

L4 Doppler direct reflectat.m

```

clearvars; close all; clc
% ===== Parametri de bază =====
f0 = 2000;          % frecvența purtătoare [MHz]
M = 16;             % fracțiune din lungimea de undă pentru eșantionare
vRx = 20;           % viteza MS [m/s]
Nfft = 128;         % număr de puncte FFT
Ns = 100;           % număr de eșantioane
R = -1;             % coeficient de reflexie
% ===== Geometrie =====
TxX = -1000;        % coordonata x a emițătorului (BS)
TxY = 0;            % coordonata y a emițătorului (BS)
RxX0 = 0;           % poziția inițială a MS pe axa x
RxY = 0;            % coordonata y a receptorului
phi1 = 0;           % unghiul față de direcția de deplasare [grade]
dSc1 = 1000;        % distanța de la origine la obstacol (scatterer 1) [m]

% ===== Poziție obstacol =====
Sc1X = dSc1*cosd(phi1); % coordonata x a obstacolului
Sc1Y = dSc1*sind(phi1); % coordonata y a obstacolului

% ===== Parametri auxiliari =====
lambda0 = 300/f0;     % lungimea de undă [m]
dx = lambda0/M;       % pas de eșantionare [m]
Ts = dx/vRx;          % interval de eșantionare [s]
Fs = 1/Ts;            % frecvența de eșantionare [Hz]
beta = 2*pi/lambda0;  % constanta de propagare [rad/m]
tAxis = Ts.*(0:Ns);   % axa timpului
RxX = RxX0 + vRx.*tAxis; % pozițiile succesive ale mobilului

% ===== Distanțele de propagare =====
dTxRx = sqrt((TxX - RxX).^2 + (TxY - RxY).^2); % directă
dTxSc1Rx = sqrt((TxX - Sc1X).^2 + (TxY - Sc1Y).^2) + ...
            sqrt((Sc1X - RxX).^2 + (Sc1Y - RxY).^2); % reflectată

% ===== Contribuția fiecărei căi =====
cRay = exp(-1j*beta.*dTxRx); % cale directă
rRay = R*exp(-1j*beta.*dTxSc1Rx); % cale reflectată

% ===== Semnalul total =====
h = cRay + rRay;

figure, plot(tAxis, abs(h), '--');

xlabel('Timp (s)')
ylabel('Amplitudinea anvelopei complexe')
xlim([0 0.02])

% ===== Spectru Doppler =====
Hf = fftshift((abs(fft(h,Nfft))).^2);
fAxis = (0:Nfft-1)*Fs/Nfft - Fs/2;

figure, plot(fAxis, 10*log10(Hf) - max(10*log10(Hf)))
xlabel('Deviație Doppler (Hz)')
ylabel('Răspuns în frecvență normalizat (dB)')

```

L4 Doppler 2 surse dispersie.m

```

clearvars; clc; close all

%% === Parametri de bază =====
f = 2e9;           % frecvența purtătoare (ex. 2000 MHz) [Hz]
F = 16;           % rata de eșantionare: fracțiune din lungimea de undă
v_ms = 20;        % viteza MS [m/s]
NFFT = 128;       % număr de puncte FFT
N = 100;          % număr de eșantioane

%% === Geometria scenariului =====
x_BS = -1000;      % coordonata x a BS [m]
y_BS = 1000;       % coordonata y a BS [m]
x_MS0 = 0;         % poziția inițială a MS (pe axa x) [m]

% Scatterer 1
x_SC1 = 100;
y_SC1 = 0;

% Scatterer 2
alpha2 = 60;       % unghiul față de traiectoria MS [grade]
d_SC2 = 100;        % distanța SC2 de origine [m]
x_SC2 = d_SC2 * cosd(alpha2);
y_SC2 = d_SC2 * sind(alpha2);

%% === Parametri derivați =====
lambda = physconst('LightSpeed') / f; % lungimea de undă [m]
Dx = lambda / F;    % pasul de eșantionare în spațiu [m]
Ts = Dx / v_ms;     % pasul de eșantionare în timp [s]
fs = 1 / Ts;        % frecvența de eșantionare [Hz]
k = 2 * pi / lambda; % constanta de propagare [rad/m]

% Axe timp
t_axis = Ts * (0:N); % axa timpului [s]

% Poziția MS în mișcare pe axa x
x_MS = x_MS0 + v_ms .* t_axis;
y_MS = 0; % MS se deplasează pe axa x

%% === Calcul distanțe de propagare =====
% Calea BS → SC1 → MS
d_BS_SC1_MS = sqrt((x_BS - x_SC1).^2 + (y_BS - y_SC1).^2) + ...
               sqrt((x_SC1 - x_MS).^2 + (y_SC1 - y_MS).^2);

% Calea BS → SC2 → MS
d_BS_SC2_MS = sqrt((x_BS - x_SC2).^2 + (y_BS - y_SC2).^2) + ...
               sqrt((x_SC2 - x_MS).^2 + (y_SC2 - y_MS).^2);
...

```

```
% == Semnalele de pe fiecare cale de propagare ==  
ray1 = exp(-1i * k .* d_BS_SC1_MS);  
ray2 = exp(-1i * k .* d_BS_SC2_MS);  
  
% Semnalul total recepționat  
r = ray1 + ray2;  
  
% == Reprezentări grafice ==  
% Variația amplitudinii în timp  
figure, plot(t_axis, abs(r), '--')  
xlabel('Timp [s]')  
ylabel('Amplitudinea anvelopei complexe |r|')  
title('Efectul fadingului cu 2 căi de propagare')  
grid on  
  
% Spectrul Doppler  
R_f = fftshift(abs(fft(r, NFFT)).^2);  
f_axis = (0:NFFT-1) * fs/NFFT - fs/2;  
  
figure, plot(f_axis, 10*log10(R_f) - max(10*log10(R_f)))  
xlabel('Variația Doppler [Hz]')  
ylabel('Răspuns spectral normalizat [dB]')  
title('Spectrul efectului Doppler (2 căi de propagare)')  
grid on
```

L4 Ray Tracing.m

```

%% Citirea clădirilor din fișierul OpenStreetMap
% Specificați numele unui fișier OpenStreetMap care conține date pentru
% mai multe clădiri din Cluj-Napoca.

filename = "map.osm";
%%
buildings = readgeotable(filename,Layer="buildingparts");

basemapName = "osm";
url = "a.tile.openstreetmap.org/${z}/${x}/${y}.png";
attribution = "@OpenStreetMap contributors";
addCustomBasemap(basemapName,url,Attribution=attribution)

figure
geobasemap(basemapName)
geoplot(buildings,FaceColor="#808080",FaceAlpha=1)
geotickformat dd
%% Specificați regiunea de interes
% Specificați un ROI (Region of Interest - regiune de interes). Puteți selecta
% interactiv patru puncte care definesc o regiune de interes.

[latROI,lonROI] = ginput(4);
%%
% Creați o formă poligonală din ROI. Afișați ROI pe aceeași hartă.

latROI(end+1) = latROI(1);
lonROI(end+1) = lonROI(1);
ROI = geopolyshape(latROI,lonROI);

hold on
geoplot(ROI,FaceColor="m",FaceAlpha=0.2,EdgeColor="m",LineWidth=2)
%% Extragerea clădirilor din regiunea de interes

[latmin,latmax] = bounds(latROI);
[lonmin,lonmax] = bounds(lonROI);
%%
% Obțineți formele poligoanelor din tabelul geospațial.

shape = buildings.Shape;
clipped = geoclip(shape,[latmin latmax],[lonmin,lonmax]);
%%
% Găsiți clădirile care se află integral sau parțial în interiorul ROI
idxInsideROI = clipped.NumRegions > 0;

```

```

% Creați un nou tabel geospațial care să conțină numai clădirile care se află
% integral sau parțial în ROI.

buildingsROI = buildings(idxInsideROI,:);
%%
% Afișați clădirile din ROI pe o hartă nouă.
figure
geobasemap(basemapName)
geoplot(buildingsROI,FaceColor="#7E2F8E",FaceAlpha=1)
%% Personalizarea clădirilor
% Personalizați clădirile din cadrul ROI prin schimbarea materialelor de
construcție ale acestora
for row = 1:height(buildingsROI)
    bldg = buildingsROI.Shape(row);
    buildingsROI.Material(row) = "concrete";
    buildingsROI.Color(row) = "#D95319";
end
%% Importarea clădirilor în Site Viewer
% Importați clădirile în Site Viewer. Afișați clădirile peste harta de bază
% OpenStreetMap.

viewer = siteviewer(Buildings=buildingsROI,Basemap=basemapName);
%% Efectuarea analizei Ray Tracing
% Creați un site emițător și un site receptor.

tx = txsite(Latitude=?,...
    Longitude=?,...
    AntennaHeight=?, ...
    TransmitterPower=?, ...
    TransmitterFrequency=?);
rx = rxsite(Latitude=?,...
    Longitude=?,...
    AntennaHeight=?);
%%
% Creați un model de propagare cu ray tracing
pm = propagationModel("raytracing",MaxNumReflections=?);
%%
% Extrageți traseele de propagare

rays = raytrace(tx,rx,pm);
rays = rays{1};

show(tx)
show(rx)
plot(rays)

% Puteți obține informații despre căile de propagare prin vizualizarea
% proprietăților obiectelor comm.Ray.

ray = rays(?) %Numărul căii de propagare

```

FIȘĂ DE LUCRU

Sarcina 1

Viteză MS (m/s)	Frecvența corespunzătoare vârfului spectral
5	
10	
20	
40	

Sarcina 2

Reprezentare grafică pentru $\alpha=180^0$, $f_D=?$

Reprezentare grafică pentru $\alpha=120^0$, $f_D=?$

Reprezentare grafică pentru $\alpha=60^0$, $f_D=?$

Sarcina 3

i) $R = -1$

Viteză MS (m/s)	Frecvența corespunzătoare vârfurilor spectrale
5	
10	
20	
40	

ii) $R = -0.5$

Viteză MS (m/s)	Frecvența corespunzătoare vârfurilor spectrale
5	
10	
20	
40	

Anvelopa semnalului recepționat ($V=40\text{m/s}$, $R=-1$):

Anvelopa semnalului recepționat ($V=20\text{m/s}$, $R=-1$):

Sarcina 4

Unghiul dintre sursele secundare	Valorile f_D pentru cele două semnale secundare
30°	
60°	
120°	
180°	

Reprezentare grafică spectru Doppler la 30° :

Reprezentare grafică spectru Doppler la 60^0 :

Reprezentare grafică spectru Doppler la 120^0 :

Reprezentare grafică spectru Doppler la 180^0 :

Sarcina 5

Reprezentare grafică Site Viewer împreună cu propagarea undelor radio folosind metoda Ray Tracing:

Caracterizarea a patru căi de propagare:

	Deviație de fază	Atenuare de propagare	Întârziere de propagare	Distanța parcursă	Număr de reflexii
Raza 1					
Raza 2					
Raza 3					
Raza 4					

Lucrarea 5 – Analiza propagării multicale

Un fenomen nedorit în ceea ce privește propagarea undelor radio este provocat în momentul apariției în calea semnalului a diferite obiecte care împiedică propagarea în regim de vizibilitate directă (Line of Sight – LOS). Acest fenomen poate conduce la apariția unor mecanisme de propagare ca reflexii, refracții, dispersii, difracții și absorbții. Toate aceste fenomene nedorite care determină variații importante în anvelopa semnalului recepționat determină apariția fenomenului de fading. În această lucrare se vor analiza diferite categorii de fading și modul în care acestea pot fi clasificate pe baza unor caracteristici specifice.

Obiective

- O1. Prezentarea unor noțiuni legate de diferitele tipuri de fading.
- O2. Prezentarea noțiunii de profil de putere al întârzierilor (Power Delay Profile-PDP).
- O3. Efectuarea unor simulări în Matlab pentru analiza fading-ului.

Durata medie de studiu individual: 30 minute.

5.1 PROFILUL DE PUTERE AL ÎNTÂRZIERILOR

Datorită mecanismelor de propagare enumerate, mai multe copii ale aceluiași semnal de interes pot ajunge la receptor, fenomenul fiind cunoscut sub numele de propagare multicale. Traseul undei directe (dacă acesta există) este cea mai scurtă cale între emițător și receptor. Orice altă undă care traversează mediul de propagare între emițător și receptor va parcurge o distanță mai mare, fiind întârziată în timp și defazată în raport cu unda directă. În consecință, semnalul radio la receptor va fi o combinație a undei directe cu undele reflectate și decalate în timp [1].

Prin modelul profilului de putere al întârzierilor (PDP), puterea semnalului pe fiecare cale de propagare este reprezentată în raport cu întârzierile de propagare aferente căii respective. Astfel, puterea semnalului (P_t dBm) este reprezentată în funcție de întârzierile de propagare de pe calea respectivă (τ_n).

În graficul din Figura 5.1, se poate observa cum un impuls transmis de către un emițător și care traversează un canal caracterizat de propagarea multicale este recepționat la nivelul receptorului ca având diferite întârzieri și diferite nivele de putere pe măsură ce se deplasează prin acest canal. PDP este de obicei furnizat sub forma unui tabel de valori, obținut din date empirice și acesta servește drept ghid pentru proiectarea sistemului de comunicații [2].

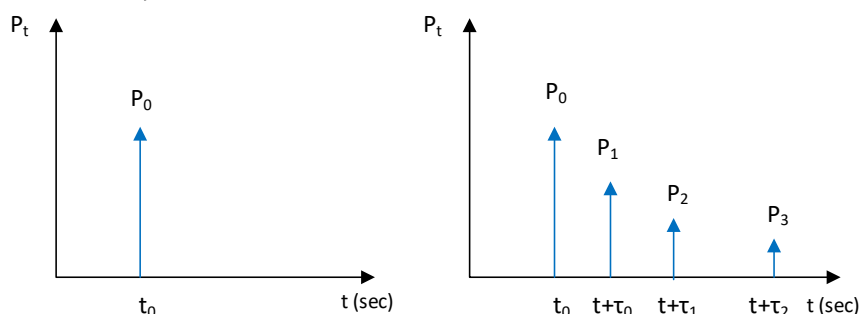


Figura 5.1 PDP pentru un canal cu propagare multicale cu 3 căi de propagare.

Valoarea medie pătratică (RMS) a dispersiei întârzierilor și întârzierea medie sunt doi dintre cei mai importanți parametri care caracterizează un canal selectiv în frecvență. Aceștia sunt derivați din profilul de întârziere a puterii.

Raportul dintre valoarea RMS a dispersiei întârzierii (τ_{rms}) și durata unui simbol (T_{sym}) cuantifică interferența inter-simbol (ISI). De obicei, atunci când durata unui simbol este mai mare de 10 ori decât valoarea RMS a dispersiei întârzierii, nu este nevoie de egalizor ISI la receptor. Fiind dat PDP, valoarea RMS a dispersiei τ_{rms} poate fi calculată ca [1]:

$$\tau_{rms} = \sqrt{\frac{\sum_i p_i (\tau_i - \bar{\tau})^2}{\sum_i p_i}} \quad (5.1)$$

unde p_i este puterea celei de a i -a căi de propagare, τ_i reprezintă întârzierea acestei căi, iar întârzierea medie relativă $\bar{\tau}$ este dată de ecuația [1]:

$$\bar{\tau} = \frac{\sum_i p_i \tau_i}{\sum_i p_i} \quad (5.2)$$

Cu ajutorul PDP, un canal se poate clasifica fie ca selectiv în frecvență, fie plat în frecvență.

Întârzierea relativă maximă (τ_{max}) este definită ca diferența de timp relativă dintre prima componentă de semnal care ajunge la receptor și ultima componentă al cărei nivel de putere este mai mare decât un anumit prag, stabilit în prealabil. Aceasta introduce interferențe intersimbol în semnalul care este transmis, distorsionându-l astfel.

5.2 FADING

Există diferite tipuri de fading, în funcție de cauzele perturbatoare ce produc acest fenomen, precum și în funcție de durata acestora. O primă clasificare a tipurilor de fading ar fi [2]:

- (1) fading la scară largă (large scale fading);
- (2) fading la scară redusă (small scale fading).

Fadingul la scară largă este în principal datorat fenomenului de umbrire a semnalului cauzată de variații ale profilului terenului sau clădirilor înconjurătoare.

Fadingul la scară redusă este caracterizat de fluctuații rapide în amplitudinea semnalului recepționat. Factorii cei mai importanți care determină acest tip de fading sunt:

- (1) propagarea multicală – prezența diferitelor obstacole pe direcția de propagare dintre emițător și receptor determină apariția unor replici (copii) multiple ale semnalului la recepție; diferențele în fază și în amplitudine de la recepție vor cauza fluctuații în puterea semnalului recepționat;
- (2) viteza de deplasare a mobilului – variația distanței dintre stația de bază și terminalul mobil determină fluctuații în frecvența semnalului recepționat, fluctuații care sunt datorate efectului Doppler.

În acest context, definim banda de coerență a canalului, B_c . Aceasta reprezintă gama de frecvențe pentru care canalul poate fi considerat uniform. Toate componentele cuprinse în această bandă vor avea aproximativ aceeași atenuare și aceeași variație a fazei. B_c este definită în funcție de dispersia întârzierilor (τ_{rms}), conform relației:

$$B_c = \frac{1}{50\tau_{rms}} \quad (5.3)$$

Pe un canal radio mobil ce introduce o atenuare și un defazaj constant într-o bandă care este mai mică decât banda semnalului transmis, semnalul recepționat va fi afectat de fading selectiv în frecvență. Prin urmare, dacă banda de coerență a canalului este mai mare decât banda semnalului transmis (B_s) ($B_s < B_c$) atunci canalul este clasificat ca fiind canal plat în frecvență. În caz contrar, ($B_s > B_c$) canalul este unul selectiv în frecvență.

Acum privind din punctul de vedere al întârzierilor apărute pe canalul radio, dacă întârzierea maximă a unui simbol transmis este mai mică decât durata aceluși simbol, adică $\tau_{max} < T_{sym}$, atunci canalul radio este clasificat ca fiind canal plat în frecvență. În acest caz, toate componentele semnalului dispersat rezultate de pe urma multiplelor căi de propagare, ajung la receptor înaintea încheierii duratei simbolului transmis. Pe de altă parte, un canal este clasificat

drept selectiv în frecvență, dacă întârzierea maximă înregistrată este mai mare decât durata unui simbol, adică $\tau_{max} > T_{sym}$. Acest tip de canal determină apariția interferenței intersimbol, provocând astfel distorsiuni asupra semnalului util. Acest lucru se întâmplă deoarece componentele semnalului rezultate de pe urma multiplelor căi de propagare, se extind dincolo de durata unui simbol [1].

Dispersia Doppler, determină două categorii majore de fading: fading rapid și fading lent. În vederea caracterizării canalului radio afectat de dispersia Doppler, este definit și timpul de coerență al canalului T_c . Acesta depinde de deviația de frecvență (f_d) datorată efectului Doppler conform relației [1]:

$$T_c = \frac{0.423}{f_d} \quad (5.4)$$

Într-un canal afectat de fading rapid, timpul de coerență al canalului va fi mai mic decât durata simbolului transmis ($T_c < T_{sym}$). În canalele afectate de fading lent, variațiile acestora sunt mult mai lente comparativ cu variațiile suferite de către semnalul din banda de bază. Astfel, timpul de coerență al canalului va fi mai mare decât durata simbolului transmis ($T_c > T_{sym}$). În Tabelul 5.1 este prezentată sintetic clasificarea tipurilor de fading descrise în paragrafele anterioare.

Tabelul 5.1 Clasificarea tipurilor de fading.

<i>Fading plat în frecvență</i>	$B_c > B_s$ $\tau_{rms} < T_{sym}$	<i>Fading rapid</i>	Dispersie Doppler mare în comparație cu banda semnalului $T_c < T_{sym}$
<i>Fading selectiv în frecvență</i>	$B_c < B_s$ $\tau_{rms} > T_{sym}$	<i>Fading lent</i>	Dispersie Doppler mică în comparație cu banda semnalului $T_c > T_{sym}$

Sarcina 1

Pentru un canal radio dat, se vor calcula: întârzierea medie de propagare, valoarea RMS a întârzierilor de pe canalul de propagare și lățimea benzii de coerență a canalului. Se ia în considerare un PDP discret dat ca o listă de valori de putere pentru cele 4 căi de propagare $P = [0, -10, -10, -20]$ (dBm) și întârzierile corespunzătoare $\tau = [0, 1, 2, 5]$ (μ sec). Utilizați scriptul [L5_PDP.m](#) pentru calculul acestor parametri.

Sarcina 2

Pornind de la scriptul [L5_Categorii_Fading.m](#) rezolvați cerințele de mai jos:

- Identificați ce tip de fading este prezent după rularea primei părți a scriptului.
- Câte componente dispersate ale semnalului transmis există conform PDP al semnalului recepționat?
- Modificați parametrii *delayVector* astfel încât să obțineți în cazul canalului Rayleigh fading plat în frecvență. Adăugați caracteristica de frecvență a canalului obținut în [Fișa de lucru](#).
- Păstrând parametrii de la punctul anterior, modificați rata de eșantionare la 5 MHz și identificați ce tip de fading este prezent.
- Modificați parametrii *delayVector* astfel încât să obțineți în cazul canalului Rayleigh fading plat în frecvență. Adăugați caracteristica de frecvență a canalului obținut în [Fișa de lucru](#).

Sarcina 3

Ray tracing [3] a devenit o tehnică uzuală de planificare și modelare a canalelor radio. Acest exemplu urmărește propagarea a undelor radio între un emițător și un receptor într-o sală de conferințe 3-D. Căile de propagare rezultate sunt utilizate pentru a construi un model de canal determinist specific pentru situația de comunicații analizată.

Pornind de la scriptul [L5_Propagare_Birou.m](#) setați frecvența de lucru la 2.4 GHz. Stabiliți ca numărul maxim de surse de reflexie să fie egal cu 1.

Cerințe:

- a) Adăugați reprezentarea grafică 3D obținută în [Fișa de lucru](#).
- b) Determinați numărul de căi de propagare simulate.
- c) Completați tabelul din [Fișa de lucru](#) cu rezultatele obținute.
- d) Considerând o putere transmisă de 10 mW, determinați întârzierea medie de propagare, valoarea RMS a întârzierilor de pe canalul de propagare și lățimea benzii de coerență a canalului. Utilizați funcția [calcul_PDP.m](#) de la Sarcina 1.

Bibliografie

- [1] E. Puschita, s.a., *Radiocomunicatii Celulare - canalul radio - antene - proiectarea sistemelor – Manual de laborator*, U.T. PRESS, ISBN 978-973-662-496-4, 170 pag., 2009.
- [2] Mathuranathan Viswanathan, *Wireless Communication Systems in Matlab*, 2nd Edition, ISBN: 9798648523210, 2020.
- [3] Z. Yun, and M. F. Iskander, „Ray tracing for radio propagation modeling: Principles and applications,” *IEEE Access*, vol. 3, pp. 1089-1100, Jul. 2015.

SCRIPT-URI MATLAB

L5 PDP.m

```

%----- Date de intrare -----%
% Puterile componentelor semnalului (în dBm)
puteri_dBm = [?, ?, ?, ?];

% Întârzierile corespunzătoare fiecărei componente (în secunde)
intarzieri_sec = [?, ?, ?, ?];

%----- Apelul funcției de analiză -----%
[intarziere_medie, dispersie_RMS, banda_coerenta] = ...
    calcul_PDP(?, ?);

%----- Afișarea rezultatelor -----%
fprintf('Rezultatele analizei canalului:\n');
fprintf('-----\n');
fprintf('Întârzierea medie      : %.2e s\n', intarziere_medie);
fprintf('Dispersia RMS           : %.2e s\n', dispersie_RMS);
fprintf('Lățimea benzii coerente  : %.2e Hz\n', banda_coerenta);
fprintf('-----\n');

function [intarziereMedie, dispersieRMS, latimeBandaCoerenta] =
    calcul_PDP(PdBm, intarzieri)
% Această funcție determină:
% - întârzierea medie
% - dispersia RMS a întârzierilor
% - rata maximă de simbol care poate fi transmisă fără interferență inter-
%   simbol (ISI)
% - lățime bandă de coerență

% Conversia valorilor din dBm în scară liniară
P_lin = 10.^(?/10);

% Calculul întârzierii medii ponderate
intarziereMedie = sum(P_lin .* ?) / sum(?);

% Determinarea dispersiei RMS a întârzierilor
dispersieRMS = sqrt(sum(P_lin .* (? - ?).^2) / sum(?));

% Calculul lățimii de bandă coerente pentru o corelație de ~0.9
latimeBandaCoerenta = 1 / (50 * ?);

stem(intarzieri, P_lin);

end

```

L5_ Categorii_Fading.m

```

%% Canal cu Fading Multipath Rayleigh
clearvars;
clc;

%% === Inițializarea parametrilor ===
sampleRate = 5e6;           % Rata de eșantionare [Hz]
maxDoppler = 200;           % Variația Doppler maximă [Hz]
delayVector = (0:5:15) * 1e-8; % Întârzieri discrete pentru cele patru căi [s]
gainVector_dB = [0 -3 -6 -9]; % Câștigurile medii ale fiecărei căi [dB]

%% === Crearea obiectului de canal ===
rayleighChan = comm.RayleighChannel( ...
    'SampleRate', sampleRate, ...
    'PathDelays', delayVector, ...
    'AveragePathGains', gainVector_dB, ...
    'MaximumDopplerShift', maxDoppler, ...
    'RandomStream', 'mt19937ar with seed', ...
    'Seed', 10, ...
    'PathGainsOutputPort', true);

M = 4;           % Modulație QPSK
phaseOffset = pi/4; % Deplasarea de fază pentru QPSK
bitsPerFrame = 20000;

%% === 1. Vizualizarea răspunsului la impuls al canalului ===
release(rayleighChan);

rayleighChan.Visualization = "Impulse and frequency responses";
rayleighChan.SamplesToDisplay = "100%";

numFrames = 2; % Număr de cadre pentru afișare
for i = 1:numFrames
    msg = randi([0 1], bitsPerFrame, 1); % Generare date aleatoare
    modSignal = pskmod(msg, M, phaseOffset, InputType='bit'); % Modulație QPSK
    rayleighChan(modSignal); % Propagare semnal
end

```

L5_Propagare_Birou.m

```

clc;
clearvars;
close all;

%% === Configurarea scenariului și a parametrilor inițiali ===
mapFileName = "conferenceroom.stl"; % Fișierul 3D al mediului de propagare

fc = ? ; % Frecvența purtătoare [Hz]
lambda = physconst("lightspeed") / fc; % Lungimea de undă [m]

%% === Definirea configurației antenelor ===
txArray = arrayConfig("Size", [1 1], "ElementSpacing", lambda); % Antenă Tx
rxArray = arrayConfig("Size", [1 1], "ElementSpacing", lambda); % Antenă Rx

%% === Configurarea stației de emisie ===
tx = txsite("cartesian", ...
    "Antenna", txArray, ...
    "AntennaPosition", [0.1; 0.2; 2], ...
    "TransmitterFrequency", fc);

%% === Configurarea stației de recepție ===
rx = rxsite("cartesian", ...
    "Antenna", rxArray, ...
    "AntennaPosition", [0; 0; 0.8], ...
    "AntennaAngle", [0; 90]);

%% === Vizualizarea mediului 3D ===
siteviewer("SceneModel", mapFileName);
show(tx, "ShowAntennaHeight", false);
show(rx, "ShowAntennaHeight", false);

%% === Definirea modelului de propagare ===
pm = propagationModel("raytracing", ...
    "CoordinateSystem", "cartesian", ...
    "Method", "sbr", ... % SBR - Shooting and Bouncing Rays
    "AngularSeparation", "low", ...
    "MaxNumReflections", ?, ... % 0 singură sursă de reflexie
    "SurfaceMaterial", "wood"); % Materialele pereților

%% === Calculul razelor de propagare ===
rays = raytrace(tx, rx, pm);
rays = rays{1,1}; % Extrage lista de raze între Tx și Rx

%% === Reprezentarea grafică a razelor ===
plot(rays, "Colormap", jet, "ColorLimits", [50, 95]);

%% === Configurarea canalului bazat pe ray tracing ===
rtChan = comm.RayTracingChannel(rays, tx, rx);
rtChan.SampleRate = 300e6; % Rata de eșantionare [Hz]
rtChan.ReceiverVirtualVelocity = [0.1; 0.1; 0]; % Viteza Rx [m/s]

%% === Vizualizarea profilului de putere în funcție de întârziere ===
showProfile(rtChan);

```

```
% == Analiza canalului ==  
numPaths = size(rays, 2); % Numărul de căi de propagare  
disp(['Numărul total de căi: ', num2str(numPaths)]);  
disp('Intarzierea pe fiecare cale de propagare este: ');  
[rays.PropagationDelay  
disp('Atenuarea pe fiecare cale de propagare este: ');  
[rays.PathLoss]  
% == Calculul caracteristicilor canalului ==  
txPower_dBm = ?; % Puterea transmisă [dBm]  
for i = 1:numPaths  
    delayList(i) = rays(i).PropagationDelay - rays(?).PropagationDelay; %  
    Întârzierea fiecărei căi  
    receivedPower_dBm(i) = ? - rays(i).PathLoss ; % Puterea recepționată  
end  
  
% Calculul întârzierii medii, a RMS, a benzii de coerență și a ratei maxime de  
simbol  
[intarziereMedie, dispersieRMS, latimeBandaCoerenta] = ...  
    meas_discrete_PDP(?, ?)
```

FIȘĂ DE LUCRU

Sarcina 1

Reprezentare grafică:

Întârziere medie (μs)	
Întârziere RMS (μs)	
Banda de coerență (MHz)	

Sarcina 2

Răspunsuri și reprezentări grafice

- a) Categoria de fading identificată
- b) Număr de componente dispersate ale semnalului transmis
- c) Caracteristica de frecvență a canalului

d) Categoria de fading identificată pentru $f_s=5$ MHz

e) Caracteristica de frecvență a canalului

Sarcina 3

a) Reprezentare grafică

b) Numărul de căi de propagare simulate

c)

Atenuare (dB)	Întârziere (μ s)

d)

Întârziere medie (μ s)	
Întârziere RMS (μ s)	
Banda de coerență (MHz)	

Lucrarea 6 – Analiza legăturii radio în medii 3D folosind Ray Tracing

Lucrarea de față are ca obiectiv aplicarea metodei Ray Tracing pentru analiza legăturilor radio dintre o stație de bază (BS) și o stație mobilă (MS), utilizând un scenariu urban real. Acesta va fi reprodus în Matlab prin intermediul unei hărți tridimensionale importate dintr-un fișier OpenStreetMap®, permițând simularea detaliată a propagării undelor radio în mediul construit.

Obiective

- O1.Importarea și vizualizarea unei hărți 3D în Matlab.
- O2.Definirea unui BS și analiza propagării LOS/NLOS utilizând metoda Ray Tracing.
- O3.Vizualizarea hărților de acoperire și evaluarea pierderilor de semnal datorate reflexiilor, difracțiilor și condițiilor meteo.
- O4.Adaptarea diagramei de radiație (Beamforming) pentru îmbunătățirea performanței legăturii radio.

Durata medie de studiu individual: 100 minute.

6.1 MODELUL RAY TRACING

Ray Tracing este o metodă geometrică de analiză a propagării undelor electromagnetice, utilizată în proiectarea și simularea rețelelor de comunicații mobile. Aceasta modelează interacțiunile undelor radio cu mediul tridimensional, luând în considerare diferite fenomene de propagare, cum ar fi reflexia, difracția și dispersia. Prin utilizarea acestei metode se pot estima puterea semnalului recepționat, pierderile de propagare și calitatea legăturii radio în scenarii urbane [1].

Pentru realizarea unei analize aprofundate și mai apropiată de realitate asupra acoperirii cu semnal radio a unei celule într-un mediu urban [2], urmați pașii de mai jos. Folosiți-vă de macheta de lucru furnizată în fișierul [L6_Simulare_Scenariu_Urban.mlx](#) și completați în paralel [Fișa de lucru](#) cu toate rezultatele cerute.

6.2 PAȘI DE IMPLEMENTARE

6.2.1 Importați și vizualizați datele clădirilor

Importați fișierul OpenStreetMap® care conține zona Campusului Observator – Sigma – Clădirea de Laboratoare UTCN. Verificați reprezentarea corectă a clădirilor în Site Viewer.

6.2.2 Definiți site-ul emițătorului

Definiți un emițător (BS) echipat cu o antenă izotropă plasată la înălțimea de 10 m și care emite cu puterea de transmisie de 5 W.

Pentru poziționarea stației de bază, extrageți datele necesare (latitudine, longitudine și frecvență) de pe site-ul [cellmapper.net](#) pentru o stație de bază situată în perimetrul hărții 3D analizate.

6.2.3 Vizualizați harta de acoperire pentru propagarea LOS

Creați un model de propagare de tip „raytracing” folosind metoda „sbr” (*shooting and bouncing rays*) și analizați harta de acoperire pentru propagarea LOS. Limitați analiza pe o rază de 250 m și setați numărul maxim de surse de reflexie la 0. Setati tipul de material de construcție al clădirilor, respectiv al terenului de tip conductor electric perfect.

6.2.4 Definiți site-ul receptorului într-o locație NLOS

După generarea hărții de acoperire pentru propagarea LOS se poate observa umbrirea cauzată de obstacole. Definiți un MS într-o locație umbrită de semnal (NLOS), în interiorul

campusului studentesc Observator. Setati înălțimea receptorului la 2 m și generați calea de propagare directă dintre MS și BS pentru a evidenția lipsa de conexiune dintre cele două entități.

6.2.5 Generați vizualizarea căilor de propagare radio utilizând metoda Ray Tracing

Ajustați modelul de propagare *Ray Tracing* pentru a include căi cu reflexie unică (*MaxNumReflections = 1*). Vizualizați caracteristicile de propagare corespunzătoare, inclusiv puterea recepționată, defazajul căilor analizate, distanța, unghiurile de plecare și de sosire ale semnalului, selectând calea de propagare dorită.

Dacă sunteți semnalat cu „Warning” în urma rulării acestei secțiuni, schimbați poziția receptorului până la remedierea situației.

6.2.6 Determinarea puterii recepționate

Determinați puterea recepționată de MS. Actualizați modelul pentru a utiliza materialul „concrete” (beton) pentru clădiri și sol. Actualizați, apoi, căile de propagare afișate în Site Viewer. Determinați noua putere recepționată de MS.

6.2.7 Includeți pierderile datorate condițiilor meteo

Actualizați mediul de propagare introducând noi condiții meteorologice. Determinați noua putere recepționată de MS.

6.2.8 Obțineți căile de propagare pentru căi cu maxim două surse de reflexie

Extindeți analiza punct la punct pentru a include două surse de reflexie și generați rezultatele simulării. Puterea totală recepționată pentru căi cu două surse de reflexie este similară cu puterea totală recepționată pentru căile cu o singură sursă de reflexie.

6.2.9 Obțineți căile de propagare pentru căi de propagare cu maxim două surse de reflexie și o sursă difracție

Extindeți analiza punct-la-punct pentru a include căi de propagare cu o singură sursă de difracție. Puterea recepționată nu se modifică în mod semnificativ prin includerea unei surse de difracție.

6.2.10 Vizualizați harta de acoperire pentru căile de propagare cu maximum o sursă de reflexie

Utilizați modelul de propagare configurat și generați o hartă de acoperire a BS care include de această dată și căile de propagare cu o singură sursă de reflexie și atenuări datorate condițiilor meteorologice.

6.2.11 Extinderea planificării radio

Pornind de la informațiile furnizate de cellmapper.net definiți un nou BS și analizați, urmărind pașii de mai sus, acoperirea noii BS și puterea recepționată de MS în condițiile de propagare enumerate mai sus.

6.2.12 (Opțional) Utilizarea tehnicii de beamforming pentru îmbunătățirea calității transmisiei

Sistemele de comunicații moderne utilizează tehnici de direcționare a lobului emițătorului pentru a obține o calitate superioară a legăturii radio. Această secțiune utilizează pachetul de lucru Phased Array System Toolbox™ pentru direcționarea lobului principal al unei rețele de antene în vederea maximizării puterii recepționate pentru o legătură fără vizibilitate directă.

Definiți o rețea de antene pentru evaluarea tehnologiei Beamforming. Creați o matrice dreptunghiulară uniformă de 8×8 cu distanța dintre elemente de $\lambda/2$. Setați numărul maxim de surse de reflexie la 1 și numărul maxim de surse de difracție la 0 pentru a efectua analiza pe calea dominantă cu o singură sursă de reflexie [3].

Obțineți unghiul de plecare pentru calea cu o singură sursă de reflexie și aplicați acest unghi pentru a orienta antena în direcția corespunzătoare pentru a îmbunătăți puterea recepționată de MS.

Generați diagrama de radiație a rețelei de antene direcționată de această dată de-a lungul căii de propagare dominante.

În concluzie, lucrarea de față a demonstrat utilizarea metodei Ray Tracing pentru evaluarea performanței unei legături radio între un MS și un BS într-un mediu urban. S-a observat influența reflexiilor și a materialelor asupra propagării. S-a evidențiat rolul atenuărilor suplimentare induse de condițiile atmosferice și importanța alegerii adecvate a parametrilor de propagare în analiza acoperirii radio, precum și. De asemenea, s-a analizat efectul de îmbunătățire a calității semnalului radio datorat tehnicii de Beamforming.

Bibliografie

- [1] Yun, Zhengqing, and Magdy F. Iskander. „Ray Tracing for Radio Propagation Modeling: Principles and Applications.” IEEE Access 3 (2015): 1089–1100. <https://doi.org/10.1109/ACCESS.2015.2453991>.
- [2] „Urban Link and Coverage Analysis Using Ray Tracing - Matlab & Simulink,” Mathworks.com, 2020. <https://se.mathworks.com/help/comm/ug/urban-channel-link-analysis-and-visualization-using-ray-tracing.html>
- [3] Report ITU-R M.2412, „Guidelines for evaluation of radio interface technologies for IMT-2020”, 2017.

SCRIPT-URI MATLAB***L6_Simulare_Scenariu_Urban.mlx***

```
%% 1. Importați și vizualizați datele clădirilor
viewer = siteviewer(Buildings="map.osm",Basemap="topographic");

%% 2. Definiți site-ul emițătorului

tx = txsite(Name="Small cell transmitter", ...
    Latitude=?,...
    Longitude=?,...
    AntennaHeight=?, ...
    TransmitterPower=?, ...
    TransmitterFrequency=?);
show(tx)

%% 3. Vizualizați harta de acoperire pentru propagarea LOS

rtpm = propagationModel("raytracing", ...
    Method="sbr", ...
    MaxNumReflections=0, ...
    BuildingsMaterial="perfect-reflector", ...
    TerrainMaterial="perfect-reflector");

% Vizualizați harta de acoperire corespunzătoare pentru o rază maximă de 250
% de metri în jurul stației de bază

coverage(tx,rtpm, ...
    SignalStrengths=-130:-5, ...
    MaxRange=?, ...
    Resolution=3, ...
    Transparency=0.6)

%% 4. Definiți site-ul receptorului într-o locație NLOS

rx = rxsite(Name="Small cell receiver", ...
    Latitude=?, ...
    Longitude=?, ...
    AntennaHeight=2);

los(tx,rx)

%% 5. Generați vizualizarea căilor de propagare radio utilizând metoda Ray
Tracing

rtpm.MaxNumReflections = 1;
clearMap(viewer)
raytrace(tx,rx,rtpm)
```

```

%% 6. Analizați puterea semnalului și efectul materialelor
% Calculați puterea recepționată.

ss = sigstrength(rx,tx,rtpm);
disp("Putere recepționată cu reflexie perfectă: " + ss + " dBm")

rtpm.BuildingsMaterial = "concrete";
rtpm.TerrainMaterial = "concrete";
raytrace(tx,rx,rtpm)

% Recalculați puterea recepționată.

ss = sigstrength(rx,tx,rtpm);
disp("Putere recepționată folosind alte materiale: " + ss + " dBm")

%% 7. Includeți pierderile datorate condițiilor meteo*

rtPlusWeather = ...
    rtpm + propagationModel("gas") + propagationModel("rain");
raytrace(tx,rx,rtPlusWeather)

ss = sigstrength(rx,tx,rtPlusWeather);
disp("Putere recepționată cu pierderi atmosferice: " + ss + " dBm")

%% 8. Obțineți căile de propagare pentru căi cu maxim două surse de reflexie

rtPlusWeather.PropagationModels(1).MaxNumReflections = ?;
rtPlusWeather.PropagationModels(1).AngularSeparation = "low";

ss = sigstrength(rx,tx,rtPlusWeather);
disp("Putere recepționată cu două căi de reflexie: " + ss + " dBm")
clearMap(viewer)
raytrace(tx,rx,rtPlusWeather)

%% 9. Obțineți căile de propagare pentru căi de propagare cu maxim două surse
de reflexie și o sursă de difracție*

rtPlusWeather.PropagationModels(1).MaxNumDiffractions = ?;

ss = sigstrength(rx,tx,rtPlusWeather);
disp("Putere recepționată cu două căi de reflexie și o difracție: " + ss + "
dBm")
raytrace(tx,rx,rtPlusWeather)

```

```

%% 10. Vizualizați harta de acoperire pentru căile de propagare cu maximum o
sursă de reflexie

rtPlusWeather.PropagationModels(1).MaxNumReflections = 1;
rtPlusWeather.PropagationModels(1).MaxNumDiffractions = 0;
clearMap(viewer)
show(tx)

coverage(tx,rtPlusWeather, ...
    SignalStrengths=-120:-5, ...
    MaxRange=250, ...
    Resolution=2, ...
    Transparency=0.6)

%% 11. Utilizarea tehnicii de Beamforming pentru îmbunătățirea puterii
recepționate

azvec = -180:180; % Unghiuri azimuth ( [°])
elvec = -90:90; % Unghiuri elevație ( [°])
SLA = 30; % Atenuare maximă lobi laterali [dB](dB)
tilt = 0; % Unghi de înclinație (deg)
az3dB = 65; % lob la 3 dB în azimuth [°]
el3dB = 65; % lob la 3 dB în elevație [°]
lambda = physconst("lightspeed")/tx.TransmitterFrequency; % lungime de undă
[m]

[az,el] = meshgrid(azvec,elvec);
azMagPattern = -min(12*(az/az3dB).^2,SLA);
elMagPattern = -min(12*((el-tilt)/el3dB).^2,SLA);
combinedMagPattern = -min(-(azMagPattern + elMagPattern),SLA); % câștig
relativ (dB)

antennaElement =
phased.CustomAntennaElement("MagnitudePattern",combinedMagPattern);
tx.Antenna = phased.URA("Size",[8 8], ...
    "Element",antennaElement, ...
    "ElementSpacing",[lambda/2 lambda/2]);

% Determinați directivitatea rețelei de antene.

antennaDirectivity = pattern(tx.Antenna, tx.TransmitterFrequency);
antennaDirectivityMax = max(antennaDirectivity(:));
disp("Directivitatea antenei " + antennaDirectivityMax + " dBi")

% Orientați antena spre nord și vizualizați diagrama de radiație a acesteia.

tx.AntennaAngle = 90;
clearMap(viewer)
show(rx)
pattern(tx,"Transparency",0.6)
hide(tx)

```

```
% Setati numărul maxim de surse de reflexie la 1 și numărul maxim de surse %
% de difracții la 0 pentru a efectua analiza pe calea dominantă cu o
% singură sursă de reflexie.

rtPlusWeather.PropagationModels(1).MaxNumReflections = 1;
rtPlusWeather.PropagationModels(1).MaxNumDiffractions = 0;
ray = raytrace(tx,rx,rtPlusWeather);
disp(ray{1})

% Obțineți unghiul de plecare pentru calea cu o singură sursă de reflexie
% și aplicați acest unghi pentru a orienta antena în direcția
% corespunzătoare pentru a îmbunătăți puterea recepționată de MS.

aod = ray{1}.AngleOfDeparture;
steeringaz = wrapTo180(aod(1)-tx.AntennaAngle(1));
steeringVector = phased.SteeringVector("SensorArray",tx.Antenna);
sv = steeringVector(tx.TransmitterFrequency,[steeringaz;aod(2)]);
tx.Antenna.Taper = conj(sv);

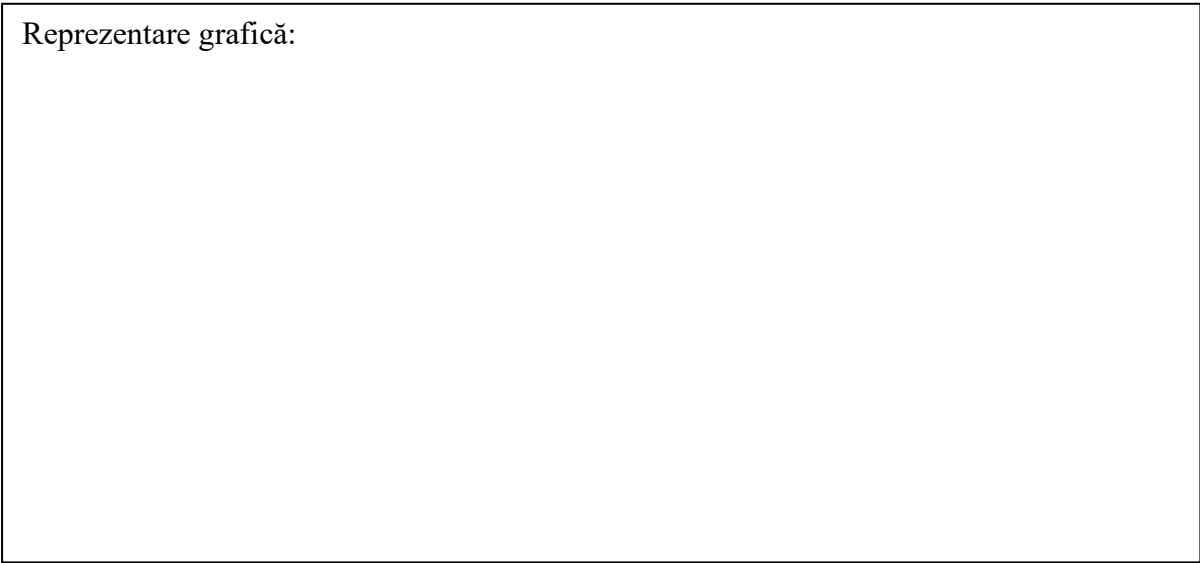
% Generați diagrama de radiație a rețelei de antene direcționată de-a
% lungul căii de propagare prin trasarea diagramei de radiație.

pattern(tx,"Transparency",0.6)
raytrace(tx,rx,rtPlusWeather);
hide(tx)

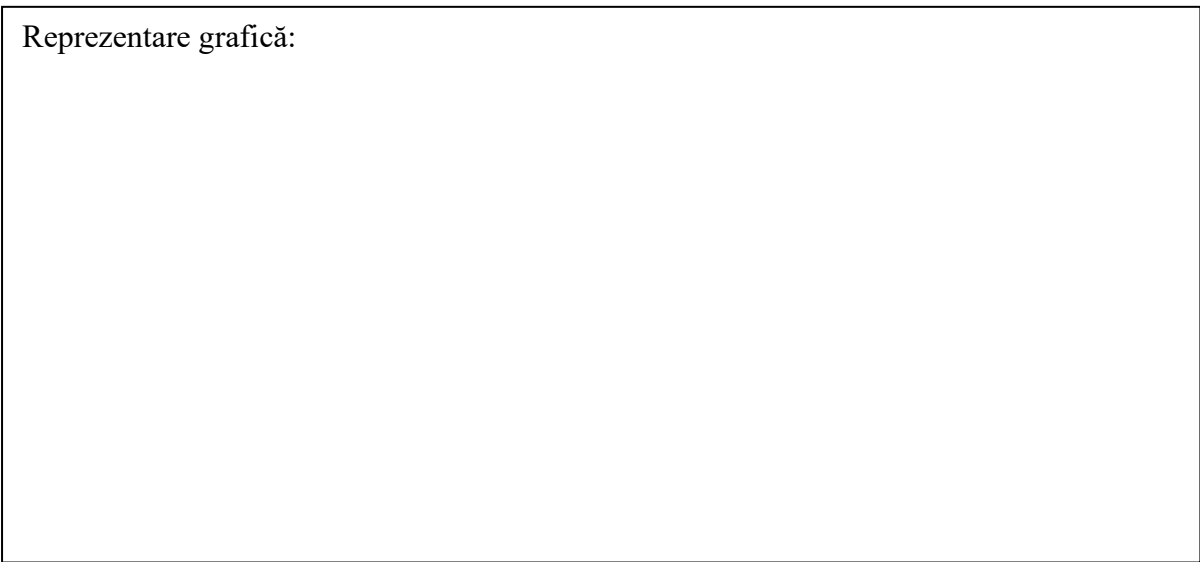
ss = sigstrength(rx,tx,rtPlusWeather);
disp("Putere recepționată după beamforming " + ss + " dBm")
```

FIȘĂ DE LUCRU**1. Importați și vizualizați scenariul 3D**

Reprezentare grafică:

**2. Definiți site-ul emițătorului**

Reprezentare grafică:

**3. Vizualizați harta de acoperire pentru propagarea LOS**

Reprezentare grafică:



4. Definiți site-ul receptorului în locație NLOS

Reprezentare grafică:

5. Trasați căile de propagare utilizând Ray Tracing

Reprezentare grafică:

6. Analizați puterea semnalului și efectul materialelor

Putere recepționată (dBm):

7. Includeți pierderile datorate condițiilor meteo

Putere recepționată (dBm):

8. Trasați căile de propagare pentru căi cu două surse de reflexie

Putere recepționată (dBm):

Reprezentare grafică:

9. Trasați căile de propagare pentru cazul cu două surse de reflexie și o difracție

Putere recepționată (dBm):

Reprezentare grafică:

10. Vizualizați harta de acoperire cu trasee cu o singură reflexie

Reprezentare grafică:

11. Utilizarea tehnicii de beamforming pentru îmbunătățirea calității semnalului

Putere recepționată (dBm):

Reprezentare grafică:

Lucrarea 7 – Conceptul celular. Reutilizarea frecvențelor. Interferențe specifice sistemelor celulare

Conceptul celular stă la baza tuturor sistemelor moderne de comunicații fără fir, deoarece permite ca un număr fix de canale să deservească un număr arbitrar de mare de abonați prin reutilizarea canalelor în întreaga regiune de acoperire. În acest laborator se vor explora și simula concepte și parametri fundamentali din spectrul proiectării rețelelor celulare. De asemenea, sunt simulate interferența co-canal și interferența canalului adiacent, evidențiindu-se sursele acestora și impactul asupra calității legăturii radio.

Obiective

- O1.Prezentarea conceptului celular.
- O2.Prezentarea conceptului de reutilizare a frecvențelor.
- O3.Prezentarea principiului de divizare a celulelor.
- O4.Efectuarea unor sarcini de lucru în Matlab privitoare la fenomenele sus-amintite.

Durata medie de studiu individual: 30 minute.

7.1 PRINCIPIILE CONCEPTULUI CELULAR

Conceptul celular a reprezentat o descoperire majoră în ceea ce privește rezolvarea problemei congestiei spectrale și a capacității unei rețele mobile [1]. Conceptul celular prezintă următoarele caracteristici majore:

1. Înlocuirea unui singur transmițător de mare putere (celulă mare) cu mai multe transmițătoare de putere mică (celule mici), fiecare dintre acestea asigurând acoperirea doar a unei mici părți din zona de serviciu.
2. Fiecărei stații de bază i se alocă o parte din numărul total de canale disponibile pentru întregul sistem.
3. Stațiilor de bază învecinate li se atribuie diferite grupuri de canale astfel încât interferențele dintre stațiile de bază (și utilizatorii mobili aflați sub controlul acestora) să fie reduse la minimum.
4. Pe măsură ce cererea de servicii crește, numărul stațiilor de bază poate fi mărit (împreună cu o scădere corespunzătoare a puterii emițătorului pentru a evita interferențele suplimentare), oferind astfel o capacitate radio suplimentară fără o creștere suplimentară a spectrului radio.

7.2 REUTILIZAREA FRECVENȚELOR

Prin limitarea zonei de acoperire a unei stații de bază la limitele unei celule, același grup de canale poate fi utilizat pentru a acoperi diferite celule care sunt separate una față de cealaltă prin distanțe suficient de mari pentru a menține nivelul de interferență în limitele rezonabile.

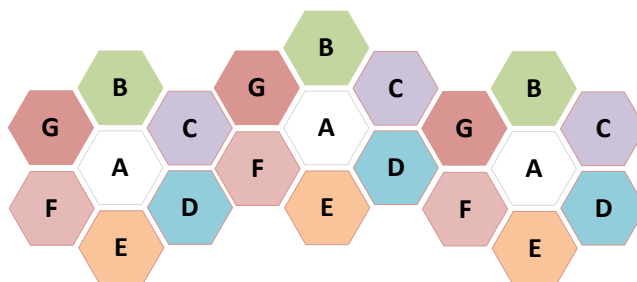


Figura 7.1 Ilustrarea conceptului de reutilizare a frecvențelor.

Procesul de proiectare, selecție și alocare a grupurilor de canale pentru toate stațiile de bază celulare din cadrul unui sistem se numește reutilizare a frecvențelor sau planificare a

frecvențelor [2].

Figura 7.1 ilustrează conceptul de reutilizare a frecvenței celulare, în care celulele etichetate cu aceeași literă folosesc același grup de canale. Pentru a înțelege conceptul de reutilizare a frecvențelor, se ia în considerare un sistem celular care are un total de S canale duplex disponibile pentru utilizare. Dacă fiecărei celule i se alocă un grup de k canale ($k < S$) și dacă cele S canalele sunt împărțite între N celule în grupuri de canale unice și disjuncte, care au fiecare același număr de canale, numărul total de canale radio disponibile poate fi exprimat ca [1]:

$$S = k \times N \quad (7.1)$$

Cele N celule care utilizează în mod colectiv întregul set de frecvențe disponibile formează un cluster. Dacă un cluster este replicat de M ori în sistem, numărul total de canale duplex din sistem, C , poate fi utilizat ca măsură a capacității sistemului și este dat de formula:

$$C = M \times k \times N = M \times S \quad (7.2)$$

După cum se poate vedea din formula anterioară, capacitatea unui sistem celular este direct proporțională cu numărul de replicări ale unui cluster într-o zonă fixă de servicii (M).

O dimensiune mai mare a clusterului indică un raport ridicat dintre raza celulei și distanța dintre co-celule. Pe de altă parte, o dimensiune mică a clusterului indică faptul că co-celulele sunt situate mult mai aproape una de cealaltă. Valoarea lui N depinde de cât de multe interferențe poate tolera o stație mobilă sau o stație de bază, menținând în același timp o calitate rezonabilă a comunicațiilor.

Din punct de vedere al proiectării, este de dorit a avea cea mai mică valoare posibilă a lui N pentru a maximiza capacitatea sistemului pe o anumită zonă de acoperire (adică pentru a maximiza C). Numărul de celule dintr-un cluster poate avea doar valori ce satisfac ecuația:

$$N = i^2 + ij + j^2 \quad (7.3)$$

unde $i, j \in \mathbb{N}^*$.

7.3 INTERFERENȚA CO-CANAL

Reutilizarea frecvenței implică faptul că într-o anumită zonă de acoperire există mai multe celule care utilizează același set de frecvențe. Aceste celule sunt numite co-celule, iar interferența generată de către aceste celule se numește interferență co-canal.

Pentru a reduce interferența co-canal, co-celulele trebuie să fie separate fizic la o anumită distanță pentru a reduce puterea semnalelor interferente recepționate de la co-celule [3]. O simplă creștere a puterii de transmisie a BS nu poate reduce efectele acestui tip de interferențe, deoarece creșterea puterii de transmisie determină creșterea interferenței în cazul co-celulelor învecinate.

Când dimensiunea fiecărei celule este aproximativ aceeași, iar stațiile de bază transmit cu aceeași putere, raportul de interferență co-canal devine dependent de raza celulei (R) și distanța dintre centrele celor mai apropiate co-celule (D). Prin creșterea raportului D/R , se mărește separarea spațială dintre co-celule în raport cu distanța de acoperire a unei celule. Astfel interferența este redusă. Parametrul Q , numit raportul de reutilizare a frecvențelor este legat de dimensiunea clusterului și este dat de expresia [2]:

$$Q = \frac{D}{R} = \sqrt{3N} \quad (7.4)$$

O valoare mică a lui Q oferă o capacitate mai mare, deoarece dimensiunea clusterului N este mică, în timp ce o valoare mare a lui Q îmbunătățește calitatea transmisiei, datorită unui nivel mai mic de interferență co-canal. Trebuie făcut un compromis între aceste două obiective în planificarea sistemului celular.

Dacă i_0 este numărul de co-celule care interferează cu stația de bază de referință, BS_{ref} , atunci raportul SIR (signal-to-interference ratio) poate fi exprimat astfel:

$$\frac{S}{I} = \frac{S}{\sum_{i=1}^{i_0} I_i} \quad (7.5)$$

Măsurătorile de propagare într-un canal radio mobil arată că puterea medie a semnalului recepționat în orice punct scade ca urmare a unei legi de putere a distanței de separare dintre emițător și receptor. Puterea medie recepționată P la distanța d de antena de transmisie este aproximată cu:

$$P_r = P_0 \left(\frac{d}{d_0} \right)^{-n} \quad (7.6)$$

unde P_0 este puterea recepționată la un punct de referință apropiat de antena BS-ului și n este exponentul pierderilor de propagare. Dacă D_i este distanța față de cel de-al i -lea semnal de interferență față de MS, puterea recepționată de MS datorită celei de-a i -a celulă de interferență va fi proporțională cu $(D_i)^{-n}$. Exponentul pierderii pe calea de propagare variază de obicei între 2 și 4 în sistemele celulare urbane [3].

Când stațiile de bază emit cu aceeași putere și exponentul de pierdere al căii de propagare este același în întreaga zonă de acoperire, raportul SIR pentru o stație mobilă poate fi aproximat ca [3]:

$$\frac{S}{I} = \frac{R^{-n}}{\sum_{i=1}^{i_0} D_i^{-n}} = \frac{\left(\frac{D}{R} \right)^n}{i_0} = \frac{\sqrt{3N}^n}{i_0} \quad (7.7)$$

Această ecuație leagă SIR-ul (raportul S/I) de dimensiunea clusterului (N), care la rândul ei determină capacitatea sistemului.

7.4 INTERFERENȚA CANALULUI ADIACENT

Interferența produsă de semnale aflate pe frecvențe adiacente față de semnalul dorit se numește interferența canalului adiacent (ACI - Adjacent Channel Interference). ACI apare din cauza filtrelor RF care nu atenuează complet semnalul din afara benzii canalului de referință. O altă cauză pentru apariția ACI este fenomenul de intermodulație produs de amplificatoarele stației de bază vecine, care determină dispersia spectrului semnalului transmis dincolo de banda alocată. Astfel, stația de bază vecină emite produse de intermodulație care se suprapun peste banda în care operează stația de bază curentă.

O situație des întâlnită este efectul de proximitate-depărtare. Problema apare, în momentul în care un utilizator $MS_{adiacent}$ campat pe un canal de frecvență adiacent unui alt utilizator MS_{ref} transmite de la o distanță foarte mică de receptorul MS_{ref} , în timp ce acesta încearcă să comunice cu BS. Astfel, $MS_{adiacent}$ „capturează” receptorul utilizatorului MS_{ref} . Alternativ, efectul de proximitate-depărtare apare și atunci când o stație mobilă $MS_{adiacent}$ situată în apropierea unui BS transmite pe un canal adiacent celui utilizat de o stație mobilă MS_{ref} depărtată de la care BS recepționează un semnal mai slab. BS poate avea dificultăți în a distinge semnalul recepționat de la MS_{ref} distant, de interferența cauzată de $MS_{adiacent}$ proxim, care emite pe canalul adiacent [1].

Pentru a limita acest efect, sistemele celulare utilizează strategii de alocare a canalelor care maximizează distanța în frecvență între canalele atribuite în aceeași celulă și evită utilizarea unor canale adiacente în celulele vecine. Astfel, în loc ca unei celule să îi fie atribuite canale consecutive în frecvență, se alocă un set de canale distanțate din punct de vedere spectral, ceea ce reduce semnificativ interferența canalului adiacent și crește calitatea recepției.

O altă soluție pentru combaterea acestui efect este controlul puterii emise de către

stațiile mobile. Acest mecanism asigură o putere recepționată aproximativ constantă la nivelul BS din partea fiecărui utilizator și diminuează astfel efectele negative ale fenomenului proximitate-depărtare. Controlul puterii este un aspect important în toate sistemele radio, deoarece reduce puterea transmisă la minimum necesar pentru ca o legătură de comunicație să funcționeze corect, menținând în același timp nivelurile de interferență la un nivel minim [1].

Sarcina 1

Dacă este necesar un raport semnal la interferență (SIR) de 15 dB pentru o performanță satisfăcătoare a canalului unui sistem celular, care este factorul de reutilizare al frecvenței (Q) și, respectiv, dimensiunea clusterului (N) care ar trebui utilizate pentru o capacitate maximă dacă exponentul pierderii pe calea de propagare este (a) $n = 4$, (b) $n = 3$? Să presupunem că există 6 co-celule plasate aproximativ la aceeași distanță față de MS.

Dacă raza unei celule este de 1 km, la ce distanță trebuie plasate co-celule pentru cele 2 cazuri de mai sus luând în calcul factorul Q determinat? Rezolvați această sarcină pornind de la macheta furnizată în [L7_Dimensionare_Sistem.m](#) și adăugați rezultatele în [Fișa de lucru](#).

Sarcina 2

Deschideți scriptul [L7_CCI_ACI.m](#) cu ajutorul căruia vom simula interferențele ACI și CCI. În Figura 7.2 găsiți reprezentată diagrama bloc a unui sistem de comunicații ce simulează interferențe ACI și CCI și pe care scriptul asociat acestei sarcini îl implementează [2]. Generați, după indicațiile din [Fișa de lucru](#), interferențe CCI și ACI. Modificați apoi parametrii indicați în [Fișa de lucru](#), pentru a combate interferențele sus-amintite. Adăugați rezultatele obținute și valorile parametrilor configurați în [Fișa de lucru](#).

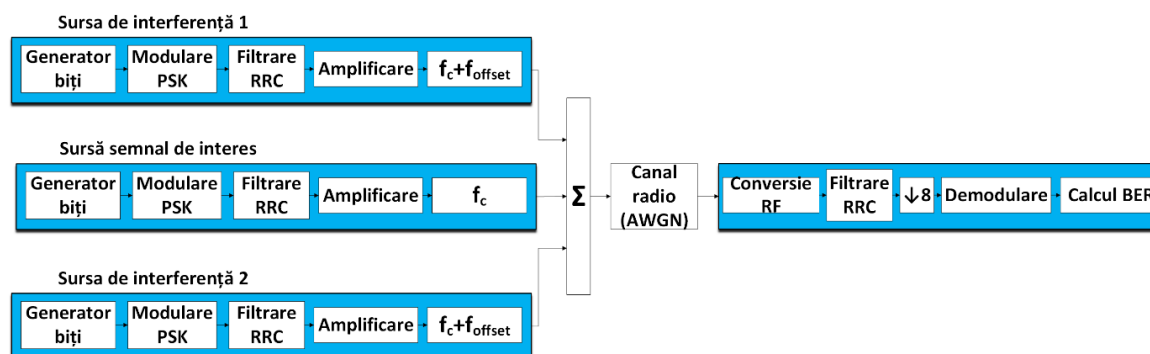


Figura 7.2 Diagrama bloc a unui sistem de comunicații ce simulează CCI și ACI.

Sarcina 3

Folosind scriptul [L7_SINR.m](#) analizați harta raportului Semnal-Interferență plus Zgomot (SINR) pentru un scenariu de comunicații celulare cu dimensiunea clusterului $N = 7$. Stabiliți poziția stației de bază centrale într-o locație la alegere, cu următorii parametri: $h_{BS} = 25$ m, $f_c = 2100$ MHz și $P_{TX} = 40$ dBm. Determinați valoarea $SINR_{mediu}$ pe o rază de 300 m față de stația BS centrală, pentru diferite combinații ale distanței dintre co-celule și coeficientul de pierdere pe calea de propagare, după cum este indicat în [Fișa de lucru](#). Adăugați atât hărțile SINR cât și valorile $SINR_{mediu}$ obținute în [Fișa de lucru](#).

Bibliografie

- [1] William C.Y. Lee, *Wireless and Cellular Telecommunications*, 3rd edition, McGraw-Hill, 2006.
- [2] T. Rappaport, *Wireless communications, principles and practice.*, Prentice Hall, 2002.
- [3] F. Perez Fontan, P. Marino Espinera, *Modelling the Wireless Propagation Channel*, John Wiley & Sons, Inc., 2008.

SCRIPT-URI MATLAB

L7 Dimensionare Sistem.m

```

clear; clc; close all;

%% Date de intrare
SIR_dB = ?;           % Raport semnal-interferență necesar [dB]
SIR_lin = 10^(?/10);   % Valoare liniara
i0 = ?;               % Număr co-celule interferente
R = ?;                % Raza celulei [km]

%% == Funcție pentru generarea clusterelor valide ==
generate_N_valid = @(imax) ...
    unique(arrayfun(@(i,j) ?^2 + i*j + ?^2, ...
        repelem(0:imax,imax+1), repmat(0:imax,1,imax+1))));

N_valid = generate_N_valid(3); % Generăm valori până la i,j = 3

%% =====
% (a) n = 4
%% =====
n1 = 4;

% N teoretic
Q_n1 = (i0 * ?)^(1/?);
N_n1_theoretical = ?^2 / 3;

% Alegem prima valoare validă > N_teoretic
N_candidates_1 = N_valid(N_valid >= N_n1_theoretical);

if isempty(N_candidates_1)
    error("Nu există valori valide N >= N_teoretic în intervalul generat!");
end

N_n1 = N_candidates_1(1); % rotunjire în sus

% Recalculăm Q și D
Q_n1 = sqrt(3 * ?);
D_n1 = ? * ?;

%% =====
% (b) n = 3
%% =====
n2 = 3;

% N teoretic
Q_n2 = (i0 * ?)^(1/?);
N_n2_theoretical = ?^2 / 3;

% Alegem prima valoare validă >= N_teoretic
N_candidates_2 = N_valid(N_valid >= N_n2_theoretical);

if isempty(N_candidates_2)
    error("Nu există valori valide N >= N_teoretic în intervalul generat!");
end

N_n2 = N_candidates_2(1);

```

```

% Recalculăm Q și D
Q_n1 = sqrt(3 * ?);
D_n1 = ? * ?;

%% =====
% (b) n = 3
%% =====
n2 = 3;

% N teoretic
Q_n2 = (i0 * ?)^(1/?);
N_n2_theoretical = ?^2 / 3;

% Alegem prima valoare validă >= N_teoretic
N_candidates_2 = N_valid(N_valid >= N_n2_theoretical);

if isempty(N_candidates_2)
    error("Nu există valori valide N >= N_teoretic în intervalul generat!");
end

N_n2 = N_candidates_2(1);

% Recalculăm Q și D
Q_n2 = sqrt(3 * ?);
D_n2 = ? * ?;

%% =====
% Afisare rezultate
%% =====
fprintf('\n===== RESULTATE =====\n');
fprintf('SIR necesar = %.1f dB | i0 = %d | R = %.1f km\n\n', SIR_dB, i0,
R);

fprintf('--- Pentru exponentul n = 4 ---\n');
fprintf('N teoretic           = %.3f\n', N_n1_theoretical);
fprintf('N valid                 = %d\n', N_n1);
fprintf('Factor reutilizare Q    = %.3f\n', Q_n1);
fprintf('Distanța co-canal D     = %.3f km\n\n', D_n1);

fprintf('--- Pentru exponentul n = 3 ---\n');
fprintf('N teoretic           = %.3f\n', N_n2_theoretical);
fprintf('N valid                 = %d\n', N_n2);
fprintf('Factor reutilizare Q    = %.3f\n', Q_n2);
fprintf('Distanța co-canal D     = %.3f km\n\n', D_n2);

```

L7 CCI ACI.m

```

clear; clc; close all;
disp('--- Configurare Simulare Interferențe ---');
disp(' ');
% --- 1. Rata de simboluri ---
symRate = 5e6;
% --- 2. Roll-off ---
prompt = 'Roll-off (0 - 1) [Implicit 0.5]: ';
t = input(prompt,'s');
if isempty(t), rolloff = 0.5; else, rolloff = str2double(t); end
% --- 3. Offset surse de interferență ---
prompt = 'Offset interferenți [Implicit 4e6]: ';
t = input(prompt,'s');
if isempty(t), baseOffset = 4e6; else, baseOffset = str2double(t); end
freqOffset1 = -baseOffset;
freqOffset2 = baseOffset;
% --- 4. Putere surse de interferență ---
prompt = 'Câștig sursă de interferență 1 (dB) [Implicit -10]: ';
t = input(prompt,'s');
if isempty(t), gain1_dB = -10; else gain1_dB = str2double(t); end
prompt = 'Câștig sursă de interferență 2 (dB) [Implicit -10]: ';
t = input(prompt,'s');
if isempty(t), gain2_dB = -10; else gain2_dB = str2double(t); end
gain1_lin = 10^(gain1_dB/20);
gain2_lin = 10^(gain2_dB/20);
fprintf("\nOK - Pornim simularea...\n\n");

%% ===== CONFIGURARE SIMULARE =====
M = 8;
k = log2(M);
sps = 8;
fs = symRate * sps;
frameSize = 1500;
bitsPerFrame = frameSize * k;
numFrames = 600;
SNR_dB = 50;
%% ===== SISTEM =====
pskMod = comm.PSKModulator('ModulationOrder', M, 'BitInput',
true, 'PhaseOffset', pi/M);
pskDem = comm.PSKDemodulator('ModulationOrder', M, 'BitOutput',
true, 'PhaseOffset', pi/M);

txFilter = comm.RaisedCosineTransmitFilter('Shape', 'Square
root', 'RolloffFactor', rolloff, 'OutputSamplesPerSymbol', sps);
rxFilter = comm.RaisedCosineReceiveFilter('Shape', 'Square
root', 'RolloffFactor', rolloff, 'InputSamplesPerSymbol', sps, 'DecimationFactor', sp
s);
freqOff1 = comm.PhaseFrequencyOffset('FrequencyOffset',
freqOffset1, 'SampleRate', fs);
freqOff2 = comm.PhaseFrequencyOffset('FrequencyOffset',
freqOffset2, 'SampleRate', fs);
noiseChan = comm.AWGNChannel('NoiseMethod', 'Signal to noise ratio
(SNR)', 'SNR', SNR_dB);
delay_syms = (txFilter.FilterSpanInSymbols + rxFilter.FilterSpanInSymbols)/2;
delay_bits = delay_syms * k;

```

```

%% ===== DISPLAY =====
span_MHz = max(baseOffset*2.5, symRate) * 2;
sa_Tx = dsp.SpectrumAnalyzer(...
    'SampleRate',fs,'Title','TX Spectrum',...
    'NumInputPorts',3,'ShowLegend',true,...
    'ChannelNames',{'Main','Int1','Int2'},...
    'FrequencySpan','Span and center frequency',...
    'Span',span_MHz,'CenterFrequency',0);

sa_Rx = dsp.SpectrumAnalyzer(...
    'SampleRate',fs,'Title','RX Spectrum',...
    'NumInputPorts',1,'ShowLegend',true,...
    'ChannelNames',{'Received + Noise'},...
    'FrequencySpan','Span and center frequency',...
    'Span',span_MHz,'CenterFrequency',0);

%% ===== MĂSURARE BER =====
totalErrors = 0;
totalBits = 0;

%% ===== SIMULARE =====
for f = 1:numFrames
    % ***** Semnal de interes *****
    bMain = randi([0 1], bitsPerFrame, 1);
    txMain = txFilter(pskMod(bMain));
    % ***** Interferență 1 *****
    bInt1 = randi([0 1], bitsPerFrame, 1);
    txI1 = txFilter(pskMod(bInt1));
    txI1 = freqOff1(txI1) * gain1_lin;
    % ***** Interferență 2 *****
    bInt2 = randi([0 1], bitsPerFrame, 1);
    txI2 = txFilter(pskMod(bInt2));
    txI2 = freqOff2(txI2) * gain2_lin;
    % ***** Spectru TX *****
    sa_Tx(txMain, txI1, txI2);
    % ***** Semnal total *****
    tx = txMain + txI1 + txI2;
    % ***** SNR real *****
    noiseChan.SignalPower = var(tx);
    rx = noiseChan(tx);
    % ***** Spectru RX *****
    sa_Rx(rx);
    % ***** RX *****
    rx = rxFilter(rx);
    bRx = pskDem(rx);
    % ***** Corectare întârziere *****
    bRx = bRx(delay_bits+1:end);
    bMain_sync = bMain(1:length(bRx));
    % ***** Calcul BER *****
    errors = sum(bRx ~= bMain_sync);
    totalErrors = totalErrors + errors;
    totalBits = totalBits + length(bRx);

    if mod(f, 100) == 0
        fprintf("Cadru %d / %d, BER curent = %.3e\n",f, numFrames,
            totalErrors/totalBits);
    end
end

```

```

%% ===== RESULTATE =====

BER = totalErrors / totalBits;

fprintf("\n===== \n");
fprintf("          RESULTATE FINAL \n");
fprintf("===== \n");
fprintf("Modulație: 8-PSK \n");
fprintf("Offset surse de interferență: ± %.2f MHz \n", baseOffset/1e6);
fprintf("Câștig sursă de interferență 1: %d dB \n", gain1_dB);
fprintf("Câștig sursă de interferență 2: %d dB \n", gain2_dB);
fprintf("SNR = %d dB \n", SNR_dB);
fprintf("BER FINAL = %.3e \n", BER);
fprintf("Erori totale: %d \n", totalErrors);
fprintf("Biți procesați: %d \n", totalBits);
fprintf("===== \n \n");

release(sa_Tx);
release(sa_Rx);

```

L7 SINR.m

```

clearvars; close all; clc;

%% -----
% 1. Definirea stației de bază centrale
% -----
% Aceasta reprezintă celula principală. În jurul ei se vor genera
% cele 6 co-celule la distanța D.

centerSite = txsite('Name','Cluj-Napoca', ...
    'Latitude',46.780487, ...
    'Longitude',23.601918);

%% -----
% 2. Definirea pozițiilor pentru cele 7 celule
% -----
numCellSites = 7;           % numărul total de celule
siteDistances = zeros(1,numCellSites);
siteAngles = zeros(1,numCellSites);
D = 800;                    % [m] Distanța co-canal [m]
% Pozițiile celor 6 celule din jur (la 60°)
siteDistances(2:7) = D;
siteAngles(2:7) = 30:60:360;

%% -----
% 3. Calcularea coordonatelor geografice pentru fiecare celulă
% -----
cellLats = zeros(1,numCellSites);
cellLons = zeros(1,numCellSites);
cellNames = strings(1,numCellSites);

cellInd = 1;
for siteInd = 1:numCellSites

    % Conversia distanță-unghi → coordonate geografice
    [lat, lon] = location(centerSite, ...
        siteDistances(siteInd), ...
        siteAngles(siteInd));
    cellNames(cellInd) = "Cell " + cellInd;
    cellLats(cellInd) = lat;
    cellLons(cellInd) = lon;
    cellInd = cellInd + 1;
end

%% -----
% 4. Definirea parametrilor stațiilor de bază
fq = 2100e6;                % Frecvența purtătoare [Hz]
antHeight = 25;             % Înălțime antenă [m]
txPowerDBm = 40;            % Putere emisă totală [dBm]
txPower = 10^((txPowerDBm - 30)/10); % Conversie în wați

% Crearea tuturor stațiilor de bază
txs = txsite('Name',cellNames, ...
    'Latitude',cellLats, ...
    'Longitude',cellLons, ...
    'AntennaHeight',antHeight, ...
    'TransmitterFrequency',fq, ...
    'TransmitterPower',txPower);

```

```

% 5. Afișarea celulelor pe harta reală (Site Viewer)
% -----
viewer = siteviewer;
viewer.Basemap = 'topographic';

show(txs)      % Afișează toate site-urile pe hartă
%% -----
% 6. Parametrii receptorului și modelul de propagare
% -----
bw = 20e6;                % Lățime de bandă [Hz]
rxNoiseFigure = 7;        % Factor de zgomot [dB]
rxNoisePower = -174 + 10*log10(bw) + rxNoiseFigure; % [dBm]

rxGain = 0;               % [dBi]
rxAntennaHeight = 2;      % [m]
Max_Range = 600;          % [m]

n= 2 % Exponent de pierderi pe calea de propagare
% Model de propagare "close-in"
pm = propagationModel("close-in", ...
    "PathLossExponent",n);
%% -----
% 7. Calculul și afișarea hărții SINR pentru întreaga rețea
% -----
sinr(txs, pm, ...
    'ReceiverGain',rxGain, ...
    'ReceiverAntennaHeight',rxAntennaHeight, ...
    'ReceiverNoisePower',rxNoisePower, ...
    'MaxRange',Max_Range);
%% -----
% 8. Calcul SINR pe întreaga zonă
% -----
SINR_map = sinr(txs, pm, ...
    'ReceiverGain',rxGain, ...
    'ReceiverAntennaHeight',rxAntennaHeight, ...
    'ReceiverNoisePower',rxNoisePower, ...
    'MaxRange',Max_Range); % doar pentru generarea datelor

data = SINR_map.Data; % format: [lat, lon, SINR_dB]

lat_vals = data(:,1);
lon_vals = data(:,2);
sinr_vals = data(:,3);
%% -----
% 9. Alege raza pentru evaluarea SINR
% -----
userRange = 300;
range_km = userRange / 1000;
fprintf('\nRaza: %d metri\n', userRange);
%% -----
% 10. Punctul central = site-ul 1
% -----
lat0 = txs(1).Latitude;
lon0 = txs(1).Longitude;

```

```

%% -----
% 11. Haversine – distanța fiecărui punct față de centrul celulei
% -----
R_earth = 6371; % km

dLat = deg2rad(lat_vals - lat0);
dLon = deg2rad(lon_vals - lon0);

a = sin(dLat/2).^2 + cos(deg2rad(lat0)) .* cos(deg2rad(lat_vals)) .*
sin(dLon/2).^2;
c = 2 * atan2(sqrt(a), sqrt(1-a));

dist_km = R_earth * c;

%% -----
% 12. Selectăm punctele din raza definită de utilizator
% -----
idx_in_range = dist_km <= range_km;

sinr_selected = sinr_vals(idx_in_range);

%% -----
% 13. Statisticile SINR
% -----
mean_sinr = mean(sinr_selected, 'omitnan');
min_sinr = min(sinr_selected);
max_sinr = max(sinr_selected);

fprintf('\n===== \n');
fprintf('Rezultate pentru raza de %d m\n', userRange);
fprintf('Puncte utilizate: %d\n', numel(sinr_selected));
fprintf('SINR mediu: %.2f dB\n', mean_sinr);
fprintf('SINR minim: %.2f dB\n', min_sinr);
fprintf('SINR maxim: %.2f dB\n', max_sinr);
fprintf('===== \n');

```

FIȘĂ DE LUCRU

Valorile indicilor i și j	Dimensiunea clusterului (N)	Raportul de reutilizare a frecvențelor (Q)
$i=1, j=1$		
$i=1, j=2$		
$i=2, j=2$		
$i=1, j=3$		

Sarcina 1

Rezultate obținute:

Script Matlab:

Sarcina 2

Parametri utilizați pentru simularea CCI:

- (a) Valoarea atenuării semnalelor de interferență: -6 dB
- (b) Offset-ul de frecvență al semnalelor de interferență: 0
- (c) Factorul de roll-off al semnalelor de interferență: 0.5
- (d) BER obținut:

Reprezentare grafică CCI:

Parametri utilizați pentru combaterea CCI:

- (a) Valoarea atenuării semnalelor de interferență: ?
- (b) Offset-ul de frecvență al semnalelor de interferență: 0
- (c) Factorul de roll-off al semnalelor de interferență: 0.5
- (d) BER obținut:

Reprezentare grafică după combaterea CCI:

Parametri utilizați pentru simularea ACI:

- (a) Valoarea atenuării semnalelor de interferență: 0 dB
- (b) Offset-ul de frecvență al semnalelor de interferență: 4 MHz
- (c) Factorul de roll-off al semnalelor de interferență: 0.5
- (d) BER obținut:

Reprezentare grafică ACI.

Parametri utilizați pentru combaterea ACI:

- (a) Valoarea atenuării semnalelor de interferență: 0 dB
- (b) Offset-ul de frecvență al semnalelor de interferență: ?
- (c) Factorul de roll-off al semnalelor de interferență: ?
- (d) BER obținut:

Reprezentare grafică după combaterea ACI:

Sarcina 3

Distanța D Coeficient n	400m	800m	1200m
2			
3			

D = 400 m, n=2

Hartă obținută:

D = 1200 m, n=3

Hartă obținută:

Lucrarea 8 – Divizarea celulei. Sectorizarea celulei. Procedura de handover

Acest laborator are ca obiectiv analizarea principalelor mecanisme utilizate în rețelele celulare pentru creșterea capacității și asigurarea continuității serviciului în condiții de mobilitate. Sunt studiate conceptele de divizare a celulei și sectorizare a celulei, ca soluții fundamentale pentru reutilizarea mai eficientă a spectrului radio și reducerea interferențelor co-canal. În același timp, procedura de transfer al legăturii este esențială pentru menținerea conexiunii utilizatorului aflat în mișcare între două celule. Acest laborator introduce aceste concepte și le analizează prin exemple și simulări Matlab pentru a înțelege impactul lor asupra performanței rețelelor mobile.

Obiective

- O1.Prezentarea mecanismului de divizare a celulei.
- O2.Prezentarea mecanismului de sectorizare a celulei.
- O3.Prezentarea conceptului de transfer al legăturii (Handover).
- O4.Efectuarea unor sarcini de lucru în Matlab privitoare la simularea fenomenelor sus-amintite.

Durata medie de studiu individual: 30 minute.

8.1 DIVIZAREA CELULEI

Divizarea celulelor este procesul de împărțire a unei celule congestionate în celule mai mici, fiecare cu propria sa stație de bază și cu o reducere corespunzătoare a înălțimii antenei și a puterii emițătorului.

Prin definirea de noi celule care au o rază mai mică decât celulele inițiale și prin instalarea acestor celule mai mici între celulele existente, capacitatea sistemului crește datorită numărului suplimentar de canale care poate fi utilizat într-o anumită arie de servicii. Capacitatea sistemului celular crește datorită creșterii numărului de reutilizări a canalelor radio [1].

În practică, nu toate celulele sunt împărțite în același timp. Este adesea dificil pentru furnizorii de servicii să găsească locații care să permită o divizare optimă a celulelor. Prin urmare, vor exista simultan celule de dimensiuni diferite. În astfel de situații, trebuie să se acorde o atenție deosebită distanței dintre co-celule.

Pentru ca noile celule să aibă dimensiuni mai mici, puterea de transmisie a acestor celule trebuie redusă. Puterea de emisie a unor celule a căror rază este jumătate din raza celulelor originale poate fi determinată prin examinarea puterii recepționate la granițele celulelor noi și vechi și egalarea celor două valori.

$$P_r(\text{pentru macro} - \text{celulă}) \propto P_{t1} R^{-n} \quad (8.1)$$

$$P_r(\text{pentru micro} - \text{celulă}) \propto P_{t2} (R/2)^{-n} \quad (8.2)$$

unde P_{t1} și P_{t2} sunt puterile de transmisie ale stațiilor de bază ale celulei mai mari și, respectiv, ale celei mai mici, iar n este exponentul pierderii de traseu.

8.2 SECTORIZAREA CELULEI

O altă modalitate de creștere a capacității sistemului celular este de a păstra neschimbată raza celulei și de a căuta metode de reducere a raportului D/R. Acest lucru se poate realiza prin reducerea dimensiunii cluster-ului. În acest sens, este necesară reducerea interferenței co-canal fără diminuarea puterii de emisie. Interferența co-canal dintr-un sistem celular poate fi redusă prin înlocuirea antenelor omnidirecționale de la nivelul stației de bază cu mai multe antene directive, fiecare radiind într-un sector specific. Această tehnică este denumită sectorizare [2].

Factorul de reducere a interferențelor co-canal depinde de gradul de sectorizare utilizat. În mod normal, o celulă este împărțită în trei sectoare de 120° sau șase sectoare de 60° , după cum este ilustrat în Figura 8.1. În acest caz, canalele alocate unei anumite celule sunt împărțite în grupuri și sunt utilizate numai în cadrul unui anumit sector, așa cum este ilustrat în Figura 8.1.

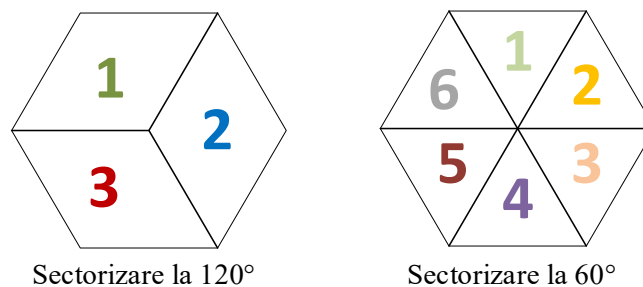


Figura 8.1 Sectorizarea celulelor.

Pornind de la scenariul ilustrat în Figura 8.2, să considerăm o sectorizare la 120° . Dacă plasăm un utilizator în sectorul 2 din celula centrală etichetată cu „A”, se poate observa că din punctul de vedere al interferențelor resimțite de acest utilizator, există 3 sectoare co-canal etichetate cu „A” în dreapta celulei centrale și 3 sectoare co-canal în stânga celulei centrale. Dintre aceste 6 co-celule, doar 2 au antene sectoriale care emit semnal de interferență în celula centrală. Prin urmare, utilizatorul din celula centrală va resimți interferențe doar din partea acestor 2 surse. Raportul semnal interferență, S/I, rezultat prezintă o îmbunătățire semnificativă față de cazul în care nu se utilizează antene sectoriale ci doar antene omnidirecționale [2].

Dezavantajele acestei tehnici sunt reprezentate de un număr mai mare de antene amplasate la nivelul fiecărei stații de bază și de o diminuare a eficienței „trunking-ului” din cauza sectorizării canalelor la nivelul stației de bază. În plus, apare nevoia unor proceduri de transfer al legăturii mai dese [2]. Multe stații de bază moderne sectorizate permit ca utilizatorii să fie transferați de la un sector la altul în cadrul aceleiași celule fără intervenția centrului de comutare mobilă (Mobile Switching Center - MSC), astfel încât problema transferului legăturii nu mai reprezintă neapărat o preocupare majoră.

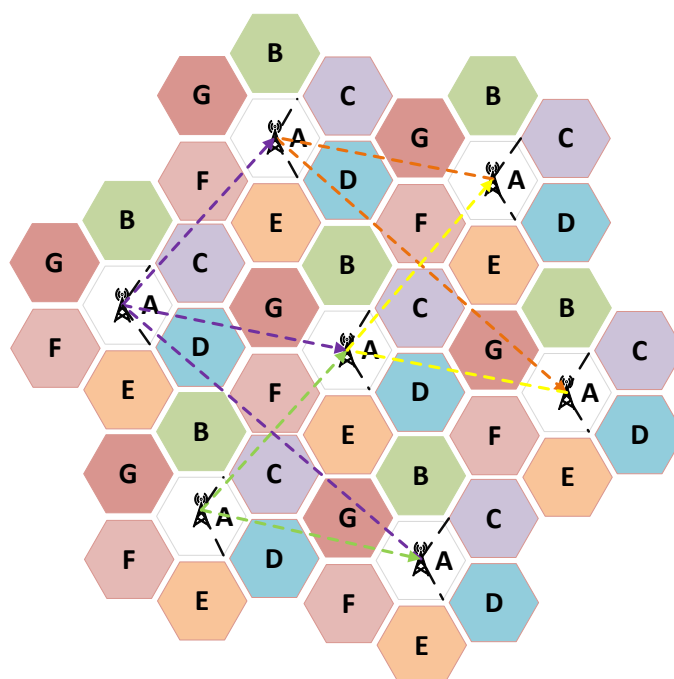


Figura 8.2 Ilustrare a modalității prin care sectorizarea la 120° reduce interferențele co-canal.

8.3 PROCEDURA DE HANDOVER

Atunci când un utilizator se deplasează dintr-o celulă în altă celulă în timpul unui apel, MSC-ul asigură transferul automat al conexiunii către un nou canal, aparținând stației de bază din celula în care a ajuns terminalul mobil. Această procedură presupune nu doar identificarea celei mai potrivite stații de bază, ci și realocarea canalelor pentru semnalul de voce și pentru semnalele de control aferente apelului.

Gestionarea transferurilor reprezintă una dintre cele mai importante funcții în rețelele celulare. În multe scheme de alocare a canalelor, cererile de transfer sunt tratate cu prioritate față de cererile de inițiere de apel, deoarece eșecul unui transfer duce la întreruperea unei convorbiri deja active. Un handover trebuie să fie realizat rapid, eficient și complet transparent pentru utilizator. Pentru a realiza acest lucru, este necesară stabilirea unui prag optim al nivelului semnalului la care transferul este declanșat.

Pornind de la puterea minimă a semnalului ($P_{r-minim}$) necesară pentru menținerea legăturii radio dintre MS și BS, nivelul de putere de la care se inițiază transferul legăturii ($P_{r-handover}$) este ales la o valoare mai ridicată. Diferența $\Delta = P_{r-handover} - P_{r-minim}$ trebuie selectată atent din două motive [1]:

1. dacă Δ este prea mare, se vor produce handover-uri dese, încărcând cu proceduri de semnalizare și alocare de resurse MSC-ul.
2. dacă Δ este prea mic, terminalul poate ajunge într-o zonă cu semnal scăzut înainte ca transferul să fie finalizat, provocând întreruperea apelului.

În plus, pentru a evita transferurile cauzate de variațiile rapide ale semnalului datorate fenomenului de fading, nivelul semnalului este monitorizat pe o perioadă scurtă de timp înainte de a decide inițierea handover-ului. Această mediere temporală trebuie configurată astfel încât să reducă numărul handover-urilor inutile, dar să nu întârzie exagerat efectuarea celor necesare.

În sistemele celulare digitale 2G, procesul de handover este asistat de terminalul mobil. În cadrul mecanismului Mobile Assisted Handover (MAHO), fiecare MS măsoară periodic puterea semnalelor provenite de la stațiile de bază învecinate și transmite aceste informații către stația de bază curentă. MAHO permite realizarea transferurilor mult mai rapid decât în rețelele analogice de primă generație, deoarece sarcina monitorizării calității semnalului este distribuită către terminalele mobile, reducând complexitatea monitorizării la nivelul entității MSC [1].

Sarcina 1

Fie un sistem GSM caracterizat de un raport semnal interferență minim (SIR) de 18 dB, $UL = [890 \text{ MHz}, 915 \text{ MHz}]$ și $DL = [935 \text{ MHz}, 960 \text{ MHz}]$, banda canalului de 200 kHz, exponentul de pierderi pe calea de propagare $n=4$, aria de acoperire de 36.4 km^2 , raza unei celule $R=1 \text{ km}$, și numărul de surse de interferență $i_0=6$, realizați un script Matlab pornind de la macheta furnizată în [L8 Analiza Sistem.m](#) care să determine:

- (a) Raportul de reutilizare a frecvențelor (Q), respectiv dimensiunea clusterului (N).
 - (b) Aria unei celule și numărul de celule necesar pentru a acoperi cu servicii suprafața dată. $Arie\ celulă = 2.6 \times R^2$
 - (c) Numărul de clustere din aria de serviciu analizată.
 - (d) Numărul de canale din sistem și numărul de canale per celulă.
 - (e) Distanța D dintre co-celule.
 - (f) Capacitatea sistemului, i.e. numărul total de canale disponibile în aria deservită.
- Dacă fiecare celulă este divizată în 4 noi celule mai mici, determinați:
- (g) Cerințele de putere a noilor stații de bază din micro-celule, știind că puterea stației de bază din celula nedivizată este de $P_{t1}=20 \text{ W}$.
 - (h) Capacitatea noului sistem.

Adăugați rezultatele obținute în [Fișa de lucru](#).

Sarcina 2

Configurați în scriptul [L8_Sectorizare.m](#) următorii parametri: dimensiunea clusterului $N=7$, distanța dintre co-celule $D=1200$ m, $P_t=44$ dBm, $f=2100$ MHz, $h_t=25$ m. Notați în [Fișa de lucru](#) valoarea medie a raportului semnal-interferență plus zgomot pe o rază de 300 m în jurul celulei centrale, obținut pentru cazul în care sectorizarea celulelor nu este utilizată.

Reduceți distanța dintre co-celule la $D=200$ m. Notați noua valoare medie a raportului semnal-interferență plus zgomot obținută.

În continuare, de-comentați porțiunea de cod responsabilă cu implementarea sectorizării celulelor. Analizați impactul sectorizării celulei. Notați raportul semnal-interferență plus zgomot mediu ($SINR_{\text{mediu}}$) obținut în acest caz. Adăugați hărțile SINR obținute în [Fișa de lucru](#).

Sarcina 3

Pornind de la amplasarea a două stații de bază situate în zona campusului studentesc Observator, vom studia un scenariu în care un utilizator mobil se deplasează de-a lungul unui traseu cuprins între acestea. În acest context, vor fi analizate procedurile de handover declanșate pe parcursul deplasării, utilizând scriptul [L8_Handover.m](#). Procedurile de handover vor fi declanșate în funcție de valoarea pragului de zgomot ($HO_Threshold$) și a marjei de zgomot (HO_Margin), permițând astfel simularea și compararea a două algoritmi diferiți de transfer al legăturii [3]. Mai întâi, analizați și descrieți în [Fișa de lucru](#) modul de lucru al celor doi algoritmi utilizați. Analizați apoi câte proceduri de handover au loc în cazul celor doi algoritmi, pentru configurațiile indicate în tabelul din [Fișa de lucru](#).

Adăugați figurile și valorile parametrilor obținuți în [Fișa de lucru](#).

Bibliografie

- [1] T. Rappaport, *Wireless communications, principles and practice*, Prentice Hall, 2002.
- [2] „Adjacent and Co-Channel Interference - Matlab & Simulink,” Mathworks.com, 2025.
<https://www.mathworks.com/help/comm/ug/adjacent-and-co-channel-interference.html>
- [3] F. Perez Fontan, P. Marino Espineira, *Modeling the wireless propagation channel: a simulation approach with Matlab*, John Wiley & Sons Ltd, ISBN 978-0-470-72785-0 (HB)

SCRIPT-URI MATLAB

L8 Analiza Sistem.m

```

clear; clc; close all;

%% === Date de intrare ===
SIR_dB = ?;
SIR_lin = 10^(?/10);
UL_band = [? ?];
DL_band = [? 960e6];
bw_canal = ?;
n = ?;
area_total = ?;           % [km^2]
R = ?;                   % [km]
i0 = ?;                  % numar surse de interferenta
Pt1 = ?;                  % [W]

%% ===== (a) Q și N =====
Q = (? * ?)^(1/n);
N_theor = ?^2 / ?;

% Construim lista completă de valori valide pentru N = i^2 + ij + j^2
validN = [];
for i = 0:10
    for j = 0:10
        N_val = ?^2 + ?*j + ?^2;
        if N_val > 0
            validN(end+1) = N_val;
        end
    end
end

validN = unique(sort(validN));

% Alegem prima valoare strict mai mare decât N_teoretic
N = validN(find(validN > N_theor, 1, 'first'));

%% ===== (b) Aria celulei și nr. celule =====
S_celula = ? * ?^2;
Nr_celule = ceil(? / ?);

%% ===== (c) Nr. clustere în aria de serviciu =====
Nr_clustere = ceil(? / ?);

%% ===== (d) Canale în sistem =====
bw_DL = DL_band(2) - DL_band(1);
Nr_canale_sistem = ? / ?;
Nr_canale_celula = floor(? / ?);

```

```

%% ===== (e) Distanța D =====
D = ? * ?;

%% ===== (f) Capacitatea sistemului =====
Capacitate_sistem = ? * ?;

%% ===== (g) Puterea microcelulelor =====
factor_divizare = 4;
R_micro = R / sqrt(factor_divizare);
Pt_micro = Pt1 * (? / R)^n;

%% ===== (h) Capacitatea noului sistem =====
Nr_celule_nou = Nr_celule * factor_divizare;
Capacitate_sistem_nou = ? * ?;

%% ===== Afișarea rezultatelor =====
fprintf('\n===== REZULTATE =====\n');
fprintf('SIR necesar          = %.1f dB (%.3f lin)\n', SIR_dB, SIR_lin);
fprintf('Q (factor reutilizare)   = %.3f\n', Q);
fprintf('N_teoretic                = %.3f\n', N_theor);
fprintf('N_ales (i^2+ij+j^2)       = %d\n', N);
fprintf('\nAria unei celule        = %.3f km^2\n', S_celula);
fprintf('Număr celule totale       = %d\n', Nr_celule);
fprintf('Număr clustere            = %d\n', Nr_clustere);
fprintf('\nBanda DL = %.1f MHz → canale total = %d\n', bw_DL/1e6,
Nr_canale_sistem);
fprintf('Canale per celulă         = %d\n', Nr_canale_celula);
fprintf('Distanța co-canal D      = %.3f km\n', D);
fprintf('Capacitate sistem        = %d canale\n', Capacitate_sistem);
fprintf('\n--- Microcelule ---\n');
fprintf('R_micro                   = %.3f km\n', R_micro);
fprintf('Putere micro-BS          = %.4f W (din %d W)\n', Pt_micro, Pt1);
fprintf('Nr. celule noi            = %d\n', Nr_celule_nou);
fprintf('Capacitate sistem nou    = %d canale\n', Capacitate_sistem_nou);
fprintf('\n===== \n');

```

L8 Sectorizare.m

```

%% ===== 1. Setare locație celulă centrală =====
% Definim stația de bază principală, poziționată în Cluj-Napoca.
% Aceasta va genera structura topologică a întregii rețele celulare.
clearvars; clc;

centerSite = txsite('Name','Cluj-Napoca', ...
    'Latitude',?, ...
    'Longitude',?);

N = ?; % Număr total de celule
siteDistances = zeros(1,N); % Vector cu distanțe față de celula centrală
siteAngles = zeros(1,N); % Vector cu unghiuri (în grade)

D = ?; % Distanță dintre co-celule [m]
siteDistances(2:7) = D; % Toate cele 6 celule sunt la aceeași distanță D
siteAngles(2:7) = 30:60:360; % Poziționare hexagonală la fiecare 60°

%% ===== 2. Definire parametri celule =====
% Aici se definesc coordonatele geografice ale fiecărei celule,
% iar fiecare celulă este sectorizată în 3 sectoare (sectors)

numCells = N;
cellLats = zeros(1,numCells*3); % Atenție – fiecare celulă devine 3 sectoare
cellLons = zeros(1,numCells*3);
cellNames = strings(1,numCells*3);
cellAngles = zeros(1,numCells*3);

cellSectorAngles = [30 150 270]; % 3 sectoare la 120°

cellInd = 1;
for siteInd = 1:N

    % Calculăm coordonatele geografice pornind de la celula centrală
    [cellLat,cellLon] = location(centerSite, siteDistances(siteInd),
    siteAngles(siteInd));

    % Creăm 3 sectoare pentru fiecare locație
    for cellSectorAngle = cellSectorAngles
        cellNames(cellInd) = "Cell " + cellInd;
        cellLats(cellInd) = cellLat;
        cellLons(cellInd) = cellLon;
        cellAngles(cellInd) = cellSectorAngle;
        cellInd = cellInd + 1;
    end
end
end

```

```

%% ===== 3. Creare site-uri transmițătoare =====
fq = ?; % Frecvență purtătoare[Hz]
antHeight = ?; % Înălțime antenă [m]
txPowerDBm = ?; % Putere TX [dBm]
txPower = 10.^(?/10); % Conversie în W

txs = txsite('Name',cellNames, ...
    'Latitude',cellLats, ...
    'Longitude',cellLons, ...
    'AntennaAngle',cellAngles, ...
    'AntennaHeight',antHeight, ...
    'TransmitterFrequency',fq, ...
    'TransmitterPower',txPower);

% Vizualizare pe hartă reală
viewer = siteviewer;
show(txs);
viewer.BaseMap = 'topographic';

%% ===== 4. Definire element antenă =====
% Modelam o antenă sectorială

azvec = -180:180; % Azimut (plan orizontal)
elvec = -90:90; % Elevație (plan vertical)
Am = 30; % Atenuare maximă [dB]
tilt = 0; % Nu introducem tilt
az3dB = 65; % Lățimea lobilor 3 dB orizontal
el3dB = 65; % Lățimea lobilor 3 dB vertical

% Crearea modelului de antenă
[az,el] = meshgrid(azvec,elvec);
azMagPattern = -12*(az/az3dB).^2;
elMagPattern = -12*((el-tilt)/el3dB).^2;

combinedMagPattern = azMagPattern + elMagPattern;

combinedMagPattern(combinedMagPattern < -Am) = -Am;

phasepattern = zeros(size(combinedMagPattern));

% Definim antena
antennaElement = phased.CustomAntennaElement( ...
    'AzimuthAngles',azvec, ...
    'ElevationAngles',elvec, ...
    'MagnitudePattern',combinedMagPattern, ...
    'PhasePattern',phasepattern);

```

```

%% ===== 5. Aplicare sectorizare cu downtilt =====
% Aplicăm downtilt pentru fiecare sector al fiecărei celule

% downtilt = 10; % [grade]
% for tx = txs
%     tx.Antenna = antennaElement;
%     tx.AntennaAngle = [tx.AntennaAngle; -downtilt];
% end

%% ===== 6. Parametri receptor & harta SINR =====
% Simulăm SINR pe toată zona de acoperire

bw = 20e6; % Lățime de bandă [Hz]
rxNoiseFigure = 7; % Factor de zgomot [dB]
rxNoisePower = -174 + 10*log10(bw) + rxNoiseFigure; % [dBm]

rxGain = 0;
rxAntennaHeight = 1.5;

if exist('f','var') && isvalid(f)
    close(f);
end

% Generăm harta SINR
sinr(txs,'close-in', ...
    'ReceiverGain',rxGain, ...
    'ReceiverAntennaHeight',rxAntennaHeight, ...
    'ReceiverNoisePower',rxNoisePower, ...
    'MaxRange',D, ...
    'Resolution',D/20);

%% ===== 7. Calcul SINR pentru întreaga zona =====

SINR_map = sinr(txs, 'close-in', ...
    'ReceiverGain',rxGain, ...
    'ReceiverAntennaHeight',rxAntennaHeight, ...
    'ReceiverNoisePower',rxNoisePower, ...
    'MaxRange',600);

data = SINR_map.Data;

lat_vals = data(:,1);
lon_vals = data(:,2);
sinr_vals = data(:,3);

%% ===== 8. Alegem raza pentru calcul SINR =====

userRange = 300;
range_km = userRange / 1000;

fprintf('\nRaza aleasă pentru analiză: %d metri\n', userRange);

```

```

%% ===== 9. Se calculează distanța față de centrul celulei =====

lat0 = txs(1).Latitude;
lon0 = txs(1).Longitude;

R_earth = 6371; % km

dLat = deg2rad(lat_vals - lat0);
dLon = deg2rad(lon_vals - lon0);

a = sin(dLat/2).^2 + cos(deg2rad(lat0)) .* cos(deg2rad(lat_vals)) .*
sin(dLon/2).^2;
c = 2 * atan2(sqrt(a), sqrt(1-a));

dist_km = R_earth * c;

%%===== 10. Se calculează SINR în punctele alese de utilizator =====

idx_in_range = dist_km <= range_km;

sinr_selected = sinr_vals(idx_in_range);

%% ===== 11. Statistici asociate SINR =====

mean_sinr = mean(sinr_selected, 'omitnan');
min_sinr = min(sinr_selected);
max_sinr = max(sinr_selected);

fprintf('\n===== \n');
fprintf('Rezultate pentru raza de %d m\n', userRange);
fprintf('Puncte utilizate: %d\n', numel(sinr_selected));
fprintf('SINR mediu: %.2f dB\n', mean_sinr);
fprintf('SINR minim: %.2f dB\n', min_sinr);
fprintf('SINR maxim: %.2f dB\n', max_sinr);
fprintf('===== \n');

```

L8 Handover.m

```

close all; clearvars; clc;

%% =====
% 1. PARAMETRI GENERALI ȘI HARTA
% =====

sv = siteviewer(Buildings="map.osm");

fc = 1900e6;                % Frecvența semnalului (Hz)
c = physconst("LightSpeed");
lambda = c/fc;              % Lungimea de undă (m)

% Antena UE (izotropă)
ueAntenna = phased.IsotropicAntennaElement;

%% =====
% 2. DEFINIREA RUTEI (UE în mișcare)
% =====
route.controlPoints.lats    = [46.7545  46.7553  46.7559];
route.controlPoints.lons    = [23.5840  23.5887  23.5957];
route.controlPoints.heights = [1.8      1.8      1.8]; % înălțime utilizator
route.spacing = 5;          % distanța dintre puncte

% Transformăm punctele în obiecte txsite
controlPts = txsite( ...
    Latitude      = route.controlPoints.lats, ...
    Longitude     = route.controlPoints.lons, ...
    AntennaHeight = route.controlPoints.heights);

% Generăm punctele UE de-a lungul rutei
rxs = computeWaypoints(controlPts, route.spacing, ueAntenna);

% Calculăm lungimea totală a traseului
totalDist = 0;
for k = 1:numel(controlPts)-1
    totalDist = totalDist + distance(controlPts(k), controlPts(k+1));
end

disp("Distanța totală " + totalDist + " m.");

% Afișare grafică puncte rută
show(controlPts, ShowAntennaHeight=false, Icon="pin.png", IconSize=[12 30]);
show(rxs, ShowAntennaHeight=false, Icon="OrangeDot.png", IconSize=[5 5]);

```

```

%% =====
% 3. DEFINIREA STAȚIILOR DE BAZĂ
% =====

tx1 = txsite(Name="BS1", ...
    Latitude=46.754817, Longitude=23.587841, ...
    AntennaHeight=10, TransmitterFrequency=fc, ...
    TransmitterPower=10);    % în dBm

tx2 = txsite(Name="BS2", ...
    Latitude=46.756189, Longitude=23.593147, ...
    AntennaHeight=10, TransmitterFrequency=fc, ...
    TransmitterPower=20);

show(tx1);
show(tx2);

%% =====
% 4. CALCUL RSS CU MODELUL DE PROPAGARE LONGLEY-RICE
% =====

propModel = propagationModel("longley-rice");
numPts = numel(rxs);

rss1 = zeros(1, numPts);
rss2 = zeros(1, numPts);

for i = 1:numPts
    % orientarea antenei BS către fiecare punct al rutei
    tx1.AntennaAngle = angle(tx1, rxs(i));
    tx2.AntennaAngle = angle(tx2, rxs(i));

    % RSS în dBm
    rss1(i) = sigstrength(rxs(i), tx1, propModel);
    rss2(i) = sigstrength(rxs(i), tx2, propModel);
end

% Vector distanță pentru plot
xAxis = route.spacing:route.spacing:totalDist;

%% =====
% 5. PLOTAREA PUTERII RECEPȚIONATĂ PE RUTĂ
% =====

figure;
plot(xAxis, rss1, "LineWidth", 1.4); hold on;
plot(xAxis, rss2, "LineWidth", 1.4);

xlim([0 totalDist]);
ylim([-80 -20]);

title("Puterea recepționată pe parcursul deplasării (Longley-Rice)");
xlabel("Distanța parcursă (m)");
ylabel("RSS (dBm)");
legend("BS 1", "BS 2", Location="southeast");
grid on;

```

```

%% =====
% 4. ALGORITMUL 1: HANDOVER SIMPLU (FARA MARJA)
% =====
HO_Threshold = -50;
HO_Margin = 15;

BSseries = [1];           % Vector istoric: Ce BS servește apelul
currentBS = 1;           % BS curent
RHO = rss1;              % Semnalul efectiv recepționat de mobil
switchBS = 0;            % Flag pentru comutare (0 = nu comuta, 1 = a comutat)
indexBS1 = length(rss1);
indexBS2 = length(rss2);

% Bucla principală de simulare pas cu pas
for ii = 2:length(rss1)

    % --- CAZUL A: Mobilul este conectat la BS1 ---
    if currentBS == 1 & switchBS == 0
        % Condiția 1: Semnalul BS1 > Prag -> Rămânem la BS1
        if rss1(ii) > HO_Threshold
            RHO = [RHO, rss1(ii)];
            currentBS = 1;
            BSseries = [BSseries; currentBS];
            switchBS = 1;
        else
            % Condiția 2: Semnal BS1 < Prag.
            if rss1(ii) < rss2(indexBS2-ii+1)
                % Handover către BS2
                RHO = [RHO, rss2(indexBS2-ii+1)];
                currentBS = 2;
                BSseries = [BSseries; currentBS];
                switchBS = 1;
            else
                RHO = [RHO, rss1(ii)];
                currentBS = 1;
                BSseries = [BSseries; currentBS];
                switchBS = 1;
            end
        end
    end
end
end

```

```

% --- CAZUL B: Mobilul este conectat la BS2 ---
if currentBS == 2 & switchBS == 0
    % Conditia 1: Semnalul BS2 > Prag -> Ramanem la BS2
    if rss2(indexBS2-ii+1) > HO_ThresHold
        RHO = [RHO, rss2(indexBS2-ii+1)];
        currentBS = 2;
        BSseries = [BSseries; currentBS];
        switchBS = 1;
    else
        % Conditia 2: Verificam daca puterea BS1 e mai mare decât puterea BS2.
        if rss2(indexBS2-ii+1) < rss1(ii)
            % Handover catre BS1
            RHO = [RHO, rss1(ii)];
            currentBS = 1;
            BSseries = [BSseries; currentBS];
            switchBS = 1;
        else
            RHO = [RHO, rss2(indexBS2-ii+1)];
            currentBS = 2;
            BSseries = [BSseries; currentBS];
            switchBS = 1;
        end
    end
end
switchBS = 0;      % Resetam flag-ul pentru urmatoarea iteratie
end

% --- FIGURA 2: Istoricul Comutarilor (Algoritm 1) ---
figure
plot(xAxis, BSseries(1:indexBS1), 'k')
a = axis; axis([a(1) a(2) 0 3]) % Ajustare axe
xlabel('Distanța parcursă (km)')
ylabel('BS care gestionează apelul (1 sau 2)')
title('Algoritm 1: Deciziile de Handover (Efect Ping-Pong)')

% =====
% 5. ALGORITMUL 2: HANDOVER CU MARJA (HISTEREZIS)
% =====
BSseries = [1];
currentBS = 1;
RHO = rss1(1);
switchBS = 0;
indexBS1 = length(rss1);
indexBS2 = length(rss2);

```

```

% Bucla principala de simulare
for ii = 2:length(rss1)

    % --- CAZUL A: Conectat la BS1 ---
    if currentBS == 1 & switchBS == 0
        if rss1(ii) > HO_ThresHold
            RHO = [RHO, rss1(ii)];
            currentBS = 1;
            BSseries = [BSseries; currentBS];
            switchBS = 1;
        else
            % Conditia de Histerezis:
            % Facem HO doar daca BS2 este mai bun decat BS1 + MARJA
            if rss1(ii) + HO_Margin < rss2(indexBS2-ii+1)
                RHO = [RHO, rss2(indexBS2-ii+1)];
                currentBS = 2;
                BSseries = [BSseries; currentBS];
                switchBS = 1;
            else
                RHO = [RHO, rss1(ii)];
                currentBS = 1;
                BSseries = [BSseries; currentBS];
                switchBS = 1;
            end
        end
    end

    % --- CAZUL B: Conectat la BS2 ---
    if currentBS == 2 & switchBS == 0
        if rss2(indexBS2-ii+1) > HO_ThresHold
            RHO = [RHO, rss2(indexBS2-ii+1)];
            currentBS = 2;
            BSseries = [BSseries; currentBS];
            switchBS = 1;
        else
            % Conditia de Histerezis:
            % Facem HO doar daca BS1 este mai bun decat BS2 + MARJA
            if rss2(indexBS2-ii+1) + HO_Margin < rss1(ii)
                RHO = [RHO, rss1(ii)];
                currentBS = 1;
                BSseries = [BSseries; currentBS];
                switchBS = 1;
            else
                RHO = [RHO, rss2(indexBS2-ii+1)];
                currentBS = 2;
                BSseries = [BSseries; currentBS];
                switchBS = 1;
            end
        end
    end

    switchBS = 0;
end

```

```

% --- FIGURA 3: Istoricul Comutarilor (Algoritm 2) ---
figure
plot(xAxis, BSseries(1:indexBS1), 'k')
a = axis; axis([a(1) a(2) 0 3])
xlabel('Distanța parcursă (km)')
ylabel('BS care gestionează apelul')
title(['Algoritm 2: Handover cu Histerezis (' num2str(H0_Margin) ' dB)'])

%% =====
% 6. FUNCTIE: GENERAREA WAYPOINT-URILOR DE-A LUNGUL RUTEI
% =====
function wayptRxsites = computeWaypoints(controlPts, spacing, ueArray)

totalDist = 0;
currDist = 0;
idx = 0;
offset = 0;

rxList = rxsite.empty;

for k = 1:numel(controlPts)-1

    totalDist = totalDist + distance(controlPts(k), controlPts(k+1));
    az = angle(controlPts(k), controlPts(k+1));

    [lat,lon] = location(controlPts(k), offset, az);

    idx = idx + 1;
    rxList(idx) = rxsite(Name=sprintf("P%d",idx), ...
        Latitude=lat, Longitude=lon, ...
        Antenna=ueArray, AntennaHeight=controlPts(k).AntennaHeight);

    currDist = currDist + spacing;

    while currDist + offset < totalDist
        [lat,lon] = location(rxList(idx), spacing, az);

        idx = idx + 1;
        rxList(idx) = rxsite(Name=sprintf("P%d",idx), ...
            Latitude=lat, Longitude=lon, ...
            Antenna=ueArray, AntennaHeight=controlPts(k).AntennaHeight);

        currDist = currDist + spacing;
    end

    offset = currDist - totalDist;
end

wayptRxsites = rxList;

end

```

FIȘĂ DE LUCRU

Sarcina 1

Rezultate obținute:

- a)
- b)
- c)
- d)
- e)
- f)

Script Matlab:

Sarcina 2

$SINR_{\text{mediu}}$ fără sectorizare, $D=1200$ m:

Hartă fără sectorizare:

$SINR_{\text{mediu}}$ cu sectorizare, $D=1200$ m:

Hartă fără sectorizare:

$SINR_{\text{mediu}}$ fără sectorizare, $D=200$ m:

Hartă fără sectorizare:

$SINR_{\text{mediu}}$ după sectorizare, $D=200$ m:

Hartă cu sectorizare:

Sarcina 3

Explicații funcționare algoritm Handover 1:

Explicații funcționare algoritm Handover 2:

HO_Threshold: -80 dbm

Figura obținută pentru primul algoritm de Handover:

HO_Margin: 9 dB

HO_Threshold: -100 dbm

Figura obținută pentru a doilea algoritm de Handover:

HO_Threshold	HO_Margin	Număr proceduri HO. Algoritm 1	Număr proceduri HO. Algoritm 2
-80 dBm	0dB		
-100 dBm	0dB		
-90 dBm	9dB		
-90 dBm	18dB		
-100 dBm	9dB		

Lucrarea 9 – Traficul celular

Trunking-ul este o tehnică des utilizată în telecomunicații, care permite unui număr mare de utilizatori să partajeze numărul relativ mic de canale dintr-o celulă, oferind acces fiecărui utilizator, la cerere, la un canal disponibil. În acest laborator ne vom concentra pe dimensionarea resurselor folosind formulele Erlang-B și pe impactul sectorizării celulei asupra eficienței trunking-ului într-un sistem celular.

Obiective

- O1. Prezentarea unor concepte legate de traficul celular.
- O2. Prezentarea procedurii de sectorizare a celulei.
- O3. Efectuarea unor sarcini de lucru în Matlab privitoare la simularea modelelor matematice ale fenomenelor sus-amintite.

Durata medie de studiu individual: 30 minute.

9.1 TRAFICUL CELULAR

Pentru analiza traficului celular, se pornește de la premisa unui comportament probabilistic al utilizatorilor de servicii celulare. În acest fel, un număr fix de canale radio poate acomoda un număr mai mare de utilizatori. De aceea trebuie luată în calcul probabilitatea ca un apel să fie blocat / respins, respectiv probabilitatea ca un apel să fie întârziat / plasat în coada de așteptare [1].

În sistemul radio "trunked", fiecărui utilizator i se alocă un canal în momentul unui apel, iar la terminarea convorbirii, canalul ocupat anterior este imediat returnat la grupul de canale disponibile. Măsura intensității traficului poartă numele de Erlang.

Un Erlang reprezintă o intensitate a traficului pe un canal care este complet ocupat (adică 1 oră de apel pe oră sau 1 minut de apel pe minut). De exemplu, un canal radio care este ocupat timp de treizeci de minute într-o oră poartă 0,5 Erlangs de trafic.

Traficul în Erlangs (A) poate fi definit ca [2]:

$$A = \frac{\text{timpul de utilizare al tuturor resurselor}}{\text{timpul total}} \quad (9.1)$$

De exemplu, fie un canal digital cu 30 de circuite vocale. Dacă fiecare circuit a fost ocupat timp de o jumătate de oră într-un interval de o oră, timpul de utilizare al tuturor resurselor este de $30 \text{ min} \times 30 = 900$ minute. Timpul total de referință este de 60 de minute. Astfel, traficul total în Erlang este de $900/60 = 15$ Erlangs.

Intensitatea traficului generată de fiecare utilizator este egală cu rata de solicitare a apelurilor înmulțită cu timpul de așteptare [3]. Adică, fiecare utilizator generează o intensitate de trafic de A_u Erlangs dată de:

$$A_u = \lambda H \quad (9.2)$$

unde H este durata medie a unui apel și λ este numărul mediu de apeluri pe unitatea de timp.

Pentru un sistem cu U utilizatori, intensitatea traficului A este dată de:

$$A = UA_u \quad (9.3)$$

Mai departe, într-un sistem trunked cu C canale, dacă traficul este egal distribuit între canale, intensitatea per canal, A_c , este:

$$A_c = UA_u / C \quad (9.4)$$

9.2 TIPURI DE SISTEME TRUNKED

Există două tipuri de sisteme trunked. Primul tip (sistem de tip Erlang B) nu oferă așteptări pentru solicitările de apel. Adică, fiecărui utilizator care solicită serviciul, i se oferă acces imediat la resurse, dacă există un canal disponibil. Dacă niciun canal nu este disponibil, utilizatorul care solicită resurse este blocat fără acces și trebuie să încerce din nou mai târziu.

Parametrul *Grade of Service* (GOS) indică abilitatea unui utilizator de a accesa un sistem de tip „trunked” la o oră de vârf. Gradul de serviciu este direct legat de probabilitatea de blocare. Un grad mai ridicat de serviciu garantat pentru un anumit client înseamnă asigurarea unei probabilități scăzute de blocare în timpul orelor ocupate. GOS este dat de probabilitatea ca un apel să fie blocat sau probabilitatea ca un apel să experimenteze o întârziere mai mare decât un anumit timp de așteptare [1]. Încărcarea sistemului variază foarte mult în funcție de ora din zi și respectiv ziua săptămânii. Obiectivul principal al dimensionării resurselor este să ne asigurăm că sistemul funcționează bine în aceste ore de vârf [2].

Formula *Erlang B* determină probabilitatea ca un apel să fie blocat și este o măsură a gradului de serviciu pentru un sistem care nu oferă coadă de așteptare pentru apelurile blocate. Numărul de canale ocupate este egal cu numărul de utilizatori ocupați, iar probabilitatea de blocare este exprimată ca [3]:

$$P[\text{blocare}] = \frac{\left(\frac{A^C}{C!}\right)}{\sum_{k=0}^C \frac{A^k}{k!}} \quad (9.5)$$

unde C este numărul de canale asociate fiecărui abonat, iar A este traficul sistemului.

Formula este valabilă în următoarele condiții [2]:

1. Numărul de clienți trebuie să fie de cel puțin 10 ori mai mare decât numărul de resurse disponibile.
2. Cererile de servicii sunt independente unele de celelalte. Această formulă nu este valabilă dacă solicitările abonaților au fost determinate de un eveniment neobișnuit, cum ar fi apelările din cadrul unui talk show, festivaluri etc.
3. Cererile de servicii ale abonaților sunt respinse numai atunci când nu sunt disponibile resurse pentru a le deservi.
4. Atunci când un client nu poate fi deservit, cererea este pur și simplu respinsă iar clientul nu reîncearcă solicitarea serviciului după ce acesta i-a fost refuzat.

Al doilea tip de sistem trunked (sistem de tip Erlang C) prevede o coadă de așteptare pentru a reține apelurile care sunt blocate. Dacă un canal nu este disponibil imediat, cererea de apel poate fi amânată până când un canal devine disponibil.

Formula *Erlang C* este utilizată atunci când se implementează o coadă de așteptare pentru a reține toate apelurile solicitate cărora nu li se poate atribui imediat un canal [2].

Formula Erlang C este dată de:

$$P[\text{delay}] = \frac{A^C}{A^C + C! \left(1 - \frac{A}{C}\right) \sum_{k=0}^C \frac{A^k}{k!}} \quad (9.6)$$

La fel ca și în cazul de mai sus (Erlang B), numărul de clienți trebuie să fie de cel puțin 10 ori mai mare decât numărul de resurse disponibile.

Sarcina 1

Folosindu-vă de funcția [inverlangb](#), determinați câți utilizatori pot fi deserviți într-un sistem cu probabilitate de blocare de 0.5% pentru următorul număr de canale: (a) 10, (b) 20, (c) 99. Considerați că fiecare utilizator generează 0.1 Erlangs de trafic.

Sarcina 2

Trei rețele mobile trunked (sistemele A, B și C) oferă servicii celulare într-o zonă populată de 2 milioane de locuitori. Sistemul A are 394 de celule cu 19 canale fiecare, sistemul B are 98 de celule cu 57 de canale fiecare, iar sistemul C are 49 de celule, fiecare cu câte 100 de canale.

Pornind de la scriptul [L9 Comparatie Sisteme.m](#), găsiți numărul de utilizatori care pot fi acomodați de sistem la o probabilitate de blocare de 2% dacă fiecare utilizator efectuează în medie 2 apeluri pe oră la o durată medie a apelurilor de 3 minute.

Sarcina 3

Fie un oraș cu o suprafață de 1300 km^2 . Acesta este acoperit cu servicii mobile de către un sistem celular care utilizează un model de reutilizare a frecvențelor cu 7 celule. Fiecare celulă are o rază de 4 km și orașului îi este alocat un spectru de 40 MHz cu o lățime de bandă a canalului duplex de 200 kHz. Să presupunem că este prevăzut un GOS de 2% pentru un sistem Erlang B.

Dacă fiecare utilizator efectuează în medie 3 apeluri pe oră cu o durată medie a apelurilor de 1 minut, folosindu-vă de scriptul [L9 Analiza Sistem](#), determinați: (a) numărul de celule din zona de serviciu, (b) numărul de canale pe celulă, (c) intensitatea traficului fiecărei celule, (d) traficul total din sistem, (e) numărul total de utilizatori care pot fi deserviți pentru 2% GOS, (f) numărul total de utilizatori care poate fi deservit simultan de către sistem.

Sarcina 4

Fie un sistem celular în care un apel durează în medie 2 minute, iar probabilitatea de blocare este de maxim 1%. Se presupune că fiecare utilizator realizează în medie 1 apel/h. Sistemul este descris de distribuția Erlang B, cu blocare. În total operatorul de servicii dispune de 395 de canale de trafic, iar dimensiunea cluster-ului este de 7 celule. Pornind de la scriptul [L9 Capacitate Sectorizare.m](#). Determinați: (a) Intensitatea traficului, numărul de apeluri/h, numărul de utilizatori din sistem și parametrul SIR pentru sistemul ne-sectorizat, (b) Intensitatea traficului, numărul de apeluri/h, numărul de utilizatori din sistem și parametrul SIR pentru sistemul sectorizat la 120° , (c) Intensitatea traficului, numărul de apeluri/h, numărul de utilizatori din sistem și parametrul SIR pentru sistemul sectorizat la 60° .

Bibliografie

- [1] William C.Y. Lee, *Wireless and Cellular Telecommunications*, 3rd edition, McGraw-Hill, 2006.
- [2] T. Rappaport, *Wireless communications, principles and practice*, Prentice Hall, 2002.
- [3] Vijay Kumar Garg, *Wireless Communications and Networking*, Morgan Kaufmann Publishers, 2007.

SCRIPT-URI MATLAB

inverlangb.m

```

function A = inverlangb(N, B)
%% =====
%      INVERLANGB - Inversa formulei Erlang B
%  A - intensitatea traficului (Erlang)
%  N - numărul de canale
%  B - probabilitatea de blocare dorită (Grade of Service)
if (length(N)~=1) || (fix(N) ~= N) || (N < 0)
    error('N trebuie să fie un întreg pozitiv.');
```

end

```

if (length(B)~=1) || (B < 0)
    error('B trebuie să fie o valoare scalară și pozitivă.');
```

end

```

if (B == 0) || (N == 0)
    A = 0;
    return;
end

An = N;
fn = erlangb(N, An) - B;

for k = 1:1000
    fprimen = erlangbprime(N, An);

    if fprimen == 0
        error('Derivata este zero - metoda nu poate continua.');
```

end

```

% Formula Newton-Raphson
An_next = An - fn/fprimen;

% Condiție de convergență
if abs(An_next - An) < 10*eps
    break;
end

% Evitarea valorilor negative
if An_next > 0
    An = An_next;
else
    % Regenerăm o valoare pozitivă aleatoare
    An = N * rand(1,1);
end

fn = erlangb(N, An) - B;

if k == 1000
    error('Nu există convergență după 1000 de iterații.');
```

end

A = An_next; % intensitatea traficului determinată

end

erlangb.m

```

function B = erlangb(N, A)
%% =====
%   ERLANGB - Probabilitatea de blocare
%
%   N - număr de canale
%   A - intensitatea traficului (Erlang)
%   Reprezintă probabilitatea ca un apel nou să fie blocat
%   într-un sistem fără coadă de așteptare.
%% =====

if (length(N)~=1) || (fix(N) ~= N) || (N < 0)
    error('N trebuie să fie un întreg pozitiv.');
```

end

esum = zeros(size(A));

for i = 0:N

 esum = esum + A.^i ./ factorial(i);

end

B = A.^N ./ (factorial(N) .* esum);

end

function eprime = erlangbprime(N, A)

%% =====

% ERLANGBPRIME - Derivata formulei Erlang-B după A

%% =====

if (length(N)~=1) || (fix(N) ~= N) || (N < 1)

 error('N trebuie să fie un întreg pozitiv.');

end

u = A.^N;

uprime = N * A.^(N-1);

v = zeros(size(A));

for i = 0:N

 v = v + A.^i ./ factorial(i);

end

vprime = zeros(size(A));

for i = 1:N

 vprime = vprime + i .* A.^(i-1) ./ factorial(i);

end

eprime = (v .* uprime - u .* vprime) ./ (factorial(N) .* v.^2);

end

L9 Comparatie Sisteme.m

```

clear; clc;

%% ===== Date de intrare =====
GOS = 0.02; % Probabilitatea de blocare (GoS)
Nr_Canale_A = ?;
Nr_Canale_B = ?;
Nr_Canale_C = ?;

Nr_Celule_A = ?;
Nr_Celule_B = ?;
Nr_Celule_C = ?;

Populatie = ?;
lambda = ?; % Nr. apeluri efectuate per utilizator / oră
H = ?; % Durata medie a unui apel [ore]

A_u = ? * H; % Trafic oferit per utilizator (Erlang)

fprintf('\nTrafic per utilizator: %.4f Erlang\n', A_u);

%% ===== Sistemul A (394 celule x 19 canale) =====
A_A = inverlangb(?, GOS); % Erlang suportat per celulă
U_A = (A_u / A_A) * ?;

fprintf('\n===== SISTEMUL A =====\n');
fprintf('Canale per celula: %d\n', Nr_Canale_A);
fprintf('Trafic maxim suportat per celula: %.2f Erlang\n', A_A);
fprintf('Numar total utilizatori suportati: %.0f\n', U_A);

%% ===== Sistemul B (98 celule x 57 canale) =====
A_B = inverlangb(?, ?);
U_B = (A_u / A_B) * ?;

fprintf('\n===== SISTEMUL B =====\n');
fprintf('Canale per celula: %d\n', Nr_Canale_B);
fprintf('Trafic maxim suportat per celula: %.2f Erlang\n', A_B);
fprintf('Numar total utilizatori suportati: %.0f\n', U_B);

%% ===== Sistemul C (49 celule x 100 canale) =====
A_C = inverlangb(?, ?);
U_C = (A_u / A_C) * ?;

fprintf('\n===== SISTEMUL C =====\n');
fprintf('Canale per celula: %d\n', Nr_Canale_C);
fprintf('Trafic maxim suportat per celula: %.2f Erlang\n', A_C);
fprintf('Numar total utilizatori suportati: %.0f\n', U_C);

%% ===== Total sistem mobil urban =====
U_TOTAL = ? + U_B + ?;

fprintf('\n===== \n');
fprintf('Capacitatea totală a rețelelor A+B+C: %.0f utilizatori\n', U_TOTAL);
fprintf('Populatia urbana: %.0f\n', Populatie);
fprintf('Gradul de penetrare în populatie: %.1f %%\n', 100 * U_TOTAL / Populatie);
fprintf('===== \n\n');

```

L9 Analiza Sistem.m

```

clc; clear;
%% === 1. Parametrii sistemului
=====
S_totala    = ?;          % Suprafața totală care trebuie acoperită [km^2]
N           = ?;          % Factorul de reutilizare a frecvenței (cluster size)
R           = ?;          % Raza unei celule [km]
bw_canal    = ?;          % Lățimea de bandă alocată unui canal RF [Hz]
bw_totala   = ?;          % Lățimea de bandă totală a sistemului [Hz]
GOS         = ?;          % Grade of Service dorit (prob. de blocare = 2%)
lambda      = ?;          % Rata medie de inițiere a apelurilor / utilizator
H           = ?;          % Durata medie a unui apel (3 minute ≈ 0.05 ore)

A_u = ? * ?;
% Traficul generat de un utilizator (trafic erlang / utilizator)

%% === 2. Numărul de celule necesare în sistem
=====
% Aria unei celule pentru o rețea hexagonală:
S_celula    = 2.6 * R^2;

Nr_celule   = floor(? / ?);
% Numărul de celule necesare pentru a acoperi zona

fprintf('Număr celule în sistem: %d\n', ?);

%% === 3. Numărul de canale disponibile pe celulă
=====
Nr_canale_celula = floor(? / (? * N));

fprintf('Număr canale pe celulă: %d\n', Nr_canale_celula);

%% === 4. Intensitatea maximă de trafic admisă / celulă
=====
A_celula     = floor(inverlangb(? , GOS));

fprintf('Trafic maxim pe celulă: %d Erlang\n', ?);

%% === 5. Traficul total al sistemului
=====
A_sistem     = ? * ?;

fprintf('Trafic total al sistemului: %d Erlang\n', ?);

%% === 6. Numărul total de utilizatori deserviți
=====
U = ? / ?;
% Numărul maxim de utilizatori care pot fi susținuți în sistem

fprintf('Număr total utilizatori în sistem: %.0f\n', U);

%% === 7. Numărul de utilizatori simultan în apel
=====
U_simultan   = ? * Nr_celule;
% Fiecare canal RF poate deservi un singur utilizator (în FDMA/TDMA clasic)

fprintf('Număr maxim utilizatori simultan: %d\n', U_simultan);

```

L9 Capacitate Sectorizare.m

```

clearvars; clc;

%% =====
% 1. PARAMETRI COMUNI
% =====

GOS      = ?;      % probabilitate de blocare (Grade of Service)
H        = ?;      % durata medie a apelului [ore]
lambda_user = ?;    % apeluri / oră
A_u      = ? * ?;  % trafic generat de un utilizator [Erlangs]

C_total  = ?;      % număr total de canale de trafic
N_cluster = ?;     % dimensiunea clusterului
i0       = ?;      % număr surse de interferența co-canal
n        = ?;      % exponentul pierderilor

fprintf("===== \n");
fprintf("  Calcul capacitate celulară și impactul sectorizării\n");
fprintf("===== \n\n");
;

%% =====
% 2. SISTEM NE-SECTORIZAT (OMNI)
% =====

channels_per_cell = floor(? / ?);
A_cell           = inverlangb(channels_per_cell, GOS); % Erlangs per celulă
calls_per_hour_cell = A_cell / H;
users_per_cell    = floor(A_cell / A_u);

% Calcul SIR
SIR_lin_ns = (sqrt(3 * N_cluster))^n / i0;
SIR_dB_ns  = 10 * log10(SIR_lin_ns);

%% =====
% 3. SISTEM SECTORIZAT 120° (3 sectoare)
% =====

sectors_120      = 3;
channels_per_sector_120 = floor(channels_per_cell / ?);

A_sector_120      = inverlangb(channels_per_sector_120, GOS);
calls_per_hour_sector120 = ? / ?;
users_per_sector120    = floor(? / A_u);

% Apeluri per celulă (3 sectoare)
calls_per_hour_cell120 = ? * sectors_120;
users_per_cell120      = users_per_sector120 * ?;

% Interferență redusă prin sectorizare
i_eff_120 = ?;
SIR_lin_120 = (sqrt(3 * N_cluster))^n / i_eff_120;
SIR_dB_120  = 10 * log10(SIR_lin_120);

```

```

%% =====
% 4. SISTEM SECTORIZAT 60° (6 sectoare)
% =====

sectors_60          = 6;
channels_per_sector_60 = floor(? / ?);

A_sector_60         = inverlangb(? , GOS);
calls_per_hour_sector60 = ? / H;
users_per_sector60   = floor(? / ?);

% Apeluri per celulă (6 sectoare)
calls_per_hour_cell60 = ? * sectors_60;
users_per_cell60      = users_per_sector60 * ?;

% Interferență redusă prin sectorizare
i_eff_60 = ?;
SIR_lin_60 = (sqrt(3 * ?))^n / ?;
SIR_dB_60 = 10 * log10(?);

%% =====
% 5. AFIȘAREA REZULTATELOR
% =====

fprintf("PARAMETRI GENERALI:\n");
fprintf("  Total canale disponibile: %d\n", C_total);
fprintf("  Mărime cluster: N = %d\n", N_cluster);
fprintf("  Trafic per utilizator: A_u = %.4f Erlangs\n\n", A_u);

fprintf("1) SISTEM NE-SECTORIZAT (OMNI):\n");
fprintf("  Canale per celulă          = %d\n", channels_per_cell);
fprintf("  Trafic per celulă (Erlangs) = %.4f\n", A_cell);
fprintf("  Apeluri/oră per celulă     = %.2f\n", calls_per_hour_cell);
fprintf("  Utilizatori susținuți per celulă = %d\n", users_per_cell);
fprintf("  SIR = %.3f (linear), %.2f dB\n\n", SIR_lin_ns, SIR_dB_ns);

fprintf("2) SISTEM SECTORIZAT 120° (3 sectoare):\n");
fprintf("  Canale per sector          = %d\n", channels_per_sector_120);
fprintf("  Trafic per sector (Erlangs) = %.4f\n", A_sector_120);
fprintf("  Apeluri/oră per sector     = %.2f\n",
calls_per_hour_sector120);
fprintf("  Apeluri/oră per celulă     = %.2f\n", calls_per_hour_cell120);
fprintf("  Utilizatori per sector     = %d\n", users_per_sector120);
fprintf("  Utilizatori per celulă     = %d\n", users_per_cell120);
fprintf("  Nr. Surse interferenta i_eff = %.2f\n", i_eff_120);
fprintf("  SIR = %.3f (linear), %.2f dB\n\n", SIR_lin_120, SIR_dB_120);

fprintf("3) SISTEM SECTORIZAT 60° (6 sectoare):\n");
fprintf("  Canale per sector          = %d\n", channels_per_sector_60);
fprintf("  Trafic per sector (Erlangs) = %.4f\n", A_sector_60);
fprintf("  Apeluri/oră per sector     = %.2f\n",
calls_per_hour_sector60);
fprintf("  Apeluri/oră per celulă     = %.2f\n", calls_per_hour_cell60);
fprintf("  Utilizatori per sector     = %d\n", users_per_sector60);
fprintf("  Utilizatori per celulă     = %d\n", users_per_cell60);
fprintf("  Nr. Surse interferenta i_eff = %.2f\n", i_eff_60);
fprintf("  SIR = %.3f (linear), %.2f dB\n\n", SIR_lin_60, SIR_dB_60);

```

FIȘĂ DE LUCRU**Sarcina 1**

Rezultate obținute:

a)

b)

c)

Sarcina 2

Script realizat și rezultate obținute:

Sarcina 3

Script realizat și rezultate obținute:

Sarcina 4

Script realizat și rezultate obținute:

Lucrarea 10 – Organizarea resurselor radio în GSM

Acest laborator are ca obiectiv descrierea sistemului GSM la nivelul organizării resurselor radio prin studierea ferestrelor de transmisie, a cadrului temporal de transmisie și a salvelor de comunicație utilizate. Vor fi prezentate ierarhia cadrelor GSM, structura acestora și tipurile de salve de comunicație implementate în standard (salvă normală, de acces, de corecție a frecvenței și de sincronizare).

Obiective

- O1. Prezentarea ierarhiei și structurii cadrelor GSM.
- O2. Calculul eficienței spectrale a cadrului TDMA.
- O3. Efectuarea unor sarcini de lucru în Matlab privitoare la simularea modelelor matematice ale fenomenelor sus-amintite.

Durata medie de studiu individual: 30 minute.

10.1 SCHEMELE DE ACCES MULTIPLU ÎN GSM

Standardul GSM a fost conceput inițial pentru banda de frecvență de 900 MHz. În această bandă, au fost rezervate inițial două sub-benzi de câte 25 MHz, și anume [935 MHz - 960 MHz] pentru sensul de transmisie de la stația de bază către mobil (DL - downlink) și [890 MHz - 915 MHz] pentru sensul de transmisie de la mobil către stația de bază (UL - uplink). Astfel, este implementată tehnica FDD (Frequency Division Duplexing). În fiecare bandă sunt definite 125 de canale radio a câte 200 KHz, numerotate de la 0 la 124, canalul „0” fiind neutilizat pentru a se evita interferența cu sistemele ce folosesc banda imediat inferioară celei alocate sistemului GSM [1].

Pe fiecare canal radio, standardul GSM definește ferestre de timp, sau time-slot-uri (TS) grupate în cadre de câte 8 TS-uri și numerotate de la 0 la 7, utilizând principiul TDMA (Time Division Multiple Access) pentru a permite unui număr de până la opt utilizatori să folosească același canal radio (Figura 10.1). Durata unui TS este de 0,577 ms, timp în care se transmit 156,25 biți. Rezultă astfel un debit pe canal de aproximativ 271 Kb/s. Aceste cadre TDMA sunt integrate în structuri ierarhice de supercadre și hipercadre, care permit rețelei să gestioneze eficient nu doar datele de voce sau utilizator transmise prin canalele de trafic (TCH), ci și informațiile de semnalizare, sincronizare și alocare a resurselor prin canalele de control (CCH) [1].

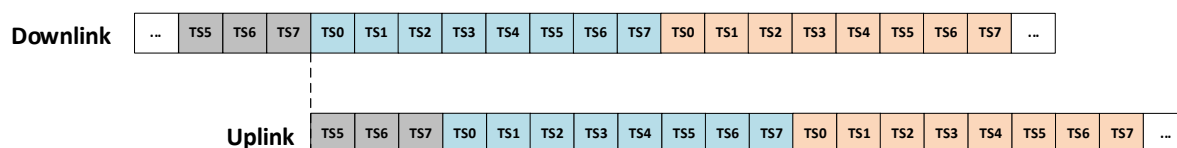


Figura 10.1 Decalarea numerotării ferestrelor pe cele două sensuri de comunicație.

Time slot-urile definite pe canalele radio sunt sincronizate în timp. Un terminal mobil utilizează TS-ul cu același număr de ordine din fiecare cadru, valabil pentru ambele sensuri de comunicație. Pentru a preveni situația în care mobilul ar trebui să emită și să recepționeze simultan, numerotarea time slot-urilor de pe UL este implementată cu un decalaj de trei poziții față de cea de pe DL.

Fiecare TS deține un număr de ordine care reflectă poziția sa în cadrul ierarhiei cadrelor TDMA: poziția TS-ului în cadrul TDMA, a cadrului în multicadru, a multicadrului în supercadru și, în final, a acestuia din urmă în hipercadru (Figura 10.2). Deoarece un hipercadru cuprinde 21.725.184 TS-uri și durează aproximativ trei ore și jumătate, este extrem de puțin probabil ca un utilizator să refolosească un TS-ul cu același număr de ordine pe parcursul unei sesiuni de comunicație. Prin urmare, acest număr unic de ordine al TS-ului de emisie este

utilizat ca dată de intrare în blocul de criptare a informației transmise. Efectul direct este că secvențele identice de biți, transmise în momente diferite pe parcursul aceleiași convorbiri, sunt criptate de fiecare dată în mod diferit, sporind astfel securitatea transferului de date [2].

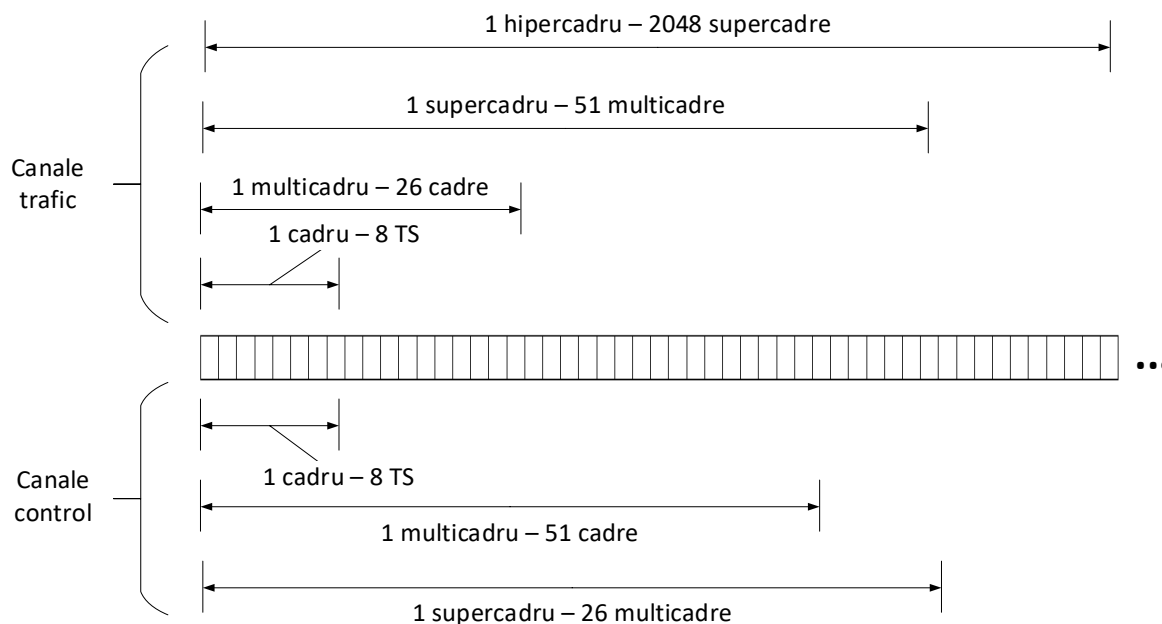


Figura 10.2 Organizarea cadrelor în sistemul GSM.

10.2 SALVELE DE COMUNICAȚIE

Datele de trafic, împreună cu mesajele de semnalizare transmise în cadrul unui TS, alcătuiesc o salvă (burst) de comunicație. Există 4 tipuri de salve de comunicație care vor fi descrise mai jos [2].

a) Salva normală

Salva normală este tipul de salvă utilizat cel mai des în timpul unei conexiuni GSM. Aceasta începe și se încheie cu câte trei biți (tail bits) de valoare zero, numiți tail bits, care formează o „zonă tampon” menită să preîntâmpine fenomenul de tranziție al puterii de emisie către și de la valoarea puterii nominale. Secvența de început mai este folosită și pentru inițializarea demodulatorului de la nivelul receptorului.

Informația utilă a utilizatorului, după trecerea prin codarea corectoare de erori, este împărțită în două blocuri de câte 57 de biți (blocurile D), separate de o secvență de antrenare de 26 de biți (A) și doi biți de semnalizare (S). Prin analiza abaterilor dintre secvența de antrenare recepționată și forma ei ideală, receptorul poate estima răspunsul la impuls al canalului radio și ajusta egalizorul pentru recuperarea corectă a datelor. Biții S indică dacă salva conține date de trafic sau de semnalizare.

La final, intervalul de gardă (G) permite reducerea treptată a puterii de emisie, prevenind ca transmisiile terminalelor să se suprapună la nivelul stației de bază. Terminalele care urmează să transmită cresc treptat puterea către nivelul nominal, în timp ce cele care finalizează transmisia o reduc progresiv.

T 3 biți	D 57 biți	S 1bit	A 26 biți	S 1bit	D 57 biți	T 3 biți	GP 8.25 biți
--------------------	---------------------	------------------	---------------------	------------------	---------------------	--------------------	------------------------

Figura 10.3 Salva normală.

b) Salva de acces

Salva de acces este prima salvă transmisă de către un terminal mobil către stația de bază atunci când dorește să inițieze o comunicație. Pentru că la acest moment întârzierea de propagare nu este încă estimată, salva transmisă este mai scurtă (aproximativ 325 μ s), astfel încât semnalul transmis să nu se suprapună peste alte transmisii. În plus, structura acestei salve diferă de cea a salvei normale prin prezența unor secvențe de capăt (T) mai lungi, a unei secvențe de antrenare (A) extinse și a unui număr redus de biți utili.

T 8 biți	A 41 biți	D 36 biți	T 3 biți	GP 68.25 biți
---------------------------	----------------------------	----------------------------	---------------------------	--------------------------------

Figura 10.4 Salva de acces.

c) Salva pentru corecția frecvenței

Salva de corecție a frecvenței este transmisă exclusiv de stațiile de bază și are o structură foarte simplă: între cele două grupuri de biți de capăt, se transmit 142 de biți de „0”. Această secvență determină emiterea unei purtătoare sinusoidale, la frecvența 67,7 kHz față de frecvența centrală a canalului.

Terminalele mobile folosesc această purtătoare pentru a-și calibra propria frecvență de emisie, astfel încât puterea radiată în afara benzii de 200 kHz să rămână în limitele impuse de standard.

T 3 biți	Secvență fixă 142 biți de „0”	T 3 biți	GP 8.25 biți
---------------------------	--	---------------------------	-------------------------------

Figura 10.5 Salva de corecție a frecvenței.

d) Salva de sincronizare

Salva de sincronizare, transmisă periodic de stația de bază, reprezintă prima secvență pe care un terminal mobil trebuie să o recepționeze și să o demoduleze. Prin structura ei, această salvă furnizează mobilului informații legate de identitatea sistemului celular, identitatea stației de bază și numărul cadrului TDMA curent. Față de salva normală, secvența de antrenare este mai lungă, pentru a garanta sincronizarea corectă a receptorului.

T 3 biți	D 39 biți	A 64 biți	D 39 biți	T 3 biți	GP 8.25 biți
---------------------------	----------------------------	----------------------------	----------------------------	---------------------------	-------------------------------

Figura 10.6 Salva de sincronizare.

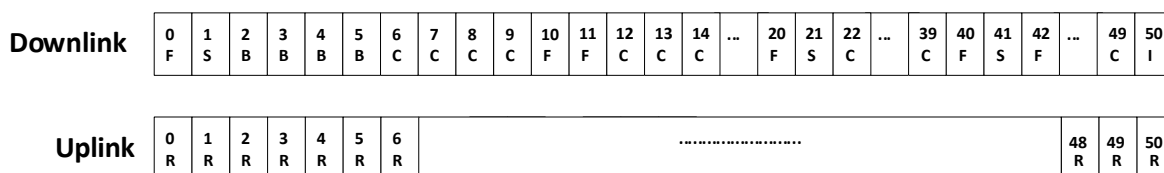
Împărțirea benzii de frecvență alocate GSM în canale radio de câte 200 KHz, urmată de divizarea accesului la resursele radio în TS-uri, constituie interfața fizică pe care se realizează transmiterea datelor. Modul în care aceste TS-uri, repartizate pe diferite frecvențe, sunt combinate pentru a transporta traficul utilizatorilor și informațiile de semnalizare este stabilit în standardul GSM și dă naștere canalelor logice. Canalele logice sunt clasificate în funcție de tipul informației transportate, iar eficiența cu care sunt folosite resursele rețelei depinde de volumul acestor date. Acestea sunt clasificate și descrise pe scurt în Tabelul 10.1.

Tabelul 10.1 Organizarea canalelor logice [2].

Categorie	Tipul canalului	Abreviere	UL/DL	Info. transp.	Tipul salvei
Canal de trafic	Traffic channel	<i>TCH/F</i> , <i>TCH/H</i>	UL&DL	Date utilizator	Normală
Broadcast channels	Broadcast control channel	<i>BCCH</i>	DL	Info. celulă (ex. semnal pilot)	Normală
	Synchronization channel	<i>SCH</i>	DL	Nr. cadru TDMA, BSIC	Sincronizare

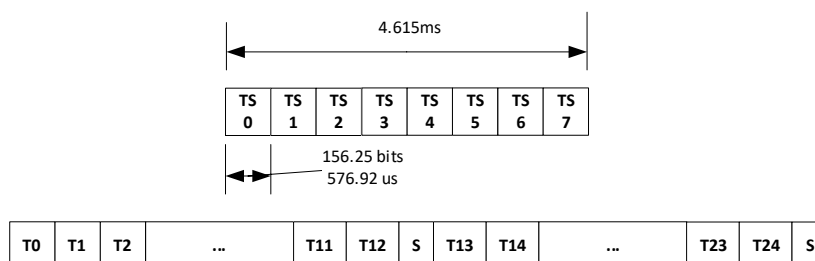
	Frequency correction channel	<i>FCCH</i>	DL	Sincro. frecv.	Corecție frecvență
Dedicated control channels	Standalone control channel	<i>SDCCH</i>	UL&DL	Info. Config. apel	Normală
	Slow associated control channel	<i>SACCH</i>	UL&DL	Info. in paralel cu TCH	Normală
	Fast associated control channel	<i>FACCH</i>	UL&DL	Doar info. critice	Normală/Acces
Common control channels	Random access control channel	<i>RACH</i>	UL	Cerere alocare SDCCH	Acces
	Paging channel	<i>PCH</i>	DL	Semnalizare apel intrare	Normală
	Access grant channel	<i>AGCH</i>	DL	Indicare alocare SDCCH	Normală

Sucesiunea canalelor logice în cadrul ferestrei temporale TS_0 de pe canalul de beacon al BS este prezentată în Figura 10.7 și Figura 10.8. Se poate remarca diferența dintre canalele de control și cele de trafic. Astfel, Figura 10.7 ilustrează multicadrul de control (format din 51 de cadre), care multiplexează canalele FCCH, SCH, BCCH și CCCH. Mai departe, Figura 10.8 detaliază structura multicadrului de trafic (format din 26 de cadre), care transportă datele utilizatorului (TCH), transmițând periodic SACCH și un cadru de tip „Idle”.



F: salvă FCCH (BCH)
 S: salvă SCH (BCH)
 B: salvă BCCH (BCH)
 C: salvă PCH/AGCH (CCCH)
 I: idle
 R: salvă RACH (CCCH)

Figura 10.7 Maparea canalelor de control pe multicadrul de control pe DL, respectiv UL.



TS_n : al n-lea time slot
 T_n : al n-lea cadru TCH
 S: cadru SACCH
 I: cadru idle

Figura 10.8 Maparea canalelor de trafic pe multicadrul de trafic pe DL, respectiv UL.

Sarcina 1

Dacă structura cadrelor utilizată în GSM constă în 8 TS, iar fiecare TS conține 156.25 bits, iar datele sunt transmise la 270.833 kbps pe canal, aflați (a) durata unui bit, (b) durata unui TS, (c) durata unui cadru TDMA GSM, (d) determinați intervalul de gardă pentru salvele de comunicații din GSM, (e) luând în calcul structura salvei normale de comunicație, determinați eficiența cadrului TDMA GSM format din 8 astfel de salve.

Sarcina 2

Formați salvele de acces GSM din scripturile [*accessBurstDescription.m*](#), [*frequencyCorrectionBurstDescription.m*](#), [*normalBurstDescription.m*](#), [*synchronizationBurstDescription.m*](#) conform descrierii din figurile Figura 10.3-Figura 10.6. Adăugați în [*Fișa de lucru*](#) configurația afișată în linia de comandă în urma rulării script-urilor.

Sarcina 3

Pornind de la scriptul [*L10_DL_Frame.m*](#), configurați un cadru GSM TDMA pe legătura DL folosind obiectul *gsmDownlinkConfig*. Transmiteți în TS₀ și TS₁ salva de corecție a frecvenței, transmiteți în TS₂ și TS₃ salva de sincronizare, iar în restul TS-urilor transmiteți salve normale. Setări atenuarea pentru TS₄ la valoarea aferentă propagării în spațiu liber a semnalului dintre MS și BS pentru distanța de 2 km la 935.2 MHz.

- Reprezentați grafic cadrul TDMA obținut.
- Determinați prin măsurători pe grafic durata perioadei de gardă pentru salva de sincronizare.
- Determinați prin măsurători pe grafic durata unui time-slot.

Pornind de la scriptul [*L10_UL_Frame.m*](#), configurați apoi un cadru GSM TDMA pe legătură UL folosind obiectul *gsmUplinkConfig*. Transmiteți în TS₀ și TS₁ salve de acces, iar în TS₂, TS₃, și TS₄ salve normale. Restul TS-urilor nu transmiteți nimic. Setări atenuarea pentru TS₄ la valoarea aferentă propagării în spațiu liber a semnalului dintre MS și BS pentru distanța de 2 km la 890.2 MHz.

- Reprezentați grafic cadrul TDMA obținut.
- Determinați prin măsurători pe grafic durata perioadei de gardă pentru salva de acces.
- Determinați prin măsurători pe grafic durata unui cadru TDMA.

Adăugați rezultatele în [*Fișa de lucru*](#).

Obs. *Normal Burst* („NB”), *Access Burst* („AB”), *Frequency Correction Burst* („FB”), *Synchronization Burst* („SB”)

Sarcina 4

Pornind de la scriptul [*L10_Mapping_CCH.m*](#) creați o structură multicadru format din primele 10 cadre de control DL din cele 51 de cadre din multicadru de control, luând ca referință maparea canalelor de control din Figura 10.7. Reprezentați structura de cadre TDMA obținută și adăugați-o în [*Fișa de lucru*](#).

Sarcina 5

În sistemul celular digital GSM800, lățimea de bandă UL/DL a sistemului este de 12.5 MHz. Lățimea de bandă a unui canal este de 200 kHz. Opt utilizatori partajează fiecare canal RF și trei canale per celulă sunt ca și canale de control. Considerați că suprafața unei celule este de 8 km², suprafața totală de acoperire cu servicii celulare este de 4000 km², numărul mediu de apeluri per utilizator (λ) este de 1.2, durata medie a unui apel (H) este de 100 de secunde, probabilitatea de blocare a apelurilor (p_B) este 2%, și se utilizează clustere de câte 4 celule.

Determinați următorii indicatori de eficiență ai sistemului pornind de la script-ul [L10_Performanta_Sistem.m](#): (a) numărul de apeluri/oră/celulă, (b) numărul de utilizatori/celulă/oră, și (c) eficiența spectrală a modulării (Erlangs/MHz/km²).

Bibliografie

- [1] Vijay K. Garg, Joseph E. Wilkes, *Wireless and Personal Communications Systems*, Prentice-Hall, 1996.
- [2] Ion Bogdan, *Comunicații Mobile*, Tehnopress, ISBN 973-8377-59-5, 292, Iași, 2003.
- [3] 3GPP TS 45.005. "GSM/EDGE Radio transmission and reception." 3rd Generation Partnership Project; Technical Specification Group Radio Access Network.

SCRIPT-URI MATLAB

Funcțiile asociate salvelor de comunicații

```

function d = normalBurstDescription()
BitNumber = ["0 - 2";"3 - ?";"? - ?";...
            "? - ?";"? - ?";"? - 156"];
LengthOfField = {3;58;26;58;3;8.25};
ContentsOfField = [...
    "tail bits";...
    "encrypted bits";...
    "training sequence bits";...
    "encrypted bits";...
    "tail bits";...
    "guard period (bits)"...
    ];
d = table(BitNumber,LengthOfField,ContentsOfField);
end
function d = accessBurstDescription()
BitNumber = ["0 - 7";"8 - ?";"? - ?";...
            "? - ?";"? - 156"];
LengthOfField = {8;41;36;3;68.25};
ContentsOfField = [...
    "extended tail bits";...
    "synch. sequence bits";...
    "encrypted bits";...
    "tail bits";...
    "guard period (bits)"...
    ];
d = table(BitNumber,LengthOfField,ContentsOfField);
end
function d = frequencyCorrectionBurstDescription()
BitNumber = ["0 - ?";"? - ?";"? - ?";"? - 156"];
LengthOfField = {3;142;3;8.25};
ContentsOfField = [...
    "tail bits";...
    "fixed bits";...
    "tail bits";...
    "guard period (bits)"...
    ];
d = table(BitNumber,LengthOfField,ContentsOfField);
end
function d = synchronizationBurstDescription()
BitNumber = ["0 - ?";"? - ?";"? - ?";...
            "? - ?";"? - ?";"? - 156"];
LengthOfField = {3;39;64;39;3;8.25};
ContentsOfField = [...
    "tail bits";...
    "encrypted bits";...
    "extended training sequence bits";...
    "encrypted bits";...
    "tail bits";...
    "guard period (bits)"...
    ];
d = table(BitNumber,LengthOfField,ContentsOfField);
end

```

L10 DL Frame.m

```

clc;
clearvars;
close all;

%% === Pasul 1: Crearea obiectului de configurare GSM downlink ===
cfg = gsmDownlinkConfig();

%% === Pasul 2: Configurați tipurile de salvă ===

% "NB" - Normal Burst
% "AB" - Access Burst
% "FB" - Frequency Correction Burst
% "SB" - Synchronization Burst

cfg.BurstType([? ?]+1)      = "?"; % prima salvă
cfg.BurstType([? ?]+1)      = "?"; % a doua salvă
cfg.BurstType([? ? ?]+1)    = "?"; % a treia salvă

%% === Pasul 3: Configurați atenuarea pentru fiecare salvă (dB) ===
cfg.Attenuation([? ? ?]+1) = [ ? ? ? ];

%% === Pasul 4: Generarea cadrului GSM ===
x = gsmFrame(cfg);

%% === Pasul 5: Extracția informațiilor despre semnal ===
wfInfo = gsmInfo(cfg);
Rs = wfInfo.SampleRate;
t = (0:length(x) - 1) / Rs * 1e3; % axă temporală [ms]

%% === Pasul 6: Reprezentarea cadrului TDMA ===
subplot(2,1,1)
plot(t, abs(x), 'LineWidth', 1.2)
grid on
axis([0 5 0 1.2])
title('GSM Downlink TDMA Frame')
xlabel('Time (ms)')
ylabel('Amplitude')

%% === Pasul 7: Spectrograma cadrului ===
subplot(2,1,2)
spectrogram(x, 500, [], [], Rs, 'centered')
title('GSM Downlink TDMA Frame - Spectrogram')
grid on

```

L10 UL Frame.m

```

clc;
clearvars;
close all;

%% === Pasul 1: Crearea obiectului de configurare GSM uplink ===
cfg = gsmUplinkConfig();

%% === Pasul 2: Configurați tipurile de burst ===
% Exemple de tipuri valide:
% "NB" - Normal Burst
% "AB" - Access Burst
% "FB" - Frequency Correction Burst
% "SB" - Synchronization Burst

cfg.BurstType([? ?]+1)      = "?"; % prima salvă
cfg.BurstType([? ?]+1)      = "?"; % a doua salvă
cfg.BurstType([? ? ? ?]+1)  = "?"; % a treia salvă

%% === Pasul 3: Configurați atenuarea pentru fiecare salvă (dB) ===

cfg.Attenuation([? ? ? ?]+1) = [ ? ? ? ? ];

%% === Pasul 4: Generarea cadrului GSM ===
x = gsmFrame(cfg);

%% === Pasul 5: Extracția ratei de eșantionare ===
wfInfo = gsmInfo(cfg);
Rs = wfInfo.SampleRate;

t = (0:length(x) - 1) / Rs * 1e3; % timp în milisecunde

%% === Pasul 6: Reprezentarea amplitudinii semnalului ===
subplot(2,1,1)
plot(t, abs(x), 'LineWidth', 1.2)
grid on
axis([0 5 0 1.2])
title('GSM Uplink TDMA Frame')
xlabel('Timp (ms)')
ylabel('Amplitudine')

%% === Pasul 7: Reprezentarea fazei semnalului ===
subplot(2,1,2)
spectrogram(x, 500, [], [], Rs, 'centered')
title('GSM Uplink TDMA Frame - Spectrogram')
grid on

```

L10 Mapping CCH.m

```

clc; clearvars; close all;

%% === Pasul 1: Configurarea cadrelor individuale ===

cfg = gsmDownlinkConfig;

% Cadru FCCH (1 x FB + 7 x NB)
cfg1 = gsmDownlinkConfig('BurstType', ["?", "?", "?", "?", "?", "?", "?", "?"]);
cfg1.Attenuation([7] + 1) = [3];

% Cadru SCH (1 x SB + 7 x NB)
cfg2 = gsmDownlinkConfig('BurstType', ["?", "?", "?", "?", "?", "?", "?", "?"]);
cfg2.Attenuation([7] + 1) = [3];

% Cadru BCCH (8 x NB)
cfg3 = gsmDownlinkConfig('BurstType', ["?", "?", "?", "?", "?", "?", "?", "?"]);
cfg3.Attenuation([7] + 1) = [3];

% Cadru CCH (8 x NB)
cfg4 = gsmDownlinkConfig('BurstType', ["?", "?", "?", "?", "?", "?", "?", "?"]);
cfg4.Attenuation([7] + 1) = [3];

%% === Pasul 2: Determinarea lungimii unui cadru ===
wfInfo = gsmInfo(cfg);
frameLength = wfInfo.FrameLengthInSamples;

%% === Pasul 3: Alocare buffer pentru 10 cadre ===
x = zeros(frameLength*10, 1);

%% === Pasul 4: Generarea fiecărui cadru în parte ===
x1 = gsmFrame(cfg1, 1); % 1 x FCCH
x2 = gsmFrame(cfg2, ?); % 1 x SCH
x3 = gsmFrame(cfg3, ?); % 4 x BCCH
x4 = gsmFrame(cfg4, ?); % 4 x CCH

%% === Pasul 5: Concatenarea în ordinea standard: FCCH → SCH → BCCH → CCH ===
x = [?; ?; ?; ?];

%% =====
%                               VIZUALIZAREA MULTICADRULUI GSM
% =====

wfInfo = gsmInfo(cfg);
Rs = wfInfo.SampleRate;

t = (0:length(x)-1) / Rs * 1e3; % axa temporală în milisecunde

figure;
plot(t, abs(x), "LineWidth", 1.2);
grid on;
title('GSM Multiframe ');
xlabel('Timp (ms)');
ylabel('Putere (dB)');

```

L10 Performanta Sistem.m

```

clc; clearvars;

%% ===== 1. Date din enunț =====
BW_total = ?;           % lățime de bandă totală (Hz)
BW_canal = ?;           % lățime de bandă canale RF (Hz)
utilizatori_per_RF = ?; % 8 utilizatori per canal RF (TDMA)
canale_control = ?;     % canale de control per celulă
Pb = ?;                 % probabilitate de blocare 2%
apeluri_BH = ?;         % apeluri per utilizator în ora de vârf
T_apel = ?;             % durata medie apel [s]
cluster = ?;           % factor de reutilizare frecvență
arie_celula = ?;        % [km^2]
arie_totala = ?;        % [km^2]

%% ===== 2. Număr canale RF =====
numar_canal_RF = floor(? / ?);

%% ===== 3. Canale de trafic în total=====
canale_trafic_total = numar_canal_RF * ?;

%% ===== 4. Canale de trafic per celulă =====
canale_trafic_celula = (canale_trafic_total / ?) - ?;

%% ===== 5. Calculul trafic per celulă (Erlang B) =====
A_cell = inverlangb(?,?); % Erlangs per celulă

%% ===== 6. Număr celule în sistem =====
numar_celule = arie_totala / ?;

%% ===== 7. Număr apeluri pe oră / celulă =====
T_apel_h = T_apel / 3600; % durata apel în ore
apeluri_pe_ora_celula = A_cell / ?;

%% ===== 8. Număr utilizatori / oră / celulă =====
utilizatori_pe_ora_celula = apeluri_pe_ora_celula / ?;

%% ===== 9. Eficiența spectrală =====
% Definiție: SE = (total Erlang) / (lățimea de bandă în MHz * aria totală)

SE = (A_cell * ?) / ((BW_total/1e6) * ?);

%% ===== 10. Afișare rezultate =====
fprintf("\n===== REZULTATE =====\n");
fprintf("Număr canale RF disponibile           : %.0f\n", numar_canal_RF);
fprintf("Număr canale de trafic totale           : %.0f\n", canale_trafic_total);
fprintf("Număr canale de trafic per celulă       : %.0f\n", canale_trafic_celula);
fprintf("\n-- Trafic și capacitate celulă --\n");
fprintf("Trafic admis per celulă                  : %.1f Erlangs\n", A_cell);
fprintf("Număr apeluri pe oră per celulă         : %.0f apeluri/h\n",
apeluri_pe_ora_celula);
fprintf("Număr utilizatori pe oră per celulă     : %.0f utilizatori/h\n",
utilizatori_pe_ora_celula);
fprintf("\n-- Eficiență spectrală --\n");
fprintf("Eficiență spectrală                     : %.2f Erlangs/MHz/km^2\n", SE);

```

FIȘĂ DE LUCRU**Sarcina 1**

Rezultate obținute:

a)

b)

c)

d)

e)

Sarcina 2

Rezultate obținute în linia de comandă:

Sarcina 3

a) Reprezentare grafică cadru DL

b) Durata perioadei de gardă determinată prin măsurători.

c) Durata unui TS determinată prin măsurători

d) Reprezentare grafică cadru UL

e) Durata perioadei de gardă determinată prin măsurători.

f) Durata unui cadru determinată prin măsurători

Sarcina 4

Script corectat și reprezentare grafică:

Sarcina 5

Script corectat și rezultate obținute:

Lucrarea 11 – Tehnici de creștere a eficienței spectrale în sistemele celulare

În comunicațiile radio celulare, semnalul recepționat este afectat atât de variații lente ale mediului de propagare (fading la scară largă), cât și de fluctuații rapide cauzate de propagarea multicală (fading la scară redusă). Aceste efecte pot reduce semnificativ calitatea legăturii radio. Pentru a le combate, sistemele wireless folosesc tehnici de diversitate spațială, prin care aceeași informație este transmisă pe mai multe căi independente, crescând probabilitatea unei recepții corecte. În acest laborator sunt introduse câteva tehnici de diversitate la transmisie și recepție, urmate de aplicații practice în Matlab.

Obiective

- O1. Prezentarea a trei metode de diversitate a combinării.
- O2. Prezentarea codului Alamouti.
- O3. Efectuarea unor sarcini de lucru în Matlab privitoare la simularea modelelor.
- O4. matematici ale fenomenelor sus-amintite.

Durata medie de studiu individual: 30 minute.

11.1 INTRODUCERE

În sistemele mobile fără fir, anvelopa semnalului recepționat este formată din suprapunerea componentelor afectate de fading. Una dintre cauzele de apariție a fading-ului este propagarea multicală. Aceasta poate fi combătută cu ajutorul tehnicilor de diversitate spațială. În cadrul tehnicilor de diversitate spațială, aceeași informație este transmisă prin canale independente și necorelate pentru a combate fenomenul de fading. Când mai multe copii ale aceluiași semnal sunt trimise pe canale diferite, fiecare dintre aceste copii va fi afectată în mod diferit de fading. La receptor, semnalul provenit de la căile independente de propagare este combinat și, prin urmare, probabilitatea de a demodula corect datele transmise crește, ceea ce îmbunătățește performanța întregului sistem. Această tehnică este denumită diversitate spațială în sistemul de comunicații [1].

11.2 DIVERSITATEA LA RECEPȚIE

Configurația canalului SIMO presupune utilizarea unei singure antene la nivelul transmițătorului și a mai multor antene la nivelul receptorului. Configurația SIMO oferă diversitate la recepție, unde aceeași informație este recepționată de pe canale afectate în mod independent de fading.

Pornind de la cazul unei transmisii M-PSK cu N antene la receptor, semnalul recepționat pe antena k este dat de expresia:

$$r_k(t) = a_k A_c \cos[2\pi f_c(t - \tau_k) + \theta_m + \varphi_k] + e_k(t), 0 \leq t \leq T_{sym}, \quad k = 1, 2, \dots, N \quad (11.1)$$

unde, A_c reprezintă amplitudinea semnalului transmis, T_{sym} reprezintă durata unui simbol, f_c reprezintă frecvența purtătoare, iar θ_m reprezintă faza semnalului util.

Canalul k afectat de fading este caracterizat de valoarea atenuării, $a_k = |h_k|$, de defazajul indus $\varphi_k = \angle h_k$ și de întârzierea semnalului pe calea de propagare τ_k . La semnalul recepționat se adaugă zgomot gaussian $e(t)$ de medie 0 și varianță $\sigma_e^2 = \frac{N_0}{2}$.

Dacă notăm cu $h_k = a_k e^{j\varphi_k}$ răspunsul la impuls al celui de-al k -lea canal și cu s semnalul util modulat din banda de bază, atunci semnalul recepționat de pe fiecare canal poate fi descris ca:

$$r_k = h_k s + n_k \quad (11.2)$$

unde n_k reprezintă zgomotul recepționat pe canalele I/Q [2].

a) Metoda Maximum Ratio Combining (MRC)

MRC maximizează produsul dintre ponderi și semnalul recepționat. Tehnica de combinare a raportului maxim, folosește toate cele N semnale recepționate, le ponderează și combină în vederea maximizării SNR-ului la ieșirea acestui sistem, γ_{MRC} . SNR-ul este maximizat dacă receptorul cunoaște perfect câștigul și faza celor N canale [3]. Cunoscând atenuarea și defazajul indus de fiecare dintre cele N canale, ponderile sunt alese astfel:

$$w_k^{MRC} = \hat{h}_k = |\hat{h}_k| e^{j\angle \hat{h}_k}, \quad k = 1, \dots, N \quad (11.3)$$

Ieșirea MRC va fi:

$$y = \sum_{k=1}^N w_k^* r_k = \sum_{k=1}^N \hat{h}_k^* (h_k s + n_k) = (h_1 \hat{h}_1^* + h_2 \hat{h}_2^* + \dots + h_N \hat{h}_N^*) s + \mathcal{N} \quad (11.4)$$

Semnalul util obținut va fi mai apoi normalizat:

$$\hat{s} = \frac{y}{|\hat{h}_1|^2 + |\hat{h}_2|^2 + \dots + |\hat{h}_N|^2} = \frac{\sum_{k=1}^N \hat{h}_k^* r_k}{\sum_{k=1}^N |\hat{h}_k|^2} \quad (11.5)$$

Pentru cazul unei estimări perfecte a canalului, SNR-ul obținut după procesarea MRC va fi egal cu:

$$\gamma_{MRC} = \frac{E_s}{N_0} (|h_1|^2 + |h_2|^2 + \dots + |h_N|^2) \quad (11.6)$$

b) Metoda Equal Gain Combining (EGC)

Dezavantajul metodei MRC constă în complexitatea ei sporită care constă în necesitatea cunoașterii stării canalului, adică a atenuărilor și defazajelor induse de fading. În cadrul metodei EGC, recepția este realizată prin însumarea semnalelor recepționate și sincronizate în fază. În cazul unei scheme de modulație M-PSK, schema EGC, nu necesită estimarea câștigului canalelor afectate de fading și, prin urmare, necesită o complexitate mai redusă în implementare față de metoda MRC [3].

Ponderile utilizate de schema EGC sunt de forma:

$$w_k^{MRC} = e^{j\angle \hat{h}_k}, \quad k = 1, \dots, N \quad (11.7)$$

Dacă presupunem o estimare perfectă a fazelor celor N căi de propagare $\angle \hat{h}_k = \angle h_k$, ieșirea sumatorului EGC este dată de expresia:

$$\begin{aligned} y &= \sum_{k=1}^N w_k^* r_k = \sum_{k=1}^N \hat{w}_k^* (h_k s + n_k) \\ &= \sum_{k=1}^N e^{-j\angle \hat{h}_k} (|h_k| e^{j\angle h_k} s + n_k) = \sum_{k=1}^N (|h_k| s + e^{-j\angle \hat{h}_k} n_k) = \\ &= (|h_1| + |h_2| + \dots + |h_N|) s + \text{zgomot} \end{aligned} \quad (11.8)$$

Semnalul util obținut va fi mai apoi normalizat:

$$\hat{s} = \frac{y}{|\hat{h}_1| + |\hat{h}_2| + \dots + |\hat{h}_N|} = \frac{\sum_{k=1}^N e^{-j\angle \hat{h}_k} r_k}{\sum_{k=1}^N |\hat{h}_k|} \quad (11.9)$$

SNR-ul instantaneu obținut după procesarea EGC va fi egal cu:

$$\gamma_{MRC} = \frac{E_s}{N_0} (|h_1| + |h_2| + \dots + |h_N|)^2 \quad (11.10)$$

c) Metoda Selection Combining (SC)

În cadrul metodei SC, se procesează doar semnalul recepționat de pe calea cu cel mai mare SNR (adică cel mai puternic semnal din cele N semnale recepționate) [3].

$$w_k^{SC} = \begin{cases} 1, & \gamma_k = \arg \max_i \gamma_i \\ 0, & \text{altfel} \end{cases} \quad (11.11)$$

Ponderea corespunzătoare căii cu $\text{SNR}_{\max}$ este stabilită la $w_k = 1$, unde $k = \arg \max_i |\hat{h}_i|$. Cu alte cuvinte, este aleasă ramura pentru care modulul răspunsului la impuls al canalului este maxim, deoarece $\text{SNR} \propto |\hat{h}_i|$.

Ieșirea sistemului care utilizează metoda SC va fi:

$$y = \sum_{k=1}^N w_k^* r_k = w_i^* r_i = r_i \quad (11.12)$$

SNR-ul instantaneu la ieșirea sistemului care utilizează metoda SC este determinat ca fiind valoarea maximă a SNR-urilor corespunzătoare tuturor ramurilor de recepție.

$$\gamma_{SC} = \arg \max_i \gamma_i \quad (11.13)$$

11.3 DIVERSITATEA LA TRANSMISIE

Atunci când receptorul este echipat cu mai multe antene, calitatea transmisiei poate fi îmbunătățită prin utilizarea tehnicilor de diversitate la recepție. Această metodă este utilă în special pentru transmisia uplink, deoarece stația de bază este echipată cu mai multe antene. Echiparea terminalului mobil cu antene multiple nu este practică din cauza limitărilor la nivel de dimensiune și complexitate. Prin urmare, este dificil să se îmbunătățească performanța legăturii downlink folosind tehnici de diversitate la recepție. În acest caz, performanța legăturii downlink poate fi îmbunătățită prin utilizarea mai multor antene la nivelul stației de bază [1].

S.M. Alamouti descrie o schemă simplă de diversitate de transmisie care folosește două antene la emițător și o antenă la receptor. Tehnica nu necesită feedback de la receptor, însemnând că informațiile despre starea canalului sunt necunoscute la nivelul transmițătorului, iar complexitatea computațională este similară cu tehnica de diversitate a recepției MRC cu o antenă de transmisie și două la recepție [1].

În această tehnică, în cadrul intervalului $[0, T_{sym}]$, sunt transmise simultan simbolul s_1 și simbolul s_2 , de către prima și, respectiv, cea de-a doua antenă. În cadrul intervalului $[T_{sym}, 2 \times T_{sym}]$, sunt transmise simbolurile $-s_2^*$ și s_1^* , de către prima și, respectiv, cea de-a doua antenă. Simbolul „*” marchează operatorul de conjugare a numerelor complexe. Prin urmare, semnalele transmise în cele două perioade de simbol T_{sym} , sunt:

$$\mathbf{x}^1 = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}, \quad \mathbf{x}^2 = \begin{bmatrix} -s_2^* \\ s_1^* \end{bmatrix} \quad (11.14)$$

Codarea Alamouti se realizează în timp și spațiu, de aceea aceasta aparține clasei codurilor spațio-temporale [3].

Semnalul recepționat în cele două intervale T_{sym} , după ce acesta parcurge un canal care îi adaugă zgomot Gaussian, este de forma:

$$r_1 = r(t) = h_1 s_1 + h_2 s_2 + n_1 \quad (11.15)$$

$$r_2 = r(t + T_{sym}) = -h_1 s_2^* + h_2 s_1^* + n_2 \quad (11.16)$$

unde h_1, h_2 sunt răspunsurile la impuls ale canalelor asociate cu cele două antene de transmisie.

Sub formă matriceală, ecuațiile de mai sus iau următoarea formă:

$$\begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} s_1 & s_2 \\ -s_2^* & s_1^* \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} \quad (11.17)$$

care se mai poate scrie și astfel:

$$\begin{bmatrix} r_1 \\ r_2^* \end{bmatrix} = \begin{bmatrix} h_1 & h_2 \\ h_2^* & -h_1^* \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} \quad (11.18)$$

sau sub forma matriceală:

$$\mathbf{r} = \mathbf{H}\mathbf{s} + \mathbf{n} \quad (11.19)$$

La receptor are loc înmulțirea semnalului recepționat cu $\mathbf{r}$ cu $\mathbf{H}^H$ (adică transpusa și conjugata matricei $\mathbf{H}$).

$$\begin{aligned} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} &= \begin{bmatrix} h_1^* & h_2 \\ h_2^* & -h_1 \end{bmatrix} \begin{bmatrix} r_1 \\ r_2^* \end{bmatrix} + \begin{bmatrix} \tilde{n}_1 \\ \tilde{n}_2 \end{bmatrix} = \begin{bmatrix} |h_1|^2 + |h_2|^2 & 0 \\ 0 & |h_1|^2 + |h_2|^2 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} \tilde{n}_1 \\ \tilde{n}_2 \end{bmatrix} \\ &= \mathbf{H}\mathbf{s} + \mathbf{n} \end{aligned} \quad (11.20)$$

unde $\tilde{\mathbf{n}} = \mathbf{H}^H \mathbf{n}$. Așadar, la ieșirea nivelului receptorului vor fi recepționate următoarele două semnale:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} h_1^* & h_2 \\ h_2^* & -h_1 \end{bmatrix} \begin{bmatrix} r_1 \\ r_2^* \end{bmatrix} + \begin{bmatrix} \tilde{n}_1 \\ \tilde{n}_2 \end{bmatrix} = \begin{bmatrix} |h_1|^2 + |h_2|^2 & 0 \\ 0 & |h_1|^2 + |h_2|^2 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} \tilde{n}_1 \\ \tilde{n}_2 \end{bmatrix} \quad (11.21)$$

$$y_2 = (|h_1|^2 + |h_2|^2)s_2 + \tilde{n}_2 \quad (11.22)$$

Prin urmare, simbolurile recepționate vor fi:

$$\hat{s}_1 = \frac{y_1}{|h_1|^2 + |h_2|^2} \quad (11.23)$$

$$\hat{s}_2 = \frac{y_2}{|h_1|^2 + |h_2|^2} \quad (11.24)$$

Sarcina 1

Pornind de la scriptul [L11_Div_Combining.m](#) simulați transmisia unor simboluri BPSK printr-un canal Rayleigh de tip flat-fading și măsurați rata de eroare de simbol (Symbol Error Rate – SER) pentru un număr de ramuri de recepție $N = 1, 2, 4, 8$. Folosiți un număr de minim 10^4 simboluri și un interval de analiză E_b/N_0 de $[-5, 20]$ (dB). Comparați performanța schemelor de diversitate la recepție descrise mai sus. Adăugați rezultatele obținute în [Fișa de lucru](#).

Sarcina 2

Pornind de la scriptul [L11_Alamouti.m](#) rezolvați cerințele de mai jos:

- Generați curba BER vs. SNR pentru un sistem BPSK cu o singură antenă de transmisie și o antenă de recepție ($N_T = 1, N_R = 1$), fără codare Alamouti, pentru un canal de fading de tip Rayleigh. Utilizați graficul obținut ca referință pentru cazurile ulterioare.
- Corectați și rulați codul dat pentru schema Alamouti ($N_T = 2, N_R = 1$) și generați curba BER vs. SNR. Comparați rezultatul obținut cu cel de la punctul anterior.
- Calculați câștigul de diversitate. Cu câți decibeli scade SNR-ul necesar pentru un BER = 10^{-3} atunci când se folosesc două antene în loc de una singură.

Adăugați rezultatele obținute în [Fișa de lucru](#).

Sarcina 3

Pornind de la scriptul [L11_Comparatie_scheme.m](#) comparați performanța E_b/N_0 vs. BER pentru scenariile de comunicații fără diversitate, cu diversitate la recepție tip MRC (1×2), și cu diversitate la transmisie folosind schema Alamouti (2×1). Adăugați rezultatele obținute în [Fișa de lucru](#).

Bibliografie

- [1] E. Puschita, s.a., *Radiocomunicatii Celulare - canalul radio - antene - proiectarea sistemelor – Manual de laborator*, U.T. PRESS, ISBN 978-973-662-496-4, 2009.
- [2] Kwonghue Choi, Huaping Liu, *Problem-Based Learning in Communication Systems Using Matlab and Simulink*, Published by John Wiley & Sons Ltd, 2016.
- [3] Mathuranathan Viswanathan, *Wireless Communication Systems in Matlab: Second Edition*, Independently Published, ISBN 979-864-852-321-0, 2020.

SCRIPT-URI MATLAB

L11 Div Combining.m

```

clearvars; clc; close all;

%% ===== Parametri =====
% nSym - numărul total de simboluri transmise
nSym = 1e4;

% N - vector ce conține numărul de ramuri de diversitate la recepție
% (ex. 1 ramură = fără diversitate, 2 ramuri, 20 ramuri etc.)
N = [1, 20];

% Intervalul Eb/N0 pentru simulare (în dB)
EbN0dBs = -20 : 2 : 36;

% Schema de modulație PSK (M = 2 → BPSK)
M = 2;
k = log2(M); % număr biți per simbol

% Conversie Eb/N0 → Es/N0
EsN0dBs = EbN0dBs + 10*log10(k);

%% ===== Modulator BPSK =====
% Generăm simboluri aleatoare uniform distribuite
d = randi([0 M-1], 1, nSym);

% Modulăm semnalul
s = pskmod(d, M); % output complex de tip BPSK

%% ===== Începerea simulării =====
figure;
hold on;

for nRx = N % parcurgem fiecare configurație de diversitate

    % Vectori pentru rata de eroare simbol (SER)
    ser_MRC = zeros(1, numel(EsN0dBs));
    ser_EGC = zeros(1, numel(EsN0dBs));
    ser_SC = zeros(1, numel(EsN0dBs));

    % Semnalul este replicat pe nRx ramuri de diversitate
    s_div = repmat(s, nRx, 1);

    idx_Es = 1;

```

```

%% ===== Buclă pe valori Es/N0 =====
for EsN0dB = EsN0dBs

    % ===== Canal Rayleigh (flat fading) =====
    h = sqrt(1/2) * (randn(nRx, nSym) + 1j*randn(nRx, nSym));

    % Aplicăm efectul canalului
    signal = h .* s_div;

    % Puterea semnalului în fiecare ramură
    P = sum(abs(signal).^2, 2) / nSym;

    % Conversie SNR → N0 necesar pentru generarea zgomotului
    gamma = 10^(EsN0dB / 10);
    N0 = P / gamma;

    % Generarea zgomotului AWGN pe fiecare ramură
    noise = sqrt(N0/2) .* (randn(size(signal))+1j*randn(size(signal)));

    % Semnalul recepționat
    r = signal + noise;

    %% ===== 1. MRC - Maximum Ratio Combining =====
    s_MRC = sum(conj(?) .* ?, 1) ./ sum(abs(?).^2, 1);
    d_MRC = pskdemod(s_MRC, M);
    ser_MRC(idx_Es) = sum(d_MRC ~= d) / nSym;

    %% ===== 2. EGC - Equal Gain Combining =====
    h_ph = exp(-1j * angle(?)); % anularea fazei canalului
    s_EGC = sum(? .* ?, 1) ./ sum(abs(h), 1);
    d_EGC = pskdemod(s_EGC, M);
    ser_EGC(idx_Es) = sum(d_EGC ~= d) / nSym;

    %% ===== 3. SC - Selection Combining =====
    [~, idx_best] = max(abs(?), [], 1);
    h_best = h(sub2ind(size(h), idx_best, 1:nSym));
    r_best = r(sub2ind(size(r), idx_best, 1:nSym));

    s_SC = r_best .* conj(?) ./ abs(?).^2;
    d_SC = pskdemod(s_SC, M);
    ser_SC(idx_Es) = sum(d_SC ~= d) / nSym;

    idx_Es = idx_Es + 1;
end

%% ===== Reprezentare rezultate =====
semilogy(EbN0dBs, ser_MRC, 'b-', 'LineWidth', 1.5);
semilogy(EbN0dBs, ser_EGC, 'r--', 'LineWidth', 1.5);
semilogy(EbN0dBs, ser_SC, 'g-.', 'LineWidth', 1.5);
end

%% ===== Format reprezentare =====
grid on;
xlabel('Eb/N0 (dB)');
ylabel('SER');
title('Performanța tehnicilor de diversitate în canal Rayleigh fading plat în frecvență');
legend('MRC', 'EGC', 'SC');

```

L11 Alamouti.m

```

clear; clc; close all;

%% ----- Parametri sistem -----
N_frame = 130;      % număr simboluri pe cadru
N_packet = 4000;    % număr pachete simulate
NT       = 2;      % număr antene Tx
NR       = 1;      % număr antene Rx
M        = 2;      % BPSK (M=2)
b        = log2(M); % biți per simbol
SNRdBs   = 0:2:30; % gama SNR
sq2      = sqrt(2);

sym_tab = pskmod(0:M-1, M); % = [-1, +1] complex

%% ----- Simulare BER -----
BER = zeros(1, length(SNRdBs));

for idxSNR = 1:length(SNRdBs)
    SNRdB = SNRdBs(idxSNR);
    sigma = sqrt(0.5 / (10^(SNRdB/10))); % zgomot Gaussian

    noeb_total = 0;

    for p = 1:N_packet

        % ----- Generare biți + modulație -----
        bits = randi([0 M-1], N_frame, NT); % simboluri aleatoare
        X = pskmod(bits, M);                % modulate (N_frame x 2)

        % ----- Bloc Alamouti 2x1 -----
        % X1 = [ x1 , x2 ]
        % X2 = [ -conj(x2) , conj(x1) ]
        X1 = X;
        X2 = [-conj(X(:,2)) conj(X(:,1))];

        % ----- Canale Rayleigh -----
        % H = canal (1 x 2) -> repetat pentru fiecare simbol
        H = (randn(N_frame,NT) + 1j*randn(N_frame,NT))/sq2;

        % ----- Transmisie prin canal -----
        R1 = sum(H .* X1, 2) + sigma*(randn(N_frame,1)+1j*randn(N_frame,1));
        R2 = sum(H .* X2, 2) + sigma*(randn(N_frame,1)+1j*randn(N_frame,1));

        % ----- Însuflare Alamouti -----
        Z1 = R1.*conj(H(:,1)) + conj(R2).*H(:,2);
        Z2 = R2.*conj(H(:,2)) - conj(R1).*H(:,1);

        % Estimare simboluri
        Xd(:,1) = pskdemod(Z1, M);
        Xd(:,2) = pskdemod(Z2, M);

        % ----- Numărare erori -----
        noeb_total = noeb_total + sum(sum(bits ~= Xd));
    end
end

```

```
        BER(idxSNR) = noeb_total / (N_packet * N_frame * NT);
end

%% ----- Afișare rezultate -----
figure;
semilogy(SNRdBs, BER, 'b-o', 'LineWidth', 1.5);
grid on;
xlabel('SNR [dB]');
ylabel('BER');
title('Alamouti 2x1 STBC - BPSK în canal Rayleigh');
axis([SNRdBs(1) SNRdBs(end) 1e-6 1]);
```

L11 Comparatie Scheme.m

```

clear; clc; close all;
rng(0);

%% ===== 1. Parametrii simulării =====
EbN0dBs = 0:2:18;          % valori SNR (dB)
Nbits_total = 2e5;         % biți utili procesați pentru fiecare schemă
batch = 2000;

M = 2;                     % BPSK
k = log2(M);               % biți / simbol

%% ===== 2. Inițializare vectori BER =====
BER_1x1 = zeros(size(EbN0dBs));
BER_MRC = zeros(size(EbN0dBs));
BER_Alam = zeros(size(EbN0dBs));

%% ===== 3. Bucla principală pe valori Eb/N0 =====
for idx = 1:length(EbN0dBs)

    % Conversie liniară
    EbN0dB = EbN0dBs(idx);
    EbN0_lin = 10^(EbN0dB/10);

    % zgomot gaussian
    sigma = sqrt(1/(2*EbN0_lin));

    % numărare erori
    errors_1x1 = 0; bits_sent_1x1 = 0;
    errors_mrc = 0; bits_sent_mrc = 0;
    errors_alam = 0; bits_sent_alam = 0;

    %% ----- Rulăm până se acumulează Nbits_total biți -----
    while bits_sent_1x1 < Nbits_total || ...
           bits_sent_mrc < Nbits_total || ...
           bits_sent_alam < Nbits_total

        %% =====
        % SCENARIUL 1: Sistem fără diversitate (1x1)
        % =====
        nDo = min(batch, Nbits_total - bits_sent_1x1);
        if nDo > 0
            bits = randi([0 1], nDo, 1);
            s = 2*bits - 1; % BPSK mapping

            % canal Rayleigh 1x1
            h = (randn(nDo,1)+1i*randn(nDo,1))/sqrt(2);
            n = sigma*(randn(nDo,1)+1i*randn(nDo,1));

            % transmisie + egalizare
            r = h.*s + n;
            s_hat = r .* conj(h) ./ (abs(h).^2);
            bits_hat = real(s_hat) > 0;

            errors_1x1 = errors_1x1 + sum(bits ~= bits_hat);
            bits_sent_1x1 = bits_sent_1x1 + nDo;
        end
    end
end

```

```

%% =====
%      SCENARIUL 2: Diversitate la recepție – MRC 1x2
% =====
nDo = min(batch, Nbits_total-bits_sent_mrc);
if nDo > 0
    bits = randi([0 1], nDo, 1);
    s = 2*bits - 1;    % BPSK

    % 2 căi independente Rayleigh
    g1 = (randn(nDo,1)+1i*randn(nDo,1))/sqrt(2);
    g2 = (randn(nDo,1)+1i*randn(nDo,1))/sqrt(2);

    n1 = sigma*(randn(nDo,1)+1i*randn(nDo,1));
    n2 = sigma*(randn(nDo,1)+1i*randn(nDo,1));

    r1 = g1.*s + n1;
    r2 = g2.*s + n2;

    % combinație MRC
    s_comb = conj(g1).*r1 + conj(g2).*r2;
    Gnorm = abs(g1).^2 + abs(g2).^2;
    s_hat = s_comb ./ Gnorm;

    bits_hat = real(s_hat) > 0;
    errors_mrc = errors_mrc + sum(bits ~= bits_hat);
    bits_sent_mrc = bits_sent_mrc + nDo;
end
%% =====
%      SCENARIUL 3: Diversitate la transmisie – Alamouti 2x1
% =====
nDo = min(batch, floor((Nbits_total - bits_sent_alam)/2));
if nDo > 0

    % generăm perechi
    bits1 = randi([0 1], nDo, 1);
    bits2 = randi([0 1], nDo, 1);
    s1 = 2*bits1 - 1;
    s2 = 2*bits2 - 1;

    % fading independent pe cele 2 antene
    h1 = (randn(nDo,1)+1i*randn(nDo,1))/sqrt(2);
    h2 = (randn(nDo,1)+1i*randn(nDo,1))/sqrt(2);

    % transmisie Alamouti (2 sloturi)
    n1 = sigma*(randn(nDo,1)+1i*randn(nDo,1));
    n2 = sigma*(randn(nDo,1)+1i*randn(nDo,1));

    r1 = (h1.*s1 + h2.*s2)/sqrt(2) + n1;
    r2 = (-h1.*s2 + h2.*s1)/sqrt(2) + n2;

```

```

% decodare
    Z1 = r1.*conj(h1) + conj(r2).*h2;
    Z2 = r1.*conj(h2) - conj(r2).*h1;

    Hnorm = abs(h1).^2 + abs(h2).^2;
    s1_hat = Z1 ./ Hnorm;
    s2_hat = Z2 ./ Hnorm;

    bits1_hat = real(s1_hat)>0;
    bits2_hat = real(s2_hat)>0;

    errors_alam = errors_alam + sum(bits1~=bits1_hat) +
sum(bits2~=bits2_hat);
    bits_sent_alam = bits_sent_alam + 2*nDo;
end
end

%% ===== Calcul BER =====
BER_1x1(idx) = errors_1x1 / bits_sent_1x1;
BER_MRC(idx) = errors_mrc / bits_sent_mrc;
BER_Alam(idx) = errors_alam / bits_sent_alam;

fprintf("Eb/N0=%2d dB → BER 1x1=%1.3e | MRC=%1.3e | Alamouti=%1.3e\n",
...
        EbN0dB, BER_1x1(idx), BER_MRC(idx), BER_Alam(idx));
end

%% ===== 4. Reprezentare grafică =====
figure;
semilogy(EbN0dBs, BER_1x1, 'k-^', 'LineWidth', 1.4); hold on;
semilogy(EbN0dBs, BER_MRC, 'r-s', 'LineWidth', 1.4);
semilogy(EbN0dBs, BER_Alam, 'b-o', 'LineWidth', 1.4);

grid on;
xlabel('E_b/N_0 [dB]');
ylabel('BER');
title('Comparație BER: 1x1 vs MRC 1x2 vs Alamouti 2x1');
legend('Fără diversitate', 'MRC 1x2', 'Alamouti 2x1', 'Location', 'southwest');
axis tight;

```

FIȘĂ DE LUCRU

Sarcina 1

Script corectat și reprezentare grafică:

Sarcina 2

Script corectat și reprezentare grafică:

Sarcina 3

Script corectat și reprezentare grafică:

Lucrarea 12 – Măsurarea și analiza parametrilor radio în rețele celulare folosind terminale comerciale

Obiectivul acestui laborator constă în prezentarea metodologiei de drive-test, larg utilizată în industria telecomunicațiilor pentru monitorizarea parametrilor radio și evaluarea performanței rețelelor celulare. Activitatea presupune colectarea parametrilor de semnal prin intermediul aplicației NetMonitor Cell Signal Logging și analiza ulterioară a acestora în Matlab. Studenții vor urmări indicatori furnizați de această aplicație, precum RSSI, RSRP, RSRQ și RSSNR și vor genera hărți de acoperire, histogramme și statistici pentru a evidenția variațiile nivelului de semnal.

Obiective

- O1.Utilizarea aplicației NetMonitor Cell Signal Logging pentru colectarea unor parametri radio ai rețelelor celulare.
- O2.Prezentarea și interpretarea parametrilor de performanță înregistrați în cadrul măsurătorilor
- O3.Procesarea și vizualizarea în Matlab a datelor colectate.
- O4.Analiza rezultatelor obținute și identificarea factorilor care afectează calitatea semnalului.

Durata medie de studiu individual: 30 minute.

12.1 PARAMETRII INVESTIGAȚI

În rețelele celulare moderne, evaluarea performanței radio și caracterizarea mediului de propagare sunt procese fundamentale pentru asigurarea calității serviciilor de comunicații mobile. În această lucrare de laborator, studenții vor utiliza aplicația NetMonitor Cell Signal Logging pe terminalul mobil personal pentru realizarea unui set de măsurători în teren în zona campusului universitar, urmând ca datele colectate să fie importate în Matlab pentru vizualizare, filtrare și analiză statistică. Activitatea are ca scop familiarizarea cu metodologia practică de evaluare a performanței unei rețele mobile și dezvoltarea competențelor necesare interpretării datelor colectate în contextul tehnologiilor LTE și 5G. Parametri precum RSSI, RSRP, RSRQ și RSSNR permit cuantificarea nivelului și calității semnalului recepționat, oferind informații detaliate despre acoperirea rețelei, interferențe, degradarea canalului și capacitatea efectivă a acestuia. În cele ce urmează vor fi prezentați schematic parametrii cheie de performanță ce vor fi analizați în cadrul acestei lucrări de laborator.

1. **RSSI (Received Signal Strength Indicator)** - Indicatorul RSSI indică puterea totală recepționată de către terminal pe o anumită lățime de bandă și include atât aportul semnalului util, cât și interferențele provenite de la alte celule, zgomotul termic și alte surse perturbatoare din mediul radio. RSSI nu reflectă calitatea semnalului, ci doar nivelul de putere recepționată în bandă și, în multe cazuri, nu este folosit direct în procedurile de handover, ci servește drept componentă în calculul altor indicatori, precum RSRQ. În sistemele LTE și 5G NR, RSSI nu este întotdeauna raportat explicit către rețea, fiind utilizat intern de către modem. În mod formal, RSSI se definește ca suma puterii semnalului util, a interferenței și a zgomotului măsurate într-o anumită bandă [1].
2. **RSRP (Reference Signal Received Power)** - reprezintă puterea medie a semnalelor de referință recepționate de la stația de bază, fiind unul dintre cei mai importanți parametri utilizați pentru evaluarea acoperirii celulei. Spre deosebire de RSSI, care măsoară puterea totală în banda de recepție, RSRP măsoară exclusiv puterea semnalului util. RSRP este un indicator esențial în procesul de selecție a celulei, în măsurătorile pentru handover și în general în caracterizarea performanțelor fizice ale rețelei LTE sau NR. Valorile tipice ale RSRP variază între -140 dBm pentru semnal foarte slab și -44 dBm pentru semnal puternic [1].

3. **RSRQ (Reference Signal Received Quality)** - este indicatorul care combină informația oferită de RSRP și RSSI. Este utilizat pentru a evalua calitatea semnalului recepționat. Acesta ia în calcul atât puterea semnalului util, cât și interferențele, respectiv zgomotul de pe canalul radio. RSRQ este determinat prin relația $RSRQ = N \times (RSRP / RSSI)$, unde N reprezintă numărul de blocuri de resurse utilizate în procesul de măsurare. Astfel, RSRQ estimează calitatea canalului radio, evidențiind atenuările semnalului datorate congestiei, interferențelor sau variațiilor mari ale mediului de propagare. Este un indicator utilizat în mecanismul de handover sau în evaluarea canalului radio, întrucât reflectă capacitatea terminalului de a demodula semnalul în condiții de interferență ridicată. Valorile RSRQ se situează în mod obișnuit între -19.5 dB (calitate foarte slabă) și -3 dB (calitate bună) [1].
4. **RSSNR (Reference Signal Signal to Noise Ratio)** - nu este definit în specificațiile 3GPP, cum e cazul metricilor de mai sus și, prin urmare, nu este raportat către celula curentă, dar este de obicei definit de producătorul terminalelor mobile. RSSNR este determinat ca media raportului dintre puterea semnalului de referință recepționat de la stația de bază și puterea medie a zgomotului, măsurate într-o anumită lățime de bandă. O valoare RSSNR mai mare (de exemplu >15 dB) indică un raport crescut semnal-zgomot, prin urmare, o calitate bună a semnalului [2].

12.2 RESURSE NECESARE

Hardware

1. Smartphone Android cu modul GPS activ.
2. Conexiune LTE/5G activă.
3. Laptop/PC cu Matlab instalat.

Software

1. NetMonitor Cell Signal Logging.
2. Matlab R2024a (toolbox necesar: Mapping Toolbox).

12.3 ACTIVITATE DE TEREN

a. Instalarea aplicației NetMonitor

1. Accesați Google Play și instalați aplicația NetMonitor Cell Signal Logging (prescurtată în continuare NetMonitor).
2. Acordați permisiuni pentru:
 - Locație.
 - Acces la fișiere.
 - Acces la informațiile rețelei.
3. În meniul aplicației, activați:
 - Data Recording.
 - GPS Logging.
 - Auto Logging Interval: 1s.

b. Deplasarea pentru drive-test

1. Porniți „Start Logging” în aplicație.
2. Plimbați-vă pe o rută de aproximativ **20 minute** în zona campusului studentesc.
3. Încercați să captați zone cu semnal bun și zone cu semnal slab:
 - a. Lângă clădiri înalte.
 - b. În zone deschise.
 - c. În interiorul unui hol.

4. La final apăsați **Stop Logging**.
5. Exportați logul în format CSV (NetMonitor → Sessions → Click prelung pe sesiunea de înregistrare a datelor → Export session to CSV File).

12.4 PROCESAREA DATELOR ÎN MATLAB

1. Importarea fișierului CSV.
2. Extragerea datelor relevante.
3. Filtrarea datelor.
4. Reprezentarea histogramei RSRP, RSRQ, RSSNR.
5. Reprezentare de tip Geo Plot după RSRP, RSRQ, RSSNR.

12.5 ANALIZĂ MĂSURĂTORI ȘI SIMULARE

a. Rezultate obținute în cadrul activității de teren

1. Fișierul CSV rezultat din drive-test.
2. Descrierea rutei parcurse.
3. 3 capturi de ecran din aplicația NetMonitor.

b. Rezultate extrase cu ajutorul Matlab

Cu ajutorul script-ului Matlab [L12_NetMonitor_Analiza.m](#), generați următoarele rezultate și adăugați-le în [Fișa de lucru](#).

1. Histogramă RSRP, RSRQ, RSSNR.
2. Grafice RSRP, RSRQ, RSSNR vs. Timp.
3. Reprezentare a indicatorilor RSRP, RSRQ și RSSNR pe hartă geografică.
4. Calcule suplimentare:
 - RSRP, RSSRQ și RSSNR mediu.
 - RSRP, RSSRQ și RSSNR minim.
 - RSRP, RSSRQ și RSSNR maxim.
 - Deviația standard a RSRP, RSSRQ și RSSNR.

c. Simularea scenariului de test în Matlab

În continuare, parametrul RSRP măsurat în cadrul activității de teren va fi comparat cu rezultatele obținute în urma simulării Matlab a unui scenariu de comunicații similar cu cel real. Scopul acestei sarcini este de a ajusta parametrii simulării pentru a obține rezultate similare cu cele măsurate.

Pornind de la scriptul [L12_Simulare_RSRP.m](#), comparați rezultatul generat de Matlab cu graficul RSRP extras la punctul anterior. Ajustați parametrii modelului din Matlab în sensul minimizării discrepanței dintre rezultatele simulate și cele reale, pentru a obține o corelație cât mai strânsă între cele două seturi de date, prin modificarea unuia sau a mai multor parametri, după cum urmează:

- Frecvența de operare: setați frecvența centrală a canalului detectată de NetMonitor.
- Poziția BS.
- Puterea de emisie: ajustați puterea de emisie până când nivelul mediu al semnalului simulat devine comparabil cu cel măsurat.
- Numărul de surse de reflexie și/sau de difracție, etc.

Notați parametrii modificați față de simularea inițială și adăugați reprezentările grafice obținute în [Fișa de lucru](#).

Bibliografie

- [1] ETSI TS 136 214, “LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer; Measurements (3GPP TS 36.214 version 14.2.0 Release 14).” [Online]. Available: https://www.etsi.org/deliver/etsi_ts/136200_136299/136214/14.02.00_60/ts_136214v140200p.pdf
- [2] O. Simpson and Y. Sun, “LTE RSRP, RSRQ, RSSNR and local topography profile data for RF propagation planning and network optimization in an urban propagation environment,” Data in Brief, vol. 21, pp. 1724–1737, Dec. 2018, doi: <https://doi.org/10.1016/j.dib.2018.08.137>.

SCRIPT-URI MATLAB

L12 NetMonitor Analiza.m

```

%% =====
%      Analiza datelor NetMonitor prin măsurători reale
%      RSRP • RSRQ • RSSNR
% =====

clear; clc; close all;

%% === 1. Încărcarea fișierului CSV exportat din NetMonitor ===
file = "log_netmonitor.csv"; % <-- modifică cu numele fișierului tău

opts = detectImportOptions(file, "Delimiter", ";");
data = readtable(file, opts);

%% === 2. Conversie și extragere coloane ===

lat = data.lat;
lon = data.long;
rsrp = data.rssi;
rsrq = data.rsrq;
rssnr = data.rssnr;

%% === 3. Eliminare valori invalide ===
valid = ~isnan(lat) & ~isnan(lon) & ~isnan(rsrp) & (lat > 0) & (lon > 0);

lat = lat(valid);
lon = lon(valid);
rsrp = rsrp(valid);
rsrq = rsrq(valid);
rssnr = rssnr(valid);

fprintf("Număr total de eșantioane valide: %d\n\n", length(lat));

%% =====
%      4. Calcul parametri statistici (RSRP, RSRQ, RSSNR)
% =====

fprintf("==== STATISTICI RSRP (dBm) ==== \n");
fprintf("Media: %.2f dBm\n", mean(rsrp));
fprintf("Deviația standard: %.2f dB\n", std(rsrp));
fprintf("Minim: %.2f dBm\n", min(rsrp));
fprintf("Maxim: %.2f dBm\n", max(rsrp));

fprintf("==== STATISTICI RSRQ (dB) ==== \n");
fprintf("Media: %.2f dB\n", mean(rsrq));
fprintf("Deviația standard: %.2f dB\n", std(rsrq));
fprintf("Minim: %.2f dB\n", min(rsrq));
fprintf("Maxim: %.2f dB\n", max(rsrq));

fprintf("==== STATISTICI RSSNR (dB) ==== \n");
fprintf("Media: %.2f dB\n", mean(rssnr));
fprintf("Deviația standard: %.2f dB\n", std(rssnr));
fprintf("Minim: %.2f dB\n", min(rssnr));
fprintf("Maxim: %.2f dB\n", max(rssnr));

```

```

%% =====
%           5. Hărți reale cu geoscatteer (OpenStreetMap)
% =====

%% – Harta RSRP –
figure;
geoscatteer(lat, lon, 25, rsrp, "filled");
colorbar; colormap(jet);
title("Hartă RSRP (dBm)");
geobasemap streets; % străzi, ideal pentru drive test

%% – Harta RSRQ –
figure;
geoscatteer(lat, lon, 25, rsrq, "filled");
colorbar; colormap(jet);
title("Hartă RSRQ (dB)");
geobasemap streets;

%% – Harta RSSNR –
figure;
geoscatteer(lat, lon, 25, rssnr, "filled");
colorbar; colormap(jet);
title("Hartă RSSNR (dB)");
geobasemap streets;

%% =====
%           6. Histograme pentru RSRP și RSRQ + distribuție RSSNR
% =====

figure;
subplot(3,1,1);
histogram(rsrp, 40);
grid on; title("Histogramă RSRP"); xlabel("RSRP [dBm]");

subplot(3,1,2);
histogram(rsrq, 40);
grid on; title("Histogramă RSRQ"); xlabel("RSRQ [dB]");

subplot(3,1,3);
histogram(rssnr, 40);
grid on; title("Histogramă RSSNR"); xlabel("RSSNR [dB]");

```

L12 Simulare RSRP.m

```

clearvars; clc; close all;

%% 1. Inițializare hartă și parametri fizici
sv = siteviewer('Buildings="map.osm",Basemap="streets_light");

fc = 2640e6; % Frecvența purtătoare [Hz]
c = physconst("LightSpeed"); % Viteza luminii [m/s]
lambda = c/fc; % Lungimea de undă [m]

%% 2. Se folosește antenă izotropă
AntennaElement = phased.IsotropicAntennaElement;
bs_ue_Antenna = AntennaElement;

%% 3. Definirea traseului utilizatorului mobil
route.controlPoints.lats = [ ...
    46.757090 46.756787 46.756335 ...
    46.755823 46.756021 46.757242 ];

route.controlPoints.lons = [ ...
    23.595999 23.593140 23.592302 ...
    23.593204 23.595922 23.596211 ];

route.controlPoints.heights = 1.5 * ones(1,6); % Înălțime UE [m]
route.spacing = 2; % Pasul de eșantionare [m]

% Conversia punctelor de control în obiecte txsite
controlSites = txsite( ...
    Latitude = route.controlPoints.lats, ...
    Longitude = route.controlPoints.lons, ...
    AntennaHeight = route.controlPoints.heights );

%% 4. Generarea punctelor Rx de-a lungul traseului
rxSites = computeWaypoints(controlSites, route.spacing, bs_ue_Antenna);

%% 5. Calculul distanței totale parcurse
totalDistance = 0;
for k = 1:numel(controlSites)-1
    totalDistance = totalDistance + ...
        distance(controlSites(k), controlSites(k+1));
end
%disp("Distanța totală parcursă este " + totalDistance + " metri.");

%% 6. Afișarea traseului și punctelor Rx pe hartă
show(controlSites, ...
    ShowAntennaHeight=false, ...
    Icon="pin.png", ...
    IconSize=[12 30])

show(rxSites, ...
    ShowAntennaHeight=false, ...
    Icon="OrangeDot.png", ...
    IconSize=[5 5])

```

```

%% 7. Definirea stației de bază
txSite = txsite( ...
    Name="NR5G", ...
    Latitude=46.7572768, ...
    Longitude=23.5941831, ...
    AntennaHeight=10, ...
    TransmitterFrequency=fc, ...
    Antenna=bs_ue_Antenna,...
    TransmitterPower=1);

show(txSite);

%% 8. Modelul de propagare - Ray Tracing
numReflections = 1; % Număr maxim de reflexii
numDiffractions = 0; % Difracția este neglijată

propModel = propagationModel("raytracing", ...
    MaxNumReflections=numReflections, ...
    MaxNumDiffractions=numDiffractions);

%% 9. Calculul nivelului de semnal în fiecare punct Rx
numRx = numel(rxSites);
rss_dBm = zeros(1,numRx);

for k = 1:numRx
    % Orientarea antenei Tx către punctul Rx curent
    txSite.AntennaAngle = angle(txSite, rxSites(k));

    rays{k} = raytrace(txSite, rxSites(k), propModel);

    % Estimarea puterii semnalului recepționat
    rss_dBm(k) = computeSignalStrength(rays{k}, txSite, rxSites(k), fc);
end

%% 10. Reprezentare grafică a rezultatelor
figure

% 10.1 Hartă RSS
subplot(2,1,1)
geoscatter([rxSites.Latitude], [rxSites.Longitude], ...
    30, rss_dBm, 'filled')
geobasemap streets
colorbar
colormap jet
title("Hartă nivel semnal recepționat (RSS)")
caxis([-100 -40]) % Domeniu tipic RSS [dBm]

% 10.2 Variația RSS în funcție de distanță
subplot(2,1,2)
distanceAxis = 0:route.spacing:totalDistance;
plot(distanceAxis(1:numel(rss_dBm)), rss_dBm, 'LineWidth',1.5)
grid on
xlabel("Distanța parcursă [m]")
ylabel("Nivel semnal [dBm]")
title("Variația nivelului de semnal de-a lungul traseului")
ylim([-100 -40])

```

```

%% ===== FUNCȚII AUXILIARE =====
function rxSites = computeWaypoints(controlPts, spacing, ueAntenna)
% Generează puncte Rx echidistante de-a lungul unui traseu definit
totalDist = 0;
currentDist = 0;
idx = 0;
offset = 0;
rxSites = rxsite.empty;

for seg = 1:numel(controlPts)-1
    segmentLength = distance(controlPts(seg), controlPts(seg+1));
    totalDist = totalDist + segmentLength;

    az = angle(controlPts(seg), controlPts(seg+1));
    [lat, lon] = location(controlPts(seg), offset, az);

    idx = idx + 1;
    rxSites(idx) = rxsite( ...
        Name=sprintf("Rx%d",idx), ...
        Latitude=lat, ...
        Longitude=lon, ...
        Antenna=ueAntenna, ...
        AntennaHeight=controlPts(seg).AntennaHeight );

    currentDist = currentDist + spacing;

    while currentDist + offset < totalDist
        [lat, lon] = location(rxSites(idx), spacing, az);
        idx = idx + 1;
        rxSites(idx) = rxsite( ...
            Name=sprintf("Rx%d",idx), ...
            Latitude=lat, ...
            Longitude=lon, ...
            Antenna=ueAntenna, ...
            AntennaHeight=controlPts(seg).AntennaHeight );
        currentDist = currentDist + spacing;
    end

    offset = max(0, currentDist - totalDist);
end
end

function rss = computeSignalStrength(rays, tx, rx, fc)
% Estimează puterea totală recepționată prin combinarea coerentă a razelor

% Dacă nu există raze valide → zonă de umbră radio
if isempty(rays) || isempty(rays{1})
    rss = -140;
    return
end

numRays = numel(rays{1});
txPower_dBW = 10*log10(tx.TransmitterPower);

linPower = 0;

```

```
for k = 1:numRays
    azTx = rays{1}(k).AngleOfDeparture(1) - tx.AntennaAngle;
    elTx = rays{1}(k).AngleOfDeparture(2);
    azRx = rays{1}(k).AngleOfArrival(1) - rx.AntennaAngle;
    elRx = rays{1}(k).AngleOfArrival(2);

    txGain = directivity(tx.Antenna, fc, [azTx; elTx]);
    rxGain = directivity(rx.Antenna, fc, [azRx; elRx]);

    rxPower_dBm = txPower_dBW + 30 + ...
        txGain + rxGain - rays{1}(k).PathLoss;

    linPower = linPower + ...
        exp(1i*rays{1}(k).PhaseShift) * 10^(rxPower_dBm/20);
end

rss = 20*log10(abs(linPower));
end
```

FIȘĂ DE LUCRU

Descrierea rutei parcurse:

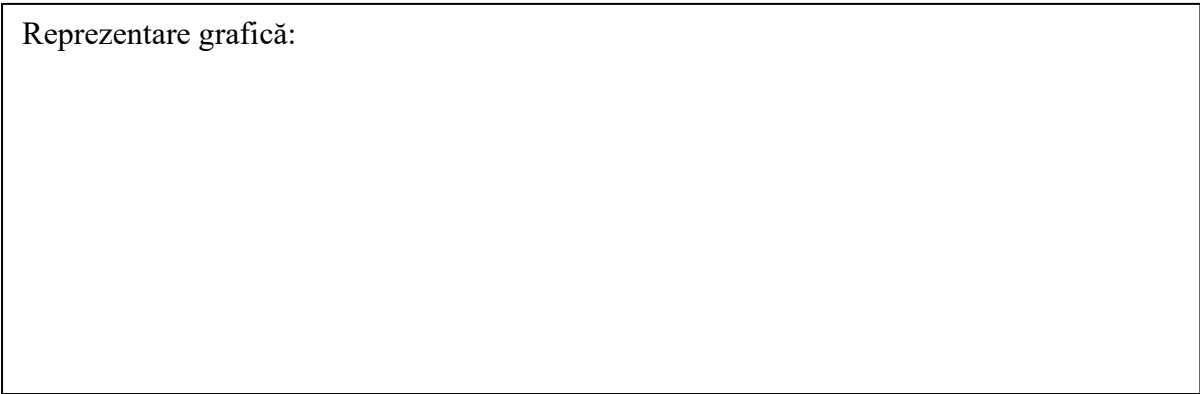
Trei capturi de ecran din NetMonitor Cell Signal Logging:

Geoplot RSRP

Reprezentare grafică:

Geoplot RSRQ

Reprezentare grafică:

**Geoplot RSSNR**

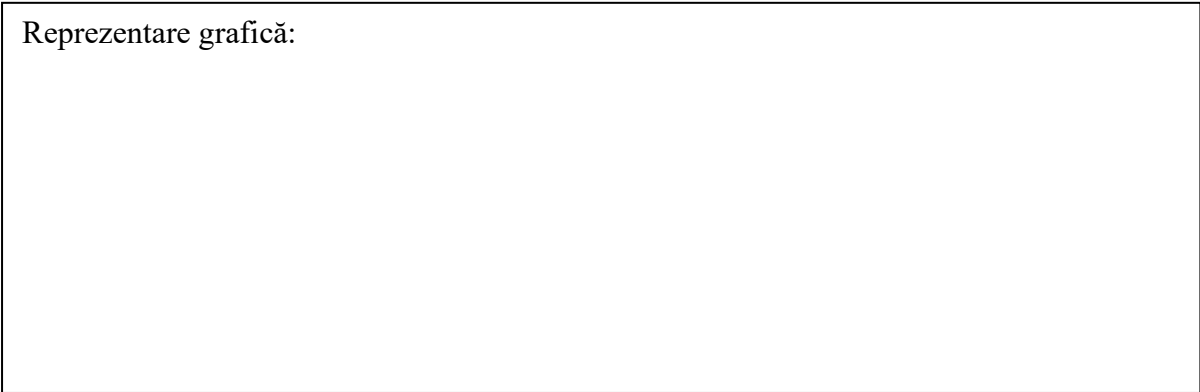
Reprezentare grafică:

**Histogramă RSRP**

Reprezentare grafică:

**Histogramă RSRQ**

Reprezentare grafică:



Histogramă RSSNR

Reprezentare grafică:

Statistici suplimentare

- RSRP mediu
- RSRP minim
- RSRP maxim
- Deviația standard a RSRP

- RSRQ mediu
- RSRQ minim
- RSRQ maxim
- Deviația standard a RSRQ

- RSSNR mediu
- RSSNR minim
- RSSNR maxim
- Deviația standard a RSSNR

Tabel parametri ajustați

Nr. crt.	Parametru ajustat	Valoare
1		
2		
3		
4		
5		

Geoplot RSRP simulat

Reprezentare grafică: