

**UNIVERSITÉ DU QUÉBEC EN OUTAOUAIS**

**GEN6153– HIVER 2024**

**PROJET DE SESSION:**

**Gradient des pertes en fonction de type d’antennes et Modélisation du canal de propagation à l’aide du Tracé des rayons**

**Étudiants:**

**Eric Stephane KAMGUIA KOUAM**

**TSOBGNY GUEGUIM Thierry ruckio**

**Professeur:**

[**Talbi Larbi**](https://moodle.uqo.ca/user/profile.php?id=1920)

**PHASE 1 :**

1. Tracer les profils de puissance z(x), et tracer par-dessus le profil des variations lentes

% Charger les données à partir des fichiers

load('Fichier\_one.mat'); % Remplacez par le nom eviter la confusion

puissance\_enregistree\_1 = z;

load('Fichier\_to.mat'); % Remplacez par le nom pour eviter la confusion

puissance\_enregistree\_2 = z;

% Tracer les profils de puissance

figure;

subplot(2,1,1); % Cas 1 : Antenne de l'émetteur omnidirectionnelle et celle du récepteur directionnelle

plot(x, puissance\_enregistree\_1, 'b', 'LineWidth', 1);

hold on;

% Calculer les variations lentes

profil\_variations\_lentes\_1 = smooth (puissance\_enregistree\_1);

plot(x, profil\_variations\_lentes\_1, 'r--', 'LineWidth', 1);

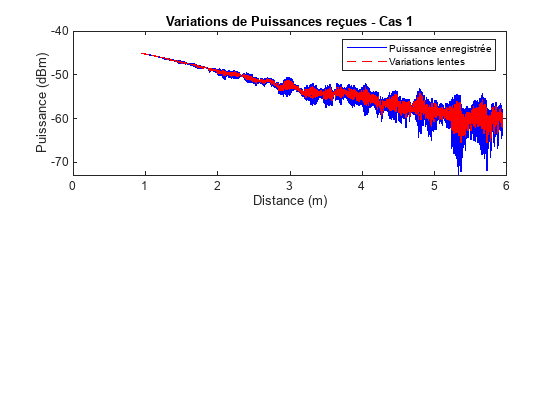
xlabel('Distance (m)');

ylabel('Puissance (dBm)');

title('Variations de Puissances reçues - Cas 1');

legend('Puissance enregistrée', 'Variations lentes');

hold off;



subplot(2,1,2); % Cas 2 : Antennes de l'émetteur et du récepteur sont omnidirectionnelles

plot(x, puissance\_enregistree\_2, 'g', 'LineWidth', 2);

hold on;

% Calculer les variations lentes

profil\_variations\_lentes\_2 = smooth (puissance\_enregistree\_2);

plot(x, profil\_variations\_lentes\_2, 'm--', 'LineWidth', 1);

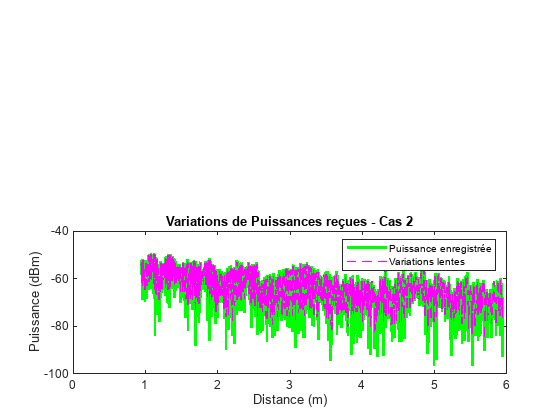
xlabel('Distance (m)');

ylabel('Puissance (dBm)');

title('Variations de Puissances reçues - Cas 2');

legend('Puissance enregistrée', 'Variations lentes');

hold off;



**2.** Déduire et tracer le profil des variations rapides.

% Calculer les variations rapides cas 1

variations\_rapides = puissance\_enregistree\_1 - profil\_variations\_lentes\_1;

% Tracer les variations rapides

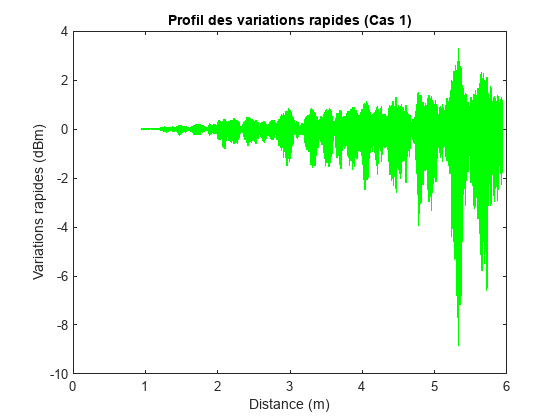
figure;

plot(x, variations\_rapides, 'g', 'LineWidth', 1);

xlabel('Distance (m)');

ylabel('Variations rapides (dBm)');

title('Profil des variations rapides (Cas 1)');



% Calculer les variations rapides cas 2

variations\_rapides = puissance\_enregistree\_2 - profil\_variations\_lentes\_2;

% Tracer les variations rapides

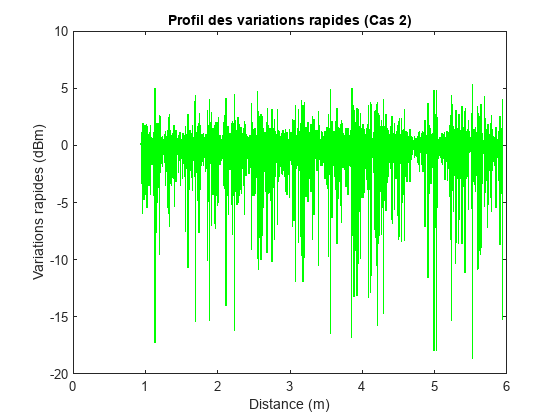
figure;

plot(x, variations\_rapides, 'g', 'LineWidth', 1);

xlabel('Distance (m)');

ylabel('Variations rapides (dBm)');

title('Profil des variations rapides (Cas 2)');



**3.** En utilisant la moyenne locale sur des intervalles de 20λ, tracer les régressions linéaires et en déduire le gradient des pertes n et la variance σ2.

%calcule de la longeur d'onde

f\_s = 37.2\*10^9;

lamda= 3^8/f\_s;

intervalle = lamda\*20;

%calcule de la moyenne locale

intervalle\_moylocal = movmean(puissance\_enregistree\_1, [intervalle intervalle]);

%regression lineaire sur les moyennes local

coeffs = polyfit (x, intervalle\_moylocal, 1);

%reg\_lin = fitlm(intervalle, intervalle\_moylocal)

reg\_dem = polyval(coeffs, x);

%traçer les données de la regression linéaire

plot(x,puissance\_enregistree\_1,'.');

hold on;

plot(x, intervalle\_moylocal, '-r', 'LineWidth', 1); %moyenne local

plot(x, reg\_dem, 'g-', 'LineWidth', 2); %regression lineaire

legend('Données originale', 'Moyenne locale', 'Régression linéaire');

xlabel('X');

ylabel('Y');

title('Régression linéaire avec moyenne locale cas 1');

%calculons l'èrreur des pertes

erreur = puissance\_enregistree\_1 - reg\_dem;

%calcule du gradient des pertes

grad\_pert = -2 \* mean (x.\* erreur);

G = mean (grad\_pert);

variance\_perte = var(erreur);

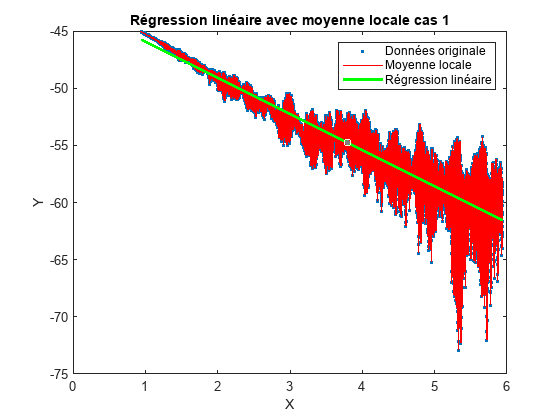
V=mean (variance\_perte);

disp('grad\_pert:');

disp(num2str(G));

disp('variance\_perte:');

disp(num2str(V));



grad\_pert: -13.1891

variance\_perte: 23.861

%calcule de la moyenne locale

intervalle\_moylocal2 = movmean(puissance\_enregistree\_2, [intervalle, intervalle]);

%regression lineaire sur les moyenne local

coeffs1 = polyfit (x, intervalle\_moylocal2, 1);

%reg\_lin = fitlm(intervalle, intervalle\_moylocal)

reg\_dem1 = polyval(coeffs1, x);

%traçer les données de la regression linéaire

plot(x,puissance\_enregistree\_2,'.');

hold on;

plot(x, intervalle\_moylocal2, '-r', 'LineWidth', 1); %moyenne local

plot(x, reg\_dem1, 'g-', 'LineWidth', 2); %regression lineaire

legend('Données originale2', 'Moyenne locale2', 'Régression linéaire2');

xlabel('X2');

ylabel('Y2');

title('Régression linéaire avec moyenne locale cas 2');

%calculons l'èrreur des pertes

erreur1 = puissance\_enregistree\_2 - reg\_dem1;

%calcule du gradient des pertes

grad\_pert2 = -2 \* mean (x.\* erreur1);

G2= mean(grad\_pert2);

variance\_perte2 = var(erreur1);

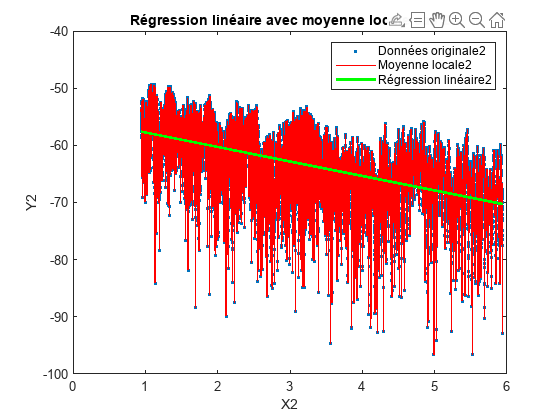
V2 = mean (variance\_perte2);

disp('grad\_pert:');

disp(num2str(G2));

disp('variance\_perte:');

disp(num2str(V2));



grad\_pert: -10.5558

variance\_perte: 44.9574

**4.** Analyser los résultats en indiquant l’impact des antennes.

Grace aux graphes des variation rapides des puissances reçues, on peut dire que l’antenne directionnelle est très sensible par rapport a l’angle de réceptions du rayonnement car nous constatons que plus l’antenne s’éloigne, plus la puissance du signale baisse ce qui n’est pas le cas des antennes omnidirectionnelles.

De plus, en regardant les droites de régressions linéaires de chaque cas on se rend compte que la pente du cas 1 est plus raide que la pente du cas 2 donc dans le cas d’une antenne émettrice omnidirectionnelle et une réceptrice directionnelle l’évanouissement de la puissance de réception est plus rapide que dans le cas où les deux sont omnidirectionnelles. Cela quand bien même que les performances de réceptions d’une antenne directionnelle restent meilleur que celle d’une antenne omnidirectionnelle.

**PHASE** **2** :

1. Antennes omnidirectionnelles : En supposant que les antennes distribuent la puissance de façon équivalente dans toutes les directions et que le coefficient de réflexion du plancher est de -1, développer un programme qui calcule la puissance reçue normalisée en ne tenant compte que de la réflexion sur le plancher. Dessiner le profil de cette puissance (dB) en fonction de la distance (de 1m à 9m avec une incrémentation de x= 1cm).

% Constantes

f = 893e6; % Fréquence en Hz

c = 3e8; % Vitesse de la lumière en m/s

lambda = c / f; % Longueur d'onde en m

R= -1; %coefficient de reflexion du planche

% Géométrie de la salle

room\_length = 12; % Longueur de la salle en m

room\_width = 4; % Largeur de la salle en m

ceiling\_height = 4.5; % Hauteur du plafond en m

antenna\_height = 1.5; % Hauteur des antennes en m

distance\_range = 1:0.01:9; % Distance de 1m à 9m avec incrémentation de 1cm

beta1 = pi; % Phase de propagation (en radians)

% Initialisation des valeurs de delta\_l et delta\_phi

delta\_l\_values = zeros(size(distance\_range));

delta\_phi\_values = zeros(size(delta\_l\_values));

% Calcul de delta\_l et delta\_phi pour chaque distance

for i = 1:length(distance\_range)

delta\_l\_values(i) = 2 \* antenna\_height / distance\_range(i);

delta\_phi\_values(i) = 2 \* pi \* delta\_l\_values(i) / lambda;

end

% Calcul de la puissance reçue normalisée pour chaque distance

received\_powers = zeros(size(distance\_range));

for i = 1:length(distance\_range)

received\_powers(i) = calculerPuissanceRecuenormalise0(R, beta1, delta\_phi\_values(i));

end

% Tracé du profil de la puissance reçue normalisée en fonction de la distance

figure;

plot(distance\_range, received\_powers);

title("Profil de la puissance reçue normalisée en fonction de la distance");

xlabel("Distance (m)");

ylabel("Puissance reçue normalisée (dB)");

grid on;

mean (received\_powers)

[max\_power, max\_index] = max(received\_powers);

fprintf('Le maximum de la puissance est %f dB, atteint à une distance de %f m.\n', max\_power, distance\_range(max\_index));

[min\_power, min\_index] = min(received\_powers);

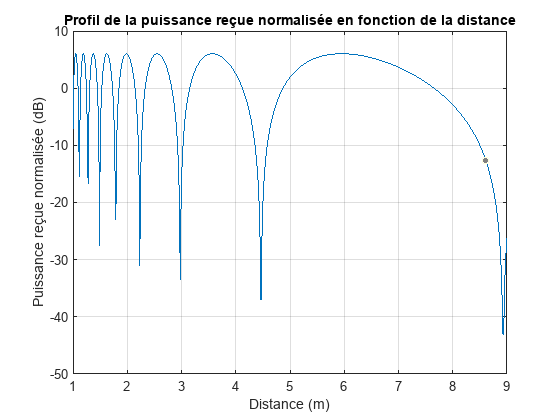
fprintf('Le minimum de la puissance est %f dB, atteint à une distance de %f m.\n', min\_power, distance\_range(min\_index));

function Pr = calculerPuissanceRecuenormalise0(R, beta1, delta\_phi\_values)

e = (1 + abs(R)^2 + 2 \* abs(R) \* cos(delta\_phi\_values + beta1)); % Calcul de e

Pr = 10 \* log10(e); % Calcul de Pr en décibels

end



ans = -Inf

Le maximum de la puissance est 6.020570 dB, atteint à une distance de 5.950000 m.

Le minimum de la puissance est -Inf dB, atteint à une distance de 8.930000 m.

1. Antenne du transmetteur directionnelle et celle du récepteur omnidirectionnel : Dans ce cas, l’antenne du transmetteur distribuera la puissance selon un diagramme de rayonnement en cos. Déterminer ensuite dessiner sur la même figure le profil de la puissance reçue normalisée en fonction de la distance pour les cas : n=1 et n=3.

% Constantes

f = 893e6; % Fréquence en Hz

c = 3e8; % Vitesse de la lumière en m/s

lambda = c / f; % Longueur d'onde en m

R = -1; % Coefficient de réflexion du plancher

% Géométrie de la salle

room\_length = 12; % Longueur de la salle en m

room\_width = 4; % Largeur de la salle en m

ceiling\_height = 4.5; % Hauteur du plafond en m

antenna\_height = 1.5; % Hauteur des antennes en m

distance\_range = 1:0.01:9; % Distance de 1m à 9m avec incrémentation de 1cm

beta1 = pi; % Phase de propagation (en radians)

% Initialisation des valeurs de delta\_l et delta\_phi

delta\_l\_values = zeros(size(distance\_range));

delta\_psi\_values = zeros(size(delta\_l\_values));

Psi\_g = zeros(size(distance\_range));

Diag\_roy = zeros(size(Psi\_g));

Diag\_roy\_3 = zeros(size(Psi\_g));

% Calcul de delta\_l et delta\_phi pour chaque distance

for i = 1:length(distance\_range)

delta\_l\_values(i) = 2 \* (antenna\_height)^2 / distance\_range(i);

delta\_psi\_values(i) = 2 \* pi \* delta\_l\_values(i) / lambda;

Psi\_g(i) = atan(2 \* (antenna\_height)^2 / distance\_range(i));

Diag\_roy(i) = cos(Psi\_g(i));

Diag\_roy\_3(i) = (cos(Psi\_g (i)))^3;

end

% Calcul de la puissance reçue normalisée pour chaque distance pour le cas n = 1

received\_powers1 = zeros(size(distance\_range));

for i = 1:length(distance\_range)

received\_powers1(i) = calculerPuissanceRecuenormalise1(R, Diag\_roy(i), delta\_psi\_values(i));

end

% Calcul de la puissance reçue normalisée pour chaque distance

received\_powers3 = zeros(size(distance\_range));

for i = 1:length(distance\_range)

received\_powers3(i) = calculerPuissanceRecuenormalise1(R, Diag\_roy\_3 (i), delta\_psi\_values(i));

end

% Tracé du profil de la puissance reçue normalisée en fonction de la distance

figure;

hold on;

plot(distance\_range, received\_powers1, 'r', 'LineWidth', 1);

plot(distance\_range, received\_powers3, 'b', 'LineWidth', 1);

hold off;

title("Profil de la puissance reçue normalisée en fonction de la distance");

xlabel("Distance (m)");

ylabel("Puissance reçue normalisée (dB)");

legend("Cas n = 1", "Cas n = 3");

grid on;

mean (received\_powers1)

[max\_power, max\_index] = max(received\_powers1);

fprintf('Le maximum de la puissance est %f dB, atteint à une distance de %f m.\n', max\_power, distance\_range(max\_index));

[min\_power, min\_index] = min(received\_powers1);

fprintf('Le minimum de la puissance est %f dB, atteint à une distance de %f m.\n', min\_power, distance\_range(min\_index));

mean (received\_powers3)

[max\_power, max\_index] = max(received\_powers3);

fprintf('Le maximum de la puissance est %f dB, atteint à une distance de %f m.\n', max\_power, distance\_range(max\_index));

[min\_power, min\_index] = min(received\_powers3);

fprintf('Le minimum de la puissance est %f dB, atteint à une distance de %f m.\n', min\_power, distance\_range(min\_index));

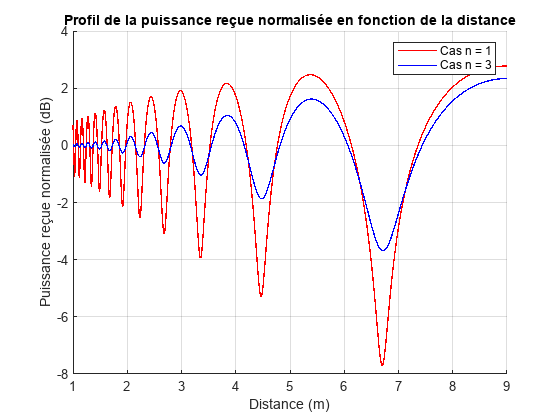
function Pr1 = calculerPuissanceRecuenormalise1(R, psi, delta\_phi)

e = (1 + psi \* R \* exp(-1i \* delta\_phi));

e1 = abs(e);

Pr1 = 10 \* log10(e1); % Calcul de Pr en décibels

end



ans = 0.1114

Le maximum de la puissance est 2.772451 dB, atteint à une distance de 8.970000 m.

Le minimum de la puissance est -7.697695 dB, atteint à une distance de 6.700000 m.

ans = 0.1029

Le maximum de la puissance est 2.341175 dB, atteint à une distance de 9.000000 m.

Le minimum de la puissance est -3.690668 dB, atteint à une distance de 6.710000 m.

3. Pour avoir plus de précision dans notre modèle de propagation, le programme Matlab doit tenir compte des deux murs latéraux et aussi des constantes diélectriques du plancher (permittivité relative  ε= 12) et des murs (ε = 3); donc des coefficients de réflexion dépendant de l’angle d’incidence. On néglige la conductivité du plancher et des murs (ꝺ =0). Vous considérez aussi les deux cas pour le diagramme de rayonnement : n=1 et n=3.

% Constantes

f = 893e6; % Fréquence en Hz

c = 3e8; % Vitesse de la lumière en m/s

lambda = c / f; % Longueur d'onde en m

R= -1; %coefficient de reflexion du planche

epsi\_1 = 12;

epsi\_2 = 3;

% Géométrie de la salle

room\_length = 12; % Longueur de la salle en m

room\_width = 4; % Largeur de la salle en m

ceiling\_height = 4.5; % Hauteur du plafond en m

antenna\_height = 1.5; % Hauteur des antennes en m

distance\_range = 1:0.01:9; % Distance de 1m à 9m avec incrémentation de 1cm

beta1 = pi; % Phase de propagation (en radians)

% Initialisation des valeurs de delta\_l, delta\_phi,Phi\_g\_p, phi\_g\_m, psi\_p3, psi\_m3, theta\_p3, theta\_m3, R\_p3, R\_m3, psi\_p1, psi\_m1, theta\_p1, theta\_m1, R\_p1, R\_m1

delta\_l\_values\_p = zeros(size(distance\_range));

delta\_l\_values\_m = zeros(size(distance\_range));

delta\_phi\_values\_p = zeros(size(delta\_l\_values\_p));

delta\_phi\_values\_m = zeros(size(delta\_l\_values\_m));

Phi\_g\_p = zeros(size(distance\_range));

phi\_g\_m = zeros(size(distance\_range));

psi\_p3= zeros(size(Phi\_g\_p));

psi\_m3 = zeros(size(phi\_g\_m));

theta\_p3 = zeros(size(psi\_p3));

theta\_m3 = zeros(size(psi\_m3));

R\_p3 = zeros(size(theta\_p3));

R\_m3 = zeros(size(theta\_m3));

psi\_p1= zeros(size(Phi\_g\_p));

psi\_m1 = zeros(size(phi\_g\_m));

theta\_p1 = zeros(size(psi\_p1));

theta\_m1 = zeros(size(psi\_m1));

R\_p1 = zeros(size(theta\_p1));

R\_m1 = zeros(size(theta\_m1));

%pour la calcule de delta\_l\_values\_p (pour plancher je suis dans le plan x0y) et pour delta\_l\_values\_m (pour les murs je suis dans le plan x0z)

% Calcul de delta\_l et delta\_phi pour chaque distance

for i = 1:length(distance\_range)

delta\_l\_values\_p(i) = (2 \* (antenna\_height)^2) / distance\_range(i);

delta\_l\_values\_m(i) = (room\_width)^2 / distance\_range(i);

delta\_phi\_values\_p(i) = 2 \* pi \* delta\_l\_values\_p(i) / lambda;

delta\_phi\_values\_m(i) = 2 \* pi \* delta\_l\_values\_p(i) / lambda;

Phi\_g\_p(i) = atan (2\*antenna\_height/distance\_range(i));

phi\_g\_m (i) = atan (2\*room\_width/distance\_range(i));

psi\_p3(i) = (cos(Phi\_g\_p(i)))^3;

psi\_m3(i) = (cos(phi\_g\_m(i)))^3;

theta\_p3(i) = (pi/2) - Phi\_g\_p(i);

theta\_m3(i) = (pi/2) - phi\_g\_m (i);

R\_p3(i) = ((epsi\_1\*cos(theta\_p3(i))) - sqrt(epsi\_1 - sin(theta\_p3(i))^2)) / ((epsi\_1\*cos(theta\_p3(i))) + sqrt(epsi\_1 - sin(theta\_p3(i))^2));

R\_m3(i) = ((cos(theta\_m3(i))) - sqrt(epsi\_2 - sin(theta\_m3(i))^2)) / ((cos(theta\_m3(i))) + sqrt(epsi\_2 - sin(theta\_m3(i))^2));

psi\_p1(i) = cos(Phi\_g\_p(i));

psi\_m1(i) = cos(phi\_g\_m(i));

theta\_p1(i) = (pi/2) - Phi\_g\_p(i);

theta\_m1(i) = (pi/2) - phi\_g\_m(i);

R\_p1(i) = ((epsi\_1\*cos(theta\_p1(i))) - sqrt(epsi\_1 - sin(theta\_p1(i))^2)) / ((epsi\_1\*cos(theta\_p1(i))) + sqrt(epsi\_1 - sin(theta\_p1(i))^2));

R\_m1(i) = ((cos(theta\_m1(i))) - sqrt(epsi\_2 - sin(theta\_m1(i))^2)) / ((cos(theta\_m1(i))) + sqrt(epsi\_2 - sin(theta\_m1(i))^2));

end

% Calcul de la puissance reçue normalisée pour chaque distance

received\_powers5 = zeros(size(distance\_range));

for i = 1:length(distance\_range)

received\_powers5(i) = calculerPuissanceRecuenormalise1(R\_p3(i), R\_m3(i), psi\_p3(i), psi\_m3(i), delta\_phi\_values\_p(i), delta\_phi\_values\_m(i));

end

% Calcul de la puissance reçue normalisée pour chaque distance

received\_powers4 = zeros(size(distance\_range));

for i = 1:length(distance\_range)

received\_powers4(i) = calculerPuissanceRecuenormalise1(R\_p1(i), R\_m1(i), psi\_p1(i), psi\_m1(i), delta\_phi\_values\_p(i), delta\_phi\_values\_m(i));

end

% Tracé du profil de la puissance reçue normalisée en fonction de la distance

figure;

hold on;

plot(distance\_range, received\_powers4, 'cyan', 'LineWidth',1);

plot(distance\_range, received\_powers5, 'r', 'LineWidth',1);

hold off;

title("Profil de la puissance reçue normalisée en fonction de la distance ");

xlabel("Distance (m)");

ylabel("Puissance reçue normalisée (dB)");

legend("Cas n = 1", "Cas n = 3");

grid on;

mean(received\_powers4)

[max\_power, max\_index] = max(received\_powers4);

fprintf('Le maximum de la puissance est %f dB, atteint à une distance de %f m.\n', max\_power, distance\_range(max\_index));

[min\_power, min\_index] = min(received\_powers4);

fprintf('Le minimum de la puissance est %f dB, atteint à une distance de %f m.\n', min\_power, distance\_range(min\_index));

mean(received\_powers5)

[max\_power, max\_index] = max(received\_powers5);

fprintf('Le maximum de la puissance est %f dB, atteint à une distance de %f m.\n', max\_power, distance\_range(max\_index));

[min\_power, min\_index] = min(received\_powers5);

fprintf('Le minimum de la puissance est %f dB, atteint à une distance de %f m.\n', min\_power, distance\_range(min\_index));

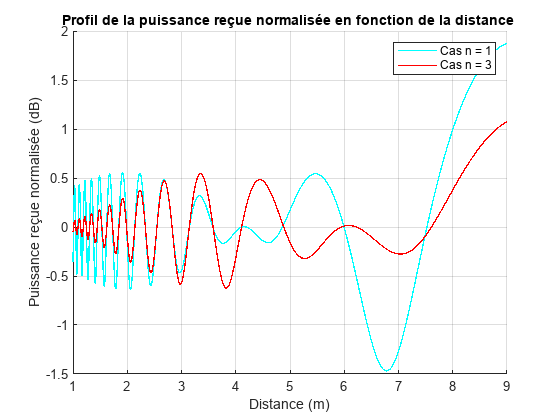
function Pr3 = calculerPuissanceRecuenormalise1(R\_p, R\_m, psi1,psi2, delta\_phi1, delta\_phi2)

e = (1 + psi1.\*R\_p\*exp(-1i\*delta\_phi1)+ 2\*psi2.\*R\_m\*exp(-1i\*delta\_phi2));

e3 = abs (e);

Pr3 = 10 \* log10(e3); % Calcul de Pr en décibels

end



ans = 0.1012

Le maximum de la puissance est 1.876045 dB, atteint à une distance de 9.000000 m.

Le minimum de la puissance est -1.467806 dB, atteint à une distance de 6.780000 m.

ans = 0.0711

Le maximum de la puissance est 1.074522 dB, atteint à une distance de 9.000000 m.

Le minimum de la puissance est -0.624402 dB, atteint à une distance de 3.820000 m.

Sur le diagramme de rayonnement d’une antenne avec un motif de cosinus a la puissance n = 1 présente une directivité modérée. Le signal est moins uniformément reparti avec une puissance qui augmente progressivement à mesure que l’onde s’éloigne de l’émetteur.

Pour un diagramme de rayonnement d’une antenne avec un motif de cosinus a la puissance de n=3, le rayonnement est beaucoup plus concentré le long de l’axe principale ce qui signifie que l’antenne est plus directive avec une porte accrue dans la direction principale et avec une couverture réduite sur les côtes.

Pour n = 1 le récepteur doit être à 9 m pour avoir une bonne réception de ce signal car a cette position la puissance du signal est meilleur et 6.78m est clairement une mauvaise position pour la réception du signal car a cette position la puissance de réception est la plus basse.

Pour n = 3 le récepteur doit être à 9 m pour avoir une bonne réception de ce signal car a cette position la puissance du signal est meilleur et 3.82m est clairement une mauvaise position pour la réception du signal car a cette position la puissance de réception est la plus basse.

Pour une **antenne émettrice directive** avec un rayonnement en cosinus simple (à la puissance 1), le faisceau est plus large et la puissance est répartie sur une plus grande zone. Cela signifie que l’antenne omnidirectionnelle réceptrice, qui capte les signaux de toutes les directions, recevra une partie de cette énergie, mais pas aussi concentrée que si le rayonnement était en cosinus cubique.

En revanche, avec un rayonnement en **cosinus à la puissance cubique**, la puissance est beaucoup plus focalisée et dirigée. Si l’antenne émettrice directive utilise ce type de rayonnement, l’antenne omnidirectionnelle réceptrice recevra une puissance plus concentrée si elle se trouve dans la direction privilégiée du faisceau. Cependant, si l’antenne réceptrice est hors de cette direction, elle pourrait recevoir beaucoup moins d’énergie comparativement au rayonnement en cosinus simple.

En termes de **puissance reçue**, cela signifie que pour un rayonnement en cosinus cubique, l’antenne omnidirectionnelle réceptrice doit être bien alignée avec la direction du faisceau pour recevoir une puissance maximale. Pour un rayonnement en cosinus simple, l’antenne réceptrice a plus de flexibilité en termes de positionnement et peut recevoir une puissance raisonnable même si elle n’est pas parfaitement alignée avec la direction du faisceau.

Il est également important de noter que le **gain** de l’antenne émettrice, qui est une mesure de sa capacité à diriger l’énergie dans une direction spécifique, affectera la puissance reçue par l’antenne omnidirectionnelle. Un gain plus élevé indique une direction plus forte et donc une puissance plus concentrée dans la direction du faisceau.