

Réalisation des fonctions électroniques en utilisant les techniques d'intelligence artificielle : Design d'un réseau d'antennes

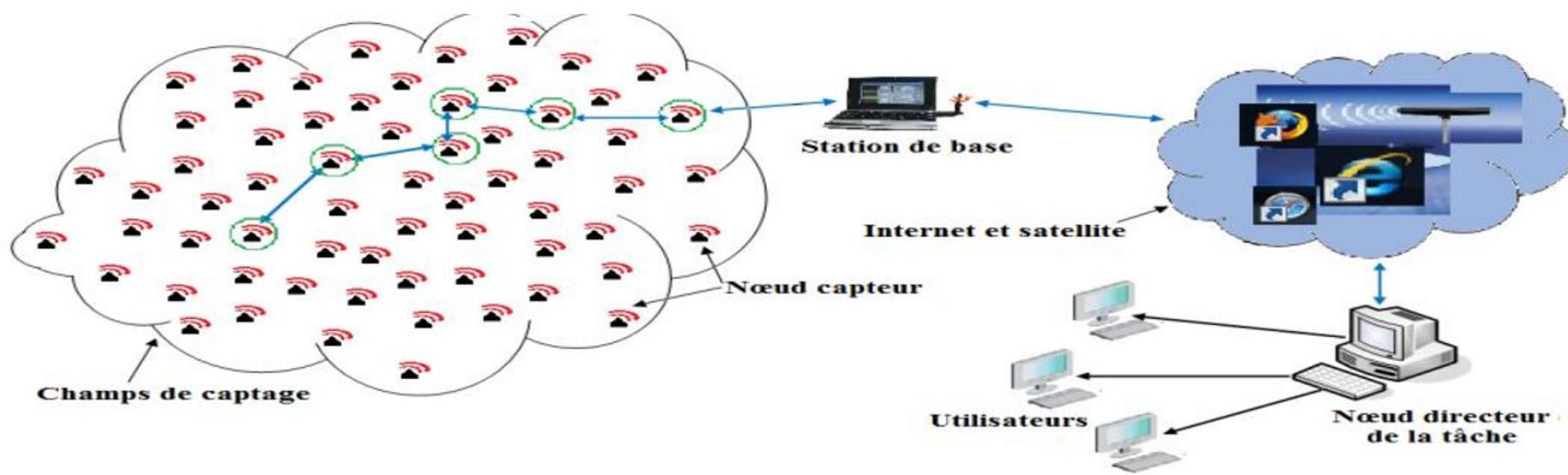
Introduction

Applications des réseaux d'antennes

Réseaux de capteurs

Un **réseau de capteurs** est un ensemble de **nœuds capteurs** déployés dans un environnement donné, capables de :

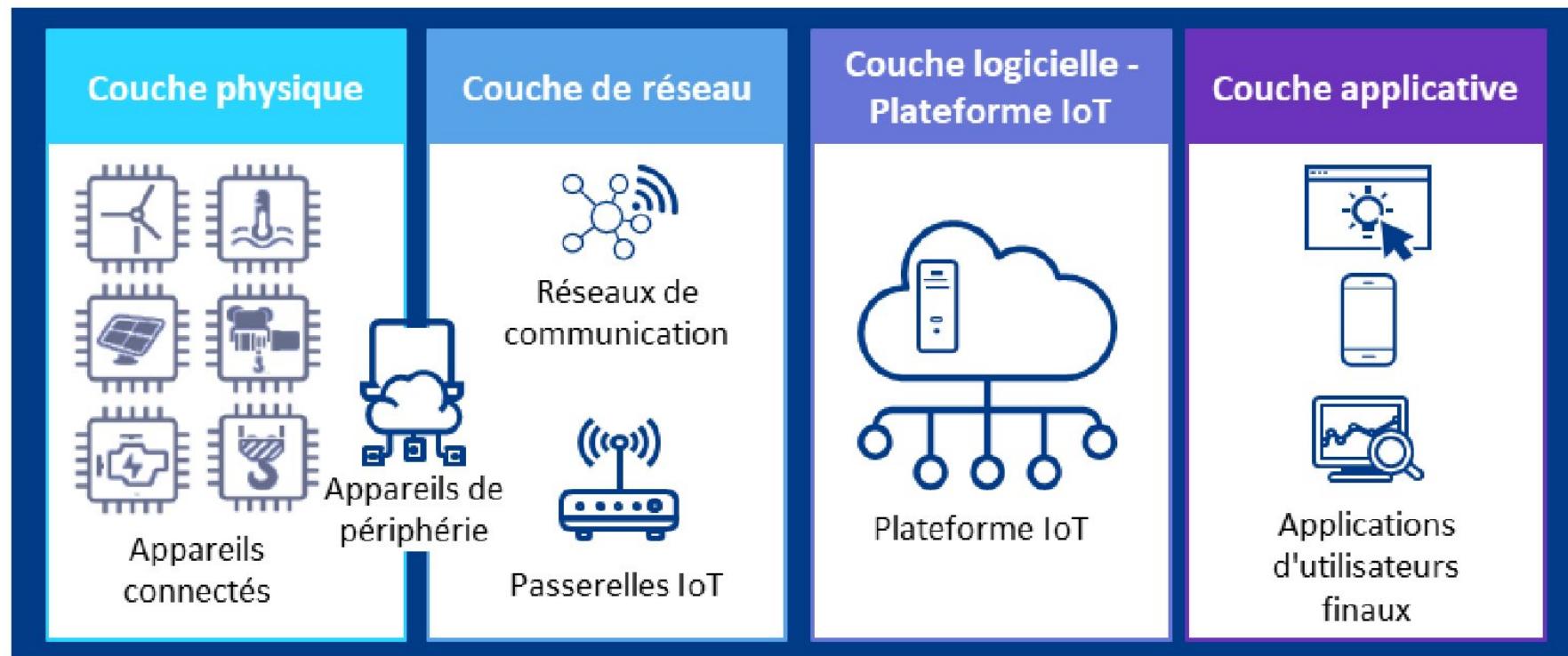
- **Collecter** des informations (température, pression, humidité, vibrations, pollution, etc.) à l'aide de capteurs intégrés
- **Traiter** localement une partie de ces données grâce à des microcontrôleurs
- **Communiquer** sans fil entre eux et avec une station de base
- **Transmettre** les informations collectées pour analyse ou prise de décision



IOT

IOT (Internet of Things) :

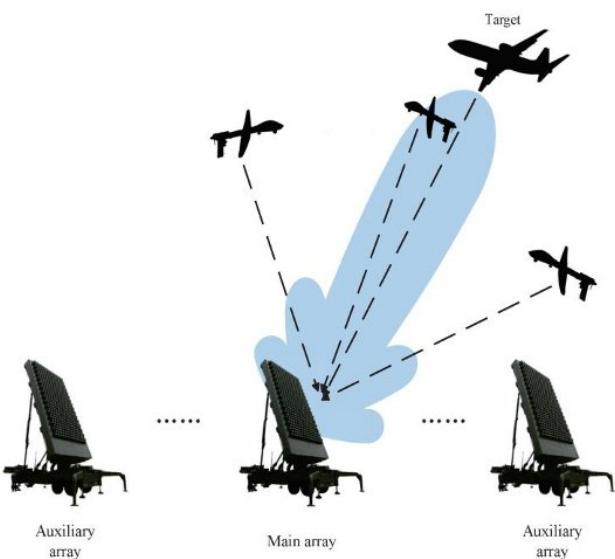
Ensemble des objets physiques connectés à Internet ou à un réseau, capables de collecter, d'échanger et de traiter des données grâce à des capteurs, des logiciels et des technologies de communication



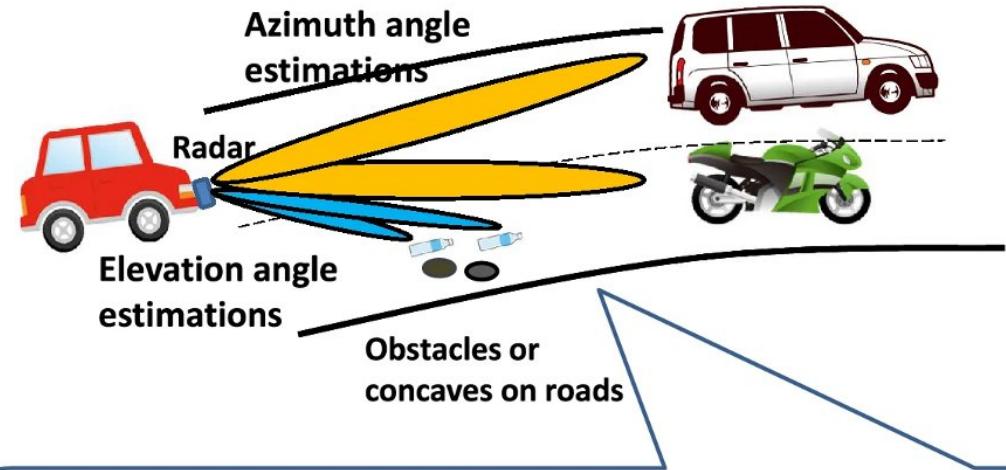
Radars militaires : Beamforming

Le **beamforming** (formation de faisceaux) consiste à piloter les phases et amplitudes des signaux transmis/recus par chaque élément d'un réseau d'antennes pour **former** des lobes directionnels, orienter le faisceau vers une direction désirée et contrôler la réponse spatiale (largeur du lobe principal, niveaux des lobes secondaires).

En contexte militaire, cela permet au radar de concentrer l'énergie vers une cible, d'améliorer la détection et la résistance au brouillage

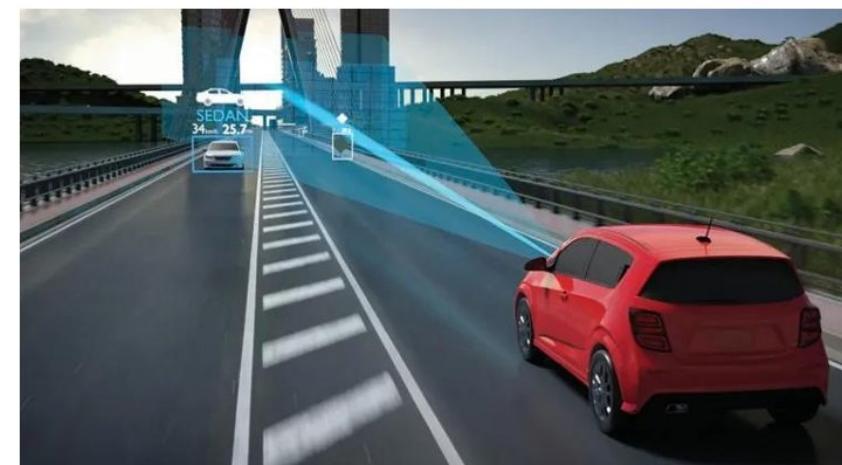


Voitures autonomes : Beamforming



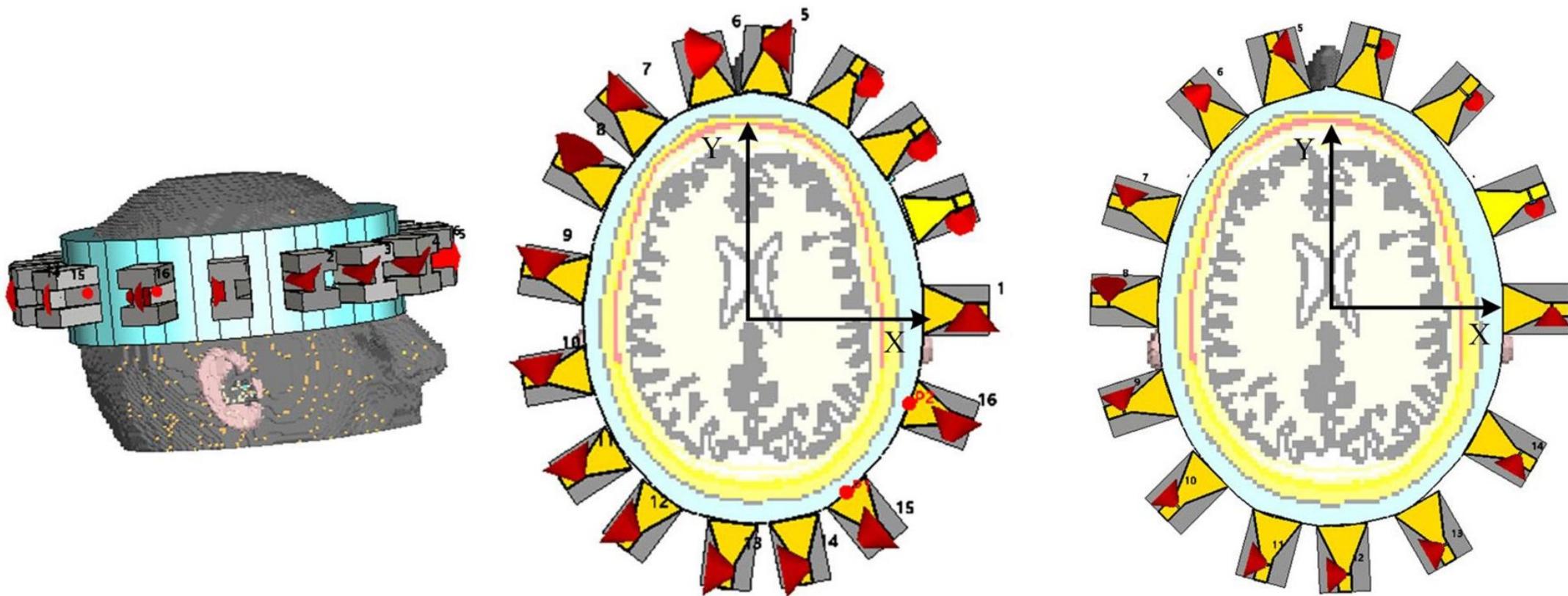
Les **radars des voitures autonomes** utilisent un **réseau d'antennes** couplé à un **beamforming** :

- **Émission et réception directionnelles** : les éléments du réseau ajustent phase/amplitude pour créer un faisceau étroit orienté vers la zone d'intérêt.
- **Balayage électronique** : possibilité de scanner rapidement l'environnement sans mouvement mécanique
- **Faisceaux multiples** : certains systèmes utilisent des faisceaux simultanés pour couvrir plusieurs directions et détecter plusieurs cibles en même temps.

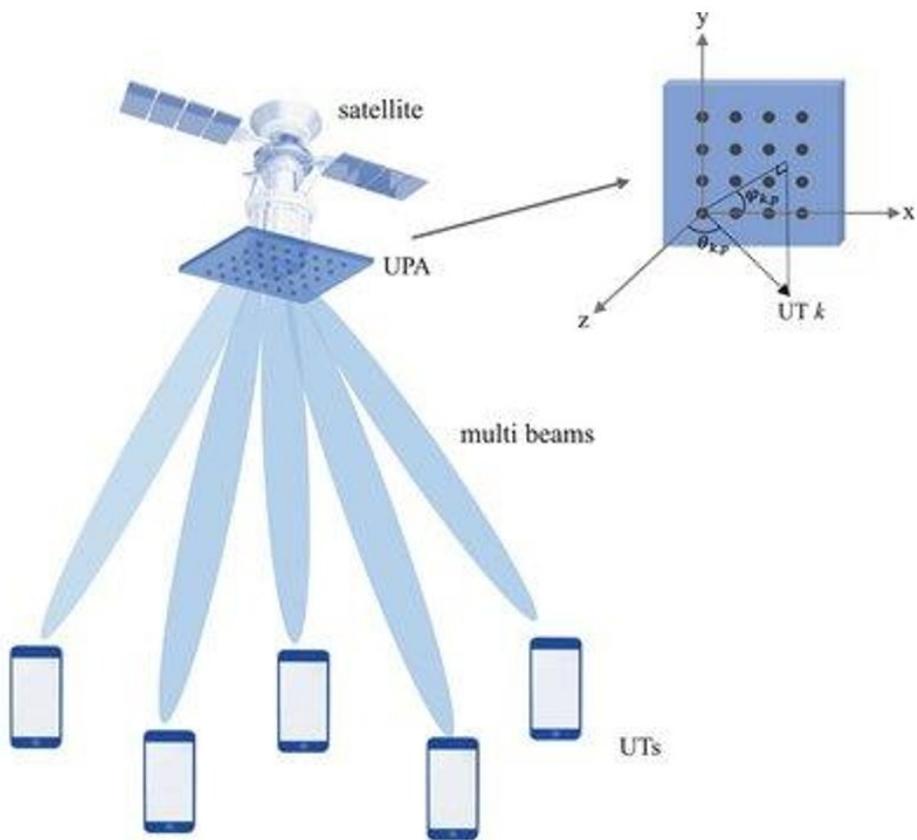


Imagerie médicale

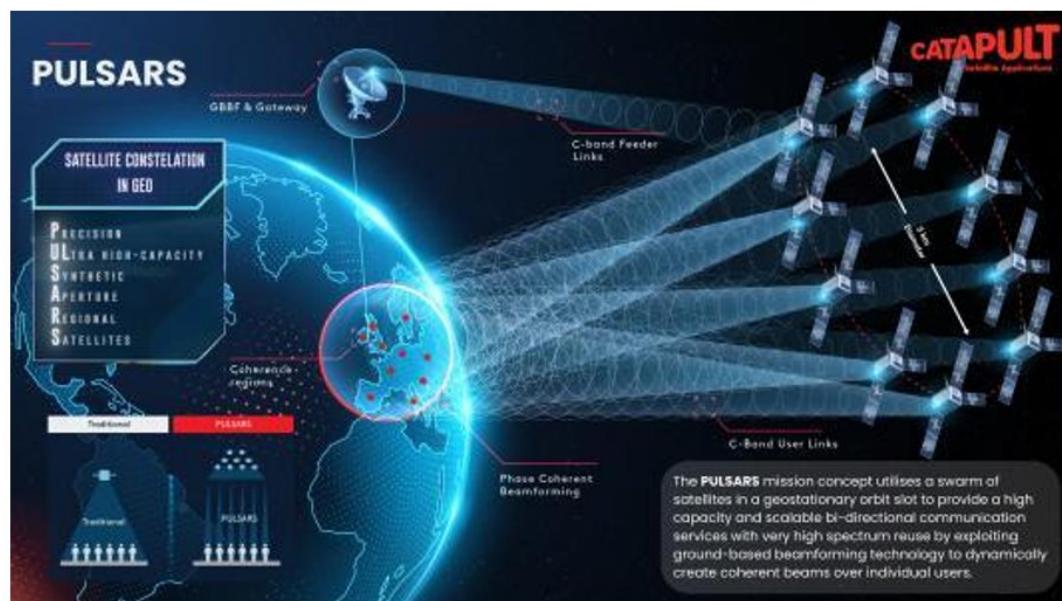
Un **réseau d'antennes** peut être utilisé pour **émettre et recevoir des ondes électromagnétiques** afin de reconstruire une image de l'intérieur du corps



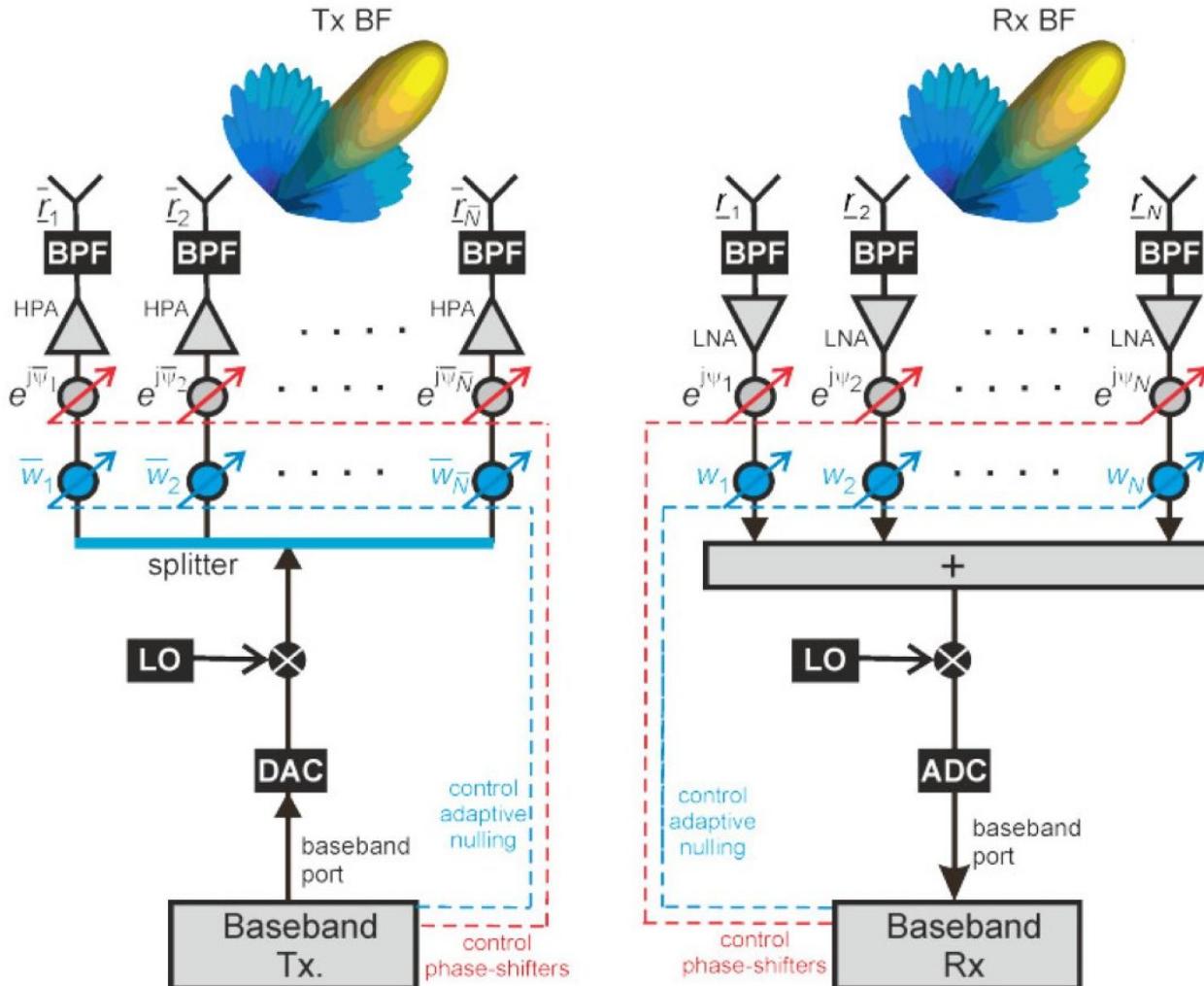
Communication par satellite



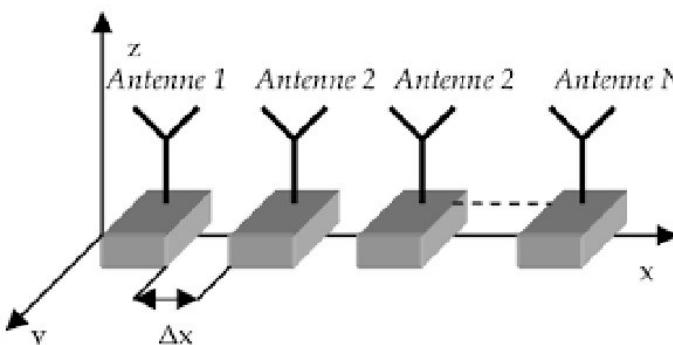
Les satellites de télécommunication utilisent des **réseaux d'antennes avec beamforming** pour fournir un **accès Internet, téléphonie ou télévision** à des zones étendues



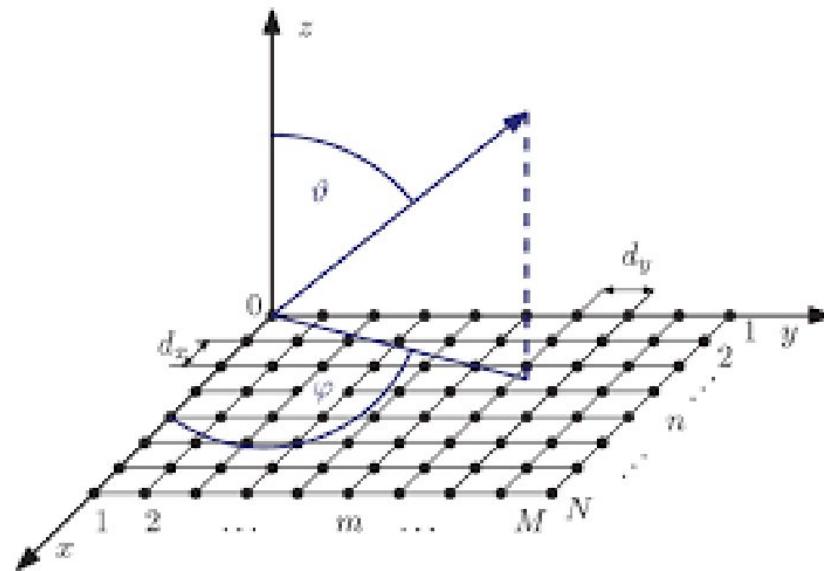
Circuits et systèmes électroniques



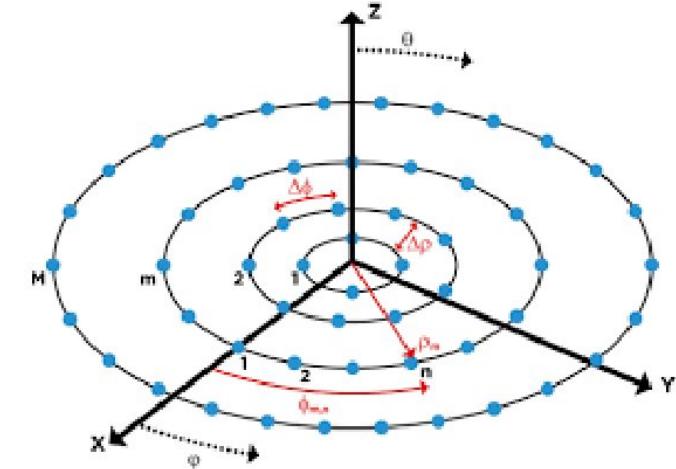
Réseaux d'antennes



Réseau d'antennes linéaire

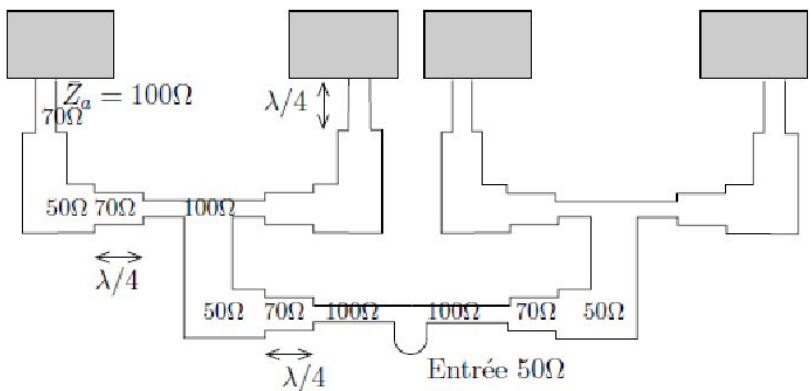
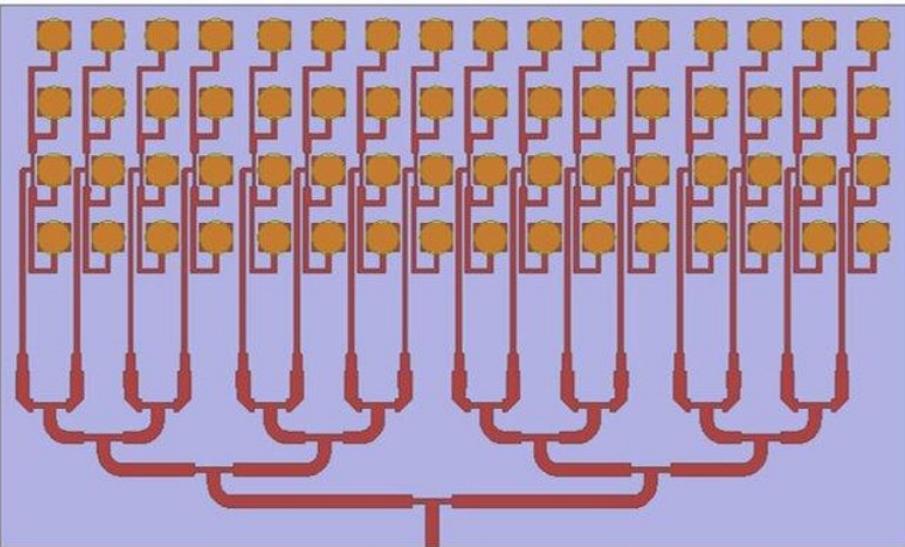
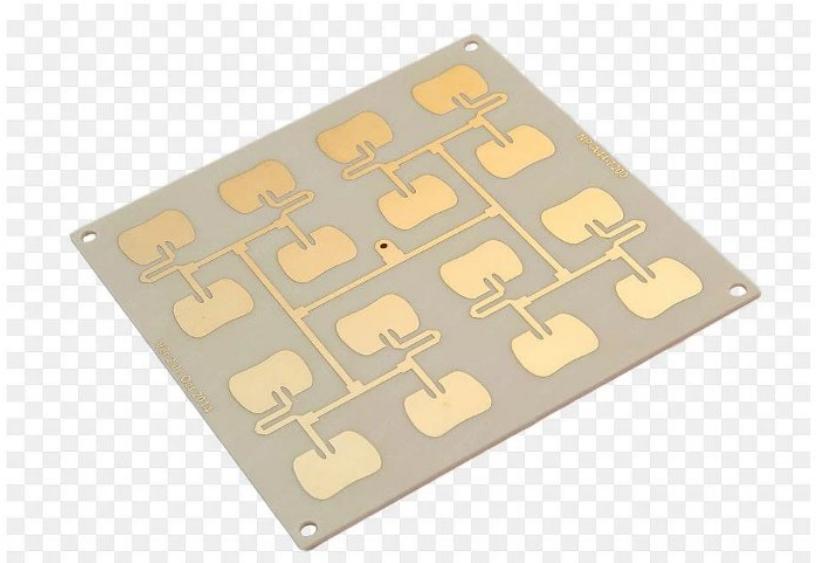


Réseau d'antennes planaire



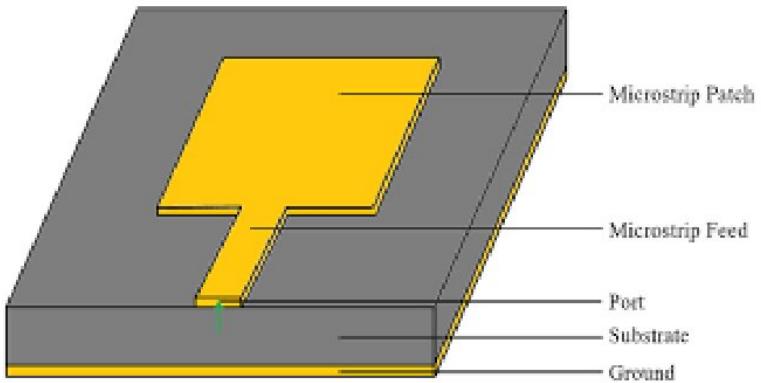
Réseau d'antennes circulaire

Réseaux d'antennes

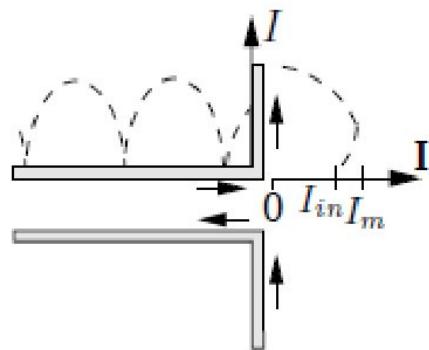


Circuit d'alimentation d'un réseau d'antennes

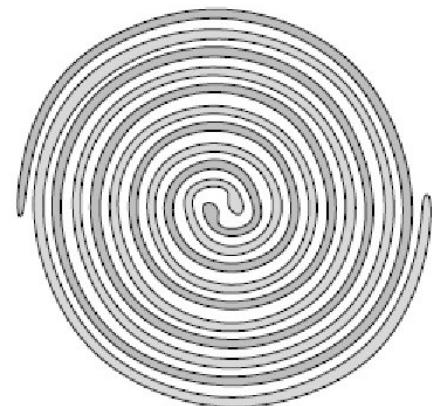
Antenne



Microstrip Patch Antenna



Antenne filiforme



Antenne spirale

Objectif

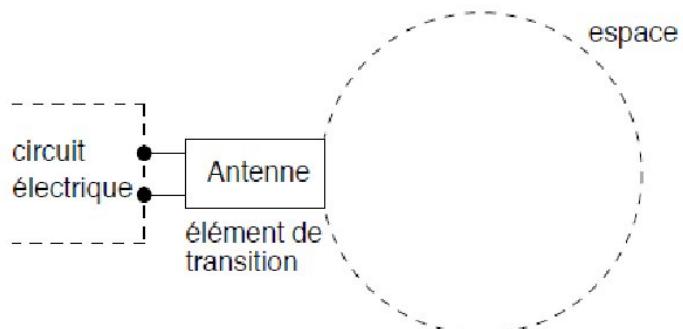
Design d'un réseau d'antennes afin d'atteindre des spécifications bien définies pour une application donnée

La méthode de design est basée sur les techniques de l'intelligence artificielle

Notions de base sur les antennes et les réseaux d'antennes

Introduction

Les antennes servent pour communiquer sur de grandes distances



Principe de fonctionnement d'une antenne :

Lorsqu'une antenne est excitée par un signal électrique, les caractéristiques géométriques et les matériaux constitutifs imposent une **distribution de courant** particulière à sa surface ou à ses conducteurs.

Cette distribution de courant agit comme source de rayonnement électromagnétique en générant des **champs électriques et magnétiques** selon les équations de Maxwell.

L'interaction de ces champs dans l'espace libre produit un motif spatial d'énergie rayonnée appelé **diagramme de rayonnement**.

La forme, l'orientation et les performances de ce diagramme (directivité, gain, largeur de lobe, etc.) dépendent directement de la distribution de courant et donc de la conception de l'antenne et définissent la façon dont l'antenne émet ou reçoit l'énergie dans l'espace.

Paramètres d'antenne

Intensité de rayonnement

on sait que le vecteur de *Poynting* \mathbf{P} indique la quantité de puissance électromagnétique instantanée qui se propage par unité de surface. Pour connaître la densité de puissance en moyenne dans le temps, il faut réaliser :

$$\langle \mathbf{P} \rangle = \frac{1}{T} \int_T \mathbf{P} dt = \text{Re} \left\{ \underbrace{\frac{1}{2} \bar{\mathbf{E}} \times \bar{\mathbf{H}}^*}_{\bar{\mathbf{P}}} \right\} . \quad (2.7)$$

$\bar{\mathbf{E}}$ Champ électrique

$\bar{\mathbf{H}}$ Champ magnétique

$$|\vec{H}| = \frac{|\vec{E}|}{\eta_0}$$

$$\langle P(r, \theta, \phi) \rangle = \frac{|\vec{E}(r, \theta, \phi)|^2}{2\eta_0}$$

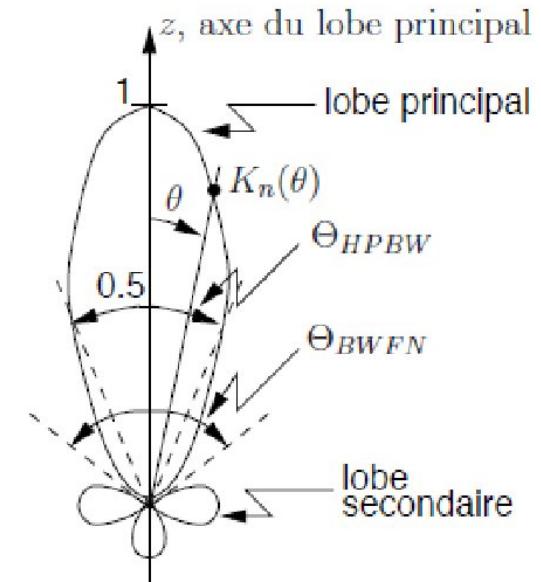
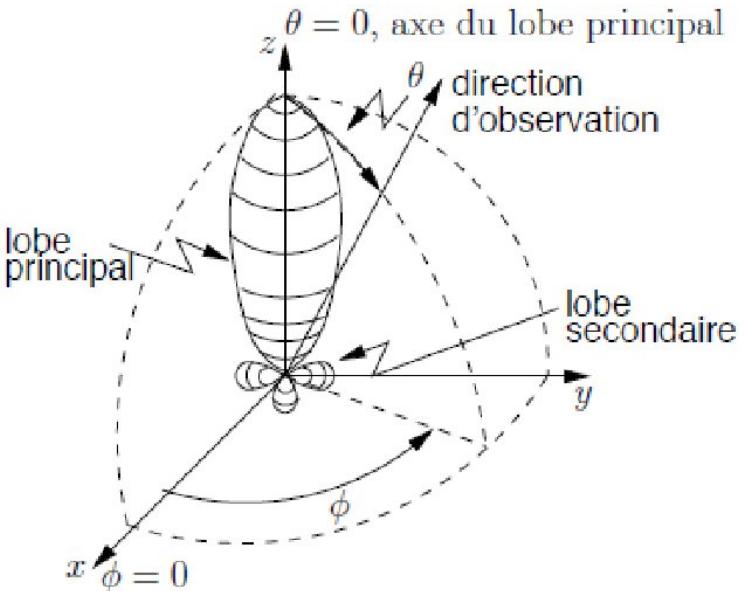
η_0 est l'impédance du vide

On définit alors l'intensité de rayonnement K en W/sr , laquelle est indépendante de r mais varie selon la direction donnée :

$$K(\theta, \phi) = \langle P(r, \theta, \phi) \rangle r^2 = \frac{E^2(r, \theta, \phi)}{2\eta_0} r^2 = \frac{E_\theta^2(r, \theta, \phi) + E_\phi^2(r, \theta, \phi)}{2\eta_0} r^2 . \quad (2.9)$$

Paramètres d'antenne

Diagramme de rayonnement



$\Psi = (\theta, \phi)$ appelée direction d'observation.

Pour une antenne en émission :

La direction d'observation correspond à la direction dans laquelle on regarde le rayonnement émis par l'antenne, c'est-à-dire, dans la pratique, la direction où se situe le récepteur.

Si un récepteur est placé dans une direction définie par l'angle (θ, ϕ) , alors le diagramme de rayonnement indique quelle quantité d'énergie sera reçue dans cette direction.

On définit alors la largeur du lobe principal entre les premiers nuls de chaque côté (Θ_{BWFN}) ou, ce qui est plus fréquent, à 3 dB (Θ_{HPBW})

Sans tenir compte de la polarisation, on n'utilise que celui représentant le module du champ électrique total assimilant le champ électrique à une quantité scalaire. L'échelle peut alors être linéaire (en V/m) ou logarithmique (en dB). Cependant, il est commode de tracer le diagramme après normalisation, ce qui est appelé fonction caractéristique de rayonnement de l'antenne³ $F_a(\theta, \phi)$. Cette fonction, obtenue en divisant $E(\theta, \phi)$ par sa valeur maximale, ne fait intervenir que les variables angulaires :

$$F_a(\theta, \phi) = \frac{E(r_o, \theta, \phi)}{E_{max}(r_o)} \quad (2.12)$$

$$K_n(\theta, \phi) = \frac{K(\theta, \phi)}{K_{max}} = F_a^2(\theta, \phi). \quad (2.13)$$

Dipôle élémentaire

Cas où Le dipôle élémentaire est orienté suivant l'axe z, de longueur dh et parcouru par un courant I_o

$$\bar{E}_{\theta_{elem}} = j\omega\mu I_o dh \frac{e^{-j\beta r}}{4\pi r} \sin \theta$$

$$\beta = \frac{2\pi}{\lambda}$$

μ perméabilité du milieu (vide : $\mu_0 = 4\pi \times 10^{-7} H/m$)

β constante de phase $rads/m$

λ longueur d'onde m

$$c = f \times \lambda$$

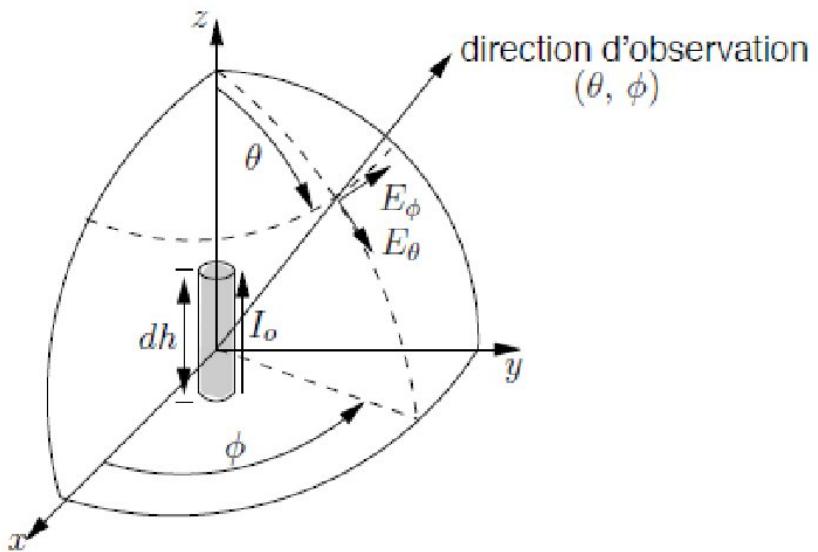
où :

- c est la vitesse de propagation des ondes dans le milieu (en m/s)
 - Dans le vide : $c = 3 \times 10^8 \text{ m/s}$
- f est la fréquence de l'onde (en hertz, Hz)
- λ est la longueur d'onde (en mètres, m)

$$E_{\theta_{elem}} = \left| \bar{E}_{\theta_{elem}} \right|$$

$$E_{\theta_{elem}} \cdot r = \omega\mu \frac{I_o dh}{4\pi} \sin \theta \approx 60\pi I_o \frac{dh}{\lambda} \sin \theta$$

$$F_{a_{elem}} = \sin \theta$$



On peut maintenant tracer les diagrammes de rayonnement dans les plans d'intérêts en prenant le module du champ \bar{E} à distance constante r , puis en normalisant pour obtenir la fonction caractéristique de rayonnement :

Dipôle élémentaire

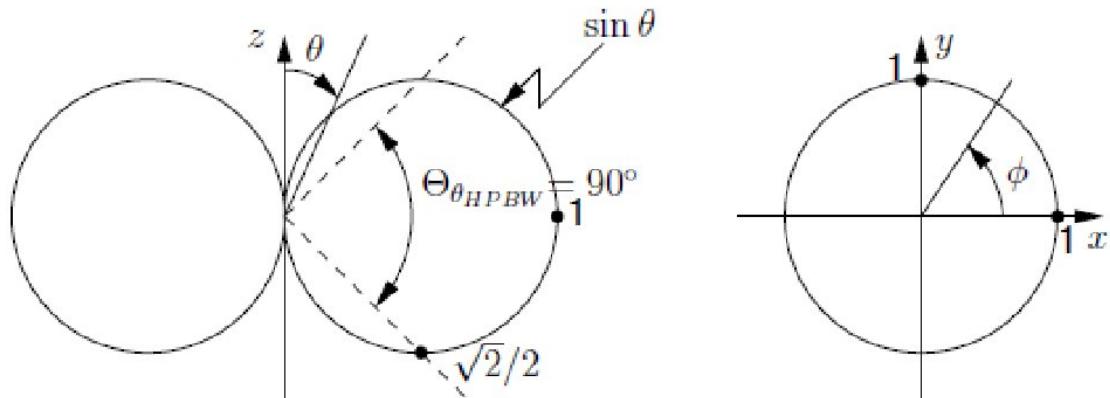


Diagramme de rayonnement du champ électrique du dipôle élémentaire selon θ (plan E) et selon ϕ (plan H).

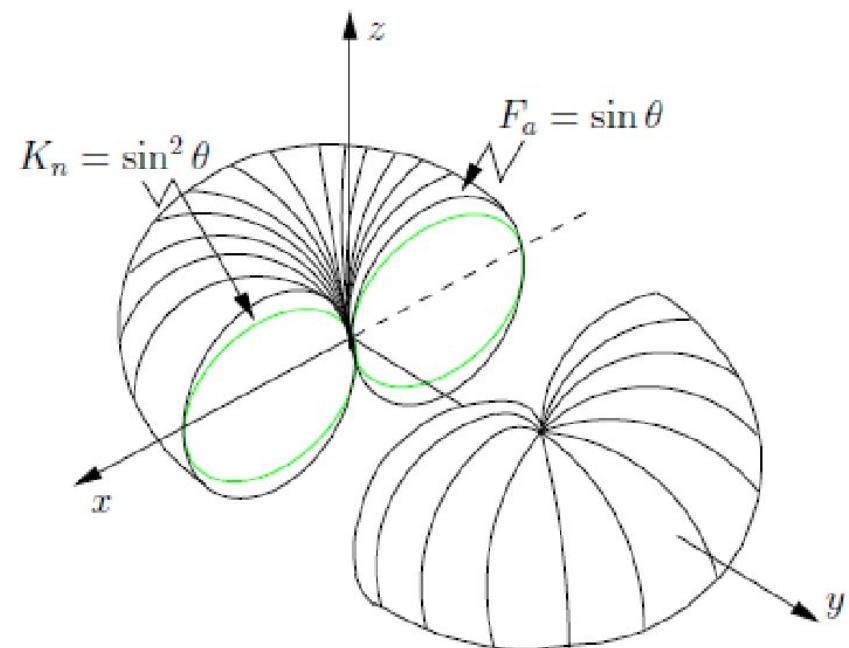
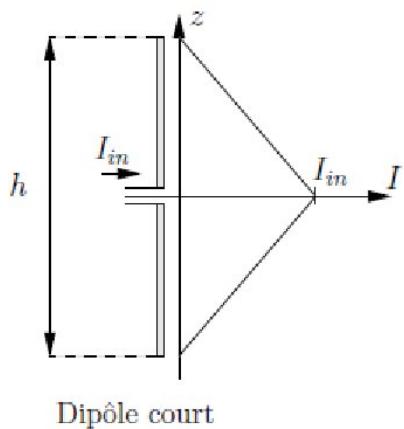


Diagramme de rayonnement du dipôle élémentaire en 3D.

Antenne filiforme

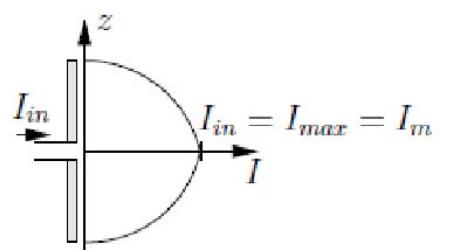
La distribution du courant dépend de h



$$h < \lambda/10$$

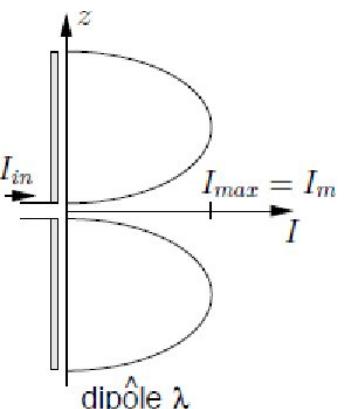
$$\bar{E}_{\theta_{court}} = j\omega\mu I_{in}h \frac{e^{-j\beta r}}{8\pi r} \sin\theta$$

$$F_{a_{court}} = \sin\theta$$



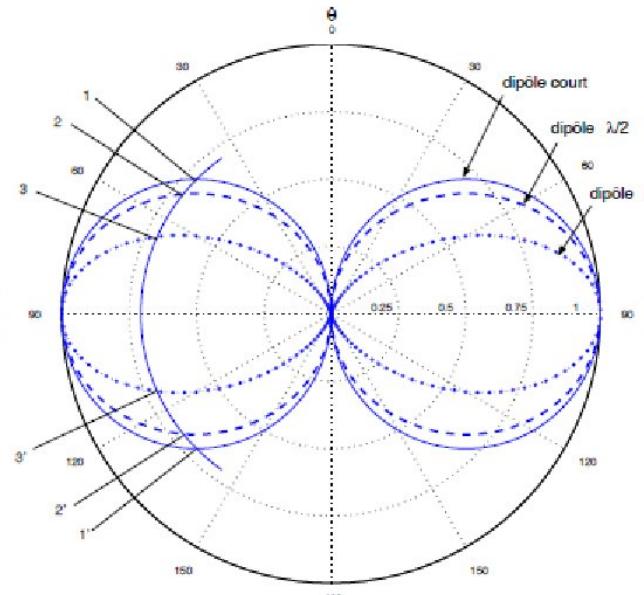
$$\bar{E}_\theta = j\eta_0 I_{in} \frac{e^{-j\beta r}}{2\pi r} \frac{\cos(\frac{\pi}{2} \cos\theta)}{\sin\theta}$$

$$F_{a_{dip/2}}(\theta, \phi) = \frac{\cos(\frac{\pi}{2} \cos\theta)}{\sin\theta}$$



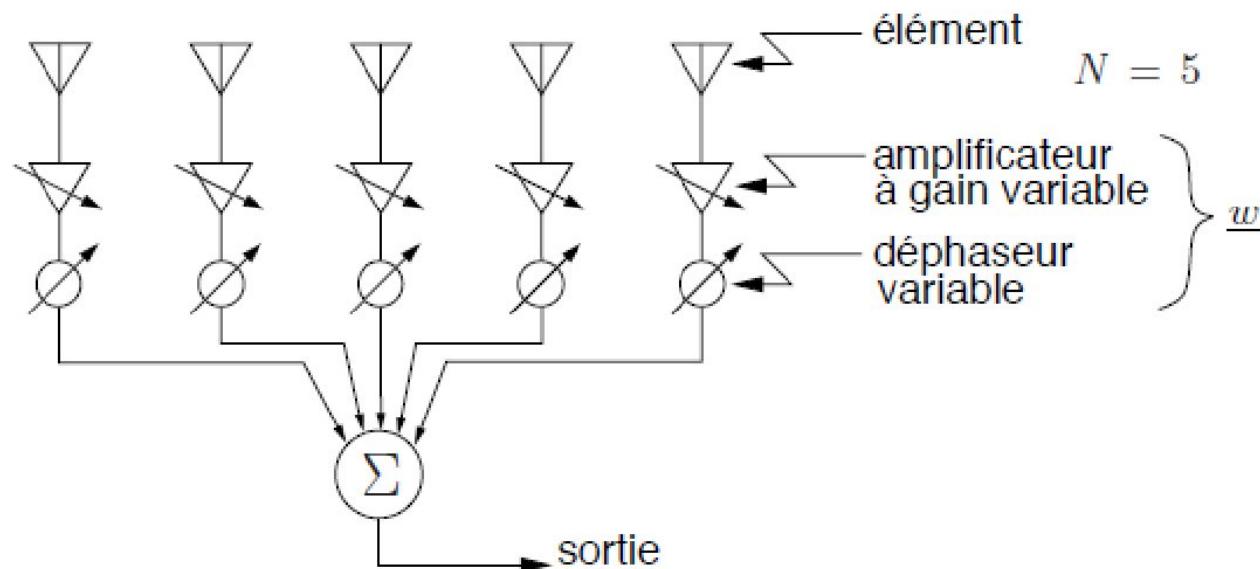
$$\bar{E}_\theta = j\eta_0 I_m \frac{e^{-j\beta r}}{\pi r} \left(\frac{\cos(\frac{\pi}{2} \cos\theta)}{\sin\theta} \right) \cos(\frac{\pi}{2} \cos\theta)$$

$$Fa = \left(\frac{\cos(\frac{\pi}{2} \cos\theta)}{\sin\theta} \right) \cos(\frac{\pi}{2} \cos\theta)$$



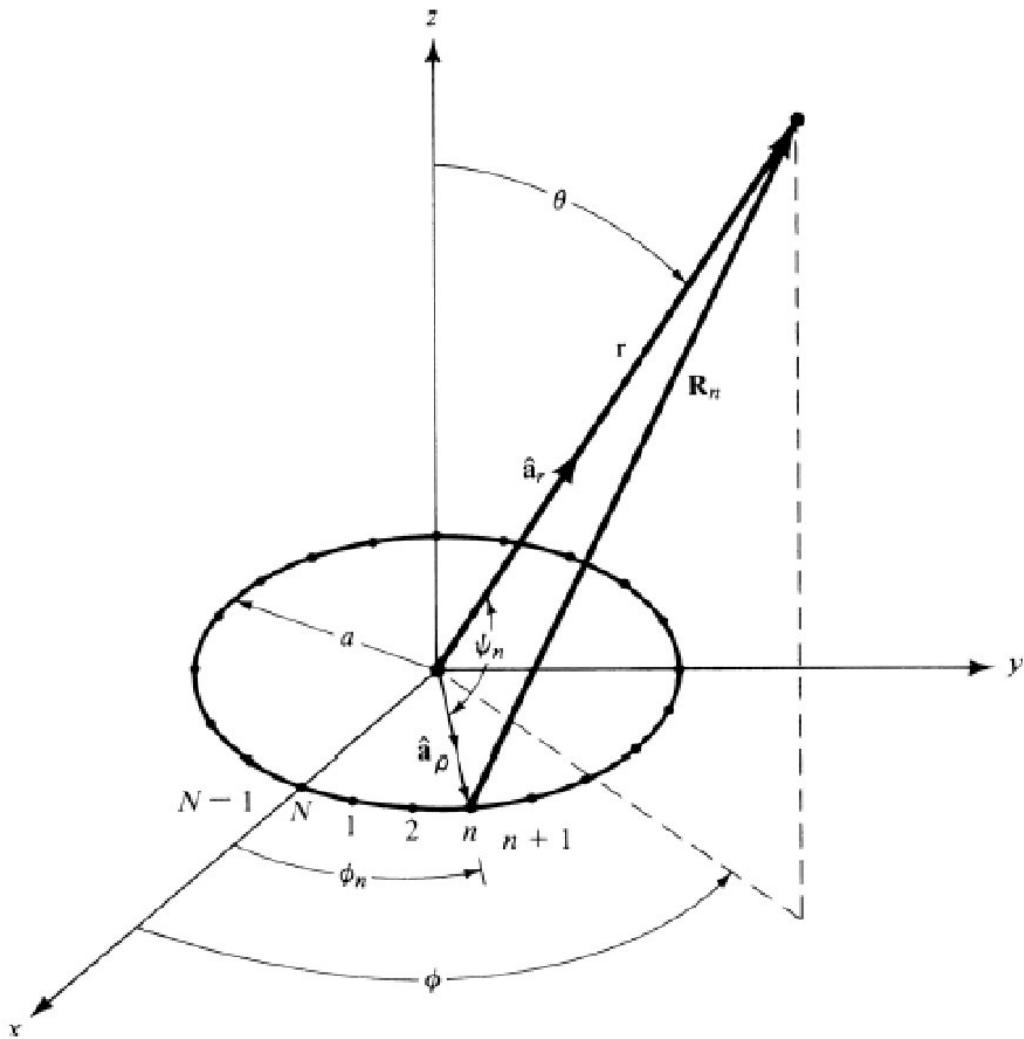
Antennes-réseaux

Une antenne-réseau est un ensemble d'antennes *identiques* (appelées “éléments”) distribuées dans un espace, de façon uniforme ou non, et reliées entre elles au niveau du récepteur. Ce dernier utilise donc, à un même instant, les données provenant de tous les éléments pour ne fournir qu'une seule donnée de sortie.



Antenne-réseau en réception

Réseau circulaire



Geometry of an N -element circular array.

Réseau circulaire

Referring to Figure 6.39, let us assume that N isotropic elements are equally spaced on the x - y plane along a circular ring of the radius a . The normalized field of the array can be written as

$$E_n(r, \theta, \phi) = \sum_{n=1}^N a_n \frac{e^{-jkR_n}}{R_n} \quad (6-112)$$

where R_n is the distance from the n th element to the observation point. In general

$$R_n = \sqrt{r^2 + a^2 - 2ar \cos(\psi_n)} \quad (6-112a)$$

$$k = \frac{2\pi}{\lambda}$$

a_n = excitation coefficients (amplitude and phase) of n th element

which for $r \gg a$ reduces to

$$R_n \simeq r - a \cos \psi_n = r - a(\hat{\mathbf{a}}_r \cdot \hat{\mathbf{a}}_r) = r - a \sin \theta \cos(\phi - \phi_n) \quad (6-112b)$$

Réseau circulaire

Thus (6-112) reduces, assuming that for amplitude variations $R_n \simeq r$, to

$$E_n(r, \theta, \phi) = \frac{e^{-jkr}}{r} \sum_{n=1}^N a_n e^{+jka \sin \theta \cos(\phi - \phi_n)} \quad (6-113)$$

where

a_n = excitation coefficients (amplitude and phase) of n th element

$\phi_n = 2\pi \left(\frac{n}{N} \right)$ = angular position of n th element on x - y plane

In general, the excitation coefficient of the n th element can be written as

$$a_n = I_n e^{j\alpha_n} \quad (6-114)$$

where

I_n = amplitude excitation of the n th element

α_n = phase excitation (relative to the array center) of the n th element

With (6-114), (6-113) can be expressed as

$$E_n(r, \theta, \phi) = \frac{e^{-jkr}}{r} [\text{AF}(\theta, \phi)] \quad (6-115)$$

where

$$\text{AF}(\theta, \phi) = \sum_{n=1}^N I_n e^{j[k a \sin \theta \cos(\phi - \phi_n) + \alpha_n]} \quad (6-115a)$$

Réseau circulaire

Equation (6-115a) represents the array factor of a circular array of N equally spaced elements. To direct the peak of the main beam in the (θ_0, ϕ_0) direction, the phase excitation of the n th element can be chosen to be

$$\alpha_n = -ka \sin \theta_0 \cos(\phi_0 - \phi_n) \quad (6-116)$$

Thus the array factor of (6-115a) can be written as

$$AF(\theta, \phi) = \sum_{n=1}^N I_n e^{jka[\sin \theta \cos(\phi - \phi_n) - \sin \theta_0 \cos(\phi_0 - \phi_n)]}$$

Réseau circulaire

```
# === Constantes ===
carrierFreq = 2.45e9
c = 3e8
lambda_ = c / carrierFreq
k = 2 * np.pi / lambda_

# === Réseau multi-anneaux ===
rings = 4
radii = np.linspace(0.2 * lambda_, 1.7 * lambda_, rings)
elements_per_ring = [3, 4, 5, 8]
assert len(radii) == len(elements_per_ring)

# === Formation de faisceau ===
theta0deg = 45
theta0 = np.deg2rad(theta0deg)
phi0 = 0
```

Diagramme de rayonnement
90°

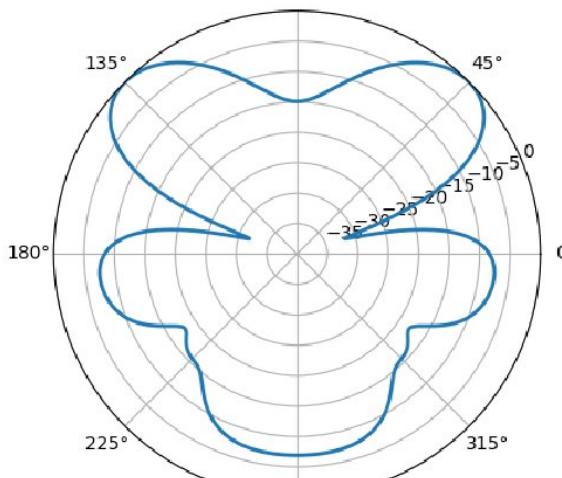
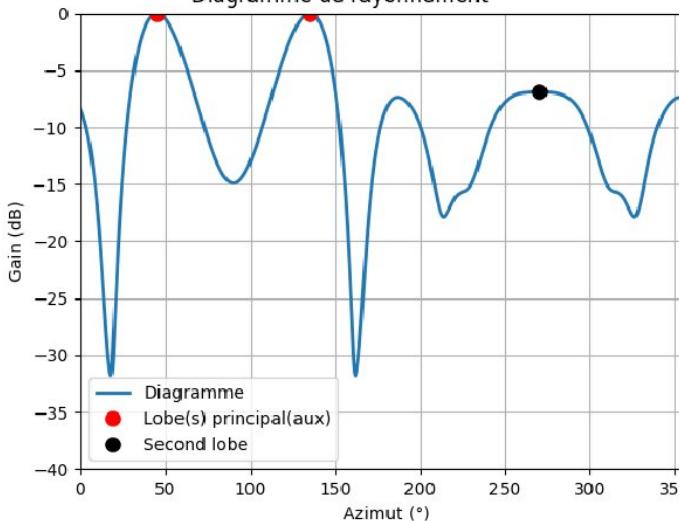


Diagramme de rayonnement



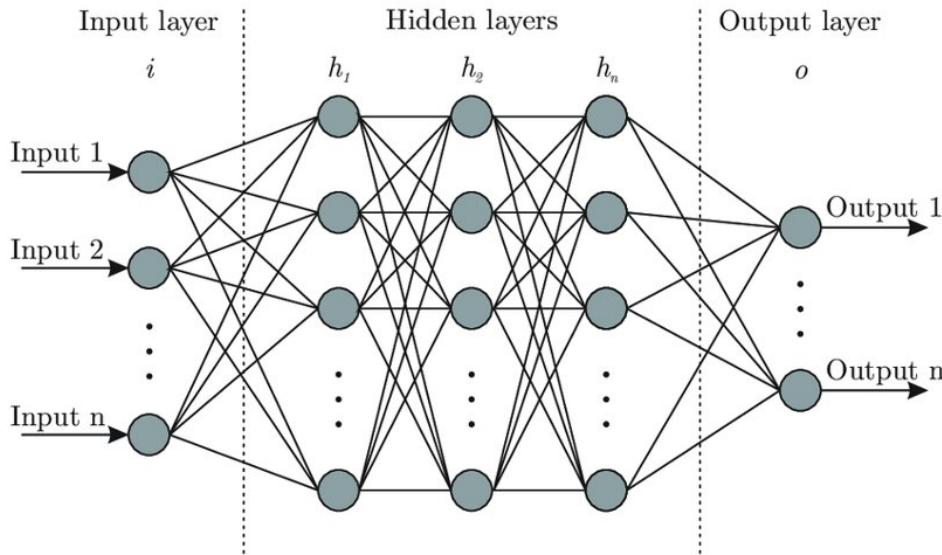
Modèles IA

Objectif

Design d'un réseau d'antennes circulaire correspondant à des spécifications bien définies:

- Gain du lobe principal
- Gain du lobe secondaire
- HPBW

Modèle FFNN (Feedforward Neural Network)



L'architecture FFNN est composée de couches interconnectées

Chaque couche contient plusieurs neurones artificiels

La couche d'entrée transmet les données aux couches cachées, qui à leur tour transmettent les données à la couche de sortie

Les biais et les poids sont ajustés de manière répétée jusqu'à ce que la sortie du réseau corresponde à la sortie correcte

Modèle FFNN (Feedforward Neural Network)

◆ Cas général – réseau à L couches :

◆ Propagation à travers chaque couche :

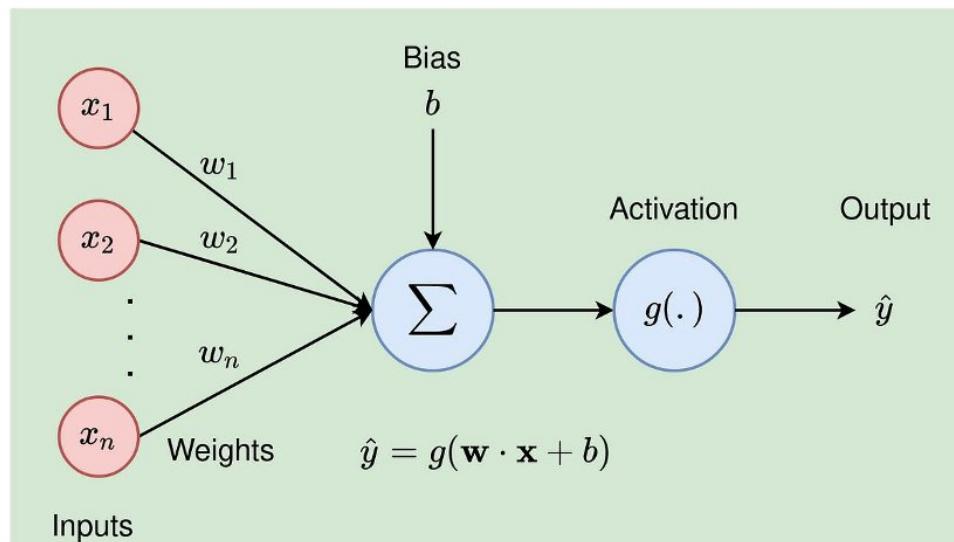
Pour la couche $l \in \{1, 2, \dots, L\}$:

$$\mathbf{z}^{(l)} = \mathbf{W}^{(l)} \mathbf{a}^{(l-1)} + \mathbf{b}^{(l)}$$

$$\mathbf{a}^{(l)} = g^{(l)}(\mathbf{z}^{(l)})$$

où :

- $\mathbf{x} = \mathbf{a}^{(0)}$ est le vecteur d'entrée,
- $\mathbf{W}^{(l)}$ est la matrice des **poids** de la couche l ,
- $\mathbf{b}^{(l)}$ est le **biais** de la couche l ,
- $g^{(l)}$ est la **fonction d'activation** de la couche l ,
- $\mathbf{a}^{(l)}$ est la **sortie (activation)** de la couche l ,
- $\mathbf{y} = \mathbf{a}^{(L)}$ est la **sortie finale** du réseau.



◆ Expression complète entrée → sortie :

$$\mathbf{y} = \mathbf{a}^{(L)} = g^{(L)} \left(\mathbf{W}^{(L)} g^{(L-1)} \left(\cdots g^{(1)} \left(\mathbf{W}^{(1)} \mathbf{x} + \mathbf{b}^{(1)} \right) \cdots + \mathbf{b}^{(L-1)} \right) + \mathbf{b}^{(L)} \right)$$

Modèle FFNN (Feedforward Neural Network)

$$z = \sum_i w_i x_i + b$$

Sans fonction d'activation, la sortie serait simplement z , une **combinaison linéaire**.

La fonction d'activation applique une transformation non linéaire à z :

$$y = f(z)$$

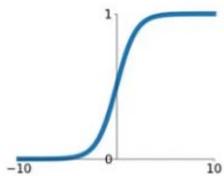
Cela permet au réseau d'apprendre des **modèles complexes et non linéaires**.

Chaque couche peut maintenant "composer" des non-linéarités, ce qui rend le réseau capable de **modéliser des fonctions très compliquées**.

Activation Functions

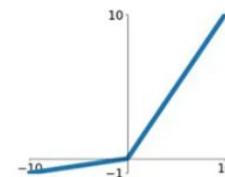
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



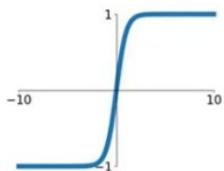
Leaky ReLU

$$\max(0.1x, x)$$



tanh

$$\tanh(x)$$

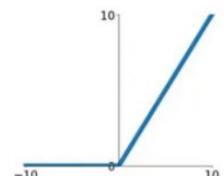


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

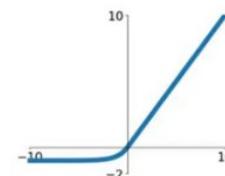
ReLU

$$\max(0, x)$$



ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Modèle FFNN (Feedforward Neural Network)

Le **training** (**entraînement**) dans un **FFNN** (**Feedforward Neural Network**) consiste à ajuster les poids et biais du réseau afin de minimiser une **fonction de perte** (**loss**), en comparant les sorties prédictives par le modèle avec les sorties réelles (cibles).

Étapes principales du training d'un FFNN :

1. Propagation avant (forward pass) :

Les données d'entrée passent à travers les couches du réseau. À chaque couche, une combinaison linéaire des poids, biais et fonction d'activation est appliquée pour produire une sortie.

$$y = f(Wx + b)$$

2. Calcul de la perte (loss) :

On calcule l'erreur entre la sortie prédictive et la vraie sortie à l'aide d'une fonction de perte (ex : MAE, MSE, cross-entropy...).

3. Rétropropagation (backpropagation) :

L'erreur est propagée en arrière dans le réseau pour calculer le **gradient** de la perte par rapport aux poids et biais, en utilisant la **dérivée de la fonction de perte**.

4. Mise à jour des poids (gradient descent) :

Les poids sont mis à jour dans le sens opposé au gradient pour réduire l'erreur :

$$w_{new} = w_{old} - \text{learning rate} \times \frac{\partial \text{Loss}}{\partial w}$$

Modèle FFNN (Feedforward Neural Network)

Rôle du batch size, epoch et learning rate :

1. Batch size (taille du lot) :

- C'est le **nombre d'exemples de données traités ensemble** avant de mettre à jour les poids.
- Trois cas :
 - **Batch Gradient Descent** : batch = taille totale du dataset (entraînement lent, mais stable).
 - **Stochastic Gradient Descent (SGD)** : batch = 1 (rapide mais bruité).
 - **Mini-batch Gradient Descent** : batch = 16, 32, 64... (compromis entre vitesse et stabilité).

3. Learning rate (taux d'apprentissage) :

- Contrôle la **vitesse de mise à jour des poids**.
- Valeur typique : 0.01, 0.001, etc.

✓ Effets du learning rate :

- Trop petit → convergence lente, voire stagnation.
- Trop grand → instabilité, oscillations, échec de la convergence.

2. Epoch :

- Une **epoch** = un **passage complet sur tout le dataset**.
- En général, on fait **plusieurs epochs** pour que le modèle apprenne correctement.

✓ Effets de l'epoch :

- Trop peu d'epochs → **sous-apprentissage (underfitting)**.
- Trop d'epochs → **sur-apprentissage (overfitting)** si on n'utilise pas de régularisation.

➡ Exemple synthétique :

Imaginons que tu as un dataset de 1000 exemples, un batch size = 100 et epochs = 200.

- ➡ À chaque epoch, le réseau verra les 1000 exemples (divisés en 10 batches de 100).
- ➡ À chaque batch, le modèle effectue une forward pass, calcule la loss, applique la backpropagation et met à jour les poids.
- ➡ Ce processus est répété 200 fois (200 epochs), ce qui signifie que chaque exemple du dataset sera utilisé 200 fois pour entraîner le modèle.

Modèle FFNN (Feedforward Neural Network)

Structure du code :

- Génération du dataset
- Training
- Test

Exemple de fonctions d'erreur pour tester le modèle :

Mean Absolute Error (*MAE*)

$$MAE = \frac{1}{M} \sum_{i=1}^M |y_{real,i} - y_{pred,i}|$$

Mean Square Error (*MSE*)

$$MSE = \frac{1}{M} \sum_{i=1}^M (y_{real,i} - y_{pred,i})^2$$

Mean Absolute Percentage Error (*MAPE*)

$$MAPE = \frac{100}{M} \sum_{i=1}^M \left| \frac{y_{real,i} - y_{pred,i}}{y_{real,i}} \right|$$

Avec :

y_{real} : Valeur réelle

y_{pred} : Valeur prédite par le modèle

M : Nombre d'échantillons utilisés dans le test

Algorithmes d'optimisation mét-heuristiques

GA : Genetic Algorithm

PSO : Particle Swarm Optimization

ABC : Artificial Bee Colony

COA : Coyote Optimization Algorithm

ACO : Ant Colony Optimization

GWO : Grey Wolf Optimizer

FA : Firefly Algorithm

Modèle FFNN (Feedforward Neural Network)

dataset :

```
Minput[0, sample_idx] = main_lobe_gain  
Minput[1, sample_idx] = true_SSL_gain  
Minput[2, sample_idx] = HPBW  
Minput[3, sample_idx] = theta0deg
```

```
Moutput[:, sample_idx] = elements_per_ring
```

training :

- 3 couches cachées
- Chaque couche contient 10 neurones
- Nombre d'epochs : 100
- Batch size : 64
- Learning rate : 0.001
- Algorithme de training : ADAM
- Fonction d'activation : RELU

```
nb_samples = 50000  
max_rings = 5  
max_elements = 10  
theta0_max_deg = 180  
carrierFreq = 2.45e9  
c = 3e8  
lambda_ = c / carrierFreq  
k = 2 * np.pi / lambda_  
r0 = 0.2 * lambda_  
delta_r = 0.5 * lambda_  
phi0 = 0
```

Modèle FFNN (Feedforward Neural Network)

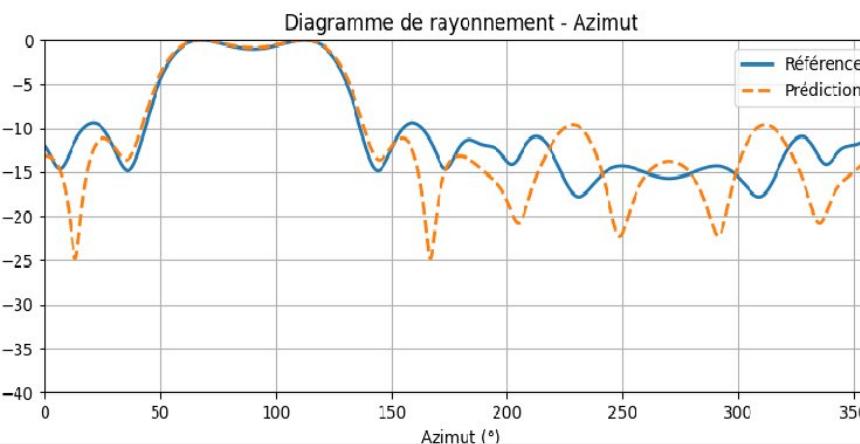
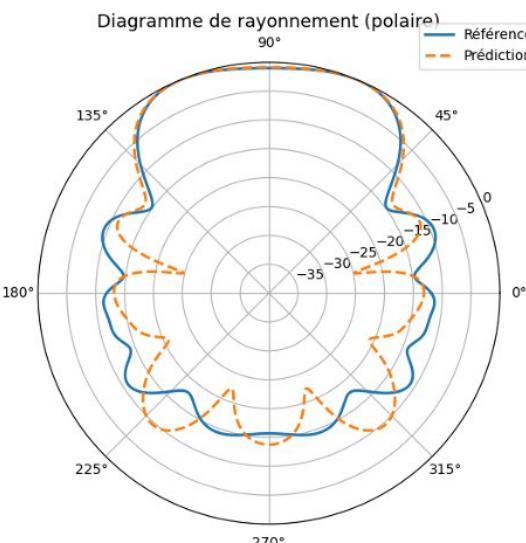
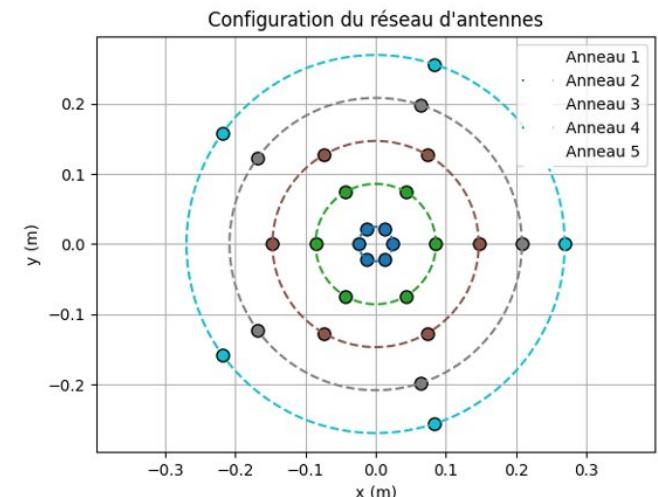
Test :

On veut faire le design d'un réseau d'antenne circulaire correspondant aux spécifications suivantes :

Gain du lobe principal : 28.94 dB
 True Side Lobe Gain : -9.37 dB
 HPBW : 74.95°
 theta0deg : 112.36°



Architecture prédictive par le modèle : Architecture prédictive : [6 6 6 5 5]



--- Performances Prédiction ---
 Gain du lobe principal : 28.94 dB
 True Side Lobe Gain : -9.58 dB
 HPBW : 76.76°

--- Erreurs absolues (MAE) entre prédiction et référence
 Erreur Gain Lobe Principal : 0.00 dB
 Erreur True SSL Gain : 0.21 dB
 Erreur HPBW : 1.80 degrés