

## RESOLUTION D'UN SYSTEME D'EQUATIONS

### 1. Système d'équations linéaires

#### 1.1. Méthode de Gauss-Jordan

##### 1.1.1. Exemple : système linéaire d'ordre 4

On considère un système d'équations linéaires d'ordre 4, que l'on écrit sous forme matricielle :

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \quad \text{ou} \quad A\vec{X} = \vec{B}$$

où  $A$  est une matrice carrée supposée inversible de rang 4,  $\vec{B}$  est un vecteur et  $\vec{X}$  est le vecteur inconnue. La méthode de Gauss – Jordan (aussi appelée méthode du pivot) consiste à prendre successivement chaque ligne comme ligne pivot, le pivot étant alors le premier élément non nul de la ligne (en général l'élément diagonal mais pas toujours). On divise alors toute la ligne  $i$  à laquelle appartient le pivot par celui-ci. Ensuite, pour les autres lignes  $j \neq i$ , on retranche la ligne  $i$  multipliée par  $a_{ji}$ . Ceci a pour effet de rendre nul le  $i^{\text{ème}}$  terme de chaque ligne. Les mêmes opérations sont effectuées simultanément sur le vecteur  $\vec{B}$ . Lorsque toutes les lignes ont été prise pour ligne pivot, la matrice  $A$  doit être transformée en matrice identité, et les solutions se lisent directement dans le vecteur  $\vec{B}$ . En prenant l'exemple ci-dessus, on aurait :

$$\ell_1^{(1)} = \ell_1 / a_{11} \quad \begin{bmatrix} 1 & a_{12}^{(1)} & a_{13}^{(1)} & a_{14}^{(1)} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} b_1^{(1)} \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}$$

puis, pour les lignes  $j \neq 1$  :

$$\ell_j^{(1)} = \ell_j - \ell_1^{(1)} \times a_{j1} \quad \begin{bmatrix} 1 & a_{12}^{(1)} & a_{13}^{(1)} & a_{14}^{(1)} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & a_{24}^{(1)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} & a_{34}^{(1)} \\ 0 & a_{42}^{(1)} & a_{43}^{(1)} & a_{44}^{(1)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} b_1^{(1)} \\ b_2^{(1)} \\ b_3^{(1)} \\ b_4^{(1)} \end{bmatrix}$$

On effectue ensuite les mêmes opérations en prenant la deuxième ligne comme ligne pivot :

$$\ell_2^{(2)} = \ell_2^{(1)} / a_{22}^{(1)} \quad \begin{bmatrix} 1 & a_{12}^{(1)} & a_{13}^{(1)} & a_{14}^{(1)} \\ 0 & 1 & a_{23}^{(2)} & a_{24}^{(2)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} & a_{34}^{(1)} \\ 0 & a_{42}^{(1)} & a_{43}^{(1)} & a_{44}^{(1)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} b_1^{(1)} \\ b_2^{(2)} \\ b_3^{(1)} \\ b_4^{(1)} \end{bmatrix}$$

pour les lignes  $j \neq 2$  :

$$\ell_j^{(2)} = \ell_j^{(1)} - \ell_2^{(2)} \times a_{j2}^{(1)} \quad \begin{bmatrix} 1 & 0 & a_{13}^{(2)} & a_{14}^{(2)} \\ 0 & 1 & a_{23}^{(2)} & a_{24}^{(2)} \\ 0 & 0 & a_{33}^{(2)} & a_{34}^{(2)} \\ 0 & 0 & a_{43}^{(2)} & a_{44}^{(2)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} b_1^{(2)} \\ b_2^{(2)} \\ b_3^{(2)} \\ b_4^{(2)} \end{bmatrix}$$

En poursuivant jusqu'à la quatrième ligne, on obtient directement la solution des équations sous la forme :

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} b_1^{(4)} \\ b_2^{(4)} \\ b_3^{(4)} \\ b_4^{(4)} \end{bmatrix}$$

### 1.1.2. Cas général

On a un système d'équation linéaire d'ordre  $n$  et on écrit l'équation matricielle sous la forme :

$$A\vec{X} = \vec{B}$$

où la matrice  $A$  est une matrice carrée de rang  $n$  supposée inversible. A la  $k^{\text{ième}}$  étape de la méthode de Gauss – Jordan, la matrice  $A$  aura la forme :

$$A^{(k)} = \begin{bmatrix} 1 & 0 & \cdots & 0 & a_{1,k+1}^{(k)} & \cdots & a_{1,n}^{(k)} \\ 0 & & & \vdots & & & \\ \vdots & & & 0 & \vdots & & \vdots \\ 0 & \cdots & 0 & 1 & & & \\ & & & 0 & a_{k+1,k+1}^{(k)} & \cdots & a_{k+1,n}^{(k)} \\ \vdots & & & \vdots & \vdots & & \vdots \\ 0 & \cdots & 0 & a_{n,k+1}^{(k)} & \cdots & & a_{n,n}^{(k)} \end{bmatrix}$$

c'est-à-dire, pour  $j \leq k$  :

$$a_{i,j}^{(k)} = 1 \text{ si } i = j$$

$$a_{i,j}^{(k)} = 0 \text{ si } i \neq j$$

On peut établir la procédure de résolution par la méthode de Gauss – Jordan, en utilisant une matrice de dimension  $(n, n + 1)$  telle que :

$$a_{i,j}^{(0)} = a_{i,j} \text{ pour } j \leq n, \text{ et } a_{i,n+1}^{(0)} = b_i$$

La procédure sera alors, pour les différentes étapes  $k = 0 \rightarrow n - 1$  :

$$* \left. \begin{array}{l} i = k + 1 \\ j = k + 1 \rightarrow n + 1 \end{array} \right\} a_{k+1,j}^{(k+1)} = \frac{a_{k+1,j}^{(k)}}{a_{k+1,k+1}^{(k)}}$$

$$* \left. \begin{array}{l} i = 1 \rightarrow n \text{ (} i \neq k + 1 \text{)} \\ j = k + 1 \rightarrow n + 1 \end{array} \right\} a_{i,j}^{(k+1)} = a_{i,j}^{(k)} - a_{i,k+1}^{(k)} a_{k+1,j}^{(k+1)}$$

Les solutions étant ensuite :

$$x_i = a_{i,n+1}^{(n)} \quad i = 1 \rightarrow n$$

Dans le cas où on a apparition d'un pivot nul (ou très petit), on peut inverser les colonnes ou les lignes ( $i, j > k + 1$ ), de façon à mettre le plus grand élément pour pivot.

## 1.2. Méthode de Gauss

La méthode de Gauss est également appelée méthode de triangularisation. Elle ressemble à la méthode de Gauss – Jordan, mais on ne l'applique pas à toutes les lignes, seulement celles d'indices  $i = k + 1 \rightarrow n$ . On aura à la fin une matrice triangulaire supérieure, dont la diagonale est constituée de 1 :

$$\begin{bmatrix} 1 & a_{1,2}^{(n)} & \cdots & a_{1,n-1}^{(n)} & a_{1,n}^{(n)} \\ 0 & 1 & & a_{2,n-1}^{(n)} & a_{2,n}^{(n)} \\ \vdots & & & \vdots & \vdots \\ 0 & \cdots & & 1 & a_{n-1,n}^{(n)} \\ 0 & \cdots & & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} b_1^{(n)} \\ b_2^{(n)} \\ \vdots \\ b_{n-1}^{(n)} \\ b_n^{(n)} \end{bmatrix}$$

Il faut ensuite rajouter une procédure de résolution d'un système triangulaire. On aura ainsi :

$$\begin{cases} x_n = a_{n,n+1}^{(n)} \\ x_{n-1} = a_{n-1,n+1}^{(n)} - a_{n-1,n}^{(n)} x_n \\ \vdots \\ x_i = a_{i,n+1}^{(n)} - \sum_{m=i+1}^n a_{i,m}^{(n)} x_m \end{cases}$$

Pour les grands systèmes, la méthode de Gauss nécessite moins de calculs que celles de Gauss – Jordan. Par contre, pour les petits systèmes il vaut mieux utiliser la méthode de Gauss – Jordan car on a le résultat en une seule étape.

### 1.3. Méthode de Cholesky

On désire résoudre le système d'équation linéaires, mis sous forme matricielle :

$$A\vec{X} = \vec{B}$$

où  $A$  est une matrice symétrique (par exemple une matrice de déformation en mécanique des milieux continus ou en mécanique des fluides) :

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & & \\ \vdots & & \ddots & \vdots \\ a_{n1} & \cdots & & a_{nn} \end{bmatrix} \quad a_{ij} = a_{ji}$$

La méthode de Cholesky consiste à mettre la matrice  $A$  sous la forme  $A = RR^t$  où  $R$  est une matrice triangulaire inférieure. On aura :

$$R = \begin{bmatrix} r_{11} & 0 & \cdots & 0 \\ r_{21} & r_{22} & & \vdots \\ \vdots & & \ddots & 0 \\ r_{n1} & \cdots & & r_{nn} \end{bmatrix} \quad R^t = \begin{bmatrix} \hat{r}_{11} & \hat{r}_{12} & \cdots & \hat{r}_{1n} \\ 0 & & & \vdots \\ \vdots & \ddots & & \\ 0 & \cdots & 0 & \hat{r}_{nn} \end{bmatrix}$$

#### 1.3.1. Calcul de la matrice $R$

De par la propriétés de la multiplication des matrices, on peut déterminer facilement le premier coefficient de la matrice  $R$  :

$$a_{11} = r_{11}^2 \quad \Rightarrow \quad r_{11} = \pm\sqrt{a_{11}}$$

On choisit :  $r_{11} = \sqrt{a_{11}}$

On peut calculer le  $j^{\text{ième}}$  élément de la première ligne de la matrice  $A$  :

$$a_{1j} = \sum_{k=1}^n r_{1k} \hat{r}_{kj} = \sum_{k=1}^n r_{1k} r_{jk} \quad (R^t \text{ étant la transposée de } R)$$

Les éléments  $r_{1k}$  étant nuls pour  $k \geq 2$ , on obtient :  $a_{1j} = r_{11} r_{j1}$

$$\text{D'où : } r_{j1} = \frac{a_{1j}}{\sqrt{a_{11}}} \quad \text{pour } j = 1 \rightarrow n \quad (\text{la première colonne est connue})$$

(Il faut bien sûr que  $a_{11} \neq 0$ )

On peut ensuite effectuer un raisonnement par récurrence. On suppose que les  $j-1$  premières colonnes de la matrice  $R$  ont été calculées. On a alors :

$$a_{jj} = \sum_{k=1}^n r_{jk} \hat{r}_{kj} = \sum_{k=1}^n r_{jk}^2 = \sum_{k=1}^j r_{jk}^2 \quad (r_{jk} = 0 \text{ pour } k \geq j+1)$$

On peut décomposer en  $a_{jj} = \sum_{k=1}^{j-1} r_{jk}^2 + r_{jj}^2$ , où seul le terme  $r_{jj}^2$  est inconnu. D'où :

$$r_{jj} = \sqrt{a_{jj} - \sum_{k=1}^{j-1} r_{jk}^2}$$

De même, on va avoir :

$$a_{kj} = \sum_{m=1}^n r_{km} \hat{r}_{mj} = \sum_{m=1}^n r_{km} r_{jm} = \sum_{m=1}^j r_{km} r_{jm} \quad (k > j)$$

$$a_{kj} = \sum_{m=1}^{j-1} r_{km} r_{jm} + r_{kj} r_{jj} \quad \text{où seul le terme } r_{kj} \text{ est inconnu}$$

$$\text{Donc : } r_{kj} = \frac{1}{r_{jj}} \left( a_{kj} - \sum_{m=1}^{j-1} r_{km} r_{jm} \right) \quad \text{pour } k = j+1 \rightarrow n$$

On peut donc ainsi déterminer tout les éléments de la matrice triangulaire  $R$ .

### 1.3.2. Résolution du système

On sépare l'équation matricielle initiale  $A\vec{X} = RR^t \vec{X} = \vec{B}$  en deux équations distinctes grâce aux matrices triangulaires :

$$R\vec{Z} = \vec{B} \quad \Rightarrow \quad \vec{Z}$$

$$R^t \vec{X} = \vec{Z} \quad \Rightarrow \quad \vec{X}$$

Les équations ci-dessus sont aisément solvables à l'aide d'une procédure de résolution de système triangulaire (voir méthode de Gauss).

Bien évidemment, cette méthode n'est valable que si les  $r_{jj} \neq 0$  et si tous les pivots sont non nuls.

### 1.4. Méthode de Crout

On veut également résoudre le système matriciel  $A\vec{X} = \vec{B}$ , mais pour cette méthode la matrice  $A$  n'a pas besoin d'être symétrique. Par contre, les pivots doivent être la aussi tous non nuls.

#### 1.4.1. Principe de la méthode

Toute matrice  $A$  peut être décomposée en un produit de matrice, sous la forme  $A = GD$ , où :

$$G = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ g_{21} & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ g_{n1} & \cdots & g_{n,n-1} & 1 \end{bmatrix} \quad D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ 0 & d_{22} & \cdots & d_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & d_{n,n} \end{bmatrix}$$

C'est pour cela que cette méthode est également appelée *décomposition L-R* (Left-Right). De même que pour la méthode précédente, on pourra décomposer l'équation matricielle  $A\vec{X} = GD\vec{X} = \vec{B}$ , d'où :

$$\begin{cases} G\vec{Z} = \vec{B} \\ D\vec{X} = \vec{Z} \end{cases}$$

Cette méthode est particulièrement intéressante lorsqu'on doit résoudre plusieurs fois le même système, avec des seconds membres différents. La décomposition étant toujours la même, il ne reste plus qu'à résoudre simplement deux systèmes triangulaires.

#### 1.4.2. Détermination des matrices $G$ et $D$

Pour une compréhension rapide de la méthode, il convient de se rappeler que pour  $j > i$  on a  $g_{ij} = d_{ji} = 0$ . On peut alors calculer :

$$* \begin{cases} i = 1 \\ j = 1 \rightarrow n \end{cases} \Rightarrow \begin{cases} d_{1j} = a_{1j} \\ g_{j1} = a_{j1} / d_{11} \end{cases}$$

$$\begin{aligned}
 * \begin{cases} i = 2 \rightarrow n \\ j = 1 \rightarrow n \end{cases} & \Rightarrow a_{ij} = \sum_{k=1}^n g_{ik} d_{kj} = \sum_{k=1}^i g_{ik} d_{kj} & (g_{ik} = 0 \text{ si } k > i) \\
 & a_{ij} = \sum_{k=1}^{i-1} g_{ik} d_{kj} + d_{ij} & (g_{ii} = 1) \\
 & a_{ji} = \sum_{k=1}^n g_{jk} d_{ki} = \sum_{k=1}^i g_{jk} d_{ki} & (d_{ki} = 0 \text{ si } k > i) \\
 & a_{ji} = \sum_{k=1}^{i-1} g_{jk} d_{ki} + g_{ji} d_{ii}
 \end{aligned}$$

En partant de  $i = 1$ , on peut déterminer successivement les différents termes  $d_{ij}$  et  $g_{ij}$  en incrémentant l'indice  $i$  :

$$\begin{aligned}
 d_{ij} &= a_{ij} - \sum_{k=1}^{i-1} g_{ik} d_{kj} \\
 g_{ji} &= \frac{1}{d_{ii}} \left( a_{ji} - \sum_{k=1}^{i-1} g_{jk} d_{ki} \right)
 \end{aligned}$$

## 1.5. Méthodes itératives

### 1.5.1. Principe des méthodes

L'équation matricielle  $A\vec{X} = \vec{B}$  est mise sous la forme  $M\vec{X} = \vec{C}$  où  $M$  est une matrice dont les éléments diagonaux sont égaux à 1. Pour cela, on crée la matrice diagonale  $D$ , telle que :

$$D = \begin{bmatrix} \frac{1}{a_{11}} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{1}{a_{n,n}} \end{bmatrix}$$

et on multiplie l'équation initiale par cette matrice :  $DA\vec{X} = D\vec{B}$ . On a donc :

$$\begin{cases} M = DA \\ \vec{C} = D\vec{B} \end{cases}$$

La matrice  $M$  pourra être décomposée en :

$$M = I - T_I - T_S$$

où  $I$  est la matrice identité,  $T_I$  une matrice triangulaire inférieure et  $T_S$  une matrice triangulaire supérieure. On peut alors écrire l'équation matricielle :

$$(I - T_I - T_S)\vec{X} = \vec{C}$$

Les méthodes itératives consistent à partir d'un vecteur arbitraire  $\vec{X}^{(0)}$  et à construire une suite  $\vec{X}^{(n)}$ , telle que :

$$\lim_{n \rightarrow \infty} \vec{X}^{(n)} = \vec{X}^* \quad \text{où } \vec{X}^* \text{ est solution de l'équation } A\vec{X} = \vec{B}$$

A toute décomposition  $M = E - F$ , où la matrice  $E$  est une matrice inversible, on peut associer la méthode itérative suivante :

$$E\vec{X}^{(n+1)} = F\vec{X}^{(n)} + \vec{C}$$

### 1.5.2. Les différentes méthodes

Différentes méthodes peuvent être obtenues suivant la décomposition en matrices  $E$  et  $F$  choisie :

\* Méthode de Jacobi :

$$\left. \begin{array}{l} E = I \\ F = T_I + T_S \end{array} \right\} \quad \vec{X}^{(n+1)} = (T_I + T_S)\vec{X}^{(n)} + \vec{C}$$

\* Méthode de Gauss – Seïdel :

$$\left. \begin{array}{l} E = I - T_I \\ F = T_S \end{array} \right\} \quad \begin{aligned} (I - T_I)\vec{X}^{(n+1)} &= T_S\vec{X}^{(n)} + \vec{C} \\ \vec{X}^{(n+1)} &= (I - T_I)^{-1}T_S\vec{X}^{(n)} + (I - T_I)^{-1}\vec{C} \end{aligned}$$

### 1.5.3. Convergence des méthodes itératives

On va écrire, de façon générale, l'équation des méthodes itératives sous la forme suivante :

$$\vec{X}^{(n+1)} = G\vec{X}^{(n)} + \vec{H}$$

La méthode va converger si  $\lim_{n \rightarrow \infty} \vec{X}^{(n)} = \vec{X}^* = A^{-1}\vec{B}$ . A chaque itération, on peut calculer le vecteur erreur :

$$\vec{\varepsilon}^{(n)} = \vec{X}^{(n)} - A^{-1}\vec{B} = G\vec{X}^{(n-1)} + \vec{H} - A^{-1}\vec{B}$$



Lorsque  $n \rightarrow \infty$ , on a  $\vec{X} = G\vec{X} + \vec{H}$ , donc  $\vec{H} = \vec{X} - G\vec{X}$ . On obtient alors :

$$\vec{\varepsilon}^{(n)} = G\vec{X}^{(n-1)} + \vec{X} - G\vec{X} - A^{-1}\vec{B} \quad (\text{or } \vec{X} = A^{-1}\vec{B})$$

$$\vec{\varepsilon}^{(n)} = G(\vec{X}^{(n-1)} - A^{-1}\vec{B})$$

On a la relation de récurrence :  $\vec{\varepsilon}^{(n)} = G\vec{\varepsilon}^{(n-1)}$

D'où :  $\vec{\varepsilon}^{(n)} = G^n \vec{\varepsilon}^{(0)}$

Il y aura donc convergence lorsque  $\lim_{n \rightarrow \infty} \vec{\varepsilon}^{(n)} = 0, \forall \vec{\varepsilon}^{(0)}$  puisqu'on part d'une valeur initiale arbitraire et l'erreur initiale peut être relativement importante. Le comportement de la méthode va donc dépendre uniquement de la matrice  $G$ . Il y a deux conditions nécessaires et suffisantes pour avoir convergence :

1/- que le rayon spectral de  $G$  soit inférieur à 1 :

$$\sigma(G) = \sup_{j \in (1, \dots, n)} |\lambda_j| < 1 \quad \text{où les } \lambda_j \text{ sont les valeurs propres de } G$$

2/- qu'il existe une norme matricielle de  $G$  inférieure à 1 :

$$\|G\| < 1$$

En pratique, on peut aussi utiliser la condition suffisante suivante : il y a convergence des méthodes itératives si la matrice  $A$  est à diagonale dominante, c'est-à-dire si :

$$\forall i, |a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|$$

Si il y a convergence, la méthode de Gauss – Seïdel convergera plus vite que celle de Jacobi, mais elle demande un peu plus de calcul au départ (inversion de matrices).

#### 1.5.4. Facteur de relaxation

On considère la suite  $x_j^{(k)}$  des valeurs déterminées par une méthode itérative, où  $j$  correspond à l'indice de la composante du vecteur  $\vec{X}^{(k)}$  et  $k$  au rang de l'itération. On peut définir un coefficient de relaxation  $\mu$  permettant de minimiser les variations d'une itération à l'autre. Dans ce cas, au lieu de continuer le calcul à l'itération suivante ( $k+1$ ) avec la valeur  $x_j^{(k)}$ , on utilisera plutôt la valeur :

$$x_j^{*(k)} = x_j^{(k-1)} + \mu(x_j^{(k)} - x_j^{(k-1)})$$

Si  $\mu=1$ , le calcul reste bien évidemment inchangé. Si  $0 < \mu < 1$ , les variations d'une itération à l'autre sont amorties, ce qui aura pour effet de stabiliser le calcul et dans certains cas de faire converger un calcul sinon divergent. Si  $\mu > 1$ , la convergence sera accélérée si le processus est initialement stable, mais cela pourra faire diverger le calcul dans d'autres cas.

La valeur du coefficient de relaxation n'est pas forcément la même pour tout l'ensemble. En pratique, s'il y a divergence, on peut suivre l'évolution des  $x_j^{(k)}$  et on essaie différentes valeurs de  $\mu$  pour voir comment évoluent les  $x_j^{(k)}$  suivants, jusqu'à obtenir la convergence (si on y arrive).

## 1.6. Calcul de déterminants

### 1.6.1. Rappels

- \* La valeur  $D$  d'un déterminant est multipliée par  $k$  si on multiplie une ligne par  $k$ .
- \* La valeur  $D$  d'un déterminant est multipliée par  $-1$  si on intervertit deux lignes ou deux colonnes.
- \* La valeur  $D$  d'un déterminant est inchangée lorsqu'on ajoute à une ligne une combinaison linéaire d'autres lignes.

### 1.6.2. Calcul à l'aide de la méthode du pivot

Soit une matrice  $A$  de rang  $n$  dont on cherche le déterminant à l'aide de la méthode de triangularisation de Gauss – Jordan. A chaque étape, on divise la ligne pivot par l'élément pivot de façon à avoir 1 sur la diagonale. La valeur du déterminant sera aussi modifiée en conséquence. Ainsi, à la  $k^{\text{ième}}$  étape, la nouvelle valeur du déterminant sera :

$$D^{(k)} = \frac{D^{(k-1)}}{p_k}$$

où  $D^{(k-1)}$  est la valeur du déterminant à l'étape  $k-1$  et  $p_k$  est la valeur du pivot à l'étape  $k$ . Puisqu'à la dernière étape ( $n$ ), la matrice  $A$  est devenue une matrice triangulaire supérieure ne présentant que des 1 sur la diagonale, on aura  $D^{(n)} = 1$ . On peut retrouver le déterminant de la matrice  $A$  par la relation de récurrence sur les déterminants, en tenant compte d'un nombre éventuel  $r$  de permutations de lignes ou de colonnes en cas de pivots nuls :

$$D = (-1)^r \prod_{k=1}^n p_k$$

## 1.7. Conditionnement d'un système linéaire

### 1.7.1. Exemple

On considère un système de deux équations à deux inconnues (correspondant à

l'équation matricielle  $A\vec{X}=\vec{B}$ ) suivant, ainsi que ses solutions :

$$\begin{cases} x_1 + x_2 = -1 \\ 1,001x_1 + x_2 = 1 \end{cases} \Rightarrow \begin{cases} x_1 = 2000 \\ x_2 = -2001 \end{cases}$$

puis un nouveau système dont un des coefficients est différent du système précédent de un millièmes seulement :

$$\begin{cases} x_1 + x_2 = -1 \\ 1,002x_1 + x_2 = 1 \end{cases} \Rightarrow \begin{cases} x_1 = 1000 \\ x_2 = -1001 \end{cases}$$

On peut alors s'apercevoir que, pour une très légère différence dans un des coefficients, on trouve des solutions différentes d'un facteur 2.

On peut interpréter ceci graphiquement par le fait que ces deux équations représentent deux droites. On s'aperçoit alors que les vecteurs directeurs de ces droites  $\vec{V}_1 \begin{pmatrix} -1 \\ 1 \end{pmatrix}$  et  $\vec{V}_2 \begin{pmatrix} -1 \\ 1,001 \end{pmatrix}$  sont quasiment parallèles et donc que l'angle  $\alpha$  entre les droites est très faible :

$$|\sin \alpha| = \frac{\|\vec{V}_1 \wedge \vec{V}_2\|}{\|\vec{V}_1\| \|\vec{V}_2\|} = \frac{|\det A|}{\|\vec{V}_1\| \|\vec{V}_2\|} = 4,9975 \cdot 10^{-4} \Rightarrow \alpha \approx 5 \cdot 10^{-4} \text{ rad} \approx 0,03 \text{ deg}$$

La solution du système correspondant au point d'intersection de ces droites, il est facile de voir que la moindre imprécision sur les coefficients (et donc sur l'angle entre les droites) va entraîner une variation importante de la localisation du point d'intersection.

### 1.7.2. Généralisation

Dans un cas réel, il arrive souvent que des mesures de quantités physiques ne soient pas d'une précision meilleure que quelques pour cent. Une faible erreur dans la mesure des coefficients des équations peut donc induire une très forte erreur dans la détermination des solutions du problèmes. On peut également avoir des erreurs importantes lorsque l'on utilise la méthode de Gauss par exemple : les coefficients sont recalculés à chaque étape et peuvent être affectés par une erreur d'arrondi. Dans un cas normal, l'incidence sur les résultats reste faible, mais pas dans le cas où la matrice est mal conditionnée.

En généralisant l'exemple précédent au cas d'une matrice carré  $A$  de rang  $n$ , on peut définir un facteur de conditionnement  $\gamma$  permettant de savoir si un système est bien conditionné ou pas :

$$\gamma = \frac{|\det A|}{\prod_{i=1}^n \sqrt{\sum_{j=1}^n a_{i,j}^2}} \quad (0 \leq \gamma \leq 1)$$

Le système sera mal conditionné si  $\gamma \ll 1$ . Si on appelle  $\vec{X}^* = A^* \vec{B}$  la solution déterminée

numériquement (approchée), au lieu de  $\vec{X} = A^{-1}\vec{B}$ , on peut vérifier l'effet du mauvais conditionnement de deux façons :

- en comparant le produit  $A^*A$  à la matrice identité.
- en comparant  $(A^*)^{-1}$  à  $A$ .

Pour résoudre ce problème, on peut utiliser deux techniques simultanément. Il faut d'abord travailler avec le plus de précision possible (du type "réels" double – précision). On peut aussi utiliser la solution approchée  $\vec{X}^*$  dans une résolution de type itératif :

- on utilise  $\vec{X}^*$  pour calculer la valeur  $\vec{B}^* = A\vec{X}^*$
- on a alors les erreurs  $\Delta\vec{X} = \vec{X} - \vec{X}^*$  et  $\Delta\vec{B} = \vec{B} - \vec{B}^*$
- on écrit l'équation  $A\Delta\vec{X} = \Delta\vec{B}$  qui permet de déterminer  $\Delta\vec{X}$
- on calcule  $\vec{X} = \vec{X}^* + \Delta\vec{X}$

On peut ensuite continuer jusqu'à obtenir la précision voulue.

## 2. Système d'équations non – linéaires

On considère un système de  $n$  équations non – linéaires à  $n$  inconnues, que l'on écrit sous la forme :

$$\begin{cases} f_1(x_1, \dots, x_n) = 0 \\ \vdots \\ f_2(x_1, \dots, x_n) = 0 \end{cases}$$

On peut également écrire ce système sous forme d'équation matricielle :

$$\vec{F}(\vec{X}) = 0 \quad \text{où} \quad \vec{X} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \quad \text{et} \quad \vec{F} = \begin{pmatrix} f_1(\vec{X}) \\ \vdots \\ f_n(\vec{X}) \end{pmatrix}$$

### 2.1. Méthode itérative

La méthode itérative est similaire à la méthode des approximations successives utilisée pour la résolution d'une équation, mais appliquée ici à une équation matricielle. On met l'équation sous la forme :

$$\vec{X} = \vec{G}(\vec{X})$$

en posant par exemple  $\vec{G}(\vec{X}) = \vec{X} + \vec{F}(\vec{X})$ . On se donne un vecteur initial  $\vec{X}_0$  et on calcule  $\vec{X}_1 = \vec{G}(\vec{X}_0)$ , et ainsi de suite, jusqu'à avoir  $\vec{X}_n = \vec{G}(\vec{X}_{n-1})$  suffisamment proche de la solution  $\vec{X}^*$ .

De façon similaire au cas d'une seule équation, on aura convergence si au voisinage de la solution l'une ou l'autre des conditions suivantes est remplie :

$$\sum_{i=1}^n \left| \frac{\partial g_i}{\partial x_j} \right| < 1 \quad \text{pour } j = 1 \rightarrow n$$

$$\sum_{j=1}^n \left| \frac{\partial g_i}{\partial x_j} \right| < 1 \quad \text{pour } i = 1 \rightarrow n$$

## 2.2. Méthode de Newton

La méthode de Newton est similaire à celle utilisée pour la résolution d'une équation, mais appliquée ici au système :

$$\begin{cases} f_1(x_1, \dots, x_n) = 0 \\ \vdots \\ f_n(x_1, \dots, x_n) = 0 \end{cases} \quad \text{ou} \quad \bar{F}(\bar{X}) = 0$$

On considère un vecteur  $\bar{X}^{(0)} \begin{pmatrix} x_1^{(0)} \\ \vdots \\ x_n^{(0)} \end{pmatrix}$  proche de la solution  $\bar{X}^* \begin{pmatrix} x_1^* \\ \vdots \\ x_n^* \end{pmatrix}$ . On peut définir un

vecteur  $\bar{H} \begin{pmatrix} h_1 = x_1^* - x_1^{(0)} \\ \vdots \\ h_n = x_n^* - x_n^{(0)} \end{pmatrix}$ . La méthode de Newton consiste à remplacer chaque fonction  $f_i$

par son développement limité à l'ordre 1 (linéarisation) :

$$f_i(x_1^*, \dots, x_n^*) = 0 = f_i(x_1^{(0)}, \dots, x_n^{(0)}) + \sum_{j=1}^n h_j \frac{\partial f_i}{\partial x_j}(x_1^{(0)}, \dots, x_n^{(0)})$$

Puisque  $\bar{X}^* \begin{pmatrix} x_1^* \\ \vdots \\ x_n^* \end{pmatrix}$  est solution du système, on aura  $f_i(x_1^*, \dots, x_n^*) = 0, \forall i$ . On obtient alors un

système linéaire par rapport aux  $h_i$ , que l'on peut mettre sous forme matricielle :

$$\left[ \frac{\partial \bar{F}}{\partial \bar{X}}(\bar{X}^{(0)}) \right] \bar{H} = -\bar{F}(\bar{X}^{(0)})$$

où  $\left[ \frac{\partial \vec{F}}{\partial \vec{X}}(\vec{X}^{(0)}) \right] = J(\vec{X}^{(0)}) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \dots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}^{(0)}$  est la matrice Jacobienne de la fonction

$\vec{F}(\vec{X})$ . On peut alors écrire une méthode itérative sous la forme :

$$\vec{X}^{(k+1)} = \vec{X}^{(k)} + \vec{H}^{(k)}$$

$\vec{H}^{(k)}$  étant alors déterminé grâce à l'équation :

$$J(\vec{X})\vec{H} = -\vec{F}(\vec{X}) \quad \Rightarrow \quad \vec{H}^{(k)} = -J^{-1}(\vec{X}^{(k)})\vec{F}(\vec{X}^{(k)})$$

Chaque itération va nécessiter le calcul des  $n^2$  éléments de la matrice  $J$ , puis la résolution d'un système linéaire de rang  $n$  donnant le vecteur  $\vec{H}$ .

On peut montrer que la méthode de Newton peut être considérée comme une méthode de sur-itération, dont le coefficient est  $J^{-1}(\vec{X}^{(k)})$ . Dans ce cas, il n'est pas nécessaire de connaître avec exactitude les éléments de la matrice  $J$ . On pourra notamment se contenter de valeurs approchées, au lieu de calculer analytiquement (puis programmer)  $n^2$  dérivées différentes :

$$\frac{\partial f_i}{\partial x_j} \approx \frac{f_i(x_1, \dots, x_j + h_j, \dots, x_n) - f_i(x_1, \dots, x_n)}{h_j}$$

De même que dans le cas d'une seule équation, chacune des fonctions  $f_i(\vec{X})$  pourra avoir une (ou plusieurs) zone de convergence. Lorsque l'on part d'un vecteur arbitraire  $\vec{X}^{(0)}$  pour calculer  $\vec{X}^{(1)}$ ,  $\vec{X}^{(2)}$ , ..., le comportement de la suite peut être aléatoire jusqu'à ce que le vecteur arrive dans la zone de convergence de  $\vec{F}(\vec{X})$ , où la convergence sera alors assurée. Plus le nombre de variables est grand et plus la probabilité d'arriver dans la zone de convergence sera réduite, et plus le risque de non-convergence augmentera.

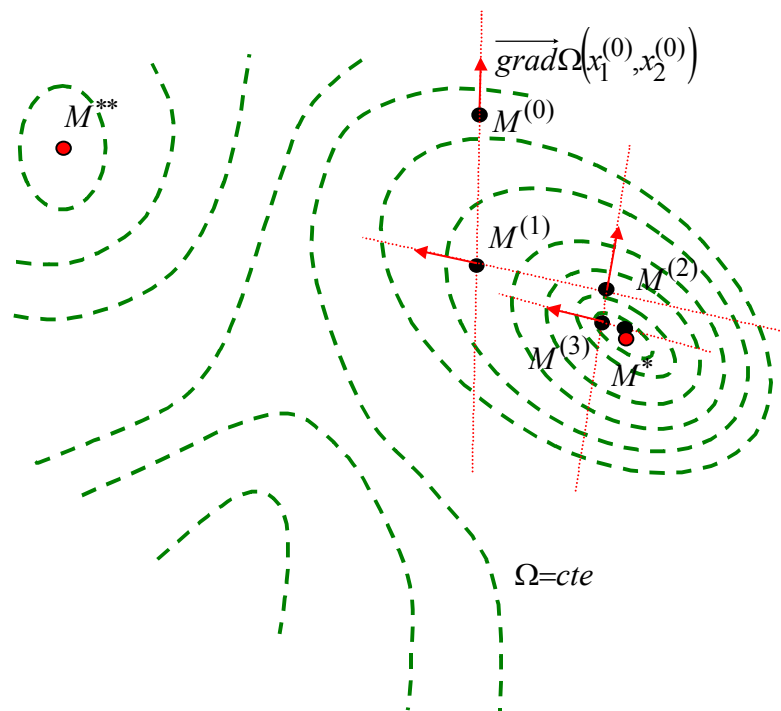
### 3. Méthodes de descente

#### 3.1. Principe des méthodes de descente

La solution du système  $\vec{F}(\vec{X}) = 0$  (linéaire ou non-linéaire) rend minimum la fonction  $\Omega(\vec{X}) = \sum_i f_i^2(\vec{X})$ . Les méthodes de descente sont des méthodes itératives qui consistent à remplacer la recherche de racines par la recherche du minimum de la fonction  $\Omega$ . Si on prend l'exemple d'un système de rang 2 :

$$\begin{cases} f_1(x_1, x_2) = 0 \\ f_2(x_1, x_2) = 0 \end{cases} \Rightarrow \Omega(x_1, x_2) = f_1^2(x_1, x_2) + f_2^2(x_1, x_2)$$

On trace les lignes de niveau de la fonction  $\Omega$  ( $\Omega = cte$ ). Les solutions recherchées ( $M^*$ ,  $M^{**}$ ) correspondent aux creux des "cuvettes". Les méthodes de descentes consistent à rechercher ces creux en se déplaçant, à partir d'un point donné, le long des droites de plus grande pente. La direction de ces droites est donnée par le gradient de la fonction  $\Omega$  ( $\overrightarrow{grad}\Omega(x_1, x_2)$ ) au point considéré. En fait la direction de descente sera donnée par  $-\overrightarrow{grad}\Omega(x_1, x_2)$  puisque le gradient est orienté dans le sens de l'accroissement maximal de la fonction).



Ainsi, le gradient  $\overrightarrow{grad}\Omega(x_1^{(0)}, x_2^{(0)})$  permet de déterminer l'équation paramétrique de la droite de plus grande pente passant par le point arbitraire  $M^{(0)}$ . Les points  $M$  situés sur cette droite auront pour coordonnées :

$$M \begin{cases} x_1 = x_1^{(0)} - t \frac{\partial \Omega}{\partial x_1}(x_1^{(0)}, x_2^{(0)}) \\ x_2 = x_2^{(0)} - t \frac{\partial \Omega}{\partial x_2}(x_1^{(0)}, x_2^{(0)}) \end{cases}$$

On cherche alors la valeur  $t_1$  du paramètre  $t$  tel que  $\Omega(x_1, x_2) = \Omega(t)$  soit minimum. On a ainsi les coordonnées du point  $M^{(1)}$  qui va remplacer le point  $M^{(0)}$  pour la détermination de la droite de plus grande pente, et ainsi de suite jusqu'à obtention de valeurs suffisamment proches de la solution  $M^*$ .

Dans le cas de plusieurs solutions, il faut partir d'un point  $M^{(0)}$  situé dans le creux de la solution à déterminer.

### 3.2. Formulation générale des méthodes de descente

Soit un système linéaire  $A\vec{X} = \vec{B}$  (ou  $\vec{B} - A\vec{X} = 0$ ) à résoudre, où  $A$  est une matrice symétrique (dans le cas contraire on résout le système équivalent  $(A^t A)\vec{X} = A^t \vec{B}$ ). On forme le vecteur résidu  $\vec{R}$  à partir d'un vecteur  $\vec{X}$  quelconque (non nécessairement solution de l'équation) :

$$\vec{R} = \vec{B} - A\vec{X}$$

La solution de l'équation sera le vecteur  $\vec{X}$  qui minimise la forme quadratique :

$$\Omega(\vec{X}) = \frac{1}{2} \vec{R}^t A^{-1} \vec{R} \quad (\text{scalaire})$$

Si on considère une petite variation  $\vec{H}$  autour du vecteur  $\vec{X}$ , et que l'on calcule la valeur de la forme quadratique en  $\vec{X} + \vec{H}$  en supposant une variation linéaire, on aura :

$$\Omega(\vec{X} + \vec{H}) = \Omega(\vec{X}) + \vec{H}^t \overrightarrow{\text{grad}}\Omega(\vec{X}) \quad (\text{similaire à un développement limité à l'ordre 1})$$

Il est aisé de voir que seuls les vecteurs  $\vec{H}$  tels que  $\vec{H}^t \overrightarrow{\text{grad}}\Omega(\vec{X}) < 0$  permettront une diminution de la fonction  $\Omega(\vec{X})$ . De tels vecteurs définissent alors une direction de descente. A partir d'un vecteur  $\vec{X}^{(k)}$  on aura le vecteur suivant  $\vec{X}^{(k+1)}$  grâce à :

$$\vec{X}^{(k+1)} = \vec{X}^{(k)} + \iota^{(k)} \vec{H}(\vec{X}^{(k)}) \quad (\text{pour l'instant } \vec{H} \text{ n'est pas défini})$$

où  $\iota^{(k)}$  est un scalaire déterminé de telle sorte que  $\Omega(\vec{X}^{(k+1)})$  soit minimal. Ceci est réalisé pour :

$$\iota^{(k)} = \frac{[\vec{R}^{(k)}]^t \vec{H}(\vec{X}^{(k)})}{[\vec{H}(\vec{X}^{(k)})]^t A \vec{H}(\vec{X}^{(k)})}$$

On peut alors définir l'algorithme de la méthode générale de descente. Si le système n'est pas linéaire, on aura une matrice  $A(\vec{X}^{(k)})$  variable et qu'il faudra peut être rendre symétrique par multiplication avec sa transposée. Dans le cas le plus général, on aura :

$$\begin{cases} \vec{X}^{(k+1)} = \vec{X}^{(k)} + \frac{[\vec{R}^{(k)}]^t \vec{H}(\vec{X}^{(k)})}{[\vec{H}(\vec{X}^{(k)})]^t A(\vec{X}^{(k)}) \vec{H}(\vec{X}^{(k)})} \vec{H}(\vec{X}^{(k)}) \\ \vec{R}^{(k)} = \vec{B}^{(k)} - A(\vec{X}^{(k)}) \vec{X}^{(k)} \end{cases}$$



En général on considère que la convergence est atteinte quand  $\frac{\|\vec{R}^{(k+1)}\|}{\|\vec{X}^{(k+1)}\|} < \varepsilon$ .

### 3.3. Les différentes directions de descente

#### 3.3.1. Méthode du gradient (ou de la plus grande pente)

On choisit pour la direction de descente  $\vec{H}(\vec{X}^{(k)}) = -\overrightarrow{\text{grad}\Omega}(\vec{X}^{(k)})$ , comme dans l'exemple précédent (système de rang 2). La convergence dans ce cas est assez lente car il n'y a pas d'optimisation.

#### 3.3.2. Méthode du gradient conjugué

On choisit la direction de descente à l'itération  $k$  de telle sorte que  $[\vec{H}(\vec{X}^{(k)})]^T A \vec{H}(\vec{X}^{(k-1)}) = 0$ , c'est-à-dire en fonction de l'itération précédente. On obtient :

$$\begin{cases} \vec{H}(\vec{X}^{(0)}) = \vec{R}(\vec{X}^{(0)}) = \vec{B}^{(0)} - A(\vec{X}^{(0)})\vec{X}^{(0)} \\ \vec{H}(\vec{X}^{(k)}) = \vec{R}(\vec{X}^{(k)}) + \frac{[\vec{R}(\vec{X}^{(k)})]^T \vec{R}(\vec{X}^{(k)})}{[\vec{R}(\vec{X}^{(k-1)})]^T \vec{R}(\vec{X}^{(k-1)})} \vec{H}(\vec{X}^{(k-1)}), \quad k \geq 1 \end{cases}$$