Décomposition LU

Une matrice A carrée inversible d'ordre n a une décomposition LU si on peut l'écrire sous la forme $A = LU$ où L est une matrice triangulaire inférieure (L comme lower) et U une matrice triangulaire supérieure (U comme upper). Si on impose de plus que L (ou U) a des 1 sur la diagonale, alors la décomposition est unique, et on peut facilement obtenir L et U par l'algorithme suivant qui a un temps de calcul en $\Theta(n^3)$ puisqu'il contient trois boucles imbriquées.

```
for i:=1 to n do
for j:=1 to n do
begin
  s:=a[i,j];
  for k:=1 to min(i,j)-1 do s:=s-l[i,k]*u[k,j];
  if i>j then l[i,j]:=s/u[j,j]
         else u[i,j]:=s
end
```

Pour $n = 3$ cet algorithme va calculer dans l'ordre :

$$u_{1,1} = a_{1,1} \qquad u_{1,2} = a_{1,2} \qquad u_{1,3} = a_{1,3}$$
$$l_{2,1} = \frac{a_{2,1}}{u_{1,1}} \qquad u_{2,2} = a_{2,2} - l_{2,1}u_{1,2} \qquad u_{2,3} = a_{2,3} - l_{2,1}u_{1,3}$$
$$l_{3,1} = \frac{a_{3,1}}{u_{1,1}} \qquad l_{3,2} = \frac{a_{3,2} - l_{3,1}u_{1,2}}{u_{2,2}} \qquad u_{3,3} = a_{3,3} - l_{3,1}u_{1,3} - l_{3,2}u_{2,3}$$

qui est bien l'unique solution du système de neuf équations :

$$a_{1,1} = u_{1,1} \qquad a_{1,2} = u_{1,2} \qquad a_{1,3} = u_{1,3}$$
$$a_{2,1} = l_{2,1}u_{1,1} \qquad a_{2,2} = u_{2,2} + l_{2,1}u_{1,2} \qquad a_{2,3} = u_{2,3} + l_{2,1}u_{1,3}$$
$$a_{3,1} = l_{3,1}u_{1,1} \qquad a_{3,2} = l_{3,2}u_{2,2} + l_{3,1}u_{1,2} \qquad a_{3,3} = u_{3,3} + l_{3,1}u_{1,3} + l_{3,2}u_{2,3}$$

En fait on a $a_{i,j} = \sum_{k=1}^{n} l_{i,k}u_{k,j}$. Mais comme L est triangulaire inférieure, $l_{i,k}$ est nul si $k > i$. De même, comme U est triangulaire supérieure, $u_{k,j}$ est nul si $k > j$. Donc $a_{i,j} = \sum_{k=1}^{min(i,j)} l_{i,k}u_{k,j}$.

Si $i > j$ on obtient $a_{i,j} = \sum_{k=1}^{j} l_{i,k}u_{k,j} = l_{i,j}u_{j,j} + \sum_{k=1}^{j-1} l_{i,k}u_{k,j}$ en sortant le dernier terme de la somme. Donc

$$l_{i,j} = \frac{a_{i,j} - \sum_{k=1}^{j-1} l_{i,k}u_{k,j}}{u_{j,j}}$$

De même si $i <= j$ on obtient

$$u_{i,j} = \frac{a_{i,j} - \sum_{k=1}^{i-1} l_{i,k}u_{k,j}}{l_{i,i}}$$

qui se simplifie car $l_{i,i} = 1$:

$$u_{i,j} = a_{i,j} - \sum_{k=1}^{i-1} l_{i,k}u_{k,j}$$

Si A est inversible et possède une décomposition LU, alors puisque le déterminant d'une matrice triangulaire est égal au produit de ses termes diagonaux, on a $\det L = 1$ et $0 \neq \det A = \det U = \prod_{i=1}^{n} u_{i,i}$. Donc tous les $u_{i,i}$ sont non nuls. Donc l'algorithme précédent ne fait pas de division par 0 et trouve bien L et U. Cela prouve donc aussi que la décomposition LU d'une matrice inversible est unique quand elle existe.

On peut remarquer que pour calculer $l_{i,j}$ ou $u_{i,j}$ on utilise $a_{i,j}$ et des $l_{i,k}$ pour $k < j$ ou des $u_{k,j}$ pour $k < i$ c'est-à-dire des coefficients de L et U qui ont des indices plus petits et qui ont déjà été calculés lors d'une itération précédente de la boucle sur i ou sur j.

On peut ranger les coefficients intéressants de L et U dans A. On pourra reconstituer L en rajoutant tous les zéros au dessus de la diagonale et les uns sur la diagonale, et U en rajoutant tous les zéros au dessous de la diagonale. L'algorithme devient :

```
for i:=1 to n do
for j:=1 to n do
begin
  s:=a[i,j];
  for k:=1 to min(i,j)-1 do s:=s-l[i,k]*u[k,j];
  if i>j then s:=s/u[j,j];
  a[i,j]:=s
end
```

résolution d'un système d'équations linéaires

Si A est une matrice inversible, alors $AX = B$ est équivalent à $X = A^{-1}B$. Donc, pour calculer $A^{-1}B$, il n'est pas nécessaire de calculer effectivement la matrice A^{-1} puis de la multiplier par le vecteur B, mais il suffit de résoudre le système d'équations $AX = B$. Autrement dit, on peut considérer que $A^{-1}B$ est seulement une façon de noter la solution du système $AX = B$.

Si $A = LU$, alors $A^{-1} = U^{-1}L^{-1}$ et donc la solution du système $AX = B$ est $X = A^{-1}B = U^{-1}(L^{-1}B)$ qui peut donc s'obtenir en deux étapes, en calculant d'abord $Y = L^{-1}B$, la solution du système $LY = B$, puis $X = U^{-1}Y$, la solution du système $UX = Y$.

Autrement dit, si on a une décomposition LU de A, on peut résoudre un système d'équations ayant A pour matrice, en résolvant d'abord un système triangulaire inférieur, puis un système triangulaire supérieur dont le membre droit est la solution du premier système.

Cela peut se faire en utilisant l'algorithme suivant qui a un temps de calcul en $\Theta(n^2)$ puisqu'il contient deux boucles imbriquées.

```
for i:=1 to n do
begin
  s:=b[i];
  for j:=1 to i-1 do s:=s-l[i,j]*y[j];
  y[i]:=s
end;
for i:=n downto 1 do
begin
```

```
  s:=y[i];
  for j:=i+1 to n do s:=s-u[i,j]*x[j];
  x[i]:=s/u[i,i]
end
```

On peut remarquer que `b[i]` n'est plus utilisé une fois que `y[i]` est calculé. De même, `y[i]` n'est plus utilisé une fois que `x[i]` est calculé. On peut donc ranger ces trois nombres dans la même variable. Au lieu d'utiliser trois tableaux B, Y et X, on en utilisera un seul qui contient initialement B et qui se transforme en Y puis en X. De plus les deux matrices triangulaires L et U peuvent être rangées dans la même matrice carrée A. L'algorithme précédent devient donc :

```
for i:=1 to n do
begin
  s:=b[i];
  for j:=1 to i-1 do s:=s-a[i,j]*b[j];
  b[i]:=s
end;
for i:=n downto 1 do
begin
  s:=b[i];
  for j:=i+1 to n do s:=s-a[i,j]*b[j];
  b[i]:=s/a[i,i]
end
```

Si on a plusieurs systèmes d'équations à résoudre avec la même matrice et des membres droits différents, on peut faire une seule fois la décomposition LU qui prend un temps en n^3 et plusieurs fois la résolution des deux systèmes triangulaires qui prend un temps en n^2.

permutations de lignes

Pour améliorer la précision des calculs, on permute les équations du système de manière à choisir le pivot de plus grande valeur absolue possible dans une colonne. On ne fait donc pas une décomposition $A = LU$ mais $A = PLU$ où P est une matrice de permutation. La solution du système est alors $X = U^{-1}L^{-1}P^{-1}B$. Il suffit donc de remplacer B par $P^{-1}B$ avant de résoudre les deux systèmes triangulaires. Autrement dit, pendant la décomposition LU de A, on note comment on a permuté les lignes de A, pour pouvoir appliquer plus tard la même permutation aux composantes de B.