

### **3 Les Fichiers de Données**

## MATLAB peut lire et écrire des fichiers de format

- ASCII
- Binaire codé en format MATLAB - MAT-files
- Des autres fichiers binaires

### **3.1 Les fichiers ASCII**

### 3.1.1 les commandes save (écriture) et load (lecture)

Les caractéristiques des fichiers ASCII écrits ou lus par ces deux commandes sont les suivants

- chiffres uniquement
- le point est le séparateur partie entière/partie décimale
- les chiffres sont séparés par des blancs ou des tabulations,
- chaque ligne du fichier doit avoir le même nombre de colonnes - ie. des matrices

En utilisant **save** sous MATLAB on ne peut sauvegarder dans des fichiers ASCII que des matrices bidimensionnelles contenant des nombres

**Syntaxe:** **save** nom\_fichier.ext X -ascii -options

où *options* : -double : 15 chiffres après la virgule  
au lieu de 7 par défaut  
: - tabs : les nombres sont séparés  
par des tabulations.

**load** nom\_fichier.ext

» <b>A = rand(3,5);</b>	créer une matrice
» <b>save temp.dat A -ascii</b>	sauvegarde de A dans temp.dat
» <b>type temp.dat</b>	affichage

9.5012929e-001	4.8598247e-001	4.5646767e-001	4.4470336e-001	9.2181297e-001
2.3113851e-001	8.9129897e-001	1.8503643e-002	6.1543235e-001	7.3820725e-001
6.0684258e-001	7.6209683e-001	8.2140716e-001	7.9193704e-001	1.7626614e-001

» <b>clear</b>	effacer la mémoire
» <b>load</b> temp.dat	chargement
» <b>whos</b>	contenu de la mémoire

Name	Size	Bytes	Class
temp	3x5	120	double array

Grand total is 15 elements using 120 bytes

### 3.1.2 La lecture et l'écriture formatées

Afin de pouvoir lire ou écrire des fichiers ASCII n'étant pas codés au format MATLAB il existe des commandes d'écriture et de lecture de fichiers ayant la même syntaxe que celle du langage C

1. La démarche est d'abord l'ouverture du fichier avec la commande **fopen**.
2. Puis la lecture des données avec **fscanf** ( ou fgetl ou fgets ) ou l'écriture avec **fprintf**
3. Finalement la fermeture avec **fclose**.

**Syntaxe :**

fid = **fopen** ('nom\_fichier.ext','permission')

où permission peut être un des suivants

'r'	read
'w'	write (create if necessary)
'a'	append (create if necessary)
'r+'	read and write (do not create)
'w+'	truncate or create for read and write
'a+'	read and append (create if necessary)
'W'	write without automatic flushing
'A'	append without automatic flushing

[A,count] = **fscanf**(fid,FORMAT,size)

A = matrice à remplir  
count = (optionelle) nombre de données lues

fid = généré par **fopen**

FORMAT = comme en C on précise le type de variable lu  
Il faut le caractère % puis une lettre qui définit le type  
d, i, x, e, f, s, c .....

size = (optionelle) il limite le nombre de données lues. Par défaut tout le fichier est lu.

N read at most N elements into a column vector.

inf read at most to the end of the file.

[M,N] read at most M \* N elements filling at least an  
M-by-N matrix, in column order. N can be inf, but not M.  
ie. M = nb de colonnes et N = nb de lignes

ATTENTION!!!! Les lignes ne sont plus prioritaires !

**Exemples :**

```
>> nbcol = 3;nbligne = 200;
```

```
>> fichid = fopen ('toto.txt','r');
```

```
>> Data = fscanf(fichid,'%e',[nbcol,nbligne]);
```

Pour remettre les données en colonnes on prend la transposée de Data.

## 3.2 Les fichiers binaires

### 3.2.1 les fichiers binaires MAT-files

Les MATfiles sont les fichiers de données binaires de MATLAB, le ".mat" étant le suffixe par défaut des fichiers. Il sont créés et lus avec les commandes **save** et **load**.

**Syntaxe :**     **save**   nom\_fichier X Y ... *-options*

X Y ... sont des variables à sauvegarder. Si aucune variable n'est précisée, MATLAB sauvegarde toutes les variables dans l'espace de travail

*options :*        -v4 : sauvegarde dans un format que MATLAB 4 peut lire  
                  -append : ajoute les variables à un fichier existant

**load** nom\_fichier (X Y ....)

X Y ... sont des variables à charger. Si aucune variable n'est précisée, MATLAB charge toutes les variables dans le fichier.

#### Exemples:

» A = **rand**(20,5);               création d'une matrice  
» **save** temp A;                 sauvegarde dans un fichier temp.mat  
» **clear** all                     effacer la mémoire  
» **load** temp                    chargement du fichier temp.mat  
» **whos**                         contenu de la mémoire

Name	Size	Bytes	Class
A	20x5	800	double array

A est reconnu !

» B = 'commentaire';            definition d'une autre variable  
» **save** temp B -append         on ajoute B dans le fichier temp.mat  
» **clear** all  
» **load** temp

» whos

Name	Size	Bytes	Class
A	20x5	800	double array
B	1x11	22	char array

A et B sont reconnus!

Grand total is 111 elements using 822 bytes

» **clear** all  
» **load** temp B                 chargement d'une variable dans le fichier .mat  
» **whos**

Name	Size	Bytes	Class
B	1x11	22	char array

Grand total is 11 elements using 22 bytes

» B  
B =  
commentaire

### 3.2.2. Les autres fichiers binaires

MATLAB peut également lire les fichiers binaires qui ne sont pas codés au format MATLAB

- 1 La démarche est l'ouverture du fichier avec la commande **fopen**.
2. Puis la lecture des données avec **fread** ou l'écriture avec **fwrite**
3. On peut aussi régler la positionnement dans le fichier avec **frewind, fseek, ftell**
4. Finalement la fermeture avec **fclose**.

Exercice:

1. Créer un fichier ASCII qui comprend deux colonnes - l'une est le temps ( $0 < t < 6\pi$ ) pas de  $\pi/10$  et l'autre qui correspond à un vecteur  $s = \sin(t)$ . Puis sauvegarder ces données dans un fichier ASCII qui comprend 2 colonnes ( la 1<sup>ère</sup> est le temps,  $t$ , et la 2<sup>ème</sup> comprend les données,  $s$ )
2. Lire un fichier ASCII. Sur la disquette il se trouve un fichier qui s'appelle ex\_data.txt. La première ligne comprend une commentaire (3 mots) qui est suivi par deux colonnes des données. Les données sont des valeurs réels (floatantes) et il y en a 10 dans chaque colonne. Qu'est ce que c'est la commentaire? et évaluer la moyenne de chaque colonne.