

Matlab : résumé du cours

Document autorisé à l'examen s'il n'est pas annoté.

Variables :

Déclaration à la volée, type adaptatif

Information sur les variables :

isxxx (ex `isreal`, `iscomplex`...)

`[ l c ] = size( M );` `L = length( M ) = max( size(M) )`

Plages :

Priorité au pas	Priorité au nombre de points
<code>Xmin : h : Xmax</code> Option : si omis <code>h=1</code>	<code>linspace(Xmin, Xmax, n)</code> Option : si omis <code>n = 100</code>

Matrices :

Matrice : Concaténation	Matrice : Sélection
<code>[]</code> : met bout à bout les éléments, <code>;</code> passe à la ligne	<code>M(plage_ligne , plage_colonne)</code> : débute à 1

Sélection : plage complète `:` / choix du dernier `end`

Matrices spéciales : `m = zeros( n_lignes, n_colonnes )` / idem `ones` et `eye` (matrice identité)

Opérateurs :

Combinent les variables : `+`, `-`, `*`, `/` etc. Ils s'adaptent au type de variables

Possibilité de forcer un comportement 'terme à terme' en préfixant avec un `.'` l'opérateur.

Opérateurs logiques :

Comme en C mais non-doublés. Opérateurs termes à termes. Retournent vrai (`=1`) / faux (`=0`).

Ex : `[ 1 2 3 ] <= 2` retourne `1 1 0`

Contrôle de flux :

<code>if</code> (condition) séquence d'instructions A <code>else</code> séquence d'instructions B <code>end</code>	<code>switch</code> (expression énumérable) <code>case</code> { liste de valeurs }, séquence d'instructions... <code>case</code> { liste de valeurs }, séquence d'instructions... ... <code>otherwise</code> , instructions... <code>end</code>
<code>while</code> (condition) séquence d'instructions... <code>end</code>	<code>for</code> index = debut : pas : fin séquence d'instructions... <code>end</code>

Saisie/affichage :

Saisie	Affichage
Variable = <code>input('message','s')</code> % 's' pour des char	* ne pas mettre de <code>;</code> en fin de ligne * OU utiliser <code>disp : disp(['a = ', num2str(a)])</code>

Algorithmes de Base :

Somme :	Produit
<pre> S = 0 % S est l'accumulateur Pour tous les Xk S = S + Xk fin </pre>	<pre> P = 1 % P est l'accumulateur Pour tous les Xk P = P * Xk fin </pre>

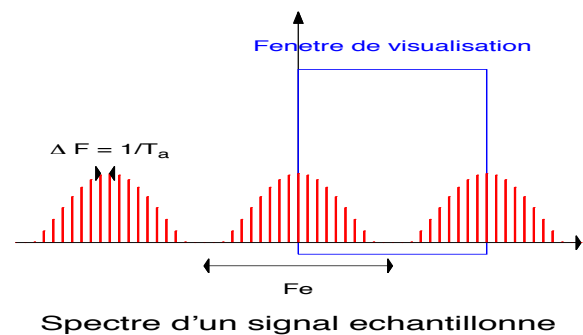
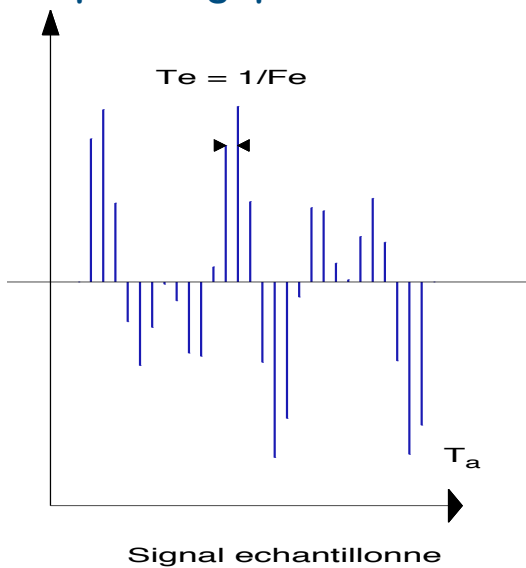
En général ces grandeurs sont calculées avec des boucles for.

Commandes Graphiques : tapez help + commande pour + d'info

Tracés :	Décors :
plot ; semilogx ; semilogy ; loglog ; stem	title ; legend ; xlabel ; ylabel ; grid on/off

Création de figures / graphes multiples	Grille, superposition, axes
Automatique a l'appel d'une commande de tracé ou figure(numéro) ou subplot(nligne, ncol, Num_fig)	grid on/off ; hold on/off ; axis tight ; axis([xmin, xmax, ymin, ymax])

Graphe magique :



Conservation de l'énergie	Equivalence de la dérivation
$\frac{1}{N} \sum_{t=0}^{N-1} x[t]^2 = \sum_{f=0}^{N-1} X[f] ^2$	$p \rightarrow \frac{1 - z^{-1}}{T_e}$

Vous pouvez (devez ?) utiliser l'aide en ligne et poser des questions au besoin même durant l'épreuve