

Charlotte Elizabeth Ameli

Sci Lab * les essentials pour moi *

INTRO

editor / scinotes - text editor / program editor

editvar - variable / table / array editor

CSVRead / CSVWrite - fonctions pour import/ex de fichiers C.S.V.

(f) CSVTextScan, read_csv, readxls, write_csv, xls_open, xls_read

matfile_open, matfile_close, matfile_listvar, - ouvrir un fichier Matlab

(f) matfile_Varreadnext, matfile_Varwrite, ~~matfile~~, saveatfile, loadmatfile

vdriv =

Export plot à fichier graphique driver ('GIF') ou driver('JPEG')

xinit ('TMPDIR' + '/myfile.gif'); plot3d(); xend(); driver(vdriv);

Xcos - affichage le schéma de plà

Xls_Open - opens excell spreadsheet fichier pour de lirez

(f) XLS_Read, readXls (reads a whole spreadsheet at a time)

audioread, wavread - reads an audio file in au or wav formats

SaveWave - save data into a sound file

analyze - frequency plot of a sound signal mapsound - plots a sound map

Mfile2sci - fonction de conversion de M-file Matlab vers Scilab.

(m-file-path [, result-path [, Recmode [, only-double [, verbose-mode [, prettyprint-output]]]]])

test-run - lance les tests unitaires et de non régression présents dans un module ou dans un répertoire. status = test-run(), test-run(module)

test-run(module, test-name), test-run(module, test-name, options, exportToFile)

GRAPHIQUE

* Graphique Export * Tous sur le g  tre graphique.

Xs2 bmp - exporter en BMP
Xs2 gif - exporter en GIF
Xs2 eps - exporter en EPS
Xs2 svg - exporter en SVG
Xs2 emf - exporter en EMF

Xs2 jpg - exporter en JPG
Xs2 ppm - exporter en PPM
Xs2 pdf - exporter en PDF
Xs2 ps - exporter en PS
stoprint - envoyer texte ou fig   l'imprimante

* Test Conditions * + other misc functions

&, && - Binary AND between integers. Logical AND over/between booleans or numbers.

Ind2sub - linear index to matrix subscript values

iscolumn - check if a variable is a column

isempty - check if a variable is an empty matrix or an empty list

isequalbitwise - comparison of objects (NaN are equal)

ismatrix - checks if an object is a non-empty 1D or 2D matrix or array.

isrow - check if a variable is a row

isscalar - check if a variable is a scalar

is square - check if a variable is a square matrix

is vector - check if a variable is a vector

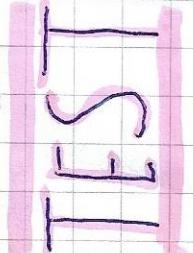
pmodulo - positive euclidian remainder modulo m

ndims - number of dimensions of an array

|, || - Binary OR between integers. Logical OR over/between booleans and numbers

size - Size of objects

sub2ind - Matrix Subscript values to Linear Index



* SCI-Lab HDF Support *

Hierarchical Data Format
- NC4 routines

h5attr - create attribute (obj, location, name, data[, tagstype]) = gchiver

h5close - Close HDF5 object

h5cp - Copy object

h5dataset - Create dataset + write data

h5dump - dump H5 object data to output stream

h5exists - test if named object exists

h5flush - flush files buffers

h5get - get named object

h5group - Create group

h5isAttr - check if HDF5 object is attribute

h5isCompound - check if HDF5 object is compound object

h5isFile - check if HDF5 object is a file

(6) h5isGroup - check if HDF5 object is a Group

h5isRef - check if HDF5 object is a Reference

h5isSet - check if HDF5 object is a dataset

h5isSpace - check if HDF5 object is dataspace

h5isType - check if HDF5 object is a Type

h5isVlen - check if HDF5 object is Variable Length Array

~~h5label~~

h5label - Label a dataset

h5ln - Link an object to another one

h5ls - List the content of an HDF5 object

h5mount - Mount a file on a group

h5mv - Move an object

H5 Objects - Describe the properties of different H5 objects

h5open - Open an HDF5 file

h5read - Read the data of HDF5 dataset

h5readattr - Read the data of HDF5 attribute

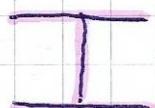
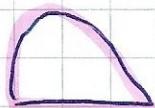
h5rm - Remove elements from an HDF5 file

h5umount - Unmount a previously mounted file

h5write - Create a dataset (if doesn't exist) and write data

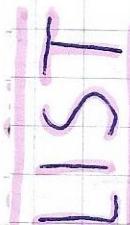
h5writeattr - Write an attribute in a group or a dataset

h5isArray - check if HDF5 object is an array



* les listes *

list (a_1, \dots, a_n)



Crée une liste avec les éléments a_i qui sont des objets Scilab arbitraires (matrice, liste, ...). Le numéro de type des objets liste (renvoyé par la fonction type) est 15. list() crée liste vide à 0 élément.

Opérations sur les listes

■ extraction

$[x, y, z, \dots] = L(v)$ où v est un vecteur indices $[x, y, z] = L(:)$

■ insertion en position i

$L(i) = a$. L'opération $L(i) = a$ avec $i > 1 + \text{size}(L)$ ne génère pas d'erreur les entrées en position $1 + \text{size}(L), \dots, i - 1$ étant alors indéfinies. Par contre l'extraction d'un élément indexé d'un

■ ajouter un élément en queue de liste

$L(\$+1) = e$

■ ajouter un élément en tête de liste

$L(0) = e$ - après ça e est en position 1, tous autres

■ Suppression

$L(i) = null()$ - Supprime l'élément en position i .

■ Concaténation de deux listes

$L3 = \text{lstcat}(L1, L2)$

■ Nombre d'éléments d'une liste

Nombre = size(L) ou encore nombre = length(L)

■ Itérer avec une liste

Piloter une boucle for $e=L, \dots, \text{end}$ boucle length(L), L(i)

Exemple

Cell Arrays + * Matrices* [Crée]

=`makecell`(`dims, a1, a2 ... an`) - creates a cell array of dimensions given by `dims`, the `a1` are stored along the last dimension first.
`a = makecell([2,3], 1, 2, 3, 'x', 'y', 'z')`
↳ `[1,x]` etc.

=`cell()`, =`cell(m1)`, =`cell(m1, m2)`, =`cell(m1, m2, ... mn)`, =`cell(x)`

creates a series of empty matrices, `x`=a vector containing the dimensions of the cell to create. `M1, M2, Mn...` dimensions of the cell to create.

`cell()` - returns a cell array of empty matrices | `cell(m1)` - returns a (`m1, m2`)

`cell(m1, m2)` - returns a (`m1, m2`) | `cell(m1, m2, ... mn)` - returns a (`m1, m2 ... mn`)

`cell(x)` - returns a cell array empty matrices, 1st dimension is `x(1)`, 2nd is `x(2)` etc..

MATRICES

[] - désigne une matrice vide. Elle est définie de façon unique - O l'ingle Oct.

y=`diag(vm)` ou y=`diag(vm, k)` - extraction de diagonale ou construction d'une matrice avec un vecteur en diagonal, `VM`: vector or matrix, `K` entier (default), y=matrice ou vecteur

y=`ones()`, =`ones(m1, m2, ...)`, =`ones(x)` - matrice remplie de 1,

`ones(m1, m2)` - renvoie une matrice de taille (`m1, m2`) remplie de 2

y=`rand(m1, m2)` - crée de une matrice de taille (`m1, m2`) aléatoire

y=`zeros(m1, m2, ...)` - crée un matrice nulle, renvoie une matrice tous termes nuls.

X=`eye(m, n)`, =`eye(A)`, =`eye()` - matrice identité

`tril(x, [,k])` - lower triangular matrix, X=matrix (real, complex, poly)

K=integer (default value 0). Lower triangle part of a matrix.

`tril(x, k)` is made by entries below the Kth diagonal. K>0 (upper diag.) and K<0 (diagonals below the main diagonal)

~~triu~~ `triu` - Upper triangle matrix - See tril above.

X=`cell2mat(c)` - converts a cell array into a matrix, X=matrix, C=cell array or matrices. can be scalars

y=`Sqrtn(x)` - matrix square root (see analysis) - (matrices)

[e, m]=`det(X)` - determinant (X=matrix, m=Base 10 determinant, e=exp)

y=`norm(x, normType)` - norms of a vector or a matrix

`normType` = 'log', or 'geo', 1, 2, %inf, -%inf (Default=2)

Arrays Etc

Vector Lignes de Nombres etc..

linspace - suite de nombres équidistants entre 2 bornes atteintes.

row = linspace(x1, x2) - x_1, x_2 : nombres réels ou complexes, les deux doivent être de même type et de mêmes dimensions.

row = linspace(x1, x2, n) - n nombre entier.

Matrix = linspace(Col1, Col2) - il précise le nombre de valeurs.

Matrix = linspace(col1, col2, n) ou de colonnes à générer entre

- **Col** - Vecteurs colonne de deux extrémités, 100 par défaut

nombres réels ou complexes de même taille. Eg: aussi

- **row** - Vecteur ligne de n nombres.

- **linspace([1:4], [5:8])** - transposition des colonnes

- **Matrix** - Matrice à n colonnes de nombres.

LINSPACE

logspace - Vecteur de valeurs logarithmiquement équidistants.

p1 = logspace(d1, d2, [n]) - d_1, d_2 : scalaire réel ou complexe (une signification particulière est donnée à la valeur %pi)

d_1, d_2
as $10^n(d_1)$
 $d = 3 \Rightarrow 10^3$
 $\Rightarrow 1000$

[n] - entier (nombre de valeurs, 50 par défaut)

p1 - Vecteur ligne réel ou complexe.

LOGSPACE

Arithmétique

factor - factor function

factorial - the factorial function

gcd - Greatest (positive) common divisor

lcm - Plus petit multiple commun (PPCM) de deux entiers

perms - retourne toutes les permutations des composantes d'un vecteur

primes - primes function

rat - approximation rationnelle d'un flottant

ARITHMÉTIQUE

* Polynômes *

bezout - équation de Bezout pour les polynômes

cheby - Chebychev polynomial

cmndred - dénominateur commun • Cofgg - co-facteurs d'une matrice carrée

Coeff - coefficients d'une matrice de polynômes

Colcompr - compression des colonnes d'une matrice de polynômes

degree - degrés d'une matrice de polynômes

denom - dénominateur

derivat - dérivée d'une matrice de polynômes

determ - déterminant d'une matrice de polynômes

detr - déterminant d'un polynôme

diophant - équation diophantienne de Bezout

factors - factorisation réelle de polynômes

hermit - forme d'Hermite

horner - évaluation d'un polynôme ou d'une fraction rationnelle

hrmt - PGDC de plusieurs polynômes

inv-coeff - construction d'une matrice polynomiale à partir de coefficients

invr - inversion d'une matrice rationnelle (ou polynomiale)

lcmdlag - factorisation diagonale par le plus petit commun multiple

ldiv - division longue d'une matrice de polynômes

numer - numérateurs d'une matrice rationnelle

pdiv - division de polynômes

pol2str - conversion polynôme => texte

polfact - facteurs minimaux

poly - définition d'un polynôme

rational - Object scilab, fraction rationnelle

residu - résidu

roots - racines d'un polynôme

rowcompr - compression de lignes d'une matrice de polynômes

sfact - factorisation spectrale en temps direct

Simp - simplification rationnelle

Simp_mode - Mode de simplification des fractions rationnelles

SyIm - Matrice de Sylvester

Vara - Variable symbolique d'un polynôme ou d'une fraction rationnelle.

SE
M
N
O
P
L
I
O
N
P
Y
N
O
R

* 2D Plot *

Champ - 2d vecteur champ plot

champ1 - 2d Vecteur champ plot avec flèches couleur.

Champ properties - desc de 2d vecteur champ properties

comet - 2d comet animated plot

Contour 2d - level curves of a surface on a 2d plot

Contour2d! - calcule les courbes de niveau d'une surface

Contour2dm - compute level curves of a surface defined with a mesh

Contourf - courbes de niveaux 2d remplies d'une surface 3d

errbar - ajoute des barres d'erreur sur un dessin 2d

fchamp - direction field of a 2d first order ODE

fec - dessin en pseudocouleurs d'une surface définie sur un maillage polygones convexes constitué de

fec properties - description of the fec entities properties

fgreyplot - 2d représentation en ps-couleurs d'une surface définie par une fonction

fplot 2d - 2d plot of a curve defined by a function

grayplot - 2d plot of a surface using colors

grayplot properties - description of the grayplot entities properties

graypolar plot - Polar 2d plot of a surface using colors

histplot - dessine un histogramme

Line Spec - to quickly customize the lines appearance in a plot

Matplot - 2d plot of a matrix using colors

Matplot properties - description of the matplot entities properties

paramfplot2d - animation dans le plan d'une courbe définie par un fonc

plot - 2d plot

plot2d - 2d plot

plot 2d 2 - 2d plot (step function)

plot 2d 3 - 2d plot (vertical bars)

plot 2d 4 - 2d plot (arrows style)

polar plot - Plot polar coordinates

Scatter - 2d scatter plot

Sgreyplot smooth 2d plot of a surface defined by a function using colors

Sgrayplot - Smooth 2d plot of a surface using colors.

2D PLOT

* Matrices - Analysis * MATRIX

Cond - $c = \text{cond}(x)$, $c = \text{cond}(x, p)$ • Conditionnement

X matrice réelle ou complexe. Si $c = \text{cond}(x, p)$, X doit être une matrice carrée réelle ou complexe.

P Scalaire ou chaîne de caractères (valeur par défaut p=2)

C Scalaire réel

■ Det - $\det(X)$, $[e, m] = \det(X)$ • déterminant

m nombre réel ou complexe, mantisse du déterminant en base 10.
e entier, exposant du déterminant en base 10

$\det(X) (m * 10^e)$ est le déterminant de la matrice carré X
pour les matrices polynomiales $\det(x)$ est équivalent à $\text{determ}(x)$
pour les matrices rationnelles $\det(x)$ est équivalent à $\text{detr}(x)$.

Q = Orth(A) - • Calcul d'une base orthogonale
A + Q Sont matrices réelle ou complexe

rank - $[i] = \text{rank}(x)$, $[i] = \text{rank}(x, tol)$ • rank
• Calcule le rang numérique, nombre valeurs singulières
tol - nombre réel positive.

cond - $\text{cond}(x)$ • Estimation de l'inverse conditionnement

rref - $R = \text{rref}(A)$ • Compte Matrix row echelon transformations
R m x n matrix, row echelon en form 'A'.

trace - $\text{trace}(x)$ • trace d'une matrice

- Identique à $\text{sum}(\text{diag}(x))$.

inv - inverse d'une matrice, pour matrices polynômes rationnelles $\text{inv}(x)$ est équivalent à $\text{invr}(x)$

pinv - Pseudo inverse - matrice réelle ou complexe, nombre réel, $\text{pinv}(x)$
• Pour les matrices rectangles.

* Differential Calculus - Integration *

D
I
N
T
E
G
R
A
T
I
O
N

- **bvodeS** - Simplified call to bvode
- **dae** - Differential algebraic equations solver
- **daeoptions** - Set options for dae solver
- **daskr** - DAE solver with zero crossing
- **daskrt** - DAE solver with zero crossing
- **dassl** - Differential algebraic equation
- **diff** - Difference and discrete derivative
- **impl** - differential algebraic equation
- **int2d** - definite 2D integral by quadrature and cubature method
- **int3d** - definite 3D integral by quadrature and cubature method
- **intc** - Cauchy integral
- **integrate** - Integration of an expression by quadrature
 - **intg** - definite integral
 - **intl** - Cauchy integral
- **intspin** - Integration of experimental data by Spline interpolation
- **inttrap** - Integration of experimental data by trapezoidal interpolation
- **numderivative** - approximate derivatives of a function (Sacobian or Hessian)
- **ode** - Ordinary differential equation solver
- **ode_discrete** - ordinary differential equation solver, discrete time step
- **Ode_optional_output** - ode solvers optional outputs description
- **Ode_root** - ordinary differential equation solver with roots finding
- **ODEDC** - discrete / continuous ode solver
- **Odeoptions** - Set options for ode solvers

* Non linear Least Squares *

N
L
S
Q
U
A
R
T

- **datafit** - Parameter identification based on measured data
- **leastsq** - Solves non-linear least squares problems
- **lsqrssolve** - Minimize the sum of the squares of non linear functions, Levenberg-Marquardt algorithm.

* Further Matrix Functions * - amending data

- ' (Quote) - opérateur de transposition, délimiteur de chaîne caractères
- cat - concatenate several arrays .! - transpose non-Conjuguée
- flipdim - reverses the order of (blocks of) rows, columns, pages of a matrix / hypermatrix
- matrix - reshapes an array with the same number and order of components
- permute - permutes the dimensions of an array
- pertrans - ~~pertrans~~ Transposition with reference to 2nd diagonal
- resize_matrix - trim or/and extend (and cast) matrix / hypermatrix
- Squeeze - removes singleton dimensions of a hypermatrix

Operations:

abs - absolute value, magnitude and - logical AND between elements of a

boolean or numerical array cross - vector cross product cumprod -

cumulative sum of array elements kron - Kronecker product (,*)

max - maximum min - minimum norm - norms of a vector

or matrix or - logical OR over the elements of a boolean

or numerical array prod - product of array elements

sign - signum function signm - matrix signum function

sum - sum of array elements tril - lower triangular part

of a matrix triu - upper triangle unwrap - unwrap a

$Y(x)$ profile or a $Z(x,y)$ surface. Unfold a $Y(x)$ profile,

Search and sort | set operations

intersect - returns the vector of common values of two vectors

setdiff - returns components of a vector which do not belong to another one

union - extract union components of a vector

unique - extracts (and sorts) distinct components of a matrix,

dsearch - distribute, locate and count elements of in categories matrix/hypermatrix

find - find indices of boolean vector or matrix true elements

qsort - sorting by quick sort algorithm

members - count (and locate) in an array each element of another array row and column

vectorfind - finds in a matrix rows or columns matching a vector.

* Matrix Trigonometry Functions *

acoshm - matrix hyperbolic inverse cosine.

acosm - matrix wise cosine inverse

asinhm - matrix hyperbolic inverse sine

asinm - matrix wise sine inverse

atanhm - matrix hyperbolic tangent inverse

atanm - square matrix tangent inverse

coshm - matrix hyperbolic cosine

cosm - matrix cosine function

cot hm - matrix hyperbolic cotangent

sinhm - matrix hyperbolic sine

sinm - matrix sine function

tanhm - matrix hyperbolic tangent

tanm - matrix tangent

* Log - Exp - Power *

exp - element-wise exponential

expm - square matrix exponential

log - natural logarithm

log10 - base 10 logarithm

log1p - computes with ~~out~~ accuracy natural logarithm of its arg, by ^{added} one

log2 - base 2 logarithm

logm - square matrix logarithm

nthroot - Real n th root of real numbers

power - (\cdot, \cdot) power operation

Polar - polar form

Sq rt - square root

Sqr tm - matrix square root

LOG-EXP:

* Standard Trigonometry Functions *

acos - element-wise cosine inverse (radians)

acosd - element-wise cosine, result in degree

acot - Computes the element-wise inverse cotangent of the argument

acotd - computes the element-wise inverse cotangent - result in degree

acoth - element wise hyperbolic cotangent inverse

acs - Computes the element-wise inverse cosecant of the argument

acs - computes the element-wise inverse cosecant - result in degree

acsch - computes element-wise inverse hyperbolic cosecant of arg.

asec - computes the element-wise inverse secant of argument

asecd - computes element-wise inverse secant - result in degree

asech - computes the element-wise inverse hyperbolic secant of arg.

asin - Sine inverse (in radians)

asind - Sine inverse - result in degree

asinh - hyperbolic sine inverse

~~atan~~ - 2 quadrant and 4 quadrant inverse tangent

atand - 2 quadrant and 4 quadrant element-wise inv. tangent - result in degree

atanh - hyperbolic tangent inverse

cos - cosine function

cosd - element-wise cosine func. argument in degree

cosh - hyperbolic cosine

cotd - element-wise cotangent func. argument in degree

cotg - cotangent

coth - hyperbolic cotangent

CSC - computes the elementwise Cosecant

CScd - computes the element-wise cosecant of argument in degrees

CSCh - computes the element-wise hyperbolic cosecant of the argument

CSgn - returns the sign of the vector of real or complex values

Sec - Computes the element-wise secant of the argument

Secl - Computes the element-wise Secant of argument given in degree

TRIG

* Floating Point Operations *

ceil - round up

clean - cleans matrices (round to zero small entries)

double - converts int type integers or booleans into decimal encoding

fix - round towards zero

floor - round down

format - set the default output format of decimal numbers

frexp - dissect floating-point numbers into base 2 exponent and mantissa

ieee - Sets or gets the floating point exception mode

int - round towards zero

isinf - tests for infinite elements

isnan - check for "Not a number" entries

nearfloat - get previous or next floating-point number

nextpow2 - next higher power of 2

number_properties - determine floating-point parameters

round - round to nearest integer

Complex and Discrete

COMPLEX NUMBERS

complex - Create a complex number

conj - Complex conjugate

imag - Imaginary part of complex numbers, polynomials or rationals

imunit - multiplication by i the imaginary unitary

isreal - check if variable is stored as a complex matrix

real - real part of a complex number, polynomial or rational

factor - factor function

factorial - the factorial function

gcd - greatest (positive) common divisor

lcm - least common (positive) ~~multiple of~~ multiple of integers or polynomials

perms - all permutations of vector components

primes - primes function

rat - floating point rational approximation

'Reshaping' * Matrix * • Redimensionne un tableau, à nombre et ordre inchangés des éléments.

$y = \text{matrix}(v, m, n)$

$y = \text{matrix}(v, m_1, m_2, m_3, \dots)$

$y = \text{matrix}(v, [\text{sizes}])$ • Par exemple pour une matrice de 4×2 origin nous obtient $6 \times 2, 12 \times 1$ matrices.

✓ → tout type de conteneur matriciel (matrice homogène de tous types de données; tableau de cellules; tableau de structures), à nombre quelconque de dimensions (vecteur, matrice, hyper tableau N-dimensionnel) de toutes tailles.

n, m, m_1, m_2, \dots → entiers positifs: nouvelles tailles attendues après redimensionnement. Une des tailles peut être fixée à -1.

sizes → Vecteur d'entiers positifs; nouvelles tailles à fixer. y tableau redimensionné, avec $y(i) = v(i)$ et ~~$\text{size}(y, " * ") = \text{size}(v, " * ")$~~

~~X~~ Pour un tableau ou un hypertableau v à $n \times m$ éléments principaux. $y = \text{matrix}(v, n, m)$ ou $y = \text{matrix}(v, [n, m])$ redimensionne v en une matrice ou un tableau 2D de taille $n \times m$, par indice linéarisé croissant des éléments de v .

De même, pour un tableau ou un hypertableau v à $m_1 \times m_2 \times m_3 \dots$ éléments principaux, $y = \text{matrix}(v, m_1, m_2, \dots)$ ou $y = \text{matrix}(v, [m_1, m_2, m_3, \dots])$ redimensionne v en une hypermatrice ou un hyper tableau à m_1 lignes, m_2 colonnes, $m_3 \dots$ pages, etc.

Dans tous les cas, l'ordre linéaire des éléments dans y est le même que celui dans v : pour tout $1 \leq i \leq \text{size}(v, " * ")$, $y(i) = v(i)$.

* Permute *

• permute les dimensions d'un tableau à N dimensions

$y = \text{permute}(x, \text{dims})$

X - Un tableau ayant un nombre quelconque N de dimensions (0, 1, 2, ou $N > 2$), de type de données quelconque : booléens, entiers encodés, nombres décimaux réels ou complexes, polynômes, fractions rationnelles, tableau de structures, tableau de cells.

dims - toute permutation des nombres entiers $1:N$ où $N > n_{\min}$ est le nombre de dimensions du tableau souhaité, au minimum $n_{\min} = \text{ndims}(x)$.

La dimension numéro ~~#~~ #i dans y était la dimension numéro ~~#~~ dims(i) dans x.

y - Un tableau ayant le même contenu que x, avec des tailles identiques à celles de x mais permutes.

• Permute (...) Permute les dimensions du tableau selon les correspondances données dans le vecteur dims

T • Dans le tableau résultant y, toute dimension d'ordre majeur réduite à une taille 1 (singleton) est automatiquement ignorée (réduction).

R • permute() est une généralisation à un nombre de dimensions N quelconque des opérateurs de transposition, 'ou'. Elle est principalement utile pour manipuler les dimensions des hypermatrices. Pour simplement transposer un matrice (permute ses lignes en colonnes et vice-versa), on utilisera de préf., 'ou'

Sparse Matrix Conversion

adj2sp - converts adjacency form into sparse matrix
full - sparse to full matrix conversion
mtlb_s - sparse - convert sparse matrix
Sp2adj - Converts sparse matrix into adjacency form
Sparse - sparse matrix definition
Spcompact - Converts a compressed adjacency representation
Spget - retrieves entries of sparse matrix

* Décompositions Creuses *

ludel - libération de la mémoire allouée à des facteurs LU creux
lufact - factorisation LU d'une matrice creuse
luget - extraction de facteurs LU creux
lusolve - solveur de système linéaire creux
spchol - Factorisation de Cholesky creuse

* Manipulation des Matrices Creuses *

issparse - determine whether input is sparse
nnz - nombre de termes non nuls dans une matrice
speye - matrice identité creuse
sprones - matrice creuse dont les termes valent 1
sprand - matrice creuse aléatoire
spzeros - matrice creuse nulle

chfact - Factorisation de Cholesky creuse
chsolve - solveur de Cholesky creux
ordmnd - Compute multiple minimum degree ordering

* Structure de Contrôle *

CONTÔL

abort - interruption d'une évaluation

break - interruption d'une boucle

case - mot-clé utilisé dans une structure "Select"

continue - Keyword to pass control to next iteration of a loop

do - Mot clé utilisé pour les boucles

else - Mot clé utilisé dans une structure if-then-else et select-case

elseif - Mot clé utilisé dans une structure if-then-else

end - mot-clé end

for - Mot clé pour les structures de boucles

halt - stop execution

if - Mot clé utilisé pour une exécution conditionnelle

pause - mode de pause, attente clavier

resume - Sortie d'une fonction ou reprise d'exécution, au

return - Sortie d'une fonction ou reprise d'exécution, avec récupération

Select - Mot clé utilisé dans une structure select-case-else

then - Mot clé dans une structure 'if' et 'select'

catch - Mot clé de début du bloc catch dans une corde "try-catch"

while - Mot clé utilisé dans une structure - while..end.

* Constantes *

%e - euler number

%eps - epsilon (floating point rel. acc.)

%f %F - boolean var for false

%i - imaginary unit

%inf - infinity

%nan - not a number

%pi - ratio circle's circumference to diameter

%S - a variable used to define polynomials

%t,%T - boolean value for True

%Z - variable used to

define polynomials

%home - User directory

SCI, WSCI - root directory for sci

SCIHOME - Path to preferences or history of session

TMPDIR - chemin du répertoire temporaire,

SIMPLEX

* Simplex - Optimisation + Simulation *

optim simplex_center - computes center

optim simplex_check - checks the consistency of internal data

optim simplex_compsomefv - computes the values of a function at vertices

optim simplex_computefv - computes value of function at vertices points.

optim simplex_deltafv - computes the difference of function values

optim simplex_deltafvmax - difference of function value between highest and lowest points.

optim simplex_destroy - destroys the simplex object

optim simplex_dirmat - computes the directions

optim simplex_fvmean - computes the mean

optim simplex_fvstddev - computes the standard deviation

optim simplex_fvvariance - computes the variance

optim simplex_getall - returns all data contained in simplex object

optim simplex_getallfv - returns all function values in the simplex object

optim simplex_getallx - returns all coordinates

optim simplex_getfv - returns the function value at given index

optim simplex_getn - returns the dimension of the space

optim simplex_getnbve - gets the number of vertices of the simplex

optim simplex_getve - gets the vertex at the given index

optim simplex_getx - gets the coordinates of the vertex at given index

optim simplex_gradientfv - returns the simplex gradient of the function

optim simplex_new - creates a new simplex object

optim simplex_reflect - returns the reflected simplex object

OptimSimplex_Setall - sets all in the simplex object

optim simplex_Setallfv - sets all the function values

optim simplex_Setallx - sets all the coordinates

optim simplex_Setfv - set the function value at given index

optim simplex_Setn - sets the dimension of the space

optim simplex_Setnbve - sets the number of vertices of the simplex

optim simplex_Setve - sets the data at a given index of the simplex object

optim simplex_Setx - sets the coordinates at a given index of simplex object

optim simplex_shrink - shrinks the simplex

optim simplex_size - computes the size

optim simplex_sort - sorts the simplex

optim simplex_xbar - returns the center of n vertices

* Optimization Base *

`optimbase_cget` - returns the value for the given key

`optimbase_checkbounds` - checks the bounds

`optimbase_checkcostfun` - checks the cost of function

`optimbase_checkx0` - checks the initial guess

`optimbase_Configure` - configures the current object

`optimbase_destroy` - resets `historyopt` and `historyxopt` field

`optimbase_function` - calls cost of function

`optimbase_get` - returns the value for a given key

`optimbase_hasbounds` - checks if the bounds are specified

`optimbase_hasconstraints` - checks if the constraints are specified

`optimbase_hasnlcons` - checks if non-linear constraints are specified

`optimbase_histget` - returns the history value

`optimbase_histset` - set the history value at iteration for given key

`optimbase_incr iter` - increments the number of iterations

`optimbase_isfeasible` - checks if the point satisfies the constraint

`optimbase_isinbounds` - checks if point satisfies bounds constraints

`optimbase_isnonlincons` - checks if point satisfies non-linear constraints

`optimbase_log` - prints the given message

`optimbase_new` - creates a new optimization object

`optimbase_outputcmd` - call back user's output command

`optimbase_outstruct` - returns a data structure of type `T_OPTDATA`

`optimbase_proj2bnds` - returns a projection point

`optimbase_set` - set the value for the given key

`optimbase_Stoplog` - prints the given stopping message

`optimbase_terminate` - checks if the algorithm is terminated

OPTIM BASE

* Non-Linear Least Squares

* Neldermead *

↓

leastsq - solve non-linear leastsq probs.
(Levenberg - Marquardt) nonlin
lsqrsolve - Min Sum of Squares functions

fminsearch - computes unconstrained min. of function with Nelder-Mead algorithm

neldermead - provides direct search optimization algorithms

nmplot - provides direct search optimization algorithms

optimget - queries an optimization data structure

optimplotfuncount - plot the number of function evaluations of optimization algorithm

optimplotfval - plot the function value of an optimization algorithm

optimplotx - plot the value of parameters of an optimization algorithm

optimset - Configures and returns an optimization data structure

* Semidefinite Programming *

aplat - flattens a list

list2vec - Concatenates list entries in a matrix

lmisolver - Solve linear matrix inequations

lmitool - graphical tool for solving linear matrix inequations

pack - compress a list of block-diagonal symmetric matrices

recons - inverse function for aplat

semidef - Solve semidefinite problems

unpack - uncompress a list of block-diagonal symmetric matrices

vec2list - inverse function of list2vec

fsolve - résout un système d'équations non-linéaires

Karmarkar - Solves a linear optimization problem

NDcost - generic external for optim, computes gradient using finite differences

optim - non-linear optimization routine

qid - linear quadratic programming solver

qp - solve - linear quadratic programming solver builtin

qpssolve - linear quadratic programming solver

readmps - reads a linear program from a MPS file

'Solvers'

* Mots Clés Scilab *

percent % - caractère spécial

backslash \ - division matricielle à gauche

brackets [] - concaténation, Récipients d'une fonction

Colon : - Générateur de séries. Adresse tous les éléments d'une liste ou d'un tableau

comma , - Vierge, séparateur d'instructions ou d'arguments

Comments // ou /* ... */ /commentaires

Comparison - relational operators

dollar \$ - last index

dot . - point

equal = - assignment, comparison, equal sign

extraction - extraction à partir d'une matrice ou d'une liste

getscilabKeywords - liste de Scilab mots clés

hat (^) - élévation à la puissance

affection - affection partielle d'une variable

greater> - greater than comparison >

minus - - opérateur de soustraction, changement de signe

not ~ - négation logique

parentheses () - parenthèses droite et gauche

plus + - opérateur d'addition

quote ' - opérateur de transposition, délimiteur de chaîne de caractères

Semicolon ; - Séparateur de lignes ou d'instructions scilab

Slash / - division à droite et feedback

Star * - opérateur de multiplication

Symbols - opérateurs de Scilab

Tilde ~ - négation logique

* Matrices Creuses *

⊕ Plus de fond

Chfact - factorisation de Cholesky creuse

Chsolve - solveur de Cholesky creux

Ocdmnd - Compute multiple degree ordering

* Console / Variables etc *

CONSOLE

- **clear** - suppression de variables
- **clearglobal** - Supprime des variables globales
- **exists** - teste l'existence d'une variable
- **global** - définition de variables globales
- **isdef** - vérifie si une variable existe
- **isglobal** - checks if a local variable has a global counterpart
- **nommage** - règles de nommage des variables et fonctions
- **predef** - protection de variables
- **who** - liste de variables
- **who_user** - listing of user's variables
- **whos** - liste de variables courantes dans un format détaillé

- **exit** - termine la session courante de scilab
- **mode** - sets or queries the mode echoing scilab instructions to console
- **quit** - quitte scilab ou décrémente le niveau de pause
- **scilab** - lance scilab en mode interactif ou batch avec options
- **startup** - fichier 'startup'

- **errclear** - 'éffacement' d'une erreur
- **error** - affichage d'un message d'erreur
- **errortable** - table of error messages
- **lasterror** - get last recorded error message
- **warning** - messages d'avertissement à l'attention des utilisateurs

- **getdebuginfo** - get information about scilab to debug
- **getmodules** - returns list of modules installed in scilab
- **recursion limit** - get or set recursion limit
- **Scilargs** - arguments sur la ligne de commande scilab
- **getScilabMode** - returns scilab mode

Chaines de Caractères

- **ascii** - strings ASCII characters
- **asciimat** - conversion d'une matrice de chaînes ASCII
- **blanks** - create string of blank characters
- **char** - converts in a character array
- **convstr** - case conversion
- **emptystr** - zero length string
- **eval** - évaluation d'une matrice de chaînes de caractères
- **evstr** - évalue des expressions Scilab et concatène leurs résultats
- **grep** - find matches of a string in a vector of strings
- **isalpha** - check that characters of a string are alphanumeric
- **isascii** - tests if character is a 7-bit US-ASCII character
- **isdigit** - check that characters of a string are digits between 0-9
- **isletter** - check that characters of a string are alphabetic letters
- **isnum** - teste si un chaîne de caractères représente un nombre
- **justify** - justify character array
- **length** - length of object
- **part** - extraction of characters from strings
- **prettyprint** - From any Scilab datatype, provide a representation to the TeX, LaTeX or MathML formats
- **regexp** - finds substring matching exp. **strchr** - finds last occurrence of character
- **Sci2exp** - converts expression to string
- **strcat** - Concatenates strings
- **Strchr** - find first occurrence of character
- **strcmp** - compare strings
- **Strcspa** - copy characters from strings
- **Strindex** - Search pos of lstring in another
- **String** - Conversion en chaîne de chrs.
- **Strings** = Scilab object, character
- **Stripblanks** - strips leading/trailing spaces
- **Strncpy** - copy characters from strings
- **strrev** - returns string reversed
- **Strsplit** - split into a vector of strings
- **strspan** - get span of char. set in string
- **strstr** - locates substring
- **strsubst** - substitute char. string for another in string
- **stetod** - get span of character set
- **stetok** - split string into tokens
- **tokenpos** - token pos in string
- **tokens** - returns the tokens of a character string

* 3D Plot *

GRAPHIQUE

- comet3d - 3d comet animated plot
- contour - level curves on a 3D surface (voir aussi contours)
- eval3d - Values of a function on a grid
- eval3dp - Calcule les facettes d'une surface 3D paramétrique
- fac3d - dessin en 3D d'une surface
- fplot3d - 3d plot of a surface defined by a function
- fplot3d1 - 3d gray or color level plot of a surface defined by a function
- gengfac3d - Calcule les facettes d'une surface 3D non-paramétrique
- geom3d - projection 3D vers 2D après une instruction dessin 3D
- hist3d - représentation 3D d'un histogramme
- Mesh - 3D mesh plot
 - nf3d - Conversion de facettes pour plot3d
 - param3d1 - plots a set of curves in a 3D cartesian frame. 1D mesh plot.
 - param3d properties - description of the 3D curves entities properties
 - plot3d - 3d surface plot
 - plot3d1 - 3d gray or color level
 - plot3d2 - plot surface defined by rectangular facets
 - plot3d3 - mesh plot surface defined by rectangular facets
 - Scatter3 - 3d scatter plot
- Secto3d - Conversion de surfaces 3D
- Soig - 3d surface plot
- Surface properties -

* Éclairage *

- light - creates a light object in the given axes
- light properties - light overview
- lighting - lighting overview

* Axes Operations *

axes properties - description of the axes entity properties

gca - return handle of current axes

gda - return handle of default axes

isoview - Règle la vue isométrique des repères graphiques

newaxes - Creates a new Axes entity

plotframe - dessine un cadre avec graduations et déf. de l'échelle

replot - recadre la zone affichée d'un ou plusieurs repères graphiques

rotate_axes - rotation interactive d'un handle de type ft

Sca - Set the current axes entity

Sda - Set default axes

Square - Échelle isométrique 2D (change la taille de la fenêtre)

Sub plot - quadrille virtuellement la figure et fixe dans une case

Unzoom - Unzoom graphiques

Xgrid - ajoute une grille sur un dessin 2D ou 3D

Xsetech - Sélectionne la sous-fenêtre d'une fenêtre pour les graphiques

Zoom_rect - Zoom Selection de graphique current.

* Couleurs *

addcolor - ajoute de nouvelles couleurs à la table des couleurs

Color - returns the color id of a color

Color_list - liste des noms de couleurs prédéfinies

Colorbar - draw a colorbar

Colormap - using colormaps

getcolor - opens a dialog to show colors in the current colormap

hsv2rgb - Converts HSV colors to RGB

name2rgb - returns the RGB values of a named color

rgb2name - returns the name of a color

Uigetcolor - opens a dialog for selecting a color

* Nuanciers *

autumn colormap - red through orange to yellow

bone colormap - gray with a light blue tone

cool colormap - Cyan to Magenta

copper colormap - black to a light copper tone

gray colormap - linear gray

hot colormap - red to yellow

hsv colormap - Hue - Saturation - Value

jet colormap - blue to red

ocean colormap - linear blue

parula colormap - blue to yellow

pink colormap - Sepia tone colorization on BW images

rainbow colormap - red through orange, yellow, green, blue - violet

spring colormap - Magenta to yellow

summer colormap - green to yellow

white colormap - Completely white

winter colormap - blue to green

* Fenêtres - Figures *

clf - clears and resets a figure or a frame window

colordef - set the color look-and-feel for graphic figures

drawlater - makes axes children invisible

drawnow - draw hidden graphics entities

gcf - return handle of current graphic window

gdf - return handle of default figure

pixel-drawing-mode - mode d'actualisation RVB des pixels dessinés

scf - set current graphic figure (window)

sdf - set default figure Xname - Charge nom fen. graph. courante

show-window - met la fenêtre une au premier plan

winstd - renvoie la liste des fenêtres graphiques ouvertes

Xdel - Supprime des fenêtres graphiques

L
I
N
O
P
H
I
C
G

* Annotation + Axes Dessin *

captions - draw graph captions

label entity properties - Label entity properties

legend - draw graph legend

Legend entity properties - legend entity properties

legends - dessine de légendes de courbes 2D

title - display a title on a graphic window

zlabel - sets the z-axis label

xtitle - add titles on a graphic window



axis properties - description of the axis entity properties



drawaxis - dessine un axe



graduate - graduations simplifiées des axes 2D



xgraduate - graduation des axes

* Bar Histogram *

bar - histogramme de barres

barh - histogramme horizontal de barres

barhomogenize - uniformise le style des barres dessinées dans un repère graphique

* Texte *

graphics_fonts - description des fonts utilisées par les graphiques

Stringbox - Calcule le rectangle englobant d'un texte label

text properties - text entity properties



title page - ajoute un titre au milieu d'une fenêtre graphique



xinfo - change le texte affiché dans la barre d'état fenêtre



xfont - load a font in the graphic context or query loaded font



xnumb - dessine des nombres



xstring - dessine des chaînes de caractères



xstringb - dessine des chaînes de caractères dans une boîte



xstringl - calcule les dimensions d'une boîte entourant des chaînes de caractères

* Matrices Further Functions *

aff2ab - linear(affine) function to A,b conversion

chol - Factorisation de Cholesky

inv - Inverse d'une matrice

linsolve - solveur d'équation linéaire

lsq - linear least square problems

lu - factorisation LU

pinv - pseudo inverse d'une matrice

qr - factorisation QR

rankqr - rank revealing QR factorization

Eigenvalue and Singular Value

balanc - matrix or pencil balancing

bdiag - bloc-diagonalisation, vecteurs propres généralisés

hess - Forme de Hessenberg

pbig - projection sur des sous-espaces propres

projspec - spectral operators

psmall - spectral projection

Schur - [ordered] Schur decomposition of matrix and pencils

Spec - Valeurs propres d'une matrice

Sva - approximation de valeurs singulières

Svd - décomposition en valeurs singulières

Noyau

Colcomp - compression de colonnes, noyau

fullrg - factorisation de rang plein

fullrfk - factorisation de rang plein de A^k

im_inv - inverse image

Kernel - noyau

range - image de A^k

rowcomp - compression de lignes, image

X
—
R
T
A

* Matrice Compagnon *

companion - matrice compagnon

ereduc - Computes matrix column echelon form by qz transform

fstair - Computes pencil column echelon ~~form~~ form by qz transform

gl ever - inverse d'un faisceau de matrices

Kroneck - Kronecker form of matrix pencil

lyap - equation de Lyapunov

pencan - canonical form of matrix pencil

penlaur - laurent coefficients of matrix pencil

quaskro - quasi-Kronecker form

rand pencil - random pencil

rowshuff - shuffle algorithm.
Factorisation

givens - transformation de Givens

householder - Matrice de Householder, Symétrique par rapport à un plan

Sq,root - factorisation hermitienne W^*W

Matrices de Markov

classmarkov - recurrent and transient classes of Markov matrix

eigenmarkov - Normalized left and right Markov eigenvectors

genmarkov - generates random Markov matrix with recurrent and transient class

State-Space Matrices

Coff - résolvante (Méthode des cofacteurs)

nlev - algorithme de Leverrier

Sous-espaces vectoriels

SpanInter - Subspace intersection

Spanplus - Sum of subspaces

Span two - Sum and intersection of subspaces

proj - projection

MATRIX

* Fichiers *

copyfile - copies a file newest - returns newest file in set of files
deletefile - deletes a file saveformat - format of files produced by 'save'
dispfiles - displays opened files properties
fileinfo - provides information about a file
findfiles - finding all files with a given file specification
fprintfMat - writes a matrix file
fscanfMat - reads a matrix text file
getmd5 - get md5 checksum
getURL - download a url (http, https, ftp..)
%io - Variable which keeps standard input and output (filedesc.)
isfile - checks whether argument is an existing file
listfiles - list of files Scanf - converts formatted input or std input
listvarinfil - list of variables in a saved data file
mclearerr - resets binary file access errors
fclose - closes an opened file
mdelete - deletes file(s) Scanf.conversion - mscanf, msscanf, mfscanf
meof - checks if end of file has been reached
merror - tests the file access errors indicator
mfprintf - converts, formats and writes data to file
msscanf - reads its input from a character string
mgeti - reads byte or word in binary format returns 'int' type
mgetl - reads lines from a text file
mgetstr - reads a character string from a file
mopen - opens a file in Scilab
movefile - move file or directory
mput - writes byte or word in a given binary format
mputl - writes strings to a text file
mputstr - writes a character string to a file
mseek - sets current position in a binary file
mtell - returns offset of current byte relative start of file

* Statistiques *

- cumulated distribution functions

binomial - binomial distribution probabilities

cdfbet - fonction de répartition de la distribution Beta

cdfbin - fonction de répartition de la distribution binomiale

cdfchi - fonction de répartition de la distribution du chi-de

cdfchn - fonction de répartition de la distribution du chi-de cent

cdfF - fonction de répartition de la distribution de Fisher

cdfFnc - fonction de répartition de la distribution de Fisher non cent

cdfgam - fonction de répartition de la distribution gamma

cdfnbn - fonction de répartition de la distribution binomiale

cdfnor - fonction de répartition de la distribution de normal

cdfpoi - distribution de Poisson • **cdfSt** - distribution de Stud

- central tendency -

geomean - Moyenne géométrique

harmean - moyenne harmonique l'inverse de la moyenne des inverses (zéros)

mean - Mean (row mean, column mean) of vector/matrix entries

meanf - weighted mean of a vector or a matrix

trimmean - trimmed mean of a vector or a matrix

- descriptive statistiques -

Center - center

Correl - correlation of two variables

COV - covariance matrix

COVAR - covar covariance of two variable

histc - calcule un histogramme

Median - médiane des termes d'une matrice

stdev - écart-type des termes d'une matrice

stdevf - standard deviation

Variance - variance d'un vecteur, matrice de nombres réels ou complexes

Variancef - variance et moyenne d'un vecteur, matrice

Wcenter - center and weight

* Statistiques Continued.*

- data with missing values

`nansum` - somme cumulative ignorant les valeurs NaNs

`nand2mean` - difference of the means of two independent samples

`max` - ~~nanmax~~ - max ignoring NaNs

`nanmean` - mean ignoring NaNs

`nanmeans` - mean ignoring NaNs with a given frequency

`nanmedian` - median of the values of a numerical vector or matrix

`min` - ~~nanmin~~ - min ignoring the NaN's

`nanreglin` - ~~nanreglin~~, régression linéaire

`nanstdev` - standard deviation ignoring the NaNs

`nansum` - sum of values ignoring the NaNs

`thrownan` - eliminates NaN values

- Hypothesis Testing -

`ftest` - Fisher ratio

`fisueq` - Fisher ratio for samples of unequal size

- measures of shape -

`cmoment` - central moments of all orders

`Moment` - non central moments of all orders

`perctl` - computation of percentiles

`quart` - computation of quartiles

- principal component analysis -

`pca` - computes principal components analysis with standardized variables

`princomp` - principal components analysis

~~show_pca~~ - visualization of principal components results.

- Échantillonnage -

`Sample` - tirage avec remise

`Samplef` - tirage avec remise d'un échantillon les valeurs fréq. données d'une population conn.

`Samwr` - tirage sans remise

`tabul` - frequency of values of a matrix or vector

`reglin` - régression linéaire.

* Listes Structures Cells *

(tout en sa page)

Cell - creates a cell array of empty matrices

Cell2mat - converts a cell array to a matrix

Cellstr - converts string vector (or strings matrix) into cell array of s

iscell - checks if variable is a cell array

iscellstr - Vérifie si une variable est une cell

makecell - creates a cell array

num2cell - converts an ~~var~~
~~array~~

boolean - objets Scilab, variables booléennes et opérateurs & |

definedfields - returns an index of list of defined

fieldnames - récupère le nom des champs pour ~~list~~ ^{tlist, mlist ou} struc

getfield - extraction d'un champ d'une liste

hypemat - initialisation d'une matrice à N dimensions via 'matrix'

hypermatrices - Objet scilab, matrices à N dimensions

isfield - teste l'existence d'un ou ^{plusieurs} champs ^{Pour une struc} dans une list / ml

isstruct - Vérifie si une variable est une structure

list - objet scilab, définition d'une liste

listcat - concaténation de listes

listsize - nombre d'éléments des objets list, tlist ~~, mlist~~, mlist

Matrices - objet scilab Matrices

Mlist - objet scilab, définition d'une liste typée

null - suppression d'un terme d'une liste

rlist - définition d'une fraction rationnelle

setfield - insertion dans un champ d'une liste

struct - créer une struct

tlist - objet scilab. Définition d'une liste typée

type - retourne le type d'un objet scilab

typename - associe un nom à un type de variable

typeof - type explicite ou code de surcharge d'un

Objet scilab.

* Traitement XML *

- htmlDump - dump a HTML document
- htmlRead - read a HTML stream from a local or distant file
- htmlReadStr - read a HTML tree from a string
- htmlWrite - write a HTML document in a file
- xmlAddNs - add a namespace to a XML element
- xmlAppend - append an element to the parent children
- xmlAsNumber - convert the result of a XPath query to row of numbers
- xmlAsText - convert the XPath query to row of strings
- xmlDelete - delete a XML document
- xmlDocument - create a new xml document
- xmlDTD - create a xml DTD object
- xmlDump - dump a XML object
- xmlElement - create a new XML element
- xmlFormat - format a scilab variable into XML
- xmlGetNsByHref - get a namespace by its href
- xmlGetNsByPrefix - get a namespace by its prefix
- xmlGetOpenDocs - get all open xml documents or open validation files
- xmlIsValidObject - test the existence of a xml object
- xmlName - retrieve the name of the elements
- xmlNs - create a new XML Namespace
- xmlRead - Read a XML stream from a local or distant file
- xmlReadStr - read a XML tree from a string
- xmlRelaxNG - create a XML Relax NG object
- xmlRemove - Remove an element or list of elements from parents
- xmlSchema - create a XML schema object
- xmlSetAttributes - set the attributes name and value
- xmlValidate - validate document using a DTD, a Relax NG, or Schema
- xmlWrite - write a XML document in a file
- xmlPath - make XPath query on a XML document

* Interpolation *

bspln3val - 3d spline arbitrary derivative evaluation function
cshp2d - bidimensional cubic shepard (scattered) interpolation
eval_cshp2d - bidimensional cubic shepard interpolation evaluation
interp - cubic spline evaluation function
interp1 - one dimension interpolation function
interp2d - bicubic spline (2d) evaluation function
interp3d - 3d spline evaluation function
interpn - linear interpolation
linear_interpn - n dimensional linear interpolation
lsg_spln - weighted least squares cubic spline fitting
Smooth - smoothing by spline functions
Spln - cubic spline interpolation
Spln2d - bicubic spline gridded 2d interpolation
Spln3d - spline gridded 3d interpolation

* Linear Analysis * - Controllability Observability

abinv - AB Invariant Subspace

Cainv - Dual of abinv

Cont_mat - controllability matrix

Contr - controllability, controllable subspace, staircase

Contrss - controllable part

ctr_gram - controllability gramian

obscont - observer based controller

observer - observer design

obsu_mat - observability matrix

obsuss - observable part

dt_ility - detectability test

obs_gram - observability gramian

Ui_observer - unknown input observer

Unobs - unobservable subspace

* Linear Analysis - Frequency Domain *

black - Black-Nichols diagram of a linear dynamical system

bode - bode plot

bode - asympt - bode plot asymptote

Calfreq - frequency response discretization

dbphi - frequency response to phase and magnitude representation

freq - frequency response

freson - peak frequencies

gainplot - magnitude plot

hallchart - draws the hall chart

nicholschart - nichols chart

nyquist - nyquist plot

nyquistfrequencybounds - computes freqs. which nyquist locus enters and leaves a rectangle

phasemag - phase and magnitude computation

phaseplot - frequency phase plot

repfreq - frequency response

Svplot - singular-value sigma plot

trzeros - transmission zeros and normal rank

* Stability - Linear Analysis *

bstab - hankel approximat

dtsi - continuous time dynamical systems stable / anti-stable composition

evans - evans root locus

g_margin - gain margin and associated crossover frequency

p_margin - phase margin and associated crossover frequency

pzr - pole zero plot

routh - ~~mat~~ routh's table

Sgrid - S-plane grid lines

Show_Margins - display gain and phase margin and associated frequencies.

stability - stabilizability test

Zgrid - Z grid plot

(Sensitivity) * - Sensi - sensitivity functions

* Linear Analysis* - Time Domain

arsimul - armax simulation

cslm - simulation (time response) of linear system

damp - Natural frequencies and damping factors

dsimul - state space discrete time simulation

flts - time response (discrete time, sampled system)

ltitr - Discrete time response (state space)

narsimul - armax simulation (using rtitr)

rtitr - discrete time response (transfer matrix)

* Linear System Factorization *

Colinout - inner-outer factorization

copfac - right coprime factorization continuous systems

dcf - double coprime factorization

fspec - Spectra factorization of cont. time dynamical system

fspecg - Stable factorization of continuous time dynamical systems

fstabst - youla's parametrization of cont. time linear dynamical systems

gtild - tilde operation

lcf - Cont. time dynamical systems normalized coprime factorizati

rowinout - inner-outer factorization

Specfact - spectral factor

Sysfact - system factorization

%sn - jacobi's elliptic func.
%K - jacobi's complete elliptic integral

* Special Functions *
amell - jacobi's am function besselh - bessel functions of the third kind (aka Hankel functions)

besselh - bessel functions of the third kind (aka Hankel functions)

beta - beta function (Euler integral of the first kind)

calerf - computes error functions

dawson - compute the dawson (scaled imaginary error) delip - complete and incomplete elliptic integral of first kind digamma - derivative of gamma function, psi function erf - the error function erfc - the complementary error function

erfcx - scaled complementary error func.

erfi - imaginary error erfinv - inverse error findm - ellip. filter

gamma - gamma gammaln - log of gamma legendre - legendre func.

* Linear System Representation *

abcd - State space matrices

arma - scilab arma library

arma2p - extract polynomial matrices from ar representation

arma2ss - transform an armax data struct, into state-space.

armac - scilab description of an armax process.

cont_gsm - transfer to controllable state-space

des2ss - descriptor to state-space

des2tf - descriptor to transfer function

freq2tf - transfer function realization from freq. response

lsslist - Scilab linear state-space function definition

markp2ss - markov parameters to state-space

pffss - partial fraction decomposition

pol2des - polynomial matrix to descriptor form

sm2des - system matrix to descriptor

ss2des - (polynomial) state-space to descriptor form

ss2ss - State-space to state-space conversion, feedback, injection

ss2tf - conversion from state-space to transfer function

ss2zp - SIMO state-space system to zero pole gain representation

ssprint - pretty print for linear system

syslin - linear system definition

syssize - size of state-space system

sysmat - System matrix

tf2des - transfer function to descriptor

tf2ss - transfer to state-space

tf2zp - SIMO transfer function to zero pole gain representation

zp2ss - zero pole gain to state-space

zp2tf - zero pole gain to transfer function

zpk - zero pole gain system representation

zpk2ss - zero pole gain to state-space

zpk2tf - zero pole gain to transfer function

A beautiful day, made perfect
For old to rule over the new
With clothes,
Made of war, money, property,
A voice, A way of speaking
A belief in them
Them. Who ruled over your parents.

So, what about you?

To follow?

Thousands of years. WARS.

Millions of lifeforms to, Money?
A belief.

A lifetime to explore yourself

And their disgusting corrupt world
Of cloth.

Find love my friend, don't seek
A response to them

Who wear one thing in day light
And another behind walls

Look to yourself.

Look for an answer.

Find God.

If everything came from nothing.
You could believe

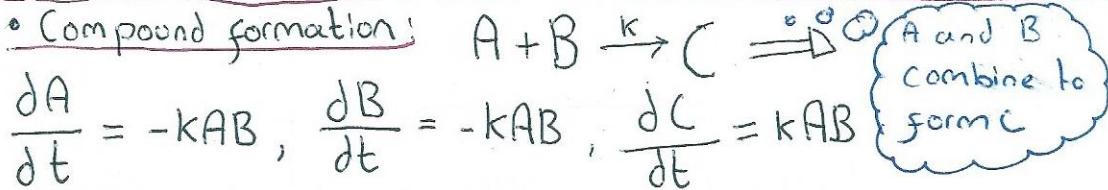
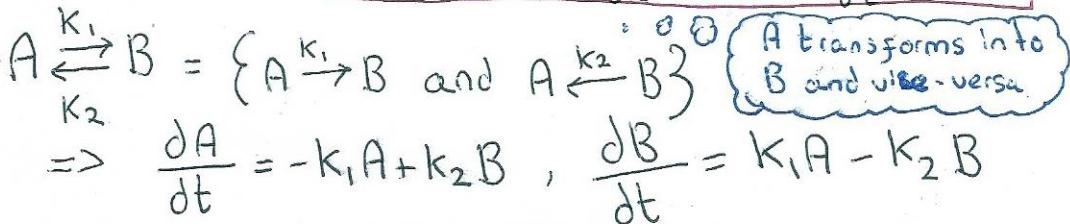
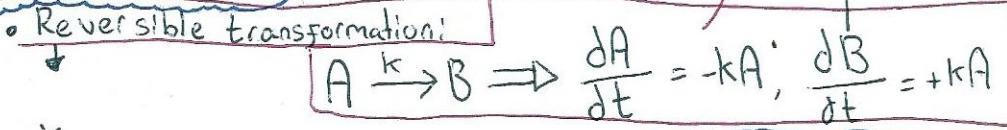
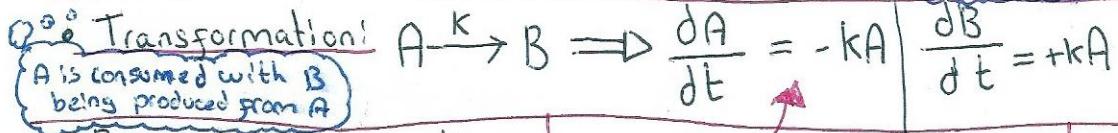
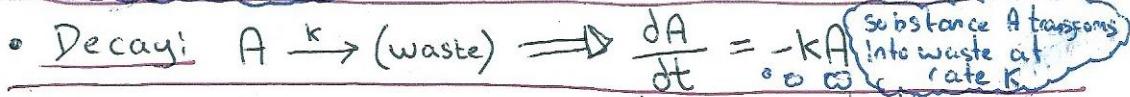
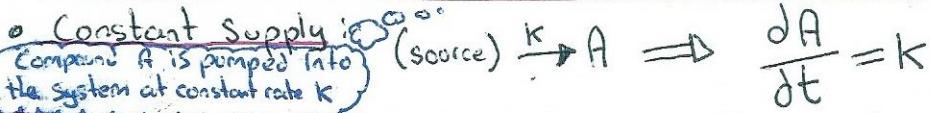
In anything.

Do you?

~ Charlotte Elisabeth
Amiel

• Motion of particles: $\frac{dX}{dt} = V, \frac{dP}{dt} = F \quad | m \frac{d^2 X}{dt^2} = F$

$$\frac{dX}{dt} = f(x) \text{ with } f = \begin{pmatrix} P/m \\ F(X, P) \end{pmatrix} \in \mathbb{R}^6.$$



37 Z_0 Impédance caractéristique

38 t température Celsius

39 G constante de gravitation de Newton

40 atm pression atmosphérique standard.

les Calculs

! factoriel $n \times (n-1) \times (n-2) \times (n-3) \cdots (1+1)$

équation quadratique: $(ax^2 + bx + c = 0)$

équation cubique: $(ax^3 + bx^2 + cx + d = 0)$