sparseMatrixType={*smType*,*smMode*,*smDiag*}

Provides optional information to describe both sparse matrix A and sparse matrix G. All of the parameters are keywords.

smType: GENERAL, SYMMETRIC, HERMITIAN, TRIANGULAR, DIAGONAL, BLOCK_TRIANGULAR, or BLOCK_DIAGONAL.

smMode: LOWER or UPPER.

smDiag: DIAG or NON_DIAG.

See **Optional Sparse Matrix Information** on page III-156 for details.

vectorX=*wx* Designates the 1D wave *wx* as vector X for the MV and TRSV operations.

The wave *wx* must be of the same data type as the sparse matrix data and it must not contain INFs or NaNs.

vectorY=*wy* Designates the 1D wave *wy* as vector Y for the MV operation.

The wave *wy* must be of the same data type as the sparse matrix data and it must not contain any INFs or NaNs.

Details

MatrixSparse supports data waves with single-precision and double-precision floating point real and complex data types. See **MatrixSparse Operation Data Type** on page III-155 for details.

Index waves must be signed 64-bit integer. See **MatrixSparse Index Data Type** on page III-156 for details.

MatrixSparse does not support waves containing NaNs or INFs.

MatrixSparse math operations (ADD, MV, MM, SMSM, TRSV) require that input sparse matrices be in CSR format. The math operations that return sparse matrices (ADD, SMSM) create output sparse matrices in CSR format.

The conversion operations (TOCOO, TOCSC, TOCSR, TODENSE) accept inputs in COO, CSC, CSR, or dense formats.

Output Variables

MatrixSparse sets these automatically created variables:

V_flag Set to 0 if the operation succeeded or to a non-zero error code.

Examples

You can find examples using MatrixSparse under **MatrixSparse Operations** on page III-156.

See Also

Sparse Matrices on page III-151 for background information about Igor's sparse matrix support.

MatrixSparse Operations on page III-156 for details on each supported operation and examples.

Matrix Math Operations on page III-138 for discussion of non-sparse Igor matrix routines.

MatrixSVBkSub

MatrixSVBkSub ***matrixU*****,** ***vectorW*****,** ***matrixV*****,** ***vectorB***

The MatrixSVBkSub operation does back substitution for SV decomposition.

Details

Used to solve matrix equation Ax=b after you have performed an SV decomposition.

Feed this routine the M_U, W_W and M_V waves from **MatrixSVD** along with your right-hand-side vector b. The solution vector x is returned as M_x.

The array b can be a matrix containing a number of b vectors and the M_x will contain a corresponding set of solution vectors.

Generates an error if the dimensions of the input matrices are not appropriate.