MatrixSparse Inputs

Inputs to the MatrixSparse operation are documented and can be understood in terms of the following conceptual matrix, vector, and scalar inputs:

Sparse matrix A is defined by the rowsA and colsA keywords and by one of the cooA, csrA or cscA keywords. In this command, from the DemoSparseMatrixMV example above, the keywords specifying sparse matrix A are highlighted in red:

```
MatrixSparse rowsA=4, colsA=4, csrA={values,columns,ptrB}, vectorX=vector, operation=MV
```

Sparse matrix A is used in all MatrixSparse operations that take one or more sparse matrix inputs. MatrixSparse math operations (ADD, MV, MM, SMSM, TRSV) require that input sparse matrices be in CSR format.

Sparse matrix G is defined by the rowsG and colsG keywords and by one of the cooG, csrG or cscG keywords. Sparse matrix G is used only in MatrixSparse operations that take two sparse matrix inputs (ADD and SMSM). These operations require that input sparse matrices be in CSR format.

Matrix B is defined by the matrixB keyword and is used in MatrixSparse operations that take one or more dense matrix inputs (currently just the MM operation).

Matrix C is defined by the matrixC keyword and is used in MatrixSparse operations that take two dense matrix inputs (currently just the MM operation).

Vector X is defined by the vectorX keyword and is used in MatrixSparse operations that take one or more vector inputs (currently the MM and TRSV operations).

Vector Y is defined by the vectorY keyword and is used in MatrixSparse operations that take two vector inputs (currently just the MV operation).

Alpha and ***alphai*** are defined by the alpha and alphai keywords and are used in all MatrixSparse operations that take one or more scalar inputs (currently the MM, MV, and TRSV operations). Alphai is used only when operating on complex data.

Beta and ***betai*** are defined by the beta and betai keywords and are used in MatrixSparse operations that take two scalar inputs(currently the MM and MV operations). Betai is used only when operating on complex data.

MatrixSparse Operation Data Type

The operation data type is the data type required for all data waves participating in the MatrixSparse command. Here "data waves" means the values wave in the representation of a sparse matrix and the matrix wave representing a dense matrix.

If a given MatrixSparse command takes sparse matrix A as an input then the values wave (the first wave specified by the cooA, cscA, or csrA keywords) determines the operation data type. If the command does not take sparse matrix A then the matrix B wave (specified by the matrixB keyword) determines the operation data type.

If there are multiple input data waves, such as for the ADD operation which adds two sparse matrices, the data type of all input data waves must be the same.

The operation data type must be single-precision or double-precision floating point and can be real or complex. MatrixSparse does not support waves containing NaNs or INFs.

Output waves are created using the operation data type.

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.