MatrixOp Operation

The **MatrixOp** operation (page V-550) improves the execution efficiency and simplifies the syntax of matrix expressions. For example, the command

```
MatrixOp matA = (matD - matB x matC) x matD
```

is equivalent to matrix multiplications and subtraction following standard precedence rules.

See **Using MatrixOp** on page III-140 for details.

MatrixSparse Operation

The **MatrixSparse** operation (page V-580) can improve performance and reduce memory utilization for calculations involving large matrices the elements of which are mostly 0. See **Sparse Matrices** on page III-151 for details.

Matrix Commands

Here are the matrix math operations and functions.

General:

MatrixCondition(*matrixA*)
MatrixConvolve *coefMatrix*, *dataMatrix*
MatrixCorr [*flags*] *waveA* [, *waveB*]
MatrixDet(*matrixA*)
MatrixDot(*waveA*, *waveB*)
MatrixFilter [*flags*] *Method dataMatrix*
MatrixGLM [/Z] *matrixA*, *matrixB*, *waveD*
MatrixMultiply *matrixA*[/T], *matrixB*[/T] [, *additional matrices*]
MatrixOp [/O] *destwave = expression*
MatrixRank(*matrixA* [, *maxConditionNumber*])
MatrixTrace(*matrixA*)
MatrixTranspose [/H] *matrix*

EigenValues, eigenvectors and decompositions:

MatrixBalance [*flags*] *srcWave*
MatrixEigenV [*flags*] *matrixWave*
MatrixFactor [*flags*] *srcWave*
MatrixGLM *matrixA*, *matrixB*, *waveD*
MatrixInverse [*flags*] *srcWave*
MatrixLUD *matrixA*
MatrixLUDTD *srcMain*, *srcUpper*, *srcLower*
MatrixReverseBalance [*flags*] *scaleWave*, *eigenvectorsWave*
MatrixSchur [/Z] *srcMatrix*
MatrixSVD *matrixA*

Linear equations and least squares:

MatrixGaussJ *matrixA*, *vectorsB*
MatrixLinearSolve [*flags*] *matrixA*, *matrixB*
MatrixLinearSolveTD [*/Z*] *upperW*, *mainW*, *lowerW*, *matrixB*
MatrixLLS [*flags*] *matrixA*, *matrixB*
MatrixLUBkSub *matrtixL*, *matrixU*, *index*, *vectorB*
MatrixSolve *method*, *matrixA*, *vectorB*
MatrixSVBkSub *matrixU*, *vectorW*, *matrixV*, *vectorB*

Sparse matrices:

MatrixSparse