Stdlib Sparse matrix API

**Goals**

To be compact instead of being exhaustive. It aims at supplying Fortran users with a minimum (yet useful) number of routines and data structures related to sparse matrices storage and operations. This library is particularly targeted at a non-expert in numerical computation public. Thus we aim at having a simple and easy to use API.

**1 Sparse matrix representations supported**

This section is based on Saad (1994). In that work, a much more complete and extensive list of formats is described. Here we take only the ones that we think are most useful at the moment.

Some questions:

Support one-based indexing?

Support zero-based indexing?

**1.1 Coordinate format (COO)**

Given an by real or complex matrix containing nonzero elements with each element denoted by this format represents using a set of three arrays: *values*, *rows*, and *columns,* as described below.

*values* A real/complex array of size containing the matrix elements in any order.

*rows* An integer array of size containing the row indices of the elements .

*columns* An integer array of size containing the column indices of the elements .

**1.2 Compressed Sparse Row (CSR)**

Given an by real or complex matrix containing nonzero elements with each element denoted by this format represents using a set of three arrays: *values*, *ja*, and *ia,* as described below.

A real/complex array of size containing the matrix elements stored row by row from row 1 to row .

An integer array of size containing the column indices of the elements as stored in the array *values*.

An integer array of size containing the index in the arrays and where each row starts. The value at always has the value .

**1.3 Compressed Sparse Column (CSC)**

This format is similar to the CSR format described previously. The difference is that instead of storing row values we store column values in the array . The exact description of this format is given below.

Given an by real or complex matrix containing nonzero elements with each element denoted by this format represents using a set of three arrays: *values*, *ja*, and *ia,* as described below.

A real/complex array of size containing the matrix elements stored *column* by *column* from column 1 to column .

An integer array of size containing the *row* indices of the elements as stored in the array *values*.

An integer array of size containing the index in the arrays and where each column starts. The value at always has the value .

**2 Creational subroutines**

**3 Conversion subroutines**

**4 Algebraic operations**

**4 Utilities**

**4 Input/Output**

References

Saad, Y., SPARSKIT: A basic tool kit for sparse matrix computation*, 1994.*<https://www-users.cs.umn.edu/~saad/software/SPARSKIT/>