**The Simple Noise Removal (snr) library**

The standalone snr library is the now two-pass “ced algorithm” for identifying noise in CLAS driftchamber data. This note does not describe the algorithm in detail; rather it describes how to *use* the algorithm.

At the time of writing, the snr package is available with its source code in the clas12 svn repository and as a jar file in the clasJlib distribution. So, as with swimmer, magfield, and splot, you can either make a dependency on the snr eclipse project or on the snr jar file.

# Using the library with default settings

To use the library you need to instantiate two objects, both in the cnuphys.snr.clas12 package in the snr jar (or source). The two objects are

Clas12NoiseResult

Clas12NoiseAnalysis

Both objects just have null constructors. The Clas12NoiseAnalysis object holds the algorithm parameters. How the parameters are modified from the default values will be discussed later.

Assume we have instantiated these objects:

Clas12NoiseResult results = new Clas12NoiseResult();

Clas12NoiseAnalysis noiseAnalysis = new Clas12NoiseAnalysis();

The algorithm is invoked with one call to the Clas12NoiseAnalysis object (for each new event):

**public** **void** findNoise(**int** sector[], **int** superlayer[],

**int** layer[], **int** wire[], Clas12NoiseResult results)

so in our case:

noiseAnalysis.findNoise(sector, superlayer, layer, wire, results)

the arrays sector, superlayer, layer and wire are the columns from the clas-io drift chamber “dgtz” bank—i.e., exactly what clas-io provides.

After this call, the results object will have a single array

boolean[] noise;

That array will be the same size and have the same ordering as the dgtz arrays. It is lock-step with the clas-io arrays. That is, hit number 7 is found at sector[7], superlayer[7], layer[7], wire[7], and if (and only if) it has been determined to be noise, then results.noise[7] will be true.

The clas-io arrays sector, superlayer, layer and wire are **not** altered.