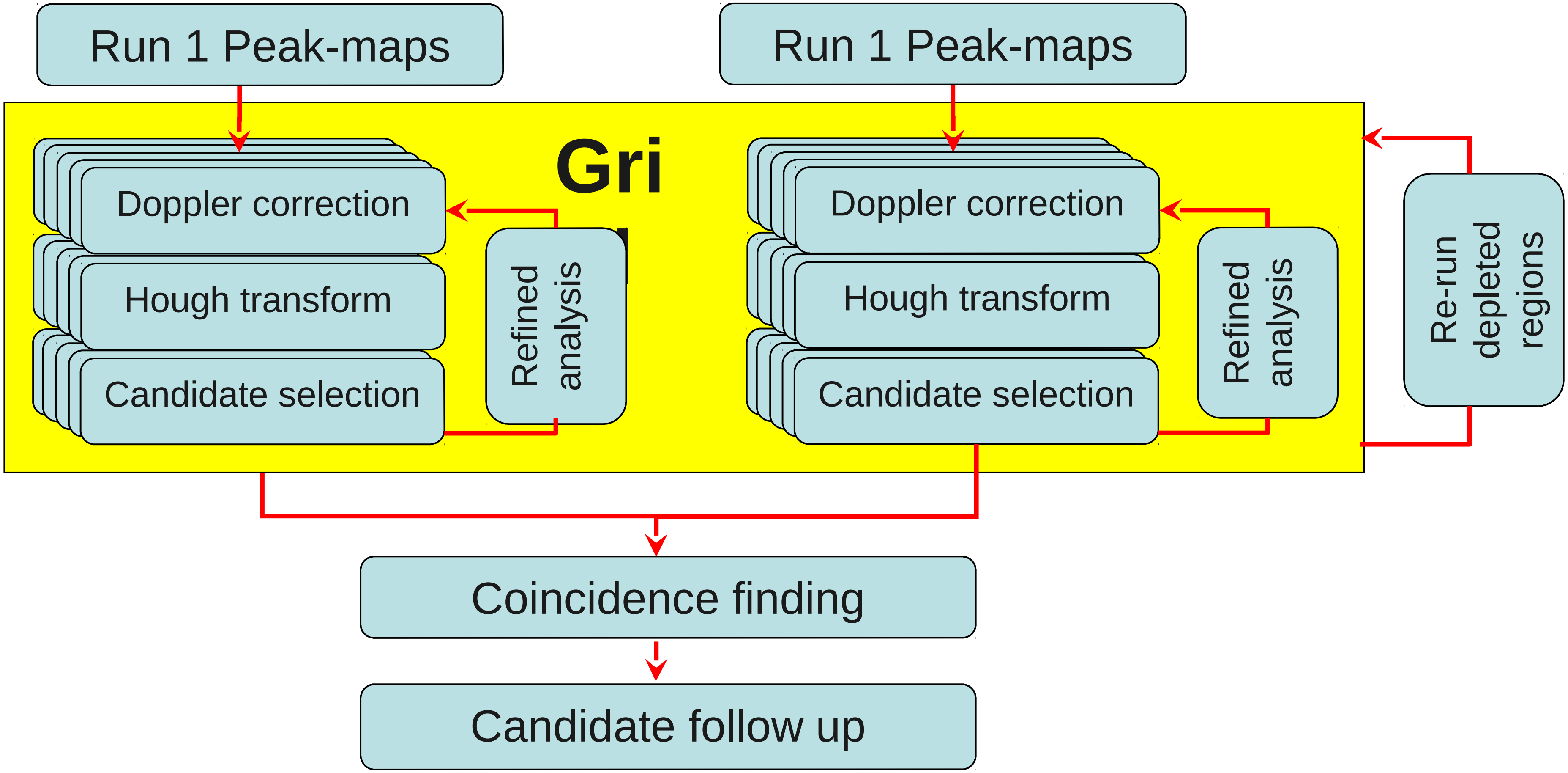
Rome group experiences on managing Grid jobs (at CNAF)

Simple tutorial to submit jobs to the Grid:

https://workarea.ego-gw.it/ego2/virgo/data-analysis/grid/simple-tutorial-for-grid-job-management/

Running the Frequency Hough All-sky continuous wave analysis on the Grid: a job submission and control framework

- Extract of a poster presented at the Amaldi meeting at Warsaw - **LIGO DCC:G1300720-v1**



Analysis configuration

- Two Virgo runs: VSR2 and VSR4
- Two frequency bands:
  - Low band: 20-128 Hz (data resampled at 256 Hz), with 8192 s long FFTs
  - High band: 128-1024 Hz, with 1024 long FFTs
- frequency and sky declination (β) divided in sub-bands
- Each *job* analyzes one frequency sub-band and one sky “slice”.

	VSR2		**VSR4**	
	Low band	**High band**	**Low band**	**High band**
No. of *f* bands	443	1452	443	1089
No. of sky slices	18	72	18	18
No. of jobs to run	**7974**	**104544**	**7974**	**19602**

Job submission and control framework

Job configuration and submission. The framework creates the specific Grid job description files (JDL), minimizing the set of input parameters to be provided by the user: run label (e.g. VSR2), frequency band, width of the *f* and β sub-bands. Storage locations of the input (peak-map files) and output (candidate files) data sets are built accordingly to the user inputs. Finally, it creates a shell script which the user can execute to directly submit the jobs.

Job type is “**parametric**”, where the parameter is frequency: a single JDL manages the execution of all frequency sub-bands. Therefore, the number of jobs description files created reduces to the number of sky slices.

Job Failure tracking. Analyzing the output data set (file name parsing), the system identifies the failed jobs and automatically recreates a new set of job description files, which allows users to submit the failed jobs only.

“sky slice splitting”. In some cases job failures are due to reaching the memory limits. A functionality of the system is to split the β parameter bandwidth of the jobs failed for memory reasons (user input).

Data management issues and solutions

- Input data for the analysis code are peak-map binary files, each one covering 10 days of data and the full frequency range, stored on disk volumes network-mounted on the worker nodes.
- Output data are text files containing the selected candidates.
- Critical increase of the execution time observed when too many jobs ran at the same time. This happened because each job needed to open all the input files to extract the frequency sub-band to analyze.
- Furthermore, the jobs wrote the output directly on the shared areas, thus contributing to the saturation of the network bandwidth.

To solve the issue:

- re-organize peakmap files splitting them in frequency, to make each job read 1 input file only;
- Write the output on the worker node local disk, and copy it on the shared areas at the end of the analysis using LCG tools.

To-do list

- Automatize the identification and re-submission of failed jobs.
- Create a web interface which to simplify the monitoring of the on-going analysis.
- Complete the implementation of Grid functionalities to access the input data set and transfer the results of the analysis. This will allow to exploit all the Grid farms accessible by the Collaboration for our analysis