SKA Pulsar Search: Technological Challenges and Best Algorithms Development

C.Baffa, E.Giani on behalf of the SKA-TDT-Group

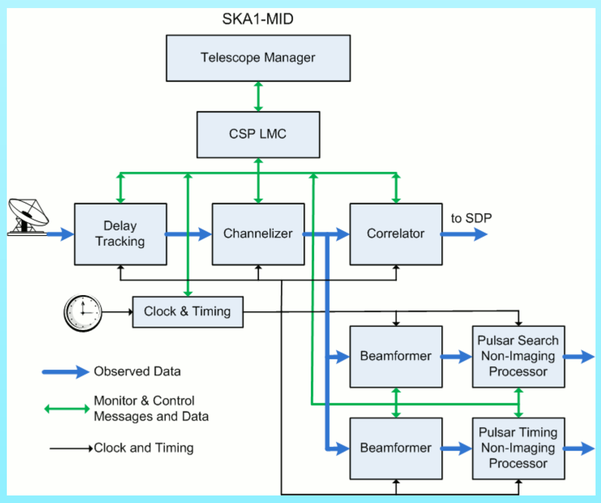
One of the key scientific projects of the SKA radio telescope is a large survey for pulsars both in isolated and binary systems. The data rate of the pulsar search engine is expected to reach 0.6TeraSamples/sec. For the purposes of extracting hidden pulses from these streams, we need a complex search strategy which allows us to explore a three dimensional parameter space and it requires approximately 10PetaFlops. This problem is well suited for a parallel computing engine, but the dimensions of SKA bring this problem to a new level of complexity. An up-to-date study shows that this operation would require more than 2000 GPUs.

SKA Overview

The Square Kilometer Array (SKA) is a giant radio telescope project and will be the largest Radio Telescope ever built
It will have two phases: A smaller one, SKA1 and the final SKA. It will be divided in two locations, South Africa and Australia.
SKA will consist of three networks of antennas:
- Low array, 1024 groups of 256 small antennas in phase 1
- Survey array, 96 medium size dish antennas in both phases
- Mid array, 256 large dish telescopes in phase 1

5 main instruments:
- Low frequency array correlator
- Mid frequency array correlator and pulsar beam-former
- Pulsar search machine
- Pulsar timing machine
- Survey array correlator

SKA MID DATA FLOW



1 - The Pulsar Search Survey

One of the SKA's main scientific drivers is a large pulsar survey. An important fraction of these objects is expected to be found in binary systems, which can provide crucial tests of general relativity in a strong field environment. In order to obtain the maximum survey sensitivity, we require a coherent sum of the SKA dishes. This procedure produces a narrow beam on the sky and, consequently, a very slow survey speed. A possible solution to avoid this inconvenience is forming a large number of beams. This would dramatically speed up the survey. The mitigation strategy that would provide the best compromise for the pulsar survey against the speed issues is to build a massive parallel searching engine able to operate in real time.

The SKA Pulsar Search beam-former, in the baseline design, will provide 2222 simultaneous beams, digitized every 50μs, on up to 15000 frequency channels, for a total of 0.7TeraSamples/sec. It is required a complex search strategy to extract the hidden pulses from these streams. To correctly add up the singles pulses, thus overcoming the overwhelming background noise, the pulsar processor needs to explore a three dimensional space: the pulse period, the interstellar medium frequency dispersion and the peculiar acceleration of the source.

This *optimization* problem is well suited for a parallel computing engine, and there are already example implementations for present-day radio telescopes. However, the dimension of the SKA project brings this problem to a new level of complexity. Based on the performances of the computer processors that will be available at the start of the SKA construction phase, we expect that this calculation will requires a considerable number of scientific-grade GPUs boards. The resulting machine will be gigantic, and will pose a large burden, both in cost and in logistics, on the available power supply.

In addition, we need to take into consideration the calculation required to perform all the operations in real time. Due to the large data flow involved, it is impossible to retain more than a tiny fraction of the raw data. The SKA Pulsar Search input is approximately 1PetaBytes on each cycle of observation which lasts up to 600s.

2 - The Pulsar Search Engine

SKA1-Mid telescopes are widely distributed and thus have a small filling factor. The maximum survey sensitivity is achieved by a coherent sum of many dishes, but this results in a narrow beam on the sky, and in a very slow survey speed. However, it is possible to speed up the survey by employing multiple beams. Each of these tied-array beams samples a different region of sky, so they are mostly independent and can be processed without communication within the PSS thus simplifying the overall design allowing us to design a modular system.

At present we aim to process in real time two beams on a GPU mounted on every node. The aim of our design is to achieve the best combination to minimizes both cost and power consumption.

Regarding the data path we are considering a large switch bank to connect the beam-former layer and the PSS layer. This would allow us to distribute the raw data between the nodes in a flexible way. Each node would be the target of two data streams and then it would forward its results to the Science Data Processor (SDP). These nodes consist of a host and a number of accelerator hardware such as FPGAs, GPUs or any other comparable device.

On completion of the processing, the pipeline will produce a list of significant events and triggers requests that will make the raw data available to the SDP for the post-processing tasks.

3- The Data Pipeline

Each processing node would operate by means of a software framework. This software coordinates the work of the different modules which perform the following stages of the data analysis pipeline. The main purpose of the framework is to ensure that the software remains modular, readable, flexible, expandable and hardware agnostic at a pipeline level.

At present, we are considering two different RT HPC Frameworks: Pelican/AMPP (Pipeline for Extensible Lightweight Imaging and CAlibratioN/Artemis Modular Pelican Pipeline), and GStreamer. Both software are already successfully used in large-scale data-intensive scientific instrument.

The main components that make up the Pulsar Search Pipeline are:

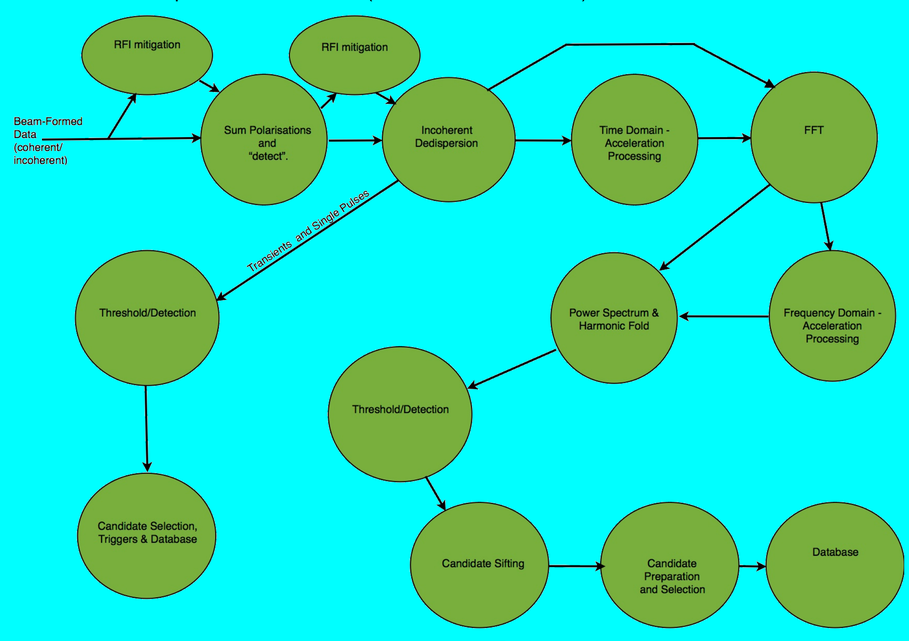
1) Data Receptor: We assume that the data coming from the beam-former arrives as a network stream and it needs to be transformed to enable pipeline processing. The task of the Data Receptor is to distribute the data to the rest of the processing elements in data chunks of a given size.

2) Radio Frequency Interference Mitigation: During the signal processing chain, we assume that some RFI mitigation will occur before and/or during the beam-forming process. However, there might be some RFI in the data which can only be perceived with the sensitivity that comes with the combination of all the dishes. The algorithm which would perform the RFI mitigation probably requires to act in both the time and frequency domain.

3) Incoherent dedispersion: As radio signals propagate through the interstellar medium, they interact with free electrons. This causes a frequency-dependent delay to the group velocity which consequently disperses the signal. In the case of a pulsed signal, such as the one from a pulsar, this means that the lower frequencies arrive later than the higher ones. If we simply sum the data over the wide bandwidths used, the pulse would be significantly smeared, reducing sensitivity. To reduce this effect, we can employ a process of dedispersion where the wide bands are divided into narrow frequency channels. The signal in each channel is corrected for the delay caused by dispersion. However, in the case of a survey the amount of dispersion (called the DM) is unknown for any given source. This forces us to make a search over a range of possible DM values. The number of dispersion measures to be processed is defined by the baseline design.

4) Acceleration processing: One of the key scientific goals of the SKA project is to find pulsars in relativistic orbits with other massive bodes, explicitly neutron stars or black holes. The relativistic aspect is important as it ensures that they are excellent test beds for theories of gravity. However, this leads to a problem in trying to detect them. The basic pulsar search model looks for strict periodicity, but an accelerating source appears to change its period. There are a number of algorithms that can partially correct this effect and this is what is meant by acceleration processing. The corrections can be made either in the time or frequency domain. The present baseline design will prescribe that 120 acceleration trials will be performed and that the correction for the acceleration will take place in the time domain. This leads to a processing requirement of almost 10PetaOps/s and this is the dominant processing step of the PSS.

5) Folding: To properly detect the source pulses, data should be integrated over time. This process (folding) can be effective only after determining a small set of possible DM and accelerations. During this stage, these parameters are refined and prepared for the selection of possible source candidates to be performed in the SDP (Science Data Processor).



The complete structure of the data pipeline

4- Future Development

The system design is predicated on a real-time processing requirement. This is derived from the very large rate of data throughput from the system. The baseline design specifies that the system works with a total PSS input data rate of 11Terabits/s and a survey integration time of 600s. Thus, to record a single typical survey observation would require about 1PetaByte of storage space, which is prohibitively large even on the timescale of the SKA
project. The PSS is consequently designed to be a real-time system which converts the enormous input data stream reduced to a just a few hundreds of megabytes of pulsar candidates data that are passed to the SDP.

As a consequence of the SKA dimension and computing requirements, technological and operational risks are still present, for instance total power consumption is still a major limitation. In the present developing phase, we are exploring the problem of mitigation in several directions.

The simplest approach is to wait for the development of more powerful GPU engines: we expect to gain roughly a factor of 2 every year. This approach has proved to work till now, and is has already (partially) been taken into account. This approach is not devoid of risks and is limited by the need to freeze the technology at the start of the construction phase.

The second direction is the exploration of more efficient algorithms. In this field the SKA TDT group has deployed the maximum effort with already good results. As this is a work in progress, extensive tests need to be performed and we have promising preliminary results.

A third direction is the development of a multiple technology environment. Mixing different parallel computation engines (such as GPUs, FPGAs and Tile-Processors) is a difficult and challenging task, but it may offer an advantage. We expect to develop specific parts exploiting the different kind of parallelism offered by each technology.

The last, and most difficult approach, is the dissolution of the boundary between the beam-former part of telescope and the correlator electronics, exploiting the possibility to reuse partial calculations in different stages. This approach poses theoretical challenges, and will be investigated outside the SKA consortium, albeit in strict contact with it.

The PSS people are affiliated with:

ASTRON

INAF ISTITUTO NAZIONALE DI ASTROFISICA NATIONAL INSTITUTE FOR ASTROPHYSICS

MANCHESTER 1824 The University of Manchester

Max Planck Institute for Radio Astronomy

NZAlliance

UNIVERSITY OF OXFORD

SWINBURNE CENTRE FOR ASTROPHYSICS AND SUPERCOMPUTING

REFERENCES

[1] Dewdney, P., "Ska1 system baseline design," in [*SKA1 Key documents*], Diamond, P., ed., SKA–TEL–SKO–DD–001, SKA Organization, Manchester, UK (2013). https://www.skatelescope.org/home/technicaldatainfo/key-documents/.
[2] Keith, M., "Ska csp ska1-mid array non-imaging processing pulsar search sub-element architecture design document," in [*SKA1 CSP Tnecnical Documents*], Carlson, B., ed., SKA–TEL.CSP.NIP.PSS–TDT–ADD–001, SKA Organization, Manchester, UK (2014).
[3] K., C., R., C., and et al., "Toward early-warning detection of gravitational waves from compact binary coalescence," *APJ* **748,2**, id136–14 (2012).
[4] D., L. and M., K., [*Handbook of Pulsar Astronomy*], Cambridge University Press, Cambridge (2005).

Address of this poster: