Automatic Mapping of Real Time Radio Astronomy Signal Processing Pipelines onto Heterogeneous Clusters

by

Terry Filiba

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor John Wawrzynek, Co-chair
Dan Werthimer, Co-chair
Professor Jan Rabaey
Associate Professor Aaron Parsons

Fall 2012

The dissertation of Terry Filiba, titled Automatic Mapping of Real Time Radio Astronomy Signal Processing Pipelines onto Heterogeneous Clusters, is approved:

Co-chair _____ Date _____

Co-chair _____ Date _____

_____ Date _____

_____ Date _____

University of California, Berkeley

Automatic Mapping of Real Time Radio Astronomy Signal Processing Pipelines onto Heterogeneous Clusters

Abstract

Automatic Mapping of Real Time Radio Astronomy Signal Processing Pipelines onto Heterogeneous Clusters

by

Terry Filiba

Doctor of Philosophy in Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor John Wawrzynek, Co-chair

Dan Werthimer, Co-chair

Traditional radio astronomy instrumentation relies on custom built designs, specialized for each science application. Traditional high performance computing (HPC) uses general purpose clusters and tools to parallelize the each algorithm across a cluster. In real time radio astronomy processing, a simple CPU/GPU cluster alone is insufficient to process the data. Instead, digitizing and initial processing of high bandwidth data received from a single antenna is often done in FPGA as it is infeasible to get the data into a single server.

I propose to develop a universal architecture where each problem is partitioned across a heterogeneous cluster, taking advantage of the strengths different technologies have to offer. I propose we take an HPC approach to instrument development with a heterogeneous cluster that has both FPGAs and traditional servers. This cluster can be reprogrammed as necessary in the same way an HPC cluster is used to run many different applications on the same hardware.

The challenge in this heterogeneous of approach is partitioning the problem. Normal HPC uses a homogeneous cluster where the nodes are interchangeable. In a heterogeneous cluster, there is an additional issue of determining how to partition the problem across different types of hardware. I propose to design a tool that automatically determines how to partition instruments for radio astronomy. In order to do this work, I will need to model the platforms and based on a description of the final instrument, generate a processing pipeline. The partitioning needs to be done using a variety of techniques to assess the hardware. A static model of the hardware is useful to determine the amount of processing available in different types of hardware. Dynamic benchmarking would also be needed to deal with varying server architectures and determine how much processing and bandwidth the cpu/gpu servers can handle. Finally, to capture any overlooked subtleties or deal with things the tools cannot handle, the user will be able to input hints as to how the instrument should be generated.

The development of this tool will be driven by 2 instruments. First, the design of the GBT spectrometer, a spectrometer designed to support many different modes using the same cluster. By using the tool to design this spectrometer, additional modes can be easily added and if the cluster is expanded the each mode can be redesigned to do additional processing that takes advantage of the extra hardware. I will also be working on the LEDA correlator. This is a low bandwidth, large N correlator, which is an ideal application for heterogeneous clusters.

We can assess the performance of this automatic partitioning tool in a number of ways. First, this tool should significantly reduce NRE and time to science. By automatically generating the instrument the need for engineers who understand both science goals and programming is removed. However, this benefit should not come with a large increase in cost. The instruments produced by this tool will be compared to optimized implementations with the same parameters on the basis of hardware utilization and power consumption.

To TODO

TODO

Contents

List of Figures

List of Tables

Acknowledgments

TODO

Chapter 1

Introduction

Chapter 2

Real Time Radio Astronomy Algorithms

Radio astronomy simply refers to the type of science that can be done by observing astronomical objects at radio wavelengths, rather than a specific scientific goal. There is a huge variety of different experiments, such as searching for gravity waves (TODO: ref nangrav), traces of the first stars (TODO: ref PAPER/LEDA), or aliens (TODO: ref SETI). But, despite this variety, the small number of algorithms detailed in this chapter serve as the first step in processing the data for many such projects.

Radio telescopes produce very high amounts of data (TODO: add estimates, how much data?). The reason for this high influx of data is twofold. First, to enable new science, new radio antenna observe increasingly higher bandwidths of data. Second, to sate the need for larger collecting area, rather than designing a large single dish, many new telescopes are being designed as antenna arrays, where the data from multiple antennas is combined to act as a single large dish. While it may be cheaper and easier to get a larger collecting area using small dishes rather than a single large dish, this adds additional complexity processing the data coming off the telescope. Rather than processing a single stream of data, now the instrument must process and combine multiple streams to make the array seem like a single large dish. As the size of the arrays and bandwidths for single dishes simultaneously increase, the data produced cannot be feasibly recorded in real-time. To cope with the progress in science and antenna technology, there is a constant need for new systems to process, rather than record, this data in real time. Each of these instruments begin processing the data immediately after it is digitized, and needs to reduce the data without loosing scientific information. Once the data is partially processed, and reduced to a manageable bandwidth, it can be stored and processed further offline.

There is a small number of real time algorithms commonly used to reduce the data. In this work, we specifically focus on spectroscopy, pulsar processing, beamforming and correlation. Spectroscopy and pulsar processing are both spectral methods of analyzing data from a single beam. They both can be used on antenna arrays but would either need to treat the array as separate dishes rather than one large dish, or the data from the dishes would need

to be combined into a single beam, which can be done using the third algorithm on the list, beamforming. The last two techniques, beamforming and correlation, are both ways of combining data from multiple antennas. Beamforming combines the data into a single beam by delaying and summing the data from each antenna. Correlation doesn't aim to form a single beam, but instead is the first step towards getting an image of the sky.

2.1 Spectroscopy

A spectrometer is simply an instrument that produces an integrated, or averaged, frequency domain spectrum from a time domain signal. A real time spectrometer works by constantly computing a spectrum over short windows of data (channelization), then each channel is summed for a predetermined amount of time to compute the average power in that channel (accumulation). Figure x shows a block diagram for a simple spectrometer design. After digitization, there is the channelization step, where the signal is processed by a digital filter bank, and then the channels are accumulated.

High resolution spectroscopy

Increasing resolution often requires an increase in complexity in the spectrometer design. Once the number of required channels is sufficiently high, it becomes infeasible to compute the spectrum using a single filter bank. To cope with this, the channelization is done in two steps as shown in figure (TODO: add hi res spectrometer block diagram). In the first step, the signal is divided into coarse channels using a filter bank. At this point the channels are much wider than intended and can't be accumulated yet. After coarse channelization, the spectrometer treats the data from a single channel as time domain data and passes it through a filter bank again. This step breaks up the wide channel into a number of smaller channels . At this point the data can be accumulated, since it has the desired resolution.

This high resolution spectroscopy technique is used in the SERENDIP V.v (Search for Extraterrestrial Radio Emissions from Nearby Developed Intelligent Populations) project. This project is part of the SETI (Search for Extraterrestrial Intelligence) effort to detect extraterrestrial intelligence. The SERENDIP V.v project is a commensal survey at the Arecibo observatory, meaning anytime the ALFA receiver is used for any observation, the SERENDIP spectrometer will also process and record data.

The project is focused on detecting strong narrow band radio signals, requiring a high resolution spectrometer installed at the observing telescope to analyze the data. The spectrometer should resolve channels of less than 2 Hz, so that natural astronomical signals that typically have a wider bandwidth can easily be distinguished from narrower, possibly extraterrestrial, signals. The SERENDIP V.v spectrometer meets this by providing 128 million channels across 200MHz of bandwidth, for an resolution of 1.5 Hz per channel. Since it would be infeasible to channelize a 200MHz into 128 million channels using a single filter

bank, the SERENDIP V.v spectrometer uses the high resolution architecture described in figure x.

2.2 Pulsar Processing

A pulsar processor is an instrument designed specifically to observe transient events, such as pulsars. A pulsar is a rotating neutron star that emits an electromagnetic beam. When the beam sweeps past Earth, due to the rotation of the star, it is observed as a wideband pulse. As the pulse travels through the interstellar medium (the matter filling interstellar space), the pulsar gets dispersed, meaning the low frequencies arrive before high frequencies, despite the fact that they were emitted at the same time.

A spectrometer, as described in Section 2.1, that accumulates the spectrum would smear the pulse, so there will need to be a few adjustments to the spectroscopy algorithm to make it suitable for processing transient events. In the case of pulsars, the algorithm starts with a high-resolution spectrometer without an accumulator. Instead, the algorithm becomes specialized to detect this type of quickly occurring event.

The high resolution data is then sent to a process called dedispersion, which undoes the dispersion caused by the ISM, realigning the pulse. There are 2 techniques to do this. First, the pulse can be dedispersed by shifting the frequency channels by different amounts to compensate for the different delays as shown in figure x, in a process called *incoherent dedispersion*. This process can't be used to reconstruct the original pulse, but due to its relatively low compute cost, is a useful algorithm to search for new pulsars. The second technique, *coherent dedispersion*, models the effect of the ISM as a convolving filter. To remove this effect, the signal is deconvolved with the model. This is more compute intensive than incoherent dedispersion, but can recover the original pulse.

After dedispersion, there is a still a lot of data and the pulse has very low SNR. The next step in processing is *folding*, or adds together many pulses, reducing the signal and improving the SNR. At this point, the data has been significantly reduced and can be recorded.

2.3 Beamforming

Beamforming is one technique for combining data from an array of antennas. The beamformer combines the data from multiple antennas into a single beam pointed at a single point in the sky. This is achieved by delaying the signal from each antenna by a different amount and then summing the delayed signals. As illustrated in figure x, the signal from the intended source will not arrive at all the antennas at the same time. The delay compensates for the disparity between the arrival times, and one the signals are summed it creates constructive interference in the direction of the source, and destructive interference in other directions. These delays can be changed to point the beam at a different source or to track a single source moving across the sky.

BF Beamforming

Since the signal is discrete, and the amount of delay might not be an integer multiple of the sample period s, the delay d is applied to the signal, $f[n]$, in 2 steps. The delay in clock cycles is represented as d/s and broken up into it's integer and fractional parts. First, the integer part is applied as a coarse delay, $n_f = \lfloor d/s \rfloor$. This can simply be implemented by buffering the signal. Applying the fractional delay $p \bmod s$ is more complex. Since there is no observed data at that time, the signal fractional samples are calculated by convolving the signal with an interpolation filter b. Applying these 2 steps, results in a new delayed signal $f[n] = f[n - n_f] * b_{fi}$

In practice, it is common to calculate the spectrum of the beamformed signal. A typical 2 element beamformer, with discrete input signals $f[n]$ and $g[n]$ is trying to calculate the spectrum of beam i, H_i, using coarse delays n_{fi} n_{gi} and delay filters b_{fi} and b_{gi}, as follows:

$$H_i(x) = FFT(f[n - n_{fi}] * b_{fi} + g[n - n_{gi}] * b_{gi})$$

We describe this as BF beamforming because the delay operation (B), happens before the FFT operation (F).

FB Beamforming

In the case where many beams are required, each beam needs a different FIR filter for every antenna, and a separate FFT. This is called FB beamforming because the FFT operations (F) occur before the delay (B). Using linearity of the fourier transform and the convolution theorem, the computation can be rearranged as follows:

$$H_i(x) = F[x] \cdot B'_{fi} + G[x] \cdot B'_{gi}$$

Where $F[x]$ and $G[x]$ are the fourier transforms of the signals f and g. The signal delay becomes a phase shift in frequency domain that is represented by B'_{fi} and B'_{gi}. In this case, there is an FFT for each antenna, rather than one FFT per beam. When forming multiple beams, there is no need to recalculate the Fourier transform of the signals. So, as the number of beams increases, it can be advantageous to use this algorithm rather than the BF algorithm described in the previous section.

2.4 Correlation

Aperture synthesis is another technique to combine that data from many antennas. The goal is to form an image of the sky, using 2 steps, correlation followed by imaging. In the correlation step, the cross-correlation of each pair of antennas is calculated. The cross-correlation of an antenna pair represents a point in the uv plane, called a *visibility*.

The uv plane represents the Fourier transform of the 2-dimensional sky image.

XF Correlation

The cross-correlation of two signals, f[n] and g[n] is defined as:

$$f[n] \star g[n] = \sum_{i=0}^{m} f^*[i]g[n+i]$$

Like in beamforming, it is often desirable to calculate a spectrum, so the correlation step typically also calculates the spectrum of the cross-correlations.

FX Correlation

Chapter 3

Past Work

3.1 Digital Signal Processing for Radio Astronomy

3.2 Automatic Mapping

Chapter 4

High Level Toolflow

Chapter 5

Algorithm Partitioning

Chapter 6

Analysis/Results

Chapter 7

Conclusions