VLBA Sensitivity Upgrade MEMO 17

Benchmarking DiFX on the Mark5A Cluster and Justification for an Upgrade

Walter Brisken (National Radio Astronomy Observatory)

November 26, 2007

Abstract

This memo describes a series of DiFX benchmarks performed on the cluster of Mark5A computers. The results are used to plan for an upgrade that will allow DiFX to run at a mean rate of about 256 Mbps (for 2-bit samples) on 10 stations.

1 The Mark5A Cluster

Currently software (SW) correlator testing has been taking place using the CPU power present inside the Mark5A units attached to the hardware (HW) VLBA correlator. Ten of these Mark5 units contain a single 3 or 3.2 GHz[1] and eight have two dual-core 2 GHz CPUs. This configuration has been sufficient for development and coarse benchmarking. Many of the possible VLBA format modes have been confirmed to properly correlate. Although the SW correlator still has some problems, there are no expected showstoppers and it is time to expand the compute capacity in anticipation of increased testing, further benchmarking, and possibly production, needs.

2 Mark5 cluster benchmarking

A series of benchmarks was run on the above mentioned Mark5A cluster. Though these benchmarks cover a large range of observing modes they are not meant to be exhaustive. The intent is to gather enough information on regularly used observing and correlator modes to make an informed decision on the immediate upgrade path. The configuration of these tests was as follows:

1. **Manager**: The manager node was 'parallax', a desktop computer on the AOC network. There is no evidence that having the manager node outside the immediate network of Mark5 units is a problem, especially at the $\sim$ 100 Mbps rates beind tested, but in the final configuration a dedicated master node with plenty of storage for staging results to be archived should be attached to the same network switch as the compute nodes.
2. **Datastreams**: In these tests, 4 or 9 of the single CPU Mark5 units were used as datastream nodes. Their function is to read requested segments of data from Mark5 modules and send them to the proper computing processes.
3. **Calculations**: Four computing processes were run on each of the Mark5s with 4 CPU cores. This sums to 64 GHz of CPU clock across the processes. These computing processes unpack data, perform station-based Fourier transforms, and perform the cross multiplies and short term accumulation.
4. **Timing**: A correlation run time was measured for each job run. The reported time includes data read and transfer, cross correlation, and writing of the results to disk. It does not include: model creation, spawning of the correlation processes on all the nodes, and FITS conversion of the results. The first and last of these can be done in parallel during correlation of other projects (and represent a very small fraction of the total processing) and the spawning of processes usually does not take much time, however streamlining this will become a priority in the near future.

Timing results for two classes of test jobs are shown in Tables 1 and 2. The first shows benchmarking for 1-bit per sample modes and the second for 2-bits per sample modes. For each test, a run time is reported. Also shown is the implied 10 station bit-rate, calculated assuming linear scaling with number of stations.

It is clear that the format of the data has an impact on the performance. Jobs with fewer IFs tend to run faster. Columns labeled "trial 1" and "trial 2" demonstrate that repeated runs of the same job

[1] One of these ten units has two such processors.

Project Job	obs time (s)	IF bw (MHz)	rate (Mbps)	nIF	nchan (per IF)	nPol	nAnt	run time (s)	10 stn rate (Mbps)	nAnt	run time (s) trial 1	run time (s) trial 2	10 stn rate (Mbps)
TM017													
4322,3	118	8	256	16	256	1	4	165.9	72.8	9	704.2	705.6	38.6
4325	119	16	256	8	256	1	4	144.6	84.3	9	456.5	460.2	59.8
4329,30	240	4	128	16	256	1	4	178.1	69.0	9	749.6		36.9
4332	239	8	128	8	256	1	4	153.7	79.6	9	488.4		56.4
4336,7	480	2	64	16	256	1	4	188.5	65.2	9	783.1		35.3
4339	480	4	64	8	256	1	4	160.8	76.4	9	515.9		53.6
4420	419	16	128	4	256	1	4	197.8	108.4	9	511.9		94.3
4422	238	8	64	4	256	1	4	75.3	81.0	9	201.8		67.9
4424,5	480	1	32	16	256	1	4	93.5	65.7	9	412.4		33.5
4427	480	2	32	8	256	1				9	265.6		52.0
4429	478	4	32	4	256	1				9	209.8		65.6
MT741													
520-7	359	4	128	8	128	2				9	626.2		66.0
521	358	8	128	4	128	2				9	500.9		82.3
								Average	78.1			*Average*	57.1

Table 1: *1-bit per sample benchmarks.*

result in nearly identical execution times suggesting that this format dependence is real and not just due to variations in job run times.

Some tests were run for both 4 and 9 stations[2]. It is clear that run time is not strictly linear in number of antennas. For the 1 and 2 bit cases respectively, the following scaling laws for timing as a function of number of antennas to be correlated are determined:

$$T_{1-\mathrm{bit}} \propto N + 0.154N^2 \quad (1)$$

$$T_{2-\mathrm{bit}} \propto N + 0.077N^2 \quad (2)$$

It should be noted that these laws were determined using an average of the tests done here and that the results will likely depend on other factors, such as number of polarization products and number of spectral channels. The breakdown into a linear and quadratic term assumes there are no constant-time portions of the algorithm. Such a scaling law determined with only two points should be considered very approximate, at best. See VLBA Sensitivity Upgrade MEMO 16 for a first-principles analysis of such scaling laws.

The factor of ~ 2 worse performance for the 1-bit case is understandable. Once unpacking of the data is complete (a small fraction of the total compute time), DiFX treats 1 and 2 bit data in the same way, so it is the total number of samples, not the number of recorded bits, that determines the run time. This is in contrast to the HW correlator, for which changing the number of bits per sample for a given recorded bit rate also changes the utilization of correlator resources. It is somewhat puzzling why the quadratic term in the 1-bit case is exactly twice that of the 2-bit case. Further benchmarks may shed light on this paltry mathematical dilemma.

[2]It would be more natural to run on 10 stations, but recent test jobs have typically 9 stations due to Saint Croix being unavailable.

Project Job	obs time (s)	IF bw (MHz)	rate (Mbps)	nIF	nchan (per IF)	nPol	nAnt	run time (s)	10 stn rate (Mbps)	nAnt	run time (s) trial 1	run time (s) trial 2	10 stn rate (Mbps)
TM017													
4320,1	419	4	256	16	256	1	4	277.2	154.8	9	1004.8	1015.9	95.5
4324	118	8	256	8	256	1	4	82.8	145.9	9	248.9	249.7	109.1
4326	119	16	256	4	256	1	4	82.6	147.5	9	205.2	206.1	133.3
4327,8	239	2	128	16	256	1	4	112.1	109.1	9	402.4	401.1	68.5
4331	239	4	128	8	256	1	4	92.7	131.9	9	264.4		104.1
4333	240	8	128	4	256	1	4	88.3	139.2	9	217.9		126.9
4334,5	481	1	64	16	256	1	4	130.2	94.6	9	424.5		65.3
4338	480	2	64	8	256	1	4	97.6	126.0	9	294.5		93.9
4340	480	4	64	4	256	1	4	88.4	139.1	9	243.4		113.6
4421	118	16	128	2	256	1	4	39.1	154.6	9	100.0		136.0
4423	240	8	64	2	256	1	4	41.2	149.3	9	130.1		106.2
4426	480	1	32	8	256	1				9	191.3		72.3
4428	478	2	32	4	256	1				9	147.2		93.5
MT741													
522-1	359	4	128	8	128	2				9	328.2		126.0
523,4	359	16	512	8	128	2				9	1216.4		136.0
								Average	135.6			*Average*	105.3

Table 2: *2-bits per sample benchmarks.*

3 Cluster upgrade

A reasonable goal is to support 10 stations at 256 Mbps (assuming 2-bit sampling). Such a computer cluster would be able to correlate a reduced number of antennas at an increased rate, or vice versa. This goal represents a factor of nearly three increase in compute power over the current Mark5A cluster.

In recent years it has become possible to cram enormous computing power into small boxes. A particular cost-effective example of this is the *twin server* 1-U rack-mount box containing up to 4 CPUs. The particular CPUs being targetted are Intel's new 45nm process quad-core XEON processors. The smaller gate sizes allow more on-chip cache memory and lower total power consumption. At the same clock speed, these CPUs are both faster and cheaper than their predecessors. Several companies offer these CPUs in 1-U boxes. For concreteness, I will assume that the *quantum* of CPU upgrade will be a *twin server* from `siliconmechanics.com` containing the following:

- 4× quad-core 2.5 GHz CPUs (Intel part E5420)
- 8× 1 GB RAM
- 4× gigabit network interfaces
- 2× 160 GB hard disks

The above choice is based on several factors including the performance/price ratio, the performace/power consumption ratio, and maintainability; comparable systems from other vendors are equally attractive. Each such box consumes 515W of power, costs \$4107[3], and has an aggregate CPU clock rate of $4\times4\times2.5 = 40$ GHz. It is somewhat difficult to determine exactly how many CPUs will be required to reach the goal

[3]It is likely an educational discount of $\sim 5\%$ would apply to our purchases.

of 256 Mbps. While the new CPUs have larger cache memories and reduced instruction latencies, their higher speeds combined with 4 cores per CPU (compared to 2) will further stress the memory bandwidth. Based purely on CPU frequency scaling arguments, 4 such units, in combination with the upgraded Mark5 units, should be sufficient to meet the 256 Mbps goal. Because of the mentioned uncertainties, one extra unit (a 25% contingency) will be added to the purchase.

A new gigabit switch will be required to handle the larger number of computers on the SW correlator network. A 32 port switch would meet the requirements, but would leave just 2 ports for future expansion. Depending on the price and recommendations of the computer division, a 32 or 48 port switch will be selected. A 48 port switch should be available for $\sim$ \$2000.

Finally, an upgraded Intel Performance Primitives software library (version 5.3 or later[4]) should be purchased that will allow the new features of these new CPUs to be more fully exploited. The estimated cost of this is $<$ \$200.

The total upgrade cost should be under \$30,000. If the processing speed goals are met, these new CPUs, in combination with the CPU power in the Mark5 units, will be sufficient to replace the HW correlator.

4 Longer term goals

The path to 4 Gbps will bring significant change to SW correlator hardware. The computer network will have to be upgraded to reach the required data transport requirements. The Mark5 units providing data to the processing nodes will need 10 Gbps network interfaces. CPUs will continue to become faster and will allow increased compute density. Graphics Processing Units (GPUs) found on modern high-end gaming PCs already offer compute capabilities for certain applications that exceed CPU speeds by between 1 and 2.5 orders of magnitude. The use of these for cross correlation is being explored by the MWA project in Australia and the KAT project in South Africa, and possibly others. Depending on the CPU or GPU technology used, the new data processing nodes may also require 10 Gbps interfaces. Currently 10 Gbps networking hardware, especially many-port switches, is very expensive, but its price is dropping very quickly; by the time it is needed it should be affordable.

Although these longer term goals paint a very different picture of the possible 4 Gbps cluster, the purchases requested here represent an important intermediate step. The processing power being purchased for this upgrade, as well as that for the upgrades to the Mark5 units, will remain useful until advances in CPU/GPU technology render them an insignificant contribution to the SW correlator cluster.

[4]See `http://www.intel.com/cd/software/products/asmo-na/eng/302910.htm`