PUBLICATIONS OF THE ASTRONOMICAL SOCIETY OF THE PACIFIC, **119**: 318–336, 2007 March

DiFX: A Software Correlator for Very Long Baseline Interferometry Using Multiprocessor Computing Environments

A. T. DELLER,[1] S. J. TINGAY, M. BAILES, AND C. WEST[2]

Centre for Astrophysics and Supercomputing, Swinburne University of Technology, Hawthorn, Victoria, Australia; adeller@astro.swin.edu.au

Received 2006 December 13; accepted 2007 February 6; published 2007 February 23

ABSTRACT. We describe the development of an FX-style correlator for very long baseline interferometry (VLBI), implemented in software and intended to run in multiprocessor computing environments, such as large clusters of commodity machines (Beowulf clusters) or computers specifically designed for high-performance computing, such as multiprocessor shared-memory machines. We outline the scientific and practical benefits for VLBI correlation, these chiefly being due to the inherent flexibility of software and the fact that the highly parallel and scalable nature of the correlation task is well suited to a multiprocessor computing environment. We suggest scientific applications where such an approach to VLBI correlation is most suited and will give the best returns. We report detailed results from the Distributed FX (DiFX) software correlator running on the Swinburne supercomputer (a Beowulf cluster of ~300 commodity processors), including measures of the performance of the system. For example, to correlate all Stokes products for a 10 antenna array with an aggregate bandwidth of 64 MHz per station, and using typical time and frequency resolution, currently requires an order of 100 desktop-class compute nodes. Due to the effect of Moore's law on commodity computing performance, the total number and cost of compute nodes required to meet a given correlation task continues to decrease rapidly with time. We show detailed comparisons between DiFX and two existing hardware-based correlators: the Australian Long Baseline Array S2 correlator and the NRAO Very Long Baseline Array correlator. In both cases, excellent agreement was found between the correlators. Finally, we describe plans for the future operation of DiFX on the Swinburne supercomputer for both astrophysical and geodetic science.

1. INTRODUCTION

The technique of very long baseline interferometry (VLBI) as a means to study the very high angular resolution structure of celestial radio sources was developed in the 1960s (Clark et al. 1967; Moran et al. 1967). Some accounts of the early developments in VLBI, the scientific motivations for the developments, and technical overviews are given in Finley & Goss (2000).

VLBI, as with all interferometry at radio wavelengths, hinges on the ability to obtain a digital representation of the electric field variations at a number of spatially separated locations (radio telescopes), accurately time-tagged and tied to a frequency standard. The digitized data are transported to a single location for processing (a correlator) and are coherently combined in order to derive information about the high angular resolution structure of the target sources of radio emission. The instantaneous angular resolution R of a VLBI array, in arcseconds, is given by $R = 2.52 \times 10^5(\lambda/D)$, where λ is wavelength of the radiation being observed (typically centimeters) and D is the maximum projected baseline (the distance between radio telescopes in the array projected onto a plane perpendicular to the source; typically thousands of kilometers). This yields typical angular resolutions on an order of milliarcseconds.

Traditionally, the "baseband" data (filtered, down-converted, sampled, and quantized electric field strength measurements; Thompson et al. 1994) generated at each radio telescope have been recorded to magnetic tape media; for example, the Mark I system (Bare et al. 1967), the Mark II system (Clark 1973), the Mark III system (Rogers et al. 1983), the Mark IV system (Whitney 1993), and the S2 system (Wietfeldt et al. 1996). After observation, the tapes from each telescope were shipped to a purpose-built and dedicated digital signal processor, the correlator. A correlator aligns the recorded data streams, corrects for various geometrical and instrumental effects, and coherently combines the data from the different independent pairs of radio telescopes. The correlator output streams, known as the visibilities, are related to the sky brightness distribution of the radio source, essentially via a Fourier transform relation (Thompson et al. 1994).

The two fundamental operations required to combine or cor-

[1] Cosupervised through the Australia Telescope National Facility, Epping, NSW, Australia.

[2] Current address: Department of Astronomy, University of Massachusetts at Amherst, Amherst, MA.

relate the recorded signals are a Fourier transform (F) and a cross multiplication (X). The order of these operations can be interchanged to obtain the same result, leading to the so-called XF and FX correlator architectures. A number of well-known descriptions of the theory and practice of radio interferometry describe the technique in varying degrees of detail and elaborate on the differences between XF and FX correlators (Thompson et al. 1994; Romney 1999), and the reader is referred to these texts for the details.

Both XF- and FX-style correlators have traditionally been highly application-specific devices, based on purpose-built integrated circuits. In the last 20 years, field-programmable gate arrays (FPGAs) have become popular in correlator designs, with one prominent example being the Very Long Baseline Array (VLBA) correlator (Napier et al. 1994). FPGAs are reconfigurable or reprogrammable devices that offer more flexibility than application-specific integrated circuits (ASICs) while still being highly efficient.

This paper deals with a departure from the traditional approach of tape-based data recording and correlation on a purpose-built processor (based on either ASICs or FPGAs). We have developed a correlator that is based on software known as DiFX (Distributed FX), which runs within a generic multiprocessor computing environment. Such a correlator interfaces naturally to modern hard disk data-recording systems, such as the Mark 5 system (Whitney 2002) and the K5 system (Kondo et al. 2003), which have now largely replaced tape-based recording systems. Specifically, we have developed this software correlator to support a new disk-based VLBI recording system that has been deployed across the Australian Long Baseline Array[3] (LBA) for VLBI. We refer the reader to a detailed discussion of the LBA hard-disk recording system (LBADR) that appears elsewhere (C. Phillips et al. 2007, in preparation). As our software correlator is more broadly applicable than to just the LBA, we do not dwell on the details of the LBA recording system in this paper, but rather concentrate on the characteristics, benefits, and performance of our software correlator, giving brief details of the recording system when required. The correlator source code, binaries, and instructions for use are available for download online.[4]

The very first VLBI observations were in fact correlated using software on a mainframe computer. Software correlators were developed simultaneously on an IBM 360/50 at the National Radio Astronomy Observatory (NRAO; Bare et al. 1967) and on an IBM 360/92 at the Goddard Space Flight Center (Moran et al. 1967). As the early experiments quickly increased in complexity, the recorded data volume also increased, and it became necessary to design custom hardware for VLBI correlation. Recent examples of such correlators include the NRAO Very Long Baseline Array correlator (Napier et al. 1994), the Joint Institute for VLBI in Europe (JIVE) correlator (Casse 1999), the Canadian NRC S2 correlator (Carlson et al. 1999), the Japanese VLBI Space Observatory Program (VSOP) correlator (Horiuchi et al. 2000), and the Australia Telescope National Facility (ATNF) S2 correlator (Wilson et al. 1996). Table 1 compares some of the basic properties of some currently operational hardware VLBI correlators.

Recently, the pace of development of commodity computing equipment (processors, storage, networking, etc.) has outstripped increases in VLBI computational requirements to the point that the correlation of VLBI data using relatively inexpensive supercomputer facilities is feasible. The correlation algorithm is "embarrassingly parallel" and very well suited to such parallel computing architectures. These facilities are not purpose-built for correlation, but are inherently multipurpose machines suited to a wide range of computational problems.

This approach to correlation gives rise to significant scientific benefits under certain circumstances. The benefits stem from the basic characteristics of correlation, software engineering considerations, and the computing environments. Software is more flexible and easier to redesign than application-specific hardware or even FPGA-based processors (although the programming tools for FPGAs are developing rapidly). The highly parallel nature of the correlation problem, coupled with the availability of high-level programming languages and optimized vector libraries, means that a reasonably general software correlator code can be written quickly and be used in a variety of different computing environments with minimum modification, or in a dynamic environment where computing resources and/or significant scientific requirements can change rapidly with time.

However, the trade-off for flexibility and the convenience of high-level programming tools is a reduced efficiency for any given task compared to an application-specific or FPGA-based solution. Put simply, the nonrecoverable engineering (NRE) costs for a software correlator are much lower than for a hardware correlator, but the cost per unit processing power is higher. Thus, the limited computation needed by a small-size correlator means a software approach will be cheaper overall, while the tremendous computational requirements of correlators on the scale required for the Expanded Very Large Array (EVLA) or Atacama Large Millimeter Array (ALMA) dictate that the substantial amounts of NRE spent optimizing hardware are worthwhile, at least in 2006.

Software also has an advantage over hardware if the additional support required for unusual or stringent VLBI experiments is impossible or impractical to implement in an existing hardware correlator. An example of this is given in § 4.3. Use of a software correlator in these cases, even at possibly reduced efficiency, is preferable to the expense of building or altering dedicated hardware.

A good example of the flexibility of software correlation and its trade-off with efficiency is spectral resolution capability. A generic modern CPU is capable of calculating multimillion-point one-dimensional fast Fourier transforms (FFTs), allowing

[3] See http://www.atnf.csiro.au/vlbi.

[4] See http://astronomy.swin.edu.au/~adeller/software/difx.

TABLE 1
COMPARISON OF EXISTING HARDWARE CORRELATOR PARAMETERS

Correlator	Type	Maximum Telescopes (in one correlator pass)	Maximum Channels (per baseline)	Minimum Integration Time (ms)	Maximum Input Data Rate (Mbps)	Maximum Output Data Rate (Mbyte s^{-1})	Pulsar Binning
VLBA[a]	FX	20	2048	131.072	256	1	Yes
JIVE [b]	XF	16	2048[c]	125[d]	1024	6[e]	No
ATNF S2[f]	XF	6	8192[g]	2000	128	0.064	Yes

[a] See http://www.vlba.nrao.edu/astro/obstatus/current/node28.html.
[b] See http://www.jive.nl/correlator/status.html.
[c] For up to eight telescopes.
[d] When using half the correlator.
[e] Data in lag space.
[f] See http://www.atnf.csiro.au/vlbi/correlator.
[g] For 0.5 MHz bandwidth, two products.

an FX-style software correlator using this CPU as a processing element to give extremely high frequency resolution: a million spectral points across the frequency bandwidth of an observation.

Such a correlation would be computationally intensive, as conventional CPUs are not optimized for such operations. However, it could be carried out using exactly the same software and hardware as is used for a generic continuum experiment. Comparison to Table 1 shows that such high spectral resolution is currently impossible on existing hardware correlators. A number of limitations on particular hardware correlator implementations, such as minimum integration times, maximum input data rates, and maximum output data rates, can be overcome in a similar fashion with software correlators.

The flexibility, inexpensive nature, and ease of production of software correlators makes them particularly useful for small- to medium-sized VLBI arrays, since development times are short, costs are low, and the capabilities are high, providing niche roles for even small facilities. These factors have led to a resurgence in software correlator applications in a number of groups around the world. In addition to the efforts described here at the Swinburne University of Technology, one group has developed a software correlator, mainly for geodetic VLBI, at the Communications Research Laboratory (CRL) in Japan (Kondo et al. 2003). This CRL code is also used for real-time fringe checks during observations on the European VLBI Network (EVN), operated from JIVE.[5] Also at JIVE, a software correlator has been developed and used to process VLBI observations that tracked the *Huygens* probe as it entered the atmosphere of Titan (Pogrebenko et al. 2003). Spacecraft-tracking with VLBI and software correlation is likely to become a more recognized technique following the *Huygens* success, for example for the Chinese *Chang'e* lunar mission.[6] Finally, the most ambitious example of a software correlator is the Low Frequency Array (LOFAR) correlator, which is implemented on an IBM BlueGene/L supercomputer containing 12,000 processors.[7] This software correlator rivals the most powerful hardware correlators currently operating or in design, but it differs from the software correlator described in this paper in that hardware-specific optimizations and large amounts of NRE were utilized.

The approach we used in the development of the software correlator was largely inspired by the previous success of a group at Swinburne who developed baseband signal-processing software for multiprocessor environments, for the purposes of pulsar studies (Bailes 2003). A prototype software correlator developed at Swinburne is described in West (2004), with initial results described in Horiuchi et al. (2007).

In this paper, we concentrate on a description of the DiFX software correlator for VLBI developed at the Swinburne University of Technology, motivated by the factors discussed above. This correlator has been used as part of the ATNF VLBI operations since 2005 and has now replaced the previously used ATNF S2 correlator. The particular architecture we have adopted (§§ 2.1, 2.2, and 2.3) is discussed only briefly, as the correlation algorithm has been discussed at length in the literature. Section 3 describes the DiFX correlator, including the details of the software implementation, verification results from comparisons with two established hardware correlators, and performance figures of merit. In § 4, we illustrate some examples of specific scientific applications that can benefit from software correlation. Finally, our conclusions are presented in § 5.

2. THE FX SOFTWARE CORRELATOR ARCHITECTURE

Many previous works develop in detail the theory of radio interferometry (Thompson et al. 1994; Thompson 1999). The reader is referred to these texts for a complete discussion of

[5] Details about the process and results can be found at http://www.evlbi.org/evlbi/tevlb8/tevlb8.html.
[6] See http://en.cast.cn.
[7] See http://www.lofar.org.

the technique. Here we discuss the main steps used to implement the correlator architecture (FX) that we have adopted.

A more extensive overview of correlator operations is given in Romney (1999). We do not describe the operations at the telescopes that convert the incident electric field at sky frequency to the filtered, down-converted, sampled, and digitized data streams that are recorded to disk ("baseband data" in our terminology).

A number of the initial operations are made on the telescope-based data streams, while a number of the later operations are baseline-based. These two sets of operations are briefly described separately and in sequence.

2.1. Antenna-based Operations

2.1.1. Alignment of Telescope Data Streams

To correlate data from a number of different telescopes, the changing delays between those telescopes must be calculated and used to align the recorded data streams at a predetermined point in space (in this case the geocenter) throughout the experiment.

The Swinburne software correlator uses CALC 9[8] to generate a geometric delay model $\tau(t)$ for each telescope in a given observation, at regular intervals (usually 1 s). CALC models many geometric effects, including precession, nutation, and ocean and atmospheric loading, and is used by many VLBI correlators, including the VLBA and JIVE correlators. These delays are then interpolated (using a quadratic approximation) to produce accurate delays ($\Delta\tau < 1 \times 10^{-15}$ s, compared to an exact CALC value) in double precision for any time during the course of the observation. The estimated station clock offsets and rates are added to the CALC-generated geometric delays.

The baseband data for each telescope are loaded into large buffers in memory, and the interpolated delay model is used to calculate the accurate delay between each telescope and the center of the Earth at any given time during the experiment. This delay, rounded to the nearest sample, is the integer-sample delay. The difference between the delay and the integer-sample delay is recorded as the antenna-based fractional sample delay (up to ± 0.5 sample). Note that the alignment of any two data streams (as opposed to a data stream alignment with the geocenter) is good to ± 1 sample.

The integer-sample delay is used to offset the data pointer in memory and select the data to be correlated (some number of samples that is a power of 2, starting from the time of alignment). The fractional sample error is retained to correct the phase as a function of frequency following alignment to within one sample, fringe rotation, and channelization (§ 2.1.3).

Once the baseband data for each telescope have been selected, they are transferred to a processing node and unpacked from the coarsely quantized representation (usually a 2 bit representation) to a floating point (single precision) representation. From this point on, all operations in the correlator are performed using floating point arithmetic in single precision, unless otherwise specified. Note that the data volume is expanded by a factor of 16 at this point. The choice of single-precision floats (roughly double the precision necessary) was dictated by the capabilities of modern CPUs, which process floats efficiently. Using sufficient precision also avoids the small decorrelation losses incurred by optimized, low-precision operations often used in hardware correlators. This is a good example of the sacrifice of efficiency for simplicity and accuracy with a software correlator.

At this point, all data streams from all telescopes are aligned to within ± 1 sample of each other, and the fractional sample errors for each of the telescope data streams are recorded for later use. A set number of samples from each telescope data stream have been selected and are awaiting processing on a common processing node (e.g., a PC in a Beowulf cluster).

2.1.2. Fringe Rotation

Fringe rotation compensates for the changing phase difference introduced by delaying the signal from each telescope to the geocenter after it has been downconverted to baseband frequencies. If the changing delay $\tau(t)$ could be compensated for at sky frequency, fringe rotation would not be required. This, however, is impractical.

The necessary fringe rotation function can be calculated at any point in time by taking the sine and cosine of the geocentric delay multiplied by the sky frequency ν_0; it is applied via a complex multiplication for each telescope's data stream.

Since the baseband data have already been unpacked to a floating point representation by this stage, a floating point fringe rotation is applied that yields no fringe rotation losses, compared, for example, to a 6.25% loss of signal-to-noise ratio (S/N) for three-level digital fringe rotation in a two-level complex correlator (Roberts 1997).

Implemented as such, fringe rotation represents a mixing operation and will result in a phase difference term that is quasi-stationary at zero phase (the desired term) and a phase sum term that has a phase rate of twice the fringe rotation function, $\sim 4\pi\nu_0\tau(t)$. The sum term vector averages to a (normally) negligible contribution to the correlator; for typical VLBI fringe rates (100s of kHz) and integration times (seconds), the relative magnitude of the unwanted contribution to each visibility point is $<10^{-5}$. In a software correlator, it would be simple to control the integration time so that the rapidly varying phase term is integrated over exactly an integral number of terms of phase, thus making no contribution to the correlator output. This feature is not currently implemented in DiFX.

We have thus far described fringe rotation as a phase shift for each sample in the time domain. If performed in this manner,

[8] See http://gemini.gsfc.nasa.gov/solve.

TABLE 2
MAXIMUM DECORRELATION INCURRED DUE TO "POST-F" FRINGE ROTATION

Observation	Maximum Baseline (km)	Frequency (MHz)	Number of Channels (16 MHz band)	Maximum Decorrelation (%)
LBA low-frequency continuum	1400	1600	128	0.003
LBA high-frequency continuum	1700	8400	128	0.13
VLBA low-frequency continuum	8600	1600	128	0.12
VLBA high-frequency continuum	8600	22,200	128	21.1
LBA water masers	1700	22,200	1024	47.6

we refer to the fringe rotation as "pre-F" (under an FX architecture), as it has been applied before the transformation to the frequency domain in the channelization process (§ 2.1.3). In this case, the geometric delay for each sample is interpolated using the delay model as described in § 2.1.1 above.

In cases where the fringe rotation to be applied changes little from the first sample in the FFT window to the last, a minimal amount of decorrelation is introduced by applying a single fringe rotation for the entire window. The decorrelation can be estimated by sinc $(\Delta\phi/2)$, where $\Delta\phi = 2\pi\nu_0\Delta\tau$ is the change in baseline phase due to Earth's rotation over the FFT window.

In this way, fringe rotation can be applied after channelization, which saves considerable computational effort ("post-F" fringe rotation). For this approach to be viable, the fringe rates should be low (i.e., low frequencies and/or short baselines), and the number of channels should be small (implying that the time range of the samples to be correlated is short compared to the fringe period). Table 2 shows the degree of decorrelation that would be incurred by utilizing post-F fringe rotation for a range of VLBI observation modes. This decorrelation is simple to calculate and could be used to correct the visibility amplitudes and alter visibility weights, although this is not presently implemented in DiFX. It is important to note that the use of post-F fringe rotation is not recommended for all situations shown in Table 2, and indeed is only intended for use when the resulting decorrelation is $\ll 1\%$.

Post-F fringe rotation is desirable in situations where the fringe rate is extremely low, when the double-frequency term introduced by the mixing operation of pre-F fringe rotation is not effectively averaged to zero over the course of an integration and makes a significant and undesirable contribution to the correlator output. Switching from pre-F to post-F fringe rotation would be beneficial for periods of time in most experiments when the source traverses periods of low phase rate. Sources near a celestial pole can have very low fringe rates for long periods of time. Alternatively, if very short correlator integration times are used, the sum term may not integrate to zero when using pre-F fringe rotation. Post-F fringe rotation would therefore be a natural choice in these circumstances.

It should be noted that it is possible to undertake the exact equivalent to pre-F fringe rotation in the frequency domain. However, this would involve the Fourier transform of the fringe rotation function and a convolution in the frequency domain, which is at least as computationally intensive as the complex multiplication of the data and fringe rotation in the time domain.

DiFX implements pre-F or post-F fringe rotation as a user-controlled option.

2.1.3. Channelization and Fractional Sample Error Correction

Once the data are aligned and phase corrected after fringe rotation, the time-series data are converted into frequency-series data (channelized) prior to cross multiplication.

Channelization of the data can be accomplished using a fast Fourier transform or a digital filterbank. If used, the filterbank is implemented in a polyphase fashion, which essentially inserts a decomposed filter before an FFT (Bellanger & Daguet 2004). This allows the channel response to be changed from the sinc^2 response that is natural to an FX correlator to any desired function. In practice, an approximation to a rectangle is applied, although the length of the filter (and hence the accuracy of the approximation) is tunable.

If pre-F fringe rotation has been applied, the data are already in complex form, and so a complex-to-complex FFT is used. The positive or negative frequencies are selected in the case of upper or lower sideband data, respectively. If post-F fringe rotation is to be applied, the data are still real, and so a more efficient real-to-complex FFT may be used. This is possible due to the conjugate symmetry property of an FFT of a real data series. In this case, lower sideband data can be recovered by reversing and conjugating the resulting channels.

The final station-based operation is fractional-sample correction (Romney 1999). This step is considerably easier in an FX correlator than an XF implementation, since the conversion to the frequency domain before correlation allows the fractional error to be corrected exactly, assuming the error to be constant over an FFT length. This is equivalent to the assumption made for post-F fringe rotation, but is considerably less stringent, since the phase change is proportional to the subband bandwidth, rather than sky frequency, as in the case of fringe rotation. The frequency domain correction manifests itself as a slope in the phase as a function of frequency across the observed bandwidth.

Thus, after channelization, a further complex multiplication is applied to the channels, correcting the fractional sample error.

In the case of post-F fringe rotation, the fringe rotation value is added to the fractional-sample correction, and the two steps are performed together.

Either simple FFT or digital polyphase filter bank channelization can be selected as a user-controlled option in DiFX.

2.2. Baseline-based Operations

2.2.1. Cross Multiplication of Telescope Data Streams

For each baseline, the channelized data from the telescope pair are cross multiplied on a channel-by-channel basis (after forming the complex conjugate for the channelized data from one telescope) to yield the frequency domain complex visibilities that are the fundamental observables of an interferometer. This is repeated for each common band/polarization on a baseline, and for all baselines. If dual polarizations have been recorded for any given band, the cross-polarization terms can also be multiplied, allowing polarization information for the target source to be recovered.

2.2.2. Integration of Correlated Output

Once the above cycle of operations has been completed, it is repeated and the resulting visibilities are accumulated (complex added) until a set accumulation time has been reached. The number of "good" cycles per telescope is recorded, which could form the basis of a data weighting scheme, although weights are not currently recorded in DiFX. Generally, on each cycle the input time increment is equal to the corresponding FFT length (twice the number of spectral points), but it is also possible to overlap FFTs. This allows more measurements of higher lags and greater sensitivity to spectral line observations, at the cost of increased computation. In this way, the limiting time accuracy with which accumulation can be performed is equal to the FFT length divided by the overlap factor. A caveat to this statement is discussed in § 3.4.

2.2.3. Calibration for Nominal Telescope $T_{\rm sys}$

Cross multiplication, accumulation, and normalization by the antenna autocorrelation spectra gives the complex cross-power spectrum for each baseline, representing the correlated fraction of the geometric mean of the powers detected at each telescope. To obtain the correlated power in units of Jy, the cross-power spectra (amplitude components) should be scaled by the geometric mean of the powers received at each telescope, measured in Jy; i.e., the $T_{\rm sys}$ in Jy routinely measured at each antenna. Calibration based on the measured $T_{\rm sys}$ is typically performed as a postcorrelation step in AIPS[9] or a similar data analysis package, and so a nominal value for the $T_{\rm sys}$ for each telescope is applied at the correlator. In addition, a scaling factor to compensate for decorrelation due to the coarse quantization of the baseband data is applied. This corrects the visibility amplitudes but of course cannot recover the lost S/N. For the 2 bit data typically processed, this scaling factor is 1/0.88 in the low-correlation limit (Cooper 1970). The relationship becomes nonlinear at high correlation, and the scaling factor approaches unity as the correlation coefficient approaches unity. The correction for high-correlation cases can be applied in postprocessing, generally at the same time as the application of measured $T_{\rm sys}$ values.

2.2.4. Export of Visibility Data

Once an accumulation interval has been reached, the visibilities must be stored in a useful format. Currently, the software correlator supports RPFITS[10] as the output format. RPFITS files can be loaded into analysis packages, such as AIPS, CASA,[11] or MIRIAD,[12] for data reduction. Ancillary information is included in the RPFITS file, along with the complex visibilities, time stamps, and (u, v, w)-coordinates. The RPFITS standard supports the appending of a data weight to each spectral point, but DiFX does not currently record weights. Additional widely used output formats, such as FITS-IDI,[13] are expected to be added in the future.

2.3. Special Processing Operations: Pulsar Binning

Pulsed signals are dispersed as they travel through the interstellar medium (ISM), resulting in a smearing of the pulse arrival time in frequency. In order to correct for the dispersive effects of the ISM, DiFX employs incoherent dedispersion (Voûte et al. 2002). This allows the visibilities generated by the correlator to be divided into pulse phase bins. Unlike hardware correlators, which typically allow only a single on/off bin, or else employ 2^N bins of fixed width, DiFX allows an arbitrary number of bins placed at arbitrary phase intervals. The individual bins can be written out separately in the RPFITS file format to enable investigation of pulse phase–dependent effects, or can be filtered within the correlator based on a priori pulse profile information.

To calculate the corresponding phase bin for a visibility at a given frequency and time, the software correlator requires information on the pulsar's ephemeris, which is supplied in the form of one or more "polyco" files containing a polynomial description of apparent pulse phase as a function of time. These are generated using the pulsar analysis program Tempo[14] and require prior timing of a pulsar. Additional software has been written by the authors to verify the pulsar timing using the generated polyco files and the baseband data (in Mark 5, LBA, or K5 format) from an experiment, allowing phase bins to be accurately set before correlation.

[9] See http://www.aoc.nrao.edu/aips.

[10] See http://www.atnf.csiro.au/computing/software/rpfits.html.

[11] See http://casa.nrao.edu.

[12] See http://www.atnf.csiro.au/computing/software/miriad.

[13] See http://www.aoc.nrao.edu/aips/FITS-IDI.html.

[14] See http://pulsar.princeton.edu/tempo/reference_manual.html.

For VLBI observations of pulsars, it is usually desirable to maximize the S/N of the observations by binning the visibilities based on the pulse phase and applying a filter to the binned output based on the signal strength in that phase. Typically, this filter is implemented as a binary on/off for each phase bin. By using the pulse profile generated from the baseband data of an observation, however, DiFX allows a user-specified number of bins to be generated and a filter to be applied based on pulse strength × bin width, allowing the maximum theoretical retrieval of signal, as described below. This also reduces the output data volume, since only an "integrated on-pulse" visibility is retained, rather than potentially many phase bins.

Consider observing a single pulse divided into M equally spaced phase bins. Let the pulsar signal strength as a function of phase bin be $S(m)$, and the noise in single phase bin to be $Z\sqrt{M}$, where Z is the baseline sensitivity for an integration time of a single pulse period. When all bins are summed (effectively no binning), the S/N will be

$$\frac{\sum_{m=0}^{M} S(m)}{Z}, \tag{1}$$

as the signal adds coherently while the noise adds in quadrature. For a simple on/off gate accepting only bins m_1 to m_2, the S/N will be

$$\frac{\sum_{m=m_1}^{m_2} S(m)}{\sqrt{\sum_{m=m_1}^{m_2} (Z\sqrt{M})^2}}. \tag{2}$$

Finally, for the case in which each bin is weighted by the pulse signal strength in that bin, the S/N will be

$$\frac{\sum_{m=0}^{M} [S(m)]^2}{\sqrt{\sum_{m=0}^{M} [S(m)Z\sqrt{M}]^2}}. \tag{3}$$

For a Gaussian-shaped pulse, this allows a modest improvement in recovered S/N of 6% compared to an optimally placed single on/off bin. On a more complicated profile, such as a Gaussian main pulse with a Gaussian interpulse at half the amplitude, the improvement in recovered S/N increases to 21%.

3. SOFTWARE CORRELATION ON THE SWINBURNE BEOWULF CLUSTER: A CASE STUDY

3.1. The Cluster Computing Environment

The Swinburne University of Technology supercomputer is an ~300 processor Beowulf cluster that is a mixture of commodity off-the-shelf desktop and server-style PCs connected via a gigabit Ethernet network. In particular, the supercomputer has five subclusters, each with 48 machines. Four subclusters are made up of single-processor, 3.2 GHz Pentium 4 PCs with 1 Gbyte of RAM per machine, while one subcluster is made up of dual-processor Xeon servers, each with 2 Gbytes of RAM. The cluster is continuously upgraded and fully replaced approximately every 3–4 years. The software correlation code must operate in this multiuser, multitasking, and highly dynamic environment.

3.2. Structure of the DiFX Code

DiFX is written in C++ but makes heavy use of the optimized vector–processing routines provided by Intel's Integrated Performance Primitive (IPP) library.[15] The use of this optimized vector library results in a factor-of-several performance gain on the Intel CPUs, compared to nonoptimized vector code. Data transfer is handled via the Message Passing Interface (MPI) standard.[16] The MPICH implementation of MPI is used.[17]

Figure 1 shows the high-level class structure of DiFX, along with the data flow. The correlation is managed by a master node (FxManager), which instructs data management nodes (DataStream) to send time ranges of baseband data to processing nodes (Core). The data are then processed by the Core nodes, and the results are sent back to the FxManager. Double-buffered, nonblocking communication is used to avoid latency delays and maximize throughput. Both the DataStream and Core classes can be (and have been) extended to allow maximum code reuse when handling different data formats and processing algorithms. The Core nodes make use of an allocatable number of threads to maximize performance on a heterogeneous cluster.

The DataStream nodes can read the baseband data into their memory buffers from a local disk, a network disk, or a network socket. Once the data are loaded into the DataStream buffer, the remainder of the system is unaware of its origin. This is one of the most powerful aspects of this correlator architecture, meaning the same correlator can easily be used for production disk–based VLBI correlation and real-time "eVLBI" testing, in which the data are transmitted in real time from the telescopes to the correlator over optical fiber. Real-time eVLBI operational modes have been tested using DiFX, transmitting data in real time from the three ATNF telescopes (Parkes, Australia Telescope Compact Array [ATCA], and Mopra) to computing resources at the Swinburne University of Technology and the University of Western Australia in Perth (a Cray XD-1 utilizing Opteron processors and onboard Xilinx FPGAs). The software correlator then correlates the transmitted data in real time. A full account of the new eVLBI capabilities of the Australian VLBI array will be presented elsewhere (C. Phillips et al. 2007, in preparation).

3.3. Operating DiFX

DiFX is controlled via an interactive graphical user interface (GUI), which calls the various component programs and helper

[15] See http://www.intel.com/cd/software/products/asmo-na/eng/perflib/ipp/index.htm.

[16] See http://www-unix.mcs.anl.gov/mpi.

[17] See http://www-unix.mcs.anl.gov/mpi/mpich1.

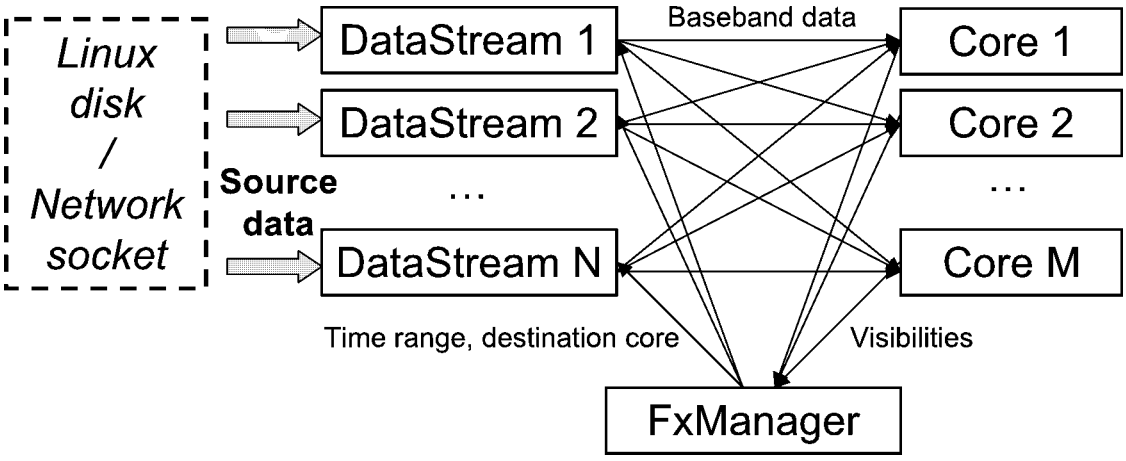

FIG. 1.—Overview of software correlator architecture. Data are loaded into memory from a disk or network connection by data-stream nodes. These nodes are directed by a master node (FxManager) to send data from given time ranges (typically several ms) to the processing elements (core nodes). The processed data are sent to the FxManager node for long-term accumulation and storage to disk.

scripts. The primary purpose of the GUI is to facilitate easy editing of the text files that configure the correlator, run external programs such as the delay model generator, and provide feedback while a job is running. Two files are necessary to run the actual correlator program. The first is an experiment configuration file containing tables of stations, frequency setups, etc., analogous to a typical hardware correlator job configuration script. The second file contains the list of compute nodes on which the correlator program will run.

While it is possible to manually run all tasks required to operate the correlator, in practice they are organized via the GUI. This consists of running a series of helper applications from the GUI to generate the necessary input for the correlator. These include a script to extract experiment information from the VLBI exchange (VEX) file used to configure and schedule the telescopes at observing time, a delay, and (u, v, w) generator that makes use of CALC 9, and scripts to extract the current load of available nodes. Pulsar-specific information, such as pulse profiles and bin settings, can also be loaded. This information is presented via the GUI, and adjustments to the configuration, such as selection computational resources to be used, can be made before launching a correlation job.

Plans for the future include incorporating some real-time feedback of amplitude, phase, and lag information from the current correlation via the GUI. This would be similar to the visibility spectra displays available continuously at connected-element interferometers.

3.4. Performance

In order to keep every compute node used in the correlation fully loaded, they must be kept supplied with raw data. If this condition is satisfied, we have a CPU-limited correlation, and the addition of further nodes will result in a linear performance gain. In practice, however, at some point, obtaining data from the data source (network socket or disk) and transmitting it across the local network to the processing nodes will no longer occur quickly enough, and the correlation becomes data-limited rather than CPU-limited. Correct selection of correlation parameters, and good cluster design, will minimize the networking overhead imposed on a correlation job and ensure that all compute nodes are fully utilized. This is discussed in § 3.4.1 below, and performance profiles for the CPU-limited case are presented in § 3.4.2.

3.4.1. Networking Considerations

As described in § 3.2, double-buffered communications to the processing nodes are used to ensure that nodes are never idle as long as sufficient aggregate networking capability is available. The use of MPI communications adds a small but unavoidable overhead to data transfer, meaning the maximum throughput of the system is slightly less than the maximum network capacity on the most heavily loaded data path.

There are two significant data flows: out of each DataStream and into the FxManager. For any high-speed correlation, there will be more Core nodes than DataStream nodes, so the aggregate rate into a Core will be lower than that out of a DataStream. The flow out of a Core is a factor of N_{cores} times lower than that into the FxManager node.

If processing in real time (when processing time equals observation time), the rate out of each DataStream will be equal to the recording rate, which can be up to 1 Gbps with modern VLBI arrays and is within the capabilities of modern commodity Ethernet equipment. The rate into the FxManager node will be equal to the product of the recording rate, the compression ratio, and the number of Cores, where the compression ratio is the ratio of data into a Core to data out of a Core. This is determined by the number of antennas (since the number of baselines scales with the number of antennas squared), the number of channels in the output cross-power spectrum, the number of polarization products that are correlated, and the

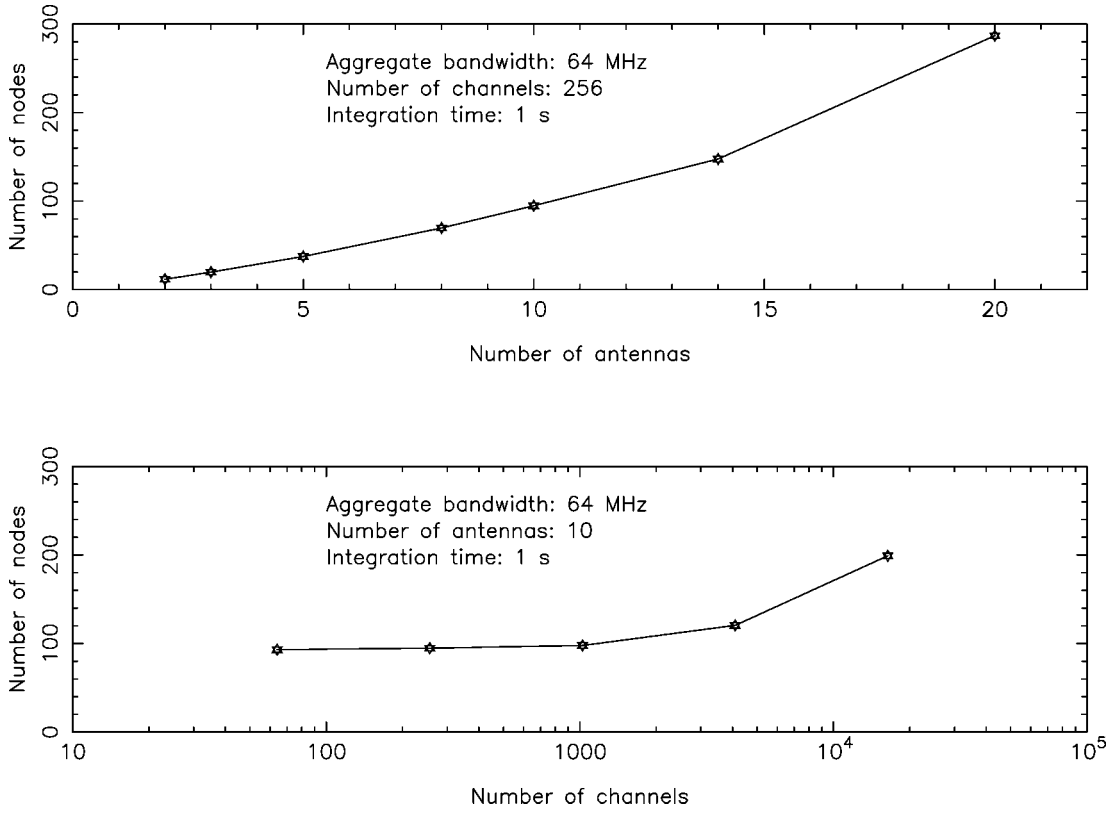

FIG. 2.—Benchmark data showing the computational requirements of DiFX to correlate in real time, as described in the text. The nodes are single-core 3.2 GHz Pentium processors with 1 Gbyte RAM, and in both benchmarks, 64 MHz of total bandwidth per station was correlated with a 1 s integration period. *Top*: Scaling of computational requirements with number of antennas, using 256 spectral points per 8 MHz subband. *Bottom*: Scaling of computational requirements with spectral points per subband for a 10 station array.

integration time used before sending data back to the FxManager node.

It is clearly desirable to maximize the size of data messages sent to a core for processing, since this minimizes the data rate into the FxManager node for a given number of Cores. However, if the messages are too large, performance will suffer as RAM capacity is exceeded. Network latency may also become problematic, even with buffering. Furthermore, it should be apparent that in this architecture, the Cores act as short-term accumulators (STAs), with the manager performing the long-term accumulation. The length of the STA sets the minimum integration time. It is important to note, however, that the STA interval is entirely configurable in the software correlator and can be as short as a single FFT, although network bandwidth and latency are likely to be limiting factors in this case.

For the majority of experiments, it is possible to set a STA length that satisfies all the network criteria and allows the Cores to be maximally utilized. However, for combinations of large numbers of antennas and very high spectral and time resolution, it is impossible to set an STA that allows a satisfactorily low return data rate to the FxManager node. In this case, real-time processing of the experiment is not possible without the installation of additional network and/or CPU capacity on the FxManager node.

It is important to emphasize that although it is possible to find experimental configurations for which the software correlator suffers a reduction in performance, these configurations would be impossible on existing hardware correlators. If communication to the FxManager node is limiting performance, it is also possible to parallelize a disk-based experiment by dividing an experiment into several time ranges and processing these time ranges simultaneously, allowing an aggregate processing rate that equals real time. This is actually one of the most powerful aspects of the software correlator, and one that would allow scheduling of correlation to always ensure the cluster was being fully utilized.

3.4.2. CPU-limited Performance

Figure 2 shows the results of performance testing on the Swinburne cluster (using the 3.2 GHz Pentium 4 machines and the gigabit Ethernet network) for different array sizes and spectral resolutions. The results shown in Figure 2 were obtained for data for which the aggregate bandwidth was 64 MHz, broken up into eight bands, each with an 8 MHz bandwidth (4 ×

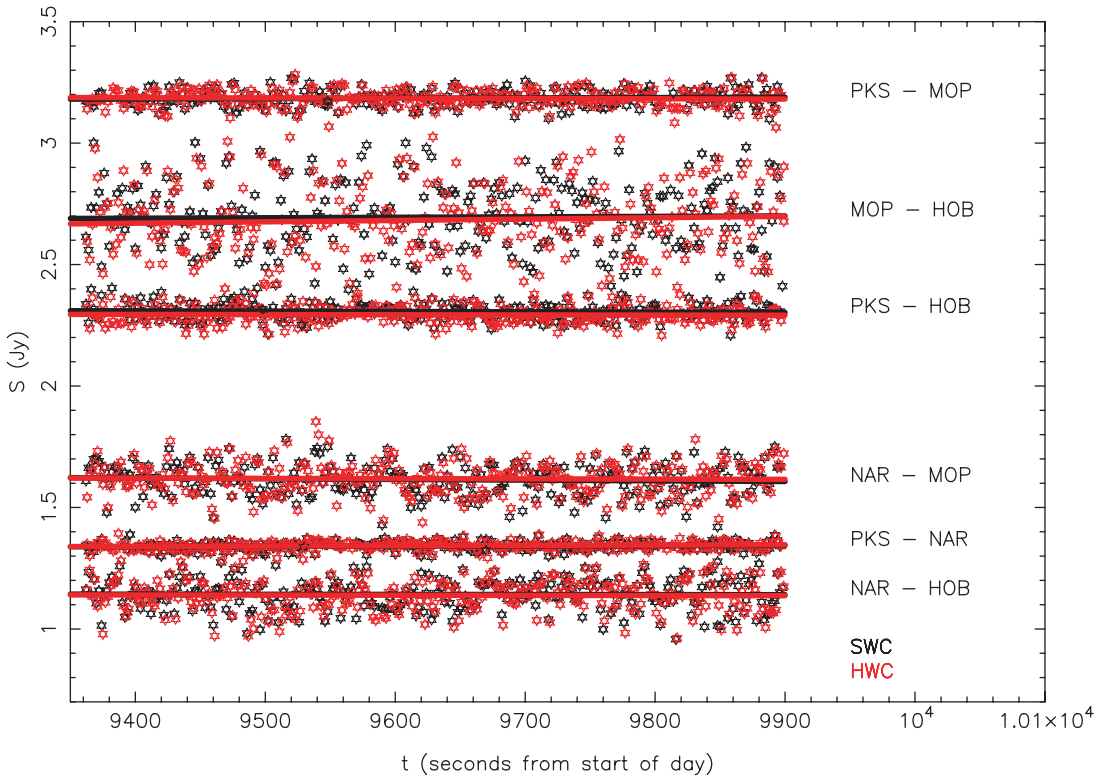

FIG. 3.—S2 (*red*) and DiFX (*black*) visibility amplitude vs. time for the 2252–2268 MHz band on the source PKS 0208−512, as described in the text (PKS = Parkes; MOP = Mopra; HOB = Hobart; NAR = ATCA). Symbols represent the actual visibilities produced by the correlators, while the lines represent linear least-squares fits to the visibilities (one line per data set).

dual-polarization 8 MHz bands; data were 2 bit sampled; antenna data rate was 256 Mbps). Node requirements for real-time operation are extrapolated from the compute time on an eight node cluster. The correlation integration time is 1 s, and all correlations provide all four polarization products. RAM requirements per node ranged from 10 to 50 Mbytes, depending on spectral resolution, showing that large amounts of RAM are unnecessary for typical correlations. It can be seen that even a modestly sized commodity cluster can process a VLBI-sized array in real time at currently available data rates.

3.5. Correlator Comparison Results

3.5.1. Comparison with ATNF S2 Correlator

Observations to provide data for a correlator comparison between the Swinburne software correlator and the ATNF S2 correlator were undertaken on 2006 March 12 with the following subset of the LBA: Parkes (64 m), ATCA (phased array of 5 × 22 m), Mopra (22 m), and Hobart (26 m). Data from these observations were recorded simultaneously to S2 tapes and the LBADR disks (C. Phillips et al. 2007, in preparation) during a 20 minute period (02:30–02:50 UT) corresponding to a scan on a bright quasar (PKS 0208−512). The recorded data corresponded to two 16 MHz bands, right circular polarization (RCP), in the frequency ranges 2252–2268 and 2268–2284 MHz.

The data recorded on S2 tapes were shipped to the ATNF LBA S2 correlator (Roberts 1997) at ATNF headquarters and were processed. The data recorded to LBADR disks were shipped to the Swinburne University of Technology supercomputer and were processed using the software correlator.

At both correlators, identical T_{sys} values in Jy were specified for each antenna and applied in order to produce nominally calibrated visibility amplitudes. Furthermore, both correlators used identical clock models in the form of a single clock offset and linear rate as a function of time per antenna. Finally, the data were processed at each correlator using 2 s correlator integration times and 32 spectral channels across each 16 MHz band.

Different implementations of the CALC-based delay generation were used at each correlator, meaning small differences exist in the delay models used, leading to differences in the correlated visibility phase. We have calculated the delay model differences and in the following discussion have subtracted the phase due to the delay model differential.

From both correlators, data were output in RPFITS format and loaded into the MIRIAD software (Sault et al. 1995) for

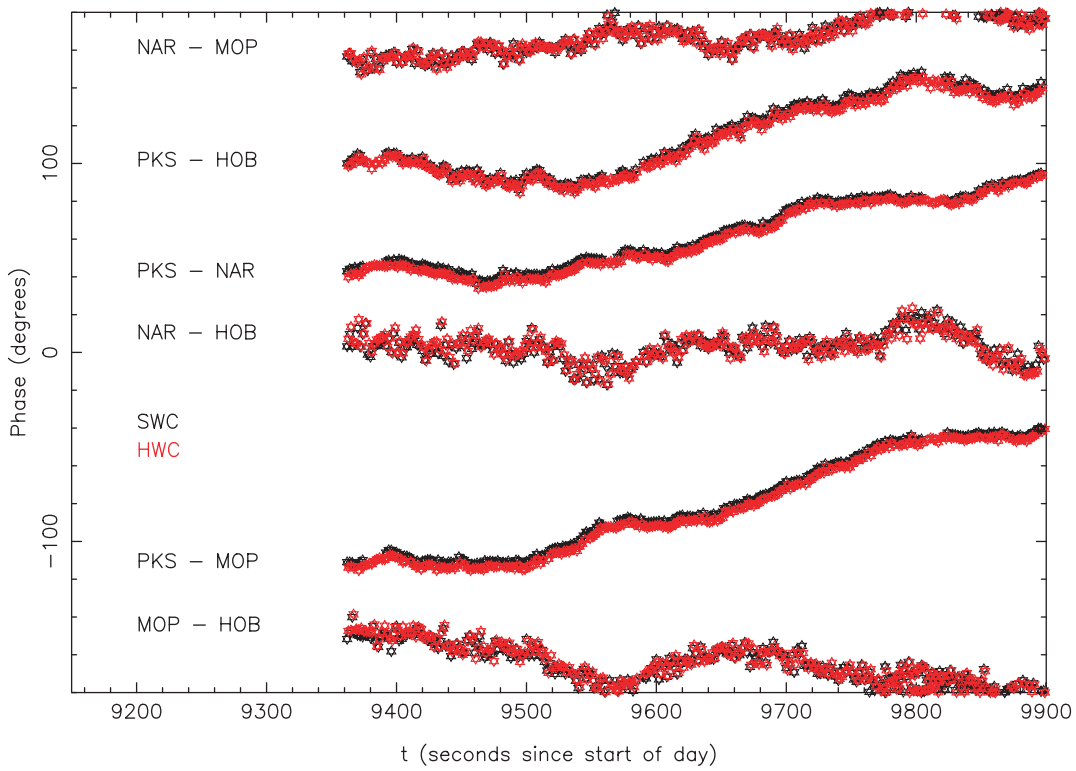

FIG. 4.—Same as Fig. 3, but for visibility phase vs. time. The PKS-NAR baseline has been shifted by −50° for clarity.

inspection and analysis. The data from the two correlators are compared in a series of images (Figs. 3–5).

Figure 3 shows the visibility amplitudes for all baselines from both correlators as a function of time over the period 02:36:00–02:45:00 UT for one of the 16 MHz bands (2252–2268 MHz). These amplitudes represent the vector-averaged data over the frequency channel range 10–21 (to avoid the edges of the band). The data for each baseline were fit to a first-order polynomial model ($S[t] = [dS/dt]t + S_0$, where S is the flux density in Jy, t is the offset in seconds from 02:40:30 UT, and S_0 is the extrapolated flux density at time 02:40:30 UT) using a standard linear least-squares routine. The rms variation around the best-fit model was calculated for each baseline. The fitted models are given in Figure 3 and show no significant differences between the S2 correlator and the software correlator. Furthermore, the calculated rms for each baseline shows very good agreement between DiFX and the S2 correlator, as summarized in Table 3.

Figure 4 shows the visibility phase as a function of time for each of the six baselines in the array. Again, the data represent the vector-averaged correlator output over the frequency channel range 10–21 within the 2252–2268 MHz band. As discussed above, small differences between the delay models used at each correlator have been taken into account as part of this comparison.

Figure 5 shows a comparison of the visibility amplitudes and phases as a function of frequency in the 2252–2268 MHz band. The data represented here result from a vector average of the two data sets over a 2 minute time range, 02:40:00–02:42:00 UT. Since the S2 correlator is an XF-style correlator, it cannot exactly correct fractional sample error in the same manner as an FX correlator such as DiFX, as the channelization is performed after accumulation. The coarse (postaccumulation) fractional sample correction leads to decorrelation at all points except the band center, up to a maximum of ~10% at the band edges on long baselines where the geometric delay changes by a sample or more over an integration period. We have corrected for this band edge decorrelation in the S2 correlator amplitudes in Figure 5.

3.5.2. Comparison with the VLBA Correlator

Data obtained as part of a regular series of VLBA test observations were used as a basis for a correlator comparison between the software correlator and the VLBA correlator (Napier et al. 1994). The observations were made on 2006 August 5 using the Brewster, Los Alamos, Mauna Kea, Owens Valley, Pie Town, and Saint Croix VLBA stations. One bit digitized data sampled at the Nyquist rate for four dual-polarization bands, each with an 8 MHz bandwidth, were recorded using the Mark

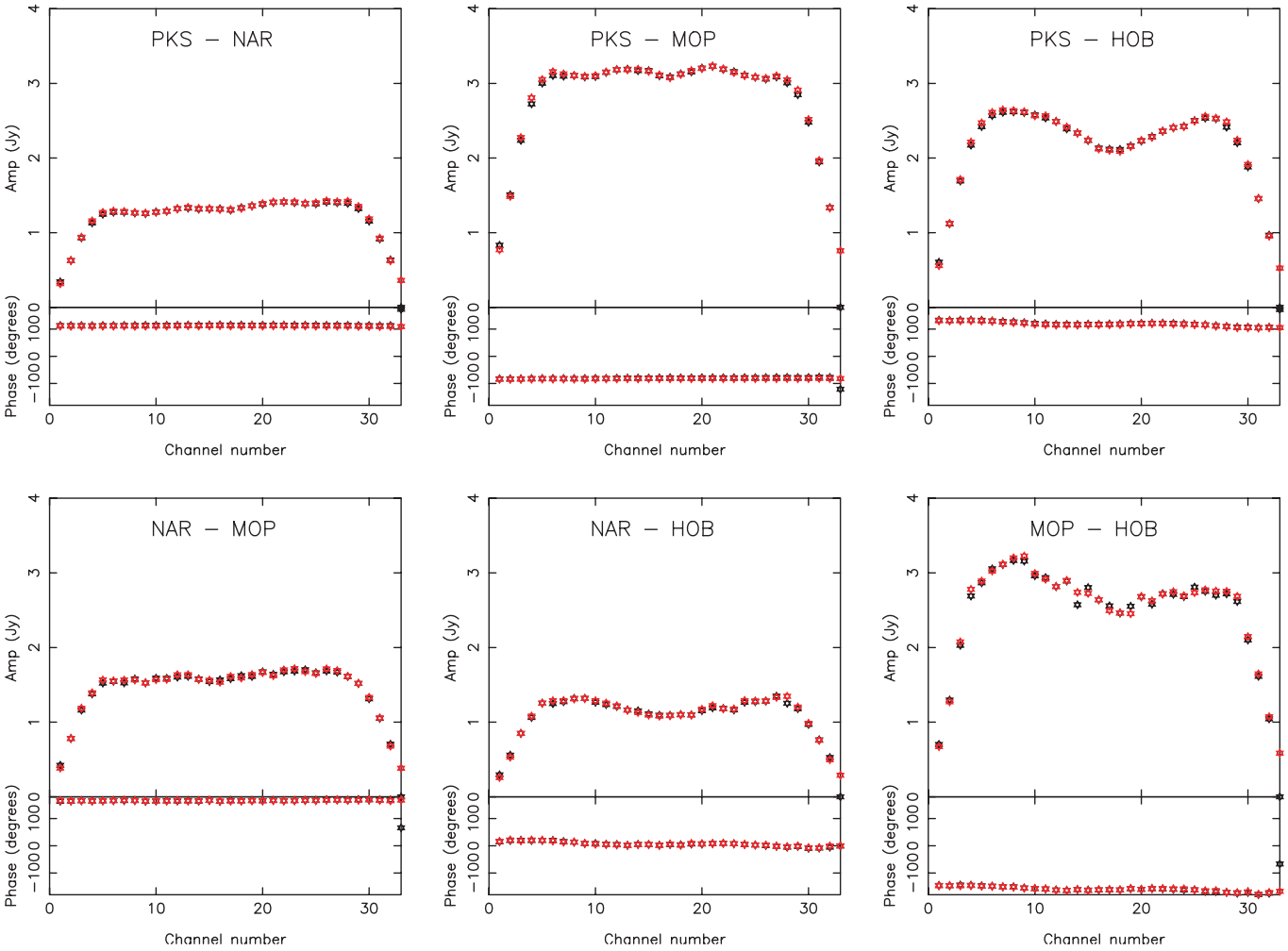

FIG. 5.—Same as Fig. 3, but for visibility amplitude and phase vs. frequency data. The S2 data have been corrected for fractional-sample error decorrelation at the band edges as described in the text.

5 system (Whitney 2003). The four bands were at center frequencies of 2279.49, 2287.49, 2295.49, and 2303.49 MHz. The experiment code for the observations was MT628, and the source observed was 0923+392, a strong and compact active galactic nucleus. Approximately 2 minutes of data recorded in this way were used for the comparison.

The Mark 5 data were correlated on the VLBA correlator and exported to FITS format files. The data were also shipped to the Swinburne supercomputer and correlated using the software correlator; the correlated data were exported to RPFITS format files. In both cases, no scaling of the correlated visibility amplitudes by the system temperatures were made at the correlators. The visibilities remained in the form of correlation coefficients for the purposes of the comparison; i.e., a system temperature of unity was used to scale the amplitudes. Each 8 MHz band was correlated with 64 spectral points, and an integration time of 2.048 s was used.

The VLBA correlator data were read into AIPS using FITLD with the parameter DIGICOR = 1. The DIGICOR parameter is used to apply certain scalings to the visibility amplitudes for data from the VLBA correlator. Furthermore, to obtain the most accurate scaling of the visibility amplitudes, the task ACCOR was used to correct for imperfect sampler thresholds, deriving corrections of ~0.5% to the antenna-based amplitudes. These ACCOR corrections were applied to the data, which were subsequently written to disk in FITS format.

The software correlator data were read directly into AIPS and then written to disk in the same FITS format as the VLBA correlator data. No corrections to amplitude or phase of the software-correlated data were made in AIPS.

Both VLBA and software correlator data were imported into MIRIAD for inspection and analysis, using the same software as used for the comparison with the LBA correlator described above. RCP from the 2283.49–2291.49 MHz band over the

TABLE 3
LINEAR-FIT PARAMETERS FOR VISIBILITY AMPLITUDE VERSUS TIME FOR THE DiFX AND LBA S2 CORRELATORS

Baseline	$\text{Offset}_{\text{DiFX}}$ (Jy)	$\text{Offset}_{\text{LBA}}$ (Jy)	$\text{Slope}_{\text{DiFX}}$ (μJy s^{-1})	$\text{Slope}_{\text{LBA}}$ (μJy s^{-1})
PKS-NAR	1.341 ± 0.030	1.343 ± 0.028	10 ± 13	14 ± 12
PKS-MOP	3.185 ± 0.058	3.185 ± 0.063	14 ± 24	−11 ± 26
PKS-HOB	2.307 ± 0.058	2.293 ± 0.061	−12 ± 24	−6 ± 24
NAR-MOP	1.616 ± 0.109	1.619 ± 0.114	−27 ± 43	−10 ± 45
NAR-HOB	1.142 ± 0.111	1.139 ± 0.116	−3 ± 44	−5 ± 46
MOP-HOB	2.694 ± 0.256	2.681 ± 0.257	18 ± 101	56 ± 101

NOTE.—Linear-fit parameters reflect 95% confidence limits.

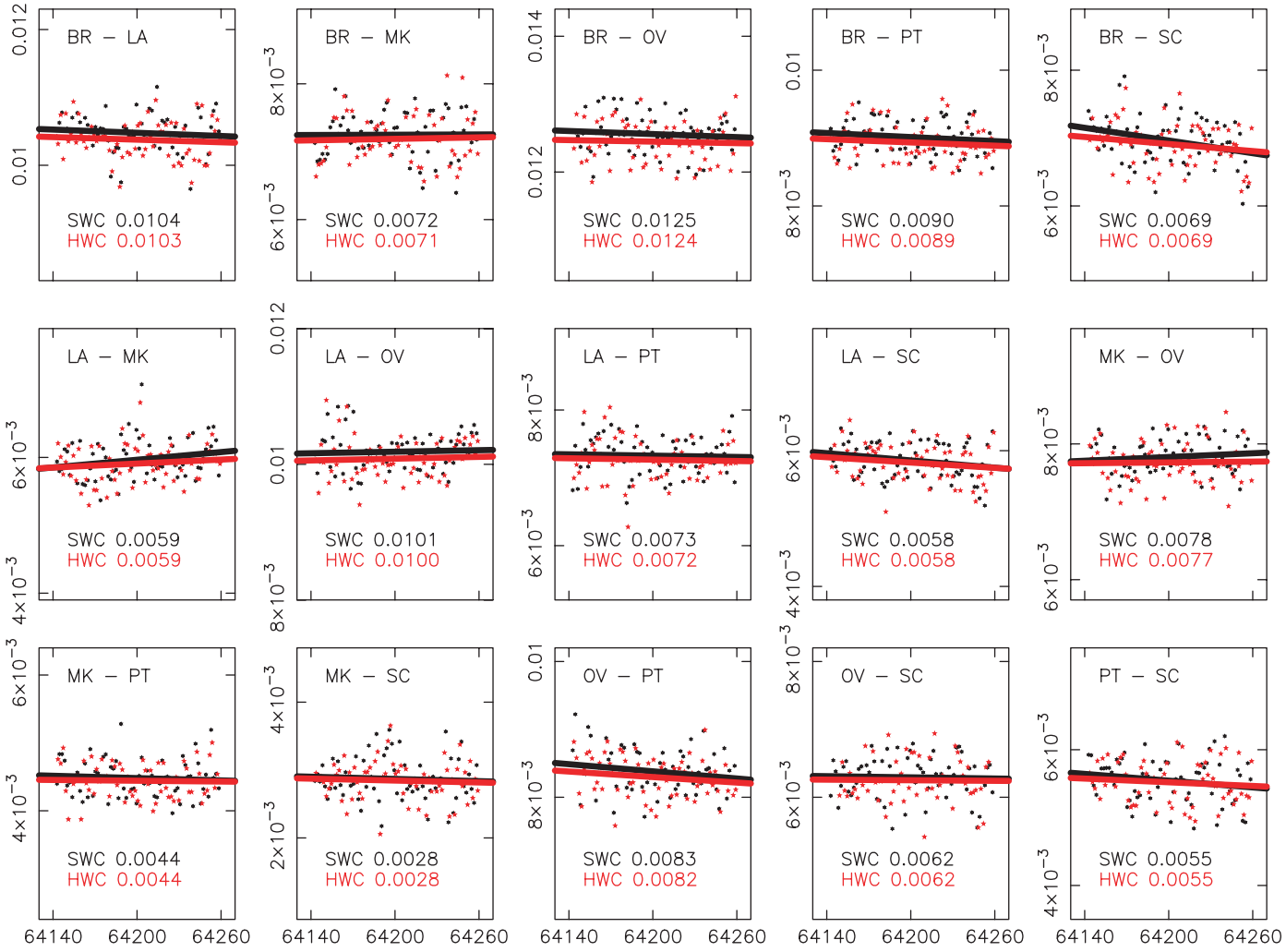

FIG. 6.—VLBA correlator (red) and DiFX (black) visibility amplitude vs time for the 2283.49–2291.49 MHz RCP band from the VLBA test observation MT628, as described in the text. The units of time are seconds from UT 00:00:00, and the amplitude scale is correlation coefficient. Symbols represent the actual visibilities produced by the correlators, while the lines represent linear least-squares fits to the visibilities. The text annotation on each panel lists the average correlation coefficient amplitude for each correlator over the time period, as tabulated in Table 4.

TABLE 4
LINEAR-FIT PARAMETERS FOR VISIBILITY AMPLITUDE VERSUS TIME FOR THE DiFX AND VLBA CORRELATORS

Baseline	$\text{Offset}_{\text{DiFX}}$	$\text{Offset}_{\text{VLBA}}$	$\text{Slope}_{\text{DiFX}}$ ($s^{-1} \times 10^{-6}$)	$\text{Slope}_{\text{VLBA}}$ ($s^{-1} \times 10^{-6}$)
BR-LA	0.0104 ± 0.0004	0.0103 ± 0.0005	−0.8 ± 1.7	−0.9 ± 1.7
BR-MK	0.0072 ± 0.0005	0.0071 ± 0.0006	0.1 ± 1.8	0.5 ± 2.0
BR-OV	0.0125 ± 0.0005	0.0124 ± 0.0005	−0.7 ± 1.7	−0.5 ± 1.8
BR-PT	0.0090 ± 0.0004	0.0089 ± 0.0004	−1.0 ± 1.3	−1.2 ± 1.5
BR-SC	0.0069 ± 0.0005	0.0069 ± 0.0005	−3.1 ± 2.0	−2.5 ± 1.8
LA-MK	0.0059 ± 0.0005	0.0059 ± 0.0005	1.9 ± 1.7	1.4 ± 1.7
LA-OV	0.0101 ± 0.0005	0.0100 ± 0.0005	0.4 ± 1.7	0.6 ± 1.7
LA-PT	0.0073 ± 0.0005	0.0072 ± 0.0005	−0.3 ± 1.7	−0.5 ± 1.8
LA-SC	0.0058 ± 0.0004	0.0058 ± 0.0004	−1.8 ± 1.5	−1.9 ± 1.5
MK-OV	0.0078 ± 0.0004	0.0077 ± 0.0005	0.9 ± 1.5	0.3 ± 1.8
MK-PT	0.0044 ± 0.0004	0.0044 ± 0.0004	−0.6 ± 1.7	−0.3 ± 1.5
MK-SC	0.0028 ± 0.0005	0.0028 ± 0.0005	−0.6 ± 1.8	−0.7 ± 1.7
OV-PT	0.0083 ± 0.0005	0.0082 ± 0.0005	−1.8 ± 1.8	−1.9 ± 1.7
OV-SC	0.0062 ± 0.0005	0.0062 ± 0.0005	−0.3 ± 1.8	−0.2 ± 1.8
PT-SC	0.0055 ± 0.0005	0.0055 ± 0.0005	−1.7 ± 2.0	−1.3 ± 1.8

NOTES.—Visibility amplitude is given in units of correlation coefficient. Linear-fit parameters reflect 95% confidence limits.

time range 17:49:00–17:51:00 UT was used in all comparison plots.

Since the delay models used by the VLBA and software correlators differ at the picosecond level, as is the case for the comparison with the LBA data in § 3.5.1, differences in the visibility phase exist between the correlated data sets. As with the LBA comparison, we have compensated for the phase error due to the delay model differences in the following comparison.

Figure 6 shows the visibility amplitudes for all baselines from both correlators as a function of time. These amplitudes represent the vector-averaged data over the frequency channel range 10–55 (to avoid the edges of the band). The data for each baseline were fit to a first-order polynomial model ($S[t] = [dS/dt]t + S_0$, where S is the correlation coefficient, t is the offset in seconds from 17:50:00 UT, and S_0 is the extrapolated correlation coefficient at time 17:50:00 UT) using a standard linear least-squares routine. The rms variation around the best-fit model was calculated for each baseline. The fitted models are given in Figure 6 and show no significant differences between the VLBA correlator and the software correlator. Furthermore, the calculated rms for each baseline shows very good agreement between the VLBA and software correlators. The results of the comparison are summarized in Table 4.

Figure 7 shows the visibility phase as a function of time for each of the 15 baselines in the array. Again, the data represent the vector-averaged correlator output over the frequency channel range 10–55 within the band. As discussed above, small differences between the delay models used at each correlator cause phase offsets between the two correlators and have been taken into account as part of this comparison.

Figure 8 shows a comparison of the visibility amplitudes and phases as a function of frequency in the band. The data represented here result from a vector average of the two data sets over a 2 minute time range. Figures 6, 7, and 8 show that the results obtained by the VLBA correlator and DiFX agree to within the rms errors of the visibilities in each case, as expected.

4. SCIENTIFIC APPLICATIONS OF THE SWINBURNE SOFTWARE CORRELATOR

4.1. High Frequency Resolution Spectral Line VLBI

As mentioned in the introduction, an attractive feature of software correlation is the ease with which very high spectral resolution correlation can be undertaken. This is particularly useful for studies of spectral line sources such as masers when mapping the distribution of the masing regions and their kinematics; i.e., near black holes in galactic nuclei (Greenhill et al. 1995).

Figure 9 shows a spectrum obtained from an LBA observation of the OH maser G345−0.2. These observations were made with an array consisting of the ATCA (phased array of 5 × 22 m), Parkes (64 m), and Mopra (22 m), recording data from a dual-polarized (RCP and LCP) 4 MHz band onto hard disk. The data were correlated using the software correlator with 16,384 frequency channels across the 4 MHz band, corresponding to 0.25 kHz per channel, or a velocity resolution of 0.038 km s^{-1} at 1.72 GHz.

These results compare with recent very high spectral resolution work done with the VLBA. Fish et al. (2006) observed OH masers with the VLBA using a 62.5 kHz bandwidth and 512 channels across this band to obtain channel widths of 0.122 kHz, or a 0.02 km s^{-1} velocity resolution. The velocity resolution of this correlated data set is almost twice as good as that shown in Figure 9. However, the VLBA bandwidth is only 0.016 times the bandwidth of the observations shown in Figure 9.

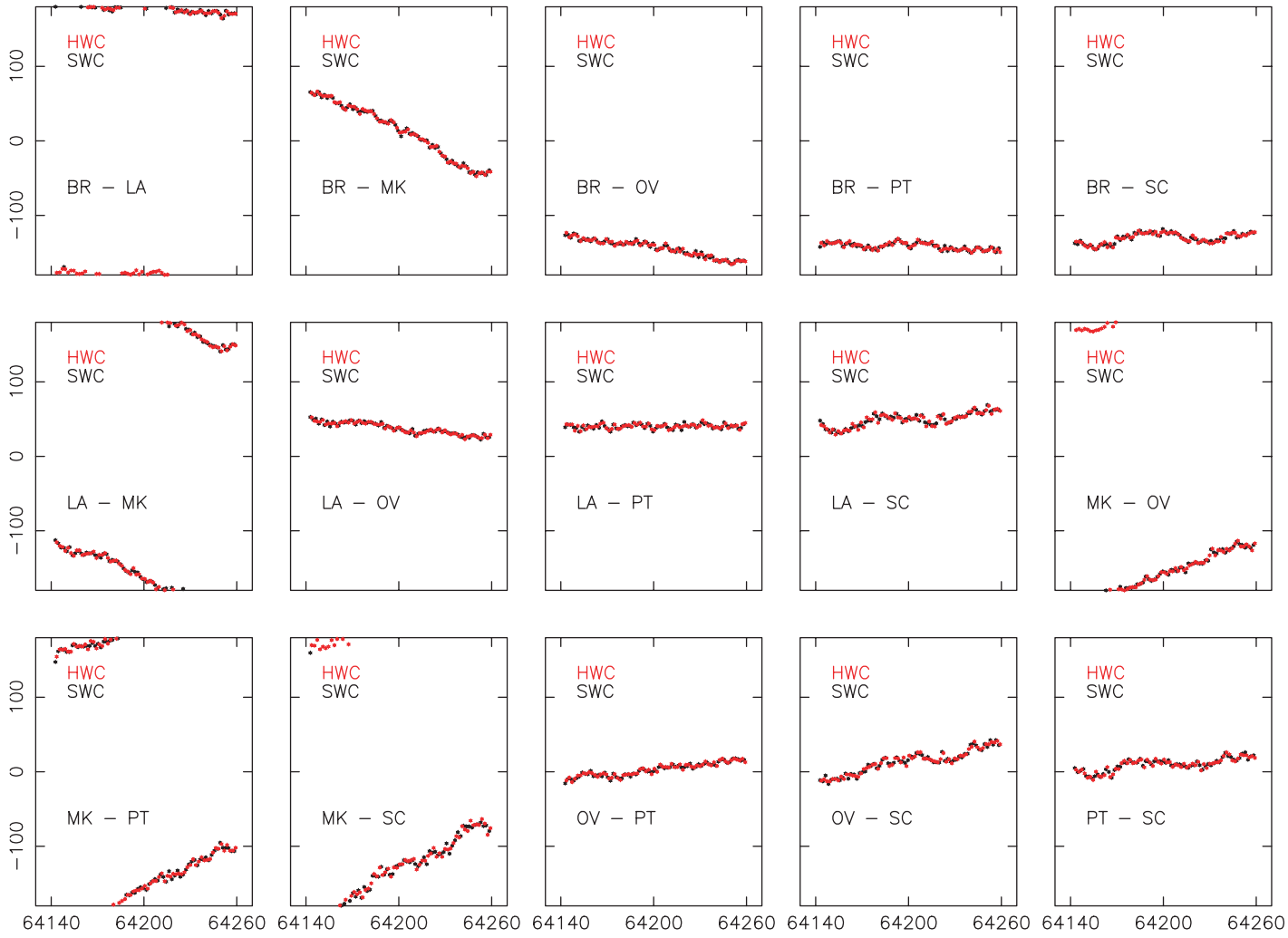

FIG. 7.—VLBA correlator (*red*) and DiFX (*black*) visibility phase vs. time for the 2283.49–2291.49 MHz RCP band from the VLBA test observation MT628, as described in the text. The units of time are seconds from UT 00:00:00, and phase is displayed in degrees.

If required, DiFX could have correlated these data with 32,768 channels, 65,536 channels, or an even higher numbers of channels. As mentioned in the introduction, the only penalty is compute time on a resource with a fixed number of processing elements. DiFX therefore has a clear advantage over existing hardware correlators in terms of producing very high spectral resolution over wide bandwidths. This capability is useful if the velocity distribution of an ensemble of masers in a field is broad and cannot be contained in a single narrow bandwidth.

4.2. Correlation for Wide Fields of View

One application that takes advantage of the frequency and time resolution of the software correlator output is wide-field imaging. To image a wide field of view, avoiding the effects of time and bandwidth smearing, high spectral and temporal resolution is required in the correlator visibility output. For example, at VLBI resolution (40 mas), to image the full primary beam of an ATCA antenna (22 m diameter) at a frequency of 1.4 GHz requires a correlator output time resolution of 50 ms and a frequency resolution of 4 kHz (allowing a 0.75% smearing loss at the FWHM of the primary beam).

Neither the JIVE nor the VLBA hardware correlators can achieve such high frequency or time resolution for continuum experiments, but DiFX can be configured for such modes in an identical manner to a normal continuum experiment.

4.3. Pulsar Studies

As compact sources with high velocities, pulsars make excellent test beds with which to probe the structure of the ISM. Scintillation due to structure in a scattering screen between the observer and the pulsar causes variations in the interferometric visibilities, which have some dependence on time and fre-

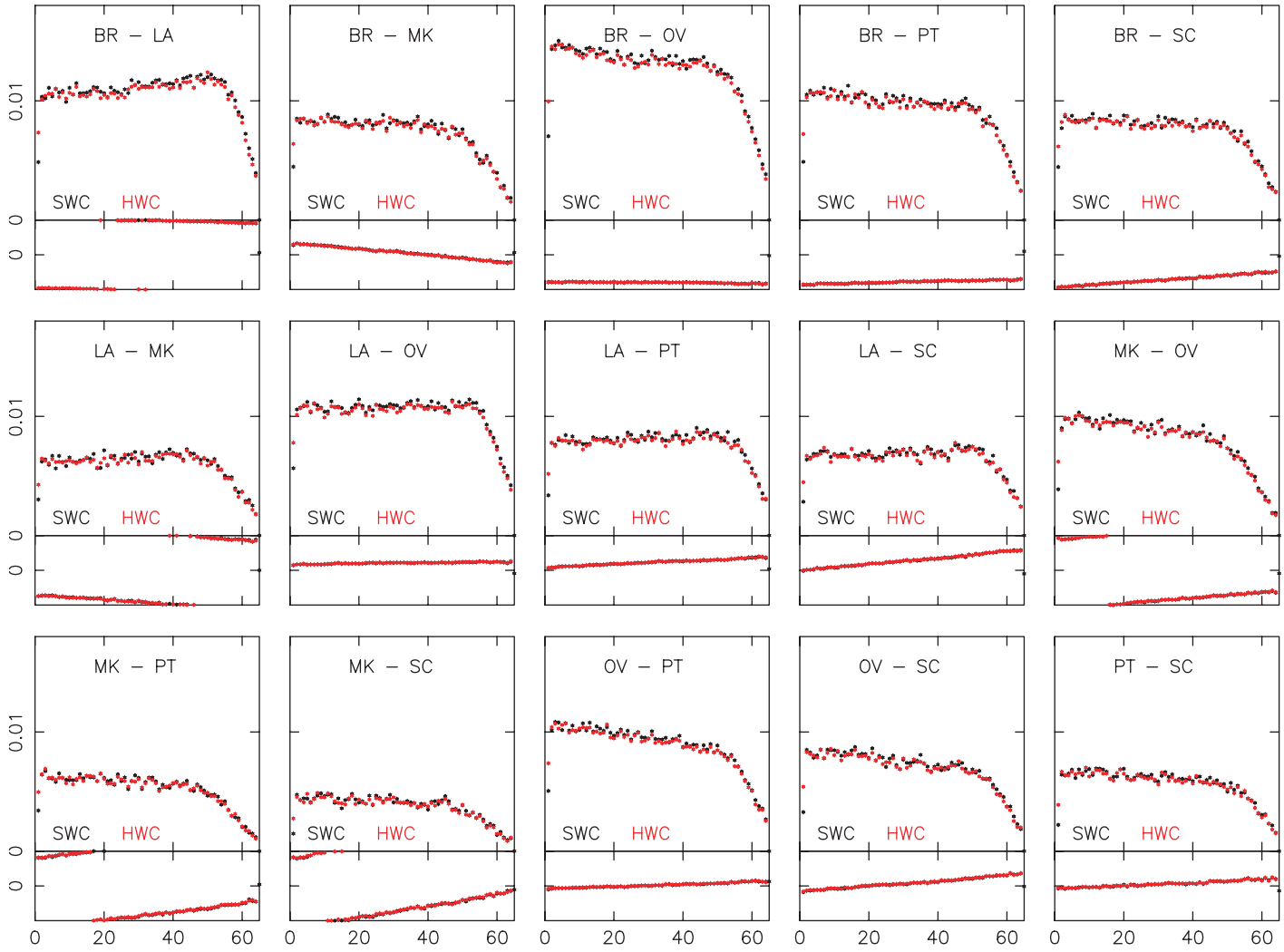

FIG. 8.—VLBA correlator (*red*) and DiFX (*black*) visibility amplitude and phase as a function of frequency for the 2283.49–2291.49 MHz RCP band from the VLBA test observation MT628, as described in the text. The vertical scale for correlation coefficient amplitude on each panel is 0–0.018, while the phase scale spans $\pm 180°$. The horizontal scale for each panel displays channels 0–64.

quency (e.g., Hewish et al. 1985). Naturally, pulsar binning is advantageous in these studies for maximizing S/Ns.

The most stringent requirement for useful studies of pulsar scintillation, however, is that of extremely high frequency resolution. W. Brisken et al. (2007, in preparation) have recently demonstrated the capabilities of DiFX for this type of analysis with observations of the pulsar B0834−04. The NRAO Green Bank Telescope (100 m), Westerbork (14 × 25 m), Jodrell Bank (76 m), and Arecibo (305 m) were used to provide an ultrasensitive array at 327 MHz. The data were recorded using the Mark 5 system and correlated on the Swinburne software correlator. The main requirement on the correlation was 0.25 kHz wide frequency channels over the broadest bandwidth available in order to maximize the S/N. For these observations, a 32 MHz band was available. The Swinburne software correlator therefore correlated the data with 131,072 frequency channels across the band.

No existing hardware correlator can provide such a high frequency resolution over such a wide bandwidth. Full details of the interpretation of the B0834−04 software-correlated data will be available from W. Brisken et al. (2007, in preparation). Figure 10 provides a section of the dynamic spectrum from this observation, showing the scintillation structure as functions of time and frequency.

4.4. Geodetic VLBI

In addition to astronomical VLBI, the software correlator can also be deployed for geodetic VLBI. Compared to astronomical VLBI, geodetic VLBI has additional requirements,

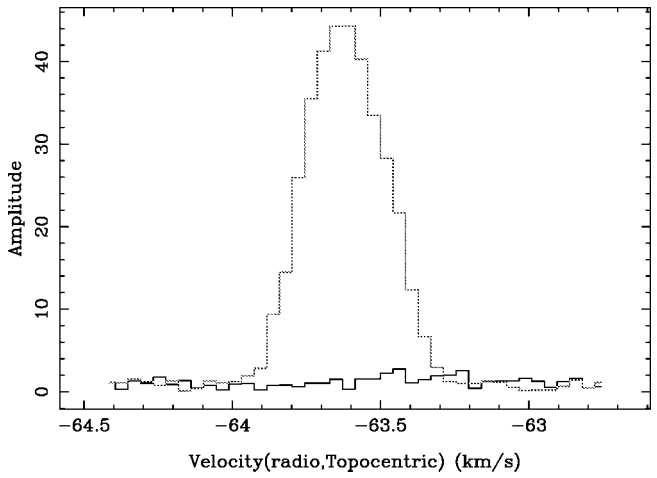

FIG. 9.—A 2 minute average of the ATCA-Parkes cross-power spectrum taken from the software-correlated data for the OH maser G345−0.2, as described in the text. The velocity resolution is 0.038 km s^{-1} at the central frequency of 1.72 GHz. The light gray line showing strong maser emission represents the LCP data, and the dark gray line with little emission represents the RCP data. The maser is highly circularly polarized.

including different output formats and the frequent use of subarraying. The flexibility and capabilities of the software correlator are well matched to this task.

The software correlator has been tested on geodetic data sets obtained using the Mark 5 recording system, consisting of 16 frequency bands. These tests form the basis of a geodetic correlation comparison between the software correlator and the geodetic correlator of the Max-Plank-Institut für Radioastronomie in Bonn, Germany. Full results of this correlator comparison will be reported elsewhere (S. J. Tingay et al. 2007, in preparation).

In particular, in Australia a new three-station geodetic VLBI array has been funded as part of the geospatial component of the federal government's National Collaborative Research Infrastructure Scheme (NCRIS). This scheme provides for three new geodetic VLBI stations 12 m in diameter, Mark 5 recording systems, and a modified version of the software correlator described in this paper. The modifications necessary to convert DiFX into a geodetic correlator consist of the addition of phase calibration tone extraction, a streamlined interface to scan-by-scan correlation for subarraying, and a capability to produce visibilities in a format convenient for geodetic postprocessing.

The new Australian geodetic VLBI array will participate in global geodetic observations, as well as undertake experiments that are internal to the Australian tectonic plate.

5. CONCLUSIONS

In this paper, we have outlined the main benefits of software correlation for small- to medium-sized VLBI arrays. They are as follows:

1. The development of software correlation is rapid and does

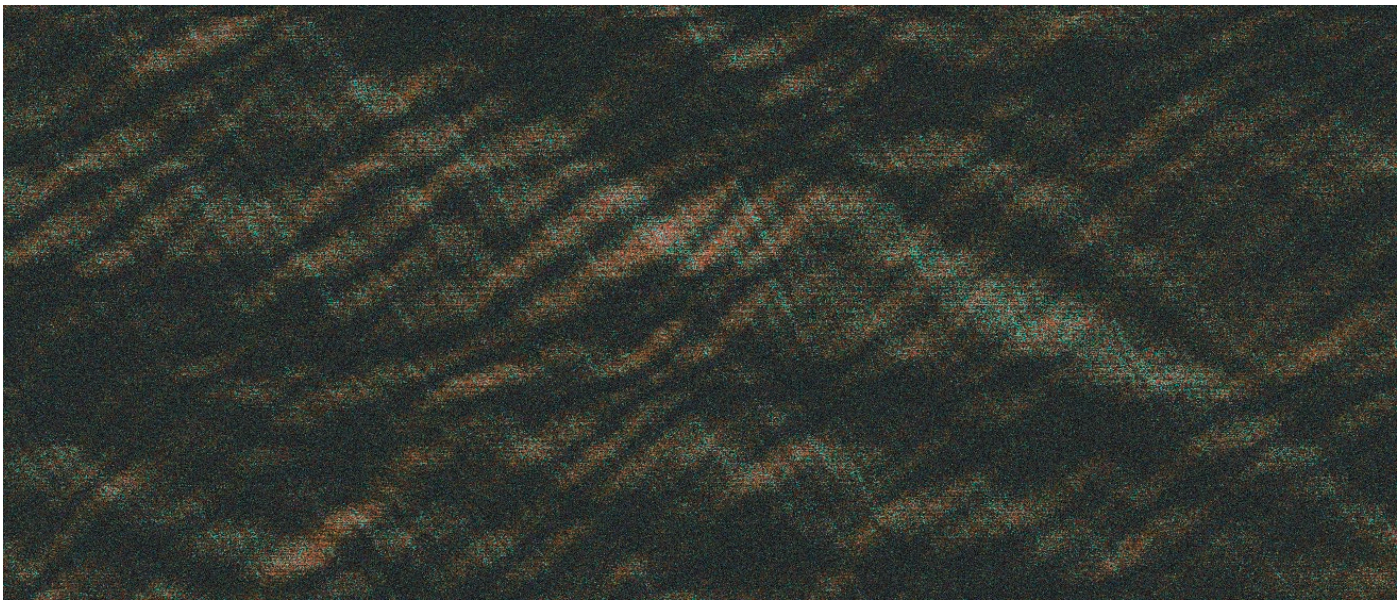

FIG. 10.—Cross-power dynamic spectrum showing scintillation variations for the pulsar B0834−04 on the Green Bank Telescope–Arecibo baseline. Brightness represents the visibility amplitude, and color represents the visibility phase. Increasing frequency runs left to right, and increasing time runs top to bottom. This section of the dynamic spectrum represents just 5% of the time span and 0.5% of the bandwidth of the observation (330 s and 160 kHz, respectively).

not depend on an intimate knowledge of digital signal-processing hardware, just the algorithms.

2. The software is flexible and scalable to accommodate a very broad range of interferometric modes of observation, including many that cannot be supported by existing ASIC-based hardware correlators. Software correlators are therefore ideal for novel experiments with very special requirements. The main trade-off for improved performance with a software correlator is the increase in computational time for a fixed number of processing elements, or the addition of extra processing elements.

3. The software can easily incorporate data recorded using mixed disk-based recording hardware.

4. Medium to large multiprocessor computing facilities are available at almost all university and government research institutions, allowing users easy entry into VLBI correlation.

5. The correlation algorithm is highly parallel and very well suited to a parallel multiprocessor computing environment.

6. The cost of commodity computing continues to fall with time, making large parallel computing facilities more powerful and less expensive.

7. Once written, the code can be ported to a wide range of platforms and recompiled with minimal effort.

We have discussed the implementation of the DiFX software correlator on a standard Beowulf cluster at the Swinburne University of Technology and have provided performance figures of merit for this implementation, showing that relatively large numbers of telescopes and relatively high data rates can be correlated in "real time" using numbers of machines that do not exceed the capabilities of moderate to large Beowulf clusters. Clear trade-offs are possible in many areas of performance. For example, if real-time operation is not important, it is possible to dramatically reduce the number of processing elements.

We have also shown the results of comprehensive testing of the software correlator, comparing its output to that of two established hardware correlators, the S2 correlator of the Australian Long Baseline Array, operated by the ATNF, and the VLBA correlator. The correlator comparisons of visibility amplitude and phase as functions of time and frequency verify that DiFX is operating correctly for astronomical VLBI observations.

DiFX now supports all Australian VLBI observations and some global VLBI experiments, at data rates up to 1 Gbps per telescope. The DiFX code can be downloaded online (see footnote 4). A number of scientific programs have already been supported by the software correlator and are briefly discussed here. Furthermore, a modified version of the software correlator will be used to support a new VLBI array in Australia, dedicated to local and global geodetic observations.

This work has been supported by the Australian federal government's Major National Research Facilities program, the Australian Research Council's (ARC) Strategic Research Initiatives (eResearch) scheme, and the ARC's Discovery Projects scheme. A. T. D. is supported via a Swinburne University of Technology Chancellor's Research Scholarship and a CSIRO postgraduate scholarship. The Long Baseline Array is part of the Australia Telescope, which is funded by the Commonwealth of Australia for operation as a National Facility managed by CSIRO. The National Radio Astronomy Observatory is a facility of the National Science Foundation operated under cooperative agreement by Associated Universities, Inc. We wish to thank Walter Brisken for kindly making available Figure 10 prior to publication, the NRAO (Walter Brisken, Craig Walker, and Jon Romney) for making available data for the VLBA correlator comparison, and Gary Scott for correlating the LBA S2 data for comparison with the ATNF S2 correlator.

REFERENCES

Bailes, M. 2003, ASP Conf. Ser. 302, Radio Pulsars, ed. M. Bailes, D. J. Nice, & S. E. Thorsett (San Francisco: ASP), 57
Bare, C., et al. 1967, Science, 157, 189
Bellanger, M., & Daguet, J. 2004, IEEE Trans. Commun., COM-22, 1199
Carlson, B. R., et al. 1999, PASP, 111, 1025
Casse, J. L. 1999, NewA Rev., 43, 503
Clark, B. G. 1973, Proc. IEEE, 61, 1242
Clark, B. G., Cohen, M. H., & Jauncey, D. L. 1967, ApJ, 149, L151
Cooper, B. F. C. 1970, Australian J. Phys., 23, 521
Finley, D. G., & Goss, W. M., ed. 2000, NRAO Workshop Number 27, Radio Interferometry: The Saga and the Science (Washington: Associated Universities)
Fish, V. L., Brisken, W. F., & Sjouwerman, L. O. 2006, ApJ, 647, 418
Greenhill, L. J., et al. 1995, ApJ, 440, 619
Hewish, A., Wolszczan, A., & Graham, D. A. 1985, MNRAS, 213, 167
Horiuchi, S., et al. 2000, Adv. Space Res., 26, 625
———. 2007, ApJS, submitted
Kondo, T., Koyama, Y., Nakajima, J., Sekido, M., & Osaki, H. 2003, in ASP Conf. Ser. 306, New Technologies in VLBI, ed. Y. C. Minh (San Francisco: ASP)
Moran, J. M., et al. 1967, Science, 157, 676
Napier, P. J., Bagri, D. S., Clark, B. G., Rogers, A. E. E., Romney, J. D., Thompson, A. R., & Walker, R. C. 1994, Proc. IEEE, 82, 658
Pogrebenko, S., Gurvits, L. I., Campbell, R. M., Avruch, I. M., Lebreton, J.-P., & van't Klooster, C. G. M. 2003, in Workshop on Planetary Probe Atmospheric Entry and Descent Trajectory Analysis and Science, ed. A. Wilson (ESA SP-544; Noordwijk: ESA)
Roberts, P. P. 1997, A&AS Ser., 126, 379
Rogers, A. E. E., et al. 1983, Science, 219, 51
Romney, J. D. 1999, in ASP Conf. Ser. 180, Synthesis Imaging in Radio Astronomy II, ed. G. B. Taylor, C. L. Carilli, & R. A. Perley (San Francisco: ASP), 57
Sault, R. J., Teuben, P. J., & Wright, M. C. H. 1995, in ASP Conf. Ser. 77, Astronomical Data Analysis Software and Systems IV, ed. R. A. Shaw, H. E. Payne, & J. J. E. Hayes (San Francisco: ASP), 433

Thompson, A. R. 1999, in ASP Conf. Ser. 180, Synthesis Imaging in Radio Astronomy II, ed. G. B. Taylor, C. L. Carilli, & R. A. Perley (San Francisco: ASP), 11
Thompson, A. R., Moran, J. M., & Swenson, G. W. 1994, Interferometry and Synthesis in Radio Astronomy (Malabar: Kreiger)
Voûte, J. L. L., et al. 2002, A&A, 385, 733
West, C. 2004, M.S. thesis, Swinburne Univ. Tech.
Whitney, A. R. 1993, in IAU Symp. 156, Developments in Astrometry and Their Impact on Astrophysics and Geodynamics, ed. I. I. Mueller & B. Kolaczek (Dordrecht: Kluwer), 151
Whitney, A. R. 2002, in Proc. 6th European VLBI Network Symp., New Developments in VLBI Science and Technology, ed. E. Ros et al. (Bonn: Max-Planck-Institut für Radioastronomie), 41
———. 2003, in ASP Conf. Ser., New technologies in VLBI, 306, 123
Wietfeldt, R., et al. 1996, IEEE Trans. Instrumentation and Measurement, 45, 923
Wilson, W., Roberts, P., & Davis, E. 1996, in Proc. 4th APT Workshop, ed. E. A. King (Sydney: ATNF/CSIRO), 16