CHARA TECHNICAL REPORT

NO. 51 31 JULY 1997

A 'Strawman' Observing Method and the Array Control System

T.A. TEN BRUMMELAAR

1. INTRODUCTION AND GENERAL INFORMATION

One of the important subsystems left to be fully specified is the overall control system for the CHARA Array. Part of the difficulty in completely specifying this control system is the remaining areas of doubt concerning what an observing session will actually be like and what an operator might expect of the control system. In an attempt to remove some of this uncertainty, this technical report describes my view of what observing with the array will consist of. It is largely based on my experience observing with SUSI and will contain my personal biases concerning the importance and ordering of the procedures. Nevertheless, this document should help us make a start at understanding what the control system needs to be able to do.

2. BRIEF DESCRIPTION OF SUBSYSTEMS

Before describing the observation procedures I will first attempt a brief description of each of the subsystems within the array. The break up of the interferometer into subsystems has been done from a control point of view and may differ from similar analyses done in the past. I will also try to list the inputs and outputs for each subsystem as well as an estimate of the digital servo cycle time and amount of processing power required.

2.1. Telescopes

Inputs : Required target coordinates, tip/tilt error signal, slew commands, stow commands, focus commands for acquisition and telescope, and acquisition data.

Outputs: Status (dome position, telescope position etc.) and video signals from acquisition and other video monitors.

Data Rates: Tracking object: minutes. Tip/tilt error: milliseconds. Status: seconds. Video: seconds.

[1]Center for High Angular Resolution Astronomy, Georgia State University, Atlanta GA 30303-3083
TEL: (404) 651-2932, FAX: (404) 651-1389, FTP: ftp.chara.gsu.edu, WWW: http://www.chara.gsu.edu

Processing Load: Easily handled by a PC or equivalent. A small stand-alone computer may be required for the tip/tilt driver. A controller box, such as those made by Compumotor, would be capable of handling most of the low-level telescope and dome functions.

The main task of the telescope is to acquire and track the required science object. There are two servos running here, one is part of the fast tip/tilt servo, controlled from inside the beam combining lab (BCL), and the other is the telescope tracking itself, basically a slave to the tip/tilt servo. In acquisition mode it must be possible for the operator to see the field and indicate which object is the desired target. Once on the target, a tip/tilt error signal used to drive the secondary tip/tilt mirror arrives from the BCL. The telescope tracking software then ensures that the secondary stays within its allowed range of movement. The tracking software must also be able to function closed-loop using the acquisition signal as feedback.

Also included as part of the telescope subsystem are the dome and enclosures themselves. The dome must be controlled to rotate at the correct times, as well as open and close both the upper and lower parts of the dome slit. The enclosures will probably be of the 'telescoping cylindrical' design proposed by Sea West, in which case the telescope control system will also need to be able to switch power on and off to this system and monitor its current position.

2.2. Pipes of Pan (PoP)

Inputs : Selected delays and fine alignment data.

Outputs: Status (current mirror positions).

Data Rates: Minutes.

Processing Load: Easily handled by a PC or equivalent. In fact, a single PC is probably adequate to handle the PoP's, the dispersion correctors and all the filter wheels and aperture stops within the BCL area.

The Pipes of Pan (PoP's) provide DC vacuum delay. This is done by switching various mirrors in and out of the system. It should not be necessary (or desirable) to change the PoP setting frequently, and it should certainly *not* be done between the science target and the calibrator.

2.3. Dispersion and Refraction Corrector

Inputs: Zenith angle, differential air path for each beam, slew commands, and stow commands.

Outputs: Status (current position of prisms).

Data Rates: Seconds.

Processing Load: See section 2.2 above.

The two dispersive correction systems are the atmospheric refraction corrector (ARC) for correcting problems caused by atmosphere differential refraction, and the longitudinal dispersion corrector (LDC) for correcting problems caused within the instrumental air paths themselves. These systems should perform all the calculations required internally and will be open loop, that is, use only the astrometric model positions.

2.4. Optical Path Length Equalizer (OPLE)

Inputs : Delay position, delay rate of change, error signals, slew commands, stow commands, and path length modulation functions.

Outputs: Status (Current positions and velocities).

Data Rates: Initially in the 0.1 second range, eventually to go as fast as millisecond rates.

The OPLEs provide the continuously variable optical delays. The signals used to calculate the required positions and rates are derived from the fringe tracker. The OPLE control system itself comes as part of the JPL contract to build the OPLE carts and the details of the interface are unknown at this time.

2.5. BCL shutters, filters

Inputs : Required shutter and filter wheel positions.

Outputs: Status (actual shutter and filter wheel positions).

Data Rates: Minutes.

Processing Load: See section 2.2 above.

There will be many shutters, aperture wheels and other adjustable elements in the BCL area. While they may not all be controlled by a single computer, they are conceptually part of a single subsystem. Included in this subsystem is the means of adjusting the position of the quadrant detectors.

2.6. Tip/Tilt detection

Inputs : Start/stop, sample rate, offsets, and current Alt/Az of object.

Outputs: Star position for each channel, status information, current rotation matrix, and r_0 estimates.

Data Rates: Milliseconds.

Processing Load: Easily handled by a PC or equivalent. This PC will probably have to be dedicated to the tip/tilt system and could do the required calculations for the r_0 measurements.

The tip/tilt detectors measure the position of the stellar image within each optical channel. These data are used as an error signal for the tip/tilt mirrors in the telescopes themselves. Because we are using an Alt/Az system for the telescopes, there will be a field rotation between the tip/tilt quadrant detectors in the BCL and the tip/tilt mirrors at the telescopes. Furthermore this rotation will slowly change as the objects moves across the sky. The tip/tilt system will be responsible for calculating the appropriate rotation matrix given the current Alt/Az of the telescopes. Due to small differences in alignment, it may be that a different matrix is required for each telescope.

The tip/tilt system is also responsible for providing the spatial seeing data r_0 used in calibrating the visibility estimates.

2.7. Fringe Tracking

Inputs : Optical bandpass, channel size, and sample time.

Outputs: Delay error for each optical line, visibility estimates, status, data for logging, and t_0 measurements.

Data Rates: Milliseconds (internal). Same as OPLE line for external signals.

Processing Load: Many DSP boards and a powerful computer are required. It is also likely that the detector will need its own PC.

The fringe tracker represents the big digital signal processing challenge for the instrument. If each of seven channels is used on each side of the beam splitter for each of fives pairs of beams, the system will need to perform $2 \times 7 \times 5 = 70$ Fourier transforms per servo cycle, which will be as fast as 10 milliseconds and hopefully down to 1 or 2 milliseconds. The visibility estimates also require a large number of calculations, although less than for the fringe tracker.

It is also essential that this system be capable of logging all the raw and processed data, not only for error checking but also for post-processing of the visibility data. The fringe tracker is also responsible for providing the temporal seeing data t_0 used in calibrating the visibility estimates.

2.8. Imaging

Inputs : Required optical bandpass, number of pixels, and sample time.

Outputs: Raw and processed counts, status, and data for logging.

Data Rates: Milliseconds (internal). Seconds (external).

Processing Load: A PC is probably adequate.

The imaging system is not part of a servo loop and need not provide output signals to the control system other than status information. The operator would, however, need to see the output produced by the system, if only as a check on what is going on. Thus, while it need not be in real time, it will be necessary to be able to output current image registers and counts for display by the control system. Internally the clock rate will be of the order of a millisecond. No strict external communication requirements exist.

As for the fringe tracker, it is necessary that this subsystem be capable of logging all raw and processed data for debugging and post-processing.

2.9. Central Scrutinizer

Inputs : All signals listed for all other subsystems.

Outputs: All signals listed for all other subsystems.

Data Rates: All rates listed for all other subsystems.

Processing Load: A powerful workstation of some kind, perhaps with a number of slave machines.

This is the central control engine for the entire array. It must be able to talk to all subsystems, display their status information, issue commands to each system, close all servo loops and be able to switch power on and off to various subsystems within the array. Making the coffee would also help. It is likely that this system also controls data logging and helps monitor non-essential subsystems (the vacuum levels, outside temperature and weather conditions, current seeing conditions and so on).

It is possible that this machine allow outside monitoring of the facility (on a web page for example) *but most certainly not outside commands.* It will be a long time before the CHARA Array is a remote observing facility, and it would be extremely dangerous to allow an external computer to issue commands. It has been suggested that the entire control system be shielded from the Internet in order to avoid the possible entry of 'hackers' onto the system. If this is the case, some flow of information outside of the Array will still be required. Perhaps the best solution is to route all outside communications through a separate server which will be able to block any undesired data packets flowing in either direction.

Another responsibility of the central scrutinizer is to run the *astrometric model*, a complete computer model of the entire Array, including all systems and subsystems. This model should provide the Alt/Az for the object, the positions and velocities of the ARC, LDC and OPLE carts, the recommended PoP set up and so on. Before closed-loop tracking can begin, or if it fails, this model can be used to keep various subsystems working until the fringes are found.

3. OBSERVING PROCEDURE

The observing procedure has been broken up into three parts. The first is a daily alignment check, probably done by 'day staff' (if we have any) last thing before they go home. The second part is a list of things to do in order to acquire and measure fringe visibility on an object. Finally, there will be periodic checks of internal alignment during the night. With any luck, the number of these checks will be small and should reduce as the system stabilizes; however, they will need to be performed at least once per night.

3.1. Afternoon Alignment and Checks

The alignment checks performed in the afternoon consist mainly of ensuring that none of the beams have moved too far out of position. In SUSI we found that, over time, this alignment settled down to checking only one or two mirrors in the system. Every few months, however, a more complete internal alignment was required.

Before any alignment checks can be performed, the various systems in the array need to be switched on and booted up. While this may seem a trivial point, it is certainly not going to be a trivial operation. For example, most moving systems, like the OPLE carts and the telescopes, will need to find their fiducial marks and slew to the zero positions. We found at SUSI that it was best to park the carts at or near their fiducial marks at the end of a night's run rather than do it first thing the next day. Communications between the various subsystem controllers will also need to be checked. Finally, powerful lasers, like that being used as a reference light source, often need a substantial warmup time.

With the system up and running, the first step in the alignment check is to insert the laser light into a beam and ensure that it is entering the OPLE and vacuum system correctly. A small mask placed over the vacuum window is helpful for this. One or two mirrors may need to be tweaked and this process repeated for each beam. If the optical train is stable, this may be all that is required; however, during commissioning of the array more will be necessary.

A second test that may need to be performed is to ensure that all beams inside the BCL are properly optically superimposed. Once again using the laser, you must now auto-collimate from various positions and check that the quadrant detectors, fringe tracker, imaging systems and reference detector all agree. If you have auto-collimated correctly, the reference detector should be centered and focused. If it isn't, you need to move the reference detector. From this point on the reference detector can be used as a remote check for each beam position. It should not be moved until another alignment is performed. Furthermore, if you find you do need to move the reference detector, it indicates an instability in the optical chain.

The correctly positioned reference detector will be used to align many other optical systems within the array. It would be very handy for it to be possible to view the output of the reference detector at numerous places throughout the optical chain. At SUSI I spent many hours running long video cables so that I could see the reference camera output at the ends of the OPLEs and so on. Even having the signal out at the telescopes will be very handy. Furthermore, small CCDs placed in various locations within the BCL will also be useful, and it would be nice to see their output in places other than near the optical tables. I therefore suggest we have a frame-grabber board, like those used in the telescope domes, on the central scrutinizer so that any computer can display images from these cameras.

The third test ensures that the 'center' of the acquisition camera field on each telescope is properly aligned with the center of the reference detector and, therefore, the rest of the optical chain. Send the laser light all the way out to the telescopes and auto-collimate from the small retro-reflector cube on the telescope secondary. With the acquisition system on, you should see the laser spot in the acquisition field. Note that this means it is essential that it be possible to have light going into both the acquisition camera and the rest of the optical chain *at the same time*. This is also a good time to focus the acquisition field. In this way the acquisition field is correctly focused with respect to the fringe tracking and imaging systems and can be used as a reference later for focusing the telescope itself. It is also necessary to record the position of the laser spot in the acquisition field as it is this position that will be used as an origin when acquiring science objects.

A fourth test, and one that should not need to be performed often, is to check the 'zero points' of the OPLE carts. This means switching from the laser to the white light source and auto-collimating from the telescope secondary retro-reflection cubes. The OPLE carts should then be slewed to their 'zero points,' and white light fringes of high contrast should be seen in both the fringe tracking and imaging systems. The fringe tracker can then be used to center the carts on the fringe envelope and the cart positions recorded. This mode is a good way to test the fringe tracker and gives us a good origin for the carts. This will need to be repeated for each of the PoP positions you intend to use. It may be a good idea to check that this will actually be possible for all PoP configurations: is there enough rail to do this for each PoP configuration?

3.2. Acquiring a New Object

There are many optical systems that need to be set up before observing an object, the most difficult of which, apart from actually finding fringes, is likely to be the telescope-tip/tilt combination.

Before setting up any system, the astrometric model needs to be started for the object required. Once this model is running, all subsystems can be told to slew to, and start tracking on, the model positions. This means that most systems will be close to or, if a miracle occurs, at the correct positions and moving at the correct velocity. The default for all systems will be to track to this model. In this way if tracking fails for some reason, no system will drift too far away from the correct position. For some subsystems, the ARC and LDC for example, there will be no closed-loop mode, and they will always track the astrometric model.

With the model running and all systems tracking the astrometric model it is time to acquire the object in each of the telescopes. The pelical that brings some of the light from the telescope beam path into the acquisition camera is inserted and the operator uses a mouse to indicate the target, probably by drawing a box around it. It may be necessary to perform a spiral search, but the telescope pointing should be good enough to get the object in the acquisition field. The acquisition system then produces an error signal that the telescope can use to guide the source into the correct position. Note that the 'correct' position will change from night to night, and it is the position found during the afternoon alignment described above that is used as an origin.

This should mean that the light from this telescope reaches all the way through to the quadrant detectors in the BCL. If it does not, the afternoon alignment checks went wrong somehow and you are in a bit of trouble; a spiral *may* help. With light reaching the quadrant detector, a sample time is chosen and the tip/tilt servo can be closed. Assuming all is well and the tip/tilt servo is stable, the telescope tracking can be switched from the acquisition system over to the tip/tilt system. Basically, if the mean tip/tilt position wanders too far from the center, the telescope needs to speed up or slow down to re-center it. The acquisition pelical can now be removed and the system is tracking. It is important that if the light level on the quadrant detectors go below a lower limit the telescopes return to open-loop tracking and follow the astrometric model.

This acquisition process needs to be repeated for each telescope, and hopefully can one day be automated. For example, once one telescope is tracking, its acquisition field can be correlated with the acquisition fields of the other telescopes in order to find the science target. This may be enough information for the remaining telescopes to center up and begin closed-loop tracking with little on no assistance from the operator. If they close on the wrong target the operator can take over manual control.

We now have light from all telescopes reaching the BCL and all other systems tracking the astrometric model. If the model is good enough (and it should be after a bit of tweaking), the OPLE carts will be close enough to their correct positions to be within the fringe envelope. This will depend on the bandwidth selected at the fringe tracker; for bright stars this can be very narrow and the fringe envelope will be large. The fringe tracker can then be turned on, having selected appropriate sample times and so on, and the OPLE carts servoed to it. As for the telescopes, if the tracker loses the signal for some reason the OPLE carts should return to tracking the astrometric model.

If the astrometric model is not good enough to get the carts within the fringe envelope, a search algorithm will be required. This will most certainly be the case for the first fringes, and quite a time beyond that, until the model has enough data to work reliably. The searches will have to be done manually for the initial commissioning of each baseline, and certainly for finding the zero positions of the OPLE carts when using auto-collimated white light. A 'jog' function for the OPLEs, similar to that used on telescopes, will be invaluable here. The ability to run at a given offset from the astrometric model position is also very useful, if not essential for calibration of the fringe tracker and tuning the servo loop.

With all this completed, it is now time to setup and switch on the imaging systems and actually collect data. Having integrated for the required time (hopefully less time than the slit size of the telescope dome allows!), the systems can be stopped, in the opposite order in which they were started, and a new object can be acquired. It will be necessary to look at a calibrator object before and after each science object, and it is very important that no large changes be made to the optical system in between. It is also important that the calibrator be in the same part of the sky as the science object and of similar magnitude and color.

3.3. Periodic Alignment Checks

There will be a number of internal alignment checks necessary during the night, although how often each check is performed is something we can only work out by experience.

At the beginning of the night, and perhaps several times during the night, we will need to check that all the quadrant detectors are optically superimposed on the reference detector. Each beam is servoed onto its own quadrant detector, and, if they are not properly aligned, the beams will have constant tilts with respect to each other, reducing the visibility. The idea is to close the shutters on the fringe tracker for all but one beam and check to see that the image formed by this beam is in the correct position in the reference detector. If it is not centered, the quadrant detector for that beam needs to be moved until the image is properly centered. Since the entire telescope-tip/tilt system is locked onto the position of the quadrant detector, moving the detector will move the entire beam alignment. If you need to move the quadrant detector too much you have a problem, and the afternoon alignment check was probably not done correctly.

The reference detector will also be useful at this time to see if the ARC is operating correctly. An elongated (banana shaped) image means that the ARC has failed and needs adjustment.

With only one beam entering the fringe tracker and imaging systems, it is also a good idea to check the amount of light entering the optical fibers and adjust optical bandwidths and pixel size. If the count is very low there could be a problem with the alignment between the reference detector and the fiber tips. It may be possible to do some remote tweaking of the fibers to fix this, although it is likely that the best way to fix it is to use the alignment laser. Once again, this sort of problem should have been found during the afternoon alignment checks.

4. GENERAL HARDWARE AND SOFTWARE ISSUES

It is not the appropriate time to decide for certain which computer type, communication protocols and operating system we should use in the Array. I believe that we should standardize as much as possible, making purchasing and spare parts easier. We should use standard PC hardware wherever possible as they are cheap, powerful enough for most subsystems, and easily repaired or replaced. Furthermore, such hardware will not restrict our choice of OS, as all the potential software systems will run on a PC.

Since we are using optical fibers between the telescope domes and the BCL/OPLE area, TCP/IP is the logical choice for almost all communications within the Array. We will have no choice in this regard for the OPLE carts as this is the communication protocol used by JPL. Some high-speed dedicated serial lines may be required for some servo systems, such as tip/tilt, but this will be communications within a single subsystem. All communications between subsystems can be done using TCP/IP and Ethernet. Of course this will mean inventing network names for the various telescopes and subsystems, but I am sure we can come up with something.

One important issue not addressed so far is synchronization of the various digital servo systems. On a large scale, synchronization will be important for observing efficiency, for example, all the telescope domes should rotate at one time. On a smaller scale, synchronization is always important in a digital control environment. If the quadrant detectors and the tip/tilt mirrors are not properly timed, the servo can fail; the same is true for the fringe tracker/OPLE combination. The easiest way to achieve synchronization is to use a distributed global timing pulse, either via the TCP/IP or through a dedicated line. This way, all servos can run in sync and no oscillations or instabilities will occur.

Another element of digital, and indeed analog, servo design is the ability to inject and monitor signals within the servo loop. This is invaluable for debugging and tuning servo systems, and the ability to send test signals to various points within the feedback loop must be a part of every control system. Monitoring the same signals within the loop is equally important. For example, in the past I have built a simple 8-bit D/A and A/D converter box which can be connected to any parallel printer port. It is easy to have the software direct a variable value to the printer port allowing you to view its behavior on an oscilloscope. Signal injection can work in a similar manner. Spectrum analyzers and signal generators can then be used to tune the servo loop. Attractive displays for VIP visitors can also be generated in this manner.

The decision concerning the OS to be used has also yet to be made, and I would suggest waiting a while longer. We should also keep in mind that it is a good idea not to have too many different flavors of OS running within the control system. Some things, however, are already known: The OS for the OPLE carts will be VxWorks, the central scrutinizer and workstations will run a flavor of Unix, and the language used will be C or C++. Since we will need to buy a VxWorks license for the site, we may decide to use it throughout, although we know it will not be fast enough for tip/tilt or fringe tracking. VxWorks will run on almost any kind of CPU, including a PC, although it also has a per CPU charge which we may decide is too expensive. Alternatively we could regard the OPLEs as a 'black box' and overlook VxWorks.

There are really only three other choices that I see: Low-level DOS-based interrupt programming, Windows NT, and RT Linux. The low-level interrupt-driven DOS programs will be a quick and dirty way of getting things up and running. Unfortunately DOS will not be supported forever (on the other hand it might; look at FORTRAN), and all the communications and timing code will need to be written from scratch. Windows NT is the solution favored by Tom Schneider, mostly for it's GUI and development tools. We are none of us Windows fans, and while all the communications software will be there, NT is not a real-time OS. Obviously I do not favor using Windows NT. RT Linux is a free real-time kernel with all the development tools and communications capability of a full blown Unix system. It is also not a commercial product, and so we will have no one to complain to if it fails. We have to think about this some more, but I am leaning towards the RT Linux solution. I recommend attempting to install RT Linux on a PC in the near future so we can try it out and test it for long term stability.

Finally, it is possible to come up with a first order guess at the number of computers required to make the control system work. Table 1 contains a list of the various control subsystems and the number of computers required for each one. In this table MC means 'motor controller' and SC means 'small control computer'. Furthermore, the number of computers required for the OPLE has been set to zero as the OPLE control system is part of the JPL contract. The fringe tracker has been listed as having two PCs. This is because the CCD camera will probably require a dedicated PC. The cost of this computer may be added to the cost of the camera, but I have listed it as part of the control system. If we

TABLE 1. Computers Required.

Subsystem	CPUs	Number of Systems	Total CPUs
Telescope	1 PC 1 MC 1 SC	5 5 5	5 PC 5 MC 5 SC
OPLE	0 PC	1	0 PC
oOP/BCL/ARC/LDC	1 PC	1	1 PC
Tip/Tilt	1PC	1	1 PC
Fringe Tracking	2 PC 5 DSP	1 1	2 PC 5 DSP
Imaging	1 PC 1 DSP	1 1	1 PC 1 DSP
Central Scrutinizer	1 PC	1	1 PC
Workstations	2 PC	1	2 PC
Total			13 PC 5 MC 5 SC 6 DSP

say that each PC costs \$3k, each MC costs \$2k, each SC costs \$0.5k and each DSP costs \$1k, the total hardware cost is \$57.5k. This does not include the cabling required or the addition of uninterruptable power supplies and so on. On the other hand, many of the PCs will not require keyboards or monitors, and perhaps not even disks, and so will be cheaper than \$3k.

5. CONCLUSION

Clearly many unknowns remain regarding the CHARA Array control system, and a great deal will have to be worked out as we go along. A consequence of building a new instrument this large is the inability to define all subsystems completely ahead of time. Some things will only be discovered when we have hardware on the ground and are trying to make things work. This means, I believe, that the central control algorithms at least, need to be written by people within the Array team. By the time we have specified the requirements of the fringe tracker to a free-lance programmer we could have written it ourselves. Besides, we will definitely change our minds about what exactly it should do, and making these changes can be very expensive unless we write the code ourselves. The conflicting argument is based on the time this requires and the long list of other things for us to do. We have limited manpower as it is; maybe we can not afford to spend too much time developing software. There is, with any luck, a happy medium here somewhere.