Pointing Control Software for the Submillimeter Array Antennas

Nimesh A. Patel

Harvard-Smithsonian Center for Astrophysics
60 Garden Street, MS 78
Cambridge, MA 02138, USA

ABSTRACT

We present the current status of the antenna control software for the Submillimeter array. This software is responsible for pointing and tracking astronomical sources and for antenna pointing calibration. We describe the various stages of the calculations, starting with source-lookup from catalogs and resulting in antenna coordinates commanded to the servo computers. We also present some preliminary results on the pointing calibration of the antenna mounts using an optical guide-scope.

The control software is distributed among several computers connected by Ethernet and an optic-fiber network known as Reflective Memory. The user interaction is through UNIX-style commands. The scripting language, Perl, is used for automating various tasks while individual commands are directly issued through a shell. We summarize the plans for further work on the control software.

Keywords: Pointing, Control software, Antennas, Submillimeter Array, SMA

1. INTRODUCTION

The Submillimeter Array (SMA) consists of 8 antennas of 6 meter diameter each, currently being deployed on the summit of Mauna Kea. The characteristics of the SMA are presented in an earlier paper by Moran (1998)[1]. The first two antennas have been in operation since the Fall of 1999. Interferometric fringes were first obtained on 30th September 1999, at a frequency of 230 GHz on Mauna Kea. Holography measurements of the surface of the antennas show an rms surface error of about 30 μm [2]. The goal is to achieve 12 μm rms surface error. Preliminary pointing measurements show a blind pointing accuracy of about 2" rms in both axes. The required pointing accuracy is 1" rms, for operation at the highest frequency of about 900 GHz. Pointing calibration measurements are currently underway, to characterize the long term stability of the pointing model.

In this paper, we describe the portion of the software that is responsible for the pointing and tracking control of the SMA antennas. We also present the status of the control commands system known as SMAsh (SMA shell) and its planned enhancements.

2. HARDWARE ARCHITECTURE

The implementation of the antenna control system involves several different computers as shown schematically in Figure 1. The various variables that are input/output are indicated along with the type of connection between the computers. The user interacts with the array through a console UNIX workstation. This computer also provides the gateway to the system over the internet for remote operations of the array. The console executes a monitoring program for the real-time status monitoring of the array, running a client program which receives data from the Central Computer over the Ethernet. The Central Computer receives these monitoring data from the antenna computers, using Reflective Memory (discussed further in section 4.1).

The GPS clock is a VME board (TrueTime VME-SG) which is common for the whole array, passing time encoded in IRIG-B over optic-fibers to the antenna pads and the correlator computer. The Central Computer reads the clock through a device driver and writes the time to Reflective Memory for time-stamping of various monitoring variables. There are 2 inclinometers in each antenna, both mounted close to the azimuth axis of the antenna, near the azimuth encoder. One of them is mounted parallel to the elevation axis and the other perpendicular to it. The inclinometers

---

SMA Technical Memo No. 139

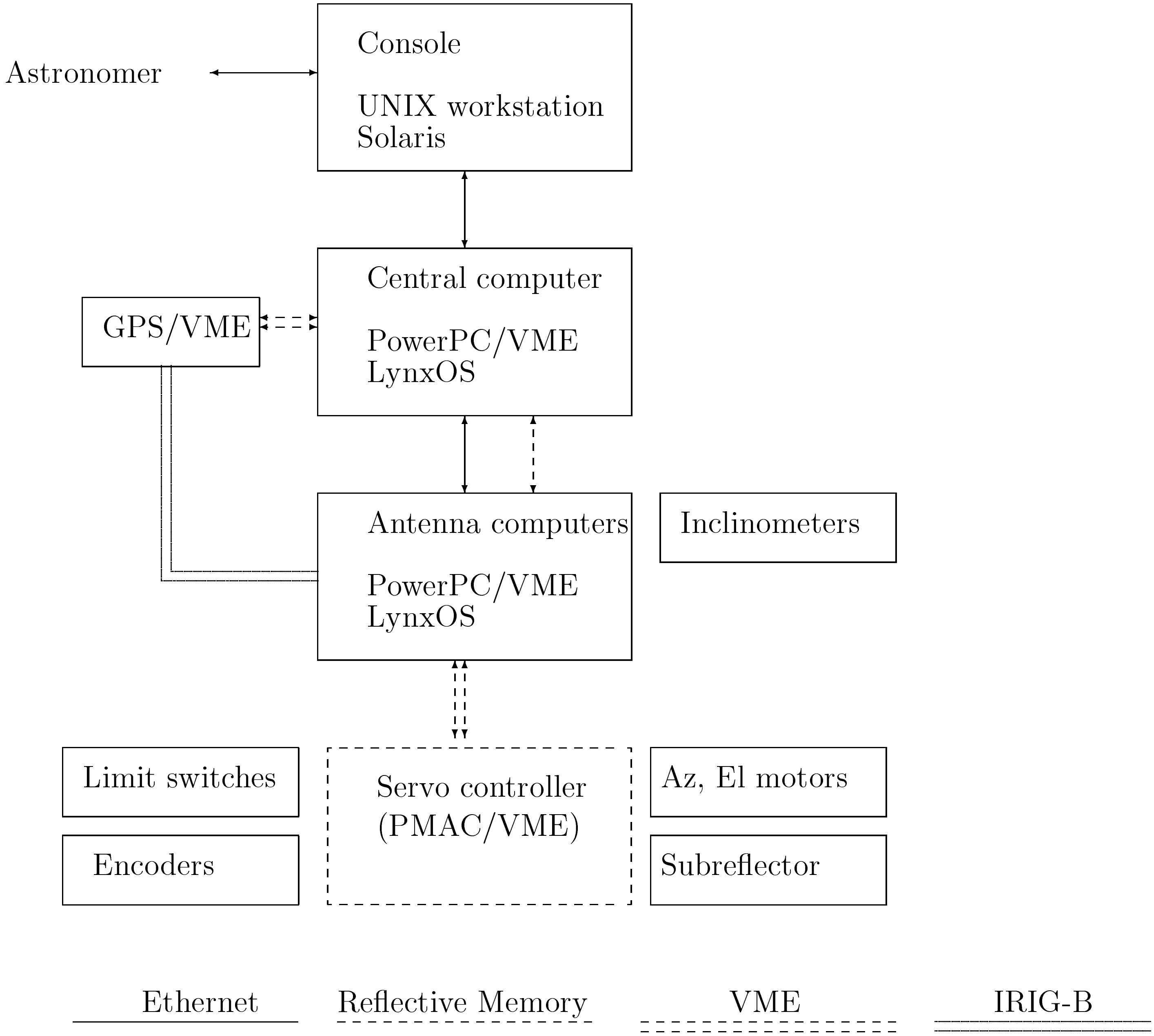

Figure 1. Hardware network for pointing calculations and control.

are made by Applied Geomechanics (model 755). There is provision in the software for real-time corrections in pointing due to the azimuth axis tilt. At present the azimuth axis tilt is measured by moving the antenna in azimuth and recording the inclinometer readings. A sinusoidal function of azimuth, up to a third harmonic, is fitted to these data. Some of these results are tabulated in Table 1, which shows good repeatability over a period of few months, and a diurnal variation of about 7" in the magnitude of the tilt.

The Antenna computer reads the inclinometers and writes the values on Reflective Memory for monitoring by the Central computer. In the first two antennas, the servo controller is a Programmable Multi-Axis Controller (PMAC), made by Delta-Tau Data Systems Inc. The PMAC will be replaced by the SMA Servo Controller Board[*] The PMAC will continue to be used as a servo controller for the axes of the subreflector. The PMAC receives the azimuth and elevation values from the Antenna computer every 0.1 seconds. The PMAC implements a PID servo filter for motion control on the azimuth and elevation axes. In the new servo control replacing the PMAC, the time-stamped azimuth and elevation values along with their rates, will be passed to the servo program through shared memory once every second. Commands to the subreflector will then be issued to the PMAC through a serial communication.

3. CALCULATIONS

3.1. Required computational accuracy

The general goal is to have the computational errors to be much smaller than any physical source of error relevant for pointing. For single-dish pointing, at the highest frequency of 900 GHz, the FWHM beam size will be $11''$. Thus, the positional error due to calculations should be $\leq 0.''1$. This translates to a relative timing error of $\leq$ 7msec or a relative position error (in say the antenna pad location) ≤ 3 m. This error is only a relative error since any fixed error would get absorbed in the pointing model. For interferometry, any errors in the phase and delay calculations must be much smaller than the actual phase error. Thus, for the highest frequency of operation, we would like to have a relative phase error due to miscalculation to correspond to a path length of $\sim 3\mu$m. For the largest baseline of $\sim$ 500 m, this corresponds to an angular error of about 1 mas, or a relative timing accuracy of about 80 μs. Again, this is to be considered as a relative error (a "jitter" in the pointing calculation, and not a fixed error).

Given these numbers for accuracies, it follows that we will have to correct for the x and y terms of the Polar wobble (the magnitude of which can be as large as $0.''3$ [3]), in addition to the usual correction of DUT in the time read from the GPS clock. The gravitational deflection of light due to the Sun must be corrected for even for sources several degrees away from the Sun (even 90° away from the Sun, the deflection is about 4 mas). Any user offset positions to the input Right Ascension and Declination of the observed source, must be added to the input coordinates before any transformations take place. All variables in the pointing calculations will be declared as doubles in the C codes. The computation of the apparent positions may be done every 30 seconds, and linearly interpolated between this time interval. The antenna azimuth and elevation calculations can be updated once every second.

3.2. Calculations of the Apparent Positions

Figure 2 is a flow-chart of the calculations of the Apparent Positions of the astronomical sources. The calculation flow closely follows the prescription given in the Explanatory Supplement to the Almanac (see sections 3.31 and 3.32 of [3]), and in a paper by Kaplan (1990) [4]. The calculations for various transformations of the coordinates such as the precession and nutation of the Earth's axis, are implemented using the Naval Observatory Vector Astrometry (NOVAS) package, which is available now in C [5]. The user-input source name is parsed to determine whether it is a solar-system object or a "star". If it is a solar-system object, its coordinates are obtained from the *DE405* ephemeris. (This is a new ephemeris referred to the International Celestial Reference System [6]). These coordinates are incremented by any user requested offsets. If the object is a star, its coordinates are looked-up from the internal standard catalog. If its coordinates are in the B1950 epoch, they are precessed to J2000. The NOVAS subroutine `topo_star` (or `topo_planet`) is used to calculate the apparent position. For a star, the proper-motion correction is applied, but the parallax is neglected. The gravitational deflection of light due to the Sun is corrected for and the the annual and diurnal aberration corrections are applied followed by precession and nutation corrections. The resulting apparent coordinates are updated in Reflective Memory for passing them on to the other programs for phase and delay calculations. This ends the first part of the calculation, which is done within a loop repeating every 30 seconds. In the second part, the apparent coordinates are converted to the local terrestrial antenna coordinates,

---

[*] developed in-house by Robert Wilson, Robert Kimberk, Steve Leiker and Todd Hunter.

Table 1. Inclinometer readings

	14 September 1999 (Day) 238.4 @ 264.°9	14 September 1999 (Night) 230.8 @ 268.°2	17 January 2000 (Day) 238.0 @ 267.°0
DC	46.5	43.9	50.3
Sin(az)	-237.5	-230.7	-237.7
Cos(az)	-21.0	-7.2	-12.8
Sin(2az)	-2.2	-3.5	-2.4
Cos(2az)	2.7	2.7	2.85

azimuth and elevation, at each antenna. This loop executes every 0.1 seconds (temporarily, faster for the PMAC servo, since there is no time-synchronization when passing these coordinates to the PMAC). Any user applied offsets in azimuth and elevation are read from Reflective Memory and added at this point. Atmospheric refraction correction is applied from weather parameters passed through Reflective Memory. The pointing model correction (as described in the next section) and the inclinometer readings are applied to obtain the azimuth and elevation values which are communicated to the PMAC through its Dual-Ported RAM.

3.3. Pointing calibration

The mount errors are characterized by a standard altitude-azimuth pointing model in which the pointing offsets in azimuth and elevation are given by

$$\Delta Az\ cos(El) = Az_0\ cos(El) + c + l\ sin(El) + I\ sin(El)\ sin(Az - Az_t) \tag{1}$$

$$\Delta El = El_0 + s\ cos(El) + I\ cos(Az - Az_t) \tag{2}$$

in which, Az_0 and El_0 are the encoder DC errors, l is the elevation axis tilt (non-orthogonality w.r.t. Azimuth axis), s is the elevation sag due to gravity, I is the magnitude of azimuth axis tilt and Az_t is the direction towards which it is tilted.

An optical guide-scope is mounted on the telescope's back-up structure, with the objective looking through a hole in one of the panels of the primary dish. The objective is an achromatic doublet lens of diameter 75mm and focal length 75 cm. The image is acquired using a CCD camera of 652x494 pixels, with a resolution 2” per pixel (EDC 1000N, manufactured by Electrim Corp.).

Optical pointing is planned to be carried out for each antenna, roughly once a month, and every time the antenna is relocated to a different pad. A Perl script, `azelstar` selects stars from the Hipparcos catalogue, in the visual magnitude range of 4 to 8, (with additional optional constraints on spectral-types), and sorts them for optimal slewing to complete the sky coverage. Each star-pointing offsets measurement takes about 1 minute, including telescope slewing time (with slew speeds of about 2 °/s and 1 °/s in azimuth and elevation, respectively). Thus, in a few hours, a nearly complete sky coverage is obtained to derive a pointing model. The optical pointing data are useful in debugging an error in timing or in the position of a given antenna pad since we can compare the azimuth axis tilt with a measured tilt using the inclinometers. Any east-west component of a discrepancy in the values of tilt is indicative of a time error (with 1 second corresponding to 15” of tilt error) or a corresponding position error along east-west. Similarly, a north-south component of the discrepant tilt would indicate an error in longitude. After obtaining reliable values of the azimuth and elevation axis tilts, these values help constrain the model in subsequent radio pointing measurements which typically have poorer sky coverage. The parameters in common between the optical and radio pointing models are the azimuth and elevation axes tilts.

Presently, the radio pointing data are obtained from continuum raster scans made across planets using the chopping subreflector to improve the signal-to-noise ratio. A good elevation coverage is required, to determine the collimation and sag terms reliably. Table 2 lists the coefficients obtained for a sample run of radio pointing on Antenna-2 on Pad-1 on 12 September 1999, using Mars, Jupiter and Saturn with only night-time data. Apart from the second-order terms [sin(2az), cos(2az)] in the azimuth axis tilt, no other departure from a standard alt-az mount model is required. Software to measure pointing offsets in spectral-line mode is currently under development.

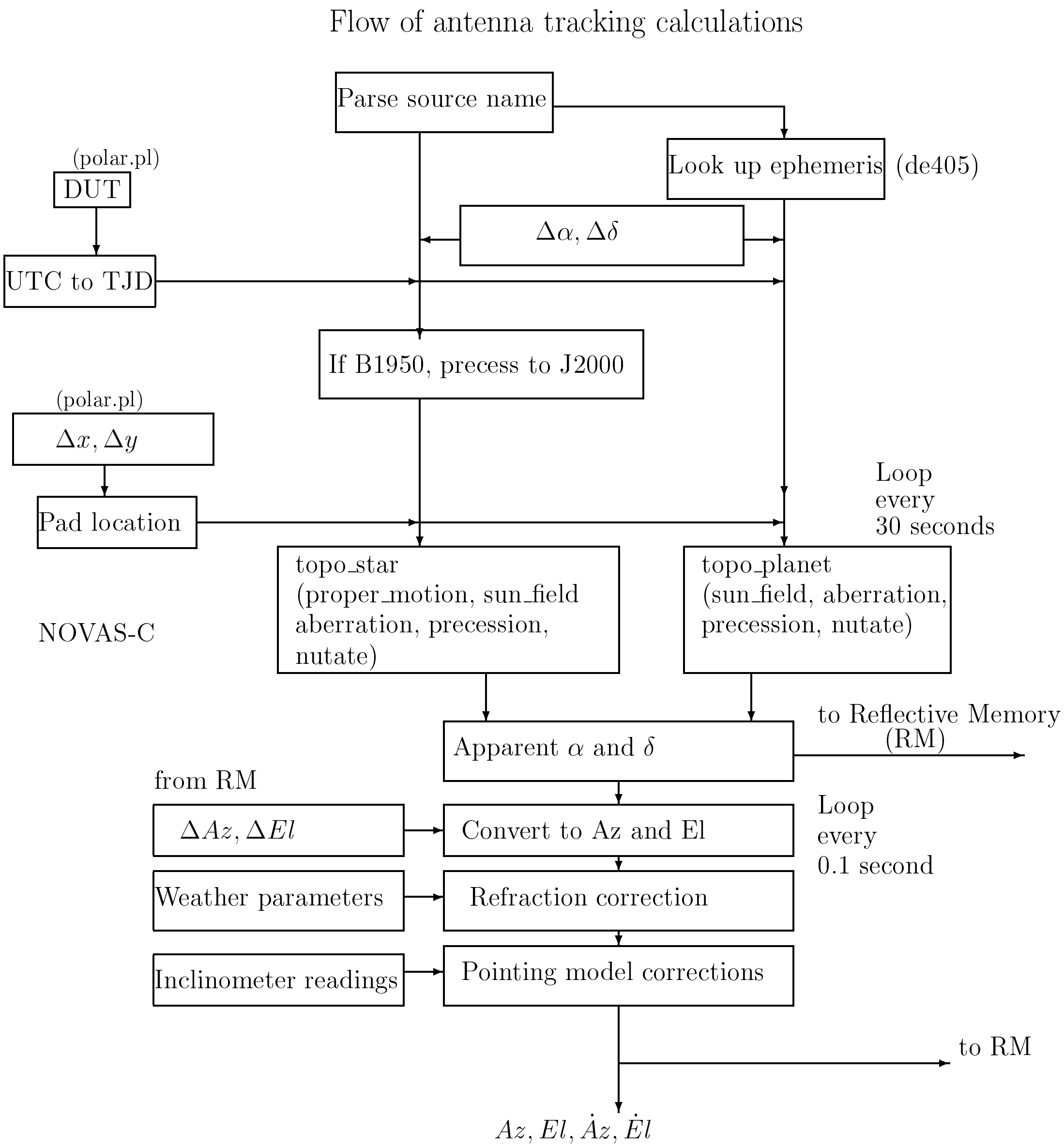

Figure 2. Flow-chart for position calculations

Table 2. Pointing model results

Results of az fit	(rms=$1\farcs28$)	Results of el fit:	(rms=$2\farcs18$)
Az DC	-1040.34±2.01	El DC	-1036.63±0.86
Az Collimation	-102.92±2.70	El Sag	0.00±0.47
El axis tilt	33.77±2.28		
Az axis tilt (sin(az))	1.16±0.36		-0.07±1.27
Az axis tilt (cos(az))	-230.09±1.89		-221.85±0.33
Az axis tilt (sin(2az))	0.90±0.42		2.92±0.66
Az axis tilt (cos(2az))	-4.63±0.86		-2.75±0.40

Table 3. SMAsh commands

`observe`	`integrate`	`integrationTime`
`offsetUnit`	`azoff`	`eloff`
`az`	`el`	`azel`
`chopperX`	`chopperY`	`chopperZ`
`chopperTilt`	`startChopping`	`dip`
`homeChopper`	`stowChopper`	`openM3`
`closeM3`	`killMotors`	`snapshot`
`standby`	`resume`	`azscan`
`elscan`	`tsys`	`tilt`
`startPS`	`stopPS`	`stopChopping`
`value`	`stow`	`shutdown`

4. COMMANDS AND AUTOMATION

The control commands for the array are executed in the Central computer running LynxOS. The commands have a UNIX style syntax and command line arguments. All the low-level commands are written in C. Some of the commands are implemented as Perl scripts. Table 3 lists the commands implemented so far.

Examples showing the command-syntax:

```
observe --source crl618
observe -a 2 -s saturn
```

The `observe` command writes the source name into Reflective Memory and sends an interrupt to the antenna computers to issue a "new source" command. Most of the commands listed in Table 1 are relevant for single-dish operations, and they have been used to carry out the two-element interferometer calibration tasks so far. More commands towards the full interferometer operations are in preparation.

A useful construct to access the value of a Reflective Memory variable, is provided by the `value` command. If for example, one needs to know the elevation of antenna-2, the following command returns this into a variable `$elevation_antenna2`, which may be used subsequently in a Perl script for conditional execution of other commands.

```
$elevation_antenna2 = `value -a 2 -v actual_el_deg`;
```

The commands are communicated from the Central computer to the Antenna computers as ASCII strings followed by an interrupt through Reflective Memory. On receiving this interrupt, the `track` program (running on the antenna computer) reads this string and carries out the commanded operation.

Some of the commands require communication with computers which do not have access to Reflective Memory, such as the computers that set the local oscillator frequency or the ones that acquire and archive the astronomical data. These computers will be running Remote Procedure Call (RPC) servers for the SMAsh commands. An example of such a command is `observe` which informs `track` to start tracking an astronomical object, and also informs the other computers for looking up the new apparent coordinates and other antenna related information for appending to the headers of the data files, via RPC client calls [8].

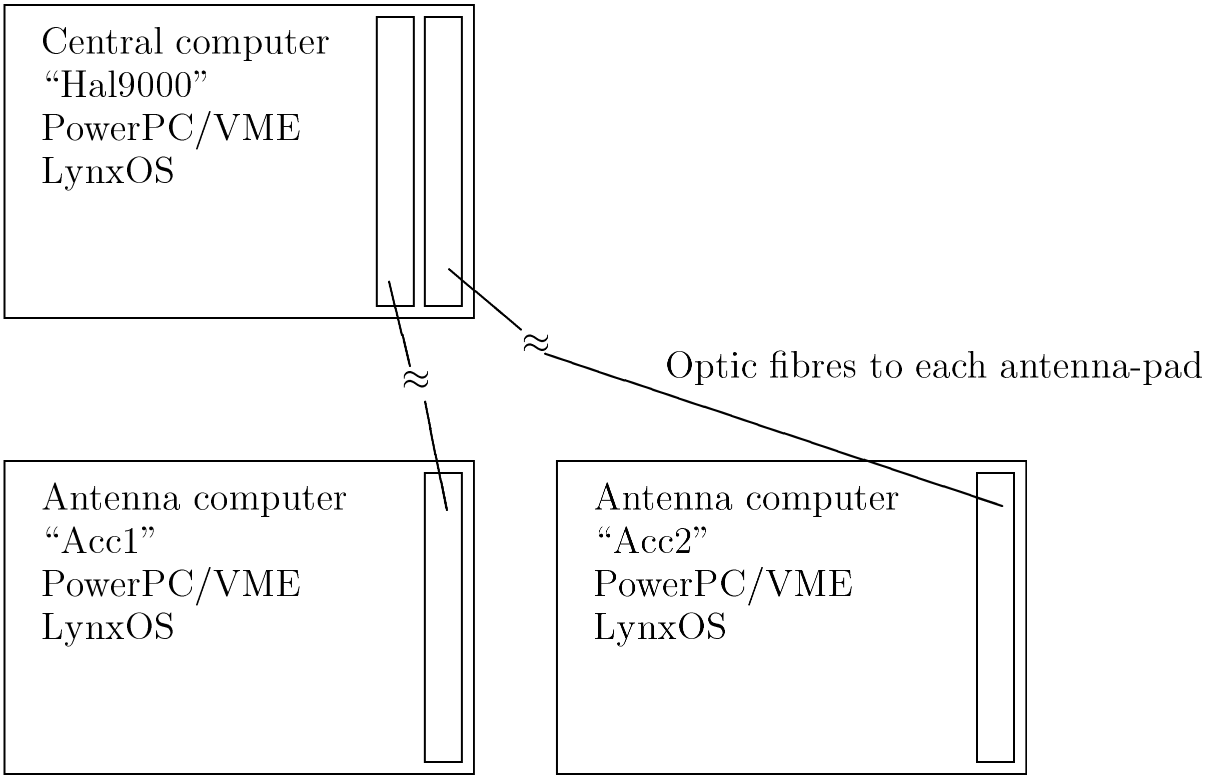

Figure 3.

4.1. Reflective Memory

The Central computer and the Antenna computers each have a VME Reflective Memory card (manufactured by VMIC, model number VME-5576)[9]. These cards are connected to each other by optic fibres, using the geometry shown in Figure 3. According to the manufacturer's specifications, the data written by one computer on this network, can become available to another computer within a time interval of a few microseconds, without an unpredictable network delay as in the case of Ethernet. The optic fibre can be as long as 2 kilometers, easily including the largest baseline for the SMA (even after joining with CSO and JCMT).

The Reflective Memory network is shown in Figure 3. There is a 256 KB buffer of real-time antenna monitoring data which is updated by the `track` program in the antenna computers every 0.1 seconds. For on-the-fly mapping purpose, faster sampling of the antenna axis encoders is required. The programs `encoderServer` running on the Central computer and the `encoderClient` running on the antenna computers. Using these programs we can do on-the-fly mapping for holography and continuum observations on the beacon as well as celestial sources. With the antenna moving at the rate of 300" per second, there was negligible delay in the data acquisition.

5. SUMMARY OF CURRENT STATUS AND PLANS

The initial implementation of the SMA pointing control software is able to provide the required functionalities towards carrying out various antenna and interferometer tests such as pointing calibration, holography and baseline determination. A command scheduler is currently in preparation, which will run on the central computer, and which will have the knowledge of various system states, via Reflective Memory, shared memory and disk files (for persistence). Such a scheduler will also interact with the Diagnostics and Error Reporting Software being developed by Zhang [10]. We are exploring the possibility of adopting the scheduler being used by the OVRO system. Tools for generating observing scripts using a GUI are also under development.

In summary, the pointing control software for the SMA antennas is operational. Several basic commands are now in operation through a standard UNIX shell and automation is achieved using Perl scripts. Reflective Memory allows fast monitoring of variables including applications such as on-the-fly mapping for holography and mapping in continuum mode. Software for spectral-line pointing calibration in single-dish and interferometric mode is currently in progress. An OVRO-style java monitoring program is being implemented by our collaborators at ASIAA in Taiwan.

ACKNOWLEDGMENTS

We are grateful to Steve Scott (OVRO) for codes and advice on various control and monitoring software. We thank Bob Wilson, Ken "Taco" Young and Eric Keto for helpful discussions. The phase and delay part of the calculations for the interferometer are carried out by programs written by Taco. Charlie Katz and Paul Peterson wrote the VME-LynxOS device drivers for the Reflective Memory [9] and true-time cards, respectively. Antony Stark wrote the software for data acquisition for the CCD camera used for optical pointing. We thank T. K. Sridharan, Masao Saito, Brad Wallace, Hiroko Shinnaga, Nagayoshi Ohashi and Ken Young for their help in carrying out pointing and tiltmeter measurements and finding bugs in the software.

REFERENCES

1. Moran, J. M., 1998, SPIE Conference on Advanced Technology MMW, Radio, and Terahertz Telescopes, Kona, Hawaii, March 1998, Editor: Phillips, T. G., SPIE Vol., 3357, p208.

2. Sridharan, T. K., Saito, M., Wiedner, M., Patel, N. A., 2000, SMA Technical Memo (in preparation).

3. Explanatory Supplement to the Astronomical Almanac, Editor: Seidelmann, P. K., University Science Books, Mill Valley, California, 1992.

4. Kaplan, G. H., Hughes, J. A., Seidelmann, P. K., Smith, C. A., 1989, Astronomical Journal, 97, 1197

5. http://aa.usno.navy.mil/AA/software/novas/novas_c/novasc_info.html

6. Arias E. F., Charlot P., Feissel M., Lestrade J. F., 1995, Astronomy & Astrophysics, 303,604
   Standish, E. M. Jr., 1982, Astronomy and Astrophysics, 114, 297
   http://ssd.jpl.nasa.gov

7. The Hipparcos Catalogue, 1997, ESA Publications Division, Noordwijk, The Netherlands.

8. Zhao J., SMA Interferometry Data Transfer and Storage, SMA Technical Memo No. 126

9. Katz, C., 2000, Using Reflective Memory, SMA Technical Memo (in preparation)

10. Zhang Q., Design and Initial Implementation of the Diagnostic and Error Reporting System, SMA Technical Memo No. 132