Per-Station Phase Computation Reconsidered

JONATHAN D. ROMNEY
National Radio Astronomy Observatory
Charlottesville, Virginia

1985 March 27

I want to reconsider a suggestion by Martin Ewing in VLBA Correlator Memo VC032. After trying for some time to arrive at a satisfactory detailed arrangement for computing the phase and delay models in the correlator, I have concluded, reluctantly, that the scheme we've been considering has several serious flaws. I say "reluctantly", because I do appreciate the virtues of using the microprocessors and the communications structure necessary for accumulating, filtering, and sorting the cross-correlated signals for the additional purpose of computing and transmitting the model parameters. In this memorandum I propose an alternative that also uses an existing signal path — the distribution network which routes the reproduced and delayed data and validity signals to the cross-correlation modules.

MODEL CALCULATIONS IN THE TREE

It's appropriate to start by elaborating on what's wrong with the current "tree" scheme, and to do so I need to refer to a particular version of the scheme. The particulars are unfortunately still vague (and not least so because of the effort to optimize the tree structure for two unrelated purposes). So I've had to choose fairly arbitrarily the following heuristic case, which illustrates my points but is general enough that the conclusions drawn apply to other variants as well. This description mentions only the tasks relevant to model computations, so some of the elements may seem a bit mysterious. Only the *phase* model is considered, for the moment. The tree extends over six recognizable levels:

Control computer. This is a VAX-11 system whose mission is overall control of the correlator system. Its involvement in the model algorithms is restricted to assembling and transmitting the appropriate top-level parameters when a new correlation scan begins.

MIOP0. Basically a communications node, this 68000 processor simply passes the parameters from the control computer to all the MIOP1's.

MIOP1. There are 8 of these 68000's linked to the single MIOP0 interface, each responsible for 2 of the correlator's 16 input channels. At each major time step (to be defined shortly), the top-level parameters from the control computer are converted to a brief series of scaled phase derivatives valid until the next step. A third-order scheme currently seems attractive, for which the major time step would be about 10 seconds. (This scheme is assumed hereafter.)

Filter. (The name has not yet been bidirectionalized.) There are 8 of these "Digital Signal Processors" (DSP's), probably TI TMS320's, per MIOP1. Their function in the model computation is simply a cascade addition of the current derivative series to produce output values for a *second*-order series at each secondary time step of 500 milliseconds.

Accumulator. At another level of fanout by 8, these DSP's perform another cascade addition at a 4 millisecond time step, yielding a *first*-order series for the VLSI correlator. Unfortunately, a rather longer time step — 32 microseconds — appears to be optimal for the counterflowing correlated data.

VLSI Correlator. Each accumulator feeds 7 of these 16-lag correlator elements. The first-order series is applied at the 0.5 microsecond phase-update rate to drive the on-chip fringe rotator.

The preceding sketch has focussed on the structure of the tree and the timing and precision relationships for the phase model, and in so doing has ignored two important but unpleasant features. First, only the phase model has been described. We have been aware that a separate branch from MIOP0 would be needed for the delay model, and have usually represented this as a MIOP1 transmitting a delay series to the Data Playback System (DPS, not to be confused with DSP). In fact, of course, to be commensurate with the phase model, the delay branch would have to continue to the Filter and Accumulator levels, since a delay update period of about 4 milliseconds is needed. These latter components would be located, presumably, within the DPS; this imposes some difficulty in ensuring synchronization of the DPS delay steps with the correlator's phase computations, and at a minimum would require that a 250 Hz clock (which we have not provided for) be transmitted through the Correlator-DPS interface.

However, that's only half the delay story. The VLSI correlator must control two "vernier delay" bits which compensate, essentially, for the loss of precision in subtracting the two station delays which have a fixed one-sample-time resolution. It must also apply a compensating phase shift in the fringe rotators when a vernier bit is switched in or out. To accomplish these vital functions, ***the tree must carry a complete delay model computation in parallel with the phase model*** in order to reproduce the delay steps in the DPS. Again, this computation chain must extend to the Accumulator level.

A second unpleasant aspect of the tree structure not exhibited in my brief description is the depth and complexity in the stations/baselines dimension. All the calculations described must be done either for 20 stations or 190 baselines; it is not clear where the differencing of station quantities to form baseline values should occur. This operation is best deferred as long as possible, of course, but can also be supported best in a 68000 — *i.e.*, in the MIOP1's — because of the conditional logic required. At the other end, it should occur before the Accumulator level, since one of these units deals with between 1 and 7 baselines. At the intermediate Filter level, we are likely to have to carry a large number of baselines to accomodate the several modes of correlator channels *vs.* resolution.

Considering both of these areas, we are faced with an enormous parallelism in the model algorithms throughout the tree. ***Exactly*** identical delay computations occur in all 9 MIOP1 branches (*i.e.*, 8 in parallel with the phase model for the vernier delays, and 1

"real" delay branch for the DPS). For high resolution correlation, where the same observed channel is delivered to several correlator channels, exactly identical phase calculations will also occur in the corresponding MIOP1 branches — and even for 16-channel observations, all the phase terms differ only by constant factors among branches.

The culprit in this redundancy, and in the awkwardness of the DPS delay calculations, is the tree structure. The values we need in one module are available right next door, but are not accessible without traversing the tree back to the root and then down another branch. In general, the model calculations are much less suited to this kind of structure than are the correlated data, which are concentrated as they flow up the tree. The model computations do not expand significantly going downward once the differencing to baselines is accomplished; to first order, we need to do fewer additions more frequently — but are forced into further expansion to parallel calculations.

In summary, using the tree for the model computations appears to have been a good idea that didn't work out. The number of levels in the tree has grown with the needs of the post-correlation "fringe processing" beyond what is useful for the models, and it has not proved possible so far to arrive at a timing scheme commensurate with both operations. I must admit to being charmed by the concept of more frequent but lower order computations at each successive level of the tree, but this baroque structure is not at all necessary to accomplish the purpose, and does evoke some queasiness as to accountability. The overall channel-by-channel architecture out of which the tree arises enforces an inefficient and unnecessary redundant parallelism in the model calculations. Finally, the existence of an isolated but equivalent DPS delay-computation chain in a rather different environment, which must parallel the results in the 8 tree branches, represents a major departure from good programming practice.

PER-STATION PHASE

In VLBA Correlator Memo VC032, Martin Ewing suggested the use of "per-station phase". This term refers to *calculating* a station-based phase rotation, just as is done for delay. For a variety of reasons, however, it is undesirable to *apply* this rotation before correlation. (See capsule descriptions in VC032, and lengthy related discussions by participants in last summer's debate on fringe rotation at the array elements.) Ewing pointed out that if pairs of these station phases could be transmitted to the appropriate baseline correlation module, they could be differenced there to derive an absolute baseline phase rotation. He also noted that the station phases would have to be switched and routed in the same manner as the associated data streams, but did not specify this further (there being then no established signal-distribution scheme).

I propose here a specific implementation of this approach, which I believe simplifies the model-computation task of the correlator *as well as* the post-correlation fringe-processing hardware, and eliminates the problems associated with routing the model parameters along the tree structure. This scheme was developed in conversations with Dave Fort, and is reminiscent of a suggestion originally made to me by Benno Rayhrer some time ago.

We currently receive 3 16-Mbit/sec data streams from the DPS for each channel of each station (2 sample bits and 1 validity bit). These streams are switched on a channel-by-channel basis to the 16 20-station "correlator arrays" via a 24×16 - to - 16×20 crosspoint

switch. In the correlator arrays the streams are routed on fixed paths to the appropriate VLSI correlator element. My proposal requires expanding these data lines and the switch by a third again, to provide a fourth stream I'll call, for obvious reasons, the *model stream.* Note that there is then one such stream for each channel/station combination.

The model stream carries serialized 8-bit data bytes at 2 Mbyte/sec. Each byte consists of a start bit, 4 station phase bits, 2 station fractional-bit delay bits, and a parity bit. The station phase is computed by an as yet unspecified station phase generator (SPG) at 0.5 microsecond intervals; possibly the JPL GPS fringe-phase generator chip could be used for this purpose. The SPG is driven by a single 68000 microprocessor, which should be able to compute a 32-bit third-order scaled derivative cascade, and transmit 2 16-bit constants for the SPG's first-order algorithm, for each of 320 channel/station combinations at 4 millisecond intervals. Note that the station phase is carried to a precision of $\pm\frac{1}{32}$ turn.

The fractional-bit delay — the remainder after truncating the station delay at the sample-interval delay resolution — need be updated much less frequently than the phase. A second 68000 should be able to do a *complete* delay calculation for each of 20 stations at 1 millisecond intervals; the integral part is then transmitted to the DPS (this aspect is discussed more thoroughly below), and the 2 leading bits of the fraction applied to the model line (even though these bits will then only change every 2000 bytes). This expresses the fractional-bit delay to a precision of $\pm\frac{1}{8}$ sample interval. It's a slight oversimplification to assume only 20 delay models are needed, since we may need to support two or more different sources (or even telescopes) on one "station" input for special purposes. Allowing for 40 delay models is probably adequate; certainly there is no need to match the full generality of 320 phase models.

These station phase/fractional-delay bytes are routed to the appropriate VLSI correlator element as described earlier. There the byte is decoded every 0.5 microsecond, and the phases and fractional delays are differenced to determine a baseline phase to be applied to the fringe rotator (with $\pm\frac{1}{16}$ turn precision) and a baseline fractional delay ($\pm\frac{1}{4}$ sample interval) for setting the vernier delay bits. This achieves a saving of several hundred gates on each chip (Dave Fort estimates); the 16-bit adder and registers of the former fringe-phase generator are replaced by a 4-bit adder and registers, and similarly for the vernier delay logic. Only the fringe rotator itself remains unchanged.

A major consequence of this alternate method of routing model parameters to the VLSI correlator is, of course, that the tree structure is bypassed completely. The communication links along the branches then need only be supported in one direction, at a substantial saving in hard- and software, and the tree structure and timing can be optimized properly to support the fringe-processing task. In fact, since the MIOP1's have no remaining function, it should then be possible to abolish the MIOP1 level and link the 64 Filter outputs directly to MIOP0, and also to downgrade the Accumulator unit to a less intelligent hardware adder, because the full DSP capabilities are needed only for model computations at this level. The proposed model stream could also carry infrequent correlator configuration commands if necessary, by providing a system-wide setup signal changing the interpretation of the model bytes; the existing link from the two 68000's used for model calculations to the model stream inputs facilitates such a mode.

CORRELATOR/DPS INTERFACE CONSIDERATIONS

The foregoing discussion has not considered several aspects of the interface to the DPS. Current specifications for this interface call for the delay to be implemented in the DPS, with the delay calculation split between the correlator, where we calculate series coefficients of unspecified type and number, and the DPS, where the series is evaluated for frequent updates. This separation is motivated, I think, by the attitude that regards the interface as the boundary to the recording-technology dependent portion of the processor. Sensible as this may be, it does impose an extremely artificial division on the software designer. The need to synchronize the parallel delay calculations on either side of the interface poses a hazard to accountabilty, and the awkward structure forces the residual-delay computations along the tree branches to follow the full absolute delay algorithm used in the DPS side to avoid introducing a second delay routine, even though only the residual is needed.

To achieve the full benefit of the per-station phase scheme, we must bring the phase and delay computations together in one logically localized area. This means either transferring the entire delay algorithm to the correlator side of the interface, or doing both delay and phase computations in the DPS. The former alternative is most logical from a software standpoint, where the core of the system resides in the correlator control computer, and the delay and phase algorithms — with, if necessary, intermediate results — are directly available for model-accountability entries in the archive data. Direct transmission of the full calculated delays through the interface to the DPS would require about 20 bits per station, at a 1 kHz rate, or 400 kbit/sec on a single line for all stations. As far as I can determine, the control path through the interface has never been specified beyond the fact of its existence, but these rates are well within the capacity of, *e.g.*, an Ethernet channel. If we were pressed we could instead transmit the full 20 bits only every 10 seconds or so and send only the low-order 10 bits in between.

The second arrangement seems quite odd — a correlator "embassy" in DPS territory — but does have some advantages too. It isolates all station-based hardware on one side of the interface, and limits the data rate on the link through the interface between the control computer and the phase-and-delay module in the DPS to occassional top-level parameter transfers. But the total traffic through the interface would increase, since the fourth set of 16-Mbit/sec lines would now have to be carried.

Finally, we could even go one step beyond the first alternative above and bring both the delay computation and the *implementation* back to the correlator side of the interface. This amounts to reinstating the "station electronics" module in the correlator, and represents a not insignificant additional burden for the correlator project, although globally it is (partially) offset by a reduction on the DPS side. This scheme offers the advantages of providing the optimum environment for the model software, and of minimizing both the data and the control traffic through the interface. In this variant the station electronics presumably would include an autocorrelation module as well, probably using the standard VLSI correlator for simplicity; the total number of VLSI correlators would not change, but each channel-based correlator array would then require about 10% fewer, and the signal routing would be simplified.

COMPARISON AND SUMMARY

The proposed per-station phase and residual-delay approach using the correlator's signal-distribution network appears to offer major improvements over the current scheme based on the fringe-processing tree. Among these are the following:

Software simplicity. The software structure is enormously simplified. The current scheme (in the particular version described earlier) would have to deal with the control computer, 10 68000's at three different levels, and more than 576 DSP's at two further levels. Each of these five levels would require different programming and timing, and the vertical structuring would impose a multiplicity of parallel and redundant computations with no other rationale than the need to fit into an inappropriate communications heirarchy. An otherwise unnecessary synchronization scheme would have to be developed to manage the separated delay and residual-delay computations. In contrast, the per-station scheme retains only the control computer and 3 68000's at two levels. The software is unified in a single environment, the computations can be limited to those actually necessary, and can be optimized without having to fit into an externally determined structure.

Hardware simplicity. The fringe phase generators in all 3520 VLSI correlators can be eliminated, and replaced functionally with fewer than 360 specialized generators — a reduction by an order of magnitude. Added to the VLSI correlator are some short registers and adders with associated control logic, but a net savings in gates is to be expected. The 512 Accumulator DSP's can be downgraded, and the entire level of 8 MIOP1's eliminated. Communications links throughout the tree can be simplified from bi- to unidirectional. Additional requirements in the crosspoint switch and signal distribution involve only expansion from 3 to 4 "planes". And the fringe-processing tree is freed for optimization without being required to coordinate with the model computations.

Accountability. The many levels in the current scheme, each with its own timing — and executing in differing processors — would be difficult to validate and debug. Intermediate results would not be easily available for inclusion in the archive. The proposed alternative allows the model coding to be as straightforward and transparent as the programmers can make it, and lends itself to easy testing, either in place or in stand-alone operation.

Interface. If we can see our way clear to reassume the responsibility for implementing the station delay, we can also simplify the interface to the DPS and eliminate the need for synchronizing the two parts of the delay computation across the interface.

I've tried to emphasize in this memorandum what I regard as unacceptable disadvantages, on the software side, of computing model phase and delay along the branches of the fringe-processing tree. The alternative suggested seems to me a logical and straightforward solution which escapes these disadvantages and introduces as well a number of improvements beyond the software area. The specific implementation using a fourth switch and data-distribution plane is not essential to the solution, and we will have to consider other possibilities before deciding on the per-station scheme. Other issues requiring further discussion are the logical location of the phase and delay modules with respect to the correlator-DPS interface, and whether to implement a station-based subsystem responsible for delay and autocorrelation.