Algorithms for Rapidly Computing the FAME Observing Sequence

George Kaplan

June 15, 2001

1 Introduction

This note outlines one way to attack the problem of determining which stars will be observed by FAME at what time. This involves projecting FAME's focal plane onto a 1.1°-wide circle in the sky as a function of time, determining which stars from the input star catalog are within that field, and relating the coordinates of star images to specific pixels on the CCDs.

This is written primarily for the relatively low-accuracy (~half pixel = 0.1 arcsecond) problem of scheduling the observations by FAME's on-board system, using as few trig functions (=CPU cycles) as possible. It is likely that the same kind of algorithm will be needed on the ground for the "quick look" system. Also, we may have to reconstruct FAME's observing sequence simply to identify which stars were observed at specific times. It is also possible that a first-order approximation to the observation geometry might be useful as part of the data analysis pipeline. I believe the approach outlined below can also be used, iteratively, as the basic framework for more precise computations (comments on that are in Section 4). The emphasis here is on linearizing the computations; with 40 million stars in the input catalog, doing traditional spherical trigonometry on the whole lot is out of the question. Fortunately, we need only deal with a fraction of the catalog at any one time, and spherical trig can be generally be avoided without compromising accuracy.

Reference is made below to equations in a 1989 AJ paper [1] that describes in detail the computation of apparent places of stars from catalog data. Much of the contents of that paper were used for the *Explanatory Supplement to the Astronomical Almanac* [2] Sections 3.31–3.33. Also listed below in several places are the names of the Fortran subroutines in the NOVAS package [3], which implement the algorithms in the 1989 paper. NOVAS is available in both Fortran and C at http://aa.usno.navy.mil/software/novas/. I'm sure there are equivalent SLALIB routines.

It's worth mentioning that a complete catalog-to-apparent-position calculation for stars, carried out in the ICRS frame, can be performed on an observation-by-observation basis by NOVAS subroutine VPSTAR, specifically through its entry point LPSTAR. VPSTAR/LPSTAR calls a series of lower-level NOVAS subroutines to handle individual effects — proper motion, parallax, aberration, etc. LPSTAR is designed for the case of Earth-based observations so the geocentric position and velocity of the observer that it uses would have to be replaced by those of FAME. Using a slightly modified VPSTAR/LPSTAR directly for each FAME observation would impose a significant computational load which might be justified for precise simulations or data analysis but not for the on-board or quick-look system. The accuracy of VPSTAR is several orders of magnitude better than what is needed for these tasks.

In this note I outline how the required calculations can be partitioned and approximated (and in some cases skipped) to yield usable focal-plane coordinates and times of transit across the relevant CCD fiducial lines. In the next section I address the individual geometric and physical effects that must be accounted for. Then, in Section 3, I describe a simple scheme for putting it all together to determine the observing sequence, given a model of FAME's rotation. Efficiency of computation is the primary consideration. Along the way, I make some comments about the differences between quick-and-dirty calculations and the computational approach needed for the much higher accuracy applications of simulation and data analysis.

2 General Considerations

2.1 Position and Proper Motion

An important aspect of the computation scheme outlined in Section 3 involves dividing the sky into boxes, called tiles, of approximately one square degree, each of which would contain on average 1000 stars from the FAME input catalog. We can store the positions of the centers of the boxes as well as those of the individual stars in the form of 3-D vectors. This takes 50% more storage than spherical coordinates, obviously, but has computational advantages since most of the required calculations downstream can be put in a form that just involves dot products. That is, we store, for each star, the vector:

$$\mathbf{p}_0 = \mathbf{p}(t_0) = r \begin{pmatrix} \cos\delta\cos\alpha \\ \cos\delta\sin\alpha \\ \sin\delta \end{pmatrix} \tag{1}$$

where t_0 refers to some convenient time within the FAME mission and the α and δ values are for that epoch, in the ICRS system. This is eqn. (30) of [1]. By storing the catalog this way, we evaluate many of the trig functions up front rather than in real time. Note that you can use the factor r (normally 1) to scale the vector for the star's distance. In NOVAS, r is the star's distance from the Sun in AU (it's done that way so the same routines can be used for stars and solar system objects). Obviously scaling the vector for distance gives us 3-D coordinates and the extra 50% storage is not wasted. The large uncertainty in the distance parameter r for most stars is actually irrelevant because it drops out whenever angles are formed. (When the parallax of a star is not known, NOVAS sets it to 10^{-7} arcsec, a rather arbitrary number.) When $r > 1$, what we're doing is storing a *position vector* rather than a *direction vector*.

However, there is a computational advantage in using direction vectors, that is, keeping r=1: we avoid the need for normalization — involving the square root function — when dot products are taken later. I think that on board FAME, the catalog should be stored this way, even though we will have a corresponding catalog on the ground stored in position-vector form.

The star position vectors will have to be updated for proper motion at intervals which depend on the accuracy requirement. For the on-board system, I assume the catalog will be updated periodically from computations done on the ground. For 0.1 arcsec accuracy, the updates need to be done only every few months, and then for only a small subset of relatively high proper motion stars. Most stars will need no proper motion update during the entire mission. For high-accuracy analysis or simulations, however, the update rate is much higher: a star with 1 arcsec/year proper motion travels 1 μas every 30 seconds. There are only a few of those, of course, and maybe we don't need 1 μas, but for the high-accuracy work, position updates several times per day will clearly be needed for many stars.

A star's position vector can be very rapidly updated for proper motion using a pre-computed space-motion vector as given in eqns. (31)–(33) of [1]. If we call the space motion vector $\dot{\mathbf{p}}$, all we're doing is using $\mathbf{p}(t) = \mathbf{p}_0 + \dot{\mathbf{p}}(t - t_0)$, which automatically takes care of the "foreshortening" of the stellar paths for nearby stars. NOVAS routines VECTRS and PROPMO do everything, using a position vector with components in AU and a space motion vector with components in AU/day. Again, in forming the space motion vector once for each star, we take care of a lot of the trigonometry up front. Nasty spherical-trig effects in proper motion near the celestial poles are totally avoided in the vector approach. However, as I said before, before any recomputed vectors are uploaded to FAME, they should be normalized.

2.2 Binary Star Motion

Note that so far we have taken care of linear space motion only. For the relatively small number of stars that are components of binary systems with known orbits, and if the computational accuracy requires it, the offset of the star from the center of mass of the system should be computed and added in. (There is no NOVAS routine for handling binary orbits, but I'm sure one can be obtained from either the USNO speckle or NPOI groups.) This assumes that the position and proper motion used above do, in fact, refer to the system's center of mass. That will not always be the case; catalog data for binaries represent a notorious hodge-podge of different reference points. For low accuracy applications with a short span of observations there is a shortcut available: for binaries with periods > several times the span of observations, we can simply absorb the orbital arc into the position and proper motion of each component, thus finessing the extra orbital calculation. If we characterize the orbital arc by a center of curvature, from which we measure the arc's radius ρ and angular span $\Delta\theta$, then linearizing the orbital motion will work if

$$\frac{\rho}{2}\,[\,1-\cos(\Delta\theta/2)\,]\;<\;\epsilon \tag{2}$$

where ϵ is the tolerable astrometric error. For example, for the FAME on-board system ($\epsilon \approx 0.1$ arcsec), folding the orbital motion of a star into its position and proper motion will work for an orbital arc of up to 90° if $\rho < 0.7$ arcsec. (For circular orbits seen face-on, ρ is the same as the semimajor axis of the orbit and $\Delta\theta = 2\pi\Delta t/P$, where Δt is the time span of observations and P is the orbital period.)[1]

The point in the calculations where proper and orbital motion should be added in depends on the update frequency. Doing it at the wrong point will result in redundant calculations. For the on-board system, where the update frequency can be a few times per year or less, all necessary updates for the whole input catalog should be performed at once. But for high-accuracy analysis or simulations, where the update frequency might be several times per day or more, it makes sense to perform the updates only on a spiral-by-spiral, or field-by-field, basis.

2.3 Parallax, Light-Bending, and Aberration

Parallax, gravitational light bending, and aberration depend on time and position on the sky, and parallax additionally depends on a star's distance. These corrections should therefore be computed only when the time that a particular field will be observed is known. The equations are given in [1] and are performed (for star position vectors with components in AU) by NOVAS subroutines GEOCEN, SUNFLD, and ABERAT. Of course, for the FAME case, the geocentric position and velocity vectors of an Earth-based observer would be replaced by the corresponding vectors for FAME. More on that below. Another note: at the highest accuracy, SUNFLD and ABERAT may need to be replaced by routines with higher-order relativity terms built in. Certainly at least we need to add the gravitational deflection by the Earth, Jupiter, and Saturn.

[1] I have a technical note about half completed where the equations for proper motion contaminated by binary orbital motion are worked out.

The good news is that for the on-board or quick-look systems, parallax and gravitational light bending can be skipped, and an efficient aberration algorithm is available with better than the needed accuracy.

There are, of course, several hundred or so stars with parallaxes > 0.1 arcsec. The most efficient scheme is simply to handle these as special cases by the on-board system. The on-board catalog could contain a one-bit flag for these stars, which would indicate that their position vectors (rather than direction vectors) were stored. Then if $\mathbf{p}$ is the position vector of the star, and $\mathbf{F}(t)$ is the position vector of FAME with respect to the solar system barycenter (the components of these vectors must be in the same units, e.g., AU) then to correct for parallax we simply form $\mathbf{p}' = \mathbf{p} - \mathbf{F}$ and normalize.

For normalized (direction) vectors, aberration can be applied equally easily, once FAME's velocity vector $\dot{\mathbf{F}}(t)$ is known. Simply compute $\mathbf{p}' = \mathbf{p} + \dot{\mathbf{F}}/c$ and renormalize. Or, to avoid the square root, use the following:

$$\mathbf{p}' = \mathbf{p} + \frac{\dot{\mathbf{F}}}{c} - \left(\frac{\dot{\mathbf{F}}}{c} \cdot \mathbf{p} \right) \mathbf{p} \tag{3}$$

In this expression, if $\mathbf{p}$ is normalized, so is $\mathbf{p}'$. The formula is accurate to better than 1 mas. (Omission of gravitational light bending results in errors of up to 10 mas for FAME.) It is tempting to set $\mathbf{p}$ to refer to the center of a FAME field and add the aberration offset (everything to the right of the + sign) to the direction vectors of all the stars in the field. Although this will work over a large part of the sky, it can result in errors > 0.2 arcsec at the edge of the field in the worst case. Even computing aberration separately for each quadrant of the field can result in errors of over 0.1 arcsec. Fortunately, the above formula is computationally light and its application to each star in the field of view is not a significant problem.

These effects require a model of the Earth's orbit with respect to the solar system barycenter and a model of FAME's orbit with respect to the center of mass of the Earth. Position and velocity vectors from the two models can simply be added together to yield FAME's position and velocity with respect to the solar system barycenter. Consider the Earth's orbit first. How good does it have to be? For the on-board or quick-look systems, not very. For the few parallax calculations that have to be performed, a circular orbit approximation is perfectly adequate. For aberration, the eccentricity of the orbit (0.017) has to be taken into account, but nothing more complex. The approximation to the Sun's coordinates on page C24 of the *Astronomical Almanac* could be adapted for this. However, there is an even simpler scheme. The velocity of the Earth in its elliptic orbit can be obtained by adding a small *constant* velocity vector to the instantaneous circular-orbit velocity vector. This means that, taken in isolation, the elliptic component of annual aberration shifts each star away from its catalog position by a constant amount. So, the effect of the Earth's eccentricity on aberration (of order 0.3 arcsec) can be embedded in the coordinates of the on-board star catalog. This scheme was used in old star catalogs, before the days of computers, so that a circular Earth orbit could be used for apparent place computations. The offsets of the star positions were known as the elliptic terms (or simply E-terms) of aberration, and are described in the *Explanatory Supplement* [2]. If the E-terms are embedded in the star coordinates in this way, a circular Earth orbit can be used for aberration as well as parallax.

As one might expect, FAME's orbit around the Earth is irrelevant for the on-board parallax calculation but a simple model of the orbit must be used for the aberration calculation. FAME's orbital speed is about 3 km/s, about 10% of the Earth's orbital speed, so FAME's motion adds as much as 2 arcsec to aberration. For the on-board system we therefore need to compute FAME's instantaneous velocity with respect to the Earth to only about 5%. FAME's nominal orbit is circular, and a circular model should work unless things go very awry.

For precise simulations or data analysis the orbital models of the Earth and FAME must be rather sophisticated. The Earth's instantaneous position and velocity vector can be retrieved efficiently from the JPL DE-405 planetary/lunar ephemeris. The JPL ephemeris is accurate enough for FAME analysis and is in the ICRS system. JPL distributes it in the form of Chebyshev series along with software to perform the series evaluations for any required time. NOVAS accesses the JPL ephemerides through a subroutine called SOLSYS. James Hilton of USNO's AA Dept. has developed software that repackages the JPL ephemerides into a more compact form, and we can probably get such a package custom built for FAME applications that would be very fast. (The JPL routines are in Fortran but Hilton's are in C.)

For the high accuracy computations, FAME's orbit around the Earth must be accounted for in both the parallax and aberration calculations. Parallax is the less stressing requirement, of course. The parallax computation for the nearest stars requires that we compute FAME's orbital position to about 500 km, about 1% of its geocentric orbital radius. For the aberration computation, however, FAME's velocity must be known to 1.5 cm/s, about 5×10^{-6} of the geocentric orbital speed. It is interesting to note that in the time it takes a star to cross FAME's field ($\sim$6 s), FAME's orbital velocity *changes* by over 1 m/s — equivalent to an aberration change of almost 1 mas! Needless to say, accounting for aberration to the accuracy needed for data analysis presents quite a technical challenge for the spacecraft tracking and orbit determination system and even affects the choice of orbit used.

2.4 FAME's Orientation and Spin

In dealing with FAME's angular motion, we must use coordinate systems that are fixed in the spacecraft and coordinate systems that are fixed in inertial space. We will use lower-case letters,

$\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$, to refer to the spacecraft-fixed axes and upper-case letters, $\mathbf{X}$, $\mathbf{Y}$, and $\mathbf{Z}$, to refer to the ICRS (inertial) system axes. If we use the coordinate system shown on Foldout 2 of the CSR, where $\mathbf{z}$ is the nominal symmetry (spin) axis (with $+z$ on the instrument side of the spacecraft) and $\mathbf{y}$ is along the main optical axis (with $+y$ bisecting the two aperture directions), then the two apertures are in the directions

$$\begin{aligned} \mathbf{q}_1 &= -\mathbf{x}\,\sin(\gamma/2) + \mathbf{y}\,\cos(\gamma/2) \\ \mathbf{q}_2 &= \mathbf{x}\,\sin(\gamma/2) + \mathbf{y}\,\cos(\gamma/2) \end{aligned} \tag{4}$$

where $\gamma = 84.3°$ is the basic angle. The direction $\mathbf{q}_1$ represents the leading aperture. (See Appendix A for a generalization of eqn. (4) for the case where the aperture directions are not perfectly aligned.)

I assume that we have, or will have, a subroutine that can provide, for any input time t, the instantaneous orientation of FAME in the ICRS system. That is, a subroutine that provides the parameters that allow us to transform between the $[\mathbf{x}, \mathbf{y}, \mathbf{z}]$ system and the $[\mathbf{X}, \mathbf{Y}, \mathbf{Z}]$ system for any time. This subroutine could obtain its information from FAME's on-board attitude control system, from analysis of observations, or some combination. For the algorithm described here, this subroutine is a black box. I am not trivializing the problem of providing such a function; it is simply beyond the scope of this note.

FAME's orientation can be characterized in a variety of ways. One way is to provide the values of three Euler angles and their rates of change. Another is to specify the (X, Y, Z) coordinates (or the equivalent (α, δ) pairs) of two of the three spacecraft axes, along with the rotation components around all three axes. Or, quaternions can be used to specify the instantaneous orientation. Most of FAME's rotation is around the $\mathbf{z}$ axis, of course, but not all — there is a 203 arcsec difference between the $\mathbf{z}$ axis and the instantaneous total angular velocity vector, caused by the spacecraft's precession. This means that the $\mathbf{xy}$-plane is not exactly orthogonal to the spin axis; this is why stars crossing the CCDs have up to a 4 pixel cross-scan drift due to precession.

If we are dealing with FAME's observed attitude, we need to be able to take a series of discrete measurements of FAME's instantaneous orientation and derive from them not just the directions of the two observing ports at any time (a straightforward interpolation problem) but also the direction of FAME's instantaneous spin vector. Suppose we have two rotation matrices, $\mathbf{A}(t_n)$ and $\mathbf{A}(t_{n+1})$, representing the spacecraft attitude at two times t_n and t_{n+1} within a short interval (seconds; see above discussion). Then

$$\mathbf{A}(t_{n+1}) = \mathbf{R}\;\mathbf{A}(t_n) \tag{5}$$

where $\mathbf{R}$ is the rotation matrix representing the infinitesimal spin of the spacecraft between times t_n and t_{n+1}. We simply extract $\mathbf{R}$ by multiplying both sides by the transpose (inverse) of $\mathbf{A}(t_n)$:

$$
\begin{aligned}
\mathbf{A}(t_{n+1})\,\mathbf{A}^t(t_n) &= \mathbf{R}\,\mathbf{A}(t_n)\mathbf{A}^t(t_n) \\
\mathbf{A}(t_{n+1})\,\mathbf{A}^t(t_n) &= \mathbf{R}
\end{aligned}
\tag{6}
$$

By Euler's theorem, the eigenvector of $\mathbf{R}$ represents the direction of the spin axis. The decomposition of a general rotation matrix, like $\mathbf{R}$, into spin axis components and spin angle is given in eqn. E-6 of [4], p. 762. Similarly, in quaternion notation, the spacecraft attitude at times t_n and t_{n+1} can be represented by two unit quaternions, $\mathbf{a}(t_n)$ and $\mathbf{a}(t_{n+1})$, and

$$\mathbf{a}(t_{n+1}) = \mathbf{a}(t_n)\,\mathbf{r} \tag{7}$$

where $\mathbf{r}$ is the quaternion representing the rotation between the two attitude states. The vector part of $\mathbf{r}$ represents the direction of the instantaneous spin axis. In fact, as the interval $t_{n+1} - t_n \rightarrow 0$, a fundamental equation of quaternions can be applied:

$$\dot{\mathbf{a}} = \frac{1}{2}\boldsymbol{\omega}\,\mathbf{a} \tag{8}$$

where $\boldsymbol{\omega}$ is the angular velocity vector (= a quaternion with scalar part 0 and vector part equal to the spin axis). Application of this equation to obtain $\boldsymbol{\omega}$ would require numerical differentiation of the sequence of attitude quaternions $\mathbf{a}(t)$, but that is a simple process with measurements spaced at short intervals.

3 Computing the Observing Sequence

I assume that the software will progress in a loop where the time t advances by of order 1 second per cycle. The time increment Δt between cycles is determined by the requirement that any curvature of the loci of the star images across the focal plane during the time Δt results in an in-scan and cross-scan error within the astrometric tolerance for the calculation. The algorithms that follow assume linear apparent motion of the stars across the focal plane, after spacecraft rotation and optical field distortion have been taken into account; for high accuracy applications, the curvature of these paths can be handled by successive iterations.

The greatest part of the nonlinearity in the stars' paths is due to the spatial variation of the optical distortion. As of mid-February 2001, the 3rd order distortion of the FAME optical design was estimated to be 3.6 arcsec/degree^3. This distortion will cause nonlinearities in stellar paths over Δt = 1 second of up to 0.1 arcsec (1/2 pixel) in one coordinate. This is just within our tolerance of error.

There are also nonlinearities due to the spacecraft spin dynamics. For example, second-order precession (the change in apparent cross-scan drift rate with spin angle) affects the instantaneous position of a star image in the field of view by at most 0.7 mas/s^2 cross-scan. From Marc's analysis, it appears that gravity gradient effects should be about a quarter this amplitude. We are lucky that the nutation period is almost as long as the spin period, so that even if it is excited to a 10 arcsec amplitude, its second-order effect would be < 100 μas/s^2 cross-scan. The second-order effects are generally smaller in-scan. Other short-term effects, generally grouped under the name "jitter", could upset this neat picture if sufficiently large, although Marc's spin dynamics simulations have not identified any. In summary, over a few seconds, the star motions are certainly linear enough for the on-board or quick-look calculations and even for more precise applications. A shorter interval for the basic loop can always be chosen.

3.1 Selecting the Stars

Rapidly calculating the observing sequence with limited computer power depends on performing as few trigonometric function evaluations as possible. With an input catalog of 40 million stars, the key is Lockheed's proposal to divide the input catalog into subcatalogs by position. Let's assume that we divide the catalog into ~40,000 fields of approximately one square degree, which are called "tiles" in the CSR (p. 4-49). Here we assume that the tiles are approximately square and all have approximately the same dimensions. Each tile will contain 1000 stars on average, although there will be a wide variation. See the Lockheed memo by van Bezooijen [5] for a description of the proposed tiling scheme. I assume that we have a subroutine that, given an input unit vector representing a point on the celestial sphere, returns the number of the tile that contains the point, along with the unit vector representing the center of the tile. It is also necessary to have a scheme that, given a specific tile number, delivers all the tile numbers of adjacent tiles.

As stated above, the software loop will cycle in time increments Δt of order 1 sec: $t_1, t_2, t_3, \ldots$. Given a time t_n, we obtain the instantaneous directions of FAME's axes: $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$, all of which are unit vectors expressed in the ICRS system. We can then compute the direction vectors of the two apertures, using eqn. (4), which we will call $\mathbf{q}_1$ and $\mathbf{q}_2$. These unit vectors point to the centers of fields 1 and 2, respectively, in the ICRS system. We also need to obtain FAME's instantaneous spin vector $\boldsymbol{\omega}$, in radians/second.

The scheme for identifying the stars that pass through field k (k=1 or 2) is as follows. We first use our tile number subroutine, providing as input the unit vector $\mathbf{q}_k$, the center of field k at time t_n. It provides the tile number that contains the center of field k along with the unit vector pointing to the tile's center. We then retrieve the tile numbers and unit vectors of all the neighboring tiles (one "ring" around the center tile is probably sufficient).

It is likely that more tiles would be identified this way than would actually be needed for a specific field. Obviously we don't want to deal with more stars than is necessary. The candidate tiles could then be easily culled by computing $\delta = \mathbf{p}_i \cdot \mathbf{q}_k$, where $\mathbf{p}_i$ is the direction of the center of tile i and $\mathbf{q}_k$ is the direction of the center of field k. Tiles would be kept for further analysis if $\delta > \cos r$, where r is an angular radius computed as illustrated in Figure 1.

Figure 1 shows one of the FAME fields of view as the smallest circle and its path across the sky as the inner parallel lines. The star catalog tile boundaries and centers are indicated. In order to ensure that all the stars within the instantaneous field are identified, it is necessary to inspect the contents of all tiles with centers within a larger circle, shown dotted. That circle has a radius equal to the FAME field radius (0.55°) plus half the largest tile diagonal (∼0.7°) plus a small allowance for aberration (0.007°). The tiles that satisfy this criterion, for the indicated position of the field, are shown shaded. The projection of the diameter h of the dotted circle along the path of the field is indicated by dotted parallel lines. As the field moves, the software must eventually inspect the contents of all tiles with centers in this wider path. The field moves smoothly, but the software doesn't. The program advances in discrete time steps of Δt seconds, during which the center of the field moves an angular distance $\Delta s = (0.15°/\mathrm{s})\,\Delta t$. At each step, then, we must inspect the tiles with centers within an *even larger* circle to guarantee that no part of the area inspected has a width less than h. This is the same as saying that the overlap region between two successive inspection circles must have width h, as shown in the figure. This largest circle has the radius r that we seek:

$$r = \sqrt{\left(\frac{\Delta s}{2}\right)^2 + \left(\frac{h}{2}\right)^2} = \sqrt{\left(\frac{\Delta s}{2}\right)^2 + \left(f + \frac{d}{2} + a\right)^2} \tag{9}$$

where Δs is the angular distance moved at each step, f is the FAME field radius, $d/2$ is half the largest tile diagonal, and a is the aberration allowance. It is obvious that keeping the step size Δs as small as possible minimizes the sky area (i.e., number of tiles) to be inspected.[2] For 1 second time steps and 1° square tiles, $r = 1.27°$, defining a circle only 0.5% in area larger than the absolute minimum possible. If the time step is increased to 5 seconds, the inspection circle area is 35% larger. The quantity $\cos r$ is a constant that needs to be calculated only once.

The same considerations apply to a distance test for individual stars, except that there is nothing that corresponds to the tile diagonal. The distance-test circle for individual stars would therefore have radius r':

$$r' = \sqrt{\left(\frac{\Delta s}{2}\right)^2 + (f + a)^2} \tag{10}$$

For 1 second steps, $r' = 0.56°$, defining a circle only 4% larger in area than the actual field of view.

[2]This holds only if the overlap area between the circles is not treated twice, that is, if there is no redundancy in calculations from one step to another.

All this assumes that we sort the tiles and stars with simple distance-from-field-center tests. Specifically, as stated above, we keep only those tiles that satisfy $\mathbf{p}_i \cdot \mathbf{q}_k > \cos r$. At this point we retrieve the data on all the stars in those tiles. This will be of order several thousand stars on average, although where the field crosses the galactic plane we might be dealing with several tens of thousands. For each star, we have a direction vector obtained from the input catalog: $\mathbf{p}_{ij}$ refers to the direction of star j in box i, already updated for proper motion where necessary. We keep only the stars that satisfy $\mathbf{p}_{ij} \cdot \mathbf{q}_k > \cos r'$.

There is some bookkeeping to be done in the software since we do not want to recompute data on stars processed in the previous time step (or several previous steps). With 1 second steps there is considerable overlap between one field and the next, and a large percentage of stars identified at each step will have been processed in a previous step. There are different schemes for performing the needed checks, the choice depending on how the star data is stored and indexed. Note that it is not sufficient just to check the tile numbers, since on successive steps different stars may be taken from the same tile.

3.2 Computing the Focal Plane Positions of the Stars

We next obtain, for time t_n, the geocentric position and velocity vectors of FAME, along with the barycentric position and velocity vectors of the Earth. Summing these vectors in the obvious way, we obtain $\mathbf{F}(t_n)$ and $\dot{\mathbf{F}}(t_n)$, the barycentric position and velocity of FAME for the current time step.

We now apply parallax and aberration, based respectively on $\mathbf{F}(t_n)$ and $\dot{\mathbf{F}}(t_n)$. As previously noted, the proportion of stars needing a parallax correction is tiny for the on-board system and they can be handled as special cases. Aberration is applied to each star using eqn. (3). Star ij's position as affected by aberration (and parallax, where necessary) will be referred to as $\mathbf{p'}_{ij}$.

FAME's action takes place in the focal plane. We need to compute the loci of the star images in this plane and compare them to the positions of pixels on the CCDs. We will use a coordinate system in which the coordinates of specific points on the physical focal plane, e.g., CCD corners or pixels, have constant coordinate values. Although we could use the spacecraft-fixed $[\mathbf{x}, \mathbf{y}, \mathbf{z}]$ system for this purpose, it is more convenient to use another spacecraft-fixed coordinate system, the [$\mathbf{u}$,$\mathbf{v}$,$\mathbf{w}$] system. The vectors $\mathbf{u}$ and $\mathbf{v}$ lie in the nominal focal plane and the origin of the system is in the center of the plane. This is obviously a very natural system for expressing the coordinates of real objects on the instrumental focal plane. For computing the coordinates of star images in this system, however, it is convenient to think of the focal plane projected onto the celestial sphere. The axes of the system are oriented such that when the focal plane is projected onto the sky, the

$\mathbf{v}$ axis points toward the instantaneous "pole" where the spacecraft $\mathbf{z}$ axis intersects the celestial sphere:

$$\begin{aligned} \mathbf{u} &= -\frac{\mathbf{z} \times \mathbf{q}_k}{|\mathbf{z} \times \mathbf{q}_k|} \\ \mathbf{v} &= \mathbf{u} \times \mathbf{q}_k \end{aligned} \tag{11}$$

Here, $\mathbf{q}_k$ is the direction vector of one of the two apertures (k=1 or 2). For stars, the focal plane is an idealized flat surface, w=0, and the star images move across the plane nearly parallel to $\mathbf{u}$. (The arrangement of CCDs on the focal plane may require use of all three coordinates.) Note that either $\mathbf{w}$ or $\mathbf{q}_k$ can be thought of as the optical axis.

To determine the (u, v) coordinates of stars we first need to account for the way that the sky is projected onto the focal plane by the optical system. The effect of the optical distortion on a star's position can be represented by a vector function $\mathbf{D}$ that maps the star's apparent direction $\mathbf{p}'$ to a distorted direction $\mathbf{p}''$:

$$\mathbf{p}'' = \mathbf{D}(\mathbf{p}') = \beta \mathbf{p}' + \boldsymbol{\delta G} \tag{12}$$

where β is a scalar function that describes the spatial variation of the focal plane scale and $\boldsymbol{\delta G}$ is a vector function that absorbs any residual distortion. Both β and $\boldsymbol{\delta G}$ are functions of focal plane position, hence implicitly functions of $\mathbf{p}'$. If $\mathbf{p}''_{ij}$ represents the distorted direction of star ij, computed as above, the coordinates of the star in the [$\mathbf{u}$,$\mathbf{v}$] system would then be simply:

$$\begin{aligned} u_{ij} &= \mathbf{p}''_{ij} \cdot \mathbf{u} \\ v_{ij} &= \mathbf{p}''_{ij} \cdot \mathbf{v} \end{aligned} \tag{13}$$

Note that if $\boldsymbol{\delta G}=\mathbf{0}$, $\mathbf{p}'$ and $\mathbf{p}''$ are in the same direction; the former is a unit vector but the latter is not. The function β is just a computational device that lengthens or shortens the direction vector of each star such that the (u, v) coordinates derived from the vector correctly reflect the radial distortion of the optical system. The coordinates u and v are analogous to the traditional astrometric focal plane coordinates ξ and η and are expressed in units of the optical system's focal length.

Four examples of the main distortion function β are:

(1)	$\beta = 1/\cos\theta$	magnification $\propto \sec^2\theta$, increases radially
(2)	$\beta = 1 + \frac{1}{6}\theta^2$	constant magnification
(3)	$\beta = 1$	magnification $\propto \cos\theta$, decreases radially
(4)	$\beta = 1 - 3.1\,\theta^2$	magnification $\propto 1 - 9.8\,\theta^2$, decreases radially

where θ is the angular distance from the optical axis in radians and $\cos\theta = \mathbf{p}'_{ij} \cdot \mathbf{q}_k$. Note that "magnification" is inversely proportional to "scale", which is usually measured in arcseconds per millimeter. Beta function (1) is taken from [6]; it corresponds to a *gnomic* projection, which yields *tangential* or *standard* coordinates. Beta function (2) is the ideal for an astronomical system, but is difficult to produce without a concave focal surface. As of mid-February 2001, the FAME optical design is described by β function (4), which is equivalent to a 3.59 arcsec/degree^3 barrel distortion. A more precise statement of β function (4) is $1 - 3.10702\,\theta^2 - 0.546\,\theta^4$, although undoubtedly the coefficients will change before the optical design is finalized. Because FAME's field of view is small, the computed (u_{ij}, v_{ij}) coordinates of a given star are the same to within 0.1 arcsec across the field for the first three β functions listed. These four functions do not, of course, exhaust the possibilities for β.

It might seem that we need to evaluate an arccos function to obtain θ for β functions 2 or 4, but actually we need θ^2, which can be computed to 5 or better significant digits over FAME's half-degree field using $\theta^2 = 2(1 - \cos\theta) = 2(1 - \mathbf{p'}_{ij} \cdot \mathbf{q}_k)$.

We assume, for the time being at least, that $\boldsymbol{\delta G}$ is zero.

3.3 Computing the Focal Plane Motions of the Stars

The next task is to compute the directions of motion of the star images across the focal plane. The time derivative of any arbitrary direction vector $\mathbf{r}$, fixed with respect to the spacecraft, is simply $\dot{\mathbf{r}} = \boldsymbol{\omega} \times \mathbf{r}$, where $\boldsymbol{\omega}$ is FAME's instantaneous angular velocity vector (with components expressed in radians/second), and $\dot{\mathbf{r}}$ is measured with respect to an inertial system (e.g., the ICRS). If we imagine $\mathbf{r}$ projected to the celestial sphere, then $d\mathbf{r} = \dot{\mathbf{r}}\,dt$ (in radians) is in the "plane of the sky". Since $\boldsymbol{\omega} \times \mathbf{r}$ represents the motion, in an inertial system, of a direction $\mathbf{r}$ fixed within the spacecraft, then the opposite vector must represent the motion, in a spacecraft-fixed system, of a constant direction $\mathbf{r}$ in an inertial frame. Thus, the motion of star ij as seen from a coordinate system fixed in FAME would be $\dot{\mathbf{p}}'_{ij} = -\boldsymbol{\omega} \times \mathbf{p'}_{ij}$. However, to obtain the instantaneous motion of the star's image across the focal plane we have to account for the spatial variation of the optical system distortion. The quantity $\dot{\mathbf{p}}'_{ij}$ is a starting point, but what we really need is $\dot{\mathbf{p}}''_{ij}$, the distorted motion. This can be accomplished in any of three ways:

(1) We can numerically difference the (u, v) coordinates of the vectors $\mathbf{D}(\mathbf{p'}_{ij})$ and $\mathbf{D}(\mathbf{p'}_{ij} + d\mathbf{p'}_{ij})$, where $d\mathbf{p'}_{ij} = \dot{\mathbf{p}}'_{ij}\,dt$ and dt is a fraction of the interval Δt. This approach may be the simplest both conceptually and computationally.

(2) We can compute the distorted direction of the star's motion given by

$$\dot{\mathbf{p}}''_{ij} = \begin{pmatrix} \nabla(\mathbf{D}\cdot\mathbf{u}) \\ \nabla(\mathbf{D}\cdot\mathbf{v}) \\ \nabla(\mathbf{D}\cdot\mathbf{w}) \end{pmatrix} \dot{\mathbf{p}}'_{ij} = \begin{pmatrix} \partial D_u/\partial u & \partial D_u/\partial v & \partial D_u/\partial w \\ \partial D_v/\partial u & \partial D_v/\partial v & \partial D_v/\partial w \\ \partial D_w/\partial u & \partial D_w/\partial v & \partial D_w/\partial w \end{pmatrix} \dot{\mathbf{p}}'_{ij} \tag{14}$$

where $D_u = \mathbf{D}\cdot\mathbf{u}$, etc., and the various partials of $\mathbf{D}$ are evaluated for $\mathbf{p}'_{ij}$. It is not clear whether the partials that contain w are important (or even what they mean in this case), since $\dot{\mathbf{p}}'_{ij}$ has a very small component in the $\mathbf{w}$ direction; it is possible that this formula would work equally well with the right column and bottom row of the matrix set to all zeros.

(3) If $\boldsymbol{\delta G}=\mathbf{0}$, we can use the equation

$$\dot{\mathbf{p}}''_{ij} = (\nabla\beta\cdot\dot{\mathbf{p}}'_{ij})\mathbf{p}'_{ij} + \beta\dot{\mathbf{p}}'_{ij} \tag{15}$$

where β and $\nabla\beta$ are evaluated for $\mathbf{p}'_{ij}$.

Note that if the maximum optical distortion at the edge of the field is smaller than the error tolerance for the calculation, it may be better to ignore the distortion entirely rather than to use the distortion gradient as part of a first-order extrapolation of the star image motion. This is especially true when $\Delta s = (0.15°/\mathrm{s})\,\Delta t$ is a significant fraction of the field width. In such cases, the gradient of the distortion at one point of a star's path across the focal plane will not provide an adequate prediction of the star's position Δt later, and more accurate results may be obtained by simply setting $\dot{\mathbf{p}}''_{ij} = \dot{\mathbf{p}}'_{ij}$.

Once the vector $\dot{\mathbf{p}}''_{ij}$ is determined, its $(\dot{u}, \dot{v})$ coordinates are simply

$$\begin{aligned} \dot{u}_{ij} &= \dot{\mathbf{p}}''_{ij}\cdot\mathbf{u} \\ \dot{v}_{ij} &= \dot{\mathbf{p}}''_{ij}\cdot\mathbf{v} \end{aligned} \tag{16}$$

which are expressed in units of the optical system's focal length per second. The star's motion makes a small angle ψ with respect to the $\mathbf{u}$ axis, where $\tan\psi = \dot{v}_{ij}/\dot{u}_{ij}$; $\dot{u}_{ij}$ is always positive. The instantaneous speed of the star image motion across the focal plane is $\dot{s}_{ij} = \sqrt{\dot{u}^2_{ij} + \dot{v}^2_{ij}}$. We will need to know the quantities $\dot{s}_{ij}$ and $\cos\psi$, which can be computed to sufficient precision using

$$\dot{s}_{ij} \approx \dot{u}_{ij} + \frac{1}{2}\,\frac{\dot{v}^2_{ij}}{\dot{u}_{ij}} \tag{17}$$

Then $\cos\psi = \dot{u}_{ij}/\dot{s}_{ij}$. These two quantities are precise to about 12 significant digits (assuming exact values of $\dot{u}_{ij}$ and $\dot{v}_{ij}$) for $|\psi| < 0.1°$, which is realistic for FAME.

It is good to remember that u_{ij}, v_{ij}, $\dot{u}_{ij}$, $\dot{v}_{ij}$, $\dot{s}_{ij}$, and $\cos\psi$ have been computed for the specific instant t_n. These are the linear parameters of star ij's instantaneous motion across the focal plane, which would slowly change with time if we tracked the star's actual curved path.

3.4 Computing Where and When a Star Crosses a CCD Row

Having defined a linear approximation to the locus of a star's image in the focal plane, when does the image intersect the readout row of pixels on one of the CCDs and on what column does that occur?

We obviously need to know the (u, v) coordinates of pixels [4095,0] and [4095,2047] on each CCD, defining the corners on the readout end of the chip. For use with this development, these coordinates should be expressed in units of the system focal length, F, nominally 15 m. Of course, we will not know these coordinates precisely enough initially for data analysis; they will have to be determined from the observations. However, the *a priori* coordinates will probably be good enough for the on-board or quick-look system. If necessary, the coordinates could be updated at any time.

It is also to be expected that for the high-precision computations, the CCDs may have non-negligible w coordinates, which may be different for each corner. That is, the CCDs may be elevated or depressed and tilted with respect to our idealized focal plane. That effect is not dealt with explicitly here because I assume that such a geometry will manifest itself in the observations as a shift of the effective (u, v) coordinates of the CCD corners, which can be solved for from the equations given here. Similarly, the entire instrumental focal plane is likely to be shifted and tilted from its nominal placement and that the usual "plate constants" will have to be determined. The plate model can then be used to correct the (u, v) coordinates of the CCD corners used in this development.

We want to know the column number that a star image will be travelling along at the readout end of the CCD and the time that the image will cross the last row of pixels there. Since both the pixel row and the star image motion are straight lines (in this approximation), this is a simple problem. Let pixels [4095,0] and [4095,2047] of a given CCD — the corner pixels on the readout end — have coordinates (u_0, v_0) and (u_1, v_1), respectively.[3] See Figure 2. Define the following quantities:

[3]The coordinates of a pixel are assumed here to refer to the center of the pixel. The "readout end of the CCD" is thus actually a line running along the centers of the pixels in the last row (row 4095) of the CCD. In this scheme, the photosensitive area of the CCD extends 1/2 pixel (approximately) beyond the lines defined by the coordinates of the pixels on the edge of the chip.

$$
\begin{aligned}
m &= \dot{v}_{ij}/\dot{u}_{ij} \\
b &= v_{ij} - m\,u_{ij} \\
k &= (u_1 - u_0)/(v_1 - v_0) \\
a &= u_0 - k\,v_0
\end{aligned}
\tag{18}
$$

The quantity m is the slope of the star motion line with respect to the $\mathbf{u}$ axis and k is the slope of the CCD row with respect to the $\mathbf{v}$ axis; both are $\ll 1$. In Figure 2, the star motion line makes an angle ψ (negative in the figure) with the $\mathbf{u}$ axis and m would then be $\tan\psi$. The quantity b is the intercept of the star motion line on the $\mathbf{v}$ axis and a is the intercept of the line defined by the CCD row on the $\mathbf{u}$ axis. The two lines in question meet at the point I:

$$
\text{Intersection point I} = (u_I, v_I) = \left(\frac{kb + a}{1 - km}, \frac{ma + b}{1 - km}\right) \tag{19}
$$

which is always well defined since $|km| \ll 1$ and the denominators are ≈ 1. The (u_I, v_I) coordinates of point I can be easily transformed back into pixel numbers. The column number, c_{ij}, that the star image crosses at the end of the CCD is simply

$$
c_{ij} = 2047\,\left(\frac{v_I - v_0}{v_1 - v_0}\right) \tag{20}
$$

Obviously if this conversion results in $c_{ij} < 0$ or $c_{ij} > 2047$ then the star will not be observed by the CCD in question. These calculations should be repeated for all 24 CCDs.

The time t_{ij} that the star ij crosses point I depends on the length of the line segment joining points (u_{ij}, v_{ij}) and (u_I, v_I) and the rate of motion, $\dot{s}_{ij}$, of the star image:

$$
\begin{aligned}
t_{ij} &= t_n \pm \frac{\sqrt{(u_I - u_{ij})^2 + (v_I - v_{ij})^2}}{\dot{s}_{ij}} \\
&\approx t_n + \frac{u_I - u_{ij}}{\dot{s}_{ij}\,\cos\psi} \approx t_n + \frac{u_I - u_{ij}}{\dot{s}_{ij}}
\end{aligned}
\tag{21}
$$

where the sign of the second term in the first equation is the same as the sign of $u_I - u_{ij}$. The second and third equations allow us to avoid the square root function. They are approximations that work quite well — to 12 and 6 significant digits, respectively — for the small values of ψ that we encounter. (Actually, the second equation is exact as shown, but our values of $\dot{s}_{ij}$ and $\cos\psi$, obtained using eqn. (17), are approximations.)

3.5 Corrections for Offset of CCD Charge Distribution

So far, we have determined the time when a given star image crosses the last row of pixels (row 4095) on a CCD chip, and the column number that it crosses. Note that this is not, in general, the exact time or place that the center of the CCD charge packet generated by the star image will arrive there. That is, the algorithm described above deals with photons, not electrons. The electron distribution moving along the CCD is smeared in both directions by a number of effects, and systematic shifts of the final centroid will occur. For example, the mismatch of TDI rate and star motion rate and the quantized nature of the charge transfer process along the CCD tend to separate the electrons from the photons in the **u** direction. Similarly, the cross-scan motion of the star as it moves across the CCD smears out the charge in the **v** direction, effectively shifting the column that will be measured for the center of the charge distribution.

First-order corrections for these effects are simple. Consider the cross-scan motion first. The slope of the line defining the linear component of the star's motion with respect to the long edge of the CCD is just $m+k$, since $|m|$ and $|k|$ are individually quite small. The number of CCD columns that the star image crosses is then $4096\,|m+k|$; this is the amount of cross-scan smearing of the PSF. The effective center of the charge distribution is thus shifted by half this amount:

$$c_{ij\,charge} = c_{ij} - 2048\,(m+k) \tag{22}$$

The in-scan smearing due to a TDI rate mismatch is similar. Suppose the star motion rate, s_{ij}, and the TDI rate, s_{tdi}, are measured in units of the system focal length per second ($F = 15$ m and 1 pixel $= 10^{-6}F$). The number of rows of in-scan smearing is just $4096\,|s_{ij}/s_{tdi} - 1|$. The time that the effective center of the charge distribution arrives at the last row of pixels is then

$$t_{ij\,charge} = t_{ij} - 2048 \times 10^{-6}\left(\frac{1}{s_{ij}} - \frac{1}{s_{tdi}}\right) \tag{23}$$

where we have neglected the small component of cross-scan motion in the scan direction.

Most stars will cross several CCDs, so there may be as many as four intersection points I for each star. Each one defines a FAME observation. All of the computed observations for all stars in both fields of view should be sorted by time $t_{ij\,charge}$ and transferred to whatever system needs to act on them. The program can then cycle to the next time step, $t_{n+1} = t_n + \Delta t$, and repeat the procedures outlined beginning at Section 3.

4 Remarks and Notes on Extension

A simple algorithm has been provided for determining FAME's observation sequence, using very simple equations. For $\Delta t = 1$ second, the accuracy of the basic algorithm is a half pixel (0.1 arcsec), which is set by the neglect of the curvature of the star's path due to the optical distortion (for the optical design of February 2001), as well as the approximations used in the corrections for the charge packet offsets. This accuracy is good enough for the on-board or quick-look system, or as a fast way to establish a first-order approximation to the observation geometry at a given time within the context of a more complete analysis.

The basic algorithm as outlined requires no trig or other transcendental functions that must be executed for each star at each step. Of course, there will be a great many "hidden" trig functions associated with determining FAME's instantaneous orientation, a non-trivial process not considered in this note. A few more are required for computing FAME's position and velocity for aberration. However, these calculations are performed only once per time step. The important thing is that at each step there are no per-star trig functions that could easily accumulate to great numbers given the size of the input catalog.

If a higher accuracy calculation is needed, the algorithm presented above would need a number of revisions. The aberration calculation would have to be made relativistically correct and the gravitational deflection of light added in. A larger fraction of the stars would need parallax corrections, and proper motion updates would have to be done much more frequently. The evaluation of the orbital position and velocity of both the Earth and FAME would become much more complex. The expression $\dot{\mathbf{p}}'_{ij} = -\boldsymbol{\omega} \times \mathbf{p'}_{ij}$ would need an extra term for the rate of change of the star's apparent position in inertial space (mainly due to aberration). More fundamentally, the above development assumes that the motion of the star images across the field of view is linear. There are a variety of effects, some involving the spacecraft's spin dynamics and others involving the distortion field, that invalidate this assumption for high accuracy calculations (see, e.g., discussion at beginning of Section 3).

The obvious way of using the scheme presented here for higher accuracy calculations is to use it iteratively. The process outlined in Section 3 results in a list of stars, observation times, and CCD column numbers, with the observables good to about 0.1 arcsecond, given our spacecraft spin model. To improve on this, it would be necessary to iterate the entire process on each observation individually. That is, for each observation (star-CCD combination), we would use the computed t_{ij} as if it were a t_n, and go back though the calculations starting at Section 3.1 (obviously the star selection process can be skipped). This allows us to obtain the spacecraft orientation from our spin model for an instant within 0.3 ms of the actual observation time (as predicted by our model), which means that we extrapolate the star's motion across the focal plane over a much shorter distance. Effectively, we iterate the nonlinearities out of the process. Of course, we are now in a computationally more intensive regime since we are dealing with each observation individually, but the equations within each iteration remain simple.

For analysis of simulated or real observations, the first cut through the process is not even necessary, since we start (after image centroiding, that is) with a list of stars, CCD identifiers, observation times, and column numbers. The iterative process will then yield C's based on our models that can be subtracted from the given O's to form $(O - C)$'s. The task yet to be done is to use the equations presented here to derive the expressions for the partials of various interesting model parameters in terms of the two orthogonal observables, time and column number.

Note that although the iterative process can produce more accurate CCD column numbers and last-row crossing times for the star images, the corrections for the offset of the charge packet are not improved by iteration. The actual offsets depend on the whole history of the star's nonlinear motion across the CCD, whereas the simple corrections presented above assume linear motion. Thus, to improve the match of the C's with the O's, better models for the offset of the charge distribution will have to be developed. Since, in the process outlined above, each star's location on the focal plane is determined a number of times, enough information on the star's actual track should be available to serve as a basis for such models. Hopefully, an accurate analysis of observations will not require us to deal with the pixel-by-pixel history of each star's image motion across the length of each CCD. If such detail is required, then the approach outlined here is inappropriate and iteration won't help. In that case, the entire process becomes significantly much more complex and the computational load increases dramatically.

References

[1] Kaplan, G. H., Hughes, J. A., Seidelmann, P. K., Smith, C. A., & Yallop, B. D. 1989, AJ, 97, 1197

[2] Seidelmann, P. K., ed. 1992, Explanatory Supplement to the Astronomical Almanac (Mill Valley, CA: University Science Books)

[3] Kaplan, G. H. 1990, BAAS, 22, 930; code and documentation available at http://aa.usno.navy.mil/AA/software/novas

[4] Wertz, J. R., ed. 1978, Spacecraft Attitude Determination and Control (Dordrecht: Reidel)

[5] van Bezooijen, R. 1999, Lockheed Engineering Memorandum of 5/5/99, "Tiles for the FAME Star Catalog"

[6] Green, R. M. 1985, Spherical Astronomy (Cambridge: University Press)

A Generalized Formula for Aperture Directions

In Section 2.4, eqn. (4) for the directions of the two spacecraft apertures is given as

$$\begin{aligned}\mathbf{q}_1 &= -\mathbf{x}\,\sin(\gamma/2)\; + \mathbf{y}\,\cos(\gamma/2) \\ \mathbf{q}_2 &= \quad \mathbf{x}\,\sin(\gamma/2)\; + \mathbf{y}\,\cos(\gamma/2)\end{aligned}$$

where $\gamma = 84.3°$ is the basic angle, and $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$ refer to the directions of the spacecraft-fixed axes expressed in the ICRS system. This assumes that the spacecraft $\mathbf{y}$ axis bisects the two apertures, and that both apertures are parallel to the $\mathbf{x}$-$\mathbf{y}$ plane.

Suppose neither of the above two assumptions holds. The direction of the leading aperture, $\mathbf{q}_1$, can be specified in the spacecraft-fixed system by two angles, γ_1 and υ_1, which can be thought of as the "longitude" and "latitude", respectively, of the aperture. (Note that we are concerned here only with the *directions* of the apertures — the extension of their optical axes to infinity — and not the physical locations of the apertures on the spacecraft.) Latitude is measured with respect to the spacecraft equator, the $\mathbf{x}$-$\mathbf{y}$ plane. The origin of longitude is the $\mathbf{y}$-$\mathbf{z}$ plane, with positive longitudes measured counterclockwise as viewed from +z. The trailing aperture, $\mathbf{q}_2$, can be specified by the angles γ_2 and υ_2. We expect that $\gamma_1 \approx -\gamma_2 \approx \gamma/2$ and that both υ_1 and υ_2 are small.

Then eqn. (4) can be replaced by

$$\begin{aligned}\mathbf{q}_1 &= -\mathbf{x}\,\cos\upsilon_1 \sin\gamma_1\; + \mathbf{y}\,\cos\upsilon_1 \cos\gamma_1\; + \mathbf{z}\,\sin\upsilon_1 \\ \mathbf{q}_2 &= -\mathbf{x}\,\cos\upsilon_2 \sin\gamma_2\; + \mathbf{y}\,\cos\upsilon_2 \cos\gamma_2\; + \mathbf{z}\,\sin\upsilon_2\end{aligned}$$

For the case $\gamma_1 = -\gamma_2 = \gamma/2$ and $\upsilon_1 = \upsilon_2 = 0$, this reduces to eqn. (4).

In the general case, the basic angle is $\gamma = \arccos(\mathbf{q}_1 \cdot \mathbf{q}_2) \approx \gamma_1 - \gamma_2$.

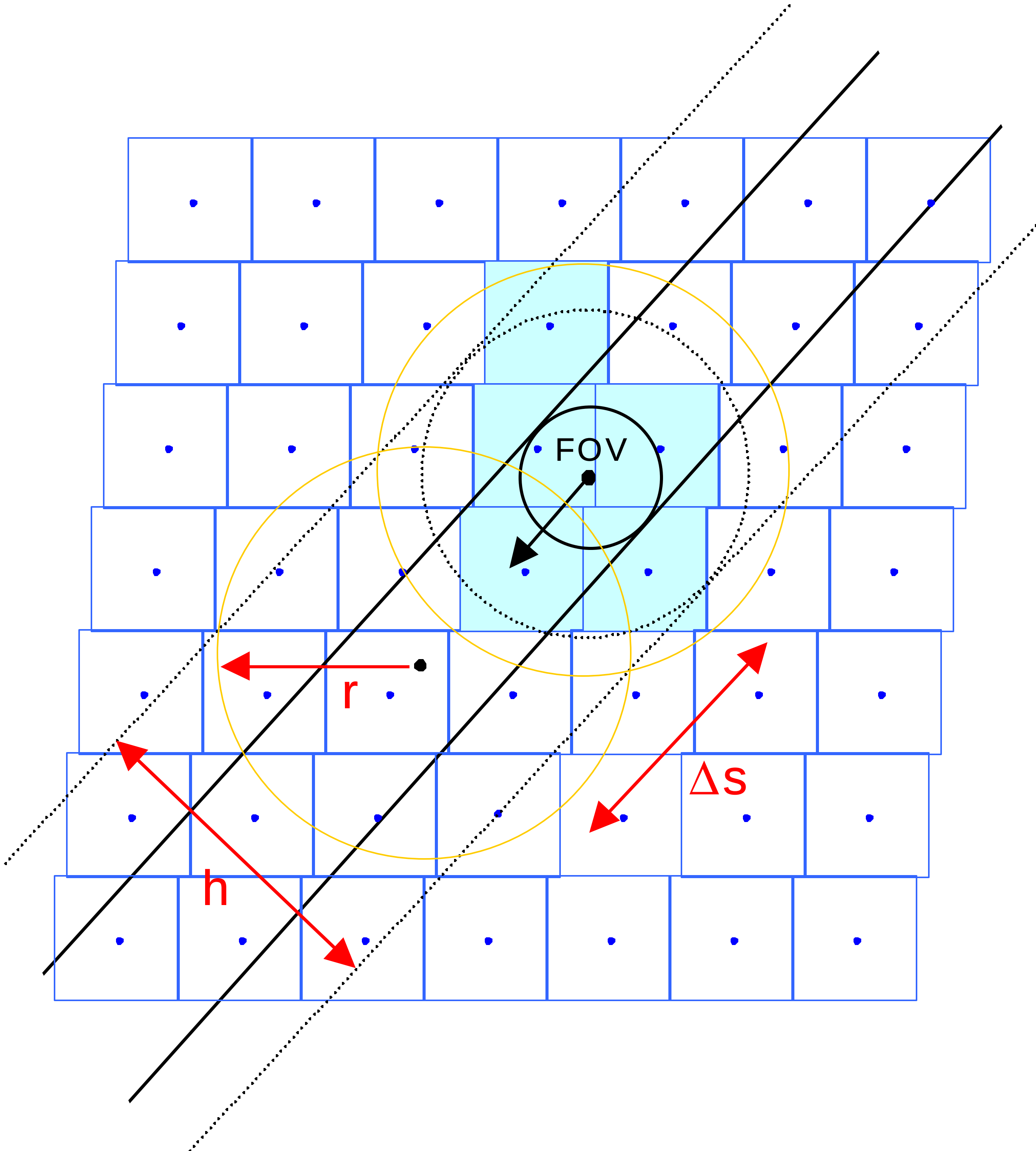

Figure 1. The path of one of FAME's fields of view across the sky. The arc length Δs is the distance that the field moves in one software cycle time Δt.

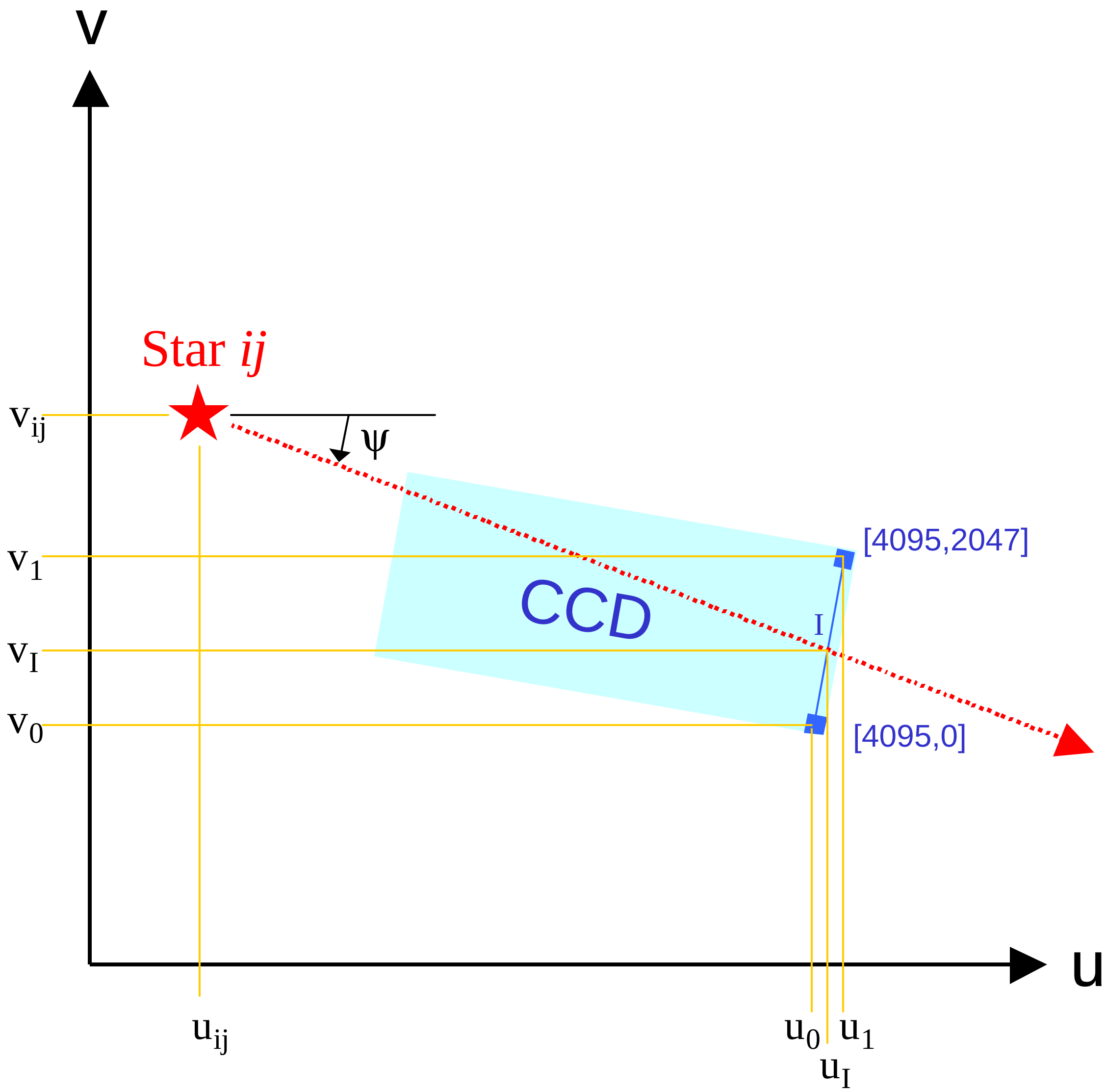

Figure 2. The path of a star across FAME's focal plane, showing its intersection point, I, with the last row of pixels on one of the CCDs.