To Boldly Go Where No Man has Gone Before: Seeking Gaia’s Astrometric Solution with AGIS

U. Lammers, L. Lindegren,¹ W. O’Mullane, and D. Hobbs¹

European Space Astronomy Centre of ESA, Madrid, Spain

Abstract. Gaia is ESA’s ambitious space astrometry mission with a foreseen launch date in late 2011. Its main objective is to perform a stellar census of the 1,000 million brightest objects in our galaxy (completeness to $V = 20$ mag) from which an astrometric catalog of micro-arcsec ($\mu$as) level accuracy will be constructed. A key element in this endeavor is the Astrometric Global Iterative Solution (AGIS) - the mathematical and numerical framework for combining the $\approx 80$ available observations per star obtained during Gaia’s 5 yr lifetime into a single global astrometric solution. AGIS consists of four main algorithmic cores which improve the source astrometric parameters, satellite attitude, calibration, and global parameters in a block-iterative manner. We present and discuss this basic scheme, the algorithms themselves and the overarching system architecture. The latter is a data-driven distributed processing framework designed to achieve an overall system performance that is not I/O limited. AGIS is being developed as a pure Java system by a small number of geographically distributed European groups. We present some of the software engineering aspects of the project and show used methodologies and tools. Finally we will briefly discuss how AGIS is embedded into the overall Gaia data processing architecture.

1. Gaia Overview

Gaia is ESA’s ambitious next-generation space astrometry mission. The launch of the satellites is currently scheduled for December 2011 from French-Guyana on a Soyuz-Fregat launcher to the second Lagrange point L2. From this vantage point Gaia will conduct an optical all-sky survey of the Milky-way lasting five or six years. The goal is to observe all “point-source like” objects brighter than 20th magnitude (including solar system bodies) which should amount to 1000–1200 million observed objects. The expected total raw telemetry data volume is about 100 TB (40 GB/day). There is the notion of a Main Database (MDB) that will hold raw and reduced data and the MDB is believed to reach perhaps a PB in size.

Gaia’s operation and observing principles are based on those of its predecessor mission Hipparcos. The main aim is to determine for each observed object the five standard astrometric parameters $\alpha$, $\delta$ (position), $\varpi$ (parallax), and $\mu_{\alpha^*}$, $\mu_\delta$ (proper motion) at an accuracy level of $\mu$as. This is roughly three orders of magnitude more accurate than Hipparcos. In addition to the astrometry there are also complementary low-resolution spectro-photometry and spectroscopic

¹Lund Observatory, Lund, Sweden
measurements. The latter will give for a subset of the 1000 million objects the sixth astrometric parameter radial velocity, $v_R$.

Design, construction, launch, and operations of the satellite and its payload is entirely under ESA’s responsibility and funding. The data processing on the other hand is organized and conducted by the Gaia Data Processing and Analysis Consortium (DPAC). This group of about 250 European scientists and software engineers is mainly nationally funded.

The end product of the mission is an astrometric catalog of unprecedented completeness and accuracy. As a typical example the targeted end-of-mission parallax standard error for 15th magnitude stars is $20 \mu$as.\(^2\) AGIS is the mathematical and numerical scheme selected by DPAC to generate the astrometric core solution which shall form the basis for the construction of the catalog.

With three years remaining until launch the Gaia project is currently in its implementation phase C/D which means that hardware is being built, integrated and tested. Routine operations are planned to start in early 2012 after the injection of the satellite into its L2 Lissajous orbit and a commissioning phase of a few months duration. The completion of the final catalog is not expected before 2020 with one or two preceding intermediate releases. DPAC’s software engineering activities started officially in 2005. The complex system is aimed to be operational by the time of the launch but development is anticipated to continue for at least some years into the mission.

Gaia’s focal plane is depicted in Figure 1: An array of 106 CCDs operated in TDI (Time Delayed Integration) mode measures about 1 m × 0.5 m in size and comprises almost one billion pixels in total. The satellite is spinning about its main symmetry axis at a rate of $60''$ s\(^{-1}\). Light of stars entering one of the two FOVs (separated by a basic angle of 106.5°) is projected through an optical system of mirrors and a beam combiner onto the focal plane and transits it horizontally (from left to right in the figure) in about one minute. Different groups of CCDs have different functions as indicated in the image. The large central area of 7 × 9 CCDs is the Astrometric Field in which the astrometric measurements take place. These are transit times (times when the centroid of the line spread function crosses over designated points on the CCDs) converted to angles in a given reference system. They constitute the fundamental input to AGIS.

During its five year mission Gaia will observe the sky following a carefully designed “Nominal Scanning Law” (NSL). Three motions are of relevance for the NSL:

1. The Earth moves together with Gaia around the sun, or in a Gaia-centered frame, the sun moves around Gaia
2. The satellite revolves about its main symmetry axis at a rate of 4 revolutions per day ($60''$ s\(^{-1}\)). The telescope viewing directions are orthogonal to the spin axis and separated by a basic angle of 106.5°
3. The spin axes performs a slow precession about the satellite-sun direction at a fixed solar aspect angle of 45°. A complete revolution takes about 63 days.

\(^2\)The arc subtended by a 20 $\mu$as angle corresponds roughly to the diameter of a golf ball on the surface of the moon viewed from the Earth.
Figure 1.: Cartoon of Gaia’s focal plane consisting of an array of 106 CCDs operating in TDI mode. There are about one billion pixels in total. Due to the satellite’s spinning motion, star images transit the focal plane from left to right (AL direction). The measurements that take place in the large central $7 \times 9$ Astrometric Field constitute the main input to AGIS.

Simulating the NSL in software yields Figure 2 (left) showing an all-sky Hammer-Aitoff projection in equatorial coordinates of the transit numbers, i.e. the number of times during the five years that each sky position is observed by Gaia’s telescopes. It can be seen that significant inhomogeneities exist: The regions around the poles and around ecliptic latitudes $\pm 45^\circ$ are most frequently seen. The lowest number of observations occurs around the ecliptic plane. This has direct consequences for the accuracies of the astrometric solution that can be expected from AGIS. Figure 2 (right) depicts this for one particular magnitude ($V=11.8$) in the case of the parallax standard error: the highest accuracies of around $5 \mu\text{as}$ can be obtained in the high-latitude regions where the number of observations is largest.

2. AGIS Overview

AGIS addresses the basic problem of determining the five astrometric parameters for each of the approximately one billion stars from the $\approx 80$ available observations in a globally self-consistent manner. In order to formulate this problem mathematically, three basic models are needed: A model for the sources (S) containing the wanted parameters, a model for the time-varying attitude (A) of the satellite, and a calibration (C) model that characterizes the geometry of the focal plane and other relevant effects. The latter two models are expressed in terms of “nuisance” parameters which are needed to couple the wanted param-
Figure 2. Left: All-sky NSL transit image showing colour-coded the frequency with each point in the sky is observed by Gaia during 5 years. The average number is around 80. The colour-bar ranges from 0 to 250. Right: Theoretically achievable standard parallax errors for magnitude $V = 11.8$ deduced from NSL transit image and other key mission parameters. The image defines in a sense the best solution that AGIS can possibly converge to (for that magnitude). The color-bar ranges from 0 to $11 \mu$as.

AGIS consists of three basic building blocks, viz.

1. Models for S, A, C jointly referred to as “observation model”.
2. A mathematical formulation of the problem whose solution gives the optimal fit of the observations to the observation model.
3. A concrete numerical method to solve the problem.

2.1. Observation Model

The source model employed by AGIS is the standard astrometric one: Each observed source has a fixed position $(\alpha_0, \delta_0)$ at a reference epoch expressed in the ICRF. It also moves through space with a uniform velocity (expressed as a proper motion with components $(\mu_\alpha^*, \mu_\delta)$) and possesses a positive annual parallax $\varpi$ from which its distance to the ICRF barycenter can be deduced.

The attitude model used by Gaia is based on a quaternion representation of the time-dependent orientation of the satellite, $q(t) = (q_1(t), q_2(t), q_3(t), q_4(t))$. Each quaternion component in turn is modeled through an expansion with short-ranged B-spline basis functions defined on a discrete time grid:

$$q_i(t) = \sum_{j=0}^{N-1} c_{ij} B_j(t), \quad i = 1, 2, 3, 4. \tag{1}$$

The number of degrees of freedom, $N$, of the spline and its order is chosen such that the resulting representation has enough flexibility to model all relevant expected attitude variations and disturbances with sufficient accuracy. At the moment in AGIS cubic splines and spline knot separations of 15s are used which gives about $4 \times 10^7$ unknown attitude parameters.

The calibration model is not described here. It suffices to say that it may be expressed in terms of some $10^5$ unknown parameters that are determined by AGIS to the same level of accuracy as the S and A parameters. There is also a fourth type of parameter set which is not considered further here: global (G)
astrophysical parameters like $\gamma$ of the Parameterised-Post-Newtonian formalism describing light-bending in the presence of gravitational bodies. In total the number of G parameters will only be of the order of a few 10 to 100.

Combining the partial models into one coherent observation model $f$ leads to the general equation:

$$t_L^{\text{pred}} = f^{(\text{AL})}_L(s_i, a_j, c_k, \text{aux}) \pm \sigma_L$$

where $L$ is the observation index, $i$ is the source index, $j$ is the attitude interval index, $k$ is the calibration unit index and with the assumption that the mappings $L \leftrightarrow i$, $L \leftrightarrow j$, $L \leftrightarrow k$ are known. The term $\text{aux}$ stands for a number of auxiliary quantities that the observation model must take into account (e.g. satellite ephemerides) but can be regarded as known from onboard and/or on-ground measurements. $\sigma_L$ is the measurement noise that each observation possesses. The result of evaluating the observation model function $f_L$ with a set of arguments is a prediction of the observable quantity $t_L$, that is, the CCD transit time of the observation $L$. (AL) stands for Along-Scan and designates one of two possible observation modes. The other one, Across-Scan (AC), is disregarded here as it is not of relevance for the discussion.

Let $p_L(x)$ be the probability density function for the noise in AL observation $L$. Then the best fit between the observations and model is given by the maximum of the total likelihood function

$$\max_{s,a,c} \prod p_L \left( t_L - f^{(\text{AL})}_L(s_i, a_j, c_k, \text{aux}) \right).$$

Assuming a centered Gaussian noise of standard deviation $\sigma_L$ this is equivalent to a least-square minimization of

$$\chi(s, a, c) = \sum_L \left( \frac{t_L - f^{(\text{AL})}_L(s_i, a_j, c_k, \text{aux})}{\sigma_L} \right)^2.$$  \hspace{1cm} (4)

This equation expresses in a common mathematical form that the sought solution vector $(s, a, c)$ gives the optimal agreement between the observational model and the actual observations.

### 2.2. Normal equations

A standard way to tackle multi-dimensional least-square problems with many more measurements than unknowns is through Normal Equations. For Eq. 4 this yields:

$$\left[ \sum_L \left( -\frac{\delta u_L}{\delta x} \right) \left( -\frac{\delta u_L}{\delta x'} \right) \right] \cdot \Delta x = \sum_L \left( -\frac{\delta u_L}{\delta x} \right) u_L$$

with

$$u_L(x) \equiv \frac{R_L(x)}{\sigma_L} = \frac{t_L - f^{(\text{AL})}_L(x, \text{aux})}{\sigma_L w(R_L(x)/\sigma_L)^{-1/2}}.$$  \hspace{1cm} (6)

The quantity $u_L(x)$ is the normalized residual for observation $L$. The standard error $\sigma_L$ of the observation is modified by a downweighting function $w$ which
Figure 3.: Structure of the Normal Equation matrix in AGIS. The different background patterns of the cells indicate different levels of sparseness: White (zero), light-hashed (sparse), dense-hashed (less sparse), solid blue (filled). The large red zeros represent simplifications (see text for details).

itself depends on the unscaled residual $R_L$. This is expected to take care of outlier observations by giving them a lower weight such that they contribute correspondingly less to the solution. Eq. 5 in matrix form reads

$$\mathbf{N} \cdot \Delta \mathbf{x} = \mathbf{b}$$

and this constitutes the basic AGIS equation. Here $\mathbf{N}$ is an $n \times n$ symmetric, positive-definite matrix with $n$ being the number of sought parameters (wanted source + nuisance). $\mathbf{b}$ forms the right-hand side — a vector of length $n$.

Eq. 7 looks very much like a standard linear algebra problem to be tackled with direct solution algorithms. A direct approach, however, cannot be employed for two reasons.

1. The observation model $f_L(\mathbf{x})$ is non-linear in the vector of unknowns $\mathbf{x}$. The non-linearity is quite weak though for small (i.e. $\mu$as) deviations from the sought solution.
2. The observation downweighting function $w(R_L(\mathbf{x})/\sigma_L)$ is strongly non-linear in $\mathbf{x}$.

2.3. Iterative Solution

Consequently, Eq. 7 has to be solved through iterations. Efficient iterative methods for solving linear systems have been in widespread use for decades. Their applicability in AGIS is limited however because the normal equation matrix $\mathbf{N}$ is simply too large to fit in memory. Figure 3 shows the actual structure of the matrix in symbolic form. The meaning of the various matrix elements is intuitively clear: for example, the very first row in the top-middle S-A block contains only non-zero elements for those attitude intervals in which source 1 was observed. An overall element sparseness exists but its non-uniformity across the different blocks prevents the use of any standard sparse-matrix techniques. In
this situation we simplify the problem by setting as many blocks to 0 as possible, viz. S-A, S-C, A-C, and C-A indicated with large overplotted zeros in Figure 3. It is clear that no other block can be zeroed in addition as this would lead to an unrealistic “unconnected” problem. The block-zeroing appears mathematically unmotivated but it should still be an adequate approach to the initial problem thanks to the outer iterations that are carried out. These are now practically organized in the following way:

1. \( i = 0 \)
2. Choose starting values for all unknowns: \( S_0, A_0, C_0 \)
3. Compute optimal \( S_{i+1} \) values using fixed \( A_i \) and \( C_i \)
4. Compute optimal \( A_{i+1} \) values using fixed \( C_i \) but \( S_{i+1} \) updated in step 3
5. Compute optimal \( C_{i+1} \) values using fixed \( A_i \) but \( S_{i+1} \) updated in step 3
6. Increment \( i \) and go to step 3

This iterative cycle is stopped if the adjustments to the sought source parameters after a completed iteration become smaller than a chosen limit.

2.4. AGIS Iterations and Standard Linear Algebra

At first sight it is not clear why this particular way of carrying out the iterations works and what it means in terms of standard linear algebra. With the simplifications to the normal equation matrix illustrated in Figure 3, Eq. 7 can be written as

\[
\begin{pmatrix} S & 0 & 0 \\ U & A & 0 \\ V & 0 & C \end{pmatrix} \begin{pmatrix} \Delta s \\ \Delta a \\ \Delta c \end{pmatrix} = \begin{pmatrix} b_s \\ b_a \\ b_c \end{pmatrix}
\] (8)

which evaluates to

\[
S \cdot \Delta s = b_s \implies \Delta s 
\] (9)
\[
A \cdot \Delta a = b_a - U \cdot \Delta s \implies \Delta a 
\] (10)
\[
C \cdot \Delta c = b_c - V \cdot \Delta s \implies \Delta c. 
\] (11)

It can be shown with relative ease that solving Eqs. 9–11 is strictly equivalent to cyclically executing the processes S, A, and C in the manner described above. Note that in doing so the partial matrices \( U \) and \( V \) are never explicitly computed: subtracting \( U \cdot \Delta s \) and \( V \cdot \Delta s \) from \( b_a \) and \( b_c \) respectively is equivalent to using the updated source parameters in steps 4+5 of the iterative scheme (Sect. 2.3).

Solving Eqs. 9–11 is done in practice through robust Cholesky decompositions. In the case of \( A \) and \( C \) this means that matrices of a few GB have to be kept in memory. For \( S \) on the other hand only \( 5 \times 5 \) matrices have to be decomposed one at a time (i.e. one source at a time) as the corresponding \( S \) sub-matrix in Figure 3 is block-diagonal.

The ad-hoc zeroing of sub-blocks in the normal equation matrix is actually a well-known technique in the theory of iterative methods for solving linear systems: If one cannot solve \( Nx = b \) then try to solve an easier system \( Mx = b \) instead, where \( M \) is some “approximation” to \( N \). In our case

\[
N = \begin{pmatrix} S & UT \\ U & A \\ V & W \end{pmatrix} \quad M = \begin{pmatrix} S & 0 \\ U & A \\ V & C \end{pmatrix}.
\] (12)
Figure 4.: Final standard parallax errors of a converged AGIS solution that started from very conservative initial conditions for a simulated data set with two million primary sources. The cycle converged in 22 iterations and the result is in satisfactory agreement with theoretical predictions (see right-hand panel of Figure 2).

\[ M \] is called a Gauss-Seidel “pre-conditioner” of the iterative solution. In terms of the normal matrix \( N \) and pre-conditioner \( M \) the AGIS iterations can formally be written as:

\[
\begin{align*}
    x^{(0)} &= M^{-1}b \\
    x^{(k+1)} &= M^{-1}(b - Nx^{(k)}), & k = 0, 1, 2, \ldots
\end{align*}
\]

This is commonly known as a “simple iteration” scheme. The employment of more advanced techniques (e.g. Conjugate Gradient) in AGIS is presently under study.

3. Selected Results from Large-scale Tests

Large-scale validation tests using simulated data sets provided by DPAC’s CU2 are being executed on dedicated processing hardware at ESAC. From the first working version of AGIS, these tests continue to produce good results and an example is given in Figure 4. It shows an all-sky Hammer-Aitoff projection in ecliptic coordinates of the parallax errors (in the sense AGIS-computed minus true value averaged over all sources in a given pixel) resulting from a recent AGIS processing cycle using a dataset with two million single, astrometrically well-behaved stars. The cycle started from very conservative initial errors in the unknown source, attitude, and calibration parameters (50 mas Gaussian errors
plus a systematic variation with an amplitude of a few 10 mas). The cycle was
considered converged after 24 iterations when the width of the parallax update
distribution became less than 1 \( \mu \text{as} \). With final errors of less than 15 \( \mu \text{as} \) the
convergence process has reduced the initial errors by more than three orders of
magnitude. The end result is close to the goal of achieving residual astrometric
errors that are consistent with the observation noise and with remaining system-
atic errors much smaller than the random errors. The error pattern seen in the
map “echoes” the Nominal Scanning Law (see Figure 2) in the sense that the
solution is most rigid in the polar regions where the number of observations is
largest. In the area between ecliptic latitudes -45° and +45° the errors are higher
because of fewer observations contributing to the solution and a less favourable
scanning geometry.

4. Software Engineering Aspects

4.1. Architecture

The design of AGIS has already been presented at a previous ADASS conference
and only a brief summary is given here. AGIS is a distributed, data-driven pro-
cessing system written in pure Java. The decision to use Java was mainly driven
by the fact that the language natively supports all features that an optimal
processing framework will need, viz. parallelism with Threads and Interprocess
Communication with RMI. In addition DPAC provides an excellent and rich
general Java library \textit{GaiaTools} that AGIS makes extensive use of.

Central to the AGIS architecture is the notion of \textit{DataTrains} and \textit{DataTakers}.
Due to the very large data volume (10^{12} observations) that AGIS has to pro-
cess, I/O is a concern and must be kept to a minimum. This means that data
needed by more than one process should be read only once and then passed
around in memory: an \textit{AstroElementary DataTrain} reads observation data that
are grouped per source, i.e. all observations belonging to a given source are
stored in a group and read together. The S, A, C, (and G) algorithms are
registered on the \textit{DataTrains} as \textit{Takers} which means the train actively passes
the read data on to them. The source update (S) is executed within the same
thread as the \textit{DataTrain}. The A and C (and G) takers on the other hand are
only processing proxies which collect observations and send them via RMI to
special \textit{UpdateServers} which are likewise distributed and multi-threaded.

The entire processing task is broken down into jobs. Each job consists typ-
ically of all observation of about 1000 sources. At the beginning of an iteration
the processing jobs are published on a Whiteboard (a DB table) by a \textit{RunMan-
ger}. \textit{DataTrains} on different nodes are launched by a \textit{Launcher}. When they
start up they ask the Whiteboard for a job, commence the processing, and when
done ask for a new one. This continues until all jobs have been processed. At
this point the \textit{RunManager} signals the A and C servers to perform a final update
and then starts a new iteration by publishing a new set of jobs. This outer GIS
iteration loop continues until a \textit{ConvergenceMonitor} senses that the computed
updates have become smaller than a chosen limit. If that happens the system
is considered converged and the processing terminates. During the cycle the
convergence status can be monitored through a range of histograms and other
result sets that are accumulated in real-time by the *ConvergenceMonitor* and rendered graphically by a Web server via Java Server Pages.

### 4.2. Development Approach

AGIS is being developed in an agile-style over four European sites using a suite of common development tools: Mantis, Eclipse, JUnit, GaiaTools, etc. ESAC hosts a central Subversion repository where all code and documents are checked in. The continuous build environment “Hudson” is running at ESAC for building/unit testing of AGIS and other DPAC modules.

DPAC follows a cyclic development approach where software is incrementally built in cycles of six months duration. A new cycle \( N \) starts with a combined \( N - 1 \)-closeout/\( N \)-kick-off meeting where milestones are reviewed and defined. This is followed by the update of a number of central (ECSS) documents: requirements (SRS), design (SDD), test plans (STP), and development plans (SDP). When enough code changes accumulate during code development, releases \( M.n.p \) are made and these are used to carry out large-scale validation tests. The results of these are documented in Validation Test Reports (VTRs).

### 5. Running AGIS at ESAC

AGIS is being run at ESAC on a cluster consisting of 18 dual-processor, single-core Xeon blades, 8 dual-processor, quad-core Xeon blades, and a 5 TB Fibre Channel SAN as main components. This system gives a floating point performance of about 400 GFLOPS. On that system the run time of one outer iteration is typically one hour for \( 10^6 \) stars, so one cycle with 40 iterations takes about two days. The CPU usage is greater than 90% thanks to the data-driven architecture. A simple FLOP count model has been developed for AGIS and the large-scale validation tests are also used as input to that model. It currently evaluates to \( 1.4 \times 10^{20} \) for the creation of the final catalog and \( 2.2 \times 10^{19} \) for the final AGIS processing cycle. The last number forecasts a total run-time of the system, on a hypothetical 10 TFLOP operational machine of around 50 days. The FLOP count model is regularly updated.

### 6. AGIS in the Overall Gaia Processing Concept

AGIS is embedded into an overall complex Gaia data processing concept and architecture. DPAC is investing significant effort and resources to implement this system until launch in late 2011. The part with most relevance for AGIS is the Initial Data Treatment (IDT) which converts raw telemetry into higher-level input data products for AGIS. See O’Mullane et al. (2009) in this volume for an overview of the Gaia data processing architecture.

### References