
A New Software Package For Computing The Time-Dependent Aperture Of The Auger Surface Detector

Aaron S. Chou

Fermilab, P.O. Box 500, Batavia, Illinois, 60510-0500, USA

Abstract

We describe a new general-purpose software package being developed for the Pierre Auger Collaboration for rapidly computing the differential and time-integrated apertures of regular arrays of surface detectors.

1. Introduction

A precise determination of the integrated aperture of the Pierre Auger Surface Detector (SD) is necessary in order to accurately convert the measured counts of high energy cosmic ray events into flux measurements. The aperture may be viewed as an efficiency for events to satisfy detector trigger requirements as well as any further event selection criteria which may be imposed for the needs of each particular analysis. Because these selection cuts may be very different depending on the analysis topic, the aperture should be re-computed for each specific application. With this goal in mind, we have developed a general-purpose software package for computing the aperture by using simulations of cosmic ray air showers and detector responses. The software allows the user to specify various parameters of the computation, including the detector simulation, the array geometry, the set of showers to be studied, and the selection criteria.

The speed of the computation is of utmost importance. For example, in a naive approach, one could throw 50K simulated showers over each local region of the entire array and compute the detection efficiency. Even with fast dedicated detector simulation programs like SDSim [1] for example, a detector simulation for a single shower takes ~ 1 cpu-minute. For the full Auger array of 1600 detectors, the full computation over each of the 1600 local regions would take ~ 1500 cpu-years. Furthermore, each time a detector or localized group of detectors goes offline, another computation of at least ~ 35 cpu-days would be needed to assess the local inefficiency in this region. Clearly a more clever approach is needed.

Operated by Universities Research Associates Inc. under contract DE-AC02-76CH03000 with the U.S. Department of Energy. Travel supported by a grant from the American Astronomical Society.

2. The Method

In our algorithm, we take advantage of the translational symmetry of a regular array of detectors. For the case of the Auger SD, the entire array can be tiled with diamond-shaped regions. For an array of 1600 detectors, 1600 local diamonds may thus be defined. Because of the array symmetry, diamonds which are sufficiently far away from the array boundary are more or less equivalent. To treat the array boundaries, the exterior of the array may also be tiled with diamonds, with each diamond site containing a non-existent detector. Functionally, these non-existent detectors are equivalent to offline detectors within the array. With this view, for the purposes of computing the local array efficiency, all diamonds may be treated equivalently by the software whether they are well within the array, near the boundary, or even outside of the array.

Therefore, instead of simulating a large set of showers impacting inside each diamond region, it suffices to simulate showers impacting within a single diamond, and later map this region to the each diamond in the vicinity of the physical array. In order to reproduce the most ideal conditions, these detector simulations are performed within a faux array of perfectly functioning detectors. The extent of the faux array is chosen to be large enough to account for all possible detector hits. Because only one local diamond region is considered, the total simulation time for a set of 50K showers is only only ~ 35 cpu-days. The collection of shower/detector simulations is stored in a database, in the same format as the real data from the observatory.

This simulated diamond is mapped to diamonds inside or nearby the physical array. Diamonds far outside of the array may need to be considered as well for analyses of large zenith angle showers. The aperture can now be determined by computing the array efficiency locally, diamond by diamond. With each diamond mapping, the hit detectors in the faux array are identified with a set of real detectors, and a calibrations database may be accessed to make appropriate modifications to the idealized simulated data. For example, when mapping to a diamond near an offline or non-existent station, the simulated data corresponding to that bad station's position is thrown away. Similarly, it may be possible to make corrections to the simulated data for real detectors with known peculiarities. For example, one can make a signal level correction to detectors which are not precisely on a regular grid position.

After all corrections are applied to the ideal simulated data, the event selection criteria is then evaluated. By repeating this process with each shower/detector simulation in the collection, a local event selection efficiency can be computed. It is important to emphasize here that because the simulated data is stored in the same format as the real data, exactly the same event selection procedure can be applied to each. For example, one can even fully reconstruct the showers and place selection cuts on the χ^2 of the fit.

3. An Example of an Aperture Computation

For our study, the above procedure is performed with a set of 25K AIRES [3] showers generated at Fermilab by Sergio Sciutto. Because the air shower simulation is very computationally intensive, this library of showers is pre-generated and stored on tape. The showers in the library are generated using a discrete set of shower parameters including energy, zenith angle, azimuthal angle, primary particle, and hadronic model. The local diamond efficiencies are calculated separately for each particular combination of shower parameters. Thus the aperture can be obtained as a function of direction, as would be necessary for an anisotropy analysis, and later integrated for a energy spectrum analysis.

To simulate the water Cherenkov tanks comprising the Auger SD, the SDSim [1] package is used. SDSim also simulates the trigger and data acquisition electronics of the detectors. For a simple study of trigger efficiency, we choose the event selection criteria to be the simultaneous trigger of three separate tanks used by the Auger Engineering Array [2]. Each individual tank can be triggered either by large PMT signals or by PMT signals extended in time. In addition, we impose a requirement that the core of the shower must be within the boundary of the array. The detector simulations are performed with the Fixed Target Farm, a 200 cpu (~ 1 GHz) Linux cluster at Fermilab.

Showers from the library are thrown with random impact points inside the unit diamond in the center of a faux array of 100 tanks. In order to increase the statistics, each shower is reused by throwing it into 5 random positions. The output of the detector simulations is stored in the real Auger data format in a shower/detector simulations database. Whereas the original AIRES shower format required ~ 20 MB of tape space per shower, this detector response data takes only ~ 20 kB of disk space. With this great reduction in required storage space, rather than requiring a storage tape silo with a tape robot, the entire simulations database can now fit easily on the hard disk of a typical laptop computer! The local efficiencies can now be rapidly computed with algorithmic complexity linear in the number of showers, the number of diamond regions, and the average number of triggered tanks per shower.

To obtain the aperture, the area integral is done by summing the local diamond efficiencies and multiplying by the area of the unit diamond. Integrals in zenith and azimuthal angles may be performed by summing the angular bins with appropriate weights. The results of our sample calculation are shown in Figure 1. Alternatively, an efficiency function may be obtained by interpolating between the efficiency values computed at the discrete angle values of the showers in the library. This efficiency function may then be numerically integrated. Similarly, the aperture as a function of energy may be obtained by interpolation between the calculated efficiencies at discrete values of energy.

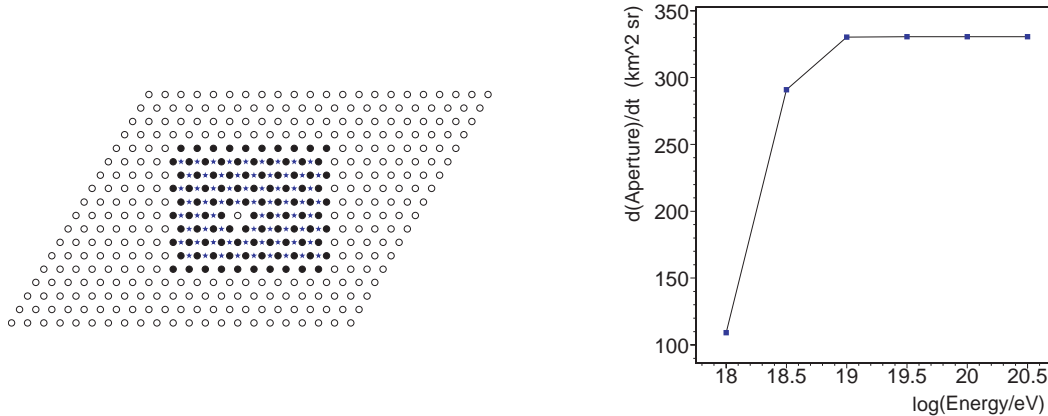


Fig. 1. Left: A single missing tank in causes inefficiencies in nearby diamonds. The black dots represent working tanks in a mini-array with 1.5 km spacing, the empty dots represent missing tanks, and the stars represent diamond regions with efficiency greater than 80% for 3 EeV small zenith angle showers. Right: The computed aperture restricted to zenith angles smaller than 60 degrees for this mini-array.

4. Computing the Time Dependence

The time dependence of the aperture may be calculated by interfacing a calibrations database containing detector status data as a function of time. For example, a detector on/off bit can be used to determine whether to throw away the simulated data which has been mapped to the that detector. Each time the status of a detector changes, the local array efficiency in a neighborhood of that detector can be recomputed. In the case of the example above of computing the trigger efficiency, this recomputation simply consists of recounting the number of triggered tanks. However, for more complicated selection criteria like shower reconstruction based cuts, the recomputation may take a significant amount of time. For example, if a shower causes N detectors to trigger, there are in principle 2^N detector on/off combinations which would affect the reconstruction of this shower. Performing a 10 cpu-second reconstruction on each of these combinations for each of the showers may not be feasible. Most of these combinations would never be seen in a real array however, and it would be sufficient to perform new reconstructions only when a new detector status combination is seen in the array. The results of the reconstruction can then be stored and referenced later when the same combination is seen again.

5. References

1. Dagoret-Campagne, S. 2002, Auger technical note GAP 2002-072
2. Pierre Auger Collaboration 2003, Submitted to Nucl. Instr. and Meth. A
3. Sciutto, S. 1999, Auger technical note GAP 1999-044, GAP 2002-047