LIKELIHOOD FREE PARTICLE FILTERING WITH APPROXIMATE BAYESIAN COMPUTATION FOR PARAMETER ESTIMATION IN COSMIC RAY AIR SHOWER STUDIES

by

M. Kutay Yabaş

B.S., Physics, Boğaziçi University, 2017

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computational Science and Engineering

Boğaziçi University

2020

ACKNOWLEDGEMENTS

I would like to thank to my thesis supervisor Prof. M. Levent Kurnaz for helping me with this research. I would like to express my sincere gratitude to Prof. V. Erkcan Özcan. His motivation, enthusiasm and immense knowledge about physics motivated and provided an objective for me in pursuing this degree.

ABSTRACT

LIKELIHOOD FREE PARTICLE FILTERING WITH APPROXIMATE BAYESIAN COMPUTATION FOR PARAMETER ESTIMATION IN COSMIC RAY AIR SHOWER STUDIES

Highly energetic particles called cosmic rays arrive at earth and interact with the atmosphere to cause other particles to emerge and fuel a chain interaction of particle production until the energy is dissipated. These cascades emerging from a highly energetic initial particle and producing secondary particles that spread over a large area at the ground are called Extensive Air Showers. These showers are studied to find anisotropy in their arrival direction and energy to unveil their source and production mechanisms in the universe. In this study we utilize the likelihood free particle filtering with Approximate Bayesian Computation to estimate the incident angle and energy of the primary particle. ABC method makes use of comparison between simulated and observed summary statistics to overcome the problem of computationally intractable likelihood function of particle physics.

ÖZET

OLABİLİRKLİK MODELSİZ PARÇACIK FİLTRESİ İLE KOZMİK IŞIN ÇALIŞMALARINDA BAYESÇİ PARAMETRE KESTİRİMİ

Kozmik ışınlar olarak adlandırılan yüksek enerjili parçacıklar, dünyaya ulaşarak atmosfer ile çarpışırlar. İlk çarpışma yeni parçacıklar üreterek zincirleme bir etkileşimi tetikler. Çarpışmalar sonucu ortaya çıkan son parçacığın etkileşim için yeterli enerjisi kalmadığında zincirmele etkileşim son bulur. Çok yüksek enerjili bir ilk parçacık tarafından başlatılan ve üretilen ikincil parçacıkların yeryüzünde geniş bir alana yayıldığı kozmik ışın gözlemlerine parçacık sağanağı denir. Kozmik ışınların kaynaklarının ve üretim mekanizmalarının keşfi amacıyla parçacık sağanaklarını oluşturan ilk parçacığın atmosfer ile etkileşim açısı ve enerjisi bakımından eşyönsüzlükler araştırılmaktadır. Bu çalışmada kozmik ışının enerjisini ve vaka açısını, olabilirlik modeli bulunmaksızın Bayesçi yöntemle tespit ediyoruz. Bu sayede hesaplamalı olarak çözülmesi imkansız veya zor olan olabilirlik modellerine sahip parçacık fiziği problemleri için alternatif bir çözüm yöntemini ortaya koyuyoruz.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

A_0	Atmospheric Depth
c	Speed of Light
D	Simulated data
D^*	Observed data
eV	Electron Volt
E	Energy
E_0	Incident Energy
E_c	Critical Energy
g	Calibration Constant
$\mathbb{I}$	Iteration Number
k	Shower Front Velocity Vector
l	Multiplier of x at the Function of a Plane
m	Multiplier of y at the Function of a Plane
$\mathcal{K}$	Probability Kernel
$\mathbb{L}$	Likelihood
N	Number of Particles
$\mathcal{N}$	Gaussian Distribution
N_{ch}	Multiplicity Constant of Charged Pions
N_{em}	Number of Particles Generated by Electromagnetic Interaction
N_μ	Number of Muons
N_π	Number of Pions
N_v	Vertical Equivalent of Number of Particles
p	Probability
r	Distance from the Shower Core
s	Summary Statistic
$S(x)$	Relative Muon Count at Distance X
$\mathbb{S}$	Summary Statistic Function
t	Time

$\mathbb{T}$	Iteration Number
v	Multiplier of z at the Function of a Plane
w	Weight
x	Location on the X Axis
y	Location on the Y Axis
z	Location on the Z Axis
ε	Acceptance Distance
θ	Zenith Angle
κ	Model Parameter
λ	Interaction Length
ρ	Distance Function
Φ	Azimuth Angle
χ	Minimization Variable
ω	Observation

LIST OF ACRONYMS/ABBREVIATIONS

ABC	Approximate Bayesian Computation
CORSIKA	Cosmic Ray Simulations for Kascade
CR	Cosmic Ray
CRs	Cosmic Rays
EAS	Extensive Air Shower
EM	Electromagnetic Shower
Kascade	Karlsruhe Shower Core and Array Detector
LDF	Lateral Distribution Function
LHC	Large Hadron Collider
MCMC	Markov Chain Monte Carlo
PA	Pierre Auger
SD	Surface Detector

1. INTRODUCTION

1.1. Cosmic Ray Air Showers

In the beginning of 20th century scientists realized that there were more radiation on the ground than there should be. This phenomenon could not be explained by the known natural sources. Speculations were made to indicate that the Sun may be the primary source of this radiation. To search for the origins of this excess radiation scientists made observations at different altitudes. In 1909 German scientist Theodor Wulf measured the difference in ionization between ground and near the top of Eiffel tower [3] using an electroscope. Later, going beyond the 300 meter height of Eiffel, Victor Hess made a balloon flight in 1911 and ascended to 1100 meters with his electroscope. Both Wulf and Hess did not observe significant changes in ionization. Discovery of the cosmic rays originated with Hess' balloon flight during a solar eclipse in 1912 [4]. Hess wanted to rule out solar radiation as a source while observing possible cosmic ray sources arriving from the space. He turned out to be right and realized that the level of radiation intensified 3 times, as he climbed up to near 5300 meters. After Hess' discovery, this radiation originating from space started to be mentioned as Cosmic Rays (CRs).

After the discovery of CRs about 100 years ago, today we know that the energies of the particles arriving at Earth can range from around $10^6 eV$ to as high as $10^{20} eV$. Around $10^{14} eV$, the frequency of events falling on solo detectors simply becomes too small because of the rapid decrease of flux with increasing energy. Instead, secondary particles spread over the ground can be observed by surface detectors such as scintillators and water Cherenkov tanks. This phenomenon of simultaneous arrival *"of many particles over a large area is called an Extensive Air Shower (EAS)"* [4]. At $10^{15} eV$ around 10^6 particles cover approximately $10^4 m^2$ and at $10^{20} eV$ around 10^{11} particles can spread over $10 km^2$.

We are curious about the origins of these highly energetic particles. While we

have so far reached $1.3 \times 10^{13} eV$ to study proton collisions at Large Hadron Collider, chance to observe natural collisions up to $10^{21} eV$ is intriguing. CR studies enable us to test our hadronic interaction models surpassing the energy range that can be tested by man made particle colliders [5]. For this reason there are many ongoing experiments to study EAS at various experiments like Pierre Auger Observatory [6], KASCADE-Grande [7], IceTop [8], AGASA [9] and GAMMA [10]. All these experiments try to accurately estimate the CR flux curve relative to primary particle energy causing the CR. With this object in mind such experiments put ground particle, air Cherenkov and fluorescence detectors in use for observation. Collected data from the detectors are reconstructed to unravel the initial properties of the particle causing the CR in the first place.

From the data collected by CR observatories around the world so far, we know that the rate of arrival of CR decreases with increasing energy. While particles with less energy arrive with a flux of 10^4 per square meter per second at $10^9 eV$, highest energy particles can have a flux of just one incident per square kilometer per century. If we take a look at the aggregated results of these observations in Figure 1.1, where many measurements of the cosmic ray flux over a wide energy range assembled by Gaisser [1], we can see that flux is roughly in proportion with the logarithm of Energy E, specifically with $\propto E^{-2.7}$.

This incidence spectrum can be divided into three regions by way of detection. We can detect the first region, particles with energies up to $10^5 eV$ by observation satellites revolving around the Earth and high altitude balloons directly, because of the high flux in this energy range. Such particles can not reach the surface as their energies dissipate by interactions with the atmosphere.

Second energy range starts roughly from $10^5 eV$ and continue up to $10^{14} eV$ level. Such energetic particles cause particle showers in the atmosphere and as a result create fluorescence in the air. Fluorescence can be detected in clear night sky and longitudinal development of the air shower can be observed. However, secondary particles created by the primary particles in this energy range still can not reach the ground.

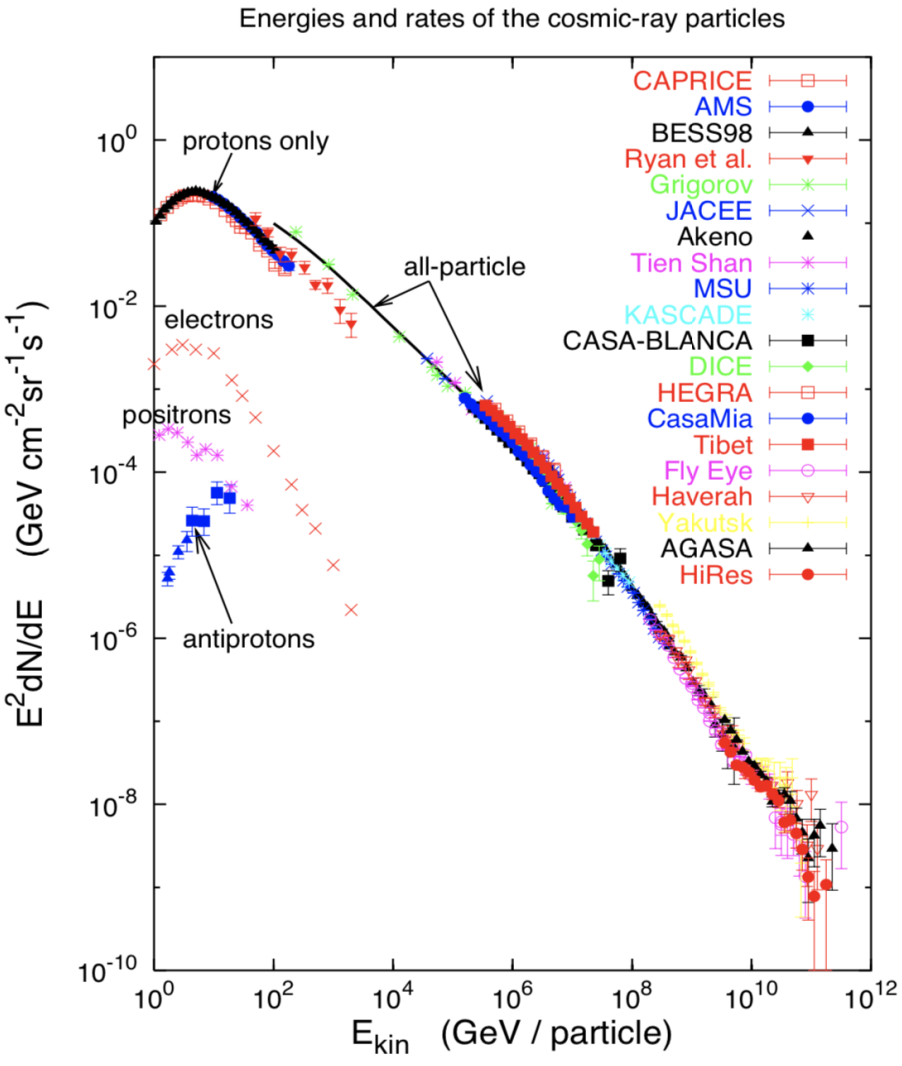

Figure 1.1. Energies and Rates of Cosmic Ray Spectrum [1]

Beyond $10^{15}eV$, the shape of the flux curve changes significantly. As can be seen in Figure 1.1, CR flux forms a knee and flattens after around $10^{15}eV$. It is postulated that such change is caused by the transition of primaries from galactic to extra galactic origin [11]. There can be various reasons like the reduced impact of galactic magnetic field over highly energetic particles; the physical limits of acceleration of proton [5] in supernova remnants, or perhaps a new particle creation with unusual behavior at high energy interactions. Hence the flux rate curve yields many questions to be answered, which makes it important to find alternative methodologies to study highly energetic particles.

1.2. EAS Reconstruction and Analysis

EAS studies are designed to identify the acceleration mechanisms and the origins of CRs in the universe. Detecting the incident parameters becomes important in order to find out potential anisotropies in the arrival direction. The accurate determination of tthe energy is also critical to explore the universe, as CRs can be deflected by intervening magnetic fields. At high energies deflections are expected to decrease and CRs may be accompanied by Gamma Rays, so we can pinpoint them back to their true sources with efficient reconstruction techniques [5].

We are interested in the arrival angle, energy and particle composition of the primary particle initiating the EAS. Interpretation of EAS requires accurate and efficient methodology of reconstruction of the collected secondary particle data.

1.2.1. Reconstruction of Incident Energy

Heitler suggests a formula for the rough estimation of the electromagnetic (EM) shower parameters [12]. He proposes a simplified shower model by assuming that a single initial particle causes the cascade. With each interaction only two new particles emerge and energy is halved for each new particle. In in each iteration along the atmosphere, the interaction length λ assumed to stay constant. Repetition of this chain interaction continues until the last particle drops below a critical energy E_c and

can no more interact with the medium to produce new particles. Thus, total number of particles created is $N = 2^n$ where n is number of interactions and energy of the last created particle is $E = 1/2^n$ which roughly equals the critical energy E_c. Then we can see that $E_0 = NE_c$ and number of particles generated basically equals to E_0/E_c. Hence, there obviously is a linear relation between N and the primary particle energy.

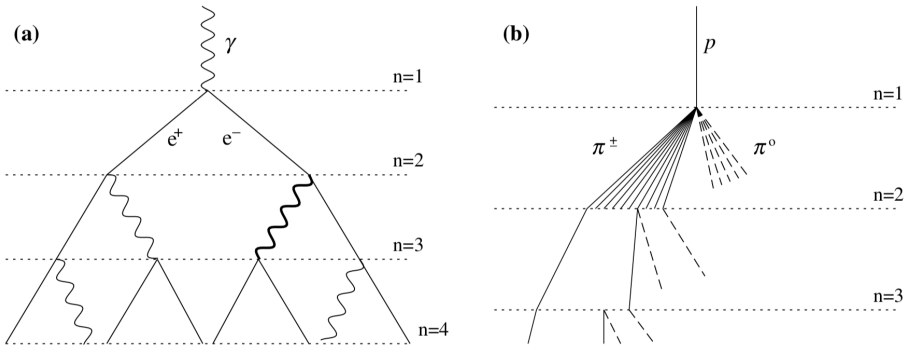

Figure 1.2. Schematic views of (a) an electromagnetic cascade and (b) a hadronic shower simplified as in Heitler's interaction model [2]

Hadronic showers are more complicated where neutral pions emerge and quickly decay without interaction. Matthews [2] adapts Heitler's basic model and derives methodology to reconstruct the initial particle energy E_0 in a hadronic EAS. π^+ and π^- travel λ and produce new pions with interaction. Similar to an electromagnetic (EM) shower, the chain production continues until a critical energy E_c but differently from pure EM showers, 1/3 of the energy is lost due to π^0 initiated electromagnetic interactions.

Thus, particle productions initiated by charged pions account for the total energy of $(\frac{2}{3})^n E_0$. In addition, charged pions have multiplicity constant of N_{ch} to account for emerging neutral pions. Therefore, the energy of a pion given at interaction number n is;

$$E_\pi = \frac{E_0}{(\frac{3}{2}N_{ch})^n} \tag{1.1}$$

When critical energy is reached, there are N_π pions and N_{em} particles generated due to EM interactions. We assume that, when charged pions can no longer interact to create new particles, they decay to muons. Thus, the number of muons N_μ is equal to number of pions N_π. In this case the primary energy can be estimated as;

$$E_0 \simeq E_c^e N_{em} + E_c^\pi N_\mu \tag{1.2}$$

These equations are parameterized, then adjusted with simulations of EAS falling over the detecting grid. This way of reproducing the real world experiment conditions yield results that are as close as possible to the ground truth. The governing idea here is that the initial energy can be approximated linearly by the final count of particles.

1.2.2. Reconstruction of the Shower Center

In a CR experiment, detector grids cover a large area to detect as many of the generated particles as possible. Thus, the EAS center does not always overlap with the center of the surface detector grid. Besides, lateral particle distribution of the EAS on the ground may be larger than the span of the limits of the detector grid. Therefore, a center should be determined for every EAS incident to make it comparable with other observed incidents.

We know the different lateral distribution characteristics of the secondary particles at different primary energy levels. This allows us to fit the observed lateral

distribution to the ground truth and interpolate the shower center even if it is not directly observed with any surface detector. In other words, ratio of signal strengths at different surface detector locations enables us to estimate the shower center.

1.2.3. Reconstruction of the Arrival Angle

The other interesting variable to reconstruct is the arrival angle to assess the CR source of creation and to study anisotropy. CR shower front moves roughly as a disc. The inclination of the disc determines the time of signal generation at a ground detector. So, the arrival angle can be deduced through the solution of a geometry problem.

Let us assume that shower front moves along the vector k of primary particle, towards the detector with speed of light c. Secondary particles are detected by the surface detectors and t_i is determined to be the time of signal generation at the ith detector where n signals are generated. The shower front will be observed by the intersection of the particles in the front plane with detectors. In the simplest form the equation of the plane is;

$$ct' = ct_0 - lx - my - vz \tag{1.3}$$

where t' is the time of arrival of the plane at the point x, y and z. We assume that detector altitudes are equal and set $z = 0$ and form the function;

$$\chi^2 = \frac{1}{n-2}\sum_{i=1}^{n} c^2(t_i - t_i')^2 = \frac{1}{n-2}\sum_{i=1}^{n}(ct_i + lx_i + my_i - ct_0)^2 \tag{1.4}$$

and minimize χ^2 by taking the following partial derivatives.

$$(\sum x_i^2)l + (\sum x_i y_i)m - (\sum x_i)ct_0 = -\sum x_i ct_i$$

$$(\sum x_i y_i)l + (\sum y_i^2)m - (\sum y_i)ct_0 = -\sum y_i ct_i$$

$$(\sum x_i)l + (\sum y_i)m - nct_0 = -\sum ct_i$$

Calculated values of l and m yield the direction of vector k. The zenith angle θ between shower axis and the vertical direction is related to l and m by the equation;

$$\theta = \arcsin\left[(l^2 + m^2)^{1/2}\right] \tag{1.5}$$

The azimuth Φ is given by;

$$\Phi = \arcsin\left[l/(l^2 + m^2)^{1/2}\right] \tag{1.6}$$

At least three different observations are required for the reconstruction of the arrival angle. One could also account for detectors located at different altitudes and curvature of the shower front. Consequently, foundation of arrival angle reconstruction is time difference in signal generation.

1.2.4. Reconstruction in Practice

If we were able to collect every and each particle that reached to the ground level, it would be trivial to compare lateral distribution of the observed particles to the lateral distribution function (LDF) of an EAS simulation. However, the EAS detector setups are like sparse matrices where there are more unobserved bins than the observed ones. That being the case, reconstruction techniques at different detector grid setups must be calibrated with simulations.

For instance, Pierre Auger experiment utilizes $S(1000)$ and $S(600)$, the reconstructed signal at 1000 and 600 meters from the shower center and N_{19} the relative muon content of an observed shower compared to mean muon content of a $10^{19}eV$ sim-

ulated proton shower [13]. $S(1000)$ and N_{19} estimators are calibrated by simulations to represent the characteristics of the Pierre Auger surface detector setup. Another EAS experiment AGASA uses the same technique with $10^{17}eV$ muon content comparison and $S(600)$ distribution [14].

We can generalize the formula of energy estimation for practical application as;

$$E_0 = g \times N_d S(r) \tag{1.7}$$

where g is a calibration constant, N_d is the LDF of an d degree simulation and $S(x)$ where the muon count is observed at r meters from the shower core.

Different parameters may be selected for the best fit. To illustrate, $S(600)$ and $S(1000)$ are used at the same experiment but for EAS arriving with different zenith angles or lower energy levels for better accuracy [15].

1.2.5. CORSIKA Simulations

CORSIKA [16] is a program for detailed simulations of the extensive air showers. It has been developed at Karlsruhe Institute of Technology for the simulations required at Karlsruhe Shower Core and Array Detector experiment. Later on, it became the state of the art EAS simulator put in use to design detector setups, to develop reconstruction techniques and to assess accuracy of the experiments. CORSIKA can treat protons, light and other heavy nuclei as primaries. The primaries are tracked through the atmosphere and their interactions are described by electromagnetic and hadronic interaction models. Particle physics simulators are essential for calibration of an experiment and benchmarking between the experiments. By this means, we can aggregate the reconstructed data from different experiments to form a cumulative knowledge about CRs.

Simulations harbor all of our understanding about particle physics but can not simply portray the ground truth. With regards to this, J. Knapp who is a developer

of particle physics simulator CORSIKA, states that simulators may not be perfect but can be used for comparison of the performance of models [16] by saying: *"Is the composition changing or not? The answer depends on the yardstick (i.e. the simulation program) used for comparison. Use the same yardstick to get consistent results, use a well-calibrated yardstick to get the correct results"* [17]. Perfecting the *yardstick* with experiments while evaluating an experiment with the very same *yardstick* is an iterative process where we extend our knowledge base of particle physics.

1.3. Approximate Bayesian Computation

Approximate Bayesian Computation (ABC) is known as a likelihood-free Bayesian parameter estimation method. It was first used in population genetics, then on different applications with computationally hard to solve problems [18]. This method makes use of comparison between simulated and observed summary statistics to overcome the problem of computationally intractable likelihood functions [19]. ABC is a promising alternative when solution can not be found and we have a forward simulation model of the problem. This opens the possibility to perform Bayesian analysis for any model that can be simulated [20].

In intricate problems such as particle physics problems, the likelihood function contains the mysteries of the universe in the shape of quantum probabilities. Particle physics experiment data sets are complex and even the simplest theoretical models have many nuisance parameters because of uncertainty. While it is very unlikely for us to solve the likelihood function in this case, we can run simulations and compare the simulation results with the observed data to infer a posterior for the initial parameters.

We can express the likelihood function of an intricate problem as a multidimensional integral [18]. For an EAS problem the likelihood becomes;

$$\mathbb{L}(\kappa|\omega) = \int \mathbb{L}^*(\kappa|\omega, u)du \tag{1.8}$$

where $\omega \in D \subseteq \mathbb{R}^n$ is observed, $u \in \mathbb{R}^p$ a latent vector and $\kappa \in \mathbb{R}^d$ the parameter of

interest. In our case, ω is the observations of physical particles on the ground of an EAS event, while u stands for quantum electrodynamics, nuclear interactions, detector geometry and κ for the parameters initiating the EAS. So, the EAS problem becomes solvable with ABC methodology.

If the input parameter κ can produce simulated data D that is close to the observed data D^*, then there is a nonzero probability that κ generated the observed data. This probability varies by the distance between simulation and the observation. Closer κ have high probability and further κ lower, which forms the distribution of the posterior [20].

Based on this idea the very primitive rejection Algorithm 1.3 has been developed by Robert and Casella [21], which is the ancestor of the ABC algorithm, where a parameter of interest is generated through a prior and the acceptance is conditional on the corresponding simulation of a sample being almost identical to the observed sample [18].

for $i = 1$ to $\mathbb{I}$ **do**
  repeat
  Sample κ' from the prior distribution $p(\kappa)$
  Generate simulation D from the simulator $p(D|\kappa')$
  until $D = \omega$
  set $\kappa_i = \kappa'$
end for

Figure 1.3. Likelihood Free Rejection Sampler.

This primitive algorithm yields outcome of posterior distributions of the parameters $\kappa \propto p(\kappa_i|\omega)$.

In brief, with Approximate Bayesian Computation methods we replace the intricate likelihood function with a simulation program. In this case we run the simulation

program CORSIKA with properly selected priors to estimate the posteriors. In the most basic form of this calculation method, we apply rejection sampling.

1.3.1. Rejection Sampling

All ABC-based methods approximate the likelihood function with simulations, then simulation outcome is compared with the observed data. The most simple form of this methodology is named as ABC Rejection Sampling [22]. Figure 1.4 explains the procedure with a diagram.

Starting with a set of input parameters κ and its associated priors $p(\kappa)$, we draw sample parameters κ^* to run a simulation program with these input parameters. Simulation yields data $D^* \sim p(D|\kappa^*)$. Then, we compare the true data D with the simulated data D^* with a distance function ρ where $\rho(D^* - D) < \varepsilon$ is less than a threshold ε_0.

This sampling procedure draws samples from $p(\kappa|\rho < \varepsilon_0)$, which in case approximates the parameter posteriors. The rejection tolerance level ε_0 can be optimized by creating train and test data-sets and minimizing the error between true data and the approximated posterior.

The performance of the approximation is closely related with the distance function and the summary statistics used to calculate the distance. While basic or weighted Euclidean distance can be used, there are studies separating the observed and true data with Support Vector Machine hyperplanes [23]. Generating efficient summary metrics and implementation of adequate tolerance metrics are open research areas. In this thesis, we try to adapt the practical methodology used in Cosmic Ray Observatories to work with ABC.

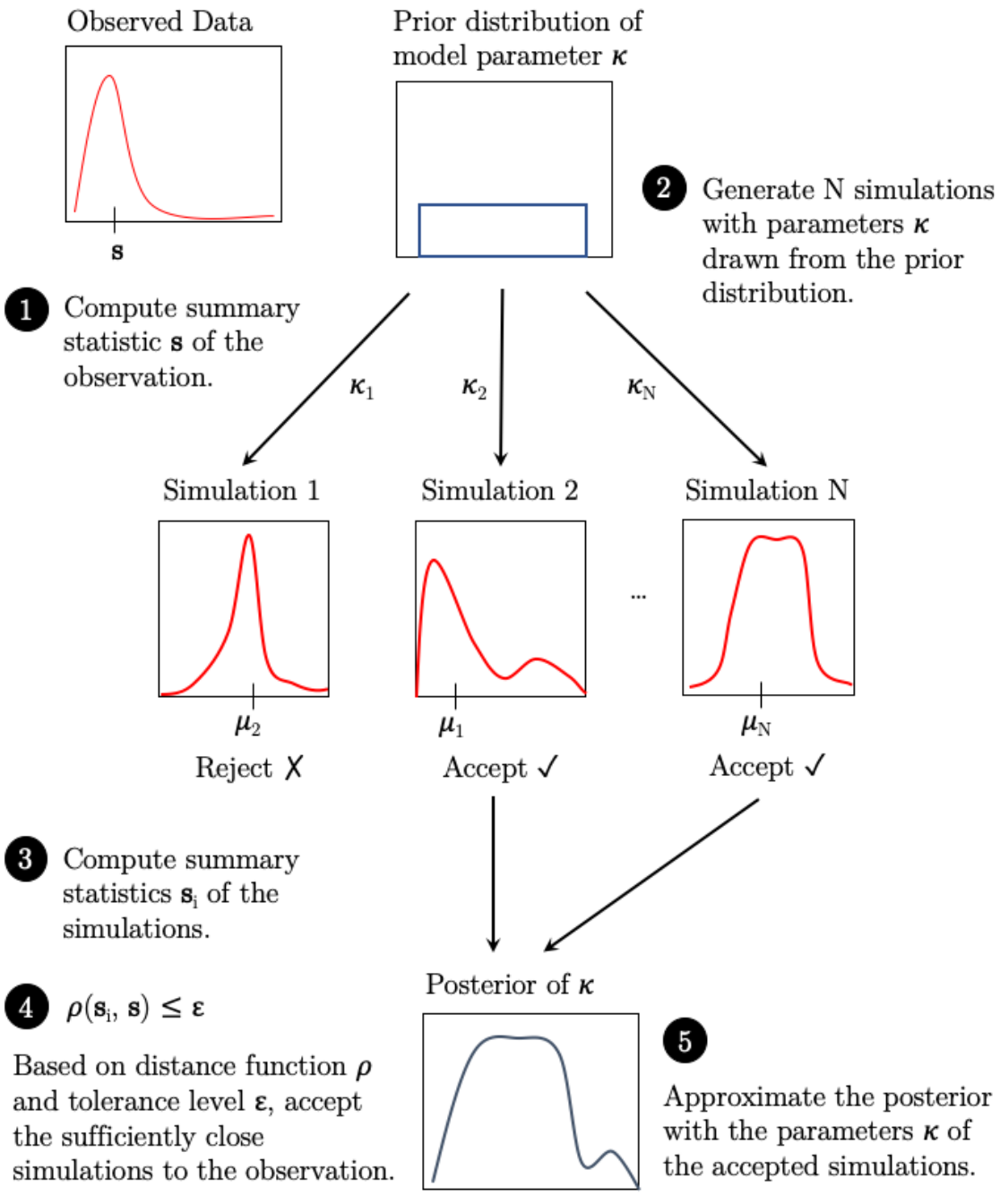

Figure 1.4. ABC Rejection Sampling Methodology Diagram

Set tolerance level ε_0
for $i = 1$ to $\mathbb{I}$ **do**
  Sample κ' from prior $\kappa \sim p(\kappa)$
  Simulate data $D^* \sim p(D|\kappa')$
  Calculate distance metric $\varepsilon = \rho(D, D^*)$
  if $\varepsilon < \varepsilon_0$ **then**
    accept and add D^* to posterior
  end if
end for

Figure 1.5. ABC Rejection Sampling Algorithm.

1.3.2. Sequential ABC

As Jennings suggests and implements [24] the ABC can be run as part of a Monte Carlo particle filter. Instead of generating huge amount of simulated data D^* at once, we can design a Markov Chain Monte Carlo (MCMC) algorithm which begins to simulate with a lot less number of parameter sets (i.e. each set of parameter is a particle in a *particle filter*) and refine the posteriors with each iteration. This can speed up the ABC algorithm by working with large pools of candidate simulations simultaneously, rather than drawing candidates one at a time. At each stage of the algorithm, the particles are perturbed and filtered using the distance metric. Eventually, this pool of particles converge to the desired posterior distribution. This approach is known as Sequential Monte Carlo or Particle Filter Monte Carlo sampling [24].

In this case, each iteration will yield posteriors for parameters and these posteriors will be used as priors of the next iteration as can be seen in Figure 1.6. The one disadvantage in this methodology may be that it can get stuck if none of the particles are accepted. To overcome this problem and to increase the acceptance probability, a weighing kernel such as a Gaussian kernel is used and randomness is added to particles by perturbation.

At iteration t=0:

for $i = 1$ to $\mathbb{I}$ **do**

  while $\rho(D, D^*) > \varepsilon_0$ **do**

    sample κ' from prior $\kappa \sim p(\kappa)$

    simulate data $D^* \sim p(D|\kappa')$

    calculate distance metric $\rho(D, D^*)$

  end while

  set $\kappa_{i,0} \leftarrow \kappa'$

  set weights $w_{i,0} \leftarrow 1/\mathbb{I}$

end for

set covariance $\Sigma_0^2 \leftarrow 2\Sigma(\kappa_{1:N,0})$

At iteration $t > 0$:

for $t = 1$ to $\mathbb{T}$ **do**

  for $i = 1$ to $\mathbb{I}$ **do**

    while $\rho(D, D^*) > \varepsilon_t$ **do**

      sample κ' from previous iteration. $\kappa \sim \kappa_{1:N,t-1}$ with weighted probabilities of $w_{1:N,t-1}$

      perturb κ' by sampling $\kappa'' \sim \mathcal{N}(\kappa', \Sigma_{t-1}^2)$

      simulate data $D^* \sim p(D|\kappa'')$

      calculate distance metric $\rho(D, D^*)$

    end while

    set $\kappa_{i,0} \leftarrow \kappa''$

    set weight $w_{i,t} \leftarrow \dfrac{p(\kappa_{i,t})}{\sum_{j=1}^{N} w_{j,t-1}\mathcal{K}(\kappa_{j,t-1}|\kappa_{i,t}, \Sigma_{t-1})}$ using kernel $\mathcal{K}$

  end for

end for

Figure 1.6. ABC Sequential Monte Carlo Algorithm.

Intractable problems often come with high-dimensional data and this is another problem that can lower the acceptance rates or increase the inference time substantially. As a solution we compare the summary statistics of the data and the distance metric becomes;

$$\rho(\mathbb{S}(D) - \mathbb{S}(D^*)) = \left(\sum_i \left(\frac{\mathbb{S}(D)_i - \mathbb{S}(D^*)_i}{\sigma_i} \right)^2 \right)^{\frac{1}{2}}$$

The algorithm runs until a desired level of acceptance level, a predefined variance for the posterior or preset iteration number $\mathbb{T}$ is achieved.

1.3.3. Choosing Summary Statistics

Summary statistics is the cornerstone of ABC methodology. In first place, the reason for us to resort to ABC is the high dimensionality of the selected problem. Summary statistics enable us to reduce the dimensionality. At the same time summary statistics must implicate distinctive information about the relation between the prior and the posterior. Thus, there is a critical balance where one should use as few statistics as possible while maintaining the entropy at the lowest. If we select too many summary statistics, we will not be able to escape from the curse of dimensionality. But, if we select so few or non-informative summaries, then we will not be able to generate accurate posteriors.

Prangle reviewed the literature for selecting summary statistics for ABC models [25]. The trivial method is to generate a training set of simulations and selecting the best performing candidates of summary statistics. While this is a common way to perform ABC Rejection, in some intricate problems the train set may not be able to capture all possible conditions.

Another methodology, *subset selection* requires candidate summary statistics to start with. The candidates are crafted with domain knowledge. Then, these candidates are introduced to the ABC algorithm until the accuracy saturates, then removed one by one until the accuracy is not significantly decreased. This parameter introduction method is called *Approximate Sufficiency*. Likewise, one can try to minimize the entropy or search for the summaries with minimum variance and maximum derivative over the posterior. These methods can be validated by generating simulations and implementing train/test procedure over the model.

Projection methodologies include dimension reduction techniques such as partial least squares, linear regression and boosting. The aim is to produce uncorrelated linear combinations of vector data features. Then, these combinations are used as the summary statistics.

2. COMPUTATIONAL EXPERIMENTS AND RESULTS

So far, we have discussed the state of the art methodology used to reconstruct the Cosmic Rays and suggested that Approximate Bayesian Computation can be a solid alternative to solve the reconstruction problem. We have explained the theory behind ABC and the algorithms developed to apply the method. Now, we try to develop significant reconstruction models for continuous and grid surface detectors. We have divided the experiment in three parts.

(i) Firstly, we use a rudimentary surface detector grid which bins any lepton that can arrive to the ground level. Then, reconstruct the EAS properties from the detector data with ABC algorithm. This first part of the experiment proves us that ABC models can be used to reconstruct CR and let us determine the ground accuracy metrics to compare with the more complicated models.

(ii) Secondly, we model the Pierre Auger (PA) Surface Detector (SD) Grid Array. Then, we run EAS simulations to generate signals from the modeled SD. In this manner we approach replicating the real world experiment conditions.

(iii) Lastly, we run the ABC algorithm over PA published real world public data to compare the performance of our reconstruction model with PA Collaboration's reconstruction accuracy.

2.1. Pierre Auger Public Reconstruction Data

The Pierre Auger observatory is a hybrid detector consists of ground water Cherenkov tanks and fluorescence detector. Cherenkov tanks are 3.6 meters in diameter and 1.2 meters in height. The tanks are filled with pure water. These 1660 surface detector tanks detect the high energy particles through the light emitted by their interaction with the pure water in these tanks. At the same time the fluorescence detector tracks the particles through the sky by ultraviolet emitted by the particles and record the longitudinal development of the EAS. Each EAS is recorded as an event, and each interaction of the secondary particles with the detector tanks are recorded as

EventId	**Stati.**	**Theta**	**Phi**	**EeV**	**Time**	**Lon**	**Lat**
620100	3	15.96	66.51	1.5952	1072936424	−117.86	9.74
620400	3	26.55	−101.79	0.3558	1072964627	−41.51	−2.21
620800	3	22.85	−30.76	0.6133	1073009810	−108.48	−24.87
621400	3	43.26	44.72	0.7503	1073079948	155.43	−60.59
622200	3	24.07	26.11	0.2830	1073168340	−158.13	−72.36
622800	3	22.23	−154.03	0.3332	1073236969	−10.81	−7.62
625800	3	34.33	−128.56	0.3524	1073583032	−25.95	−11.50

Table 2.1. PA Public Data Events Index File

TankId	**Signal**	**Time(sec)**	**Time(ns)**	**Easting**	**Northing**	**Altitude**
1359	8.17	1298214067	368674418	487375.69	6129513.23	1395.12
1348	4.84	1298214067	368671285	486624.45	6128212.97	1382.93
1369	3.22	1298214067	368672558	485872.04	6129517.65	1400.99

Table 2.2. PA Public Data Event Detail File

event details. We are using the PA Public dataset, built on Wed, 08 Jun 2011. The dataset includes an index of the events, and signal detail files for each event. The event index file consists of the event id, number of stations in the event, θ (zenith angle) of incoming reconstructed CR, ϕ (azimuth angle) of incoming CR, reconstructed Energy denoted in EeV, unix time of the event in seconds, galactic longitude and galactic latitude. Sample rows from the event index file can be viewed in Table 2.1.

The signal detail files consist of the list of tank id that has generated the signal, the signal generation time denoted in seconds and nanoseconds, easting and northing in meters and the altitude of the tank. A Sample event detail file can be viewed in Table 2.2.

The reconstructed energy of the events are in the range of $0.05 \times 10^{18} eV$ and $50 \times 10^{18} eV$ and the distribution of the energy can be seen in Figure 2.1. The reconstructed zenith angle of the events is in the range of 0-60° and the distribution of the zenith angle can be seen in Figure 2.2. The reconstructed azimuth angle of the events are between -180-180° and the distribution is almost uniform as can be sen in Figure 2.3.

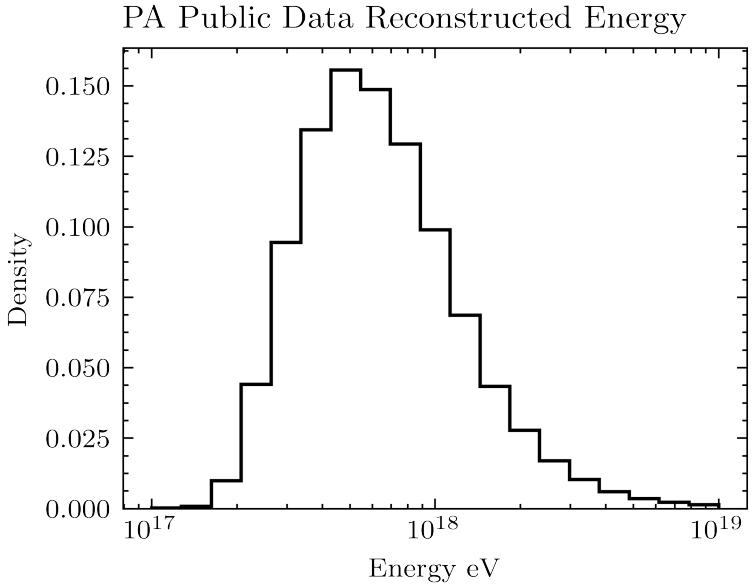

Figure 2.1. Pierre Auger Public Data Reconstructed Energy Histogram

2.2. Codes and Implementation

We generated the simulations with CORSIKA. To run the simulator we used the input format described in the user guide. An example can be seen in Appendix A. To read the CORSIKA output we used a Python library named SAPPHiRE which is authored by Javier Gonzalez and Arne de Laat. After reading the data we simulated the continuous surface detector and grid detector with a binning algorithm. We used the binned data to generate the summary statistics. Lastly we implemented the ABC algorithms with Python and utilized a grid search algorithm to optimize the models. Sample code of the aforementioned implementations can be seen in Appendix B.

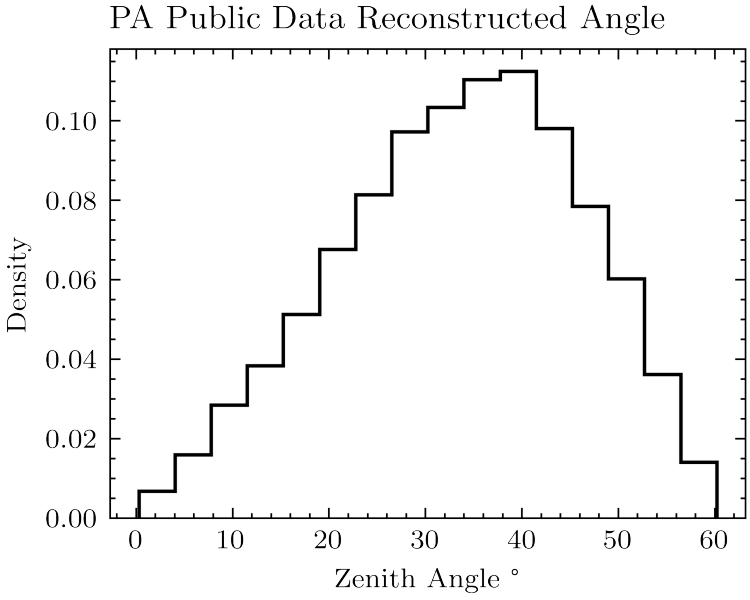

Figure 2.2. Pierre Auger Public Data Reconstructed Zenith Angle Histogram

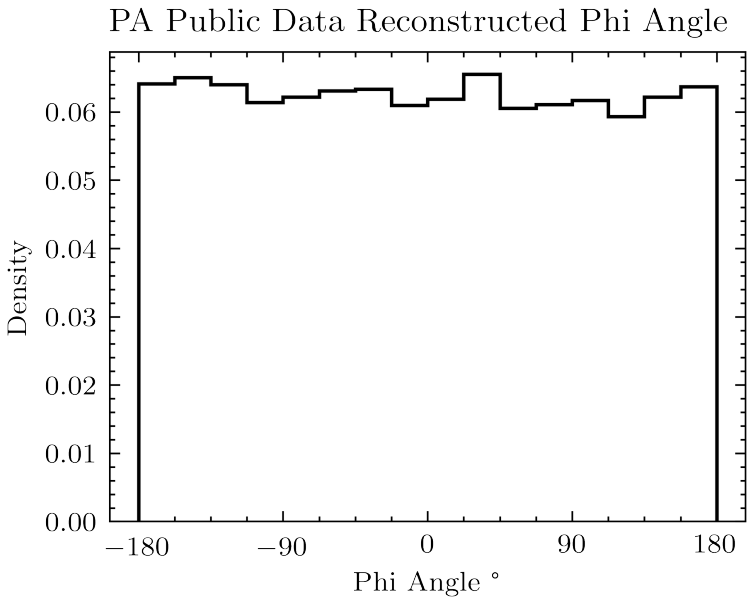

Figure 2.3. Pierre Auger Public Data Reconstructed Azimuth Angle Histogram

2.3. Continuous Surface Detector Array

In this first part of our experiment we define a hypothetical continuous detector grid which can perfectly bin any lepton arriving at the ground level (Figure 2.4). We have mentioned that the Auger grid is designed to detect very high energy EAS in the range of $10^{18}eV$. However, simulating the EeV range is very time extensive. It can take up to 40 hours just to simulate a single full event on a 3 GHz single core CPU. Besides, the secondary particle data, just with a single observation level, requires gigabytes of storage. To develop our model we need tens of thousands of simulations, thus for this reason we downscale the energy by factor of 10^4. This allows us to generate a single event in around 4 minutes and significantly reduce the storage constraint. In the second part of the experiment we will also downscale the surface detector grid in required proportion with the downscaled energy.

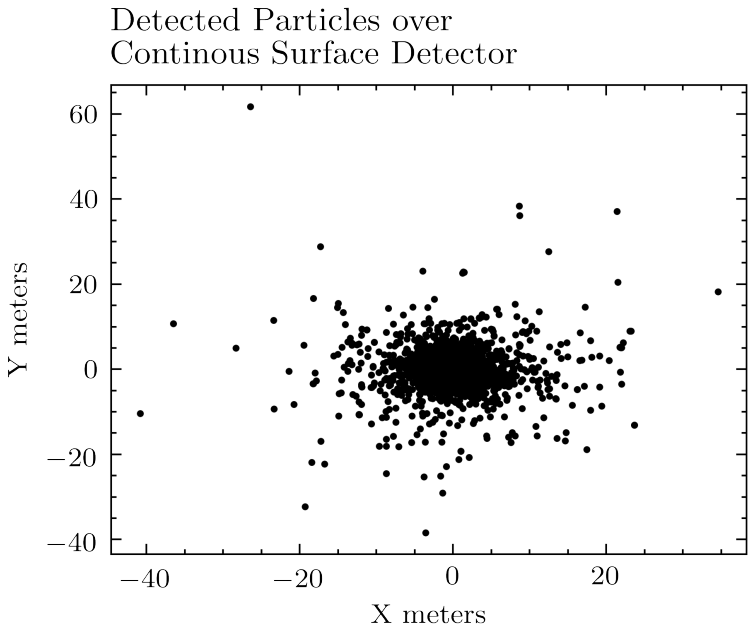

Figure 2.4. A Sample Simulated Event of $0.822 \times 10^{14}eV$ and 0° Zenith Detected by the Hypothetical Continuous Surface Detector

From the public data of the Pierre Auger observatory, we observe that the reconstructed EAS range is between $0.05 \times 10^{18}eV$ and $50 \times 10^{18}eV$ (Figure 2.1). The number of reconstructed EAS incidents peaks near $0.4 \times 10^{18}eV$ and then decrease

logarithmically. As very high energy events have very low frequency in the PA public data, we limit this density function to the range between $10^{17}eV$ and $10^{18}eV$ (Figure 2.5). Accordingly, we generate our energy downscaled simulations between $10^{14}eV$ and $10^{15}eV$ range.

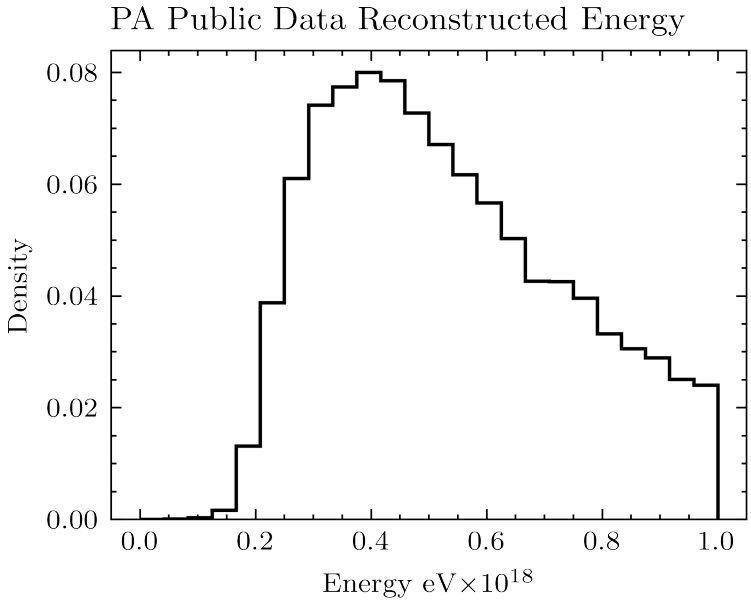

Figure 2.5. Pierre Auger Public Data Reconstructed Energy Histogram limited in the range of $10^{18}eV$

To generate EAS simulations we utilize CORSIKA with *QGSJET* interaction model. Primary particle is set to proton. At first we generate fixed zenith angle simulations where all EAS arrive perpendicular to the ground. The Earth's magnetic field effect is set to $19.52\mu T$ horizontally and to $14.17\mu T$ vertically to reflect the conditions at the PA Observatory. The observation level is set to $1452m$ which is the mean altitude of Water Cherenkov detectors. We use a uniform Energy prior (Figure 2.6) between $0.5 \times 10^{14}eV$ and $1 \times 10^{15}eV$. Next, we generate a dataset of fixed energy and variable Zenith angle. PA public dataset Zenith angle distribution is between $0 - 60°$ (Figure 2.2). We set the same range for variable angle and set fixed Energy level to $0.1 \times 10^{15}eV$.

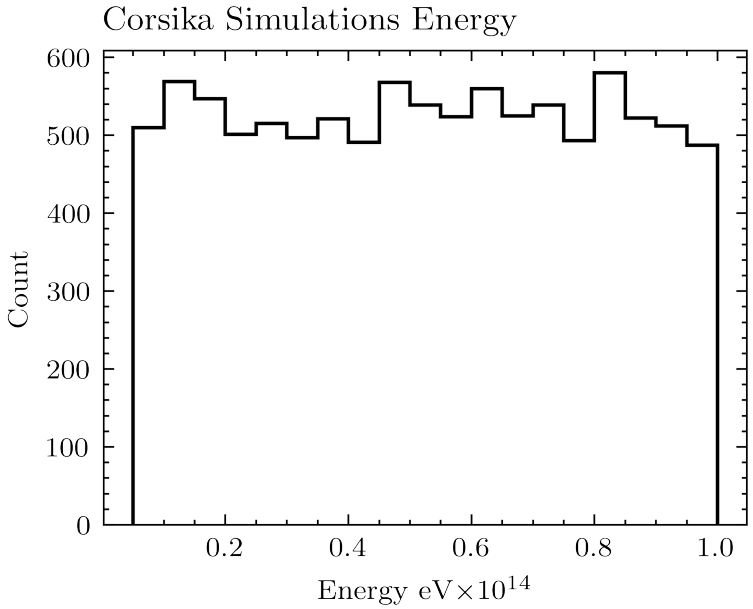

Figure 2.6. Energy Prior for Fixed Zenith Angle Corsika Simulations

2.3.1. Summary Statistics

We need informative particle data summaries to use with the algorithm. The summaries should contain the information of Energy and Zenith angle of the event. As we have discussed in the EAS Reconstruction section, the Energy and particle count have a linear relation and the ratio of particle count at different distances from the shower core is used to reconstruct the particle energy. Thus, we generate summary statistics of the total particle count and the distance between two farthest particles detected in the event to summarize the shower size. In addition, we generate summary statistics of particle count at different number of quantiles of X and Y distances at 0.625, 0.750 and 0.875 to summarize the density of surface detected particles. Quantile statistics are thought to be analogous summaries for $S(600)$ and $S(1000)$ metrics used in conventional reconstruction methodology. Histograms of these summary statistics can be seen in Figure 2.7.

Next, we introduce the candidate summary statistics to the ABC Rejection algorithm utilizing the *Approximate Sufficiency* approach. At first we minimize the

reconstruction error of number of particles N. The acceptance limit is set to $\varepsilon_N = 250$ (Figure 2.8).

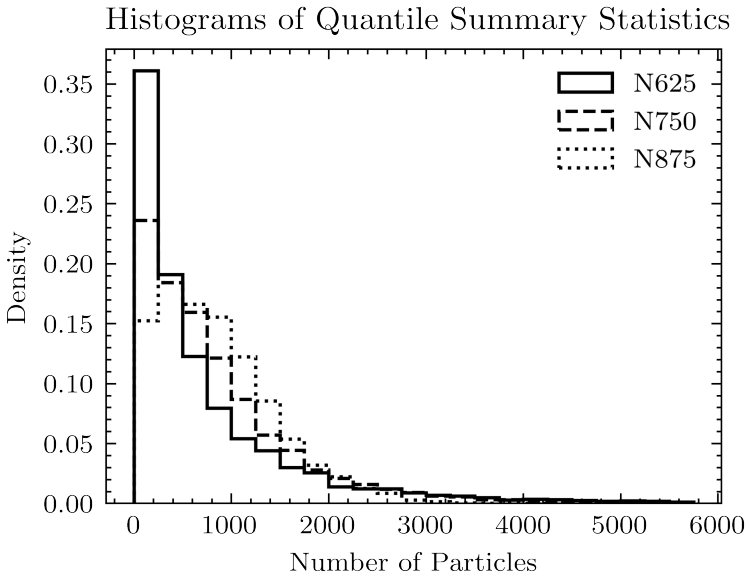

Figure 2.7. Histograms of the Quantile Summary Statistics

Then, we introduce the 0.625 quantile of N, namely the summary statistic $N625$. The minimum is achieved at $\varepsilon_{N625} = 50$ (Figure 2.9). The lower acceptance values prevent the formation of posterior, thus preclude the reconstruction of the observations.

At the optimization of next quantile statistic $N750$, a minimum can not be achieved (Figure 2.10). Stated in other words, this summary statistic does not contain any distinctive information. The minimum error achieved by the combination of N and $N625$ could not be enhanced.

Addition of $N875$ to the acceptance criteria increase the accuracy at $\varepsilon_{N875} = 25$ (Figure 2.11). However, this better accuracy comes with the cost of, observations that can not be reconstructed at all. Posteriors could not be generated for 11% of the events. Thus, we do not include this statistic in our acceptance criteria.

Lastly, we introduce the shower size as summary statistic. Like $N625$, shower size does not add any value to the model (Figure 2.11).

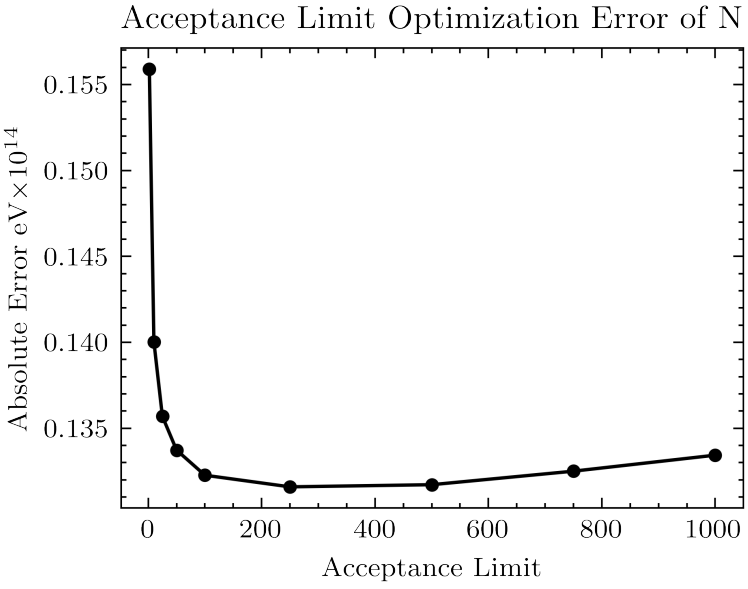

Figure 2.8. Mean Absolute Error of Reconstructed Energy by the Acceptance Limit of N for ABC Rejection Algorithm run over Continuous Surface Detector

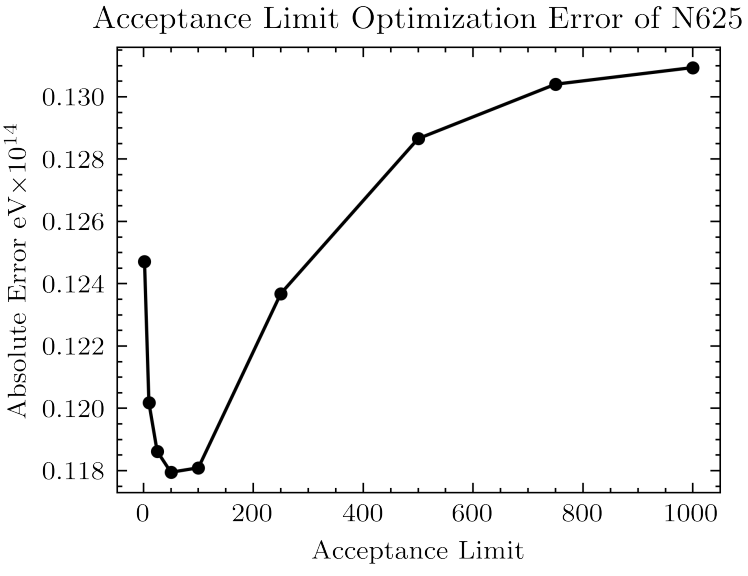

Figure 2.9. Mean Absolute Error of Reconstructed Energy by the Acceptance Limit of N625 for ABC Rejection Algorithm run over Continuous Surface Detector

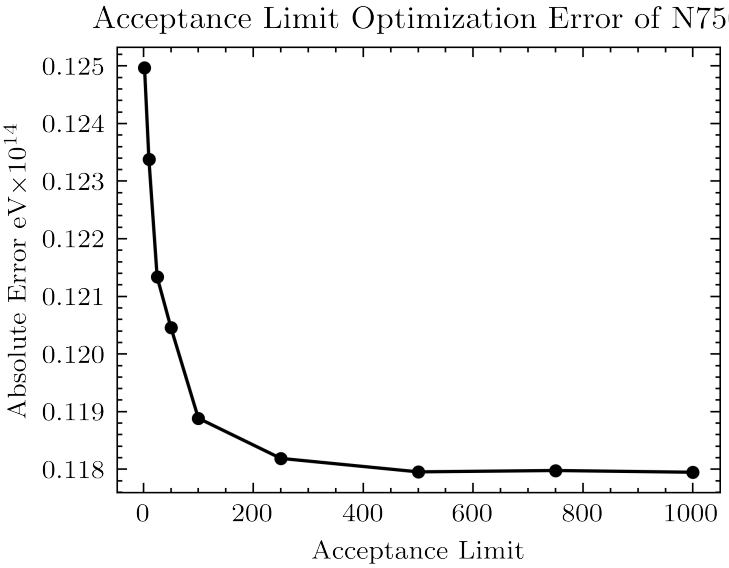

Figure 2.10. Mean Absolute Error of Reconstructed Energy by the Acceptance Limit of N750 for ABC Rejection Algorithm run over Continuous Surface Detector

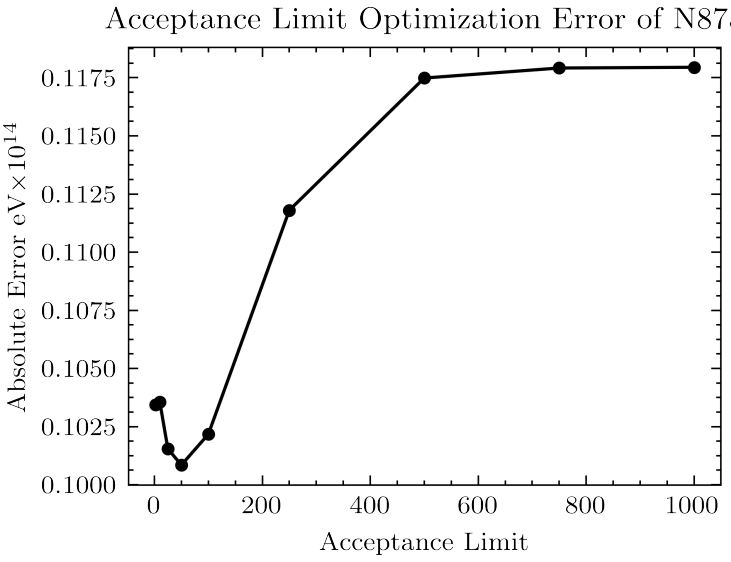

Figure 2.11. Mean Absolute Error of Reconstructed Energy by the Acceptance Limit of N875 for ABC Rejection Algorithm run over Continuous Surface Detector

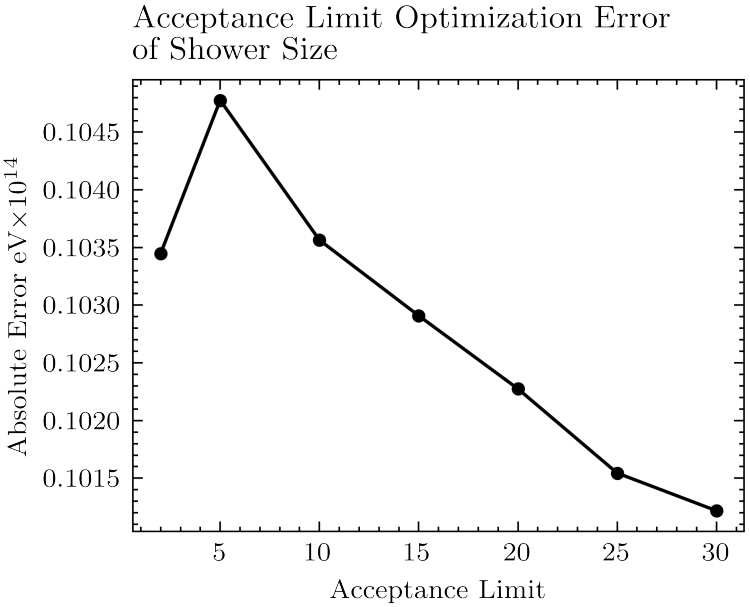

Figure 2.12. Mean Absolute Error of Reconstructed Energy by the Acceptance Limit of Shower Size for ABC Rejection Algorithm run over Continuous Surface Detector

2.3.2. ABC Rejection

With our parameter introduction approach, we have selected N and $N625$ from the candidate summary statistics. Reconstruction of the Energy is based on the acceptance of the sample both by ε_N and ε_{N625}. By the criteria we run the ABC Rejection algorithm and reconstruct the observations.

Distribution of the approximation error of Energy can be seen in Figure 2.13. We have also binned the accuracy of the ABC Rejection algorithm by incident Energy in Figure 2.14. We observe that the observations with near mean Energy levels can be estimated with higher accuracy, while the accuracy decrease with the increasing distance to the mean Energy level. In addition, the increasing error deviation stands out with increasing Energy. The variability of the signal generation pattern increases with increasing energy and we perform these experiments in a limited range of energy. If simulations were to span the full range of energy, higher energy events would also be selected for the posterior and the under prediction bias would disappear.

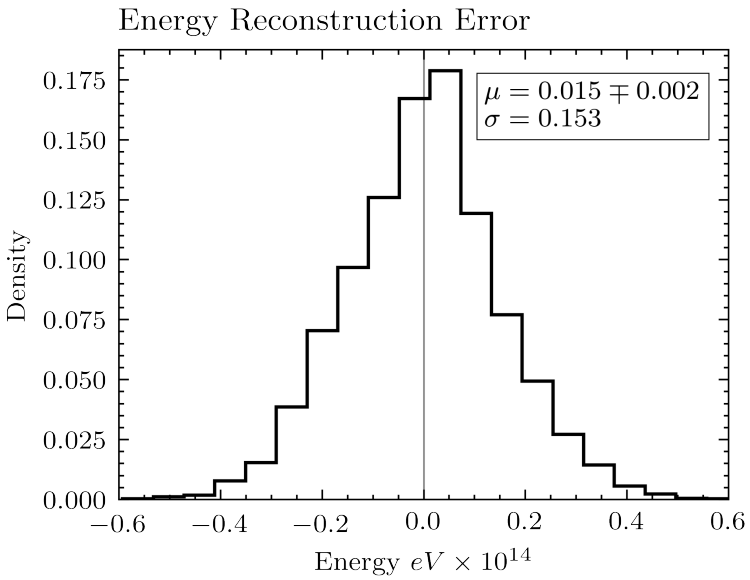

Figure 2.13. Error Histogram of Energy Reconstruction for ABC Rejection Algorithm run over Continuous Surface Detector

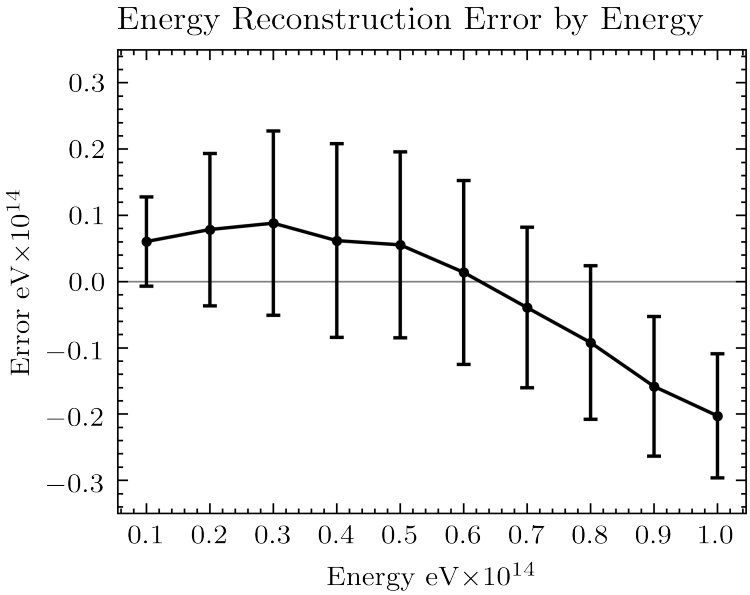

Figure 2.14. Error of Energy Reconstruction by Reconstructed Energy Range for ABC Rejection Algorithm run over Continuous Surface Detector

To assess the significance of our model, we use a random reconstruction algorithm, where we draw uniformly random samples from the prior for every posterior generated by the ABC Rejection algorithm. Sample size is defined as the mean posterior sample count. The comparison of error distribution histograms of the ABC Rejection and Random Model can be seen at Figure 2.15. Our reconstruction model is significant compared to random reconstruction.

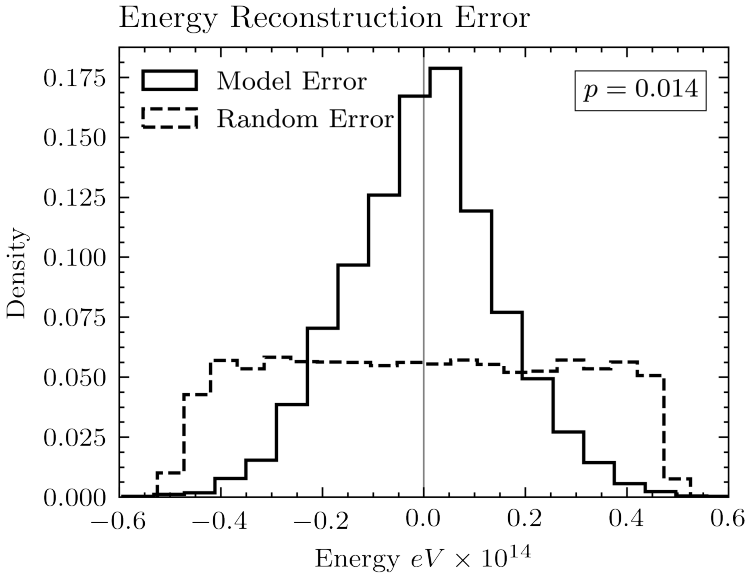

Figure 2.15. Energy Reconstruction Error Histogram Comparisons of ABC Rejection Algorithm and Random Model run over Continuous Surface Detector

Next, we try to select summary statistics to reconstruct the Zenith angle of the EAS. We make educated guesses of candidates. The time difference in generation of signals at different tanks in the same event is used to reconstruct the Zenith angle. Thus, we generate the moments of the signal generation times as candidate summary statistics.

First, we introduce the mean time of signal generation statistic *tmean* to the ABC Rejection algorithm. Mean absolute error of Zenith angle reconstruction immediately drops to 4.1° at $\varepsilon_{tmean} = 500$. However, below 500 the selection criteria become too strict that posteriors of the events can not be constructed (Figure 2.16).

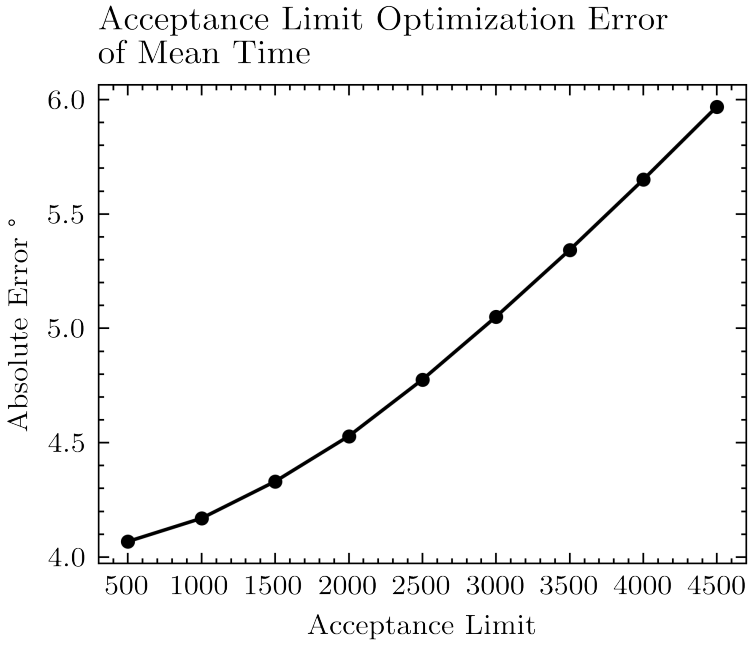

Figure 2.16. Mean Absolute Error of Reconstructed Zenith Angle by the Acceptance Limit of *tmean* for ABC Rejection Algorithm run over Continuous Surface Detector

Secondly, we introduce the standard deviation of signal generation time as *tstd*. We achieve the best gain at $\varepsilon_{tstd} = 400$ (Figure 2.17). Next, we try the maximum signal time *tmax*. Accuracy increase at $\varepsilon_{tmax} = 4000$ (Figure 2.18). We also introduce the skewness of the signal generation time, standard deviation and skewness of locations of the signals but did not gain any increase in the accuracy. In addition, strict selection criteria left too many events unreconstructed.

Finally, $\varepsilon_{tmean} = 500$, $\varepsilon_{tstd} = 400$ and $\varepsilon_{tmax} = 4000$ are used to run the ABC Rejection Algorithm for Zenith reconstruction. The histogram for net Zenith angle reconstruction error can be seen in Figure 2.19. Accuracy and deviation of the error vary by different incident angles (Figure 2.20). The small angle range below 10° is over predicted because of a systematic bias caused by a simulation parameter called RCUT. We use a radius cut parameter for simulations to minimize the need of data storage by discarding the particles at the very center of the shower. Secondary particles concentrate near the shower center especially at zenith angles below 10°.

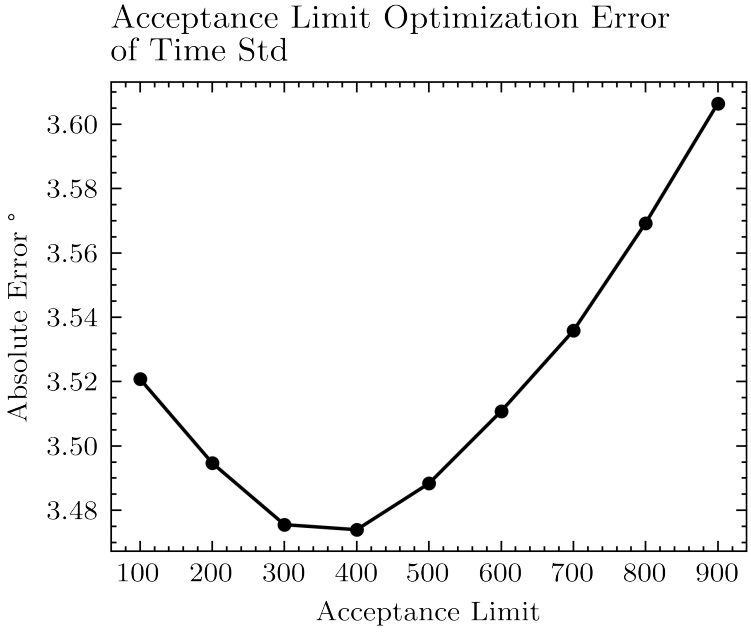

Figure 2.17. Mean Absolute Error of Reconstructed Zenith Angle by the Acceptance Limit of *tstd* for ABC Rejection Algorithm run over Continuous Surface Detector

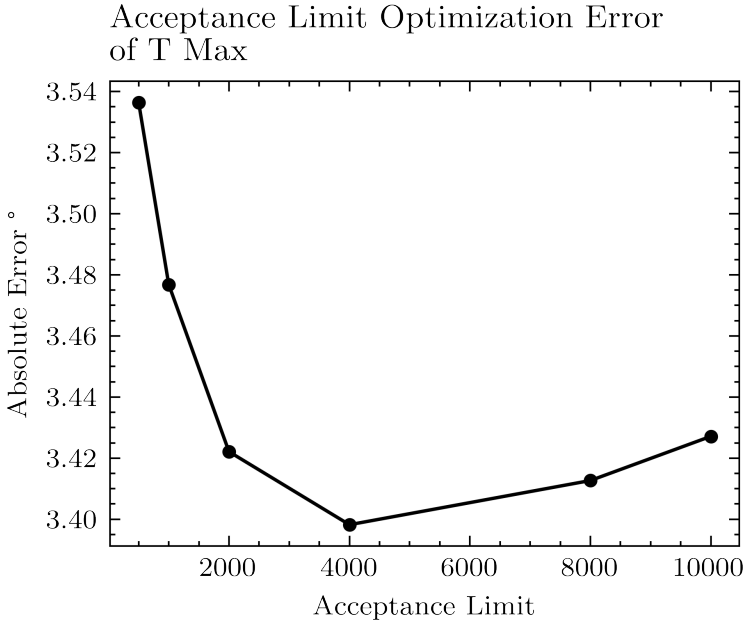

Figure 2.18. Mean Absolute Error of Reconstructed Zenith Angle by the Acceptance Limit of *tmax* for ABC Rejection Algorithm run over Continuous Surface Detector

We reconstruct the same observations with a random model to assess the significance and gain of our model. Our model is significant and error comparisons between ABC and random models can be seen in Figure 2.21.

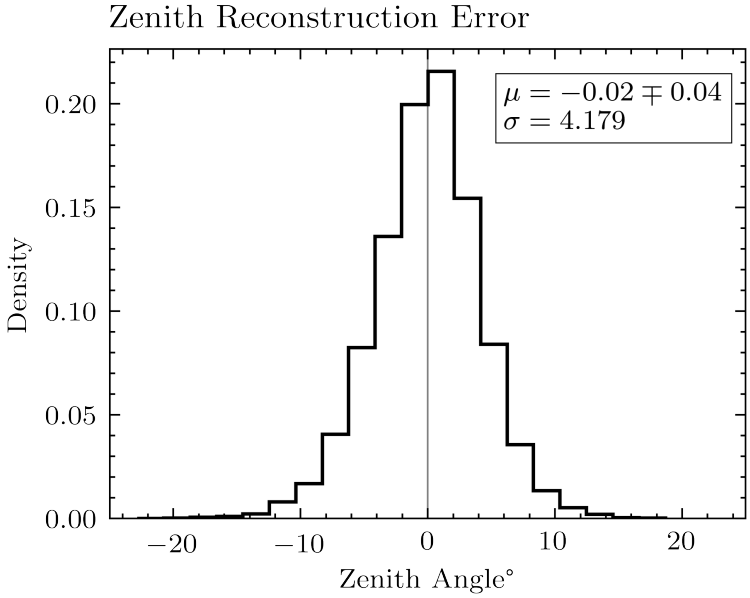

Figure 2.19. Error Histogram of Zenith Angle Reconstruction for ABC Rejection Algorithm run over Continuous Surface Detector

2.3.3. Sequential ABC

In the ABC Rejection approach we have used a set of data to optimize the acceptance criteria of the summary statistics. However, the train set may not contain all the possible range of observations. This may cause the model to underperform in unexplored input ranges. In addition, the optimized parameters are optimized to perform globally, while different ranges of priors may perform better with different acceptance levels. So, we design a Markov Chain Monte Carlo procedure, where we change the simulation prior at every iteration, based on the distance of posterior to the observation of the previous iteration. We increase the weight of parameters in the prior which generate a closer posterior to the observation. Distance is based on summary statistics separately, or a function of the summary statistics like Euclidean distance. We assume that with enough iteration the posterior will converge to the true parameters that have caused the observation.

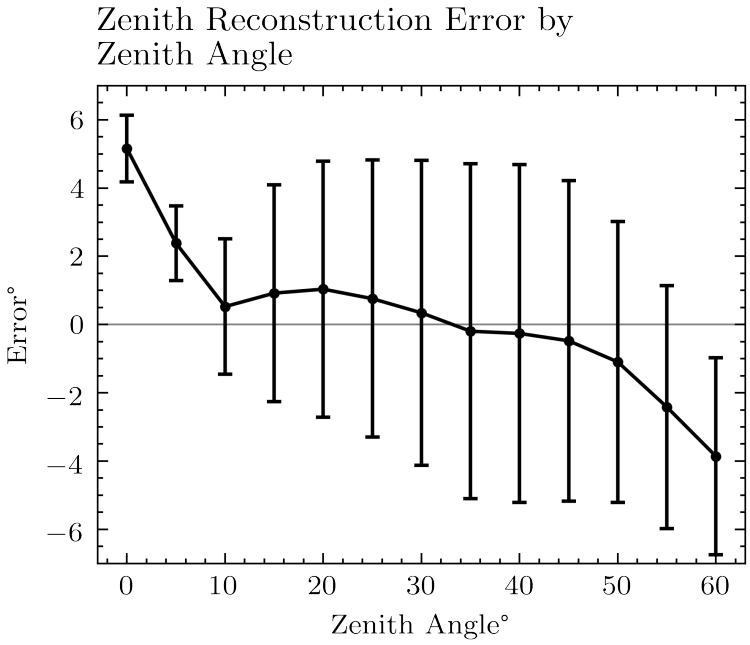

Figure 2.20. Error of Zenith Angle Reconstruction by Reconstructed Incident Angle Range for ABC Rejection Algorithm run over Continuous Surface Detector

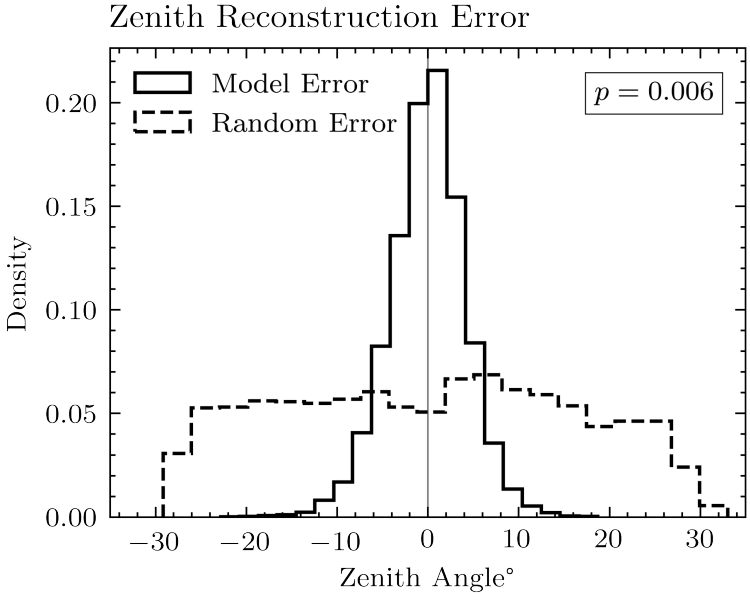

Figure 2.21. Zenith Angle Reconstruction Error Histogram Comparisons of ABC Rejection Algorithm and Random Model run over Continuous Surface Detector

We use the same N and $N625$ summary statistics to run the Sequential ABC algorithm. Euclidean distance between the posteriors is set as the distance metric. We start by sampling from the prior randomly, and at each iteration we weight the prior by the inverse of distance.

The algorithm yields a similar but slightly worse performance than the ABC Rejection algorithm (Figure 2.22). The model is still significant (Figure 2.23). The key feature of the sequential algorithm is to generate simulations with a weighted prior, updated at each iteration. This way the posterior of the simulations converge faster and more accurately to the parameters of the observation. Thus, the sequential algorithm can show its superiority where it can generate enough number of simulations for the reconstruction of an observation. However, we can not generate the required amount of simulations because of the time and storage limitations at $10^{15}eV$.

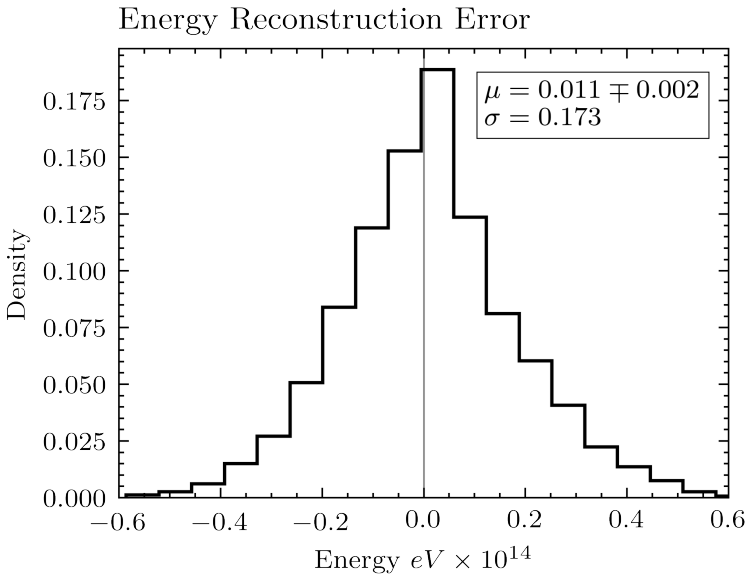

Figure 2.22. Error Histogram of Energy Reconstruction for Sequential ABC Rejection Algorithm run over Continuous Surface Detector

So for the sake of argument, we compare the rejection algorithm with the sequential algorithm at a lower energy level, by generating simulations of the CR events in the range of $0.5 \times 10^{12}eV$ and $1 \times 10^{13}eV$. First, we optimize the rejection model and set $\varepsilon_N = 20$ and $\varepsilon_{N625} = 5$ as the best distance parameters. Then, we get the same set

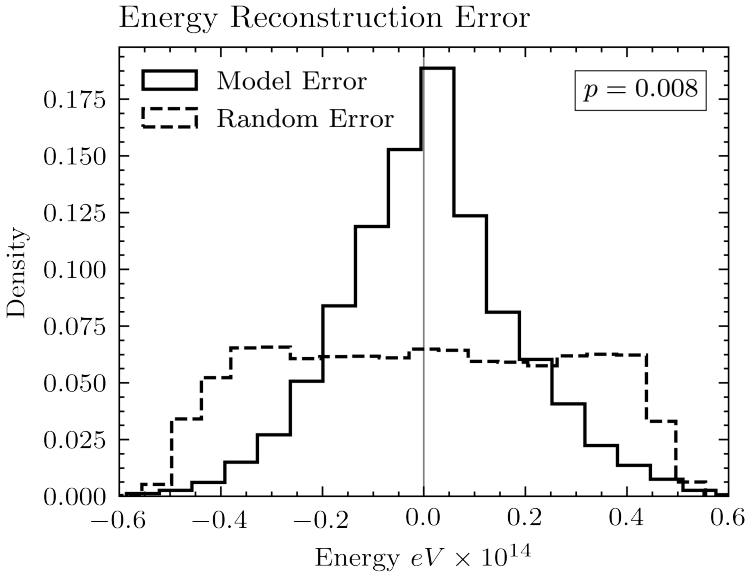

Figure 2.23. Energy Reconstruction Error Histogram Comparisons of Sequential ABC Algorithm and Random Model run over Continuous Surface Detector

of simulated observations and reconstruct them with the sequential algorithm.

Sequential algorithm starts running by drawing uniformly random samples from the simulation set and compare the distance between the simulations and the simulated observation. The distance is set as the Euclidean of N and $N625$ summaries. Prior of the next iteration is weighted by inverse of the distance. The samples from the first iteration is discarded and is not added to the posterior. The second iteration draws fresh simulations with the input parameters sampled from the weighted prior. The new samples are added to the posterior and the prior of the next iteration is reweighed. This loop continues until convergence. Reconstruction error decreases with each iteration (Figure 2.25).

We show that sequential approach performs better than the rejection methodology (Figure 2.24) to reconstruct the Energy. We see that the reconstruction bias is corrected and error deviation is decreased compared to reconstruction performed by rejection algorithm.

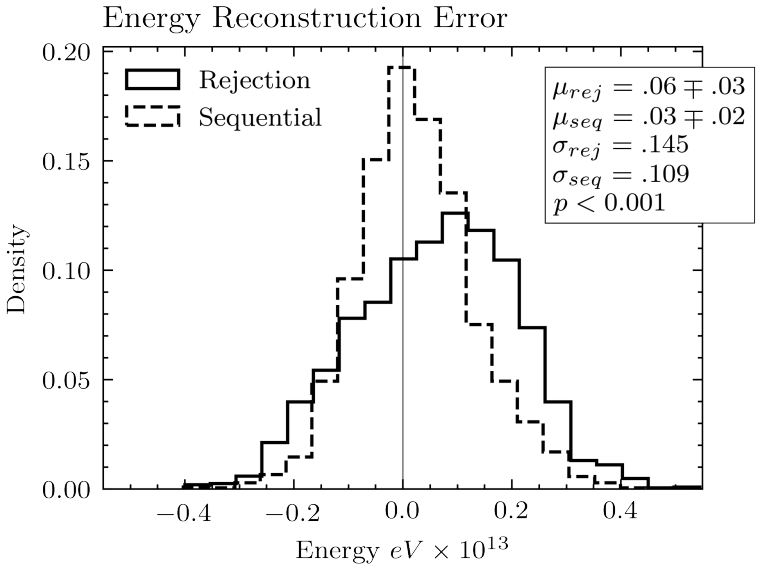

Figure 2.24. Energy Reconstruction Error Comparisons of Sequential ABC Algorithm and ABC Rejection Model run over Continuous Surface Detector for Low Energy Events

2.4. Surface Detector Grid Simulation

In the second part of the experiment, we model the Pierre Auger Surface Detector grid from the public data. We aggregate the signals generated in tanks and average their coordinates to pinpoint the locations of the Water Cherenkov Surface Detectors. From the PA SD design paper [26] we know that, SDs are located $1500m$ apart from each other and they each have a diameter of $3.6m$. Our modeling of the SD locations (Figure 2.26) from the PA public data agrees with the design.

As we have downscaled the Energy level in the first part of our experiment, this time we miniaturize the PA SD grid, to make it compatible with the Energy range we work with. We down scale the grid by 10^3. The distance between the tanks is reduced to $2.59m$. The effective area of the tanks are also reduced in proportion. Tank diameter is decreased to $0.11m$.

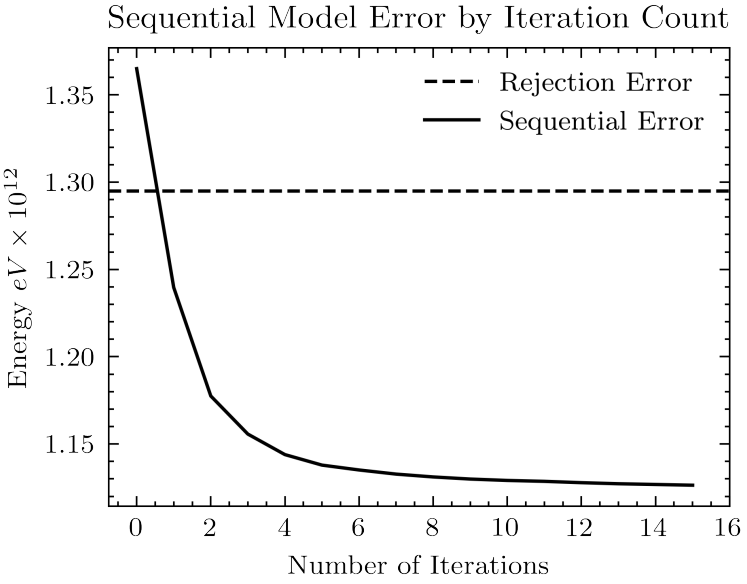

Figure 2.25. Error of Energy Reconstruction by Number of Iteration for Sequential ABC Algorithm run over Continuous Surface Detector for Low Energy Events

We use the miniaturized version of the grid to generate the signal data from particles detected by the SDs. We take PA public data format as in Figure 2.1 and Figure 2.2 as an example to report the signals. Generated signals of PA Modeled Grid Array are sparse matrices of, generated signals of continuous SD grid in the first part of the experiment.

We adapt the N and $N625$ summary statistics to work with the reconstruction of the Energy from signals generated at tanks. We aggregate signals by tank, and determine the tank location as the signal coordinate. Optimization of ε_N yields 6 as the best acceptance level (Figure 2.27).

Next, we introduce $N625$ and gain better overall accuracy at $\varepsilon_{N625} = 4$ (Figure 2.28). Even with the second summary statistic, we are unable to reconstruct a few observations, thus we do not introduce any new acceptance criteria to the model.

We compare the model, to reconstruct the energy with ABC Rejection Algorithm over surface detector grid, with the corresponding random model. T-test between errors

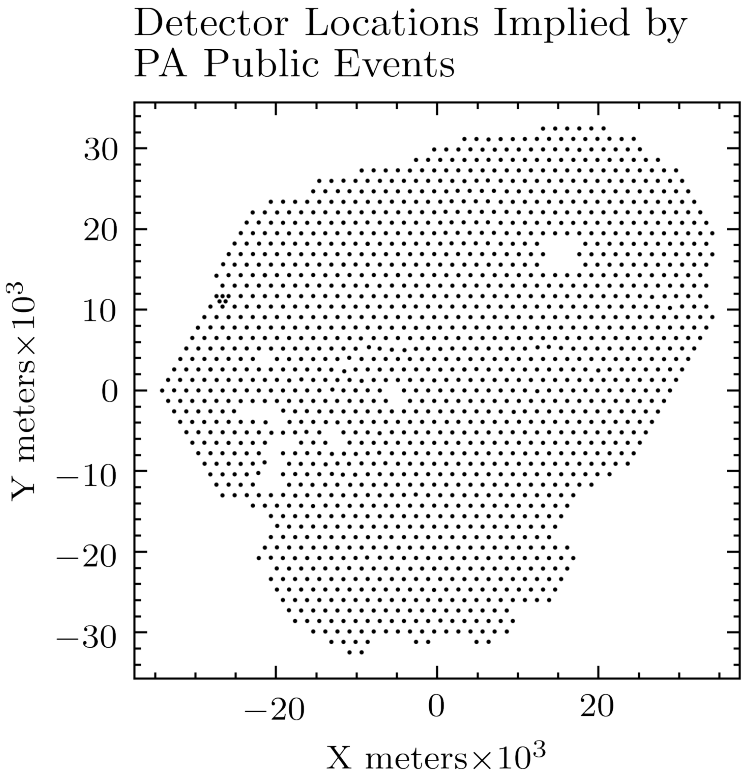

Figure 2.26. The Physical Layout of the Pierre Auger Surface Detector Grid Inferred from Public Data

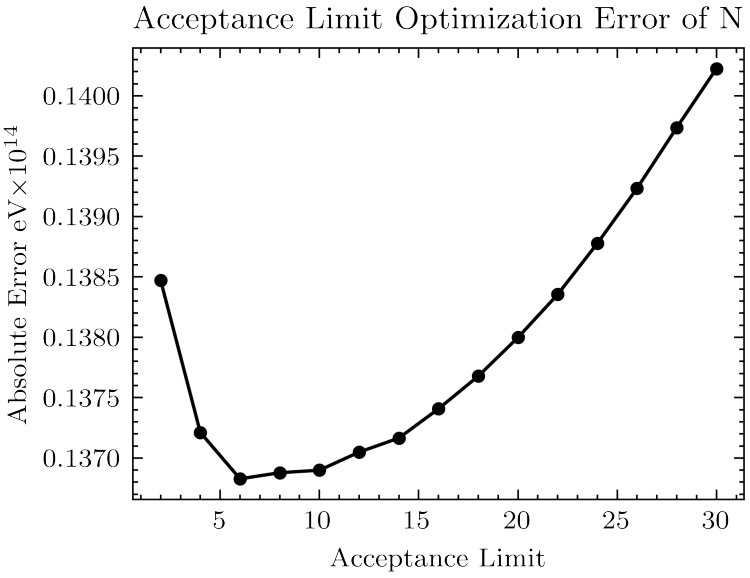

Figure 2.27. Mean Absolute Error of Reconstructed Energy by the Acceptance Limit of N for ABC Rejection Algorithm run over Surface Detector Grid

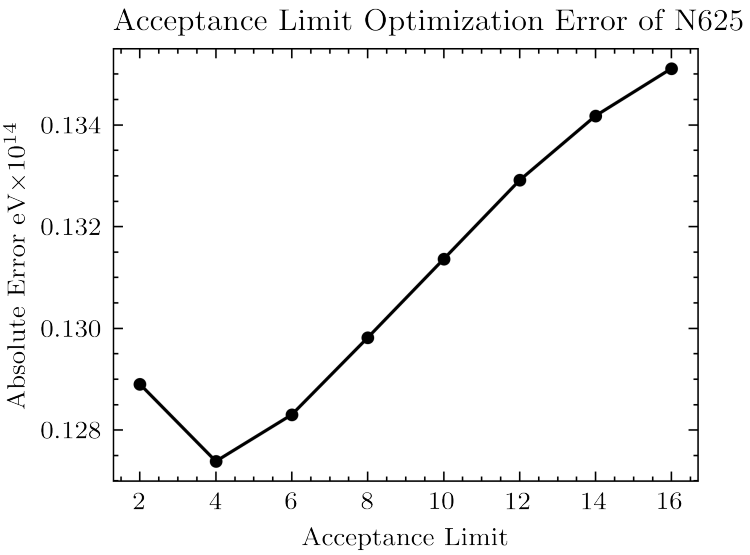

Figure 2.28. Mean Absolute Error of Reconstructed Energy by the Acceptance Limit of N625 for ABC Rejection Algorithm run over Surface Detector Grid

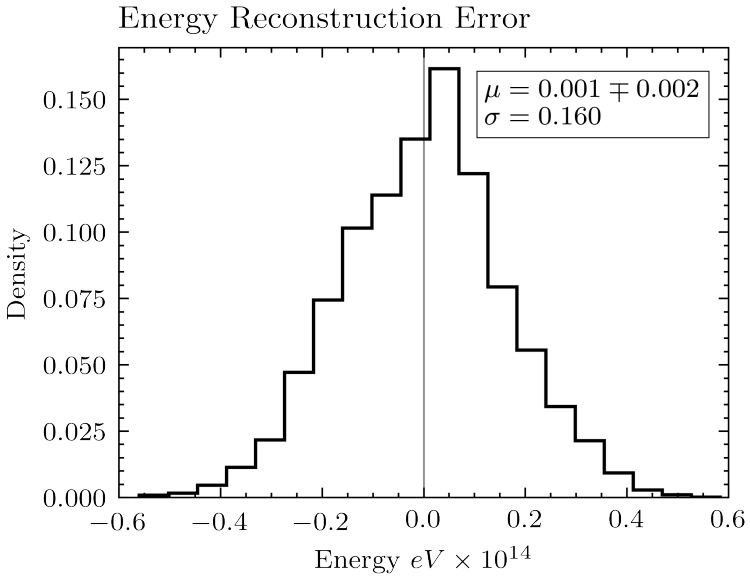

Figure 2.29. Error Histogram of Energy Reconstruction for ABC Rejection Algorithm run over Surface Detector Grid

of the aforementioned model and random model does not yield significant results. Reconstructed posterior mean sample size is increased almost 3 fold, to 548 from 187 samples. With 10^4 samples in the train set, the model generates posteriors sampling just the 5.48% of all the simulations. This means that acceptance criteria of the model is not strict enough to differentiate the energy level of the samples. In other words, binning the signals and discarding information because of the nature of a grid detector decreased the signal to noise ratio substantially. Thus, we introduce the previously discarded summary statistics of $N750$ and $N875$ to the model. With more strict acceptation criteria, mean sample size of the reconstructed posteriors decrease to 101 and model is unable to reconstruct 7.65% of the samples. In return we get a significant reconstruction model (Figure 2.32).

We analyze the unreconstructed events and realize that, the count of the unreconstructed events increase with increasing Energy (Figure 2.33). The percentage of the unreconstructed events in each energy range can be seen in Figure 2.34. EAS with more energy increase the variability of the particle distribution at the observation level. Thus, more simulations are needed at higher energy to populate close enough samples to pass the acceptance criteria.

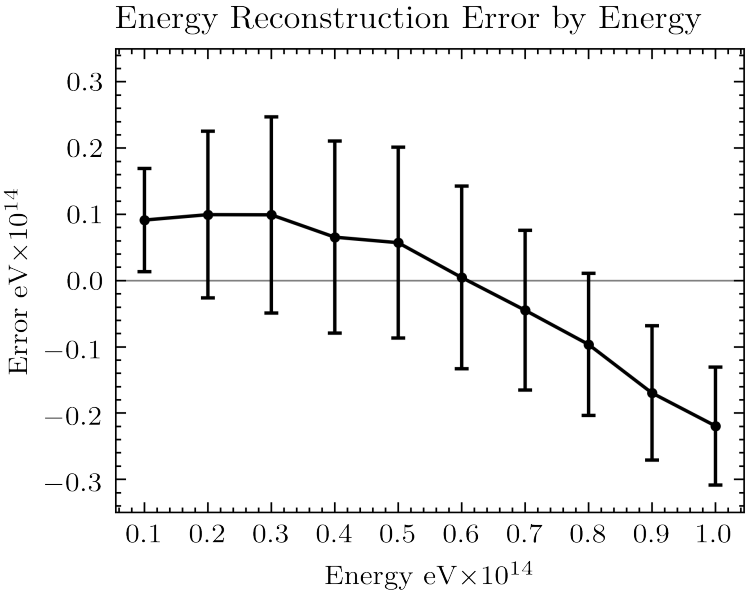

Figure 2.30. Error of Energy Reconstruction by Reconstructed Energy Range for ABC Rejection Algorithm run over Surface Detector Grid

We advance to reconstruction of the Zenith over the grid SD. We utilize the same summary statistics identified by the continuous SD model. These statistics are *tmean*, *tstd* and *tmax*. Optimization of the summary statistics yield the minimum reconstruction error at $\varepsilon_{tmean} = 32$ (Figure 2.35), $\varepsilon_{tstd} = 64$ (Figure 2.36) and $\varepsilon_{tmax} = 128$ (Figure 2.37).

The reconstruction error is increased as expected (Figure 2.38). The model is significant compared to corresponding random model (Figure 2.40). There is almost three fold increase in the mean absolute reconstruction error from 3.21° to 8.47°. This is a huge increase, compared to Energy reconstruction error increase of 1.71% between continuous and grid detectors. Surface Grid Detector can generate signal only if the shower particle falls over the detector tank and the signals are summed over to create an aggregated signal strength. On the other hand, the Continuous Surface Detector generate separate signal for each particle arriving to the observation level. The aggregation of the signal strength have a lesser effect compared to arrival time, as it just adds a location uncertainty of half the diameter of the detector tank. However, we can

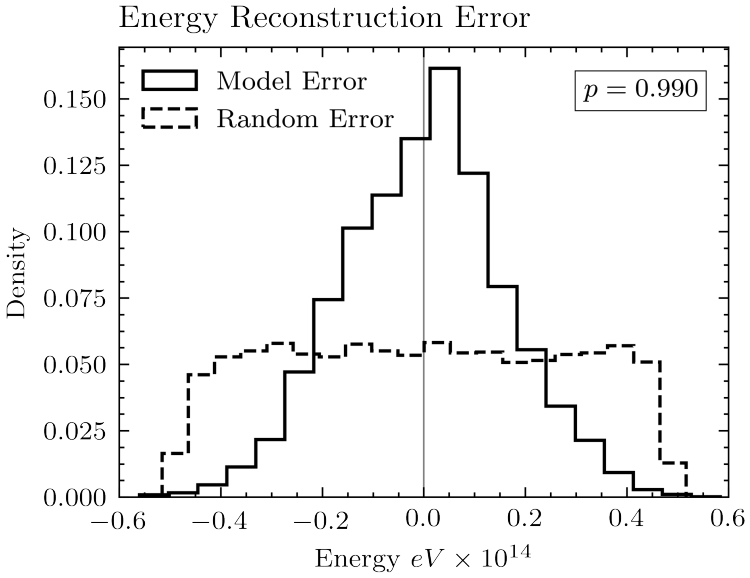

Figure 2.31. Energy Reconstruction Error Histogram Comparisons of ABC Rejection Algorithm and Random Model run over Surface Detector Grid

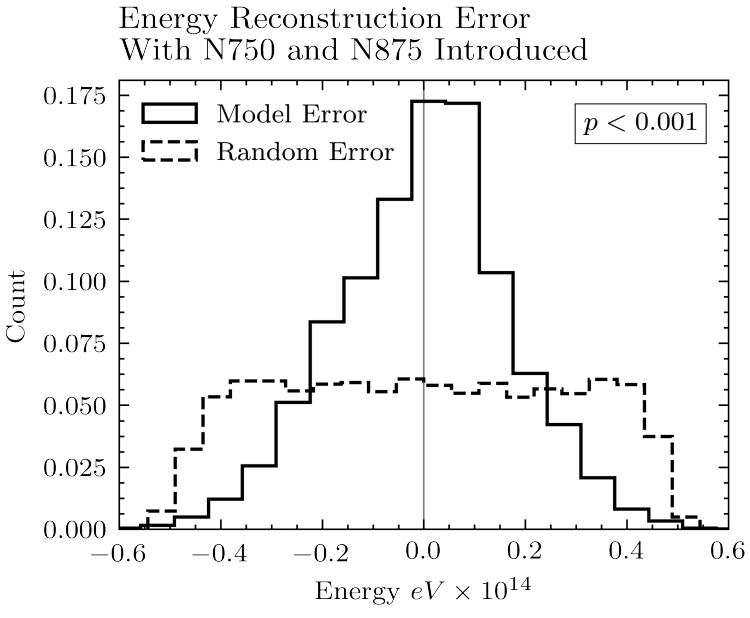

Figure 2.32. Energy Reconstruction Error Histogram Comparisons of ABC Rejection Algorithm With More Strict Criteria and Random Model run over Surface Detector Grid

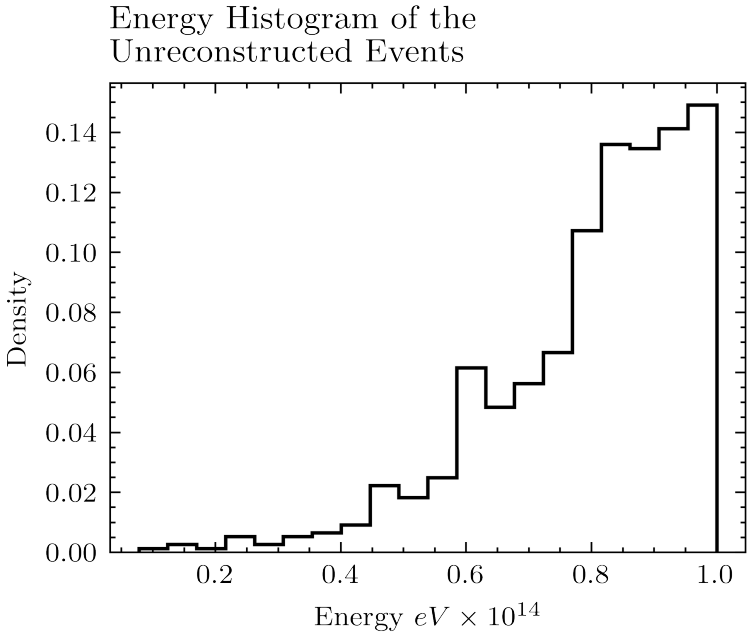

Figure 2.33. Energy Histogram of the Unreconstructed Events of ABC Rejection Algorithm With More Strict Criteria Run Over Surface Detector Grid

just get the arrival time of the first arriving particle and lose the rest of the information in our model. This loss of information causes worse Zenith reconstruction accuracy. This can also be seen in Figure 2.39 where smaller zenith angles are over predicted and larger zenith angles are under predicted.

2.4.1. Randomized Detector Locations

PA have an orderly and an almost symmetrical surface detector grid with equally distanced water Cherenkov tanks. However, the cost to build detector grid over a flat and large land is humongous. Low budget experiments may be designed to utilize rooftop detectors, detectors spread over a larger area or maybe even with detectors with variable locations. We wondered if we could develop reconstruction models for randomly distributed detectors thus, randomly shifted the locations of the tanks in the PA Grid. The random shift is based on independent samples of x and y distances sampled from a zero mean normal distribution with the standard deviation of located detectors in the grid. Resulting random grid can be observed in Figure 2.41.

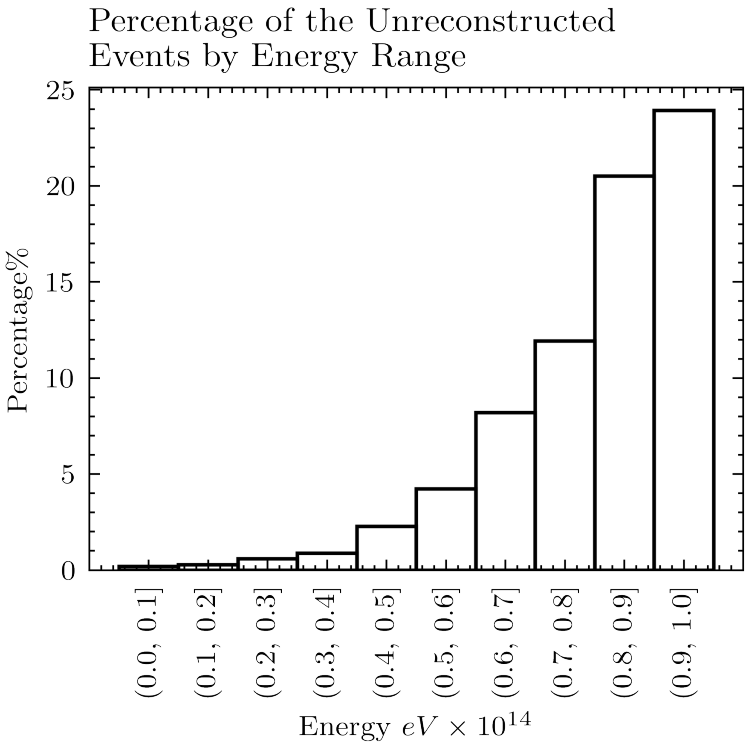

Figure 2.34. Percentage of the Unreconstructed Events of ABC Rejection Algorithm With More Strict Criteria Run Over Surface Detector Grid

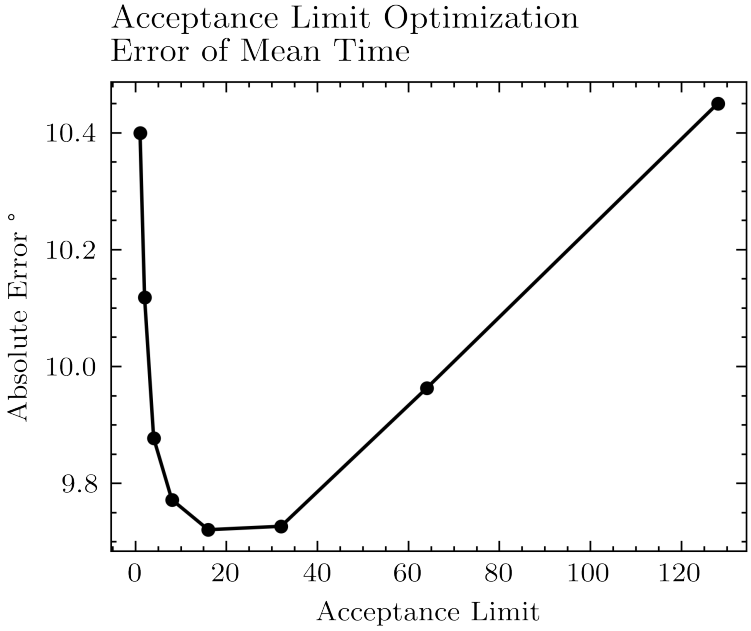

Figure 2.35. Mean Absolute Error of Reconstructed Zenith Angle by the Acceptance Limit of *tmean* for ABC Rejection Algorithm Run Over Surface Detector Grid

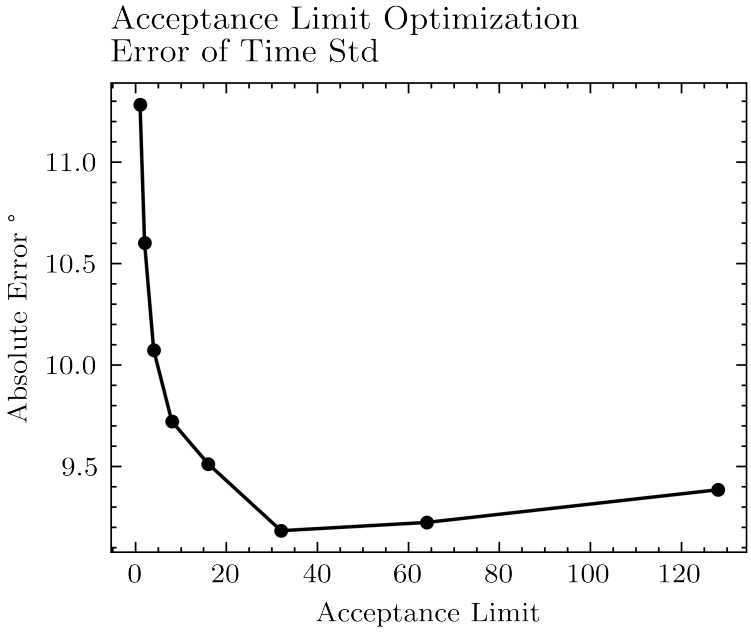

Figure 2.36. Mean Absolute Error of Reconstructed Zenith Angle by the Acceptance Limit of *tstd* for ABC Rejection Algorithm Run Over Surface Detector Grid

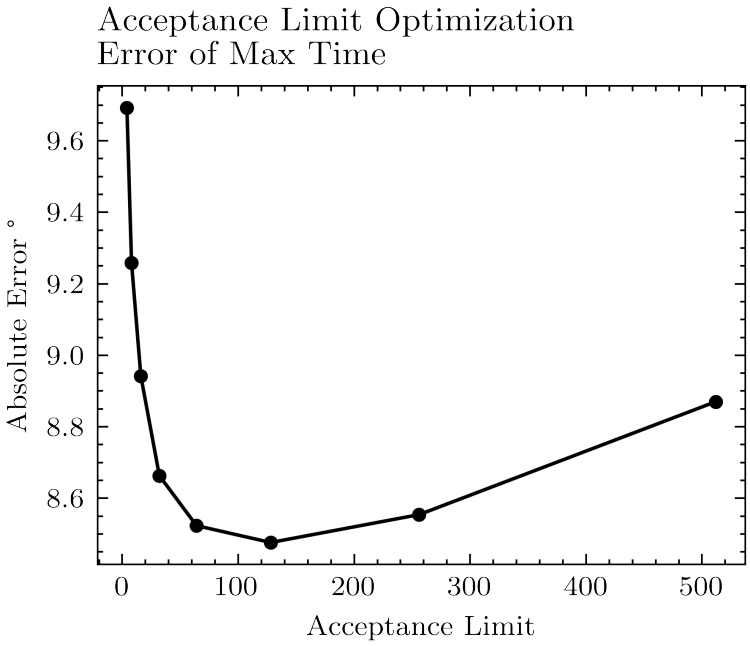

Figure 2.37. Mean Absolute Error of Reconstructed Zenith Angle by the Acceptance Limit of *tmax* for ABC Rejection Algorithm Run Over Surface Detector Grid

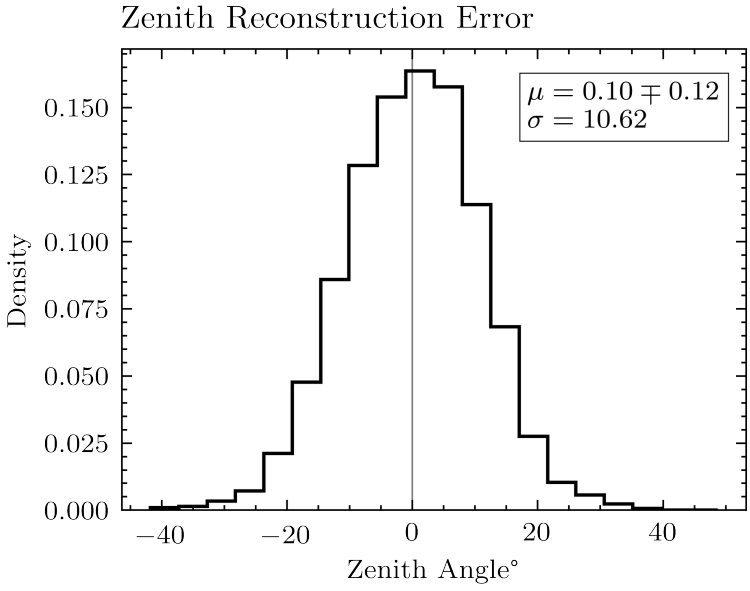

Figure 2.38. Error Histogram of Zenith Angle Reconstruction for ABC Rejection Algorithm run over Surface Detector Grid

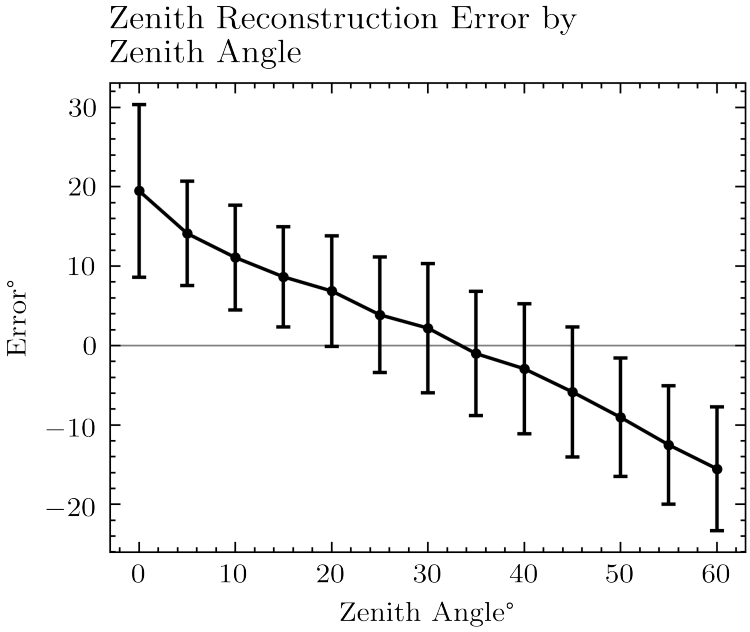

Figure 2.39. Error of Zenith Angle Reconstruction by Reconstructed Incident Angle Range for ABC Rejection Algorithm run over Surface Detector Grid

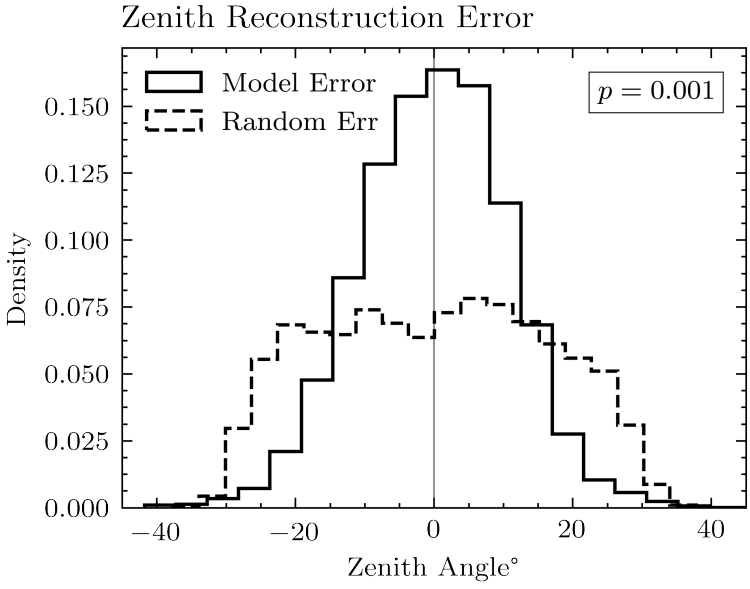

Figure 2.40. Zenith Angle Reconstruction Error Histogram Comparisons of ABC Rejection Algorithm and Random Model run over Surface Detector Grid

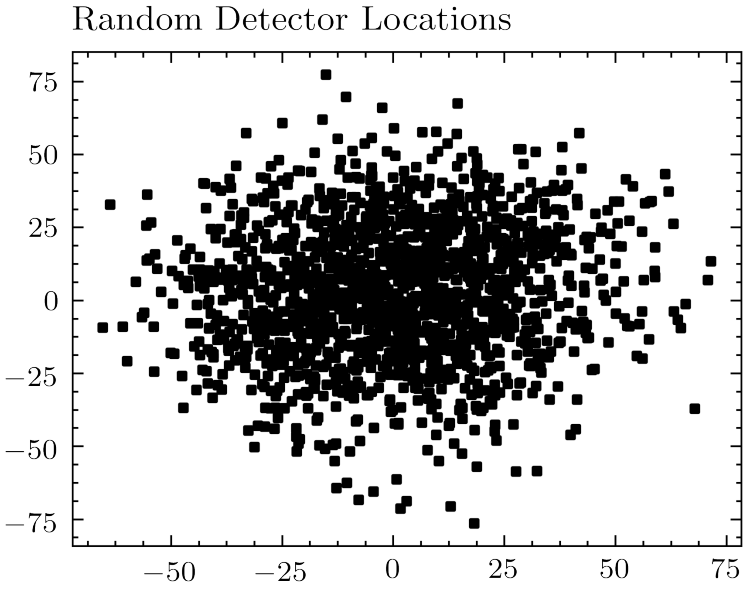

Figure 2.41. Surface Detector Grid with Randomly Shifted Tank Locations

We have re-optimized the parameters of N and $N625$ and attained the same best values with surface detector grid ABC rejection energy reconstruction model. We have adapted the energy reconstruction model to run over the random detector grid setup. The model turned out to be significant as can be seen in Figure 2.42. Ave et al. suggests that the tanks in a grid detector setup roughly measures the same point because of the proportionality between the footprint of an EAS and span of the detector grid [15]. On the other hand, the experimental results in EAS studies are based on large statistical samples rather than the properties of a single EAS event. It seems that significant statistical comparisons can also be made over detector grids with randomly distributed tanks.

2.5. ABC over Pierre Auger Public Data

In the third and last part of our experiment we reconstruct the Pierre Auger recorded public events. We reconstruct each event with the other events in the data set with ABC rejection algorithm. First, we start by optimizing the model for reconstruction of Energy. We have generated summaries of N, $N625$, $N750$ and $N875$.

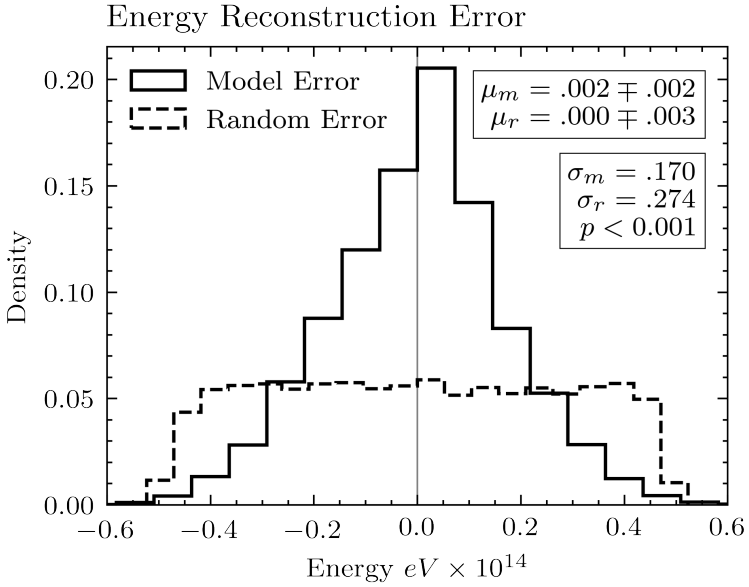

Figure 2.42. Error Histogram of Energy Reconstruction for ABC Rejection Algorithm run over Surface Detector Grid with Random Tank Locations

Different then the previous experiments, the Zenith is not perpendicular in Auger data set. Thus, we derive the equivalent vertical particle count of a shower arriving with an angle with the following equation.

$$N_v = N \exp\left(A_0 \left(\sec\theta - \sec\theta_0\right) \lambda\right) \tag{2.1}$$

In Equation 2.1, A_0 is atmospheric depth and λ is attenuation length. θ is the arriving zenith angle and θ_0 is the conversion angle. $\theta_0 = 0$ in our case as we want to calculate equivalent shower size of a vertical incidence and set $A_0\lambda = 1$. This method of *constant intensity cut* has been devised by found by MIT group [4]. We generate N_v accordingly and use it as an alternative summary statistic (Figure 2.43).

In the optimization procedure we can not determine a minimum for N (Figure 2.44) and N_v. Accuracy increases with stricter selection criteria, whereas unrecon-

structed events start to emerge. In this case, the minimum distances is set to $N = 50$ and $N_v = 50$ to reconstruct all the events with best reconstruction accuracy. Energy reconstruction accuracy is 13.7% without any bias (Figure 2.45). However, the model significance is questionable with $p = 0.21$.

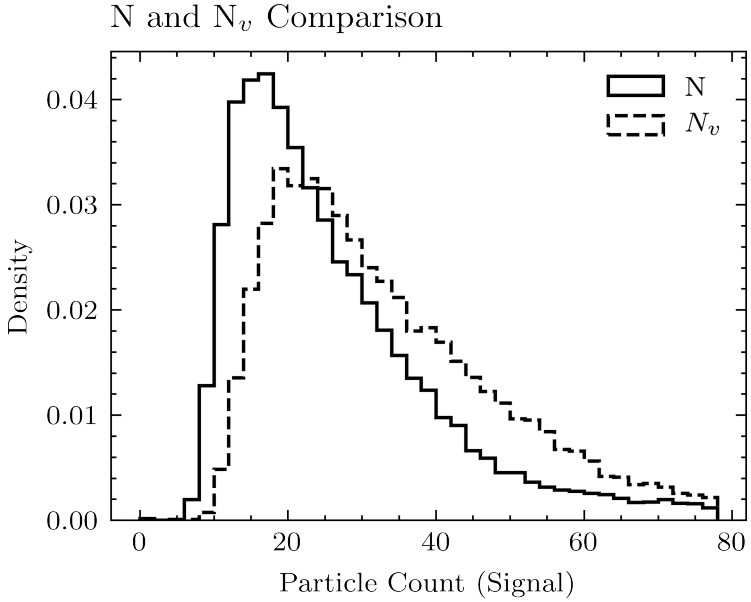

Figure 2.43. Histograms of N and N_v Summary Statistics of PA Public Dataset

We continue with reconstruction of the Zenith angle. Likewise with the optimization of N, the optimization of *tmean* do not yield a minimum. So, we set $\varepsilon_{tmean} = 50$ as the minimum value, without compromising reconstruction of any event. Next, we optimize *tstd* (Figure 2.48) and *tskew* (Figure 2.49). Respectively the distance metrics are set to $\varepsilon_{tstd} = 50$ and $\varepsilon_{tskew} = 0.1$.

2.6. Results and Discussion

We have reconstructed Energy and Zenith of CR showers with various energy levels over a hypothetical continuous SD, a grid detector modeled from PA Observatory SD and the publicly published events of PA Observatory. We have utilized the ABC Rejection and Sequential methodologies by selecting summary statistics with approximate sufficiency approach. We report all results in Table 2.3. Table consists of detector

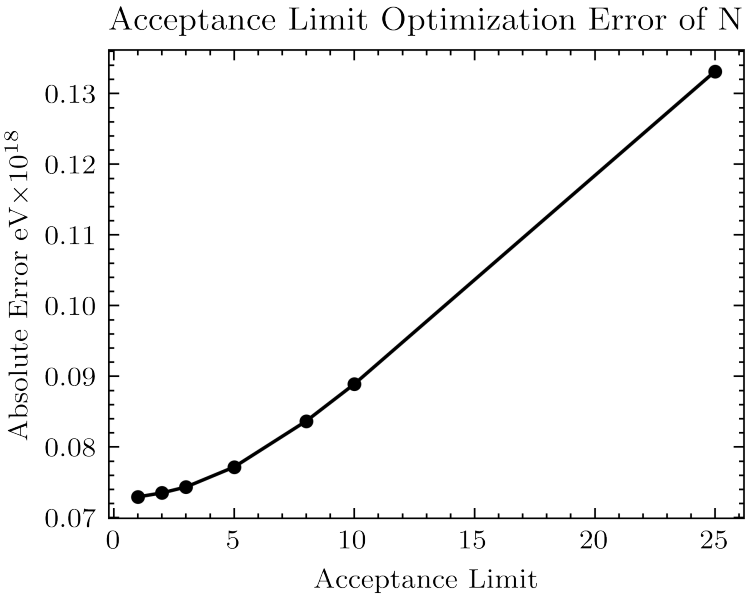

Figure 2.44. Mean Absolute Error of Reconstructed Energy by the Acceptance Limit of N for ABC Rejection Algorithm run over PA Public Dataset

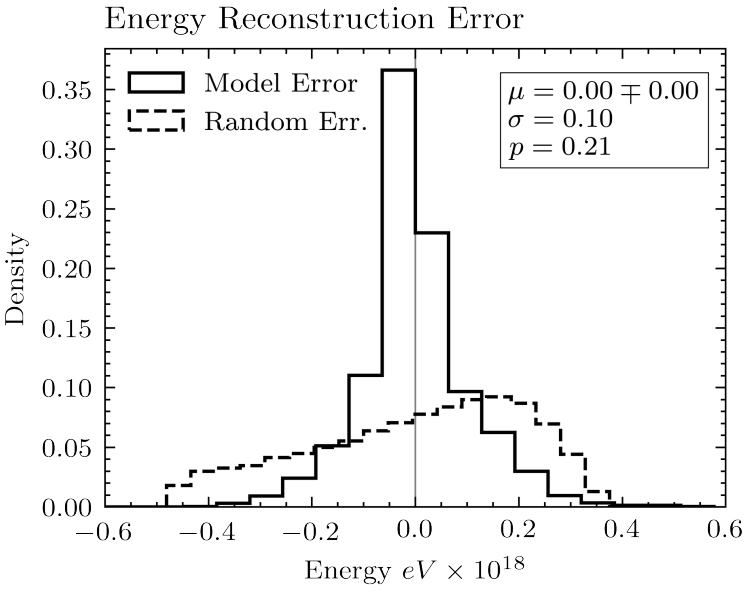

Figure 2.45. Error Histogram of Energy Reconstruction for ABC Rejection Algorithm run over PA Public Dataset

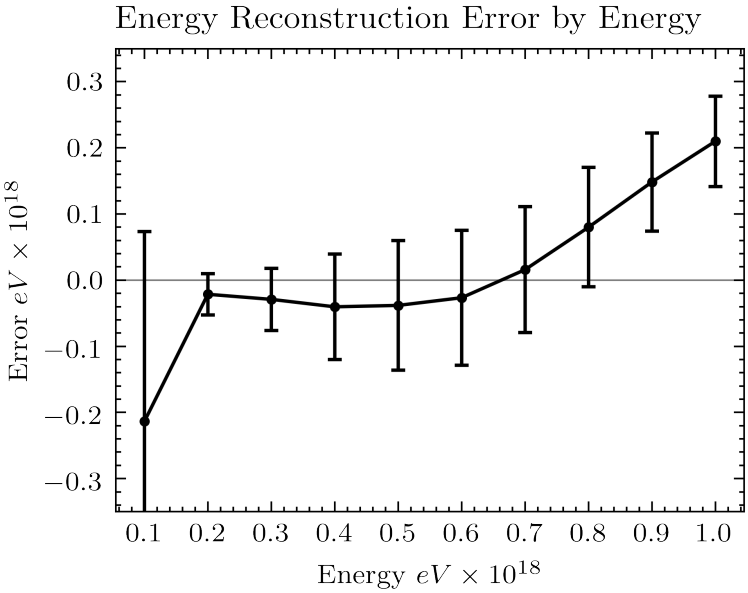

Figure 2.46. Error of Energy Reconstruction by Reconstructed Energy Range for ABC Rejection Algorithm run over PA Public Dataset

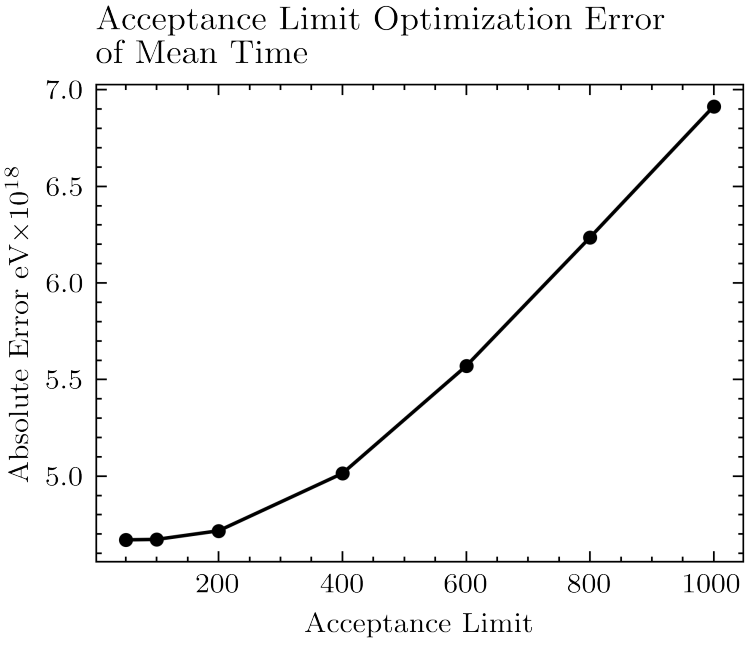

Figure 2.47. Mean Absolute Error of Reconstructed Zenith Angle by the Acceptance Limit of *tmean* for ABC Rejection Algorithm run over PA Public Dataset

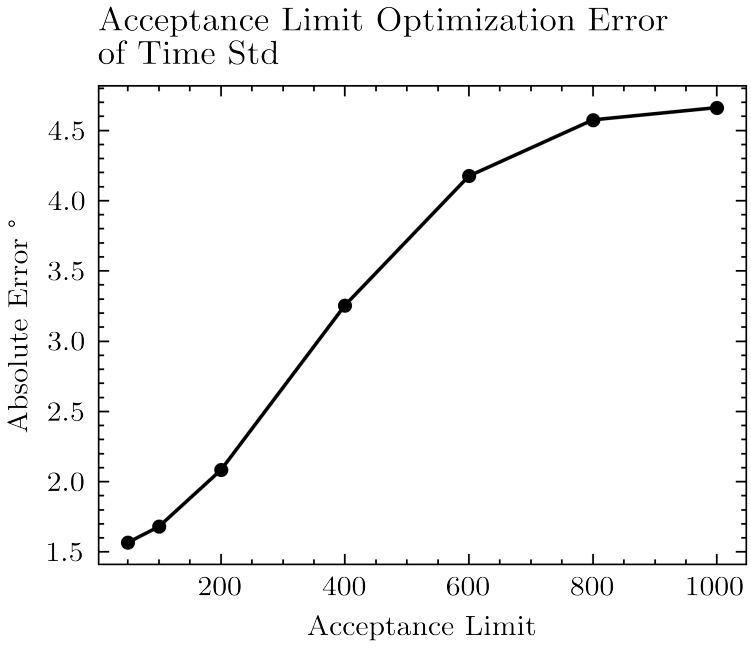

Figure 2.48. Mean Absolute Error of Reconstructed Zenith Angle by the Acceptance Limit of *tstd* for ABC Rejection Algorithm run over PA Public Dataset

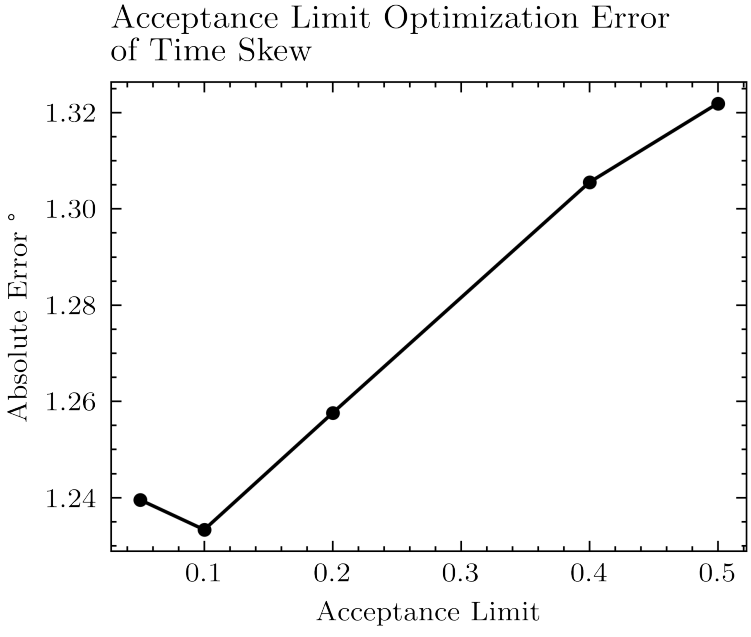

Figure 2.49. Mean Absolute Error of Reconstructed Zenith Angle by the Acceptance Limit of *tskew* for ABC Rejection Algorithm run over PA Public Dataset

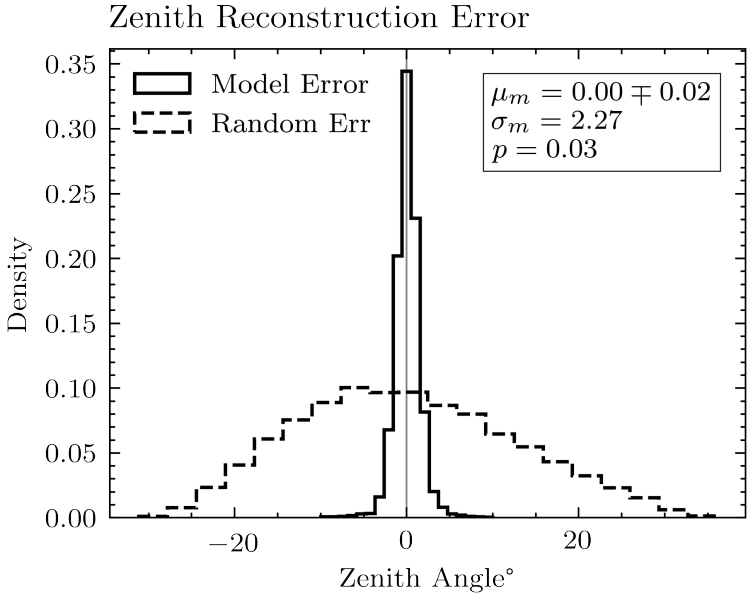

Figure 2.50. Error Histogram of Zenith Angle Reconstruction for ABC Rejection Algorithm run over PA Public Dataset

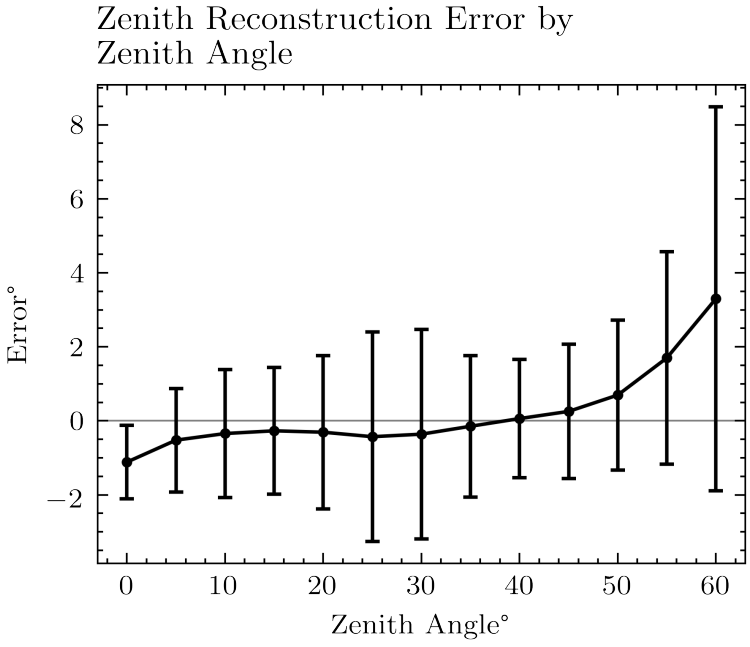

Figure 2.51. Error of Zenith Angle Reconstruction by Reconstructed Incident Angle Range for ABC Rejection Algorithm run over PA Public Dataset

type, reconstructed parameter, methodology used, energy range, mean error, standard deviation of error, absolute error, percentage of reconstructed events and p-value of the reconstruction model. Now we will elaborate and discuss these results.

We have achieved significant models both for reconstruction of the Energy and Zenith with ABC rejection model run over continuous surface detector without bias. For 3.1% of the events energy could not be reconstructed and for 1.6% of events zenith angle could not be reconstructed. There is a balance between number of simulations, strictness or number of acceptance criteria and the number of unreconstructed events. In theory, if we were able to generate infinite number of simulations to reconstruct the observations, then we could set the acceptance distance ε near zero. The simulations would represent all the possible observations and the probability to draw an identical simulation to the observation would be greater than zero. Then, we would be able to reconstruct any observation. With limited number of simulations, we used a set of informative summary statistics and set the acceptance distance as small as possible to build significant models. Additional summary statistics can be used with smaller acceptance distance when more simulations are available.

Next, we tried to implement the model based on a MCMC Particle Filter combined with ABC, namely the Sequential ABC. We used the same dataset used previously for ABC Rejection model. This means that the sequential algorithm draws samples from a fixed set of simulations, which is contradictory with the sequential approach as it needs to generate fresh simulations at each iteration. We thought an intermediate step to be helpful while developing the sequential model. We had expected similar results compared to ABC Rejection model in this transition, if we were to implement the algorithm properly. The results were slightly worse. Sequential model optimizes itself with each iteration by drawing closer samples to the observation. Thus, within a limited set of samples, sequential algorithm can not find close enough simulations to the observation and finish the loop before convergence is satisfied. On the other hand, ABC rejection model is optimized to its best performance over the dataset, so the slightly worse result of the sequential algorithm were expectable.

We demonstrated the superiority of the Sequential ABC at a lower energy range where we could generate simulations faster. To form a baseline, first we created a low energy events dataset with CORSIKA and ran the ABC rejection algorithm with optimized distance metrics. We got a significant model in return. However, the energy reconstruction accuracy at $10^{13}eV$ energy level is worse than the $10^{15}eV$ level. The only change in the experimental setup is the energy level of the simulations. The observation level and the distribution of the grid SD remained exactly the same. Thus, CR showers with lower energy create smaller number of particles to be detected by the SD at the same observation level. Consequently, the distinction between the observations diminish with decreasing Energy. This decreasing signal-to-noise ratio caused the worse reconstruction accuracy. However, this is not a concern as we compare the rejection and sequential models at the same energy levels. Next, we ran the Sequential ABC algorithm to reconstruct the same low energy events, but this time with ability to generate new samples as required by the sequential algorithm. Sequential algorithm corrected the bias in reconstruction with the rejection model and achieved a far better accuracy. The reconstruction accuracy increased with each iteration (Figure 2.25). Put another way, the model self optimized itself for each individual observation by advancing iterations. We accomplished to present the superiority of the Sequential ABC model.

We continued the experiments by modeling the PA Grid SD. Creation of a grid, rather than using a continuous surface caused decreasing signal-to-noise ratio. We have almost lost the 98% of the particle information. Nevertheless, we were still able to develop significant models to reconstruct Energy and Zenith. At first, to reconstruct the Energy we have tried to use the same model that we have used to reconstruct the Energy over continuous SD. We have re-optimized the parameters and ran the algorithm. The model turned out to be insignificant. We revised the model to be more strict in acceptation of the samples to create the posteriors. We achieved this by introducing additional summary statistics. This stricter acceptation criteria caused 18.5% observations left to be unreconstructed, but in return we were able to construct a significant model for reconstruction of Energy over a grid detector. We analyzed the unreconstructed events an realized that higher energy events were predominate. This

is intelligible as secondary particles of higher energy spread over a larger area and with higher dispersion. As the variability of the signal pattern increases, the models require more samples in that range to find comparable simulations to the observation. In point of fact, this is a drawback of the rejection algorithm. It can be resolved by running the sequential ABC model, as it draws new samples from the required prior. Next, we have developed our model to reconstruct the Zenith over grid SD. We have re-optimized the acceptation criteria for the same summary statistics used in reconstruction of Zenith over continuous grid. While no significant bias is introduced, the standard deviation of the error increased from 4.18° to 10.6°. In addition, we were not able to reconstruct 47.5% of the events, without compromising the significance of the model. We can say that loss of information affects the reconstruction of the Zenith more than the Energy. This is perceivable as, in addition to lost particle count information, arrival of time information is also lost, which is important for the reconstruction of Zenith. A single detector on a grid aggregates the signal of multiple particles arrived to give in a single time of incidence. Lastly we have randomized the tank locations in the grid and tried to reconstruct the energy. The randomization process extended the span of the grid and increased the tank density in the center. In spite of, the loss of orderly fashion and fixed distance between the tanks, the ABC model reconstructed the Energy significantly, with slightly worse accuracy compared to orderly grid.

Finally, we reconstructed the events from the PA Public Dataset with the knowledge attained from our experiments. Both in the parameter optimization of Energy and Zenith reconstruction we have realized that a minimum distance could not be achieved. We set the parameter minimum at the level where the models can reconstruct most of the events. Furthermore, fewer summary statistics were enough for significant and highly accurate results. There may be a few reasons for this. Firstly, the PA Public Data set may have been put together with reconstructed events with higher likelihood. Secondly, the observatory may have cut some events below a predefined signal-to-noise ratio and do not reconstruct them at all. Lastly, the events in the dataset may be the mean representations of the observations with same reconstructed parameters. The significance of the Energy reconstruction model is questionable with $p = 0.21$. We think that this is caused by the shape of Energy prior of the events in the dataset.

As the prior is not uniform a selection bias is introduced, thus the random model results can draw samples from the dataset which has a mean closer to the reconstructed observation. We can observe the bias in the random model caused by the shape of the prior in Figure 2.45. On the other hand, Energy reconstruction model is highly accurate with absolute error of $0.072 \times 10^{18} eV$ compared to $0.173 \times 10^{18} eV$ error of the corresponding random model.

Key performance metrics of the parameter reconstruction at Pierre Auger Observatory are presented in the collaboration paper [27]. EAS arriving with zenith angles below 60° have energy reconstruction accuracy between 12% and 22% when the EAS is only reconstructed with the data collected from the surface detectors. Low energy EAS starting from $4 \times 10^{18} eV$ have lower accuracy and EAS with highest energies above $10^{19} eV$ have better energy reconstruction accuracy. Accuracy of zenith angle reconstruction changes with the count of signal bearing surface detectors. Angular resolution is 1.6° when 3 stations are involved in measurement and 0.9° for more than 5 detectors. Our reconstruction errors over the public dataset are in the same range as can be seen in Table 2.4. The Energy reconstruction error is slightly better, save that we discarded the highest energy events with fewer number of observations, as we were not be able to reconstruct them.

To conclude, we were able to develop significant reconstruction models both with ABC Rejection and Sequential ABC methodologies run over various detector setups. We have showed that this experimental methodology is applicable to the real world observations.

Detector	**Recons.**	**Model**	**Range**	**mu**	**std**	**abs**	**rec%**	**p**
Continuous	Energy eV	Rejection	14	.015	.153	.118	96.9	.014
Continuous	Zenith	Rejection	14	-.022	4.18	3.22	98.4	.006
Continuous	Energy eV	Rejection	13	.060	.145	.130	97.6	.001
Continuous	Energy eV	Sequential	13	.026	.109	.086	100	.001
Grid	Energy eV	Rejection	14	.001	.160	.127	99.7	.990
Grid	Energy eV	Rejection(Str.)	14	.001	.168	.132	81.5	.001
Grid	Zenith	Rejection	14	.100	10.6	8.48	52.5	.001
Random Grid	Energy eV	Rejection	14	.002	.170	.133	92.2	.001
PA Public	Energy eV	Rejection	18	.000	.105	.073	99.3	.210
PA Public	Zenith	Rejection	18	.003	2.27	1.23	97.9	.030

Table 2.3. Reconstruction Accuracy and Significance Results of the Computational Experiments

	Benchmark	**Reconstruction**
Energy	$14 - 22\%$	13.6%
Zenith	$0.9 - 1.6^\circ$	1.23°

Table 2.4. PA Benchmark Comparison

3. CONCLUSION

This research aimed to utilize a likelihood free particle filter namely the Approximate Bayesian Computation methodology to reconstruct the initial parameters of Energy and Zenith angle of Extensive Cosmic Ray Air Showers. The results clearly indicate that significant ABC models can be built for parameter reconstruction in particle astrophysics. Where, we have defined the significance of our models with t-test between the reconstruction accuracy distributions of developed and random models. We see that the count of simulations, selection of summary statistics and their acceptance criteria are important factors when designing significant reconstruction algorithms.

In the first place, we picked ABC methodology for the reconstruction problem as, CR development and particle generation do not have a closed-form solution. This approach aligns with the previous studies which utilized the ABC method to solve problems related to population genetics, evolution and intricate dynamical systems. All of these problems can be forward simulated, but reaching to a mathematical solution is impracticable or impossible. The same approach proved to be successful for astroparticle physics problems within this research.

It is revealed that there should be a subjectivity derived from the knowledge of the domain for generating the summary statistics. The common moments like mean, standard deviation and skewness are not always the best picks as candidates. Subjective summary statistics are proved to be meaningful with statistical tests. ABC algorithms should be tailor-made per different area of research and the specific problem to be solved.

To better understand the implications of these results, future studies could study the higher energy range with supercomputers. Simulations could be generated with a wider range of input parameters to safely generalize the results. More simulations used to develop the ABC models may ensure better distinction between posteriors and better model accuracy. Thus, the accuracy limits of our approach can be analyzed

with a computational power that we could not attain. In spite of the limited available computational resources, we have successfully built significant reconstruction models.

REFERENCES

1. Hillas, A. M., "Cosmic rays: Recent progress and some current questions", *arXiv preprint astro-ph/0607109*, 2006.

2. Matthews, J., "A Heitler model of extensive air showers", *Astroparticle Physics*, Vol. 22, No. 5-6, pp. 387–397, 2005.

3. CernNews, "Cosmic rays discovered 100 years ago", *https://home.cern/news/news/physics/cosmic-rays-discovered-100-years-ago*, 2012.

4. Kampert, K.-H. and A. A. Watson, "Extensive air showers and ultra high-energy cosmic rays: a historical review", *The European Physical Journal H*, Vol. 37, No. 3, pp. 359–412, 2012.

5. Mollerach, S. and E. Roulet, "Progress in high-energy cosmic ray physics", *Progress in Particle and Nuclear Physics*, Vol. 98, pp. 85–118, 2018.

6. "The Pierre Auger Cosmic Ray Observatory", *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, Vol. 798, pp. 172 – 213, 2015.

7. Apel, e. W., J. Arteaga, A. Badea, K. Bekk, M. Bertaina, J. Blümer, H. Bozdog, I. Brancus, P. Buchholz, E. Cantoni *et al.*, "The KASCADE-grande experiment", *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, Vol. 620, No. 2-3, pp. 202–216, 2010.

8. Bai, X. and IceCubeCollaboration, "The IceCube/IceTop air shower experiment", *Nuclear Physics B-Proceedings Supplements*, Vol. 175, pp. 415–420, 2008.

9. Chiba, N., S. Kawaguchi, K. Hashimoto, K. Honda, N. Kawasumi, I. Tsushima, N. Hayashida, M. Honda, M. Nagano, H. Ohoka *et al.*, "Akeno Giant Air Shower Array (AGASA) covering 100 km sup 2 area", *Nuclear Instruments and Methods in Physics Research*, Vol. 311, pp. 338–349, 1992.

10. Gallant, Y., A. Garyaka, L. Jones, R. Martirosov and N. Nikolskaya, "Search for PeV-ray sources with the GAMMA experiment", *Proceedings of the 29th International Cosmic Ray Conference*, pp. 101–104, 2005.

11. Gaisser, T. K., "Cosmic rays at the knee", *Energy Budget in the High Energy Universe*, pp. 45–55, World Scientific, 2007.

12. Heitler, W., *The quantum theory of radiation*, Oxford University Press, 1954.

13. Aab, A., P. Abreu, M. Aglietta, E. Ahn, I. Al Samarai, I. Albuquerque, I. Allekotte, J. Allen, P. Allison, A. Almela *et al.*, "Searches for anisotropies in the arrival directions of the highest energy cosmic rays detected by the Pierre Auger Observatory", *The Astrophysical Journal*, Vol. 804, No. 1, p. 15, 2015.

14. Nagano, M., D. Heck, K. Shinozaki, N. Inoue and J. Knapp, "Comparison of AGASA data with CORSIKA simulation", *Astroparticle Physics*, Vol. 13, No. 4, pp. 277–294, 2000.

15. Ave, M., P. Bauleo, A. Castellina, A. Chou, J. Harton, R. Knapik, G. Navarra, P. A. Collaboration *et al.*, "The accuracy of signal measurement with the water Cherenkov detectors of the Pierre Auger Observatory", *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, Vol. 578, No. 1, pp. 180–184, 2007.

16. Heck, D., J. Knapp, J. N. Capdevielle, G. Schatz and T. Thouw, *CORSIKA: a Monte Carlo code to simulate extensive air showers.*, 1998.

17. Falkenburg, B. and W. Rhode, *From Ultra Rays to Astroparticles*, Springer, 2012.

18. Marin, J.-M., P. Pudlo, C. P. Robert and R. J. Ryder, "Approximate Bayesian computational methods", *Statistics and Computing*, Vol. 22, No. 6, pp. 1167–1180, 2012.

19. Blum, M. G., M. A. Nunes, D. Prangle, S. A. Sisson *et al.*, "A comparative review of dimension reduction methods in approximate Bayesian computation", *Statistical Science*, Vol. 28, No. 2, pp. 189–208, 2013.

20. Turner, B. M. and T. Van Zandt, "Hierarchical approximate Bayesian computation", *Psychometrika*, Vol. 79, No. 2, pp. 185–209, 2014.

21. Casella, G., C. P. Robert, M. T. Wells *et al.*, "Generalized accept-reject sampling schemes", *A Festschrift for Herman Rubin*, pp. 342–347, Institute of Mathematical Statistics, 2004.

22. Sunnåker, M., A. G. Busetto, E. Numminen, J. Corander, M. Foll and C. Dessimoz, "Approximate bayesian computation", *PLoS computational biology*, Vol. 9, No. 1, p. e1002803, 2013.

23. Alshamlan, H. M., G. H. Badr and Y. A. Alohali, "Abc-svm: artificial bee colony and svm method for microarray gene selection and multi class cancer classification", *Int. J. Mach. Learn. Comput*, Vol. 6, No. 3, p. 184, 2016.

24. Jennings, E. and M. Madigan, "astroABC: an approximate bayesian computation sequential Monte Carlo sampler for cosmological parameter estimation", *Astronomy and Computing*, Vol. 19, pp. 16–22, 2017.

25. Prangle, D., "Summary statistics in approximate Bayesian computation", *arXiv preprint arXiv:1512.05633*, 2015.

26. Allekotte, I., A. Barbosa, P. Bauleo, C. Bonifazi, B. Civit, C. Escobar, B. García, G. Guedes, M. G. Berisso, J. Harton *et al.*, "The surface detector system of the Pierre Auger Observatory", *Nuclear Instruments and Methods in Physics Research*

Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, Vol. 586, No. 3, pp. 409–420, 2008.

27. Aab, A., P. Abreu and M. Aglietta, "Pierre Auger Collab", *Phys. Rev. D*, Vol. 91, p. 092008, 2015.

APPENDIX A: Sample CORSIKA Input

```
RUNNR 1 run number
EVTNR 1 number of first shower event
NSHOW 10000 number of showers to generate
PRMPAR 14 particle type of prim.  particle
ESLOPE 0 slope of primary energy spectrum
ERANGE 5.E3 1.E5 energy range of primary particle
THETAP 0.  0.  range of zenith angle (degree)
PHIP 0.  0.  range of azimuth angle (degree)
SEED 1 0 0 seed for 1.  random number sequence
SEED 2 0 0 seed for 2.  random number sequence
OBSLEV 1452.E2 observation level (in cm)
CORECUT 2.E2 core radius cut
MAGNET 19.52 -14.17 magnetic field
EXIT terminates input
```

APPENDIX B: Sample Codes

Read CORSIKA Output and Bin Secondary Particles (Detector Simulation)

```
from sapphire import corsika
import matplotlib.pyplot as plt
import numpy as np
import math
import pandas as pd
from datetime import datetime
import pickle

event_dictionaries = []
event_signals = []
datfile = "DAT000506"
cfile = corsika.reader.CorsikaFile('fixed-theta-events/{}'.format(datfile))
event_id_prefix = int(datfile.replace("DAT",""))
for event in cfile.get_events():
   event_id = int("{}{}".format(event_id_prefix, event_counter))
   zenith = round( event.get_header().zenith * 180 / np.pi , 1)
   energy = event.get_header().energy

   raw_particles = []
   for particle in event.get_particles():
      raw_particles.append(particle)

   prt_df = pd.DataFrame(raw_particles)
   prt_df.columns = ['Px', 'Py', 'Pz', 'X', 'Y', 't', 'Particle', 'R',
'hadron_generation', 'observation_level', 'phi']
   prt_df['X'] = prt_df['X'] - prt_df['X'].mean()
   prt_df['Y'] = prt_df['Y'] - prt_df['Y'].mean()
   prt_df['X'] = (prt_df['X']) / 100
   prt_df['Y'] = (prt_df['Y']) / 100 # cm to meters


   lepton_filter = [True if part in (2, 3, 5, 6, 131, 132) else False for part in
prt_df['Particle'].values]
   filtered_df = prt_df[lepton_filter]
```

```
    signals = {}
    for key, row in minigrid.iterrows():

        xtankmin = row[0] - r_minitank
        xtankmax = row[0] + r_minitank

        ytankmin = row[1] - r_minitank
        ytankmax = row[1] + r_minitank

        signal = filtered_df[
            (filtered_df["X"] >= xtankmin) & (filtered_df["X"] <= xtankmax) &
            (filtered_df["Y"] >= ytankmin) & (filtered_df["Y"] <= ytankmax)]

        if len(signal > 0):
            signals[key] = signal['t'].values
    if len(signals) < 1:
        print("no signal", event_id)
        continue

    event_signals_dict = {
        "EventId": event_id,
        "TankId": list(signals.keys()),
        "Signal": [len(val) for val in signals.values()],
        "ns": [min(val) for val in signals.values()]}
    event_signals_df = pd.DataFrame(event_signals_dict)
    event_dict = {
        "EventId": event_id,
        "Theta": zenith,
        "Phi": 0,
        "Energy": energy / 10**14}
    event_dictionaries.append(event_dict)
    event_signals.append(event_signals_df)

event_dictionaries_df =
pd.DataFrame(event_dictionaries).reset_index(drop=True)
event_signals_df = pd.concat(event_signals).reset_index(drop=True)
```

```
event_dictionaries_df.to_csv("run_{}_events.csv".format(event_id_prefix))
event_signals_df.to_csv("run_{}_signals.csv".format(evaent_id_prefix))
```

Generate Summary Statistics

```
tank_signals['X'] = minigrid.loc[tank_signals['TankId']]['Easting'].values
tank_signals['Y'] = minigrid.loc[tank_signals['TankId']]['Northing'].values

q625 = tank_signals['X'].quantile(0.625)
q750 = tank_signals['X'].quantile(0.750)
q875 = tank_signals['X'].quantile(0.875)

n625 = tank_signals[tank_signals['X'] >
q625].groupby("EventId")['Signal'].sum()
n750 = tank_signals[tank_signals['X'] >
q750].groupby("EventId")['Signal'].sum()
n875 = tank_signals[tank_signals['X'] >
q875].groupby("EventId")['Signal'].sum()
events['n625'] = n625.reindex(events.index).fillna(0)
events['n750'] = n750.reindex(events.index).fillna(0)
events['n875'] = n875.reindex(events.index).fillna(0)
events['n'] = n.reindex(events.index).fillna(0)

t_mean = {int(es.iloc[0]['EventId']): (es['ns'] - es['ns'].min()).mean() for es in
tank_signals}
t_std = {int(es.iloc[0]['EventId']): (es['ns'] - es['ns'].min()).std() for es in
tank_signals}
t_max = {int(es.iloc[0]['EventId']): (es['ns'] - es['ns'].min()).max() for es in
tank_signals}
```

ABC Rejection

```
accept = {'n': 16, 'n625':4, 'n750':2, 'n875': 1}
features = accept.keys()
e_errors = []
posteriors = []
for i,observation in df.iterrows():
```

```
    dfo = df[df.index != i] # do not include the tested sample
    errors = abs(dfo - observation)
    errors['id'] = dfo['id']
    slx = None
    for feature in features:
        if slx is None:
            slx = errors[feature] < accept[feature]
        else:
            slx = slx & (errors[feature] < accept[feature])

    posteriors.append(dfo[slx])
    energy = dfo[slx]['Energy'].mean()
    e_errors.append(observation['Energy']-energy)
```

ABC Rejection Optimization

```
opt_n = [50,100,125,250,500,1000]
opt_n625 =  np.arange(25,250,25)
opt_n750 = np.arange(150,250,25)
opt_n875 = np.arange(5,60,5)
opt_maxmin = np.arange(5,40,5)
optim_params = list(itertools.product(opt_n,opt_n625, opt_n750, opt_n875,
opt_maxmin))

optim_results = []
for i, params in enumerate(optim_params):
    accept = {'n': params[0], 'n625': params[1], 'n750':params[2], 'n875':
params[3], 'maxmin': params[4]}
    posteriors, e_errors = abc_reject(accept, df)
    optim_results.append(e_errors)
```

Sequential ABC

```
sample = 1000
iterate = 15
prior_ratio = 0.8
random_ratio = 1 - prior_ratio
```

```
acceptance = 10

prior_count = int(prior_ratio * sample)
random_count = int(random_ratio * sample)

errors = []
iteration_errors = {}
for i in range(iterate):
   iteration_errors[i] = []
posteriors = []
count = 0
for i, observation in observations.iterrows():
   if count % 100 == 0:
      print(count, "{:.0f}%".format(count/2133*100), datetime.today())
   count = count + 1
   dfo = alld[alld.index != i]
   dfo_init = df[df.index != i].set_index("id")
   posterior = []
   difference =  dfo_init.sample(init_sample) - observation
   difference = difference.drop_duplicates()
   euclidean = np.sqrt(difference['n']**2 + difference['n625']**2)
   euclidean = euclidean[euclidean > 0]
   accepted = difference.loc[euclidean.sort_values().head(acceptance).index]
   posterior.append(accepted.index) # do not accept the initial smaple.
   euclidean_lim = euclidean.mean()


   for n_iterate in range(iterate):
      accepted_euc = euclidean.loc[accepted.index]
      accepted_euc = accepted_euc[accepted_euc > 0]
      weights = 1 / accepted_euc
      weights = weights / weights.sum()
      weights = (weights * prior_count).astype(int)

      priors = []
      for i, energy in alld.loc[accepted.index]['energy'].iteritems():
```

```
            prior = dfo[(dfo['energy'] > (energy - 0.005) ) & (dfo['energy'] < (energy
+ 0.005))]
            samplesize = weights.loc[[i]].iloc[0]
            if samplesize > len(prior):
               samplesize = len(prior)
            priors.append(prior.sample(samplesize))
         if len(priors) < 1:
            priors.append(dfo.sample(random_count)) # randomize
         prior = pd.concat(priors)

         difference = prior - observation
         difference = difference.drop_duplicates()
         euclidean = np.sqrt(difference['n']**2 + difference['n625']**2)
         euclidean = euclidean[euclidean > 0]
         accepted = difference.loc[euclidean[euclidean < euclidean_lim].index]
         if len(accepted) > acceptance:
            accepted =
difference.loc[euclidean.sort_values().head(acceptance).index]

         if len(accepted) > 0:
            if len(posterior) > 0:
               novel = [a for a in accepted.index if a not in
np.concatenate(posterior)]
            else:
               novel = accepted.index
            if len(novel) > 0:
               posterior.append(novel)
            euclidean_lim = euclidean.mean()

         if len(posterior) > 0:
            energy_it = alld.loc[np.concatenate(posterior)]['energy'].mean()
            error_it = (energy_it - observation['energy'])
            iteration_errors[n_iterate].append(error_it)

      if len(posterior) > 0:
         energy = alld.loc[np.concatenate(posterior)]['energy'].mean()
         error = (energy - observation['energy'])
```

```
        errors.append(error)
        posteriors.append(np.concatenate(posterior))
    else:
        errors.append(None)
        posteriors.append(None)
```