A GENETIC ALGORITHM FOR ASTROPARTICLE PHYSICS STUDIES - REVIEW

ALI AHMAD AND FLORENT I. MUSTAFAJ

Introduction

The energy spectra of cosmic rays (CR) have been fundamental in the investigations of astrophysical phenomena in the universe. Several experiments in high altitudes or in space (such as AMS-02, ATIC-2, CALET, CREAM, DAMPE, Fermi-LAT & PAMELA) as well as ground experiments (such as HESS, CTA & LHAASO) have yielded spectra containing information on the physical processes behind cosmic rays (mainly transport effects & acceleration processes in CR), as well as detections due to additional sources, such as dark matter. Traditionally, models of these spectra do not account for spectral hardening or softening present in some nuclei spectra. To take these effects into account, more sophisticated models with a higher number of free parameters must be used. Obtaining these free parameters from data is an optimization problem that cannot be solved analytically due to high dimensionality and thus, numerical methods must be employed. A common approach to this is Markov-Chain Monte Carlo (MCMC) methods [2].

An alternative to the outlined traditional approach to this optimization problem is to use a genetic algorithm. Genetic algorithms are a global optimization strategy useful for solving optimization problems with large/high-dimensional and ill-behaved search spaces. Genetic algorithms seek to maximize what is usually dubbed a *fitness* function through successive evolutionary steps. The relevant concepts are "mutations", which, like in the biological case, describes a random change in a member of a population (the population of solutions, in this case), "crossover" (i.e. "breeding") which describes some way of combining solutions to get new solutions and "natural selection", which describes selecting solutions according to fitness [3]. In this work, we review an attempt to apply genetic algorithms to obtain the free parameters in models describing the energy spectra of cosmic rays.

Model of Cosmic-Ray Propagation

The model under consideration is the following two-dimensional transport equation

$$\frac{\partial \psi}{\partial t} = Q + \vec{\nabla} \cdot (D \vec{\nabla} \psi) - \psi \Lambda + \frac{\partial}{\partial E}(\dot{E} \psi).$$

Where $\psi(E, r, z)$ is the particle number density, a source term $Q(\Lambda_j, \psi_j)$ with both a primary and a secondary production term stemming from spallation of heavier nuclei types, $\Lambda(\beta, n, \sigma)$ is then the destruction rate for collisions of gas, $\dot{E}$ contains information on the ionization, Coloumb energy loss and the radiative cooling of CR leptons & finally $D(D_0, \rho_0, \rho, r, z, \delta(z))$ is the spatial diffusion coefficient (with information on the particle magnetic rigidity, diffusion coefficient of rigidity, and normalisation parameters). Furthermore, the model is given with some boundary conditions for $r = r_{max}$ and $z = \pm L$.

In total there are 8 parameters to be estimated, substantiating the complexity of the problem, which are given as D_0 (the normalisation on the spatial diffusion coefficient), ξ & L (limit parameters for the spatial-dependent propagation model used for $\delta(z)$), χ (additional coefficient in outer halo for the normalisation D_0), δ (power-law index of diffusion coefficient of rigidity), ν (spectral indices for $z > 1$), and finally Δ & $\Delta\nu$ (primarily differences in inner & outer halo, as well as for the spectral indices in these two regions).

The Genetic Algorithm

In order to implement the genetic algorithm, one needs a fitness function to be maximized, which in this case is the likelihood. This has been written in terms of the χ^2 as

$$F(\vec{P}) = e^{-\frac{1}{2}\chi^2(\vec{P})}.$$

Presumably, this comes from Wilk's theorem, that relates the ln-likelihood ratio to the chi square by a factor of -2 [1]. However, it is not clear and has not been specified why the likelihood was written in terms of the chi square, as opposed to simply using the more exact, direct expression for the likelihood. Perhaps this was done to take into account the errors on the data, as this is included in the calculation of the chi square. However, since only the parameters, and not the errors, are estimated using the genetic algorithm, it is not clear why this would be a benefit.

The algorithm starts with an initial guess, $\vec{P}$, of the parameter values, after which a sequence of steps are performed. The first of these is *mutation.* Here, an orthonormal basis (ONB) of n vectors is generated by the Gram-Schmidt process. Then, each vector in the ONB is multiplied by a random number, yielding a set of random vectors $\vec{N}_i$. Each of these are added to $\vec{P}$, such that $\vec{P}$ mutates into n different vectors. Then, one can get the set of parameters with the maximum fitness $\vec{P}_m$ and repeat this process which will iteratively improve the fitness (likelihood). However, if this process falls into a local maximum, one needs a new way of selecting solutions for the next generation, which is dubbed *roulette wheel selection.* Here, the sum of all fitness functions is computed

$$\mathcal{F} = \sum_{i=1}^{n} F(\vec{P_i}),$$

after which a random number $l \in [0, \mathcal{F}]$ is generated. Finally, one calculates $\mathcal{F}_k = \sum_{i=1}^{n} F(\vec{P_i})$, iterating over k until $\mathcal{F}_k > l$, after which the process is stopped and $\vec{P}_k$ is selected. This way, the probability of a solution being selected is proportional to its fitness.

As mentioned in the introduction, one feature of genetic algorithms is the *crossover* operation, where new solutions ”child solutions” are generated by, in some way, combining ”parent solutions”. This is one of the central features that contributes to the efficiency of genetic algorithms [3]. It seems that this aspect has not been focused on in the above algorithm. The roulette wheel selection in a very loose sense does combine solutions by considering the sum, however the solution that is ultimately chosen for the next generation is simply one of the solutions in the parent generation; not a combination of them. Perhaps, the algorithm could be further improved by implementing crossover.

As can be seen in table A, the solution from the genetic algorithm matches the solution from the MCMC method. However, the genetic algorithm is a lot more efficient. It turns out that in order to get a fit with a p-value of around 0.9 (calculated from a chi square test), it takes 120 trials for the genetic algorithm, whereas for the MCMC process takes around 3000 trials. Thus, the genetic algorithm can find a solution with a p-value greater than 0.9, 70 times faster as compared to the MCMC (see figure 1).

However, one downside of the genetic algorithm is that it does not provide errors on the extracted parameters, whereas the MCMC process does. Since the errors serve a crucial role in the analysis, the genetic algorithm cannot replace the MCMC method in this case. However, it is suggested that it should be used in conjuction with the MCMC method; first using the genetic algorithm to estimate the parameters and then use those as the priors for the MCMC method.

Perhaps, the idea behind this is that having already estimated the parameters using the genetic algorithm, it will be much quicker to extract the errors from the MCMC, as opposed to performing the entire optimization with MCMC. However, it does not seem that this was tested by the authors. Thus, considering the *whole* calculation of parameters and errors, it is not clear the extent to which using the genetic algorithm in conjunction with the MCMC allows for faster execution or the use of lower-powered hardware.

Conclusion

The use of genetic algorithms in finding the parameters in high-dimensional problems involving cosmic ray propagation seems promising in the light of the massively improved execution time as compared to traditional MCMC methods. This is strengthened by the fact that there is likely still room for further improvement of the genetic algorithm. However, the lack of ability to estimate errors using the genetic algorithm is a major obstacle that, at present, limits the practical benefits associated with using the genetic algorithm as compared to the traditional MCMC approach. Further research into the topic should take a closer look at the practical benefits of applying genetic algorithms in conjunction with MCMC methods and perhaps also considering ways to extract error estimates from the genetic algorithm without MCMC.

Appendix A. - GA vs. MCMC fit results

Param.	GA best-fit	MCMC best-fit
GA vs. MCMC fit results (Table 1 in review article)		
L [kpc]	6.70	6.70
D_0 $10^{28}[cm^2s^{-1}]$	2.22	$2.18^{+0.69}_{-0.34}$
δ [...]	0.17	$0.19^{+0.04}_{-0.07}$
Δ [...]	0.55	$0.56^{+0.12}_{-0.05}$
ξ [...]	0.17	$0.22^{+0.11}_{-0.07}$
χ [...]	0.33	$0.30^{+0.09}_{-0.21}$
$\Delta\nu$ [...]	0.104	$0.096^{+0.023}_{-0.06}$
ν [...]	2.27	$2.29^{+0.11}_{-0.09}$

Table 1. Table 1 in reviewed article comparing the results of GA vs. MCMC scans for the parameters used for the CR propagation model. Best-fit values & their corresponding 1σ uncertainties (68% C.L, with the exception of D_0) provided.

Appendix B. - Ga vs. MCMC χ^2/p-value as a func. of gen/trials

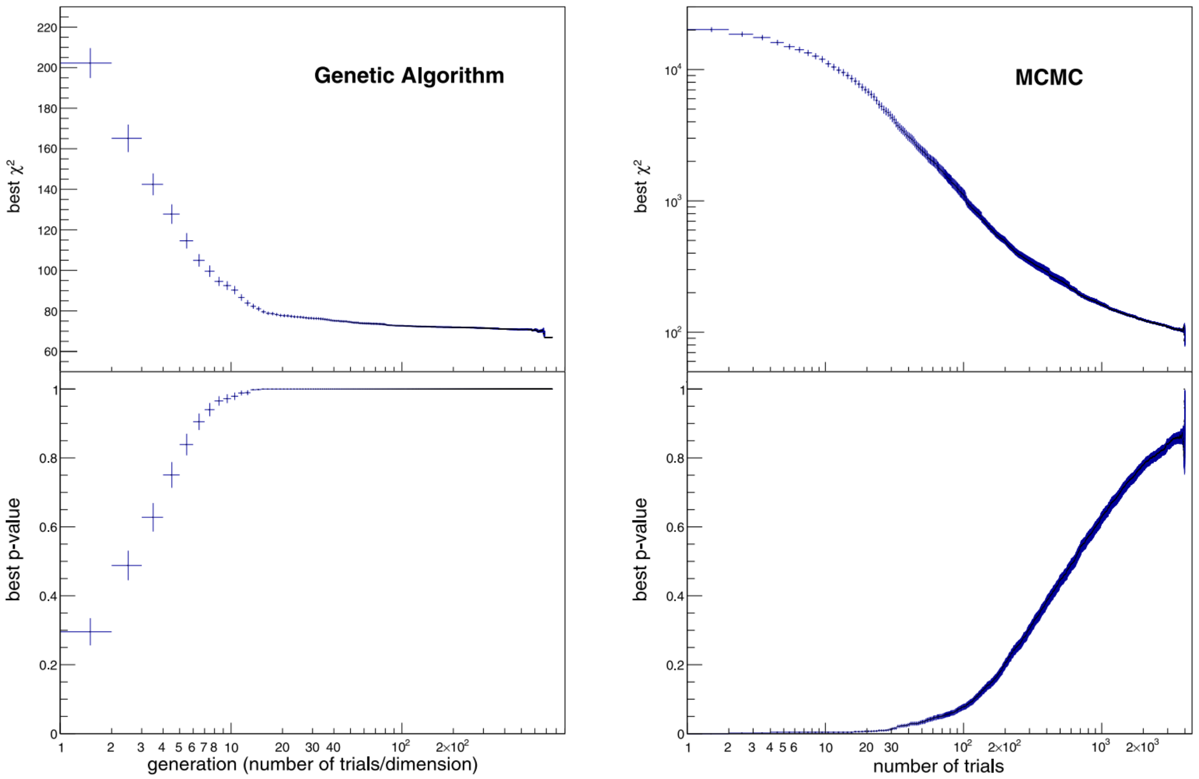

Figure 1. **Left figure:** Plot of the χ^2 & p-value as a function of generation for the GA algorithm (with 1 generation corresponding to $n = 8$ number of trials, where n is the number of dimensions). **Right figure:** Plot of the χ^2 & p-value as a function of trials for the MCMC algorithm. (Fig. 5 & Fig. 6 of reviewed article)

References

1. Glen Cowan, *"Goodness of fit and Wilks' theorem"*, Lecture notes (2013), National Institute for Nuclear Physics, Italy.
2. Jie Liu, Qiang Yuan, Xiao-Jun Bi, Hong Li, and Xinmin Zhang, *"Cosmic ray Monte-Carlo: A global fitting method in studying the properties of the new sources of cosmic $e^{+/-}$ excesses"*, Physical Review D **85** (2012), no. 4.
3. Kevin Richard Williams, *"Applications of Genetic Algorithms to a Variety of Problems in Physics and Astronomy"*, Master's thesis (2005), University of Tennessee.