Program Summary: Tracking Magnetic Monopoles Through the Galaxy

Felix Feist

GOALS / TIMELINE

due: approx.. april 24 – 8 months from today.

goal for 9/26: finish runge-kutta alg, PLUS an euler to test their consistency.

PAPER TO DO:

-\*discussion of mass and other chosen properties

-\*discussion of error sources and bounds. sources:

-sigfigs, python numerical errors

-GMF model

-mathematical approximations, including nonrelativistic

-\*add: assumes r is never precisely 0 because, well, it makes infinities. and the xfield is weird.

-\*figure numbers, equation numbers

-\*make mathematica things in word

-\*change image of diagram to computer image (photoshop?)

-\*read through as if unaware of anything and check for clarity (but also eliminate over-explaining)

-\*add healpix section with diagrams

-\*make all hats bold

-\*add section about nonreal gamma and v>c for too small timestep

-error: we got divide by zero for r-rp=0, even when z=/= 0 because of python rounding errors. check for the possibility of other such errors.

-explain as if to an undergrad, but then at end move the more obvious to appendix

-redo eqn for disk field for 3<r<5, making it just phihat, and explaining later what that is.

-read about radiation and whether or not it’s relevant to the analysis

OTHER TO DO:

-check if I’m running extraneous processes on ria

-check formulas in word doc

-run for negative charge (antimonopole), 2 charge, 1/3 or 2/3 charge, different masses, etc

-relativistic version: beware of not using copyto for acceleration etc, acc might change when you change vel or something

-error: we got divide by zero for r-rp=0, even when z=/= 0 because of python rounding errors. check for the possibility of other such errors.

-change units to cgs Gaussian

-consider simple situations with predictable results and test program output on those.

-asyncpy parallel computing

-read about parity and CP violation, they are fundamental to nature!

-rounding error must be checked. see Wikipedia for “machine epsilon”

-optimization: if necessary to optimize, have a “smart check” for the disk regions: only check adjacent regions to the one which I was in previously. Note that a large timestep (or changing timestep) can mess this up.

-check this: pos / vel goes to NaN if you start at x=-19.99kpc, y=z=0, and v0=(ce-66, 1, 1

-too-big timestep makes it go to NaN. Adaptive timestep which includes a check if vel + velstep makes a speed > c could be implemented. Halve the timestep each time this occurs.

goals:

I. map all of the possible conditions of the earth-hitting monopoles to their positions and velocities on the 20kpc sphere.

1. Is the magnetic field even important? Does it change their velocities significantly at all? AKA do we even need this computational business?

-check for global vars in a local function problem http://stackoverflow.com/questions/10851906/python-3-unboundlocalerror-local-variable-referenced-before-assignment

-if computation is necessary: numerical recipes book.

2. find errors. stepsize error, computer rounding error. human error in building the program – test for cases in which you know something about the outcome.

3. check for other monopole charges

II. find out if there is a decent chance of having monopoles bound in the galaxy

III. Making an actual paper stuff

Bring up w Farrar, or other questions:

**-What is the intention in making a plot of where monopoles came from? What is it going to help?**

**-Even with the upper bound of mass, the final velocity ends up as 0.3c just on one nonrelativistic run. Seems I should, in general, run the relativistic version**

**-are we sure about using 16.7 for r\_s, not the mean value? Refer to plot in paper, pg 7**

**-statistical analysis: what kind of error are we looking at due to our uncertainty in the bfield?**

**-Newq:**

**-Wick[6]: “We emphasize…relativistic.” Let me make sure I understand their argument: If monopoles pass through extragalactic sheets, the regions with largest energy changes, they will gain about 1e14GeV in a coherence length. Monopoles are presumed to have gone through these sheets at least once (is that because they are presumed to have been created in the early universe during a phase transition?), and so their energies are presumed to be of that order.**

**-equation 9, where does it come from, what’s the deal?**

**-what portion of a galaxy is empty space? I ask to know whether collisions would be relevant.**

**-why are flux units 1/(cm^2 s sr)? why is the sr in there too (I know the definition of the sr, that is not the issue)**

**-GZK cutoff not applicable for monopoles? (see notes, wick paper). somewhat of a comment, not a question.**

**-equation 2.1… it says below what those two symbols are but I don’t know enough to say from that what their orders of magnitude would be and thus what the mass bound of equation 2.4 is**

**-wondering: is it worth it to find the potential of the magnetic field? I was going to do so to find out which areas have a negative difference in potential great enough that a relativistic monopole of some mass entering the galactic sphere would be slowed to nonrelativistic speeds before hitting the earth. It would be better than just running the program for different directions, I would actually have a full answer as to which parts of the galactic sphere have enough of a potential difference. I could then also run the program to see which parts of the area on the galactic sphere would actually result in earth collisions. BUT if I did find that the difference in potential was NOT great enough to change the velocity of a particle by a significant amount, the paths would be nearly straight lines. Or would they be? Just because the energy difference wouldn’t be much, does that mean anything about the *direction* of the particles as they hit the earth? What I am trying to find out is: are all of the particles just going to go in a straight line anyway? (this would make our analysis kinda pointless).**

**-what about the magnetic field for rho <1 kpc? Since the black hole is there I would think that there are some extremely strong magnetic fields that we couldn’t ignore in that region. Is there an analysis of that region?**

**Introduction**

The program takes input parameters and initial conditions, and tracks a particle backward in time, tracing out the path it took through the galaxy for a specified distance. It then plots the path in 3D along with a yellow dot where the sun is, and a blue dot at the center of the galaxy. The 3D plot is interactive and can be rotated by dragging. Below the plot of the particle trail is a plot of its kinetic energy as a function of time. The program works for particles with arbitrary properties, but by default it has theoretically likely properties of a magnetic monopole. The galactic magnetic field is given by Jansson, Farrar (2012).

The input parameters are magnetic charge, mass, initial position and velocity, timestep, and distance to track the particle. The program uses base units of distance in kpc, time in s, current in A, and mass in kilograms. All other units are built from these. In other words, all units are SI units except those with factors of distance. The kinetic energy, however, is plotted in GeV.

A few assumptions were made in designing this program:

1. There are no collisions – space is empty. While this is not true, it is somewhat unlikely for a particle to collide with an object, and if it did, its path from that point on would be uninteresting. It is also worth noting that realistically, the only collisions that could take place are collisions with objects which act as a source of magnetic monopoles, because the particle was tracked *backwards* in time.

2. The magnetic monopoles have no electric charge.

3. All forces are considered negligible except for the magnetic force.

4. The particle cannot decay during its travel.

In addition, some mathematical assumptions were made in the theory, in order to make the program possible. Most, if not all of these assumptions rely on a small timestep. To test whether the chosen timestep is small enough that the assumptions make negligible difference, divide the timestep by some factor and see if the final position is significantly different than it was with the initial timestep.

The mass of the default particle is chosen to be the lower mass bound of a magnetic monopole from Wick (2002) – 40TeV/c2.

The charge is one elementary unit of magnetic charge, given by the dirac quantization condition. This is further described in the Theory section.

By default, the particle starts at the position of the earth. The galactic center is not plotted by default because if the particle is only tracked a short distance, showing both the center and the trail would require the window to be so large that the trail cannot be seen in detail.

Because of the limitations on numbers of significant digits in python, the Lorentz factor is precisely equal to 1 until . In this case the kinetic energy is calculated using .

The program currently uses a constant timestep. But if one were to vary the timestep so that when the velocity is great, the timestep gets smaller, it would prevent large changes in distance in a single step and increase accuracy.

**Theory**

**I. Monopole Charge, Mass, Initial Velocity**

Confusingly, the Dirac quantization condition depends whether the SI units you are using are using the weber convention of the ampere\*meter convention.

The dirac quantization condition, with magnetic charge in units of Ampere meters, gives (from wikipedia for magnetic monopole):

Mathematica gives 3.29106e-9 Ampere meters ([link](http://www.wolframalpha.com/input/?i=h%2F(mu_0+*+q_e)&rawformassumption=%22UnitClash%22+-%3E+%7B%22h%22,+%7B%22PlanckConstant%22%7D%7D))

Here, qb is the elementary magnetic charge. Converting to Ampere kiloparsecs yields ([link](http://www.wolframalpha.com/input/?i=3.29106e-9+m+to+kpc)):

(we use 8 digits, the accuracy to which is known)

In Gaussian units, the quantization condition, from wikipedia, is

Or,

As for the units, I get:

(based on wikipedia’s definition of the erg and statC)

2/8/16 – confirmed, this agrees with Terry Sloan, too. He has the gaussian version up. Interesting that they are numerically different... I do not understand why.

Wikipedia says the A\*m equation ...

GAH! I asked on stack exchange the diff btw Am and Weber.

Initial velocity:

Wick (2002) estimates the energy a monopole has picked up since its creation as ~1e14GeV. I don’t know about the accuracy of his coherence length or field strength and the justification of the number of coherence lengths traversed, but the rest is reasonable. Should ask Farrar about it.

**II. Relativistic Particle Dynamics**

For comparison, here is the non-relativistic derivation, in one dimension, with the assumption that force is constant over the interval :

To track the particle backwards, we just reverse the sign of the equations for and , which are the only equations appearing in the program itself.

We now consider the relativistic case. We use 3-vectors, as coordinate transformations are not necessary. All quantities are classical (so, for example, ).

The force on a magnetic monopole in a magnetic field is:

I assume that this holds true, even relativistically, because of the way the Coulomb law was used in Griffiths (pg 524). However, here, F is the “classical” force, , as opposed to the nicely-transforming .

Thus we have

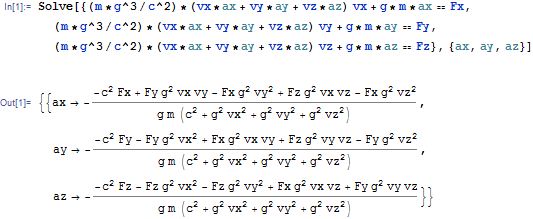
Newton’s second law in the form does not hold. Instead, (still with classical variables and three-vectors) we use

With , its derivative is:

Plugging this into the force yields:

The above equation reduces to if .

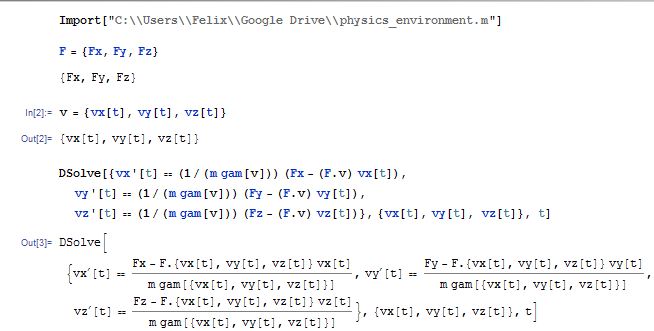
The equations for the three components of yield a system of three equations and three unknowns ( and ). Mathematica solves for the three accelerations:



dividing by γ^2 and expanding out the definition of γ:

**C:\Users\Felix\Downloads\CodeCogsEqn (15).gif**

It seems that this system of differential equations is not solvable; mathematica knows no solution, even with a constant force.



We now have solved for , and integrating component-by-component yields , the change in velocity over the time interval . In order to do so, we make the assumption that our timestep is small enough that the 6 dependent variables of **a**, namely **F** and **v**, are constant over the interval, and thus the acceleration is as well. We do so with the x-component below:

With the above assumption, we integrate, and solve for

We then integrate again to find Δx:

In our program, then, we repeat the following process:

1. Calculate the force

2. Using the force and initial velocity, calculate the acceleration

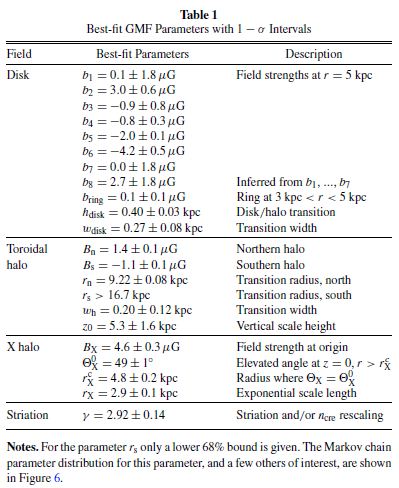
3. Using the above expressions for and , update the positions and velocities to their new values at

**III. The magnetic field from Jansson, Farrar (2012): Summary and Calculations**

Using Faraday rotation measures, it was possible to extrapolate an approximate value of the magnetic field throughout the galaxy. The field everywhere is a sum of three functions: The disk component. the toroidial halo component, and the out-of-plane component (referred to as the X field because of its shape). The paper mentions a fourth component (the striated field), which was not used in this program. The different components are defined for different regions. Where the regions overlap, the field is a sum of the components.

This paper uses right-handed coordinate systems. They are centered at the galactic center. In rectangular coordinates, the sun is at x=-8.5kpc, and the z axis is perpendicular to the galactic plane. In cylindrical, r is the distance from the galactic center axis, and z is the same as in rectangular coordinates. The paper uses a convention in which the cylindrical coordinate extends from the negative x-axis, but this program adopts a convention in which extends from the positive x-axis. Thus, we replace all instances of in the paper by . From this point, our convention for the angle will be written , and the paper’s convention will be written .

The magnetic field depends on a variety of best-fit parameters. The table of parameters from the paper is below:



In a sphere of radius kpc, and outside the cylinder with , the field is defined to be 0. Below is a summary of its components in the nonzero region.

**IIIa. The Disk Component:**

The disk field is 0 for .

For , the disk field is azimuthal and takes the form

C:\Users\Felix\Downloads\CodeCogsEqn (5).gif

is a function which transitions the disk field into the halo field. The halo field is multiplied by , and the disk field is multiplied by . Its definition is

When , the disk component is a piecewise function, dependent on the “region” in the galaxy wherein the function is evaluated. The regions are defined as follows:

8 logarithmic spirals on the x-y plane form boundaries (thus creating 8 regions). The equations of the spirals take the form NO THEY DONT

Here, is the opening angle of the logarithmic spiral (it is the same for each spiral).

\*(make this a bottom comment) Note: In the paper it’s:

But that is an error.

is the position at which the spiral crosses the negative x-axis. The values it takes on are 5.1, 6.3, 7.1, 8.3, 9.8, 11.4, 12.7, and 15.5, in kpc. Regions are numbered such that Region 1 is between the functions with and , region 2 is between and , and regions 3-8 follow.

The disk component of the magnetic field in region j is

C:\Users\Felix\Downloads\CodeCogsEqn (4).gif

C:\Users\Rick\Downloads\CodeCogsEqn.gif

Where the direction is:

C:\Users\Rick\Downloads\CodeCogsEqn (1).gif

C:\Users\Rick\Downloads\CodeCogsEqn (4).gif

C:\Users\Rick\Downloads\CodeCogsEqn (2).gif

C:\Users\Rick\Downloads\CodeCogsEqn (3).gif

is a constant parameter which takes on one of the 8 values listed in the table at the beginning of section III.

In rectangular coordinates, the disk component in region j is



The disk field is nonzero in the region with . In this region it is azimuthal and takes the form

C:\Users\Felix\Downloads\CodeCogsEqn (5).gif

where

C:\Users\Felix\Downloads\CodeCogsEqn (12).gif

**IIIb. The Toroidial Halo Component**

The toroidial halo component is a simple piecewise function:

C:\Users\Felix\Downloads\CodeCogsEqn (6).gif

It is defined to be 0 for r = 0, where its direction is not well-defined.

**IIIc. The X-Field Component**

The X-field has no azimuthal component, giving it the structure for which it was named:

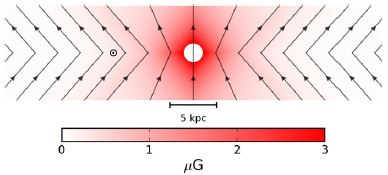


Diagram above from Jansson, Farrar (2012). This is the xz cross section, so and it includes the sun at x=-8.5kpc, but the field lines the same for any .

The field is given as a function of , the radius at which a field line hits the x-y plane. Before , the field lines are variable in slope with elevation angle . Afterwards, the slope remains constant at a value of .

Thus, in the constant elevation region,

and in the varying elevation region,

The respective fields are

C:\Users\Felix\Downloads\CodeCogsEqn (6).gif

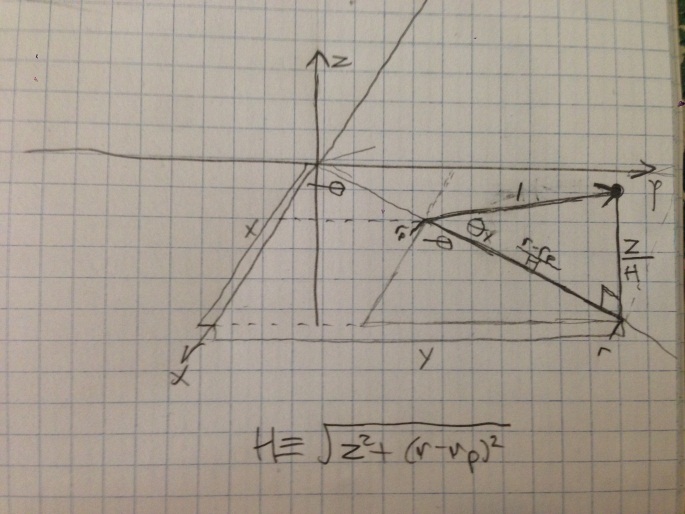
where C:\Users\Felix\Downloads\CodeCogsEqn (4).gif is a unit vector defined by the field direction in the X-field image. (give figure number), which, for , comes out to

C:\Users\Felix\Downloads\CodeCogsEqn.gif

C:\Users\Felix\Downloads\CodeCogsEqn (5).gif

For , C:\Users\Felix\Downloads\CodeCogsEqn (4).gif is straight upward in z. This was not in the paper, but was chosen because a similar and slightly better-fit xfield function has a bfield straight upward in z. For r=0, C:\Users\Felix\Downloads\CodeCogsEqn (4).gif normally would reach a singularity but is defined as C:\Users\Felix\Downloads\CodeCogsEqn (10).gif for z>0 and -C:\Users\Felix\Downloads\CodeCogsEqn (10).gif for z < 0. Note that for r=0, where , is still well-defined (just divide the formula for rp by r).

eq# (above) can be derived from the diagram below, where .



The program checks which equation to use for by checking the inequality

The inequality divides the two regions: if true, use the varying elevation region.

This concludes all three components of the magnetic field.

**Method of ODE Solving**

We use a 5th order Runge-Kutta with an adaptive timestep method and coefficients suggested by Cash and Karp, taken from Numerical Recipes in C (page 740).

The general procedure:

dt = guess step

//startloop

allowedDelta[] = maximum allowed error per step allowed in each component of **r** and **v**, estimated beforehand

Delta[] = actual errors in each of the components after taking the guess step dt. //this is estimated by doing rungekutta5 – rungekutta4 = estimateDelta.

if (any of the actual errors (Delta) are greater than the allowed errors) {

#retake the step

dt <- 0.9\*dt\*(allowedDelta[theWorstOffenderBy%]/Delta[theWorstOffenderBy%])^0.25

xnew, vnew = rk4&5(xold,vold,dt) //rk4&5 is a combination of the terms in 4 and 5 that yields 6th order.

else {

//set dt in preparation for next step:

dt = 0.9\*dt\*(allowedDelta[theWorstOffenderBy%]/Delta[theWorstOffenderBy%])^0.2

}

…repeat the process until finished.

Let be the average error per step in the algorithm – this could be in a position or velocity value, and be the total allowed error at the end of the algorithm, where each can be negative. Then by definition,

It remains to find and N.

We find a typical N by :

1. Run the Euler method a few times with a tiny timestep to approximate a perfect accuracy. We will know that our timestep is small enough when a 5x change in timestep does not change our result significantly.

2. Run the Euler method on the same trajectories with larger timesteps until an error of size is achieved.

3. The N for the run which achieves an error is multiplied by 1.5 and used as a threshold value. If a run’s N exceeds this value, the run is re-done with a doubled N, and all nearby points in phase space are assumed to require a doubled N as well.

As for , it is a choice of ours: what is the maximum allowed error in position and in velocity? With what resolution do we need to see the map of bigsphere conditions corresponding to littlesphere collisions? First consider spacial conditions, under a particular velocity. The flux would be calculated as the number of particles hitting the bigsphere for that velocity, divided by the

In a distance over which the magnetic field does not change much, we will get similar results. So we only need our resolution to be less than the distance over which the magnetic field changes significantly. What is significantly?

But what should we choose as E, the maximal allowed error in the velocity? The problem really highlights a problem with the backtracking approach: we want to be able to make the assumption that the incoming velocity of monopoles is a particular number, and find the percent of the initial phase space with that initial velocity which hits the earth. But with backtracking, we give a velocity to start with and end up with a finishing (or starting – in either case, earlier in time) velocity which is not of our choosing. It doesn’t suit the problem at hand. So for now, I will choose an arbitrary E – that is the lesser of one one-hundredth the speed of light or of the initial velocity.

Resolution: The problem was if many bigsphere points correspond to few littlesphere points. Let’s say many correspond to just one littlesphere point for argument. Physically this is impossible but with a coarse enough resolution of littlesphere conditions, it could be that adjacent points in littlesphere space correspond to far-away points in bigsphere space, so that many bigsphere points are “missed”, or not marked as earthbound points.

This is not a problem as long as I am careful with my choice of initial (littlesphere) conditions. I start with one point in the (bounded) littlesphere phase space and find its corresponding bigsphere point. I repeat for a nearby littlesphere point. If the 2nd bigsphere point is too far away\* (beyond some tolerance) from the 1st bigsphere point, I try another condition in between the two points to get within the tolerance of my 1st point. For each bigsphere point I fill in, I “paint red” a sphere of bigsphere points; these “red” points are the ones within my bigsphere tolerance. By shooting more and more particles, I fill in more and more of the “bigspace” as red. In truth, only some fraction of the bigspace is within my capability to paint; the points I am capable of painting are the ones whose trajectories can hit the littlesphere. My goal, then, is to paint every point in bigspace that can be painted red, red.

\*But how do I define this distance? I could try to come up with some kind of definition of a sphere in phase space. But this would be more complicated and less useful than just drawing a cube around each point in space (equivalently, painting all bigspace points which are within a certain distance of *each coordinate* red, where coordinates are θ, φ, vx, vy, vz. ). Cubes would be more useful than spheres because they pack better in ideal conditions.

I am likely to end up with unpainted holes in my bigsphere space which are surrounded by red space. I have two reasonable alternatives to fill the unpainted space:

1.Within a certain tolerance, I can just color these holes red, because it is unlikely that small punctures in red bigspace (“redspace”) are not also in redspace. I can call these points pink points (in between red and white). I must be careful in doing this, but even if I color some punctures incorrectly, it should not have a big effect on the result of the paper (the flux) as long as pinkspace has a much smaller volume than redspace, which is an easy requirement to satisfy. I can just choose my tolerance when coloring space pink to make this happen.

2. I can use the smallspace points corresponding to nearby, filled points to guess other smallspace points which are likely to fill in the bigspace gaps. This is more costly in computing time but eliminates a source of error.

Next, the proposed problem of Liouville’s theorem. I will answer it in message format to copypasta it to Farrar.

From what I understand, the proposed problem due to Liouville’s theorem is that it implies that in any physical scenario, the flux of particles into the bigsphere (r=20kpc about the galactic center) must be equal to the flux of particles into the littlesphere (the earth). This would mean that the existence or nonexistence of the magnetic field would not affect the flux of mpoles into the littlesphere.

I do not understand Liouville’s theorem very well, but I believe I have a counterargument. If this really applies to every physical scenario, it would also apply to a scenario in which we tracked positive electrically charged particles through the same scenario, but with a divergent electric field like the field of a point charge. I propose that in this scenario, the flux of particles into an earth-size sphere surrounding the divergent particle would be less than the flux into the r=20kpc bigsphere. My rationale is:

Compare two scenarios: the divergent field exists or it doesn’t.

Statement 1: All particles which would not hit the littlesphere if no field existed, would also not hit the littlesphere if a field did exist, because their trajectories would only diverge from the littlesphere. (they are positive particles in a divergent field)

Statement 2: There exist particles which would hit the littlesphere in the no-field scenario, but would not hit the littlesphere if a divergent field were present. Consider a particle which, in the no-field scenario, only barely hits the outer volume of the littlesphere and is not aimed toward the center of the littlesphere. In the divergent field scenario, the path of this particle would be perturbed away from the littlesphere, so that no collision would occur.

Thus, the flux of particles into the littlesphere would be *smaller* in the scenario with a divergent field, than if there were no field. And so it is clear that scenarios exist in which the field affects flux values. So Liouville’s Theorem cannot imply that universally, fields do not have an effect on flux. And so, the GMF may have an effect on earthbound monopole flux.

**Notes**

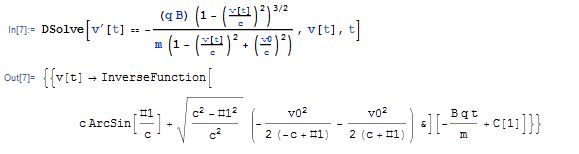
I. Future improvement: The biggest fault in the program is the time it takes to process. The time varies greatly with the initial velocity, because accelerations are small. Tracking the particle for a distance of order 1kpc would take days. To improve the program, I will continue seeking ways to better optimize the processing time. In addition, there may be a more accurate way to track the particle by making a higher-order approximation than just a constant velocity to calculate the acceleration. It is possible that such an approximation would take more time to calculate than just changing the timestep by an amount that adds the same accuracy, but it is still worth comparing the two.

II. Personal note: In retrospect, the theory doesn’t seem bad at all (other than somewhat lengthy expressions), but having been pretty unfamiliar with relativity (we learned about length contraction and time dilation in Physics I but that’s about it), I made a lot of errors and it took a lot more time than I had anticipated. I also spent a lot of time fixing bugs and making small adjustments.

**Questions**

1. On the line x=0,y=0 the magnetic field reduces to (BX is a parameter from the model). So the force is entirely in the z direction for a positive charge, and the equation for the z-acceleration reduces to (with vx, vy =0). Let the starting velocity in the z direction simply be and the z-acceleration be a.

Which has the solution:

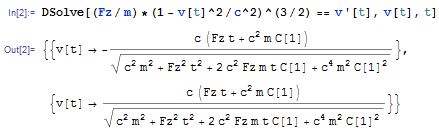


(not good!)

(incorrect solution: algebra error, below)

I had

Which has the solution:

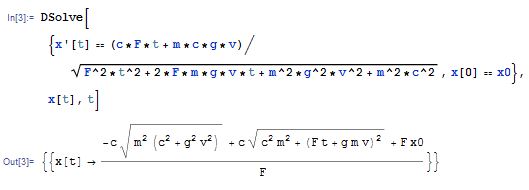


With the initial condition we get that

This yields

Simplifying this yields the condition that the from must be a +. We assume that a positive force in z must yield a positive velocity (if is positive as well), and from there we deduce that the in the differential equation solution must also be a +. Thus we have:

Solving this for x(t) with the condition x(0)=x0, we get



Sig figs reasoning:

in kpc/s, c9.715611713408621\*10-12 . To identify the number of sig-figs necessary to keep, we found the limiting velocity at which python considers (v/c)^2 to be 1, or γ=infinity. This happens soon after v=9.7156…0862 \*10-12. So we let c=9.7156…08621, a value of v at which γ=inf. Any more sig figs are redundant. (sum this bit up in a smaller amount). (question this – we are inaccurate in velocity so I don’t think we need that many)

-Low mass (1000(?)TeV) leads to almost immediate relativistic velocities, with accelerations of ~300m/s/s.

**Masses:**

>~ 10^4GeV: avoids standard model violations (wick 2002, pg3)

~10^5GeV: 3-family non-susy models

~10^7GeV: extra dimensions a la kaluza-klein, maybe(?) compact, like ~1mm leads to this mass.

~10^8GeV: exists in SU(15)

<10^11GeV: a comparison of the kibble flux to the parker limit (Wick 2003,pg666 RHS)

< 10^13GeV: from observations of curvature of universe(wick 2002, pg4 bottom)

<~ 10^15GeV: lack of proton decay mandates this

~10^17GeV: minimal SU(5) breaking

in short: , with some more likely spots in between

Grand unification magnetic monopoles have masses of the order 1e15GeV

src: http://moedal.web.cern.ch/content/search-magnetic-monopole

**Monopole Density**

From sloan’s paper on finding particles, a suggested density based on the kibble mechanism is “one monopole per (100km)­3

Conclusions:

\*\*THE BELOW DISCUSSION IS WRONG BECAUSE I HAD AN EXTRA SQRT IN MY ACC\_RELATIVISTIC FUNCTION\*\*

Through the plot, you can see that the points with the greatest Bfield strength. Run a monopole through these points. The trajectory will be altered more than any other trajectory. If the trajectory is entirely not bent, then you can be confident that monopoles at this velocity would not have their earthbound flux altered. Another requirement here is that the field is relatively monodirectional throughout the trajectory; this can be seen by the vector field plot.

I ran the program with:

r0\_default = np.array([-19.8, 1.5, -0.01])

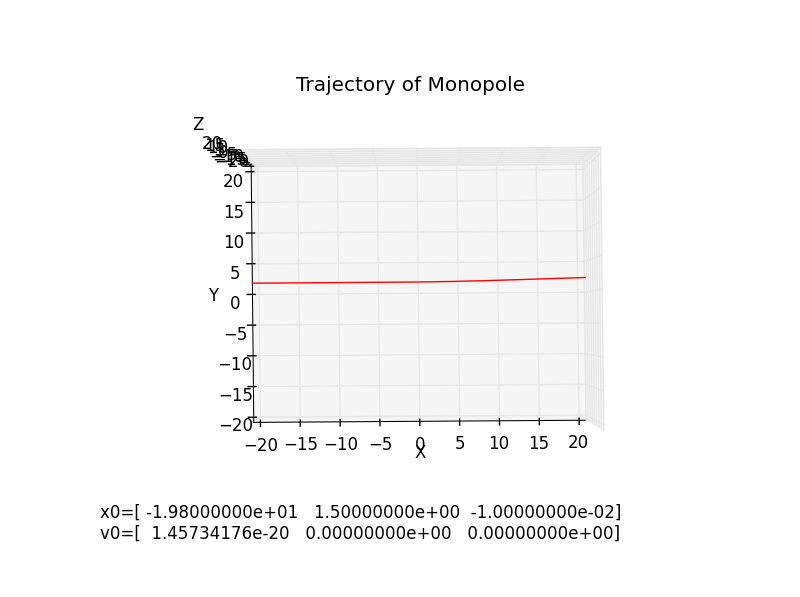
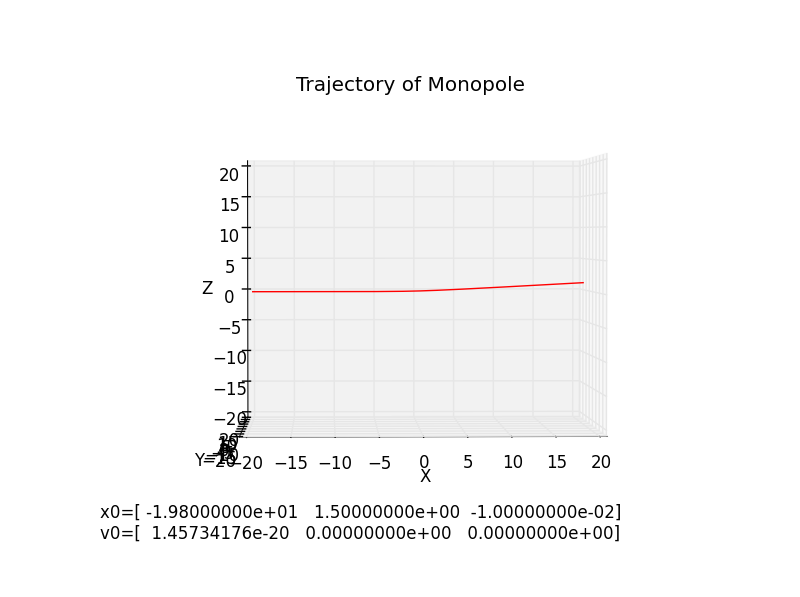
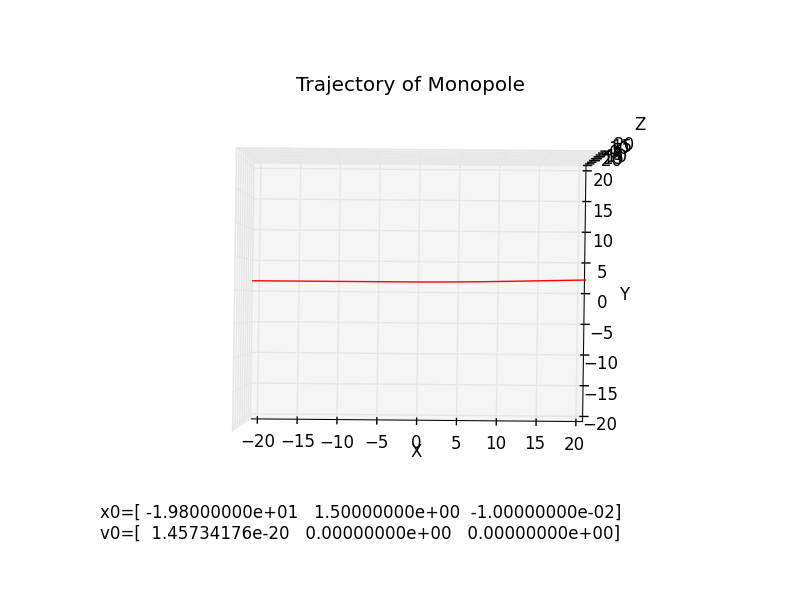
v0\_default = np.array([1.5e-9\*c,0,0])

dt\_defult = 1.0e18

m=1e13GeV

This resulted in 2726 iterations, enough to form a smooth trajectory. Pictures of the slightly bent trajectory (titled trajectory1-xz.png or similar and shown below) show that the highest velocity at which the trajectories are still somewhat bent is around 1.5e-9\*c = 0.45 m/s. So for velocities greater than this, the Bfield is unlikely to have any effect on the flux.

For theoretical derivations of this result, it’s relevant that over 1e6 random points, the smallest and largest Bfield in Tesla were 4.84e-13 and 3.47e-10.



Screenshots at v=0.45m/s. Lower than this, the trajectories start to bend significantly. Decreasing the velocities yields the following outcomes:

INITIAL CONDITIONS:

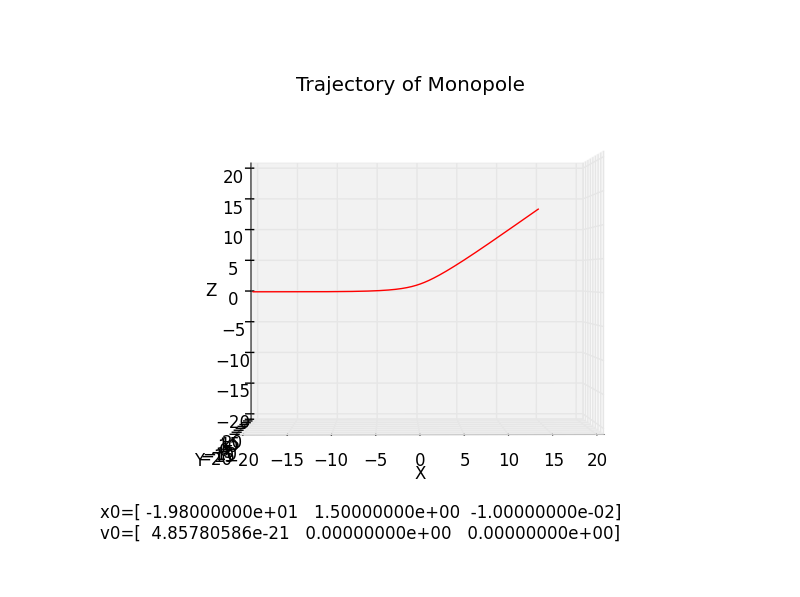
r0\_default = np.array([-19.8, 1.5, -0.01])

v0\_default = np.array([5e-10\*c,0,0])

dt\_default = 5.0e18

m=1e13GeV

graphs below.



(screenshot is from an interactive display and was taken at the most bent angle)

initial velocity of v=0.15m/s

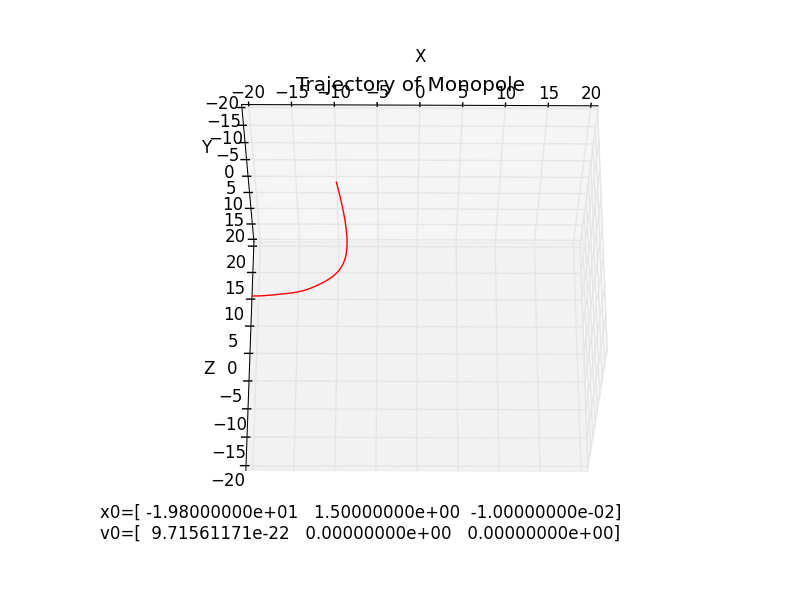
r0\_default = np.array([-19.8, 1.5, -0.01])

v0\_default = np.array([1e-10\*c,0,0])

dt\_default = 5.0e18

m=1e13GeV

graphs below.



4498 iterations. large bending. v0 = 0.03m/s

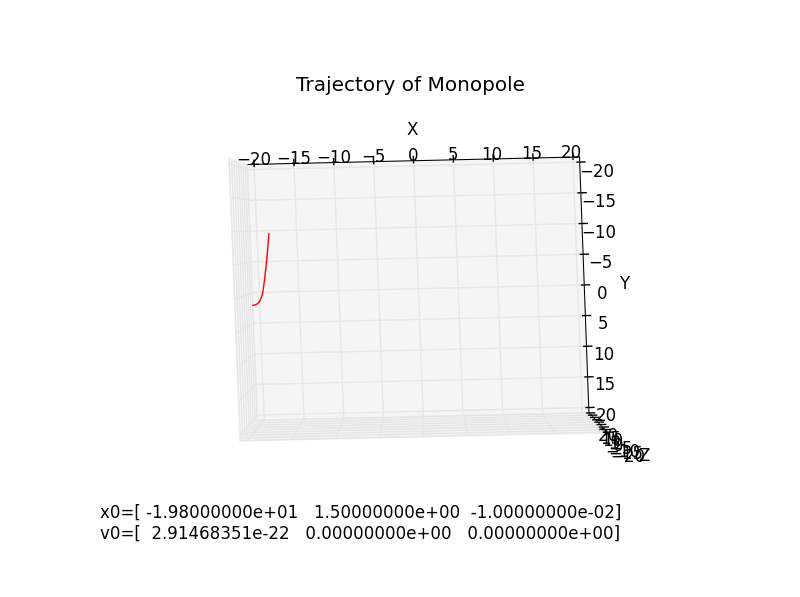
r0\_default = np.array([-19.8, 1.5, -0.01])

v0\_default = np.array([3e-11\*c,0,0])

dt\_default = 5.0e18

m=1e13GeV

graphs below.



v0 = 0.009 m/s ~= 1e-2 m/s

3538 iterations before exiting the galaxy.

The bending occurred for velocities between 1e-2 m/s and 0.45 = 4.5e-1 m/s. With a mass of 1e15GeV, the kinetic energies of these velocities correspond to kinetic energies of

to

With a monopole mass of 1e13GeV, this is tiny.

Above 2.5MeV, trajectories must not be bent, as they were not bent by the regions with the strongest field.

Below 56keV, the earth is shielded from monopoles from the upper hemisphere of the galaxy, because of the outward flux of field lines. The direction of the field in the lower hemisphere is generally toward the earth, so no conclusions can be made.

Thus, there is certainly shielding of monopoles with energies below 2.5MeV, which reaches at least 50% shielding at energies below 56keV because an entire hemisphere is blocked\*

\*56keV is not an appropriate number. The upper hemisphere has to be checked for the region with the lowest field strength. It was tested in the region with the strongest field strength.

\*\*THE ABOVE DISCUSSION IS WRONG BECAUSE I HAD AN EXTRA SQRT IN MY ACC\_RELATIVISTIC FUNCTION\*\*

**Theoretical Bounds for when monopoles can enter:**

Below I try to theoretically estimate the kinetic energy required to tunnel through the magnetic field and reach the Earth, in the upper hemisphere of the field where it opposes entry. Relevant to this is an estimate by Wick (2002) that the kinetic energy of a monopole which has existed since the beginning of the universe is of order 1e13GeV, independent of its mass.

**Theoretically, at which energies can the monopole never tunnel all the way to Earth?**

AKA: At which energies does the monopole get repelled before tunneling to r=8.5kpc?

The boundary is when just as much radial kinetic energy is imparted by the field as the monopole had to begin with. If kinetic energy is T, then:

q=3.3e-9 A\*m is the charge.

Δx is the distance that a particle would have to penetrate into the milky way in order to reach the earth, namely 11.5kpc = 3.5e20m.

B is 3.8e-12T. This is a typical value of the radially outward component of B which a particle would experience if it chose one of the “easiest” paths to penetrate toward the earth. In other words, the region had a relatively weak field. The value was found by sampling a point from the xfield in the weak region, in particular at (-15.7, 0, 2.6). This point was located visually by looking at plots of |B|. The xfield was chosen as it is the only component of the field which pushes particles radially outward from the earth.

Note: It was checked that the special relativistic versions of the work-energy theorem and Coulomb’s law in the form F=qE are the same as the classical versions. Evidence in Serway chapter2 (in papers folder) and Griffiths E&M, page 438-439.

The conclusion from the work above: if the Wick estimate is even somewhat correct, monopoles will have no difficulty entering the galaxy, and their flux will not be affected by the presence of the field whatsoever.

**Question 2: Okay, so monopoles can make it in at just the right angle. At what point will a monopole *reliably* be able to penetrate the upper hemisphere?**

Repeating the above calculation with a slightly above-average x-field strength (1e-10T, ~1 stdev above average) and choosing Δx as a relatively long tunneling distance (22kpc), the required energy to reach earth is:

It is clear from this that any monopole with a kinetic energy as described by Wick would have no difficulty penetrating the Galaxy (whether they are relativistic or not), but as monopoles would have deviations in energy and the magnitudes are only off by 1 order, there are likely to be small percentages of monopoles which would be deflected.

**Now let’s do it computationally (question 1):**

The following mass & speeds barely made it:

mass, GeV/c2 speed , Joules

(entered) (tested for, (calculated from m, v)

computationally)

1e9 (\*\*\*) 7.8

1e13 (1e-1)\*c 8.0

1e15 (1e-2)\*c 8.0

\*\*\* the computer could not handle this many 9s and exploded. In particular, at speeds this close to c, the acceleration easily pushes v > c unless the timestep is very small. So it takes a long time to handle ultrarelativistic velocities.

Now for the case where monopoles are reliably able to tunnel to the earth. This happens approximately when there is no deflection at all throughout the trajectory. I will computationally shoot monopoles into the regions of heaviest field and look at the deflection. Monopoles enter through the upper hemisphere, where experience radial pushback.

**Replicating Sloan:**

Sloan uses a starting velocity . With this velocity, he should find that the field repels monopoles of masses lower than (in SI units):

His paper lists the boundary at which the mpole is repelled as approximately 1017GeV/c2, consistent with the theory. The kinetic energy this corresponds to is , consistent with computational results (sloan did not check every angle it seems).

**Order of Magnitude Consistency Check of**

min: 0

typical:

But typical should actually be smaller since field doesn’t always push the right direction so we could easily lose a factor of ~4:

(calc: [http://www.wolframalpha.com/input/?i=(3.3e-9+A\*m)\*(7.0e-11+T)\*(10kpc)+to+GeV](http://www.wolframalpha.com/input/?i=(3.3e-9+A*m)*(7.0e-11+T)*(10kpc)+to+GeV) )

max:

but really no path should be this high, there is no coherent path with this amplitude of |B|.

(calc: [http://www.wolframalpha.com/input/?i=(3.3e-9+A\*m)\*(3.5e-10+T)\*(20kpc)+to+GeV](http://www.wolframalpha.com/input/?i=(3.3e-9+A*m)*(3.5e-10+T)*(20kpc)+to+GeV) )

So for all runs.

**TESTING FIBONNACI\_SPHERE FUNCTION**

Does fibfunc generate a good, even distribution of points?

Is it better with randomize = True?

How do results vary with number of samples?

Is the algorithm stable (will it sometimes produce extremely bad results?)

Programmatically I did the following:

1. For every sample size in [5,100], generate 20 random samples and one nonrandom sample.

2. For the nonrandom sample, record the average distance to a point’s nearest neighbor. Also record the min and max nearest neighbor distance (NND), and the stdev of NNDs divided by their mean.

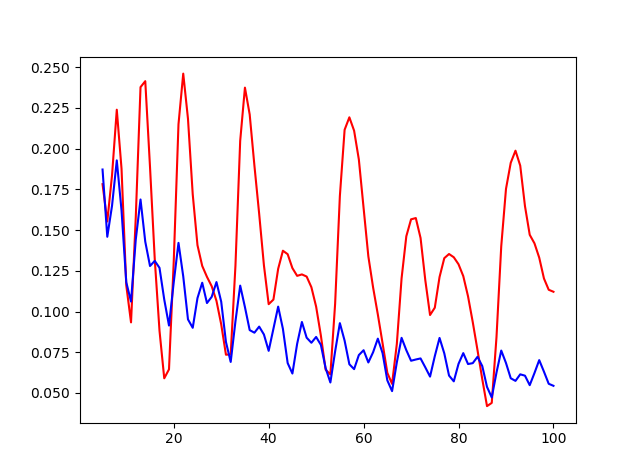
3. For the random sample, record the average of these values over the 20 random samples.

4. “Normalize” these values so that they don’t depend on the number of samples, by dividing by sqrt(SphereArea/nSamples). sqrt() was taken in order to get a length from an average area per point. Now the avg, min, and max can be compared across sample sizes.

5. Plot these results across the sample sizes [5,100].

**Nonrandom did better than random for essentially ANY sample size. While the minimum nearest neighbor distance was approximately the same for random and nonrandom, the stdev/mean of the NNDs was significantly better for nonrandom.**

The graphs that led to those conclusions are in the fibsphere tests folder. Random in Red, nonrandom in Blue. Sample size on x-axis. The most important graph is below: given a sample size on the x axis, it shows the typical spread of distances to a point’s nearest neighbors, normalized by dividing by the average nearest-neighbor distance:



Above: number of points on x-axis, stdev/mean of NNDs on y-axis. Lines been smoothed out by averaging to make it easier on the eye,