Dark Matter Halos in the Milky Way

Mikael Kardell

September 16, 2008

Abstract

In this thesis I present a generator of dark matter (DM) sub-halos inside a Milky Way-like galaxy. I use the radial density profile of a DM halo presented by Navarro, Frenk and White (the NFW profile which is $\propto r^{-1}$ when $r \to 0$ and $\propto r^{-3}$ when $r \to \infty$) combined with the statical Roche criteria for tidal forces. Theoretically sub-halos could be as small as $10^{-6} \cdot M_{\odot}$ but computational problems limit me to having a lower limit of around 10^{-3} solar masses at the smallest sub-halo.

The DM sub-halo distributions generated are visualized by calculating a sky map of the strength of annihilation radiation as seen from the Earth. These annihilation strengths are proportional to ρ^2 (where ρ is DM density), so the more lumpy the galaxy is the clearer the visible structure will be. All maps are generated as both Mollweide projections and $\theta\phi$-maps. I compare how different setups affect the sky maps, especially how scaling affects the sub-halo visibility compared to the host halo.

The sky maps are computed for discrete points forming a grid, resulting in the loss of small sub-halos that fall between evaluation points. This phenomenon is also studied with the result that you need to choose a reasonable trade-off between computational time, and missed sub-halos. The smaller the grid size is the longer the computational time becomes.

Contents

1 Introduction

It has long been known that there is something missing in the Universe. The rotation velocities of galaxies, gravitational lensing and temperature distribution of hot gas inside galaxies all tell the same same story. The Universe should contain more mass. This missing mass has come to be called "Dark Matter", DM, since it has to be electromagnetically neutral, and thus does not shine.

1.1 Galactic rotation velocities and gravitational lensing

As early as in 1933 an astronomer by the name of Fritz Zwicky measured the rotations of distant galaxy clusters, especially the Coma cluster. Using the virial theorem he computed the amount of mass in the clusters and compared it to the mass of luminous objects showing a discepancy of at least a factor of a hundred [17, 18].

It did however take until the early 1970's until there was mainstream acceptance that you needed some kind of extra matter in the Universe. Astronomers started measuring the rotation velocities of different spiral galaxies and found that, instead of having velocities that decreased at larger distances, you had an almost constant velocity out to some cut-off distance. This indicates that galaxies had much more mass than what could be observed and that it was uniformly distributed well beyond the visible disc of stars.

One effect of relativity theory is that light is bent in gravitational fields, much like light is bent in a common lens. Observations of these gravitational lensing effects, on distant stars and galaxies, imply large chunks of matter floating around in the universe. However since there are no visible stars or anything else that seems to cause the gravitational lensing dark mattes seems necessary.

Recently, the existence of DM has been verified in a number of experiments of which the WMAP satellite is one.

1.2 The supersymmetric solution

Since dark matter cannot be seen, hence the word "dark", it cannot interact electromagnetically. This means that DM-particles have to be electromagnetically neutral (charge, $q = 0$). In the standard model of particles there is only one such type of particle, and that is the family of neutrinos, ν_e, ν_μ and ν_τ. These particles could account for the missing mass in the universe. However due to the low mass of these particles, next to zero, this kind of DM would be hot, since neutrinos travel near the speed of light. Hot DM will not be able to form the small scale structures that we see in the Universe today and therefore neutrinos are not likely to be responsible for the abundance of DM in the Universe.

The specification of our desired DM particle is a charge neutral, but still heavy, particle. A class of such particles is the WIMP or "Weakly Interacting Massive Particle" which arise from the supersymmetric extension to the standard model.

One such particle is the neutralino family, χ , a supersymmetric cousin to the Z^0, γ and Higgs bosons. These particles should be quite massive, up to several hundred GeV, and only be weakly interacting[1]. Fortunately these particles are their own antiparticles and wherefore we should be able to detect the light emitted from a DM-annihilation. Of course there are lots of other objects in the galaxy: stars, clouds of gas, etc that also shine. Since the DM particles are visible only through their self-annihilation we would not expect them to outshine luminous objects like stars, however there are ways around this.

Studies made by Bergström et. al. using the DarkSusy package show what the energy spectra of a DM annihilation could look like. One way of finding the DM annihilations is to look for the sharp cutoff in energy that should be produced at an energy of m_χ. No other known source has this kind of spectrum and thus such a spectrum could serve as a fingerprint of DM. Since χ is its own anti particle you could have and annihilation process that is $\chi\chi \rightarrow \gamma\gamma$. In this reaction the γ-photons will have an energy of $E_\gamma = m_\chi$.

A possibility of direct detection could be to look for parts of the sky that only glow in the spectrum near m_χ. If you find a region of the sky that shine at $E_\gamma \approx m_\chi$ but not in optical or any other wavelength you would have strong evidence for indirect detection of DM. The reason that we can be certain that we are actually observing a DM-annihilation is that usually other stellar object shine in many different wavelengths. DM-annihilations could however have a very specific energy range and maybe even have monochromatic lines.

1.3 Via Lactea - The great simulation

Many simulations have been made to simulate the development of the known universe. Also simulations of our own galaxy has been undertaken. One of the largest and best is the Via Lactea simulation made by Diemand et.al.[3, 2, 4] A galaxy similar to our Milky Way has been tracked from the days of the cosmic microwave background, CMB, to today. In this simulation they started with a distribution of mass matching the cosmic microwave background (CMB). Since DM dominates in mass over the "common" hadronic mass in the universe, the later has been omitted and only collisionless matter has been used in the simulations. This simulations had recently been updated using the Via Lactea II simulation of which analysis is still being conducted.

The results are that DM is not smoothly distributed in the galaxy but rather quite lumpy. The lumpiness comes in all resolved mass scales and is surprisingly self similar to scales as small as $10^6 M_\odot$, where $M_\odot$ is the solar mass. It has been speculated that this self similarity could be preserved even down to mass scales as small as $10^{-6} M_\odot$.

[1] As comparison the mass of the proton and neutron are both slightly below 1 GeV and the electron mass is half an MeV.

1.4 The GLAST satellite

The Gamma-ray Large Area Space Telescope or GLAST is a space observatory that will study high energy γ-rays from some of the most exotic events in the Universe. Some of these events are pulsars and super massive black holes. The GLAST satellite will hopefully answer some very old questions about the universe. The satellite which was launched from Cape Canaveral on June 11th 2008 [7], will also make an all sky survey of our own galaxy in the hope of detecting a DM annihilation signal.

1.5 This Thesis

My contribution to the search for DM is to theoretically and numerically predict how many DM clumps you could find if you from the Earth were to look at the Milky Way. I will use the density profile adopted by Navarro, Frenk and White, the NFW profile. I will use the same approach as Pieri. et. al.[11] and assume that the DM clumps follow a NFW distribution matching the host halo but with tidal forces taken into account. To more accurately describe the DM halo distribution it is necessary to take into account effects such as tidal forces, virialization and the formation of structures in the early universe.

2 Theory

From the Via Lactea simulation you can see that the radial distribution of mass follows the smooth component of the DM halo, but with an important exception. Due to tidal forces, large sub-halos cannot exist near the center of the galaxy without being spagettified and broken up into smaller pieces, much like the Shoemaker–Levy comet was ripped apart by the tidal field around Jupiter.

The sub-halofication means that you can decompose the density ρ at a point $\mathbf{r}$ like

$$\rho(\mathbf{r}) = \rho_H(r) + \sum_i \rho_i([\mathbf{r} - \mathbf{r}_i])$$

if we put origo at the center of the galaxy. Here ρ_H is the density of the smooth host halo while ρ_i is the density of the i^{th} sub-halo. The luminosity from a small volume of space is proportional to ρ^2 which will complicate the numerical evaluation luminosity integral.

2.1 The NFW profile

The density profile developed by Navarro, Frenk and white, is a broken power law that is r^{-1} close to the center, but r^{-3} further out. Usually it is written as

$$\rho_{NFW}(r) = \frac{\rho_s}{\frac{r}{r_s}\left(1 + \frac{r}{r_s}\right)^2} \tag{1}$$

where r_s is the radius at which the profile goes from r^{-1} to r^{-3} [9, 11]. The NFW profile does have some problems of which one is that the volume integral of the function is divergent. If we want to calculate the mass of a halo with parameters ρ_s and r_s inside a radius R we get

$$\begin{aligned} M(R) &= \int \rho_{NFW}(r)\, d^3r = \int_0^{\pi} \sin\theta\, d\theta \cdot \int_0^{2\pi} d\phi \cdot \int_0^R \frac{\rho_s}{\frac{r}{r_s}\left(1+\frac{r}{r_s}\right)^2} r^2\, dr \\ &= 4\pi \int_0^R \frac{r_s^3 \rho_s r}{(r_s+r)^2}\, dr = 4\pi r_s^3 \rho_s \left[\ln\left(1+\frac{R}{r_s}\right) - \frac{R}{r_s+R}\right] \qquad (2) \end{aligned}$$

We can clearly see from equation (2) that $M(R) \to \infty$ as $R \to \infty$. In figure 1 we can see how the broken power law makes the density profile go from r^{-1} to r^{-3} in the region of $R = 1$. We can also see how the encapsulated mass goes from growing as r^2 to the slower but still divergent $\ln r$.

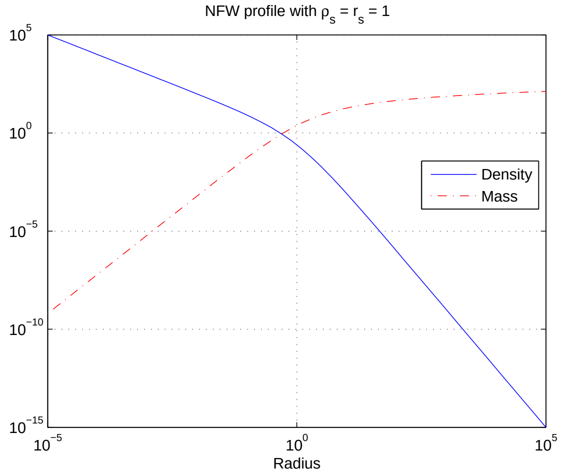

Figure 1: The density $\rho(R)$ at, and encapsulated mass $M(R)$ within, the radius R.

To handle the divergence we have to choose a cut-off radius outside which we say that the density of the halo is zero. There are no absolute way to do this but a common feature is to say that the density of equation (1) is non zero only within a virial radius. A cluster of objects is said to be virialized when all of them are held together by their combined gravity. In reality the halos of several different sub-halos will mix and affect each other.

2.1.1 Virial radius and virial mass

The virial radius can be defined in many ways, but a common definition is the following. Assume an object with radial density $\rho(r)$ and encapsulated mass

$m\left(R\right)=4\pi\int_0^R f\left(r\right)r^2\,dr$. The mean density of the object inside R is

$$\left\langle\rho\right\rangle\left(R\right)=\frac{m\left(R\right)}{V\left(R\right)}=\frac{4\pi\int_0^R\rho\left(r\right)r^2\,dr}{\frac{4\pi}{3}R^3}=3R^{-3}\int_0^R\rho\left(r\right)r^2\,dr$$

The virial radius, r_{vir}, is defined as the radius at which the mean density inside of r_{vir} is k times the critical density of the universe, ρ_0. Usually $k=200$ [11]. The critical density, ρ_0, is obtained from the Friedman equation of a flat universe

$$H^2=\left(\frac{\dot{a}}{a}\right)^2=\frac{8\pi G}{3}\rho_0$$

where G is Newtons gravitational constant and H the Hubble constant. The variable a is called the scale factor and tracks the expansion of the universe [1]. Since $H^2=\left(\frac{\dot{a}}{a}\right)^2$ you can say that H is related to the fractional expansion of the universe.

Reorganizing the equation above we get

$$\rho_0=H^2\frac{3}{8\pi G}\tag{3}$$

Using the definitions above and defining $m\left(r_{\text{vir}}\right)\equiv m_{\text{vir}}$ we have the relation

$$m_{\text{vir}}=\frac{4\pi}{3}k\rho_0 r_{\text{vir}}^3\tag{4}$$

The above equation is interesting since it says that for a given mass the virial radius is well defined by

$$r_{\text{vir}}=\sqrt[3]{\frac{3m_{\text{vir}}}{4\pi k\rho_0}}\tag{5}$$

Observe that the relation (5) does not depend on the specific form of $\rho\left(r\right)$ but is a direct consequence of the definition of r_{vir}. This means that you could if you want use (5) as the definition of virial radius. In the future analysis the simple form of r_{vir} will be of great benefit.

2.1.2 The concentration parameter

In the NFW profile we have the parameter ρ_s and r_s which govern the scale and shape of the NFW density. However r_s is tricky to extract from formula (2). You can invert $M\left(R\right)$ with respect to r_s but the final formula involves the Lambert function, $L\left(x\right)$, which is the solution to $we^w=x$, and this function is not on closed form.

Instead we can use the concentration parameter c which is defined as

$$c\equiv\frac{r_{\text{vir}}}{r_s}$$

This means that if we have the virial mass and concentration parameter of a halo, we can easily calculate the scale density, ρ_s and scale radius r_s. If we insert that $r_{\text{vir}} = cr_s$ into $M\left(r_{\text{vir}}\right)$ we get

$$\begin{aligned} m_{\text{vir}} &= 4\pi r_s^3 \rho_s \left[\ln\left(1+\frac{cr_s}{r_s}\right) - \frac{cr_s}{r_s + cr_s}\right] = \\ &= 4\pi r_s^3 \rho_s \left[\ln\left(1+c\right) - \frac{c}{1+c}\right] \end{aligned}$$

We can rearrange the equation above and insert (4) to get

$$\begin{aligned} \rho_s &= \frac{m_{\text{vir}}}{4\pi r_s^3}\left[\ln\left(1+c\right) - \frac{c}{1+c}\right]^{-1} \\ &= \frac{k\rho_0 c^3}{3}\left[\ln\left(1+c\right) - \frac{c}{1+c}\right]^{-1} \qquad (6) \end{aligned}$$

and

$$r_s = \frac{r_{\text{vir}}}{c} = \frac{1}{c}\sqrt[3]{\frac{3m_{\text{vir}}}{4\pi k\rho_0}} \qquad (7)$$

The concentration parameter is very useful since it is dimensionless. The scale of c is usually around $c \approx 10$ and larger. In the early universe structure formed hierarchically and thus small structures formed earlier when the universe was more dense. These small sub-halos usually have a higher concentration than larger sub-halos.

2.1.3 Units

Usually you measure the critical density in g/cm^3 where you usually write ρ_0 as

$$\rho_0 = 1.9 \cdot h^2 \cdot 10^{-29}\ \text{g} \times \text{cm}^{-3}$$

where the uncertainty in the value of ρ_0 is in h. Present observations bound $h \in [0.6, 0.8]$ [1]. If we want to transform the units of ρ_0 into solar masses and parsec which will be used in the simulations. The relation between gram and solar masses $M_\odot$ as well as between cm and parsec is

- 1 centimeter $= 3.2407 \times 10^{-19}$ parsec $\leftrightarrow$1 parsec $= 3.0857 \times 10^{18}$ centimeter
- 1 gram $= 5.027 \times 10^{-34}$ solar masses $\leftrightarrow$ 1 solar mass $= 1.989 \times 10^{33}$ gram [10]
- 1 gram$\times$centimeter^{-3} $= 1.477187 \times 10^{22}$ solar masses$\times$parsec^{-3}

In these units the critical density is

$$\rho_0 = 2.1 \cdot h^2 \cdot 10^{-7}\, M_\odot \times \text{pc}^{-3}$$

In the numerical simulation ρ_0 will be in units of $M_\odot \times \text{pc}^{-3}$.

2.2 The sub-halo distribution

In several numeric simulations of the mass distribution of sub-halos within Milky Way-like galaxies you can see that the mass distribution is very self-similar over large mass scales ranging from $10^6 \cdot M_\odot$ and above to around $10^{12} \cdot M_\odot$. A span of six orders of magnitude where the number of halos per mass decade seems to fall of like $M^{-\lambda}$, where λ is in the vicinity of $\lambda \approx 2$. The reason you do not have masses bellow $10^6 M_\odot$ is because of the numerical resolution of the simulations. Theoretical calculations say that you should be able to have sub-halos down to the size of $10^{-6} M_\odot$. It seems reasonable that you could extend the self similar distribution of sub-halos down to a size near the theoretical lower bound.

As the simulations imply the radial number density of the sub-halos should follow the radial density of the Milky Way. However there is one important exception due to the fact that DM is collisionless matter. Large sub-halos will probably be torn apart by tidal forces in the inner of the galaxy. This fact is important to take into considerations and will make the number density function look like

$$\rho_N\left(r, M\right) = AM^{-\lambda} \frac{\Theta\left(r - R_T\left(M\right)\right)}{\frac{r}{r_{MW}}\left(1 + \frac{r}{r_{MW}}\right)^2} \tag{8}$$

where r_{MW} is the scale radius of the Milky Way and $R_T\left(M\right)$ is the radius at which a sub-halo with mass M is destroyed by tidal forces. See section 2.5 for more details. In this thesis I will primarily use $\lambda = 2$ and will state whenever I use another value of λ. The function $\Theta\left(x\right)$ is the usual Heaviside function

$$\Theta\left(x\right) = \begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases} \tag{9}$$

The constant A is chosen so that

$$1 = \int_{MW} \int_{M_{\min}}^{M_{\max}} \rho_N\left(r, M\right)\, d^3r\, dM$$

however due to the Heaviside function which make the two integrals over space and mass non-multiplicative the normalization A is hard to calculate, see Appendix A. That we cannot find the normalization of ρ_N will not be a problem since in the end we want only to generate a distribution according to $\rho_N\left(r, M\right)$.

2.3 Generating distributions

We will first discuss how to generate random numbers. Assume that you have a stochastic variable $X = x$ in the region between A and B, that you know has the normalized probability density function $f\left(x\right)$. Since $f\left(x\right)$ is a density function we know that it fulfills the relation

$$\int_A^B f\left(x\right)\, dx = 1$$

If we want to change variables from x to y the density function for y will be $g(y) = f(x(y)) \cdot \frac{dx}{dy}$ since

$$\int_A^B f(x)\ dx = [x = x(y)] = \int_{y(A)}^{y(B)} f(x(y)) \cdot \frac{dx}{dy}\, dy$$

Usually most programming languages can only generate a limited amount of distributions. Some languages can only generate according to the uniform distribution. This means that we need to find a variable that is uniformly distributed from which we can calculate the value x. Such a variable is p, defined as

$$p(x) \equiv \text{Prob}(X \leq x) = \int_A^x f(x')\ dx'$$

which is the cumulative probability that $X \leq x$. It is easy to see that $p(A) = 0$ and $p(B) = 1$. The interpretation of p is the probability of obtaining a value of X that is smaller than x. That p will be uniform we can see if we look at the density function of p, $g(p)$. From the definition of p we have $dp = f(x)\ dx$ which means that

$$g(p) = f(x(p)) \cdot \frac{dx}{dp} = g(p) = f(x(p)) \cdot \frac{1}{f(x(p))} = 1$$

which is the uniform distribution. To generate the value of x we now only need to invert the relation $p = p(x)$ to $x = x(p)$. As an example we can take the value of θ that is sinusoidally distributed as $f(\theta) = \sin\theta$. Our $p_\theta(\theta)$ becomes

$$p_\theta = \int_0^\theta \sin\theta'\, d\theta' = [\cos\theta']_\theta^0 = 1 - \cos\theta$$

Given a random p_θ we now calculate θ as $\theta = \arccos(1 - p_\theta)$.

If our function $p(x)$ would not be invertable other methods have to be applied, such as the ones mentioned in section 2.4.

2.3.1 Multiple variables

Sometimes you have two variables x and y with the joint distribution $f(x, y)$ such that $A_x \leq x \leq B_x$ and $A_y \leq y \leq B_y$. Still we have

$$\int_{A_x}^{B_x} \int_{A_y}^{B_y} f(x, y)\ dx\, dy = 1$$

if we want to generate x and y we can make the same procedure as above only we do it in two steps. First we retrieve the normalization as

$$1 = \int_{A_x}^{B_x} \left(\int_{A_y}^{B_y} f(x, y)\ dy \right) dx = \int_{A_x}^{B_x} f_{\text{eff}}(x)\ dx$$

where $f_{\text{eff}}(x)$ now is the effective distribution of x as a single variable. Assuming that $p_x = \int_{A_x}^{x} f_{\text{eff}}(x')\, dx'$ is invertible we now define p_y like

$$p_y(y) = \frac{\int_{A_y}^{y} f(x, y)\, dy}{\int_{A_y}^{B_y} f(x, y)\, dy}$$

and we get the correct normalization giving $p_y(A_y) = 0$ and $p_y(B_y) = 1$. This procedure can of curse be generalized to N dimensions in an analogous manner to the generalization from one to two.

2.4 Metropolis algorithm

Sometimes however you are not so lucky that you can invert the function $p(x)$, or even worse, it is impossible to find an analytical form of $p(x)$. In these cases something else is needed, like the metropolis algorithm. Assume that you have a density function $f(x)$ that need not be normalized. Sometimes it cannot even be normalized. The idea is that if you let a variable x_n make a random walk over the defined space you will retrieve a list of numbers whose distribution matches $f(x)$.

We start by choosing a starting value of x_n. Now let x_n make a random jump to a new point x_r by letting $x_r = x_n + \epsilon$ where ϵ is distributed symmetrically around zero[2]. Two examples of distribution for ϵ are the Gaussian and uniform distribution. In our simulations we choose a uniform distribution between $-\sigma$ and σ.

We construct the quotient, q, between the values of the density function in the points x_r and x_n by

$$q = \frac{f(x_r)}{f(x_n)}$$

Now one of two things happens.

- If $q \geq 1$ let $x_{n+1} = x_r$.
- If $q < 1$ let $x_{n+1} = x_r$ with probability q.
- Else let $x_{n+1} = x_n$.

Usually one trows away the N first x-values to give the number generator a chance to forget about the initial guess.

The trick in using the metropolis algorithm is

1. to choose the initial guess x_0 correctly so that you are on a place where $f(x_0) \neq 0$

[2] The metropolis algorithm can handle distributions that are not symmetrical but here we do not bother about that.

2. to choose the range of randomness such that the range of x_r is roughly over the range of the interval of interest. If the range of x_r should be to small you will need a large N to forget the initial value of x_0. If on the other hand you choose a range of x_r that is to large you might find yourself with a series of x_n that are identical.

A usual way to check that the choosen parameters are good for the algorithm is to count the acceptance rate, i.e the percentage of numbers accepted by the algorithm. Usually one would like an acceptance rate near 60%.

You can easily generalize the metropolis algorithm to handle vector arguments by letting $x \to \mathbf{x}$ and $x_r = x_n + \epsilon \to \mathbf{x}_r = \mathbf{x}_n + \bar{\epsilon}$ where $\bar{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)$ and every ϵ_j is randomly distributed unrelated to the other components of $\bar{\epsilon}$ [14, 5].

It is important to point out that this form of the Metropolis algorithm *does not* produce numbers that are independent from each other. Rather the numbers gained from the Metropolis algorithm is a Markov chain with a total distribution shaped like $f(x)$. Therefore it is not advisable to use this kind of Metropolis algorithm if it is important that the generated numbers are independent of each other as well.

In our case we do not have this problem, since it is only the placements of the stars that are interesting, not the order in which they where placed on the sky.

2.4.1 The NFW distribution

As we pointed our earlier we cannot use standard generation procedure described in section 2.3 to generate a distribution according to equation (8). We will need the metropolis algorithm, but since the value of ρ_N falls of like a power law at higher masses and greater distances from the center of the galaxy it will be hard go get an efficient algorithm when the part with the highest values of ρ_N also takes up the smallest area. What we can do is to change coordinates to logarithmic coordinates such that $M \to e^m$ and $r \to e^l$ where $dM \to e^m\, dm$ and $dr \to e^l\, dl$. If we insert this and integrate over (8) we get

$$\begin{aligned}
1 &= A \int_{M_{\min}}^{M_{\max}} \int_0^{r_{\mathrm{vir}}} \frac{r}{M^2} \frac{\Theta\left(r - R_T(M)\right)}{\left(r_{MW} + r\right)^2}\, dr\, dM \\
&= A \int_{\ln(M_{\min})}^{\ln(M_{\max})} \int_{-\infty}^{\ln(r_{\mathrm{vir}})} e^{l-2m} \frac{\Theta\left(e^l - R_T(e^m)\right)}{e^l \left(r_{MW} + e^l\right)^2}\, e^l dl\, e^m dm \\
&= A \int_{\ln(M_{\min})}^{\ln(M_{\max})} \int_{-\infty}^{\ln(r_{\mathrm{vir}})} e^{2l-m} \frac{\Theta\left(e^l - R_T(e^m)\right)}{\left(r_{MW} + e^l\right)^2}\, dl\, dm
\end{aligned}$$

This means that we can use Metropolis algorithm with

$$f(l, m) = e^{2l-m} \frac{\Theta\left(e^l - R_T(e^m)\right)}{\left(r_{MW} + e^l\right)^2} \tag{10}$$

It is easy to see if the distribution is correct by simply plotting the function on top of a histogram of the generated numbers and making sure that they coincide. I have made sure that the Metropolis algorithm gives the right distribution. The acceptance rate is usually around 53%.

One has to be careful when one chooses that starting guess. If you pick a l_0 and/or m_0 where $f(l_0, m_0) = 0$ then you might never have any other values than the starting guess depending on how the computer interprets $\frac{1}{0}$ and $\frac{0}{0}$. Especially the case with both $f(l_0, m_0) = 0$ and $f(l_r, m_r) = 0$ is hazardous if q is interpreted as $\neq 0$ since you could have a random walk taking you away from the region where $f \neq 0$. In GalaxyHalo the starting guess is set at $m = \ln\left(5 \cdot 10^6 M_\odot\right)$ and $l = \ln\left(\frac{r_{\text{vir}}}{10}\right)$ and it has worked fine during all the simulations.

Of course we have to give our sub-halo a ϕ and θ value as well but to generate them we can use standard methods since ϕ is uniform between 0 and 2π and θ is sinusoidal between 0 and π.

2.5 Tidal forces and tidal disruption

There is one more thing we need to clear out before we can generate the ρ_N distribution. That is the so called Roche distance, R_T, which is the radius inside of which the sub-halo is torn apart by tidal forces. Let us take it from the start. Assume that you have a distribution of mass that has accumulated mass $M(r)$ inside the radius r. Assume that you have a piece of mass m at radius r. The force that m experiences from the rest of the halo is

$$F_G = G\frac{M(r)\,m}{r^2}$$

Now assume that the small halo is embedded within a larger central halo. This central halo has an accumulated mass function of $M_H(d)$ where d is the distance from the halo center. Our small mass m, which we assume is as close to the center halo as possible, will of course feel a gravitational pull from the center galaxy as well. That force is

$$F_H = G\frac{M_H(d-r)\,m}{(d-r)^2} \approx \frac{GM_H(d-r)\,m}{d^2} \cdot \left\{1 + \frac{r}{d}\right\} = F_V + F_T$$

The first term, F_V, is the usual gravitational force that comes from the mass at distance d. The second term is the one we are interested in since it says how big the tidal force acting on the piece of mass is. If the tidal force is greater than the self-gravity the sub-halo will be torn apart by the central halo [16]. Therefore we seek the radius where $F_T < F_G$. This gives

$$\begin{aligned} G\frac{M_H(d-r)\,mr}{d^3} &< G\frac{M(r)\,m}{r^2} \\ &\Updownarrow \\ \frac{M_H(d-r)}{d^3} &< \frac{M(r)}{r^3} \end{aligned} \tag{11}$$

which is the Roche criterion in a general form.

2.5.1 Roche criterion for NFW profile

Let us apply equation (11) to the NFW profile. Both M_H and M are on the form $M_H(R) = 4\pi r_{s_{MW}}^3 \rho_{s_{MW}} \left[\ln\left(1+\frac{R}{r_{s_{MW}}}\right) - \frac{R}{r_{s_{MW}}+R}\right]$ where $s_{MW} \to s$ for the sub-halo. Remember that $M(r)$ is divergent so we need to define the cut-off distance for r. The choice we will make so that we are consistent with what we have done earlier is to let $r = r_{\text{vir}}$.

With the use of r_{vir} the right hand side of equation (11) becomes just a constant. Recall formula (4) on page 8 and insert it into the left hand side and we have

$$\left.\frac{M(r)}{r^3}\right|_{r=r_{\text{vir}}} = \frac{m_{\text{vir}}}{r_{\text{vir}}^3} = \frac{4\pi}{3}k\rho_0 \tag{12}$$

which is a constant. This is good since the only part where r_{vir} enters now is in $M_H(d - r_{\text{vir}})$. If we let $r_{\text{vir}} \to 0$ we get the equation

$$\frac{M_H(d)}{d^3} = \frac{4\pi}{3}k\rho_0 \tag{13}$$

The solution to this equation is the distance inside of which no sub-halos survive tidal forces. In Appendix A we will see that for small values of r_{vir} the Roche distance is $R_T = d_0 + \alpha r_{\text{vir}}$ where d_0 is the solution to equation (13) and α is a factor close to one.

2.5.2 Bypassing the numerical solution

Let us return to the original Roche criteria that says

$$\frac{M_H(d - r_{\text{vir}})}{d^3} < \frac{4\pi}{3}k\rho_0$$

Remember that in equation (8) we are really only interested in if you are inside the radius of tidal disruption or not. This means that we do not really need to find the exact value of d that solves the Roche criterion. It is really enough to see if the inequality is fulfilled or not. Therefore you can change $\Theta(r - R_T(M))$ to $\Theta\left(\frac{4\pi}{3}k\rho_0 - \frac{M_H(r - r_{\text{vir}}(M))}{r^3}\right)$ where r_{vir} is defined by equation (5).

2.6 The Line of Sight

When we have generated all of the sub-halos in the Galaxy we would like to calculate what a telescope somewhere in the Galaxy would be able to see. The rate of detection for a telescope depends on many factors such as the geometry of the detector, the astrophysical properties of the target and the physical properties of the medium of interest.

Assume a small volume element dV at a distance r from the Earth. This volume has a source density $S(E_\gamma, \mathbf{r})$ that determines the number of photons of a specific energy that are emitted per volume time and energy. The photons will be emitted in all directions and thus when they have traveled the distance to the Earth be spread over a sphere with area $4\pi r^2$. Since the distance to dV is much greater than the dimension of the detector we can assume that the photon enter parallel to each other. The number of photons detected from the source will thus be $\frac{G(E_\gamma)}{4\pi r^2} S\left(E_\gamma, \mathbf{r}\right)$ where $G\left(E_\gamma\right)$ is the effective area of the detector.

If we denote the total amount of photons with energy E_γ that are detected per time unit, by $R\left(E_\gamma\right)$ from the entire universe we have the relation

$$\begin{aligned}
R\left(E_\gamma\right) &= \int_{\Delta V} \frac{G\left(E_\gamma\right)}{4\pi r^2} S\left(E_\gamma, \mathbf{r}\right) \, dV \\
&= G\left(E_\gamma\right) \int_{\Delta\Omega} \int_{\text{l.o.s}} \frac{S\left(E_\gamma, \mathbf{r}\right)}{4\pi\lambda^2} \lambda^2 \, d\lambda \, d\Omega \\
&= G\left(E_\gamma\right) \int_{\Delta\Omega} \int_{\text{l.o.s}} \frac{1}{4\pi} S\left(E_\gamma, \mathbf{r}\right) \, d\lambda \, d\Omega
\end{aligned}$$

where $\int_{\text{l.o.s}} = \int_0^\infty$ is the "line of sight" integral in the direction of ψ.

Let us look a little bit closer at the source $S\left(E_\gamma, \mathbf{r}\right)$. This factor actually consists of two separate parts and can be written as

$$\frac{1}{4\pi} S\left(E_\gamma, \mathbf{r}\right) = \frac{d\Phi^{PP}}{dE_\gamma}\left(E_\gamma\right) \cdot \rho^2\left(\mathbf{r}\right)$$

where ρ is the mass density of DM and $\frac{d\Phi^{PP}}{dE_\gamma}\left(E_\gamma\right)$ is the factor scaling with the amount of photons (of energy E_γ) that will be emitted from a volume dV with density squared ρ^2. This factor does only depend on the particle physics involved and can be written as

$$\frac{d\Phi^{PP}}{dE_\gamma}\left(E_\gamma\right) = \frac{1}{4\pi} \frac{\sigma_{\text{ann}} v}{2m_\chi^2} \cdot \sum_f \frac{dN_\gamma^f}{dE_\gamma} B_f$$

where m_χ is the MD particle mass, $\frac{dN_\gamma^f}{dE_\gamma}$ the differential photon spectrum and B_f the branching ratio. The v is just the mean speed of the DM particles and σ_{ann} the cross section for annihilation [11]. I will not engage in this term as the outcome depends strongly on what kind of DM particle that is considered.

If we just look at the differential detection rate for a certain angle ψ, we get

$$\begin{aligned}
\frac{dR}{d\Omega}\left(E_\gamma, \psi\right) &= G\left(E_\gamma\right) \int_{\text{l.o.s}} \frac{d\Phi^{PP}}{dE_\gamma}\left(E_\gamma\right) \cdot \rho^2\left(\lambda, \psi\right) \, d\lambda \\
&= G\left(E_\gamma\right) \cdot \frac{d\Phi^{PP}}{dE_\gamma}\left(E_\gamma\right) \times \int_{\text{l.o.s}} \rho^2\left(\lambda, \psi\right) \, d\lambda \\
&= G\left(E_\gamma\right) \cdot \frac{d\Phi^{PP}}{dE_\gamma}\left(E_\gamma\right) \times J\left(\psi\right) \qquad (14)
\end{aligned}$$

It is the last factor $J\left(\psi\right)$ that we will focus on since the other factors in the end are only numbers or distributions independent of the DM distribution within the Universe. The factor we will calculate is thus

$$J\left(\psi\right) = \int_{\text{l.o.s}} \rho^2\left(\lambda, \psi\right) \, d\lambda$$

which will be in units $[J] = M_\odot^2 \cdot \mathrm{pc}^{-5}$.

2.6.1 A small clarification

When one looks at $J(\psi)$ above one might naively get the impression that the distance to the volume element dV does not matter any more. This is partly true and partly not true. Intuition tells us that an object becomes fainter the further away you are from it which is perfectly true. However when it comes to the value of $J(\psi)$ for a specific angle ψ the distance to dV does not matter. The whole thing becomes much clearer if we remember that $J(\psi)$ is an angular density.

Let us make an example to straighten things out. Here we only use planar polar coordinates for simplicity, but the effect is the same for spherical polar coordinates. Imagine a circle of radius R, placed at $x = d$ on the x axis. How does this sphere look in polar coordinates (centered at $x = y = 0$)? The further away the circle is placed from origo the smaller the area is occupied in polar space. The equation describing our circle is

$$(x-d)^2 + y^2 = R^2$$

If we change to polar coordinates using $x = r\cos\psi$ and $y = r\sin\psi$ we get

$$R^2 = (r\cos\psi - d)^2 + r^2\sin^2\psi = d^2 + r^2 - 2rd\cos\psi \tag{15}$$

This is really the trigonometric cosine rule which we could have otherwise derived from strict geometrical considerations. Let us investigate what happens when d is small, and when d is large. If $d = 0$ then equation (15) becomes $R^2 = r^2$ which is just a line in polar space. The value of ψ ranges from 0 to 2π. When $d \neq 0$, we can rewrite (15) as

$$\cos\psi = \frac{d^2 + r^2 - R^2}{2rd}$$

If parametrize $r^2 = x^2 + y^2$ by $x = R\cos\theta + d$ and $y = R\sin\theta$ we get $r^2 = R^2 + d^2 + 2Rd\cos\theta$. Inserting this into the equation above gives us

$$\cos\psi = \frac{d + R\cos\theta}{r}$$

If we write $R = kd$ we get

$$\cos\psi = \frac{1 + k\cos\theta}{\sqrt{1 + k^2 + 2k\cos\theta}}$$

It is now easy to see that when $k \ll 1$ then $\cos\psi \approx 1$ and thus $\psi \approx 0$. Later we will find out that the maximum angle an object at distance d and radius R occupies on the sky is $2\Delta\theta = 2 \cdot \arcsin\left(\frac{R}{d}\right)$. This means that the maximum differential intensity of an object is unchanged no matter the distance. The spread of the object will however be smaller the further away it is and therefore the total luminosity integral will be smaller.

2.7 Coordinate transformations - The short version

Since any measurements made by GLAST or any other telescope will be centered at Earth we need to move the coordinate system in such a way that origo is at the Earth and not in the center of the galaxy (CoG). We also need to decide how to orient this coordinate system to get a good view. Since we want to use spherical polar coordinates presented in a Mollweide projection we would like the CoG to be centered in the picture. This means that we should orient the coordinate system of the Earth so that CoG has coordinates $\theta = \frac{\pi}{2}$ and $\phi = \pi$.

The way we do this is that we first transform the polar coordinates of the CoG to Cartesian coordinates using the standard transformation

$$\begin{aligned} x &= r \sin\theta \cos\phi \\ y &= r \sin\theta \sin\phi \\ z &= r \cos\theta \end{aligned}$$

Then we the move the coordinate system to the Earth by letting $(x_E, y_E, z_E)_C = (0,0,0)_E$. By choosing a clever position for the Earth we do not need to rotate the coordinate system when we have transposed it. In this case the clever position of the Earth is at $(r_E, 0, 0)_C$ which will make the CoG positioned at $(-r_E, 0, 0)_E$ which is precisely $\theta = \frac{\pi}{2}$ and $\phi = \pi$. It is worth to point out that since all stars are spherically symmetrically distributed we can choose the angles of the Earth without loosing the randomness. The same cannot be said for the value of r_E. If r_E is small we will be closer to CoG and the view will be more homogeneous than if r_E is large. In this case the sky map will be noticeably more luminous in the direction of CoG and fall of rapidly outwards.

The coordinate transform is therefore the simple

$$\begin{aligned} x' &= x - r_E \\ y' &= y \\ z' &= z \end{aligned}$$

When the new Cartesian coordinates are in place we can calculate the new polar coordinates as

$$\begin{aligned} r' &= \sqrt{x'^2 + y'^2 + z'^2} \\ \theta' &= \arccos\left(\frac{z'}{r'}\right) \\ \phi' &= \arctan\left(\frac{y'}{x'}\right) + \Theta\left(-x'\right)\pi \end{aligned}$$

The addition of $\Theta\left(-x'\right)\pi$ is because the range of ϕ' is 2π while the image of $\arctan\alpha$ is only π.

2.8 Angular integration

Like mentioned before the luminosity from a small volume in space is proportional to the square of the density. $\Delta L \propto \rho^2 \Delta V$. This means that the total luminosity can be written as

$$\begin{aligned} L &= A \int_V \frac{\rho^2(\mathbf{r})}{r^2} d^3 r = A \int_\Omega \left(\int_0^\infty \rho^2(\mathbf{r})\, dr \right) d\Omega \\ &= A \int_\Omega J(\psi)\, d\Omega \end{aligned}$$

where ψ is a general coordinate of both θ and ϕ. We are not really interested in the entire value of L, rather we are interested in the angular distribution of the astrophysical factor

$$J(\psi) = \int_{\text{l.o.s}} \rho^2(\mathbf{r} = (\lambda, \psi))\, d\lambda$$

Even for a single halo the integral above is complicated and the fact that we have a lot of lumpiness does not help. With the problems of doing an analytical integration of $J(\psi)$ we will need to numerically evaluate the function for certain values of ψ.

The full form off $\rho(\mathbf{r})$is

$$\rho(\mathbf{r}) = \rho_{Hosthalo}(|\mathbf{r} - \mathbf{r}_O|) + \sum_i \rho_i(|\mathbf{r} - \mathbf{r}_i|)$$

where ρ_i is the density of the ith sub-halo centered at $\mathbf{r}_i$. The argument $|\mathbf{r} - \mathbf{r}_i|$ can be calculated as

$$|\mathbf{r}_i - \mathbf{r}|^2 = r^2 + r_i^2 - 2rr_i \{\cos(\theta_i - \theta) + \sin\theta \sin\theta_i [\cos(\phi_i - \phi) - 1]\}$$

A special case is $|\mathbf{r} - \mathbf{r}_O|$ and since we have chosen $\theta_O = \frac{\pi}{2}$ and $\phi_O = \pi$ we get

$$\begin{aligned} |\mathbf{r}_O - \mathbf{r}|^2 &= r^2 + r_O^2 - 2rr_O \left\{\cos\left(\frac{\pi}{2} - \theta\right) + \sin\theta \sin\frac{\pi}{2} [\cos(\pi - \phi) - 1]\right\} \\ &= r^2 + r_O^2 - 2rr_O \{\sin\theta - \sin\theta [\cos\phi + 1]\} \\ &= r^2 + r_O^2 + 2rr_O \sin\theta \cos\phi \end{aligned}$$

2.8.1 Integrating over the sum

If we assume that the host halo has index 0 we get

$$\rho(\mathbf{r}) = \sum_{i=0} \rho_i(|\mathbf{r} - \mathbf{r}_i|)$$

and thus

$$J(\psi) = \int_0^\infty \left[\sum_{i=0} \rho_i(|\mathbf{r} - \mathbf{r}_i|)\right]^2 d\lambda \tag{16}$$

We will need to numerically integrate and evaluate the integral in (16) and to do so we will use a simple rectangular approximation that

$$\int_0^\infty f(x)\;dx = \sum_{j=1}^{\infty} f(x_j)\cdot\Delta x$$

It is not practical to have an infinite sum when you do a numerical evaluation. Fortunately the choice of where to cut of the sum is rather easy to make since we know that there are no halos outside the virial radius of the host halo[3]. In this case we choose the cutoff at $\lambda_{N_r} = 2r_{\mathrm{vir\,Halo}}$ which makes $\Delta\lambda = \frac{2r_{\mathrm{vir\,Halo}}}{N_r}$ and $\lambda_j = j\cdot\Delta\lambda$. We choose to evaluate $J(\psi)$ at the angles $\phi_l = (l-1)\cdot\Delta\phi$ and $\theta_k = \left(k-\frac{1}{2}\right)\cdot\Delta\theta$ with $\Delta\phi = \frac{2\pi}{N_\phi}$ and $\Delta\theta = \frac{\pi}{N_\theta}$. All indexes j, k and l start at 1. Putting all these thing together we get

$$J(\psi_{kl}) = J(\theta_k,\phi_l) = \frac{2r_{\mathrm{vir\,Halo}}}{N_r}\cdot\sum_{j=1}^{N_r}\left[\sum_{i=0}^{N}\rho_i\left(|\mathbf{r}_{jkl}-\mathbf{r}_i|\right)\right]^2 \tag{17}$$

where $\mathbf{r}_{jkl} = (\lambda_j\sin\theta_k\cos\phi_l, \lambda_j\sin\theta_k\sin\phi_l, \lambda_j\cos\theta_k)$.

2.8.2 Scaling the sub-halo

As mentioned in 2.2, usually the number of halos necessary to get the correct total mass is more than you you can generate. If we let M_S be the total mass of the sub-halos and M_H be the mass of the host halo, the total mass of the galaxy is $M_T = M_S + M_H$. We would like to rescale M_S with a factor q in such a way that the fractional mass of $M_S' = qM_S$ is p. The definition of p is

$$p = \frac{M_S}{M_S + M_H} \tag{18}$$

which gives

$$M_S = M_H\left(\frac{p}{1-p}\right)$$

If we insert that $M_S = N\cdot\langle M\rangle$ where $\langle M\rangle$ is the average mass of all the sub-halos and N the number of sub-halos and let $M_S \to M_S'$ we get an equation for q that is

$$q = \frac{M_H}{N\cdot\langle M\rangle}\left(\frac{p}{1-p}\right) \tag{19}$$

How we should implement this normalization is not clear cut. As I see it there are two possibilities. The first one is to simply let $\rho(r) = \sum q\cdot\rho_S(M_{\mathrm{vir}}, r) + \rho_H$, and the other is to let $\rho(r) = \sum\rho_S(q\cdot M_{\mathrm{vir}}, r) + \rho_H$. These two ways of rescaling the sub-halos both have advantages and disadvantages.

[3] There might in fact be some parts of the sub-halos that are outside of the radius since the restriction is limited to the center of mass.

In the first case we just scale the density of the sub-halos with a factor of q. This will make the sub-halos stronger than they actually are and enhance the lumpiness. This will effectively change the concentration of the sub-halos since letting $\rho_S\left(r\right) \rightarrow q \cdot \rho_S\left(r\right)$ is related to letting $\rho_s \rightarrow q \cdot \rho_s$.[4] Remember that ρ_s is a function of concentration only and if we change ρ_s the concentration will change as well.

In the second case we scale the mass of the sub-halo. This means that the viral radius will change since there is a one-to-one relation between virial mass and virial radius. Changing the tidal radius means that some of the halos will be inside of their virial radius.

I have chosen the first way of renormalizing since it is slightly easier to implement. As for the value if p I have chosen $p = 0.15$ since that is what Pieri et. all. uses [11].

One must be careful when one uses the renormalization because you can get really ugly effects if the scaling factor is large. An example of this can bee seen in figure 2 where two distribution of $N = 10^6$ sub-halos with a lower mass of $M_{\text{min}} = 10^{-3} M_{\odot}$ are compared. In this case the q-factor is approximately $1.6 \cdot 10^7$ which greatly distorts the image.

[4] Note that ρ_S is the density profile of the sub-halo whereas ρ_s is the density scale of the sub-halo. This unfortunate resemblance of variables will only occur here though.

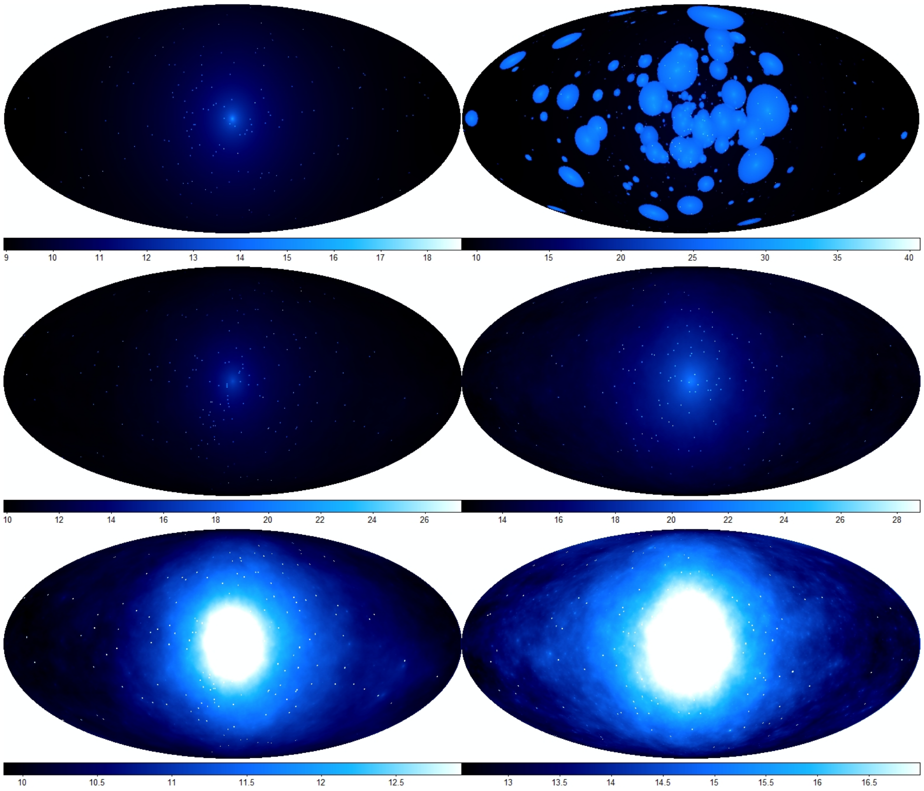

Figure 2: Comparison between two distributions of $N = 10^6$ sub-halos. The projections on the right shows how the sky map transforms when we scale the sub-halos with the correction factor q. The tow pictures are with a lowest mass of $M_{\text{min}} = 10^{-3} M_\odot$ and corresponding scaling factor $q \approx 1.6 \cdot 10^7$. The middle and bottom pictures have a lowest mass of $M_{\text{min}} = 10^3 M_\odot$ with corresponding scaling factor $q = 120$. In the central region you do not see much but at the sides you can see much more structure using the scaling factor. All units are in $\ln \left[J\left(\psi\right) \cdot M_\odot^{-2} \cdot \text{pc}^5 \right]$.

2.8.3 Partial angles

In order to reduce the CPU time we do not want to integrate over the entire sky of very little sub-halo, especially when most of them do not even span more than a few degrees on the sky. Assume a sub-halo with radius R originated at (r, θ, ϕ) on the sky. We are interested in knowing how large part of the sky this halo occupies.

Since the halo is spherical any line that touches the surface will automatically be normal to the orientation of the surface. If we draw a line from the center of the sub-halo to the surface at an angle ψ from the x-axis, this line will have orientation $(R \cos\psi, R \sin\psi)$. Another line, that originates from the center of the origo and touches the surface of the sphere at the same place as the fist line will have orientation $(r\cos\theta + R\cos\psi, r\sin\theta + R\sin\psi)$. Since the two lines are orthogonal their scalar product will be zero.

We can thus write

$$\begin{aligned}
0 &= (r\cos\theta + R\cos\psi, r\sin\theta + R\sin\psi)\cdot(R\cos\psi, R\sin\psi)^T \\
&= rR\cos\theta\cos\psi + R^2\cos^2\psi + rR\sin\theta\sin\psi + R^2\sin^2\psi \\
&= rR\cos(\psi-\theta) + R^2 \\
&= R\left(r\cos(\Delta\psi) + R\right)
\end{aligned}$$

where $\Delta\psi = \psi - \theta$ is related to $\Delta\theta$ by $\Delta\theta = \Delta\psi - \frac{\pi}{2}$ where $\Delta\theta$ is the angular spread in the θ coordinate. If we rewrite $0 = r\cos(\Delta\psi) + R$ and insert $\Delta\theta$ we get

$$\begin{aligned}
-\frac{R}{r} &= \cos\left(\Delta\theta + \frac{\pi}{2}\right) = \cos\Delta\theta\cos\frac{\pi}{2} - \sin\Delta\theta\sin\frac{\pi}{2} \\
&= -\sin\Delta\theta
\end{aligned}$$

with solution

$$\Delta\theta = \arcsin\frac{R}{r}$$

We can use the same argument as above to find the spread in the ϕ variable but this time we will have to scale r by $\sin\theta$ because the effective radius at $z = r\cos\theta$ is $r \rightarrow r_{eff} = r\sin\theta$. With this taken into account the angular spread in the ϕ coordinate is

$$\Delta\phi = \arcsin\frac{R}{r\sin\theta}$$

which means that the higher the elevation above the xy plane the larger the spread in the ϕ variable.

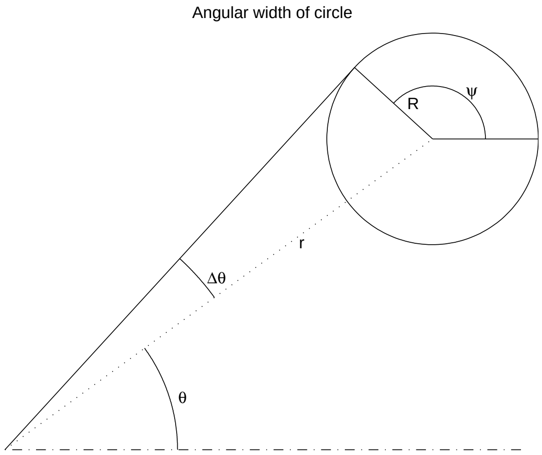

Figure 3: Calculation of partial angles. The principle is the same for both $\Delta\theta$ and $\Delta\phi$ but for $\Delta\phi$ the radial distance is changed as $r \rightarrow r_{eff} = r \sin\theta$.

2.9 Mollweide projections

We have earlier found the form of $J(\psi)$ but the form of it is not the best, since we would like an all sky projection that preserves area and solid angles since the actual area of $J(\psi)$ is proportional to $\sin\theta$. One such projection is the Mollweide projection widely used in astronomy. The projection is onto an xy area of $2l \times 2s$ where

$$\begin{aligned} x &= \frac{2\sqrt{2}}{\pi}\lambda\cos\xi \\ y &= \sqrt{2}\sin\xi \end{aligned}$$

where ξ is an effective latitude which fulfills

$$2\xi + \sin 2\xi = \pi \sin\varphi$$

and φ is the real latitude [15, 12]. If we want to project our function $f(\theta, \phi)$ using Mollweide we first have to transform polar coordinates to latitudes an longitudes. The relation is

$$\begin{aligned} \varphi &= \frac{\pi}{2} - \theta \\ \lambda &= \phi - \pi \end{aligned}$$

We know that the projection is on an ellipsis with width $2\sqrt{2}$ and height $\sqrt{2}$ so the total area is 4π just like the total area of Ω. We can now map a specific point (x, y) onto (θ, ϕ) by

$$\begin{aligned}
\xi &= \arcsin\left(\frac{y}{\sqrt{2}}\right) \\
\varphi &= \arcsin\left(\frac{2\xi}{\pi} + \frac{\sin 2\xi}{\pi}\right) \\
\lambda &= \frac{\pi x}{2\sqrt{2}\cos\xi}
\end{aligned}$$

If we discretize we say that

$$M(x, y) = \begin{cases} f(\theta, \phi) & \text{if } x^2 + 4y^2 \leq 8 \\ 0 & \text{else} \end{cases}$$

Of course, since $f(\psi)$ is only sampled in discrete points the the mapping from x, y to θ, ϕ will not always be perfect. We therefore need to choose how to transform theses points in the $\theta\phi$-grid to points on the Mollweide grid even though these points will not always overlap. There are two simple routines which we will use since we do not a priori know which one is preferable. The first method involves interpolating between points and will be discussed in the next section. The second method is to choose the ψ_{ij} point that is closest to the real point ψ.

2.9.1 Interpolation

As mentioned above when we transform from $\theta\phi$ to Mollweide we will need to make an interpolation between ψ_{ij} points lying around the real point ψ. Assume the we want to know the value of M in the point (x_0, y_0), but we only have the values in $M_{00} = M(0, 0)$, $M_{a0} = (a, 0)$, $M_{0b} = (0, b)$ and $M_{ab} = M(a, b)$. We can now interpolate between the points using the curve $z = A + Bx + Cy + Dxy$. The last term is to make sure the place fitted with the three points also touches the fourth. It is easy to determine the coefficients for A, B, C and D when we solve the equation system

$$\begin{aligned}
A &= M_{00} \\
A + Ba &= M_{a0} \\
A + Cb &= M_{0b} \\
A + Ba + Cb + Dab &= M_{ab}
\end{aligned}$$

with solution

$$\begin{aligned}
A &= M_{00} \\
B &= \frac{M_{a0} - M_{00}}{a} \\
C &= \frac{M_{0b} - M_{00}}{b} \\
D &= \frac{M_{ab} - M_{a0} - M_{0b} - M_{00}}{ab}
\end{aligned}$$

If we rearrange the solutions we get

$$\begin{aligned} z &= M_{00} + \frac{M_{a0} - M_{00}}{a} x + \frac{M_{0b} - M_{00}}{b} y + \frac{M_{ab} - M_{a0} - M_{0b} - M_{00}}{ab} xy \\ &= \frac{M_{00}}{ab} \left(ab - bx - ay - xy\right) + \frac{M_{a0}}{ab} \left(b - y\right) \cdot x + \frac{M_{0b}}{ab} \left(a - x\right) \cdot y + \frac{M_{ab}}{ab} xy \\ &= \frac{1}{ab} \left[M_{ab} \left(ab - bx - ay - xy\right) + M_{a0} \left(b - y\right) x + M_{0b} \left(a - x\right) y + M_{ab} xy\right] \end{aligned}$$

When we change variables so that $x = \theta - \theta_k$ and $y = \phi - \phi_l$ with $a = \Delta\theta$ and $b = \Delta\phi$, we get

$$\begin{aligned} z\left(\theta, \phi\right) &= M_{kl} + \frac{M_{k+1,l} - M_{kl}}{\Delta\theta} \left(\theta - \theta_k\right) + \frac{M_{k,l+1} - M_{kl}}{\Delta\phi} \left(\phi - \phi_l\right) \\ &+ \frac{M_{k+1,l+1} - M_{k+1,l} - M_{k,l+1} - M_{kl}}{\Delta\theta\Delta\phi} \left(\theta - \theta_k\right) \left(\phi - \phi_l\right) \end{aligned}$$

This interpolation cannot be used on the sides and in the corners of the $\theta\phi$-plane. At these places one has to extrapolate beyond the four points lying closest thus changing the lowering or raising the index of either k or l or both by one.

The differences between these two sorts of interpolations are not large but they do exist. Mainly in the distribution of $J\left(\psi\right)$ values. As we can see in figure 4, when we have the skew Mollweide projection we preserve the pixel distribution while this distribution becomes more smeared out when we have the interpolation.

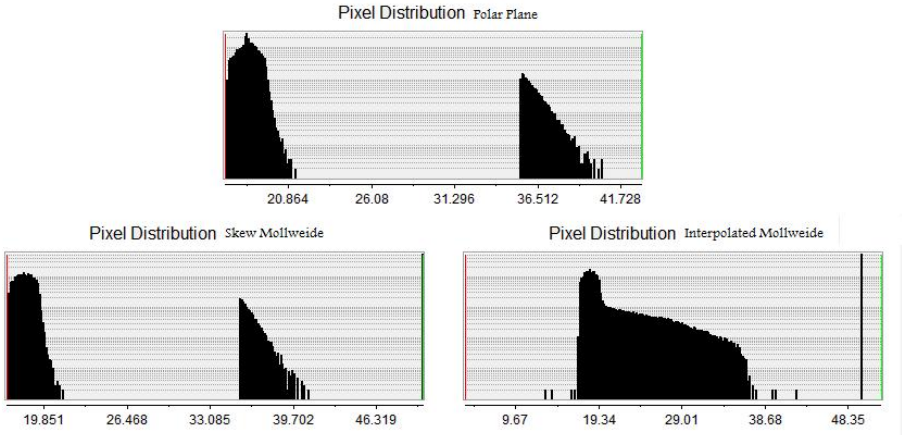

Figure 4: Comparison of how the pixel distribution of $J\left(\psi\right)$ transforms when we go from a $\theta\phi$ representation to a Mollweide projection. This particular distributions is for $N = 10^8$ sub-halos with $M_{\text{min}} = 10^0 M_\odot$. We can see that the skew Mollweide projection preserves the pixel distribution while the interpolated Mollweide projection smears the distribution together.

3 Computational information

The entire program is written in FORTRAN, a language constructed for speed, which requires special care from the programmer. Some of the features of FORTRAN are listed in Appendix C.

3.1 Subroutine breakdown

The code of my program is divided into several subprograms, to make the program more easily navigated. The thought is that you should be able to alter in one subprograms without causing direct errors in another. The subprograms are also written so that you can remove some of them without affecting the possibility of running of the program as a whole. An example is that the contribution to the density from the sub-halos and the host halo are added using different functions. This gives the programmer the option to remove either the lumpy och smooth component of the density by just removing the function that adds the corresponding term.

In figure 5 there is a schematic picture of how all the functions as subroutines are linked together in my program. Single arrows is a function call where the arrow points away from the called function. Double arrows are subroutines where information travels both ways since some subroutines change the value of variables in the main program. Lines without any arrows denote the next stage within the same program/function.

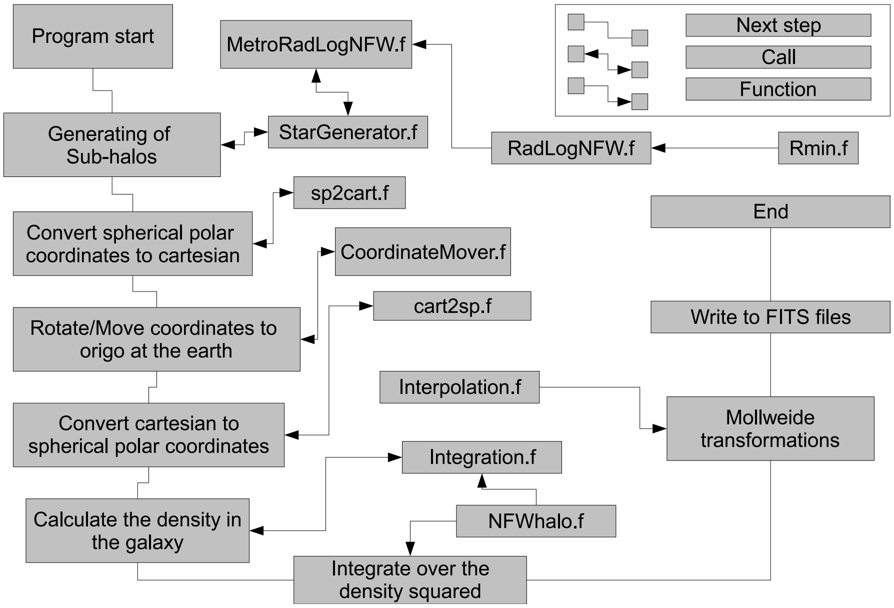
Program start
MetroRadLogNFW.f
Next step
Call
Function
Generating of Sub-halos
StarGenerator.f
RadLogNFW.f
Rmin.f
sp2cart.f
End
Convert spherical polar coordinates to cartesian
CoordinateMover.f
cart2sp.f
Rotate/Move coordinates to origo at the earth
Write to FITS files
Interpolation.f
Convert cartesian to spherical polar coordinates
Mollweide transformations
Integration.f
NFWhalo.f
Calculate the density in the galaxy
Integrate over the density squared

Figure 5: Schematic picture of how my program is build.

The functions and their use are:

- GalaxyHalo: The main program where all the parameters are stored and initiated. If you want to change any parameters, here is the place to do it.
- StarGenerator: A subroutine that generates N sub-halos distributed spherically symmetric and radially according to equation (8) on page 10. Calls MetroRadLogNFW.
- MetroRadLogNFW: Uses the Metropolis algorithm to generate a radial distribution according to (8). The Metropolis density function is calculated in logarithmic units and therefore calls RadLogNFW for the specific value of the density function.
- RadLogNFW: Calculates the specific value of the density function. RadLogNFW is zero whenever $M < M_{\text{min}}$ or $r \notin [R_T(M), R_{\text{max}}]$. Rmin is used to know when $r < R_T(M)$.
- Rmin: A function calculating $\Delta F = F_H - F_T(M)$. Whenever ΔF is negative then $r < R_T(M)$ and RadLogNFW is zero.
- sp2cart: Subroutine transforming spherical polar coordinates to Cartesian coordinates.
- CoordinateMover: Transforms the Cartesian coordinates from being centered at the galaxy center to having their origo at the Earth.
- cart2sp: Subroutine transformation Cartesian coordinates back to spherical polar coordinates.
- Integration: Adds all the sub-halo contributions (except the host halo contribution which is separately added) to the density. The density contribution is calculated using NFWhalo.
- NFWhalo: Function that return the density at distance r from the sub-halo center.
- Interpolations: A function that interpolates between the discrete points in the $\theta\phi$-plane sky map when you transform the to a Mollweide projection.

In my program I have limited the use of global variables to just a few, in order to prevent mix ups. Instead my functions and subroutines have an extensive number of arguments in order transport the information.

3.2 Some interesting tricks

Sometimes the algebra described in the previous sections are not directly translatable to program code. Sometimes the naive implementation will result in huge loss of efficiency. In these cases you have to tweak the formulas or rewrite the program in such a way that you save CPU time or memory.

3.2.1 Integrating over the sub-halos

In the original formulation of the calculation of the density in the galaxy we had (17) where we first sum over space and the over the sub-halos. This equation is fine, but often $\rho_i\left(|\mathbf{r}_{jkl} - \mathbf{r}_i|\right) = 0$, which happens whenever $|\mathbf{r}_{jkl} - \mathbf{r}_i| > r_{\mathrm{vir}}^{(i)}$ where $r_{\mathrm{vir}}^{(i)}$ is the virial radius of the ith sub-halo. This means that many of the iteration are really unnecessary. To save CPU-time I reorganized the sums so that the contributions for every single sub-halo are added separately.

3.2.2 Dividing the volume

When you calculate the volume in which the sub-halo exist, you sometimes find yourself in a position where parts of the ϕ interval lies outside of the $[0, 2\pi)$ range. Because of periodicity this means that you really have a sub-halo in the region $[0, \alpha] \cup [\beta, 2\pi)$. The FORTRAN for-loop does only allow you to have evenly spaced iterations. The naive solution to this problem is to say that the sub-halo occupies the entire range $[0, 2\pi)$ and simply iterate over it. This procedure however requires a lot of CPU time since we will need to iterate over large parts of the sky, especially if $\alpha + (2\pi - \beta)$ is much smaller than 2π.

What we can do to improve performance is that instead of having one for-loop that runs from 0 to 2π, we use two loops that run over $[0, \alpha]$ and $[\beta, 2\pi)$ respectively. If all the sub-halos had $d\phi = \alpha + (2\pi - \beta)$ the quotient of CPU-time gained can sometimes approach $\frac{2\pi}{d\phi}$. The improvement did still speed things up by a factor of at least 10.

3.3 The FITS file format

The dominating file format for astronomers and other people staring at the sky is the FITS file format. Invented in the 70's this file format is good for storing both images and information regarding the images. The form of FITS is rather strict which makes it hard to write FITS files by hand. If you are a FORTRAN user, you can in principle print to a FITS file row by row but it is cumbersome to say the least. We can however use a library package originally intended for C programming but altered to fit FORTRAN. The package in question is called CFITSIO and is not a standard package in most FORTRAN compilers [6].

All the sky maps made in this thesis are stored in FITS format [8].

4 Results and Discussion

What we get out of our simulations is, as we have said earlier, sky maps that show the luminosity density for the θ and ϕ coordinates. These maps, as we can see in figure 7 on page 32, show that the highest density is in the region nearest the center of the galaxy and decreases at higher distances. What we would like to do is to compare our DM luminosity sky maps with the cosmic background spectrum. Such a spectrum can "easily" be generated using the GALPROP package [13].

There will however not be time enough for me to do this comparison in this thesis, and so I will only lay down the basics for such a comparison.

4.1 The Parameters

The program I have developed is written in such a way that you can easily change the major aspects of the simulation like

- N: The number of sub-halos in the system
- N_θ, N_ϕ: The angular resolution of the simulation. Changing either N_θ or N_ϕ alters the size of the sky map since these parameters define the number of pixels in the FITS file.
- N_r: Related to the radial resolution, Δr by $N_r \cdot \Delta r = r_{\text{max}}$. Usually I choose $r_{\text{max}} = 2r_{\text{vir}}$ where r_{vir} is the viral radius of the host halo.
- M_H: The mass of the host halo, in units of solar masses ($M_\odot$). The mass of the Milky Way is roughly $10^{12} \times M_\odot$ [10].
- p: The percentage of the total mass that is located in sub-halos. If nothing else is stated the choice is $p = 0.15$.
- r_E: The radius at which the Earth is circulating. In the simulations this is chosen to be 7660 parsec[5].
- r_{vir}: The virial radius of the Milky Way. With the definition of virial radius we have $r_{\text{vir}} \approx 2 \cdot 10^5$ parsec. This is a rather large number since many sources say that the radius of the Galaxy is $3 \cdot 10^4$ parsec and the radius of the halo is around $6 \cdot 10^4$ parsec [10].

Other parameters that are not as easy to change but can be altered on a global scale is

- k, ρ_0: In the definition of viral radius the critical density of the Universe enters, as well as the quotient between the mean density and the critical density. Changing any of these two would yield effects in the virial radius. I have always used $k = 200$ and $\rho_0 = 10^{-29} \times \text{g/cm}^3 = 1.477 \times 10^{-7} \times M_\odot/\text{pc}^3$.

[5] The Earth is at approximately 25 000 light years making 7660 parsec. Sources are not in agreement to late i found a more reliable distance of 8.5 kpc[10].

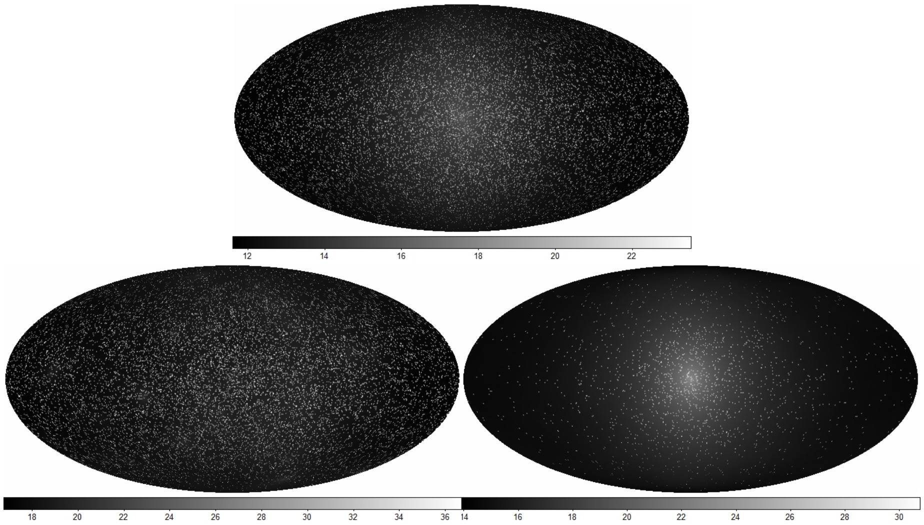

Figure 6: Comparative sky maps of $M_{\min}$ with 10^{-3} (top), 10^{0} (bottom left) and 10^{3} (bottom right) solar masses, $M_{\odot}$. All figures uses a constellation with $N = 10^6$, $N_\phi = 720$, $N_\theta = 360$ and $N_r = 400$. Many more star are visible then we lower $M_{\min}$ but this has partly to do with the scaling factor boosting these stars. The scales are as usual in $\ln\left[J\left(\psi\right) \cdot M_{\odot}^{-2} \cdot \mathrm{pc}^5\right]$.

4.2 Output

One of the questions, which we would like to have answered, is what happens when you scale down the size of the smallest sub-halo. Several runs have been made to get a feeling for what happens when you allow for smaller sub-halos. Some of the results can bee seen in figure 6. The conclusion we can draw is that when you lower the minimum mass the NFW profile becomes more visible. This has several reasons. When $M_{\min}$ is large only heavy sub-halos are allowed and these are, because of tidal forces, condemned to be in the outer regions of the galaxy. This increases the chances of the Earth being inside the cloud of sub-halos. The luminosity density will thus be more even over the sky, just like when you are inside of a cloud of fog. When the minimum mass, $M_{\min}$, is small most sub-halos are never observed, the are simply to faint and small. In this case the luminosity from the cost halo will be the biggest contributor.

The fact that you do not see most sub-halos because they fall between different evaluation points is problematic. You can make a rough estimate about how many sub-halos will be detected with a specific number of evaluation points. For instance a sub-halo with radius R at a distance d from the coordinate system origo will have a chance of approximately $\frac{\frac{4}{3}\pi R^3}{\Delta V}$ to be counted. Here $\Delta V = d^2 \sin\theta \cdot \Delta\lambda \cdot \Delta\phi \cdot \Delta\theta$ is the volume spanned by the integration points at radius d and elevation θ. Spheres with a radius larger than the longest dimension of ΔV will always be detected.

The smooth background component should look something like figures 11 on page 47.

The units used in the calculations are mainly parsec and solar masses. Thus the dimension of ρ is $[M_\odot/\mathrm{pc}^3]$. With $\Delta\lambda$ having the dimension of [pc] the dimension of $J(\psi)$ is $[\mathrm{pc} \times \left(M_\odot/\mathrm{pc}^3\right)^2]$=$[M_\odot^2/\mathrm{pc}^5]$. Since the scale of $J(\psi)$ ranges over several orders of magnitude all sky maps have values of $\ln[T(\psi)]$, where $T(\psi) = J(\psi) \cdot M_\odot^{-2} \cdot \mathrm{pc}^5$

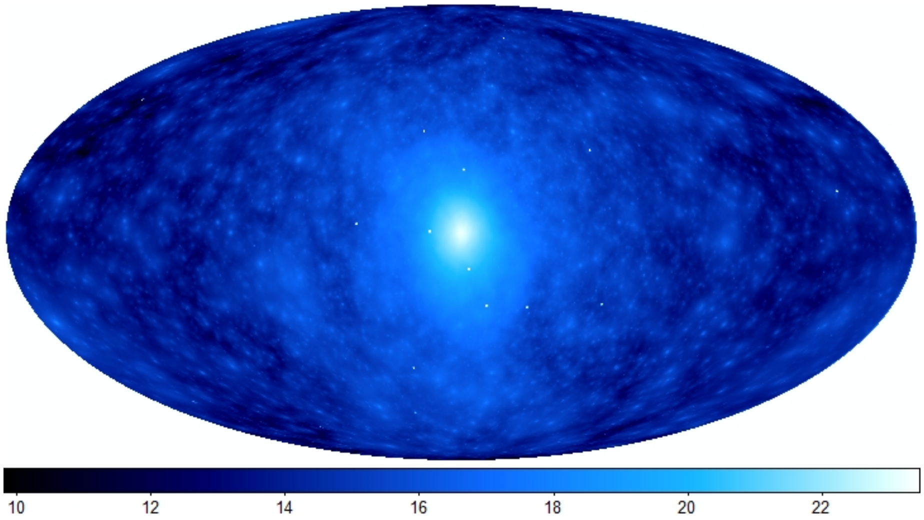

Figure 7: Sky map of $\ln[J(\psi)]$ as seen from the Earth. We can see lots of structure and identify several satellites. The q-factor in this picture is ≈ 200 which boosts the structure, but the same type of structure can bee seen in sky maps where the q-factor is approximately unity. The scale is logarithmic of $J(\psi)$, namely $\ln\left[J(\psi) \cdot M_\odot^{-2} \cdot \mathrm{pc}^5\right]$.

For flexibility in the analysis GalaxyHalo has several different sky maps in its output. You have two $\theta\phi$-maps showing $T(\psi)$ and $\ln[T(\psi)]$, as well as Mollweide projections of both $T(\psi)$ and $\ln[T(\psi)]$. The Mollweide projections are made in two copies using different techniques for coordinate transforms, one with interpolated values and one where the nearest pixel point is choosen as described in section 2.9.1.

4.3 Comparison with cosmic radiation

If the only visible thing in the Universe was the DM-annihilation luminosity people around the world would not be sitting thinking about if DM-annihilation can actually bee seen. However DM-annihilations have to outshine several other sources of radiation in the universe to be able to be detected. What we would like to do is to compare the luminosity spectra created by us with known galactic spectra or models for galactic spectra.

The main idea is that if you have a model for the spectral background and add it to the DM-annihilation spectra we can see if any DM-clumps are visible against the background. One package that generates spectral data is the GALPROP package by Strong et al. [13].

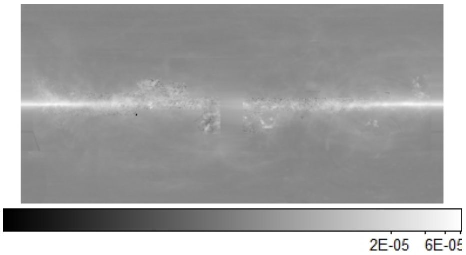

Figure 8: GALPROP generated picture of Bremsstrahlung. Note that the center of the galaxy is located at $\phi = 0$.

It is important to note that a DM-clump not only has to outshine other galactic sources but also has to shine stronger than the smooth component. The latter task might sometimes be hard enough as Pieri et. al. saw for some configurations of DM distributions [11].

However, whether or not the cosmic background or the smooth component is the dominating we will still need some way of determining whether or not we can see any lumps. This business with finding lups is not an easy task. There are some algorithms for identifying stars and such like the "Friend of Friend" algorithm used to identify stars in the Via Lactea simulations [2].

Other approaches might be to, in a small region of space, fit the spectral luminosity to some model describing a small part of space. Then the deviations by experimental data from the theoretic curve can be compared to look for large deviations.

4.3.1 Error propagation

No matter which method we choose to make our analysis, the error analysis are much the same. Assume that we in energy bin E_k have found a potential DM-annihilations signal at θ_i, ϕ_j, denoted D_{ijk}. The corresponding theoretical value from some statistical analysis is called G_{ijk}.

We would like to know what a GLAST like experiment would be able to detect if we where to make an all sky survey over a period of time. The number of photons a detector will see is related amongst other things the area and effectiveness of the detector and the particle physics specifics as written in equation (14) on page 16.

The variables sD_{ijk} and G_{ijk} thus describes the detection rate of photons and we assume that D_{ijk} and G_{ijk} are in units [counts/time interval]. We can assume that both D and G[6] are Poisson distributed and therefore they have an error of $\sigma_D = \sqrt{D}$ and $\sigma_G = \sqrt{G}$.

Our null hypothesis is that there is no difference between D and G, meaning that $D = G$. We can reformulate the problem as $\Delta = D - G = 0$. The error of Δ is $\sigma_\Delta^2 = \sigma_D^2 + \sigma_G^2 = D + G$. The t value, which is the number is standard deviations that Δ is from 0 is $t = \frac{\Delta}{\sigma_\Delta}$.

Since we are not making a focused search it would be possible to get a high t value even if the true value of D really is G. We do, of course, want to minimize the false positives of this kind. It is customary in these kinds of searches to choose that the threshold t is 5. It is sometimes referred to as a 5σ search.

If we insert the 5σ limit in the equation for t and we get $5 \leq \frac{D-G}{\sqrt{D+G}}$. We can assume that $D = kG$ to get a feeling for how much larger than G, D has to be to find a 5σ abnormality. Rewriting we get

$$\begin{aligned} t^2 (k+1) G &= (k-1)^2 G^2 \\ &\Downarrow \\ k &= 1 + \frac{q}{2} \pm \sqrt{2q + \frac{q^2}{4}} \end{aligned}$$

where $q = \frac{t^2}{G}$ which means that $q \to 0$ (∞) when $G \to \infty$ (0). From the last line, we can see that when G is large then D does not have to be significantly larger than G to be detected. When G decreases then k increases almost linearly with G^{-1}.

From this we can draw the conclusion that the quality of the detector as well as the particle physical properties are of paramount importance for the detection of DM-annihilations.

4.4 Outlook

Since this paper is only a master thesis and time prevents me from doing everything I have chosen to focus on the problem of generating DM-distributions and the necessary algorithms for that purpose. It is my hope that GalaxyHalo will be easy to use and that someone will continue to build on it. There are still many features that needs to be added and/or improved. For instance now all clumps have the same concentration, something you would like to be related to mass. Another important improvement would be to give the option of how you want the scaling of the sub-halos to be done. Right now you can choose whether or not to have scaling but you cannot choose to have scaling of the viral mass instead of having scaling of the density as mentioned in section 2.8.2.

In the present form of GalaxyHalo you can make a complete simulation of a galaxy with arbitrary number of sub-halos at any mass range. You could therefore compute $J(\psi)$ for a galaxy with mass $M = 10^{12} M_\odot$ and sub-halo mass $M_{\text{min}} = 10^{-6} M_\odot$. This would require, with a percentage of $p = 0.15$, between approximately $10^{11} - 10^{13}$ sub-halos at the present configuration of parameters.

[6] We remove the indexes from now, whey will not add anything to the analysis.

The task is not impossible for me, since using the GLAST machines it would take $2 \cdot 10^7$ seconds or 231 days which is a little to long to wait. No systematic efforts have been made to minimize the run time nor has compilations been made with flags that mark for speed.

One feature that has to be corrected before GalaxyHalo can be really user friendly is to make all the parameters easily accessible without having to make changes within the FORTRAN code. As it stands now the parameters are accessible within the "head" of the "main" but you still need to recompile everything whenever you change a parameter.

References

[1] L. Bergström and A. Goobar. *Cosmology and Particle Astrophysics.* Springer, Berlin, Germany; New York, U.S.A., 2nd edition, 2004.

[2] Jurg Diemand, Michael Kuhlen, and Piero Madau. Early supersymmetric cold dark matter substructure. *Astrophys. J.*, 649:1–13, 2006.

[3] Jurg Diemand, Michael Kuhlen, and Piero Madau. Dark matter substructure and gamma-ray annihilation in the milky way halo. *Astrophys. J.*, 657:262, 2007.

[4] P. Madau, J. Diemand, and M. Kuhlen. Dark matter subhalos and the dwarf satellites of the milky way. *Astrophys. J.*, 679:1260–1271, June 2008.

[5] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, 21:1087–1092, June 1953.

[6] NASA. Cfitsio - a fits file subroutine library. HEASARC: http://heasarc.gsfc.nasa.gov/docs/software/fitsio/fitsio.html, 07 2008.

[7] NASA. Nasa's glast launch successful. NASA press release: http://www.nasa.gov/news/releases/archives/index.html, 11/6 2008.

[8] NASA/GSFC. Definition of the flexible image transport system (fits). The FITS Support Office: http://fits.gsfc.nasa.gov/, 07 2008.

[9] Julio F. Navarro, Carlos S. Frenk, and Simon D. M. White. The structure of cold dark matter halos. *Astrophys. J.*, 462:563–575, 1996.

[10] Carl Nordling and Jonny Österman. *Physics Handbook For Science And Engineering.* Studentlitteratur AB, Lund, 2004.

[11] L. Pieri, G. Bertone, and E Branchini. Dark matter annihilation in substructures revised. *Mon. Not. Roy. Astron. Soc.*, 384:1627, 2008.

[12] John Parr Snyder. *Flattening the Earth: Two Thousand Years of Map Projections.* The University of Chicago Press, 1997.

[13] A. W. Strong and I. V. Moskalenko. Models for galactic cosmic-ray propagation. *Adv. Space Res.*, 27:717–726, 2001.

[14] Wikipedia. Metropolis - hastings algorithm. On-Line Encyclopedia: http://www.wikipedia.org/, 06 2008.

[15] Wikipedia. Mollweide projection. On-Line Encyclopedia: http://www.wikipedia.org/, 06 2008.

[16] Wikipedia. Roche limit. On-Line Encyclopedia: http://www.wikipedia.org/, 06 2008.

[17] F. Zwicky. Die rotverschiebung von extragalaktischen nebeln. *Helvetica Physica Acta*, 6:110–127, 1933.

[18] F. Zwicky. On the masses of nebulae and of clusters of nebulae. *Astrophys. J.*, 86:217–+, October 1937.

A The Normalization

This chapter is about how and why we cannot find the normalization A for equation (8) on page 10. To find the normalization we need to evaluate the integral

$$\begin{aligned}
1 &= \int\int AM^{-2}\frac{\Theta\left(r-R_T\left(M\right)\right)}{\frac{r}{r_{MW}}\left(1+\frac{r}{r_{MW}}\right)^2}\,dM\,d^3r \\
&= 4\pi A\int_{M_{\min}}^{\infty}\left(\int_{R_T(M)}^{r_{\max}}\frac{r^2M^{-2}}{\frac{r}{r_{MW}}\left(1+\frac{r}{r_{MW}}\right)^2}\,dr\right)\,dM \\
&= A\int_{M_{\min}}^{\infty}M^{-2}\left[\frac{r_{MW}}{r_{MW}+r}+\log\left(r_{MW}+r\right)\right]_{R_T(M)}^{r_{\max}}\,dM \\
&= A\int_{M_{\min}}^{\infty}M^{-2}f\left(M\right)\,dM
\end{aligned}$$

where

$$f\left(M\right)=\frac{r_{MW}}{r_{MW}+r_{\max}}+\log\left(r_{MW}+r_{\max}\right)-\frac{r_{MW}}{r_{MW}+R_T\left(M\right)}-\log\left(r_{MW}+R_T\left(M\right)\right)$$

If we had the explicit form of $R_T\left(M\right)$ we might be able to solve the integral above.

A.1 Calculating $R_T\left(M\right)$

Remember the Roche criterion for tidal forces acting on a sub-halo. The equation of interest is

$$\frac{M_H\left(d-r\right)}{d^3} \quad < \quad \frac{4\pi}{3}k\rho_0\equiv C_0$$

Here we skip the index "vir" appearing in equation (12) since r is the only radius other than that of d that we are talking about. Remember that r really is a function of M defined through (5). It is good that the left hand side is a constant sine if we wanted to numerically evaluate the solution d_r in the region where $r\ll d$ we could calculate d_r from d_0 which we only need to calculate once.

If $M_H\left(d-r\right)$ would have been $M\left(d\right)$, then d_r would be independent of r. This is not the case, however since $r<d$, and often $r\ll d$, we can make a Taylor expansion around $M_H\left(d\right)$. This expansion gives $M_H\left(d-r\right)\approx M_H\left(d\right)-M'_H\left(d\right)\cdot r$. From (2) on page 7 we get

$$M'_H\left(d\right)=4\pi\frac{r_{MW}^3\rho_{MW}d}{\left(r_{MW}+d\right)^2}$$

which leads to the inequalities

$$4\pi r^3_{s_{MW}}\rho_{s_{MW}}\left[\ln\left(1+\frac{d}{r_{MW}}\right)-\frac{d}{r_{MW}+d}\right]-4\pi\frac{r^3_{MW}\rho_{s_{MW}}d}{\left(r_{MW}+d\right)^2}r \quad < \quad \frac{4\pi}{3}k\rho_0$$

or

$$\frac{1}{d^3}\left\{\ln\left(1+\frac{d}{r_{MW}}\right)-\frac{d}{r_{MW}+d}-\frac{rd}{\left(r_{MW}+d\right)^2}\right\} \quad < \quad \frac{k\rho_0}{3r^3_{MW}\rho_{MW}} \qquad (20)$$

Now we want to find the solution to (20). We can use a kind of linear approximation which will give us a good result when $r \ll d$.

If we define $c \equiv \frac{k\rho_0}{3r^3_{s_{MW}}\rho_{s_{MW}}}$ and $a\left(d\right) \equiv \ln\left(1+\frac{d}{r_{s_{MW}}}\right)-\frac{d}{r_{s_{MW}}+d}$ so $a'\left(d\right) = \frac{d}{\left(r_{s_{MW}}+d\right)^2}$, we can construct the functions $f\left(d\right) = \frac{a(d)}{d^3} - c$ and $g\left(d\right) = \frac{1}{d^3}\left\{a\left(d\right)-a'\left(d\right)r\right\} - c = f\left(d\right) - \frac{1}{d^3}\left\{a'\left(d\right)r\right\}$. The function $f\left(d\right)$ is what you get when $M_H\left(d-r\right) \rightarrow M_H\left(d\right)$. We first find d_0, the solution to $f\left(d_0\right) = 0$. This will have to be done numerically, using any of the well known formulas for numerical calculations like Newton-Raphsons method[7].

Since $r \ll d$, $f\left(d_0\right) \approx g\left(d_0\right)$. We can extrapolate the straight line of the tangent to $g\left(d_0\right)$ like $y = kd + m$ where $k = g'\left(d_0\right)$ and $m = g\left(d_0\right) - g'\left(d_0\right)d_0$. The solution, $d_r$to $kd_r + m = 0$ is

$$d_r = -\frac{m}{k} = \frac{g'_0 d_0 - g_0}{g'_0} = d_0 - \frac{g_0}{g'_0}$$

where $g_0 \equiv g\left(d_0\right)$ and $g'_0 \equiv g'\left(d_0\right)$. From the definition of d_0 and g_0 we can immediately see that

$$g_0 = \frac{1}{d^3}\left\{a\left(d_0\right)-a'\left(d_0\right)r\right\} - c = f\left(d_0\right) - \frac{a'\left(d_0\right)r}{d^3} = -\frac{a'\left(d_0\right)r}{d^3}$$

The value of g'_0 is

$$\begin{aligned}
g'\left(d\right) &= \frac{\partial}{\partial d}\frac{1}{d^3}\left\{a\left(d\right)-a'\left(d\right)r\right\} \\
&= \frac{1}{d^3}\frac{\partial}{\partial d}\left\{a\left(d\right)-a'\left(d\right)r\right\} + \left(\frac{\partial}{\partial d}\frac{1}{d^3}\right)\left\{a\left(d\right)-a'\left(d\right)r\right\} \\
&= \frac{1}{d^3}\left\{a'\left(d\right)-a''\left(d\right)r\right\} + \left(-3\frac{1}{d^4}\right)\left\{a\left(d\right)-a'\left(d\right)r\right\} \\
&= \frac{1}{d^3}\left[a'\left(d\right)-a''\left(d\right)r-\frac{3}{d}a\left(d\right)+\frac{3}{d}a'\left(d\right)r\right] \\
&= \frac{1}{d^3}\left[a'\left(d\right)\left\{1+\frac{3r}{d}\right\}-a''\left(d\right)r-\frac{3}{d}a\left(d\right)\right]
\end{aligned}$$

[7] Because of the shape of $f\left(d\right)$ Newton-Raphson is very sensitive to the initial guess. The method is divergent except in a small area around d_0. A more robust but slower approach would be the to use the secant formula.

The quotient between g_0 and g_0' becomes

$$\begin{aligned} \frac{g_0}{g_0'} &= \frac{-a'(d_0)\, r}{a'(d_0)\left\{1+\frac{3r}{d_0}\right\} - a''(d_0)\, r - \frac{3}{d_0} a(d_0)} \\ &= -\frac{r}{1+\frac{3r}{d_0} - \frac{a''(d_0)}{a'(d_0)} r - \frac{3}{d_0}\frac{a(d_0)}{a'(d_0)}} \\ &= -\frac{r}{\left\{1-\frac{3}{d_0}\frac{a(d_0)}{a'(d_0)}\right\} + r\left\{\frac{3}{d_0} - \frac{a''(d_0)}{a'(d_0)}\right\}} \end{aligned} \tag{21}$$

Now we remember, that if you have $\frac{r}{a+br}$ you can Taylor expand it as $\frac{r}{a+br} = \frac{r}{a}\sum_{n=0}\left(-\frac{br}{a}\right)^n = \frac{r}{a} - \frac{br^2}{a^2} + \frac{b^2r^3}{a^3} + \cdots$ and so to first power we have

$$\frac{g_0}{g_0'} \approx -\frac{r}{1-\frac{3}{d_0}\frac{a(d_0)}{a'(d_0)}}$$

Thus our value of the Roche distance is

$$d_1 \approx d_0 - \frac{g_0}{g_0'} \approx d_0 + \frac{r}{1-\frac{3}{d_0}\frac{a(d_0)}{a'(d_0)}} \tag{22}$$

This is good because we only need to calculate d_0 and $1-\frac{3}{d_0}\frac{a(d_0)}{a'(d_0)}$ at one time and can be used for all small values of r. Of course we could have used (21), but (22) is much simpler and can be used when we want to calculate the normalization.

A.2 Returning to normalization

Now we return to the normalization of (8) where we have an explicit form of $R_T(M) = d_0 + \frac{r_{\text{vir}}}{1-\frac{3}{d_0}\frac{a(d_0)}{a'(d_0)}} = d_0 + \alpha r_{\text{vir}}$ which gives

$$\begin{aligned} 1 &= 4\pi A \int_{M_{\min}}^{\infty} M^{-2} \cdot \left[\gamma - \frac{r_{MW}}{r_{MW} + R_T(M)} - \log\left(r_{MW} + R_T(M)\right)\right] dM \\ & 4\pi A \int_{M_{\min}}^{\infty} M^{-2} \cdot \left[\gamma - \frac{r_{MW}}{\beta + \alpha r(M)} - \log\left(\beta + \alpha r(M)\right)\right] dM \end{aligned}$$

We can change variables from M to r to make the integral easier to solve. Using that $M = \eta r^3 \Rightarrow dM = 3\eta r^2\, dr$ we get s

$$\begin{aligned} 1 &= 4\pi A \int_{r_{\text{vir}_{\min}}}^{r_{\text{vir}_{\max}}} \frac{3\eta r^2}{\eta^2 r^6} \cdot \left[\gamma - \frac{r_{MW}}{\beta+\alpha r} - \log(\beta+\alpha r)\right] dr \\ &= \frac{12\pi A}{\eta} \int_{r_{\text{vir}_{\min}}}^{r_{\text{vir}_{\max}}} \frac{1}{r^4} \cdot \left[\gamma - \frac{r_{MW}}{\beta+\alpha r} - \log(\beta+\alpha r)\right] dr \\ &= \delta \int_{r_{\text{vir}_{\min}}}^{r_{\text{vir}_{\max}}} \frac{1}{r^4} \cdot \left[\gamma - \frac{r_{MW}}{\beta+\alpha r} - \log(\beta+\alpha r)\right] dr \end{aligned}$$

where $\delta = \frac{12\pi A}{\eta} = \frac{9A}{k\rho_0}$, $\gamma = \frac{r_{s_{MW}}}{r_{s_{MW}} + r_{\mathrm{vir}_{MW}}} + \log\left(r_{s_{MW}} + r_{\mathrm{vir}_{MW}}\right)$ and $\beta = r_{s_{MW}} + d_0$. We get three integrals $1 = I_1 + I_2 + I_3$ that are

$$\begin{aligned}
I_1 &= \delta \int_{r_0}^{\infty} \frac{\gamma}{r^4}\, dr \\
I_2 &= \delta \int_{r_0}^{\infty} \frac{1}{r^4} \frac{r_{MW}}{\beta + \alpha r}\, dr \\
I_3 &= \delta \int_{r_0}^{\infty} \frac{1}{r^4} \log\left(\beta + \alpha r\right)\, d
\end{aligned}$$

The first integral is

$$I_1 = \delta\gamma \left[\frac{3}{r^3}\right]_{\infty}^{r_0} = \frac{3\gamma\delta}{r_0^3}$$

The second is

$$\begin{aligned}
I_2 &= r_{MW}\delta \left[-\frac{\alpha^2}{\beta^3 r} + \frac{\alpha}{2\beta^2 r^2} - \frac{1}{3\beta r^3} + \frac{\alpha^3}{\beta^4} \ln\left(\frac{\beta}{r} + \alpha\right)\right]_{r_0}^{\infty} = \\
&= r_{MW}\delta \left(\frac{\alpha^2}{\beta^3 r_0} - \frac{\alpha}{2\beta^2 r_0^2} + \frac{1}{3\beta r_0^3} - \frac{\alpha^3}{\beta^4} \ln\left(\frac{\beta}{r_0} + \alpha\right) + \frac{\alpha^3}{\beta^4} \ln\alpha\right) \\
& \quad r_{MW}\delta \left(\frac{\alpha^2}{\beta^3 r_0} - \frac{\alpha}{2\beta^2 r_0^2} + \frac{1}{3\beta r_0^3} - \frac{\alpha^3}{\beta^4} \ln\left(\frac{\beta}{\alpha r_0} + 1\right)\right)
\end{aligned}$$

The third integral is

$$\begin{aligned}
I_3 &= \delta \left[-\frac{\alpha}{6\beta r^2} + \frac{\alpha^2}{3\beta^2 r} - \frac{1}{3r^3} \ln\left(\beta + \alpha r\right) - \frac{\alpha^3}{3\beta^3} \ln\left(\frac{\beta}{\alpha r} + 1\right)\right]_{r_0}^{\infty} \\
&= \delta \left(\frac{\alpha}{6\beta r_0^2} - \frac{\alpha^2}{3\beta^2 r_0} + \frac{1}{3r_0^3} \ln\left(\beta + \alpha r_0\right) + \frac{\alpha^3}{3\beta^3} \ln\left(\frac{\beta}{\alpha r_0} + 1\right)\right)
\end{aligned}$$

By putting this all together and letting $q = \frac{\alpha}{\beta}$ we get

$$\begin{aligned}
1 &= \delta \left\{\frac{3\gamma}{r_0^3} + r_{MW} \left(\frac{q^2}{\beta r_0} - \frac{q}{2\beta r_0^2} + \frac{1}{3\beta r_0^3} - \frac{q^3}{\beta} \ln\left(\frac{1}{q r_0} + 1\right)\right)\right. \\
& \quad \left. + \left(\frac{q}{6r_0^2} - \frac{q^2}{3r_0} + \frac{1}{3r_0^3} \ln\left(\beta + \alpha r_0\right) + \frac{q^3}{3} \ln\left(\frac{1}{q r_0} + 1\right)\right)\right\} \\
&= \delta \left\{q^3 \left(\frac{1}{3} - \frac{r_{MW}}{\beta}\right) \ln\left(\frac{1}{q r_0} + 1\right) + \frac{q^2}{r_0} \left(\frac{r_{MW}}{\beta} - \frac{1}{3}\right)\right. \\
& \quad \left. + \frac{q}{r_0^2} \left(\frac{1}{6} - \frac{r_{MW}}{2\beta}\right) + \frac{1}{3r_0^3} \left(9\gamma + \frac{r_{MW}}{\beta} + \ln\left(\beta + \alpha r_0\right)\right)\right\}
\end{aligned}$$

Since $\delta = \frac{9A}{k\rho_0}$ it is easy to find the normalization from here. Using this normalization we could be interested in calculating properties of the distribution such as the expectation of the mass. We can use the ordinary formula for expectation values $\langle f(x) \rangle = \int f(x) \rho(x)\ dx$ and get

$$\begin{aligned}
\langle M \rangle &= 4\pi A \int_{M_{\min}}^{\infty} M \cdot M^{-2} \cdot \left[\gamma - \frac{r_{MW}}{\beta + \alpha r} - \log(\beta + \alpha r) \right] dM \\
&= 12\pi A \int_{r_0}^{\infty} \frac{1}{r} \cdot \left[\gamma - \frac{r_{MW}}{\beta + \alpha r} - \log(\beta + \alpha r) \right] dr \\
&= \lambda \int_{r_0}^{\infty} \frac{1}{r} \cdot \left[\gamma - \frac{r_{MW}}{\beta + \alpha r} - \log(\beta + \alpha r) \right] dr
\end{aligned}$$

Here, the third term poses a bit of a problem since

$$\lambda \int_{r_0}^{\infty} \frac{1}{r} \log(\beta + \alpha r)\ dr = \lambda \left[\operatorname{dilog}\left(-\frac{\alpha r}{\beta} \right) + \ln(\beta + \alpha r) \ln\left(-\frac{\alpha r}{\beta} \right) \right]_{r_0}^{\infty}$$

where $\operatorname{dilog}(x)$ is the dilogarithm function

$$\operatorname{dilog}(x) = \int_1^x \frac{\ln t}{1-t}\, dt$$

You might say that the dilogarithm function have made us run into a brick wall. Even if we could evaluate it numerically, it still is not defined for negative arguments, which is just what we have since α, β and r are all positive.

B Polar transformations

When we have generated our sub-halos, we have the problem that they are in polar coordinates with origo at the center of the galaxy. What we would like, is to transform to polar coordinates around the Earth, and further more orient the z- axis of the Earth so that it is pointing towards the center of the Galaxy. Another orientation would be to point it perpendicular to the origin of the galaxy and let the x axis point away from the center of the galaxy.

Whichever orientation of our coordinate system we choose, the general procedure is as follows. First we transform our polar coordinates $(r, \theta, \phi)_C$ to their Cartesian counterparts $(x, y, z)_C$ via the usual

$$\begin{aligned}
x &= r \sin\theta \cos\phi \\
y &= r \sin\theta \sin\phi \\
z &= r \cos\theta
\end{aligned}$$

The next procedure is to move the origo to the center of the Earth $(x_E, y_E, z_E)_C = (0,0,0)_E$ and rotate it so that $(0,0,0)_C = (x_C, y_C, z_C)_E$. In our case, we are interested in $(0,0,0)_C = (0,0,R_C)_E$ or $(0,0,0)_C = (\pm R_C, 0, 0)_E$ depending on the preferred orientation of our coordinate system. To change the coordinate system like this, we use matrix algebra and the transformation $\mathbf{r}' = R \cdot \mathbf{r} + \mathbf{C}$

where $\mathbf{r}'$ is the coordinates of the Earth, $\mathbf{r}$ is the coordinates from the center of the galaxy, R is a rotation matrix and $\mathbf{C}$ is a vector constant.

When the new Cartesian coordinates are in place, we can calculate the new polar coordinates as

$$\begin{aligned} r' &= \sqrt{x'^2 + y'^2 + z'^2} \\ \theta' &= \arccos\left(\frac{z'}{r'}\right) \\ \phi' &= \arctan\left(\frac{y'}{x'}\right) + \Theta\left(-x'\right)\pi \end{aligned}$$

where $\Theta\left(-x'\right)\pi$ adds a π whenever x' is negative.

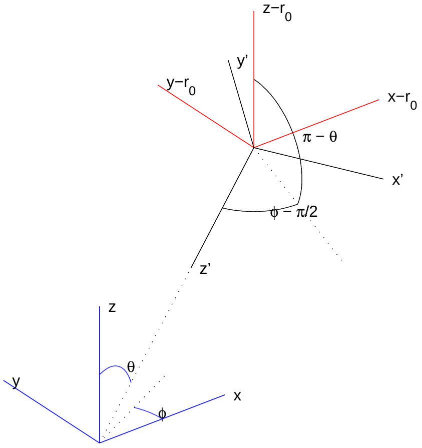

Figure 9: A coordinate transform where the new axes are $\overline{r}'$.

B.1 The z'-axis on center

Let us first turn to the choice where the z'-axis points towards the center of the galaxy. In this representation the host halo distribution is only dependent on θ' and r'. A drawback is however that the center of the halo, which will have small θ' values, will be heavily distorted in the $\theta'\phi'$ coordinate system. Because of the presumed symmetry around the center of the galaxy there is no preference for how the x' and y'-axes are oriented around the z'-axis. We start by defining

$\mathbf{r}_E = (x_E, y_E, z_E)_C$ as the coordinate system of the Earth. If we move the origo to the Earth any generic star at a generic position $\mathbf{r}_s = (x_s, y_s, z_s)_C$ will have the coordinates $\mathbf{r}_s - \mathbf{r}_E$. We now want to turn these coordinate so that $\mathbf{O} = (0,0,0)_C$ has coordinates $(0,0,r_E)_E$ seen from the Earth. This means that in the new coordinates for $\mathbf{r}_s$ can be written

$$\mathbf{r}'_s = R \cdot (\mathbf{r}_s - \mathbf{r}_{:E})$$

where R is the transformation mentioned earlier. The rotation, R, moves the z-axis from pointing upward to pointing in the direction of the center of the galaxy. What we will do first is calculate the rotations that move a star from position $(0,0,r_E)$ to position $(-x_E, -y_E, -z_E)$. Then we will take the inverse of the rotation since it is the same thing to rotate vectors one way as to rotate coordinates the other way.

We accomplish this by first rotating $\theta_{\text{rot}} = \pi - \theta_E$ around the y-axis. After that we rotate $\phi_{\text{rot}} = \phi_E + \pi$ around the z-axis as figure 9 implies. The matrix that describes a rotation around the y-axis an angle α is

$$R_y(\alpha) = \begin{pmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{pmatrix}$$

and an angle β around the z axis is

$$R_z(\beta) = \begin{pmatrix} \cos\beta & -\sin\beta & 0 \\ \sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

If we first rotate around y, and then z we get

$$\begin{aligned} R_{zy} &= R_z(\beta) R_y(\alpha) = \begin{pmatrix} \cos\beta & -\sin\beta & 0 \\ \sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{pmatrix} \\ &= \begin{pmatrix} \cos\alpha\cos\beta & -\sin\beta & \sin\alpha\cos\beta \\ \cos\alpha\sin\beta & \cos\beta & \sin\alpha\sin\beta \\ -\sin\alpha & 0 & \cos\alpha \end{pmatrix} \end{aligned} \tag{23}$$

As we said earlier, we are actually interested in rotating the coordinate axes and not rotating the coordinates. Fortunately rotating the coordinate axes is that same thing as rotating the coordinate backwards. In the case of R_{zx} it is easy to find the inverse. Since both R_x and R_z fulfills $R^{-1}(\gamma) = R(-\gamma) = R^\dagger$ we have

$$\begin{aligned} R_{zy}^{-1} &= (R_z(\beta) R_x(\alpha))^{-1} = R_y^{-1}(\alpha) R_z^{-1}(\beta) \\ &= R_y(-\alpha) R_z(-\beta) = R_y^\dagger R_z^\dagger = (R_z R_y)^\dagger \\ &= R_{zy}^\dagger \end{aligned}$$

The final rotation we are interested in thus becomes

$$R_{z-\text{axis}} = R_{zy}^{-1} = R_{zy}^{\dagger} = \begin{pmatrix} \cos\alpha\cos\beta & \cos\alpha\sin\beta & -\sin\alpha \\ -\sin\beta & \cos\beta & 0 \\ \sin\alpha\cos\beta & \sin\alpha\sin\beta & \cos\alpha \end{pmatrix} \tag{24}$$

Now, if we put in $\alpha = \pi - \theta_E$ and $\beta = \phi_E \pm \pi$ and use the trigonometric relation to get

$$\begin{aligned} \cos\alpha = & \; - & \cos\theta_E \\ \sin\alpha = & & \sin\theta_E \\ \cos\beta = & \; - & \cos\phi_E \\ \sin\beta = & \; - & \sin\phi_E \end{aligned}$$

By inserting the above relations into (24) we finally get

$$R_{z-\text{axis}} = \begin{pmatrix} \cos\theta_E\cos\phi_E & \cos\theta_E\sin\phi_E & -\sin\theta_E \\ \sin\phi_E & -\cos\phi_E & 0 \\ -\sin\theta_E\cos\phi_E & -\sin\theta_E\sin\phi_E & -\cos\theta_E \end{pmatrix} \tag{25}$$

We can check that this R is the one we are looking for. If we transform $\mathbf{O}$ we should get $(0, 0, r_E)$. Let us try it.

$$\begin{aligned} \mathbf{O}' &= R_{z-\text{axis}} \cdot (\mathbf{O} - \mathbf{r}_E) = -R_{z-\text{axis}} \cdot \mathbf{r}_E \\ &= -\begin{pmatrix} \cos\theta_E\cos\phi_E & \cos\theta_E\sin\phi_E & -\sin\theta_E \\ \sin\phi_E & -\cos\phi_E & 0 \\ -\sin\theta_E\cos\phi_E & -\sin\theta_E\sin\phi_E & -\cos\theta_E \end{pmatrix} \begin{pmatrix} r_E\sin\theta_E\cos\phi_E \\ r_E\sin\theta_E\sin\phi_E \\ r_E\cos\theta_E \end{pmatrix} \\ &= -r_E \begin{pmatrix} \cos\theta_E\cos\phi_E\sin\theta_E\cos\phi_E + \cos\theta_E\sin\phi_E\sin\theta_E\sin\phi_E - \sin\theta_E\cos\theta_E \\ \sin\phi_E\sin\theta_E\cos\phi_E - \cos\phi_E\sin\theta_E\sin\phi_E \\ -\sin\theta_E\cos\phi_E\sin\theta_E\cos\phi_E - \sin\theta_E\sin\phi_E\sin\theta_E\sin\phi_E - \cos\theta_E\cos\theta_E \end{pmatrix} \\ &= -r_E \begin{pmatrix} \cos\theta_E\sin\theta_E\left[\cos^2\phi_E + \sin^2\phi_E - 1\right] \\ \sin\phi_E\sin\theta_E\cos\phi_E - \cos\phi_E\sin\theta_E\sin\phi_E \\ -\sin^2\theta_E\cos^2\phi_E - \sin^2\theta_E\sin^2\phi_E - \cos^2\theta_E \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ 0 \\ r_E \end{pmatrix} = C_{z-\text{axis}} \end{aligned}$$

The result is that we expected. Even though (25) is the most general rotation for our specific form of orientation we can without loss of randomness choose the origo of the Earth to be $\theta_E = \pi$ so that

$$R_{z-\text{axis}} = \begin{pmatrix} -\cos\phi_E & -\sin\phi_E & 0 \\ \sin\phi_E & -\cos\phi_E & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The choice of ϕ_E is arbitrary but for simplicity we can say $\phi_E = \pi$ because then $R_{z-\text{axis}}$ is the unit rotation.

B.2 The x'-axis off center

The second choice is to let the x'-axis be pointing away from the galaxy center. This means that the galaxy will be placed at $\theta = \frac{\pi}{2}$ and $\phi = \pi$. Orienting the coordinate system this way causes the galaxy origo to be in the middle of the $\theta\phi$- plane, thus reducing the distortion at small angles to the galaxy origo. This way we will get pictures like figure 10 and figure 11.

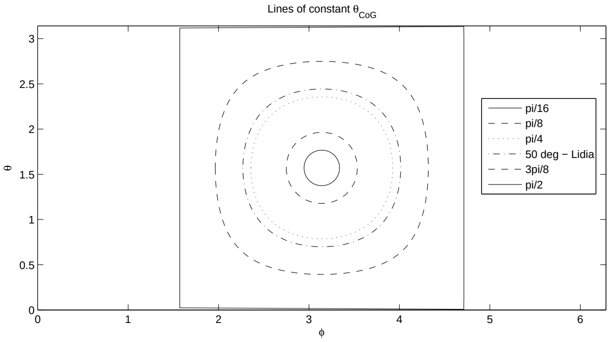

Figure 10: Lines of constant θ_{CoG}. This is the angle compared to the Center of the Galaxy.

Here we take advantage of the work done in B.1. What we need to do is to rotate a point $(-r_E, 0, 0)$ to $(-x_E, -y_E, -z_E)$, and we can do so by first rotating the point to $(0, 0, r_E)$ and then using the rotation (23) derived earlier. The initial rotation is simple since it is just the R_y rotation with angle $\alpha = \frac{\pi}{2}$ giving

$$R_y\left(\frac{\pi}{2}\right) = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{pmatrix}$$

The final rotor for this scenario is thus

$$\begin{aligned} R_{x-\text{axis}} &= \left(R_{yz} R_y\left(\frac{\pi}{2}\right)\right)^{-1} = R_y^{-1}\left(\frac{\pi}{2}\right) R_{yz}^{-1} \\ &= R_y^{\dagger}\left(\frac{\pi}{2}\right) R_{z-\text{axis}} \end{aligned}$$

which explicitly is

$$
\begin{aligned}
R_{x-\mathrm{axis}} &= \begin{pmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} \cos\theta_E\cos\phi_E & \cos\theta_E\sin\phi_E & -\sin\theta_E \\ \sin\phi_E & -\cos\phi_E & 0 \\ -\sin\theta_E\cos\phi_E & -\sin\theta_E\sin\phi_E & -\cos\theta_E \end{pmatrix} \\
&= \begin{pmatrix} -\sin\theta_E\cos\phi_E & -\sin\theta_E\sin\phi_E & -\cos\theta_E \\ \sin\phi_E & -\cos\phi_E & 0 \\ -\cos\theta_E\cos\phi_E & -\cos\theta_E\sin\phi_E & \sin\theta_E \end{pmatrix}
\end{aligned}
$$

We can try to calculate $\mathbf{O}'$ in this coordinate system and get

$$
\begin{aligned}
\mathbf{O}' &= R_{x-\mathrm{axis}} \cdot (\mathbf{O} - \mathbf{r}_E) = -R_{x-\mathrm{axis}} \cdot \mathbf{r}_E \\
&= -\begin{pmatrix} -\sin\theta_E\cos\phi_E & -\sin\theta_E\sin\phi_E & -\cos\theta_E \\ \sin\phi_E & -\cos\phi_E & 0 \\ -\cos\theta_E\cos\phi_E & -\cos\theta_E\sin\phi_E & \sin\theta_E \end{pmatrix} \begin{pmatrix} r_E\sin\theta_E\cos\phi_E \\ r_E\sin\theta_E\sin\phi_E \\ r_E\cos\theta_E \end{pmatrix} \\
&= -r_E \begin{pmatrix} \sin^2\theta_E\cos^2\phi_E + \sin^2\theta_E\sin^2\phi_E + \cos^2\theta_E \\ \sin\phi_E\sin\theta_E\cos\phi_E - \cos\phi_E\sin\theta_E\sin\phi_E \\ \cos\theta_E\sin\theta_E\left[\cos^2\phi_E + \sin^2\phi_E - 1\right] \end{pmatrix} \\
&= -\begin{pmatrix} r_E \\ 0 \\ 0 \end{pmatrix} = C_{x-\mathrm{axis}}
\end{aligned}
$$

To simplify the form of $R_{x-\mathrm{axis}}$, without loosing randomness, we can here choose $\theta_E = \frac{\pi}{2}$ and $\phi_E = \pi$ which yields $R_{x-\mathrm{axis}} = 1$.

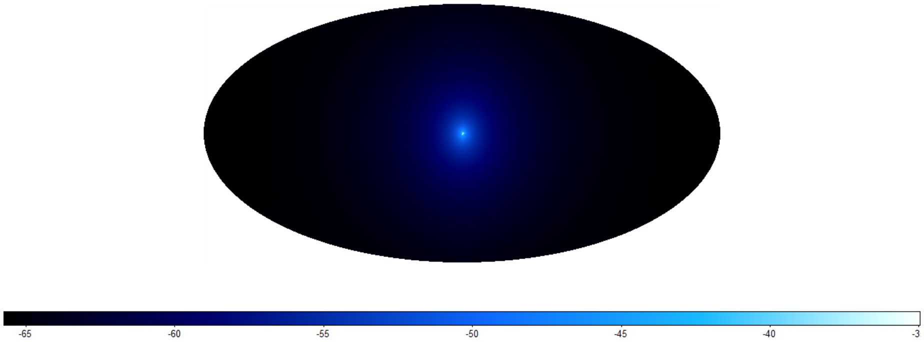

Figure 11: Sky map of the smooth component as seen from the Earth. The scale is logarithmic.

B.3 The general axis on center

If we would like to orient our coordinate system so that a generic (θ', ϕ')-axis points towards the galaxy origo the general procedure is as follows. Here we can reuse what we have done in B.1 and B.2. First, we rotate a point at (x'_E, y'_E, z'_E) to $(0,0,1)$ and the rotate to (x_E, y_E, z_E). The first rotation is really the inverse of $R_{yz}(\theta', \phi')$, since $R_{yz}(\theta', \phi')$ rotates $(0,0,1)$ to (x'_E, y'_E, z'_E). Therefore the rotation here is

$$R = R_{yz} R_{yz}^{-1}(\theta', \phi')$$

and therefore we have

$$\begin{aligned} R_{\text{genral axis}} &= \left(R_{yz} R_{yz}^{-1}(\theta', \phi')\right)^{-1} = R_{yz}(\theta', \phi') R_{yz}^{-1} \\ &= R_{yz}(\theta', \phi') R_{z-\text{axis}} \end{aligned}$$

or explicitly

$$R_{\text{genral axis}} = \begin{pmatrix} \cos\theta' \cos\phi' & -\sin\phi' & \sin\theta' \cos\phi' \\ \cos\theta' \sin\phi' & \cos\phi' & \sin\theta' \sin\phi' \\ -\sin\theta' & 0 & \cos\theta' \end{pmatrix} \begin{pmatrix} \cos\theta_E \cos\phi_E & \cos\theta_E \sin\phi_E & -\sin\theta_E \\ \sin\phi_E & -\cos\phi_E & 0 \\ -\sin\theta_E \cos\phi_E & -\sin\theta_E \sin\phi_E & -\cos\theta_E \end{pmatrix}$$

We can as in the previous subsections choose $\theta_E = \pi - \theta'$ and $\phi_E = \phi' \pm \pi$ to get $R_{\text{genral axis}} = 1$. This is because $R_{z-\text{axis}}(\theta_E, \phi_E) = R_{zy}^{-1}(\alpha, \beta) = R_{zy}^{-1}(\pi - \theta_E, \pi + \phi_E)$. When we take $R_{yz}(\theta', \phi') R_{z-\text{axis}} = R_{yz}(\theta', \phi') \times R_{zy}^{-1}(\pi - \theta_E, \phi_E \pm \pi)$ and then we can identify $\theta_E = \pi - \theta'$ and $\phi_E = \phi' \pm \pi$.

If we insert $\theta' = 0$ and $\phi' = 0$ we receive $R_{z-\text{axis}}$ and if we choose $\theta' = -\frac{\pi}{2}$ and $\phi' = 0$ we get $R_{x-\text{axis}}$.

C Some notes about FORTRAN

FORTRAN is a programming language that goes back to the 50's. The language was developed by IBM around 1956 and quickly became the language of choice for technical and scientific programming. In 1966 the first standard FORTRAN, FORTRAN 66, was released and eleven years later in 1977 the FORTRAN 77 standard was off the ground. Later standards are the FORTRAN 90 and FORTRAN 95 standard. My code is made using the FORTRAN 77 standard.

FORTRAN is built for speed, and has thus made some sacrifices to get it. First of FORTRAN almost never asks you for anything. If you have a function that has an integer as input it will assume that you use integers and not floats. You are at liberty to call any function with the wrong type of variable but surprising things might happen.

Another feature is that while C and other languages usually have call by value when you use a function, FORTRAN does not. Instead FORTRAN uses call by reference making it possible to alter the value of the calling variables. As an example, if you in C call function $f(x)$, the variable x that is used to call f is not the same as the variable that f uses inside the function. C makes a copy of x and uses that copy for calculations. In FORTRAN the x that is used in the call of f and the x that is used inside f is one and the same. This means that the function f can change that value of x.

All functions in FORTRAN need to be called as an assignment, like $y = g(x)$. Usually this does not pose a problem, but if you would like to take advantage of the call by reference structure it does. An example is when you want to make a coordinate transform from spherical to Cartesian coordinates. A way to to this would be to define the function sp2cart$(\theta, \phi, r, x, y, z)$ by

```
function sp2cart(theta,phi,r,x,y,z)
x=r*sin(theta)*cos(phi)
y=r*sin(theta)*sin(phi)
z=r*cos(theta)
end
```

This function does the coordinate transform of interest, but since you have to do it as an assignment, with

```
a=sp2cart(theta,phi,r,x,y,z)
```

therefore you have to add a row

```
sp2cart=0
```

or something similar to the function, somewhere above the `end`. This procedure is rather cumbersome and irritating at best. The way around it is to use a cousin to the `function` called `subroutine`. A `subroutine` is not allowed to have a functional value opposed to the `function`. Any subroutine can be called using call, like

```
call sp2cart(theta,phi,r,x,y,z).
```

In FORTRAN the type of a variable is important, as it is in quite a lot of other languages as well. The FORTRAN standard gives the programmer the choice of how he/she wants to typecast his/her variables. You can either declare the type in the beginning of the code, but you can also skip the declaration and use the automatic declaration built into FORTRAN. This means that whenever the compiler finds a variable that has not been typed FORTRAN type sets the variable itself using a simple rule. Variables starting with A-H and O-Z are `real` while variables starting with I-N are `integer`. This auto-typesetting can be quite a nuisance if you are a frequent miss-speller. Even a small error in spelling can be hard to find. The solution is to write `implicit none` at the beginning of the code, meaning that all variables have to be typed explicitly.