A new ray tracing algorithm for complex ice models and the analysis of ice properties using weather balloons

Arthur Adriaens

Department of Physics and Astronomy

Promotor: Prof. dr. Dirk Ryckbosch   Dirk.Ryckbosch@ugent.be

Supervisor: Bob Oeyen   Bob.Oeyen@ugent.be

Master's dissertation submitted in partial fulfilment of the requirements for the degree of master in Physics and Astronomy

SUMMARY

Outside our earth various kinds of events take place which we wish to observe: Black hole outbursts, supernovae, cosmic jets, ... These events produce various kinds of messengers which are useful to obtain information on the event: Gravitational waves, gamma rays, protons,... But one particle within this set of particles is quite unique and the subject of our study: The neutrino.

The neutrino is unique in that it points back to the event itself. Due to the neutrino not having any charge, nearly no mass and interacting weakly it doesn't get bend nor absorbed and re-emitted on it's way to us unlike, say, the proton. This means that if we observe a neutrino back here on earth it's very likely that the direction we observe it in is the direction it came from.

We wish to detect a particular type of neutrinos: The ultra high energy (UHE) neutrino. There have been lots of neutrino detectors around but none of them have been able to observe *cosmogenic neutrinos* which would live past the 3 PeV energy range, this is probably caused by the exponentially falling flux with increasing energy, implying that a really big detector volume is required to detect neutrinos with such high energies.

Due to cost requirements, it was necessary to work in the radio regime. As much as we'd like a really big detector working in the visible spectrum, it would cost way to much. As neutrinos can produce radio waves upon interacting in ice through the Askaryan effect and as radio waves can travel for way longer distances in there before interacting than visible light, it was previously descided in experiments like ARA and ARIANNA to detect on the principle of radio waves. The Radio Neutrino Observatory in Greenland or RNO-G wich is currently under construction and the subject of this thesis builds on the knowledge of these experiments to make a quite complex detector which should be capable of detecting UHE neutrinos.

The ice properties have an impact on how radiowaves propagate. As mentioned before, RNO-G is a detector built in the Greenland icecap. As the radiowaves get produced in the ice we need to figure out how they propagate towards our detector. An important part in figuring out how they propagate is the local index of refraction which seems to be linearly related to the density of the ice. The density of the ice seems to vary continuously with depth, a function describing

this overall relation of the index of refraction with the depth is called an *ice model* and it is crucial for future studies to understand this ice model.

Due to the ice model seeming to deviate from the theoretically expected single exponential, it was necessary to develop a new algorithm to find the paths the radio waves could take with more complex ice models. The software used within this thesis to figure out the path radiowaves take is called *radiopropa*. Even though this software can be used to work with most kinds of ice models, it still needs to be used within some kind of algorithm to be useful. Such an algorithm has already been developed called the *Iterative ray tracer* as is explained in section 3.3 but it had its shortcomings, that's why in this thesis a new algorithm was developed called the *hybrid ray tracer* which is explained in chapter 4. The hybrid ray tracer makes it possible to both find a more accurate solution and find it faster.

It is, however, crucial to make sure we actually need a new kind of ice model. The measuring of the index of refraction can be accomplished indirectly through weather balloon fly-bys as is explained in chapter 5. Every day in the summer, two times per day a weather balloon is launched from the base camp. This weather balloon is equiped with an antenna which sends out a 403MHz sinusoidal signal, upon close fly-bys with detectors this signal can be observed in the detectors (as is visualized on the cover). By looking at the difference in arrival time of the signal, the difference in timing can be made out from which a plane wave reconstruction can be done. As this plane wave reconstruction is heavily dependent on the local index of refraction in the ice, it can be indirectly measured through this reconstruction. After this analysis the following data was found:

Depth (m)	Station id	channels	Run:Event	$n_{\text{exponential}}$	ϵ	n_{fitted}
-48.155	21	6&7	1441:117	1.6400	-0.02%	1.6310 ± 0.001
-58.38	21	5&7	1441:117	1.6736	-0.23%	1.6688 ± 0.0006
-68.2	21	5&6	1441:117	1.6983	0.02%	1.6916 ± 0.0003
-93.237	23	phased array	800:1867	1.7383	4.6166%	1.7393 ± 0.0013

As can be seen all of the undeep channels show a discrepency with the exponential model.

SAMENVATTING

Er vinden verscheidene gebeurtenissen plaats in het heelal waarover we meer wensen te weten zoals Black hole outbursts, supernovae, cosmic jets,... Al deze gebeurtenissen produceren vormen van informatie dewelke we kunnen infereren door hun op een manier te detecteren zoals de protonen en fotonen maar ook meer exotische bronnen van informatie zoals zwaartekracht golven. Maar één bepaald deeltje vinden wij bijzonder interressant: de neutrino.

De neutrino heeft als erg unieke eigenschap dat het terugwijst naar het event waar ze gecreëerd werd. In tegenstelling tot andere deeltjes zoals protonen heeft de neutrino geen lading, bijna geen massa en interageert ze zeer zwakjes. Door deze eigenschappen zal ze op de weg naar de aarde niet afgebogen worden door magneetvelden en hoogstwaardschijnlijk niet interageren met tussenmedia zoals gaswolken. Deze eigenschappen impliceren dat, als we erin slagen een neutrino the detecteren, de richting waarin ze gedetecteerd wordt hoogstwaarschijnlijk eenzelfde richting is als waar ze vandaan komt.

Neutrinos komen in verschijdene energieën voor maar wij zijn geïnteresseert in één bepaald gebied, boven de 3 PeV oftewel de ultra hoge energie (UHE) neutrino. In de verscheidene experimenten die neutrinos hebben gedetecteerd is er nog geen een in geslaagd neutrinos te observeren met energieën boven de 3 PeV, dit is het regime waar zogenaamde *cosmogenic neutrinos* zouden leven en dus uiterst interessant om te bestuderen. Maar, om zo'n hoge neutrino energie waar te nemen, blijkt een zeer grote detector nodig te zijn.

Aangezien het zeer rap zeer duur zou zijn om een groot genoege detector te bouwen om deze neutrinos te detecteren in het visuele spectrum, wat bijvoorbeeld gebruikt wordt in detectoren zoals IceCUBE. Werd besloten om te leunen op het Askaryan effect, dit is het effect die zorgt voor de productie van radiogolven bij interactie van een neutrino in ijs. Aangezien radio golven verder vrij kunnen propageren in ijs dan zichtbaar licht maakt dit het mogelijk om de verscheidene antennes verder van elkaar te plaatsen. Enkele experimenten zijn op dit principe gebouwd zoals ARA en ARIANNA, uit de expertise van deze en soortgelijke experimenten werd dan de Radio Neutrino Observatory in Greenland of RNO-G gemaakt. RNO-G moet het mogelijk maken om UHE neutrinos te observeren en deze thesis wenst bij te dragen aan de ontwikkeling van deze detector.

Het blijkt dat de eigenschappen van het ijs een effect hebben op hoe radiogolven zich propageren door het ijs. Aangezien RNO-G gebouwd is in groenland op een groot volume ijs en ze werkt op het principe van radiogolven is het nodig dit te onderzoeken. Het blijkt dat de propagatie sterk afhankelijk is aan de locale refractieve index van het ijs, dewelke lineair lijkt gerelateerd te zijn aan de dichtheid van het ijs, hoe de locale refractieve index afhangt van de ruimtelijke positie wordt het *ijsmodel* genoemd. Theoretisch verwacht men een exponentiele afhankelijkheid van de dichtheid met de diepte, maar uit metingen blijkt deze af te wijken van het theoretische model.

De ray tracing simulaties die gebruikt worden om na te gaan waar de neutrino vandaan komt zijn sterk afhankelijk van het ijsmodel. Er werd vroeger een algoritme gemaakt dewelke enkel met het exponentieel model kan omgaan, maar door de recent gevonden tekortkomingen aan dit ijsmodel bleek het nodig een nieuw algoritme te bedenken. Zo'n algoritme, genaamd de *iterative ray tracer* werd daarom uitgevonden (zie sectie 3.3) maar bleek enkele tekortkomingen te hebben. Daarom vonden wij het nodig om een nieuw soort algoritme uit te vinden dewelke steunt op dat algoritme genaamd de *hybrid ray tracer*, de werking van dit algoritme kan gevonden worden in hoofdstuk 4. Dit algoritme maakt het mogelijk om beide een meer accurate oplossing te vinden maar deze ook sneller te vinden.

Maar om over te schakelen op een nieuw ijsmodel met een anders algoritme is het nodig aan te tonen dat er effectief een groot verschil is tussen het theoretische exponentiële model en de echte refractieindex, om dit aan te tonen kunnen we gebruik maken van weerballonnen. Elke dag in de zomer, 2 maal per dag, wordt een weerballon gelanceerd op de RNO-G basis. Deze weerballon beschikt over een antenne dewelke een sinusoïdaal signaal uitstuurt aan 403MHz, als zo'n ballon toevallig dicht bij een detector komt zal dit signaal kunnen gedetecteerd worden. Door een plane wave reconstructie te doen van dit signaal en gebruik te maken van de positie van de ballon kunnen we infereren wat de locale refractieve index is in het ijs. Deze procedure wordt volledig uitgelegd in hoofdstuk 5. Na deze procedure op enkele evenementen toe te passen werd de volgende data gevonden:

Depth (m)	Station id	channels	Run:Event	$n_{exponential}$	ϵ	n_{fitted}
-48.155	21	6&7	1441:117	1.6400	-0.02%	1.6310 ± 0.001
-58.38	21	5&7	1441:117	1.6736	-0.23%	1.6688 ± 0.0006
-68.2	21	5&6	1441:117	1.6983	0.02%	1.6916 ± 0.0003
-93.237	23	phased array	800:1867	1.7383	4.6166%	1.7393 ± 0.0013

Hier is een duidelijk verschil tussen de gemeten index van refractie (n_{fitted}) en die uit het exponentieel model ($n_{exponential}$) te zien.

DANKWOORD

CONTENTS

INTRODUCTION

Outside our earth various kinds of events take place which we wish to observe: Black hole outbursts, supernovae, cosmic jets, ... These events produce various kinds of messengers which are useful in detecting them: Gravitational waves, gamma rays, protons,... But one particle within this set of particles is quite unique and the subject of our study: The neutrino.

The neutrino is unique in that it points back to the event itself. Due to the neutrino not having any charge, nearly no mass and interacting weakly it doesn't get bend or absorbed and re-emitted on it's way to us unlike say the proton. This means that if we observe a neutrino back here on earth it's very likely that the direction we observe it in, is the direction it came from.

We wish to detect a particular type of neutrinos: The ultra high energy (UHE) neutrino. There have been lots of neutrino detectors around but none of them have been able to observe *cosmogenic neutrinos* which would live past the 3 PeV energy range, this is probably caused by the exponentially falling flux with increasing energy, implying that a really big detector volume is required to detect neutrinos with such high energies.

Due to cost requirements, it was necessary to work in the radio regime. As much as we'd like a really big detector working in the visible spectrum, it would cost way to much. As neutrinos can produce radio waves upon interacting in ice through the Askaryan effect and as radio waves can travel for way longer distances in there before interacting than visible light, it was previously descided in experiments like ARA and ARIANNA to detect on the principle of radio waves. The Radio Neutrino Observatory in Greenland or RNO-G wich is currently under construction and the subject of this thesis builds on the knowledge of these two experiments to make a quite complex detector which should be capable of detecting UHE neutrinos.

The ice properties have an impact on how radiowaves propagate. As mentioned before, RNO-G is a detector built in the Greenland icecap. As the radiowaves get produced in the ice we need to figure out how they propagate towards our detector. An important part in figuring out how they propagate is the local index of refraction which seems to be linearly related to the density of the ice. The density of the ice seems to vary continuously with depth, a function describing this overall relation of the index of refraction with the depth is called an *ice model* and it is

crucial for future studies to understand this ice model.

It has become apparent that the ice model seems to deviate from the theoretically expected single exponential density expectation. In this thesis a new algorithm will be developed which makes it possible to work with more complex ice models and the verification of the shortcomings of the exponential model, which is now the only model that gets used, will be layed out.

CHAPTER

1

NEUTRINO AS ASTROPARTICLE

When looking at phenomena outside our earth the astronomer will turn to electromagnetic radiation, but he's missing out on a big part of the full picture. Not only do interesting events emit photons but also muons, nuclei, gravitational waves,... All kinds of particles which might also be of interest, that's where the astroparticle physicist comes in.

Of all these particles there is one particle which has properties we're quite interested in: the neutrino. Neutrinos don't have any charge, meaning that they are not deflected by magnetic fields. Also neutrinos interact very weakly, because of this they are often called "ghost" particles; on average 100 trillion neutrinos pass through your body per second, none of them having any effect. You'd even need a light year of lead to give you just a 50% chance of stopping a neutrino. These properties makes them ideal messenger particles as, when we detect a neutrino and it's arrival direction, we can be quite sure it came to our detector unhindered from a far away event in the exact same direction. Neutrinos can serve as unique clues about what's happening elsewhere in the universe including the cosmic collisions, galaxies, supernovae, Gamma-ray bursts (GRBs),... where they are created.

In this chapter we'll give an overview of neutrinos and the various kinds of energies we expect to see them in when coming from outside our earth.

1.1 Discovery

When researching β^- decay, the decay of a neutron, researcher detected a proton and an electron coming from the neutron. However on closer inspection it became apparent that energy was lost somewhere in violation with the conservation law of energy, and angular momentum wasn't conserved either. The solution postulated by Wolfgang Pauli was to introduce a new, really hard to detect particle with no charge and a very small mass: the neutrino. The neutrino comes in three flavours: electron, muon and tau neutrinos, each corresponding to their respective lepton denoted as

$$\nu_e \quad \nu_\mu \quad \nu_\tau \tag{1.1}$$

and each also having an anti-particle.

$$\bar{\nu}_e \quad \bar{\nu}_\mu \quad \bar{\nu}_\tau \tag{1.2}$$

Now with the introduction of the neutrino the full β^- decay becomes

$$n \to p^+ + e^- + \nu_e \tag{1.3}$$

The inverse can then also be used to detect neutrinos:

$$n \to p^+ + e^- + \nu_e \tag{1.4}$$

Which is called beta capture and was first experimentally detected in 1956 [12] also making it the first experimental detection of a neutrino.

1.2 Charged and neutral currents

As with the strong and electromagnetic interaction whom interact via gluons and photons respectively, the weak interaction which is the primary way neutrinos interact, is also mediated by the exchange of elementary spin-1 bosons. Namely the charged bosons W^+ and W^- and the neutral Z^0 boson. Contrary to the gluons and photons however, these bosons are very massive:

$$M_W = 80.40 \text{ GeV} \qquad M_Z = 91.19 \text{ GeV} \tag{1.5}$$

Following general arguments, it can be derived that the range of any force is given by the compton wavelength of the particles transmitting it [28], as the compton wavelenght is given by

$$\lambda = \frac{h}{mc} \tag{1.6}$$

The very large mass of the "force carriers" imply a very short-range interaction, in the order of 10^{-3}fm and can even be treated as a zero-range interaction at low energies.

These bosons can couple to leptons (e.g neutrinos, electrons, tauons and muons) but also to quarks, if they couple via the charged bosons then the coupling process is called a *charged current* reaction and if they couple via the Z^0 boson than the process is called a *neutral-current* interaction.

A simple interaction is sketched below

$$\mu^- \to e^- + \bar{\nu}_e + \nu_\mu \tag{1.7}$$

here the muon decays via a charged current reaction, note that lepton number needs to be conserved. This means that, as on the left side of the reaction we have $\ell_\mu = 1$ and $\ell_e = 0$ this needs to be the case on the right side. The lepton number is conserved as a muon neutrino counts for $\ell_\mu = 1$ and an electron anti-neutrino counts for $\ell_e = -1$

1.3 Neutrino sources

As shown in figure 1.1 there are various kinds of neutrino sources, we'll discuss these one by one in order leaving out the reactor anti-neutrinos and the terrestrial anti-neutrinos as we're only interested in neutrinos of astrophysical nature.
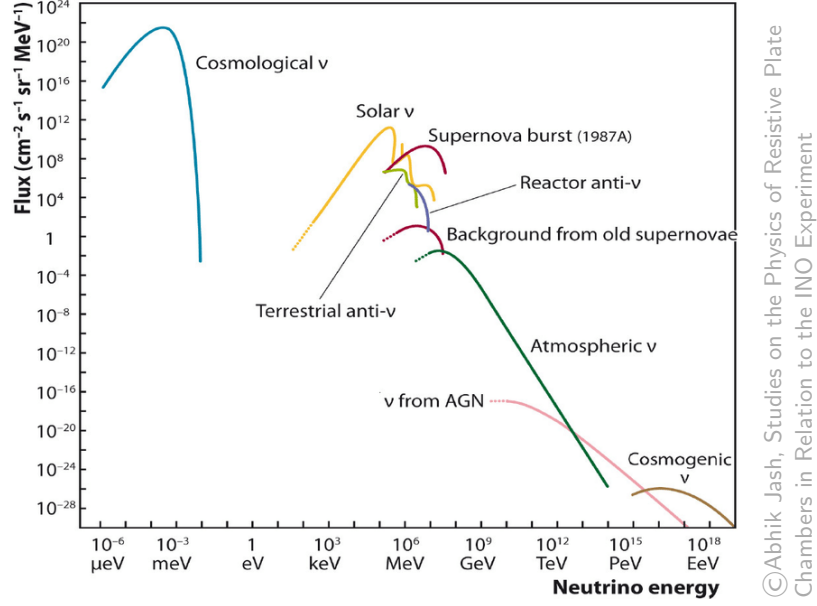
Figure 1.1: Predicted neutrino flux for various sources both natural and man-made

1.3.1 Cosmological/Primordial neutrinos

The first source of neutrinos we'll talk about is the one in blue to the left of the figure termed the *Cosmological neutrinos*: the neutrino version of the CMB. To understand this source we'll have to go back all the way to just after the big bang: The very early universe was hot and dense. As a result, interactions among particles occurred much more frequently than they do today. As an example, a photon today can travel across the observable universe without deflection or capture, so it has a mean free path greater than 10^{26} m. When the universe was 1 second old, though, the mean free path of a photon was about the size of an atom. Thus in the time it took the universe to expand by a factor of 2, a given photon interacted many, many times. These multiple interactions kept the consitituents in the universe in thermal equilibrium. But as the universe expanded there were times when reactions could not proceed rapidly enough to maintain equilibrium conditions, these particles then fall out of thermal equilibrium. This falling out of equilibrium is termed *decoupling*. And we're interested in when neutrinos decoupled. Neutrinos were kept in equilibrium through the interaction

$$\nu e \leftrightarrow \nu e \tag{1.8}$$

up until the universe cooled down to about 1MeV when they decoupled. To estimate the temperature of the neutrinos who decoupled at the start of the universe, we can take a look at conservation of entropy [13] from which we'll find that the temperatures are related by:

$$T_\nu = \left(\frac{4}{11} \right)^{1/3} T_\gamma \tag{1.9}$$

Note that they decoupled before the photons making them lower in temperature. As T_γ is the CMB temperature which, nowadays, is measured to be around 2.7K or 2.3×10^{-4}MeV. This would imply $T_\nu = 1.66 \times 10^{-4}$MeV which is rougly where the peak flux is located. these primordial neutrinos are thus very low in energy.
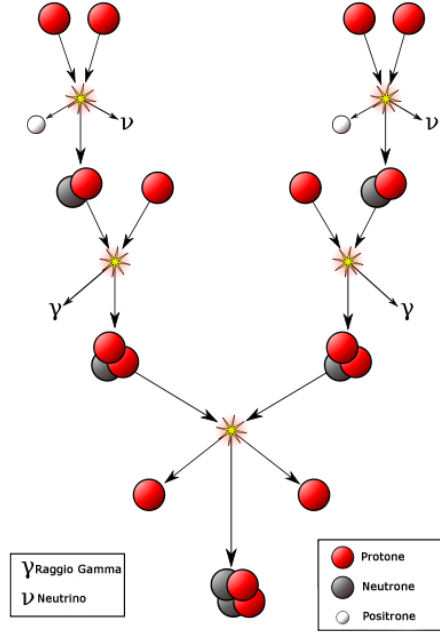
5

Figure 1.2: illustration of the full fusion cycle in the sun

1.3.2 Solar neutrinos

The sun fuses elements to release energy and thus keeping itself from collapsing in on itself, with most of the various ways particles get fused, neutrinos get released as is shown in figure 1.2 where the neutrinos are electron neutrinos. Now with this and some information about the sun like the pressure and mass, the *standard solar model* was made. This model predicted a certain amount of electron neutrinos to be hitting the earth from the previously mentioned thermonuclear fusion, it was however 3 times higher than the observed amount of electron neutrinos back at our planet. This led to a little bit of hysteria as this could've meant that the sun was dying and we'd see the aftermath only in a couple of years. Through various experiments however, it became apparent that this was due to the different kinds of neutrinos oscillating into each other on their way to earth, i.e 2/3 of the original electron neutrinos had oscillated into mu and tau neutrinos. But for them to oscillate into eachother, they not only require mass but each flavor also should have a different mass as can be seen from an example 2D approximation to the transition probability [9]:

$$P(\nu_e \to \nu_\mu) = |\langle \nu_\mu | \psi(L,T) \rangle|^2 = c_\mu c_\mu^* = \sin^2(2\theta)\sin^2\left(\frac{\Delta\phi_{12}}{2}\right) \tag{1.10}$$

with

$$\Delta\phi_{12} \approx \frac{m_1^2 - m_2^2}{2p}L \tag{1.11}$$

In full generality (3D):

$$|\nu_\alpha\rangle = \sum_i U_{\alpha i} |\nu_i\rangle \tag{1.12}$$

With $U_{\alpha i}$ the Pontecorvo-Maki-Nakagawa-Sakata (PMNS) matrix. This phenomenon has been experimentally verified e.g through the descrepency from the observed and expected neutrino events coming from a nuclear reactor [14].

1.3.3 Supernovae

A star starts its life as a ball of pure hydrogen. At the core, due to the gravitational pressure of the outside plasma, fusion of hydrogen into deuterium and helium happens. Thus converting mass into energy. The pressure of this energy counteracts the pressure of gravity and the star is stable.

When the hydrogen at the core runs out no more hydrogen can be fused. For stars with masses between $8M_\odot$ and $30M_\odot$ ($M_\odot$ being the mass of the sun) the fusion of heavier elements starts, this can't keep going on however as at some point the star starts to form the most stable element: iron.

It costs energy to both make lighter elements than iron and heavier ones as can be seen on figure 1.3. As the iron core builds up the outside pressure from the core starts to decrease as no new
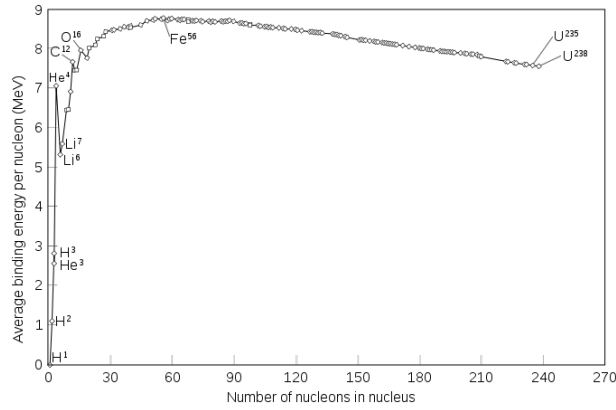

Figure 1.3: Energy curve showing that iron is the most stable atom

energy is released. This goes on until the treshold of an iron core with a mass of $1.4M_\odot$ known as the Chandrasekhar limit is reached and the the inwards pressure becomes too large compared to the outwards pressure, making the electrons surrounding the iron core fuse with the protons (uud), creating neutrons (udd) and neutrinos, diagramatically shown in figure 1.4.
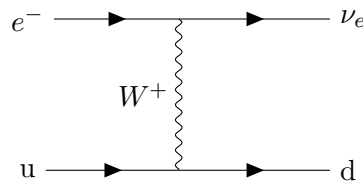
Figure 1.4: fusion of protons with surrounding electrons into neutrons via charged current exchange

This last part happens in a split second as the collapse goes at 25% the speed of light, creating a very dense neutron star (3000km in diameter iron core to 30km in diameter neutron star) and up to 10^{52} ultra-relativistic neutrinos, carrying up to 99%[1] of the released energy [29]. As the density has suddenly increased so much there's a huge distance of pure vaccuum between the plasma outer layer and the (now) neutron star, this plasma starts free-falling inwards, also at 25% the speed of light whilst the neutrinos carrying tremendous amounts of energy start going outwards from the neutron stars core.

---

[1]$\approx$1% is released as kinetic energy, only 0.001% as electromagnetic radiation

The neutrinos then collide with the plasma resulting in what we observe as a "supernova", wrongly thought of by Kepler as being a "new (nova) star" rather being a violent death of an old star.

This is quite unexpected as neutrinos rarely interact, it's only as the incoming plasma is so dense and due to the tremendous amount of neutrinos that collissions happen at all. Some, however, escape and will be visible on earth in our neutrino detectors $\approx$ 18h before the light escapes the exploding star.

Neutrino observatories are thus also useful to know where to point our various telescopes before the supernova is actually visible in the night sky.

1.3.4 Background from old supernovae

Also termed the *diffuse supernova neutrino background* (DSNB), as the universe is quite old various supernovae have happened over it's lifetime, each generating a lot of neutrinos as was discussed in section 1.3.3. This is postulated to have generated a continuous neutrino background.

1.3.5 Atmospheric neutrinos

Before we can talk about atmospheric neutrinos it's necessary to discuss *cosmic rays*. Cosmic rays are ionized nuclei of which 90% are protons, 9% are alpha particles and the rest are heavier nuclei. Almost all of them originate from outside the solar system but from within our galaxy, the few particles that do come from our solar system can be temporally linked to violent events on the sun. In contrast to this the particles coming from outside our solar system show an anti-correlation with the sun as they can more easily reach the earth if solar activity is low. It has been observed that they roughly follow a power-law spectrum $N \propto E^{-\gamma}$ [16].

Cosmic rays hit the Earth's atmosphere at a rate of about 1000 per square meter per second and interact with atomic nuclei in the Earth's atmosphere, creating showers of particles, many of which are unstable and produce neutrinos when they decay, these neutrinos are what's called *Atmospheric neutrinos*. Most notably neutrinos can be produced together with muons in the two-body decays of charged pions and kaons wherever these hadronic interactions occur. The most important production channels and their branching ratios for neutrinos are:

$$\pi^{\pm} \rightarrow \mu^{\pm} + \nu_{\mu}(\bar{\nu}_{\mu})(\sim 100\%) \tag{1.13}$$

$$K^{\pm} \rightarrow \mu^{\pm} + \nu_{\mu}(\bar{\nu}_{\mu})(\sim 63.5\%) \tag{1.14}$$

Neutrinos are subsequently also produced when these muons decay:

$$\mu^{\pm} \rightarrow e^{\pm} + \nu_e(\bar{\nu}_e) + \bar{\nu}_{\mu}(\nu_{\mu}) \tag{1.15}$$

which is a process mainly happening at low energies in the atmosphere. The atmospheric neutrino spectrum shown in figure 1.1 roughly follows a power spectrum as the cosmic ray flux follows a power spectrum but the correspondence isn't one-to-one as, due to the difference in kinematics, the contribution from kaons to neutrinos is significantly more important than to muons, especially at high energies.

1.3.6 neutrinos from AGNs

An AGN (active galactic nucleus) is deemed to be the reason why several abnormal galaxies exist with an extra bright (and mostly variable) light source in their core which even the biggest

of telescopes can't spatially discern. The general concensus is that this phenomenon is caused by one particular kind of object: a supermassive black hole (a black hole with a mass of at least $105M_\odot$) surrounded with a close torus of dust and gas. This torus of gas is called an *accretion disc* and is an enormous source of energy. The conversion of potential energy of the incoming gas to highly energetic radiation is a very complex physical process with which we have to account for various factors like gravitational instabilities, magnetic fields, hydrodynamical turbulence,... And thus produces a spectrum that's quite complex. It would appear that the luminocity of an AGN would increase indefinitely with incoming mass, but this process is limited: if too much matter accretes on the black hole the radiative pressure becomes too massive and the matter on the disc gets blown away, this phenomenon is termed a *black hole outburst.*

The emission of high energy neutrinos from AGNs rests solely on the premise that relativistic protons of sufficiently high energy and energy density in the AGN's accretion disc will be present [26] as they may interact to create e.g pions whom decay into neutrinos. A direct consequence of the occurence of these relativistic protons is the production of γ-rays of similar energies to those of the neutrinos, thus high energy neutrino- and γ-ray astronomy are closely related. However, even though γ-ray photons can be produced even in the absence of relativistic protons (e.g via high energy electrons), neutrinos can not. Thus the detection of these high-energy neutrinos (which might have already been detected [10]) will provide unique information about the workings of AGNs. As these ultra high energy neutrinos get produced near the source (the AGN) they are what's called *astrophysical neutrinos*

1.3.7 Cosmogenic neutrinos

In contrast to the previous source of UHE neutrinos which was generated at the source, called *astrophysical neutrinos* we'll now talk about UHE neutrinos whom are generated through the interaction of ultra-high energy cosmic rays during propagation with the cosmic microwave or other photon backgrounds termed *cosmogenic neutrinos*. The mechanism by which these get created is quite simple, if a proton has sufficiently high energy the cross section to interact with the CMB (Cosmic Microwave Background) becomes non-negligable. These protons can scatter off the photons to resonantly produce a Δ^+ baryon. This resonance has enough mass to dominantly decay to a pion and a nucleon:

$$\Delta^+ \to \pi^0 + p \quad (2/3) \tag{1.16}$$

$$\Delta^+ \to \pi^+ + n \quad (1/3) \tag{1.17}$$

Of which the charged pion decays to neutrinos as previously mentioned in 1.3.5, these neutrinos carry most of the original energy.

1.3.8 How do they fit into the full detector spectrum?

The origin of the most energetic cosmic rays is still not conclusively identified. One approach to solving this problem is *multi-messenger astrophysics*, where several types of cosmic particles are used to identify the sources of these ultra-high energy cosmic rays (UHECRs). E.g we simultaneously measure gravitational waves with the Einstein telescope, neutrinos with RNO-G, photons with various telescopes and muons with a muon detector.

CHAPTER

2

RADIO NEUTRINO DETECTION

Neutrinos on their own aren't detectable, unlike say protons we can't expect them to produce brehmsstrallung or any other kind of directly observable radiation as they aren't charged and thus don't couple to photons. We'll need to convert them to charged particles first which then can generate some kind of detectable signal.

To this end we'll need a big volume in which they can interact to produce charged particles, we'll be choosing ice for this. Even though there are a lot of volumes such as the water or rock the ice has a few particular properties which make it more usable:

- contrary to some kind of sediment, ice is mostly see-through for electromagnetic radiation but it's still possible to differentiate between neutrino signals and other particles

- contrary to water as a medium which is used e.g in the Super-Kamiokande detector [15], ice was naturally formed and it isn't needed to construct some kind of dome

2.1   Neutrino interactions in ice

As neutrinos propagate through ice they can interact weakly with the nuclei. The main mechanisms of interaction is via charged- and neutral currents [17] as is also depicted in figure 2.1.

Figure 2.1: Most prominent ways of neutrino-nucleus interaction

With the produced leptons in the W boson mediated interaction being either electrons, resulting in an electromagnetic shower, muons wich typically go undetected as they live too long or tauons wich will decay via

$$\tau^- \to e^- + \bar{\nu}_e + \nu_\tau \tag{2.1}$$

or, less ideally

$$\tau^- \to \mu^- + \bar{\nu}_\mu + \nu_\tau \tag{2.2}$$

In both of the possible interactions (W or Z exchange) the resulting nucleus will result in an hadronic shower, for the neutral current interaction (mediated by the Z boson) the fraction of the neutrino energy that gets transferred to the nucleon is described by the inelasticity y and is heavily shifted towards small values of y [1]. This causes a big, irreducable uncertainty when trying to estimate the original neutrino energy from these kinds of events. With the charged current interaction (mediated by the $W^\pm$ bosons) this isn't a problem however as the full neutrino energy ends up in the resulting cascades.

2.2 Askaryan effect

For a particle shower to emit strong radio signals, two conditions have to be met:

- There needs to be a separation of positive and negative charges in the shower front

- The signals produced over the length of the shower profile need to overlap coherently.

The *Askaryan* [6] effect, which is responsible for the production of Askaryan radiation describes the effect at radio frequencies which abides by both of these conditions. In general it's a quite difficult effect but we'll give a crude overview. The previously described interactions of the neutrinos with nuclei create a shower of secondary charged particles containing a charge anisotropy. This charge imbalance is a result of medium electrons either Compton scattering into the advancing shower or annihilating with shower positrons. In the end you have a moving charge anisotropy, propagating faster than the speed of light in the medium, creating Cherenkov radiation.

Cherenkov radiation is like the electromagnetic equivalent of a sonic boom, a sonic boom happens when something goes faster than the speed of sound in the medium; A particle emits Cherenkov radiation if it goes faster than the speed of light in the medium. Choosing the particle trajectory to lie along the z axis an approximate equation can be found [25] for $\frac{\mathrm{d}^2 \mathscr{J}}{\mathrm{d}\omega \mathrm{d}\Omega}$, which is the energy radiated per elementary unit solid angle and per elementary unit frequency interval:

$$\frac{\mathrm{d}^2 \mathscr{J}(\omega)}{\mathrm{d}\omega \mathrm{d}\Omega} = \frac{q^2}{4\pi}\sqrt{\frac{\mu}{\epsilon}}\beta^2 \omega^2 \delta^2 [\omega(1 - \beta \mathbf{e}_r \cdot \mathbf{e}_z)] |\mathbf{e}_r \times \mathbf{e}_z|^2 \tag{2.3}$$

Now we can re-write this equation in spherical coordinates, which gives $1 - \beta \mathbf{e}_r \cdot \mathbf{e}_z = 1 - \beta \cos(\theta_c)$ in the delta function. We thus only expect radiation if

$$\cos(\theta_c) = \frac{1}{\beta} = \frac{c'}{u} = \frac{c}{n} \cdot \frac{1}{u} \tag{2.4}$$

With u the local speed of light in the medium and n the index of refraction, an optical property of a medium which we'll later on thoroughly discuss. If $u > \frac{c}{n}$, Cherenkov radiation will be emitted along a cone surface with half angle $\frac{\pi}{2} - \theta_c$ as illustrated in figure 2.3. Integrating equation 2.3 over the solid angle and formally dividing by the time interval we get:

$$\frac{\mathrm{d}^2 \mathscr{J}}{\mathrm{d}\omega \mathrm{d}t} = \frac{q^2}{4\pi}\sqrt{\frac{\mu}{\epsilon}}\beta\omega \left(1 - \frac{1}{\beta^2}\right) \tag{2.5}$$
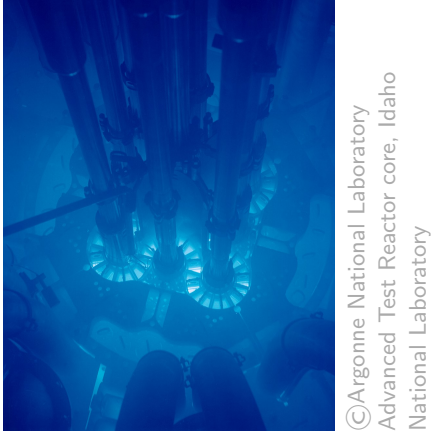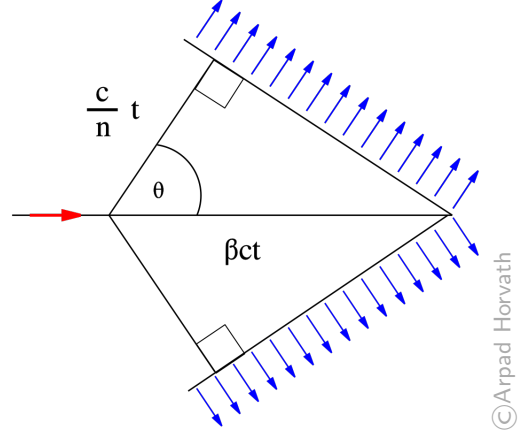
Figure 2.2: Cherenkov radiation in a nuclear reactor

Figure 2.3: Diagrammatic representation of Cherenkov radiation

We see that the energy is proportional to ω, so we expect that most radiation will be emitted "in blue" with a cut-off frequency above which the equation $\cos\theta = 1/(n\beta)$ can no longer be satisfied, this "in blue" characteristic is responsible for the blue glow seen in nuclear reactors as seen in figure 2.2 . For ice the index of refraction is roughly 1.78 in deep ice, so we expect an ultra-relativistic particle to produce the most radiation at around 56° opening as

$$\cos(\theta_c) \approx \frac{1}{n} \implies \cos^{-1}\left(\frac{1}{1.78}\right) \approx 56° \tag{2.6}$$

 Of course this is just an estimate, as the actual index of refraction is depth-dependent which we'll get to in section 3.2. Now this explains how the signals get generated but logically, from only knowing this we'd expect radio waves to almost be non-existent due to the "in blue" nature of Cherenkov radiation.

To understand why this isn't the case we'll need another piece of the puzzle: coherent overlap. Coherent overlap in the radiowave spectrum range can be intuitively explained as follows: generally the shower is of length $\mathcal{O}(10\text{cm})$ [24], over this length the radiation gets emitted, most frequencies decoherently interfering, but radio waves with wavelengths of $\approx 10\text{cm}$ coherently interfere, and it's these waves we then wish to detect. The radiation generated trough this Askaryan effect is called *Askaryan radiation*.

The Askaryan radiation is polarized perpendicular to the cherenkov cone, this can be useful to discern between different cherenkov cones whom, timely, would generate the same response. This concept is illustrated below in 2D where two neutrinos from different directions would generate the same signal in the detector.
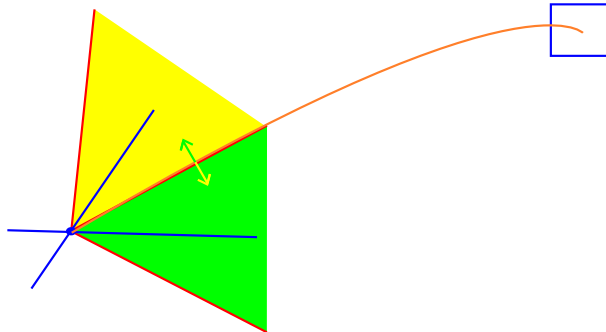
Figure 2.4: orthographic projection of Greenland with the red star representing the approximate RNO-G location

If the detector has a way to differentiate between polarization however, there would be no doubt where the neutrino originated from as the one producing the yellow cherenkov cone would have a downwards polarization and the one producing a green cherenkov cone would have an upwards polarization, note that in 3D infinite different cherenkov cones could generate the same timely signal (think of rotating the cone around a line on the cone) so both vertical and horizontal polarization information is needed.

2.3 RNO-G

Both cosmic ray and neutrino detectors face the same main problem at the highest energies: the steeply falling flux (as was previously seen in figure 1.1) requires large effective areas, which leads to the construction of neutrino detectors with volumes on the cubic kilometer scale like IceCube [3] which works on the principle of detection neutrinos with visible light. But even IceCube has it's limitations, it's still too small to observe neutrino events above the 10 PeV scale [2], that's why a new detector was needed which was even bigger and able to observe Cosmogenic neutrinos. We could just make IceCube bigger but this would cost a lot of money as the individual detectors need to be spaced closely as the attenuation length is quite short for light in the visible spectrum propagating through ice.

The proposed solution was to work with radiowave detectors, leveraging the previously discussed Askaryan effect (see section 2.2). Besides the advantage radiowaves have due to their abundance, they can also propagate way further in ice than visible light making it possible to space the individual detectors further apart. The proposed location was Greenland, an island country in North America and part of the Kingdom of Denmark which has large ice sheets. The proposal for RNO-G, which was later funded and now in the construction phase, is for it to be an array of autonomous radio stations each of which having both surface channels and various deep channels resulting in a total of 24 channels per station. The whole project builds heavily on the knowledge obtained through previous radio based neutrino detectors like the NuMoon [11] project, ANITA [31], ARA [5],ARIANNA [7] and RICE [27].
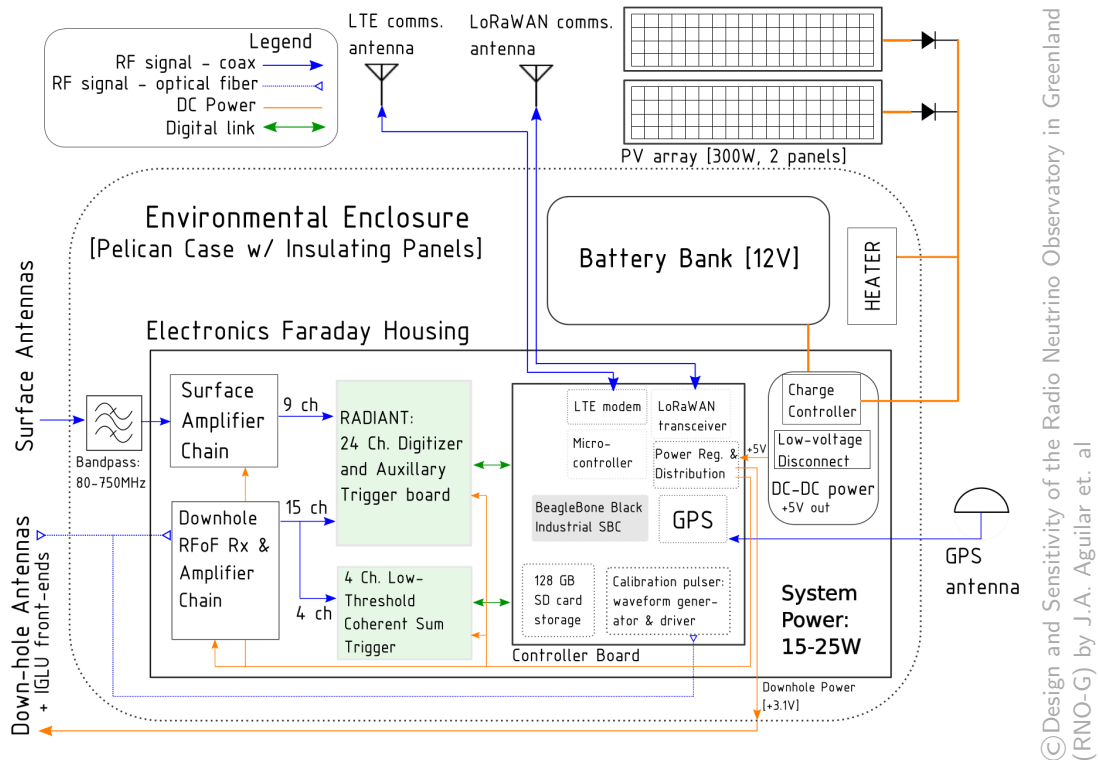
Figure 2.5: System diagram for a RNO-G station

2.3.1 Hardware

One such detector is illustrated in figure 2.7, the plan is to build 35 of these as is shown in figure 2.8[1]. Looking closely at one such detector we see just below the surface 9 Log Periodic Dipole Antennas (LPDAs), these are used to detect air shower muon signals as muons will also generate cherenkov radiation in the ice whose signals can then be filtered out. Aside form these surface detectors there are also deep components of the detector which can be split up in three parts: Two *helper strings* and the *power string*.

The helper strings are the 2 vertical cables shown on the right of figure 2.7 each housing 2 vertically polarized antennas (Vpols), one quadslot antenna for the horizontal polarization component (Hpol) and one radio pulser on each helper string which can be used to generate calibration signals. As was previously mentioned the polarization can be used to distinguish between possible cherenkov cones generating the same resulting pulse, that's why both the helper and power string are equiped with both kinds of polarized antennae.

The power string (the leftmost vertical cable) is more densely instrumented than the helper strings: At the bottom it houses a set of four Vpol and two Hpol antennas with a spacing of 1m called the *phased array*, arguably the most important channels in a station as it's spaced closely together and can thus use the principle of phasing the channels to get higher precision. Further up the string, with a spacing of 20m, are three more Vpol antennas.

A full system diagram for a RNO-G station is shown in figure 2.5, I'll give a walktrough of how data gets collected. The signal from each of the deep antennae are fed into a low-noise amplifier directly above it, from there the signal is send to the data acquisition (DAQ) system

---

[1]note that all the individual detectors are numbered but also named after various species living in greenland (in the native tongue)

14

Figure 2.6: example simulation showing both direct and refracted path from a neutrino vertex (bottom left) to a detector (top right)

at the surface via a Radio Frequency over Fiber (RFoF) cable. The signals coming from the surface antennae are first passed through a Bandpass filter of 80-750MHz[2] prior to both them and the deep component signal ending up in the RAdio DIgitizer and Auxiliary Neutrino Trigger (RADIANT), there it's again amplified, digitized and, if a trigger treshold is reached, saved onto an SD card. This data is then transmitted via a Long Term Evolution (LTE) telecommunications network to a local server[3], from where it is sent via a sattelite link.

As a power source, battery banks are used whom are charged via solar panels. But as there is't enough light during the Greenland winters, there're plans to build wind turbines (with one of the problems being the possibly detectable RF noise the 'engine' would produce).

As was previously explained the radio signal from a neutrino often travels along both direct and refracted paths, designated DnR, to the deep array (as shown on figure 2.6). This double pulse characteristic would be a smoking-gun signature of an in-ice source. The two helper strings are needed for a full direction reconstruction. Three independent measurements are needed for azimuthal information, which is provided by the Vpol (Vertical polarization) antennas and placing the Hpol (Horizontal polarization) antennas at different depths on every string, both zentih and azimuth information will be provided for those signals. The helper strings' calibration pulsers, as well as one on the surface, will ensure regular monitoring of the performance of the station and provide information useful for precise calibration of the antenna geometry.

2.3.2   Reconstruction: Lookup tables

The main simulation code we'll be using consists of 2 parts: NuRadioMC [18] and NuRadioReco [19]. NuRadioMC uses Monte Carlo simulations to generate neutrino events in the ice and simulates how they propagate to the various channels, which will be covered more in-depth in the next chapter. NuRadioReco is reconstruction software, it simulates how the various detectors would respond to the detected radiowaves. This simulation software is needed as the plan is to simulate a lot of neutrino events and record the detector responses in a giant database, then, when an actual neutrino event occurs, we'll only have to look in the database and match the actual detector response to the simulated detector responses, thus finding the origin.
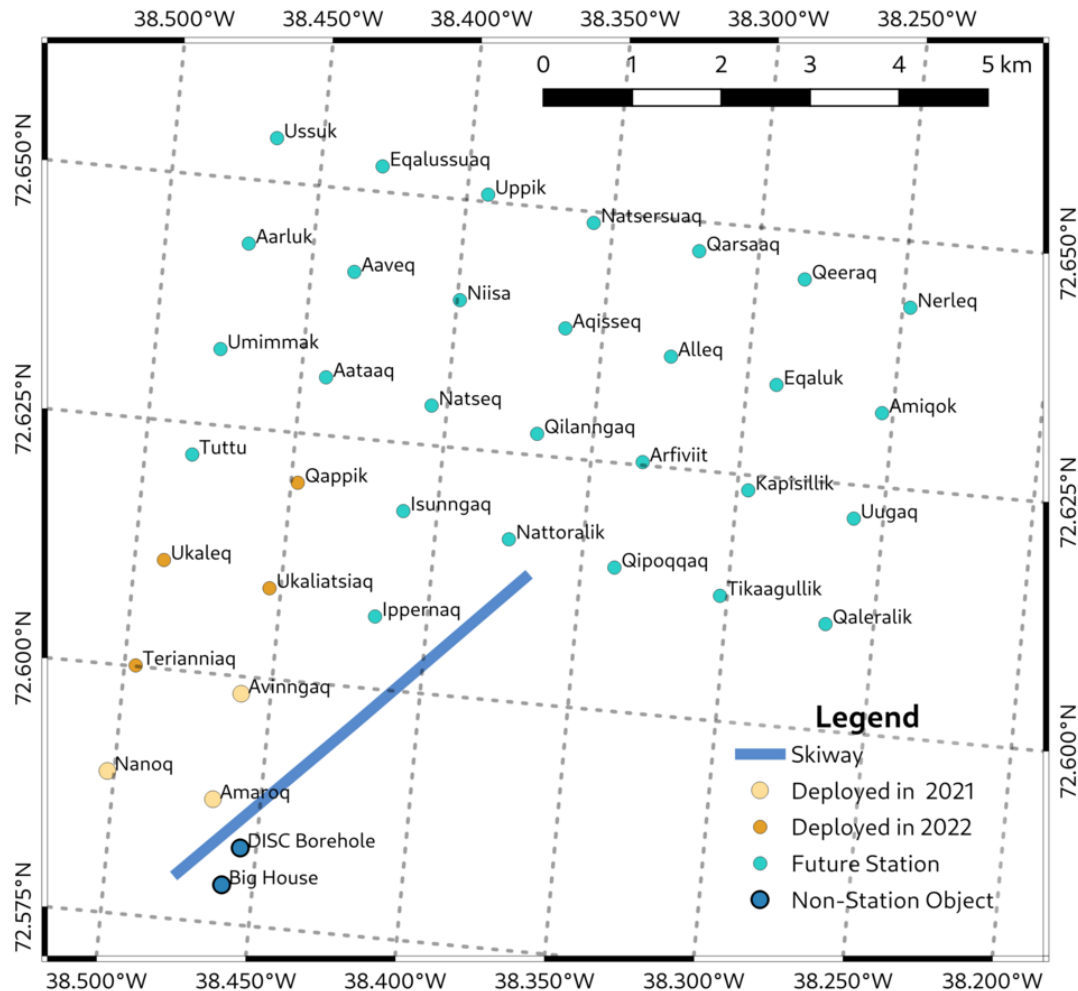
---

[2]i.e a filter that only lets frequencies in this range pass

[3]There is additionally a Long Range Wide Area Network (LoRaWAN) antenna as backup in case of problems with the LTE network

Figure 2.7: diagram showing the numbering and locations of the various antennae

Figure 2.8: planned station layout

CHAPTER

3

RAY TRACING

As previously hinted it is necessary to be able to simulate how radio waves propagate through the ice for various reasons like building up a database, calibrating channels,...

It will also become necessary in the last chapter for doing actual measurements, that's why in this chapter we'll give a brief introduction to the way radiowave paths are found within the NuRadioMC framework.
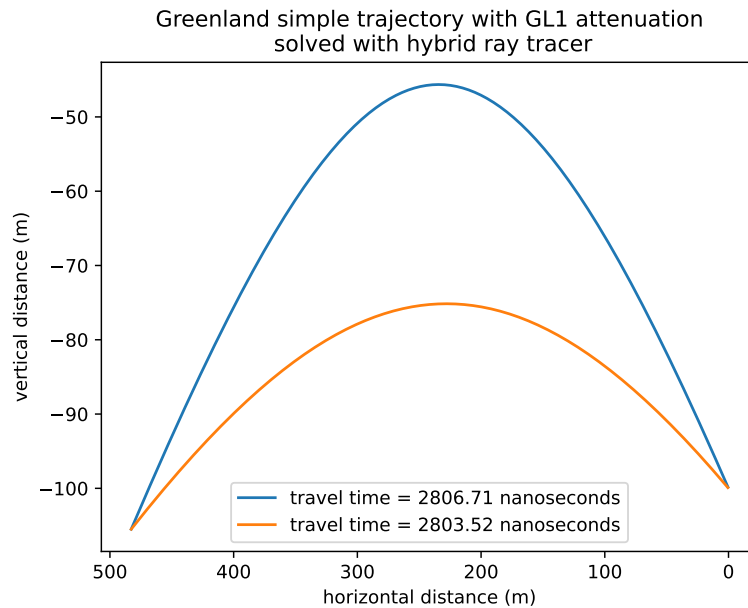
3.1 Wave propagation

Figure 3.1: illustration of radiowave paths generated by a neutrino event

The way we simulate the strongest waves propagating to the detector from a radio source is

through ray tracing, an illustration of such a simulation is shown in figure 3.1. Here the detector is located at (0,-100) and a radio source at (480,-108), note that there are two possible paths leading to the detector.

The amount of solutions and how the waves are bent are consequences of the properties of the ice we work in. In a dielectric medium a ray propagates with it's signal wave-speed determined by the local index of refraction as $v = c/n$. The effect on speed isn't the only effect the index of refraction has which we'll need to concern ourselves with however, if a ray propagates towards a boundary dividing 2 media with different indexes of refraction, the ray will refract and the refracted angle can be found from Snell's law:

$$n_i \sin\theta_i = n_o \sin\theta_o \tag{3.1}$$

Where n is the index of refraction, θ the angle with respect to the surface normal and "i" and "o" indicating incoming and outgoing respectively. The system we'll consider however, isn't homogeneous with some specified boundary, it's continous: ice in greenland has a continously varying density and index of refraction.

How do we know how the waves propagate in a medium? The software we'll be using to simulate how the radio waves behave is called *Radiopropa* [32]. As simulations of the wave propagation in full detail with the finite-differences-time-domain (FDTD) technique [33] are, even though they are more accurate, too time consuming. Have the authors of radiopropa opted to build their program on geometrical optics, i.e ray tracing. A path of a ray $\mathbf{r}(s)$ with path parameter s in a medium with index of refraction n($\mathbf{r}$) is described by the eikonal equation [21]:

$$\frac{d}{ds}\left(n(\mathbf{r})\frac{d\mathbf{r}}{ds}\right) = \nabla n \tag{3.2}$$

in radiopropa the local paraxial approximation (small angle approximation) is used, i.e if we assume that in any individual step of the algorithm the change of the refractive index along the path ds is small it's possible to re-write the equation as:

$$n(\mathbf{r})\frac{d^2\mathbf{r}}{ds^2} \approx \nabla n \tag{3.3}$$

Which is then iteratively solved using the Cash–Karp method. The way you would go about using this program is thus find a start and end point (e.g a supposed neutrino interaction point and a detector respectively), "shoot" your ray in a certain direction for which the path will then be iteratively solved using radiopropa and if you chose your direction right you have the path a ray might take from your start to the end point. The difficulty is this direction choosing which we'll get to later. If there are boundaries (such as defects or the surface) these are treated seperately using Snell's law.

3.2 Ice model

Ice has a density gradient which we'll need to account for. Due to the way the ice bonds there will be more air trapped inbetween the molecules closer to the surface than at greater depths where the pressure due to the overhead ice prevents this. Due to this air being trapped the density of ice will be smaller closer to the surface than at greater depth.

Purely from classical gravity and density considerations it can be derived that the density scales exponentially. To see this let's consider a sheet of ice in the Greenland firn with a surface A and a height dz, the extra pressure this sheet of ice exerts on the ice just below it is:

$$d\sigma = \frac{dF}{A} = -\frac{gdM}{A} = -g\frac{A\rho(z)dz}{A} = -g\rho(z)dz \tag{3.4}$$
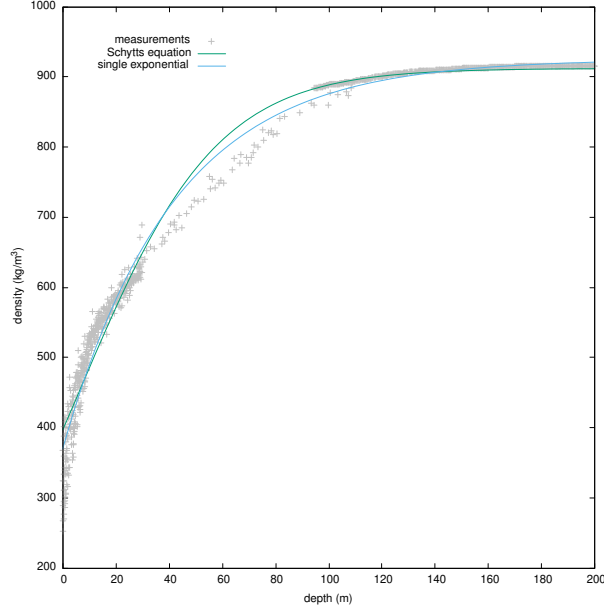
Figure 3.2: Illustration of the shortcomings of the analytical models

with $\rho(z)$ the depth-dependent density. Schytt assumed the proportional change in air space to be proportional to the change in pressure:

$$\frac{dV}{V} \propto d\sigma \tag{3.5}$$

As the volume scales inversely with the density let's assume the relation $V \propto (\rho_i - \rho)$ with ρ_i the density of pure ice, this yields [22]:

$$\frac{d\rho}{\rho_i - \rho} \propto \rho dz \tag{3.6}$$

$$\frac{d\rho}{\rho(\rho_i - \rho)} \propto dz \tag{3.7}$$

$$\frac{\ln\left(\frac{\rho}{\rho_i - \rho}\right)}{\rho_i} + C = Az \tag{3.8}$$

$$\ln\left(\frac{\rho}{\rho_i - \rho}\right) = A\rho_i z + C \tag{3.9}$$

$$\frac{\rho}{\rho_i - \rho} = e^{A\rho_i z + C} := Ae^{z/z_0} \tag{3.10}$$

$$\rho = \frac{A\rho_i e^{z/z_0}}{1 + Ae^{z/z_0}} \tag{3.11}$$

Note that Schytt worked with equation 3.7 but we found the further derivation useful. Schytt also empirically fitted the following function:

$$\rho = \rho_0 e^{z/z_0} + B \tag{3.12}$$

Figure 3.2 shows how both of these functions fit the density curve. There is one big downside however, it seems that the ice actually doesn't follow these exponential curves perfectly but would more closely follow some kind of higher order function.

Equation 3.3, and thus the path, depends on the index of refraction on a given location. The dependence of the index of refraction on density for ice can approximately be given by the Schytt
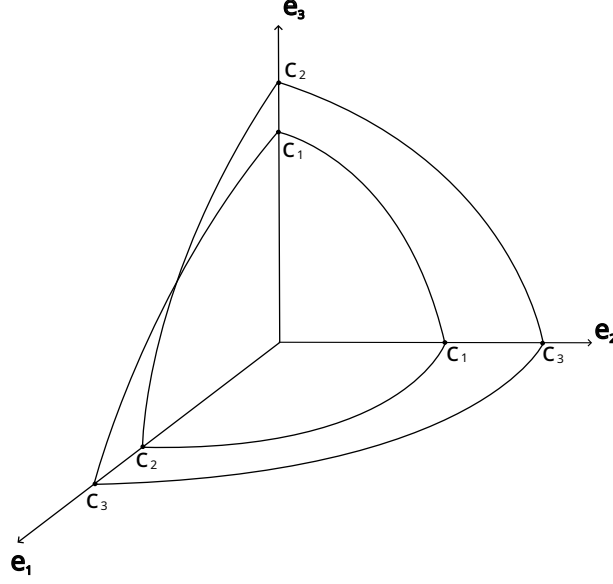
Figure 3.3: Wave surface of Fresnel with two sheets

equation [8]:

$$n(x, y, z) \approx 1 + 0.78\rho(x, y, z)/\rho_0 \tag{3.13}$$

Where $\rho(x, y, z)$ is the local ice density and ρ_0 is the density for solid ice (917 kg/m^3). For the development of the simulation software equation 3.12 was taken and after assuming Schytt's equation (equation 3.13) to hold we find that the index of refraction abides by

$$n(z) = n_{ice} - \Delta n e^{z/z_0} \tag{3.14}$$

with n_{ice} the refractive index of solid ice and $\Delta n = n_{ice} - n_s$ with n_s the index of refraction of snow. This exponential dependency of the index of refraction on depth is called the *single exponential model*.

This single exponential model has a huge advantage as it's analytically solvable, meaning that we can know which direction we'll have to shoot our ray in (as mentioned previously) after the location of the neutrino interaction and the detector are specified, the ray tracing algorithm developed using this exponential index is called the *analytic ray tracer*.

The discrepency between the single exponential model and the actual data for the density implies that the analytic ray tracer will make the wrong predictions. This is why the development of a different ray tracer was needed which will be able to handle more complex ice models, one such ray tracer will be explained in section 3.3 but this ray tracer has it's shortcomings. That's why the development of a new ray tracer was needed which is the partial work of this thesis and we'll get to that ray tracer in chapter 4.

Lastly there is an effect which might become important in the future: birefringence. Up until now have implicitly assumed that ice is isotropic meaning that both it's permittivity ε and it's permeability μ are scalars but these could very well be tensorial in nature for radio waves in ice. In general, after calculating this tensorial nature through you'd find that in every direction two different indices of refraction can be found implying two different types of waves each propagating with a different speed as illustrated in figure 3.3. Which of the two speeds in a certain direction is then dependent on the polarization of the wave, which in our case thus depends on the Askaryan effect (see section 2.2). The optical property coming from the anisotropic nature of the material
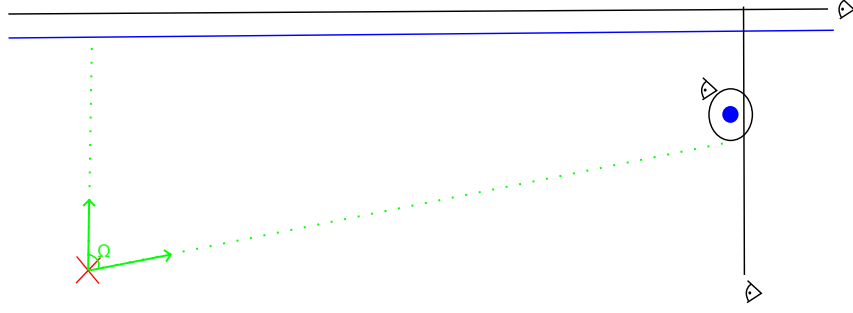
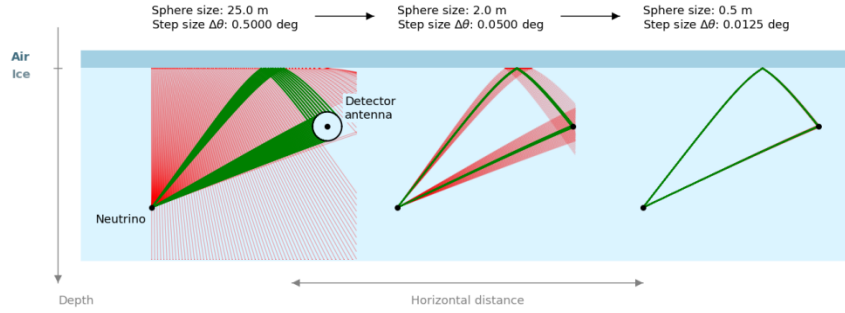Figure 3.4: Illustration of the workings of the iterative algorithm

Figure 3.5: Illustration of the steps of the iterative ray tracer

is what's called *birefringence*. Birefringence has been extensively researched for implementation in the simulation software NuRadioMC used in the RNO-G group [23].

3.3 Iterative ray tracer

The iterative ray tracer [30], as can be derived from it's name, iteratively searches the path a ray might take. The workings of the first part of the explanation is illustrated in figure 3.4. Say we have a neutrino interaction point $\mathbf{X}_i$ (the red cross on the figure) and a detector located at $\mathbf{X}_d$ (the blue dot on the figure), the algorithm starts by constructing an *observer* sphere with radius r_1 with the detector, located at $\mathbf{X}_d$, as the center. This means that any ray that gets shot and propagates to the sphere will get stopped and counts as a solution. Then, to reduce the time spent simulating, there are also observers placed whose purpuse is to stop the ray tracing but not count the observed signal as a solution. One such observer is placed just above the ice surface as a ray that escapes the ice won't be able to make it back, and one is placed just behind the detector looking from the point of the interaction vertex $\mathbf{X}_i$ as a ray is not able to reach the detector anymore after it has passed it in the lateral direction.

Finally it's noted that due to the way the ice's index of refraction continuously increases with depth, rays can't propagate upwards, this means that we only have to look for solutions within the angle Ω which is just the zenith angle the detector makes with the bottom of the observer sphere.

Now that we have our setup we'll just iteratively shoot rays from the neutrino interaction point starting at an angle $\Delta\theta_1$ then at an angle $2\Delta\theta_1$, $3\Delta\theta_1$,... Until we have reached Ω. This process is illustrated on the left side of figure 3.5. After we have gone over all the different launch angles we'll have some solutions (marked in green) and some whom don't end up on the detector (marked in red) we can now make the observer sphere's radius smaller ($r_2 < r_1$) and the step

size of the angle smaller ($\Delta\theta_2 < \Delta\theta_1$). And again iteratively find the rays which end up on the sphere, only this time looking within the angles of the solutions of the previous step. We can keep on making the sphere and step size smaller in iterative steps until we've reached the precision we want. We then take, for each bunch of solutions within an angle interval, the most normal to the detector as the final solutions.

CHAPTER

4

HYBRID RAY TRACER

4.1 Shortcomings of the exponential ice model

As mentioned in section 3.2, complex ice models will be necessary moving forward as the exponential ice model fails to fit the density curve. The software for radio wave propagation through ice the RNO-G team chose is radiopropa [32], but due to the way it works you'll have to know the start point, the end point and the launch angle of your ray to work out the path. This isn't difficult for the analytic model as it's exactly solvable but for a general ice model you'll somehow have to find where to *shoot* the ray. An algorithm was developed to work out the path the rays would trace out in more complex ice models called the *iterative ray tracer* [30], which was discussed in section 3.3. The way this algorithm works is however a sub-optimal solution in python as an optimisation library will generally work faster and more accurate, work had been done on trying to implement such an algorithm deemed the "minimizer" but this attempt failed. As we saw this work the idea came to mind to combine the iterative ray tracer and the code using the optimization libraries. This way we built the algorithm which will be discussed in this chapter: The hybrid ray tracer, in the source code called the "hybrid minimizer" which can be found here under the radiopropa/hybrid_minimizer branch.

It succeeds in more rapidly finding the path from the event to the detector, is more accurate and also arrives closer to the detector as the final result is not limited by the final drawn sphere size but by a given tolerence making it useful for simulations in which high time and space precision is needed like plane wave reconstruction, which we'll do in the next chapter.

4.2 How it works

The hybrid minimizer can be seen as an extension of the iterative raytracer as it starts out the same way: Say our source of radiation is at position $\mathbf{X}_1$ and our detector is located at position $\mathbf{X}_2$, we start by defining the vector $\mathbf{v} = \mathbf{X}_2 - \mathbf{X}_1$, then we clone it as a new vector $\mathbf{u}$ and set it's z coordinate to 0, making it a normal vector of a plane parallel to the z direction in the lateral direction of the detector. We now want to constrain the search to where solutions are actually possible, looking at figure 3.1 we see that no solutions below the direct path are possible as
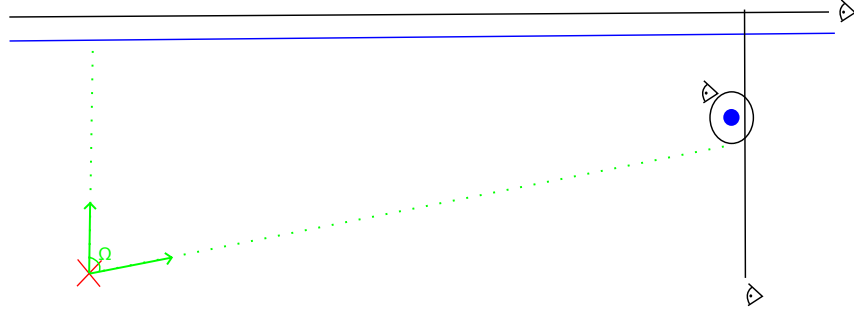
Figure 4.1: Illustration of the workings of the hybrid algorithm

there would need to be upwards reflection, so we convert our vector **v** representing the path from the source to the detector to spherical coordinates, giving us a polar angle (zenith angle) "theta-direct". With this we know that from the source the ray should propagate with an initial zenith angle lying within the angle interval 0° to theta-direct° $:= \Omega$ ($\theta \in]0, \Omega]$).

Next we need to define our "observers", if you shoot a ray with the radiopropa module from a certain point at a certain angle the ray path will get simulated until it interacts with this "observer". Ideally we would like to a priori know where to shoot our ray and have the detector be an infinitesimally small observer in our simulation, but as we'll be working with general ice models this can't be done.

The algorithm of finding the possible paths is then as follows: We define a spherical observer at the location of the detector, with a radius of fair size[1]. We place an observer plane directly behind the detector with normal vector **u** (as no rays can propagate back after passing the detector) and an observer above the surface (as no rays could make it back after escaping the ice) our full setup is then what's illustrated in figure 4.1

Here the red cross is the radio source, the blue horizontal line is the ice-air boundary surface, the blue dot to the right is the detector and the green Ω indicates the range over which solutions to the problem are possible.

We start off by just iteratively guessing: given a certain angle stepsize $\Delta\theta$ shoot rays at the angles $\{0, \Delta\theta, 2\Delta\theta, ..., \Omega\}$ And see which ones get detected at the sphere around the detector, this process is illustrated in figure 4.2.

If 2 distinct launch regions are found, say in the regions $[\theta_1, \theta_2]$ and $[\theta_3, \theta_4]$ the rays end up on the sphere (drawn in green), then the so called *minimization* procedure will start. Using scipy's module optimize.minimize the optimal solution will be found. First we get rid of the spherical observer and place the vertical observer **exactly** at the detector, now to be able to use the minimize module we'll need a function to minimze, for this reason we first define the function *delta_z* as, given a certain launch angle, propagating the ray onto the vertical observer and returning the distance from the point where it lands on the plane to the detector, as illustrated on figure 4.3 (i.e it returns the value Δz). The function we'll minimize is then delta_z_squared which just squares the value which delta_z returns as we want to do away with negative values (so we can minimize to $\Delta z = 0$). We use the previously found angle boundaries as the boundaries for this minimization algorithm, meaning that if the 2 zenith angle intervals $[\theta_1, \theta_2]$ and $[\theta_3, \theta_4]$ were found in the first step of the algorithm it will first minimize Δz^2 within the first angle interval and find it's solution and then within the second angle interval. With this our algorithm is done, it does have a fail-safe as well for if the first step, finding the launch regions, doesn't find the 2

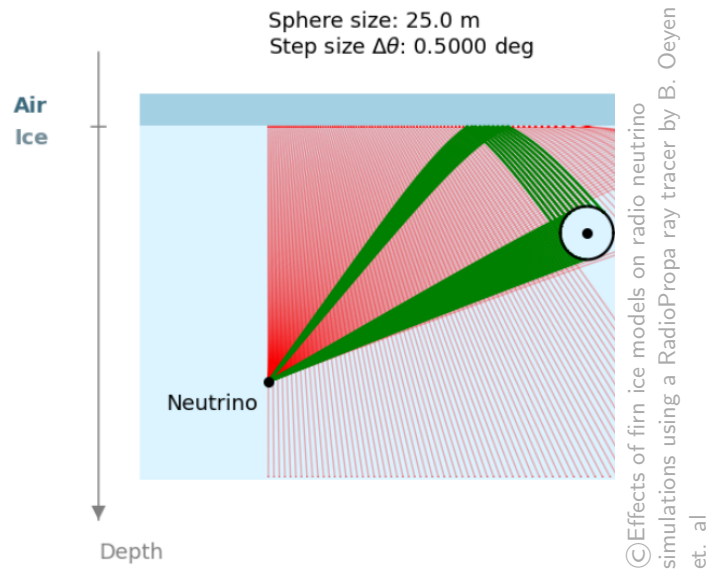[1]We'll get back to how this "fair size" gets chosen
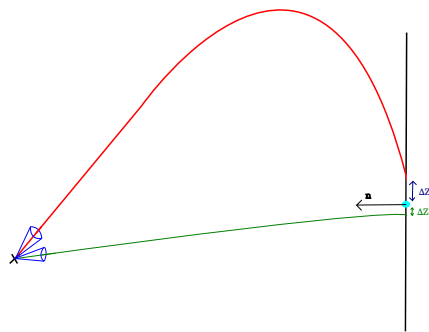
Figure 4.2: First step of the hybrid ray tracer

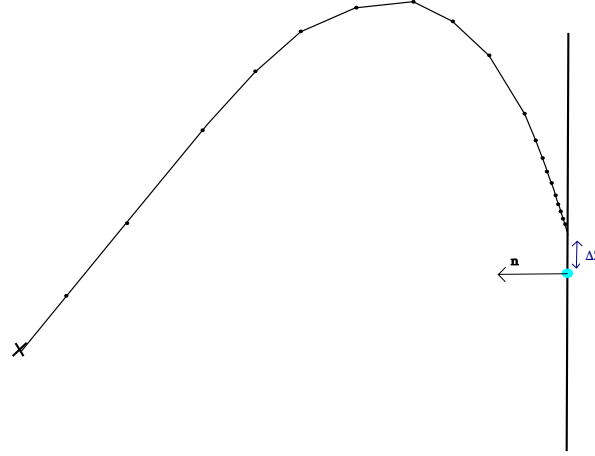Figure 4.3: Principle behind the minimization in the hybrid ray tracer

Figure 4.4: how normal vector size influences the stepsize

launch regions. Namely it reverts back to being the iterative ray tracer.

4.3 Performance Optimalisation

We wish to optimize this algorithm to make it as fast as possible. To consistently test the algorithm after every particular change in a parameter we'll be randomly generating vertex interaction positions, keeping the detector at the fixed (0,-100) value. To this end the numpy random module was used to generate random coördinates, the considered square (as there is only a z component to the ice model the 3D problem is cilindrically symmetric and thus essentially only a 2D problem) is x:0.1km,4km and z:-0.1km,-3km[2]. Every simulated point shown in the following subsections consists of at least 500 random initial positions. As the speed of the algorithm is computer dependent the algorithm's speed is always plotted relative to the iterative ray tracer's speed, simulated with the same coordinates and under equal computer load.

Now we don't want to just achieve higher speed in our algorithm, we also want to at least have the same accuracy as the iterative ray tracer, if not even better. To this end we need to check with an "exact solution". There is only one candidate that fits this role: *the analytic ray tracer*. The plan is thus to use the exponential ice model as a testing ground for the hybrid ray tracer and solving each vertex-detector ray tracing problem with both the iterative, analytic and hybrid ray tracer. After having solved for the **propagation time**, meaning the time it takes the ray to propagate from the source to the detector. And the **arrival zenith angle**, the zenith angle the ray makes at the detector, for each of the algorithms we can infer the accuracy of a particular algorithm by how much it differs from the solution found through the analytic ray tracer. For example, if the hybrid ray tracer finds a particular solution with a propagation time of 1001ns, the iterative finds one with 1002ns and the analytic one with 1000ns then the hybrid ray tracer is more accurate than the iterative ray tracer.
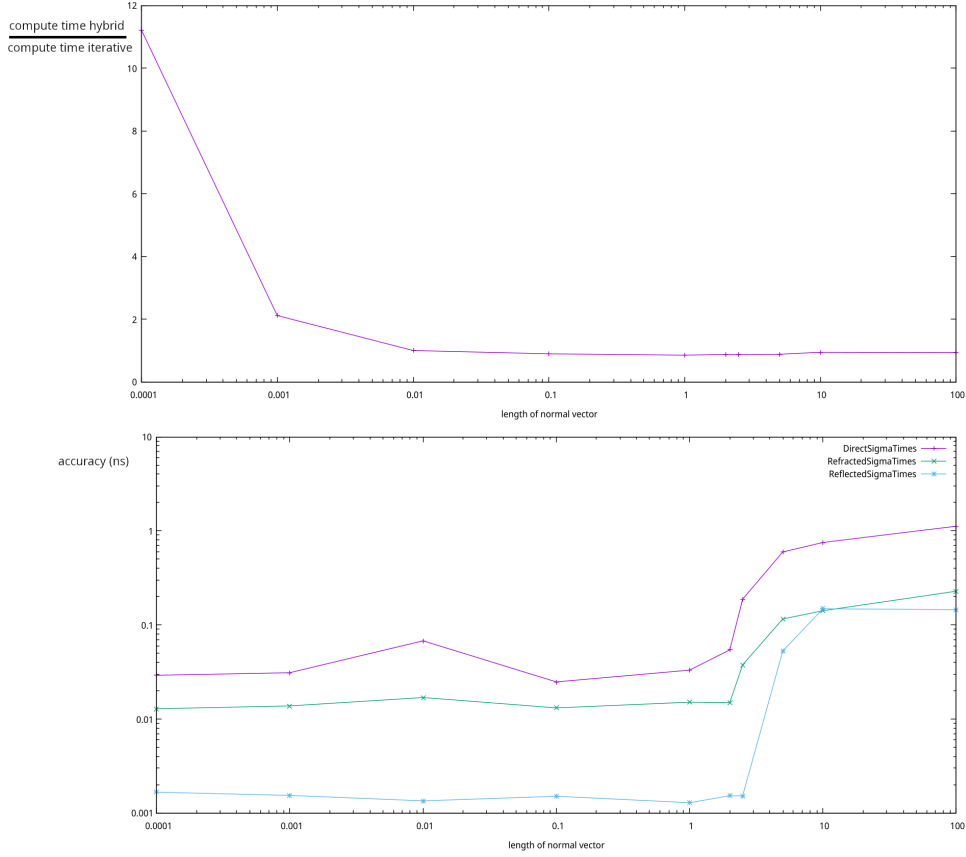
Figure 4.5: influence of the length of the normal vector

4.3.1 Length of the normal vector

Whilst figuring out what was wrong with the initial *minimizer* we stumbled on the fact that radiopropa has a little weird quirk. As visually explained in figure 4.4, the size of the normal vector seems to influence how big radiopropa's ray tracer's step size is taken close to the detector. This thus influences the accuracy and total computational time taken. The results of varying the length of this normal vector, comparing the accuracy (in timing and arrival zenith) with the analytic ray tracer and the computational time with the iterative ray tracer, are shown in figures 4.5 and 4.6. Looking at these figures we can conclude a, what would normally be rather obvious but is interesting nonetheless, first optimization conclusion: take the normal vector length to be 1 meter. We can conclude this as the calculation speed seems to be minimal there (after zooming in on the graph) and the accurace dropping rapidly after 1 meter.

4.3.2 ztol

We'll now change the *tolerance* on the vertical distance away from the detector which is deemed accepted i.e in figure 4.4 if Δz is below this threshold, the minimization procedure is stopped and the last found answer is the accepted answer. The results from analogous comparisons as previously discussed are shown in figures 4.7 and 4.8. There seems to be a minimum at 0.05m, looking at the accuracy for this tolerance it seems sufficient to be used. We can thus infer the second optimization conclusion: take ztol to be 0.05 m.

---

[2]This start at 100m depth was to get around issues concerning events that won't even trigger in a full simulation
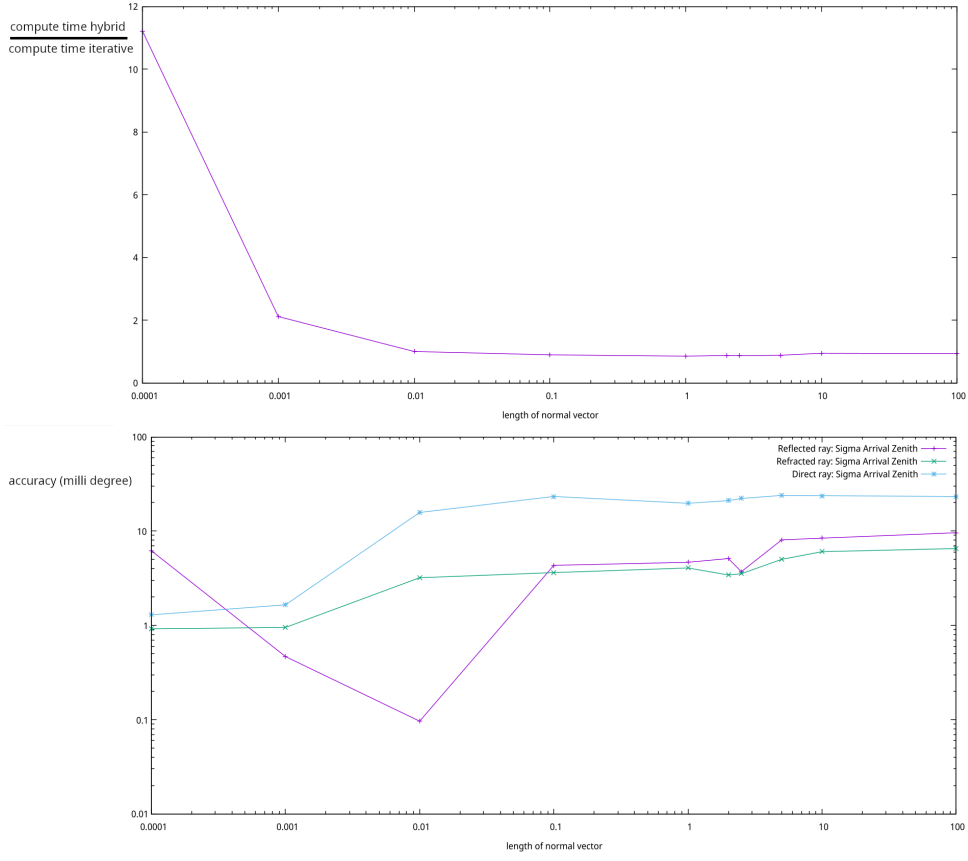
Figure 4.6: influence of the length of the normal vector
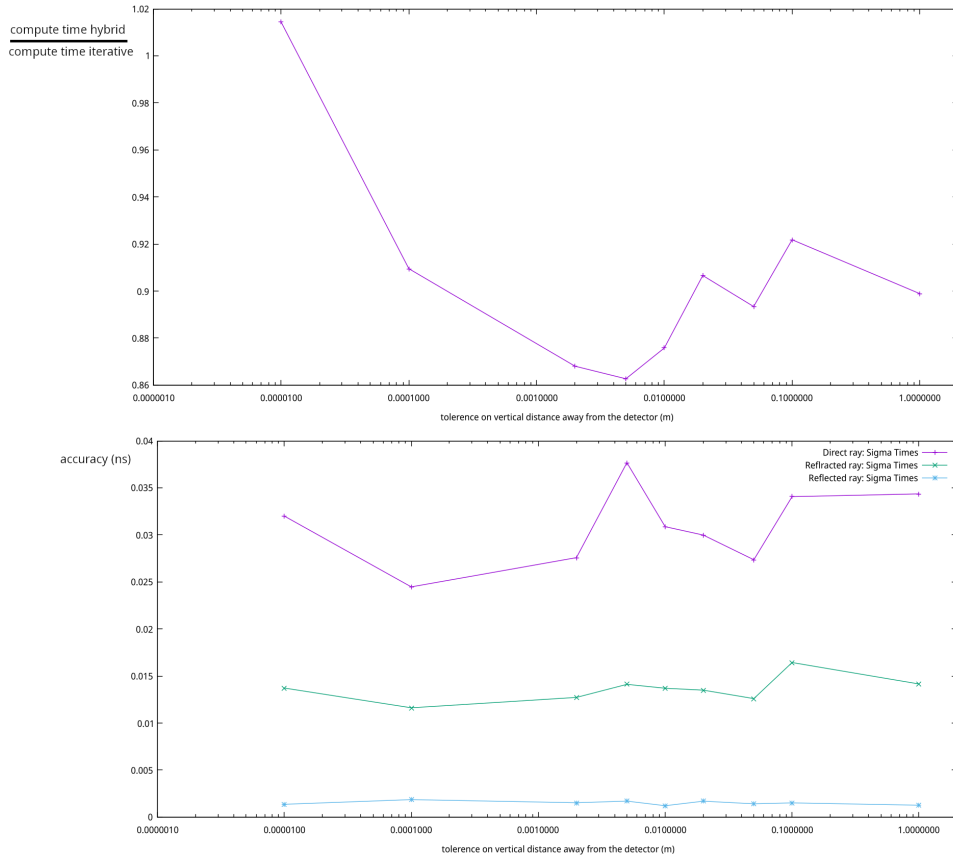
Figure 4.7: influence of the tolerence on vertical distance
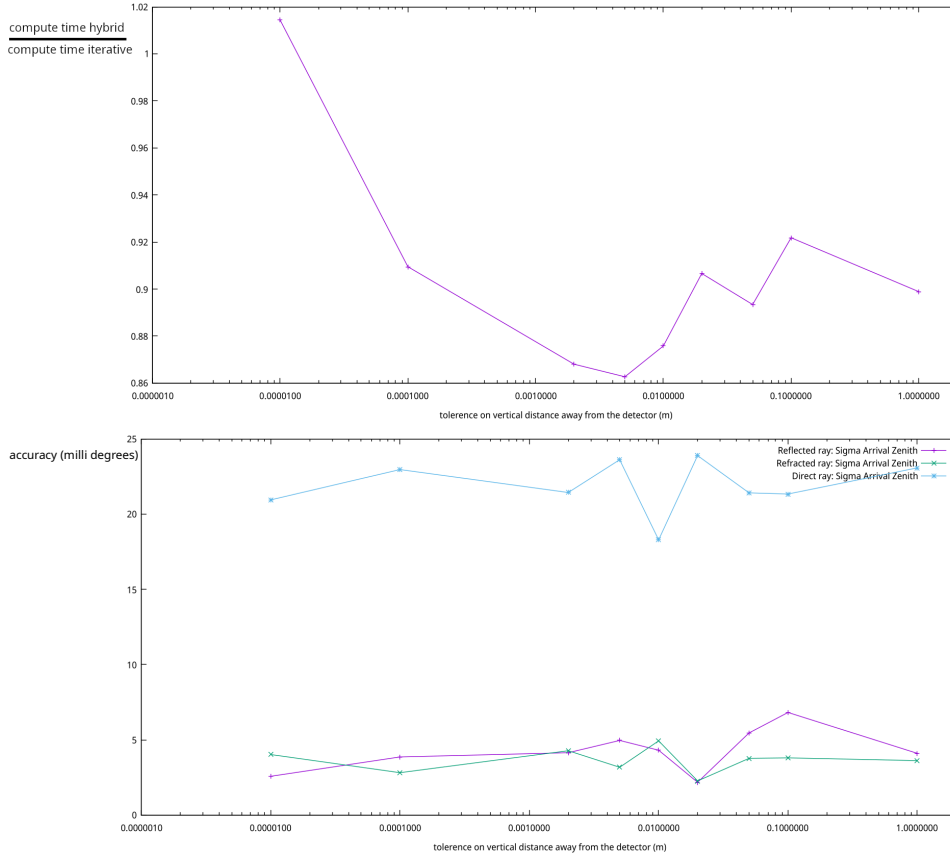
Figure 4.8: influence of the tolerence on vertical distance

4.3.3 Sphere Size & Step Size

The initial rays are sent out in steps of a certain angle and with a sphere around the detector of a certain size, varying the parameters of this first step of the algorithm (e.g the step size and sphere size) might thus have an effect on both the accuracy and the computational time. As this initial search for launch angle regions is also the slowest step in the hybrid ray tracer it's also the most important step to optimize. The optimization procedure is as follows: change the sphere size and loop over various step sizes, recording the speed. Note that, as we have to vary 2 parameters, we can't really look at accuracy here. This isn't needed however as we'll see that after finding the optimal solution the accuracy is still superb, apart from one part where it is needed: **if insufficient solutions are found**, as we wish this algorithm to best the iterative ray tracer, it'll need to at least find as many solutions as the iterative ray tracer does. If, for a particular random interaction vertex location, less solutions are found by the hybrid ray tracer than for the iterative ray tracer, this parameter choice will be thrown away (and colored differently). The results of going through this are shown in figure 4.9, the points colored red are inusable as previously discussed.

The lower on the graph the points lie, the better. If we combine all the green points into a single plot we get what is shown on figure 4.10, zooming in onto the lowest point , we see that an optimal sphere size seems to be at 45m accompanied by a stepsize of 0.7°.

The programs used in making these plots can be found here under the folder "testhybrid".

Figure 4.9: Variation in Sphere and angle step size with report on relative time.

Figure 4.10: Green values in variation in sphere and angle step size, max speed (minimum on figure) at 0.7° as stepsize and 45m spheresize
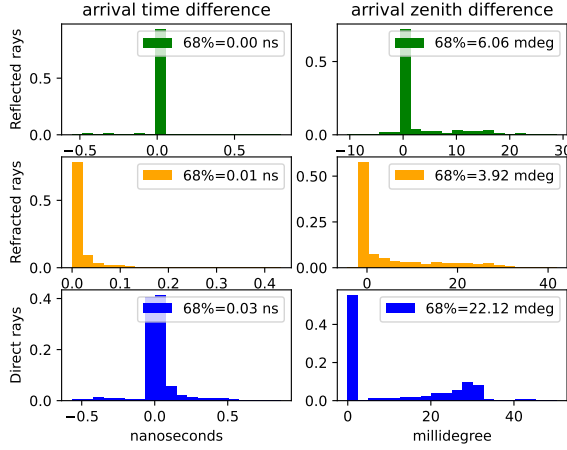
Figure 4.11: Hybrid
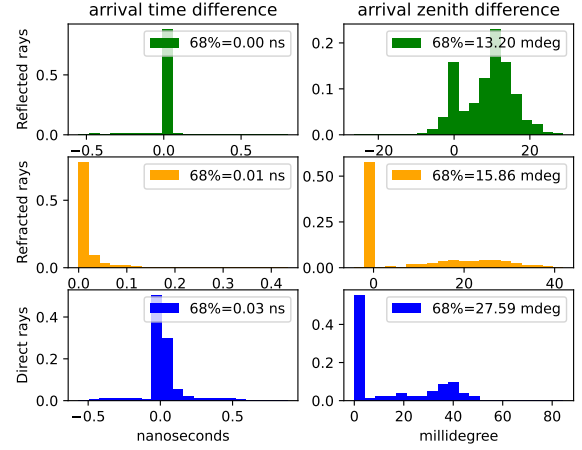
Figure 4.12: Iterative

4.3.4 Conclusion of optimization

How does the hybrid ray tracer perform relative to the iterative ray tracer if we use the previously optimized variables? After doing a simulation of 1000 random s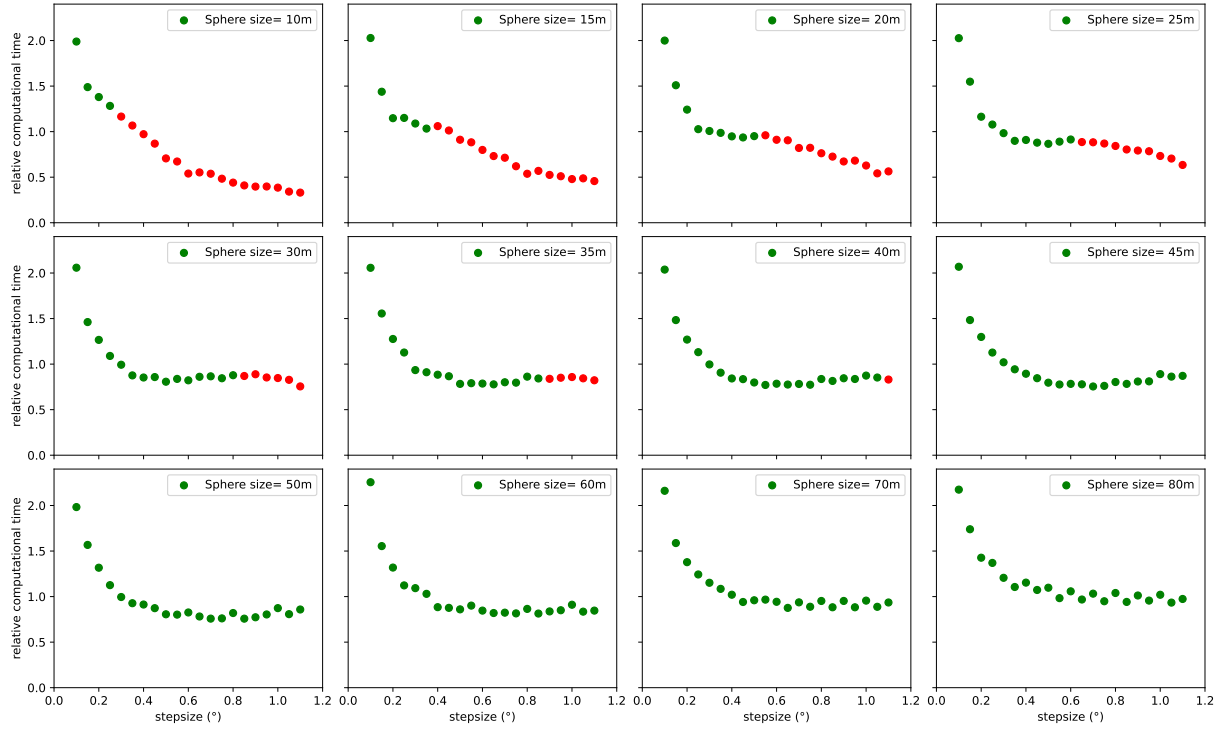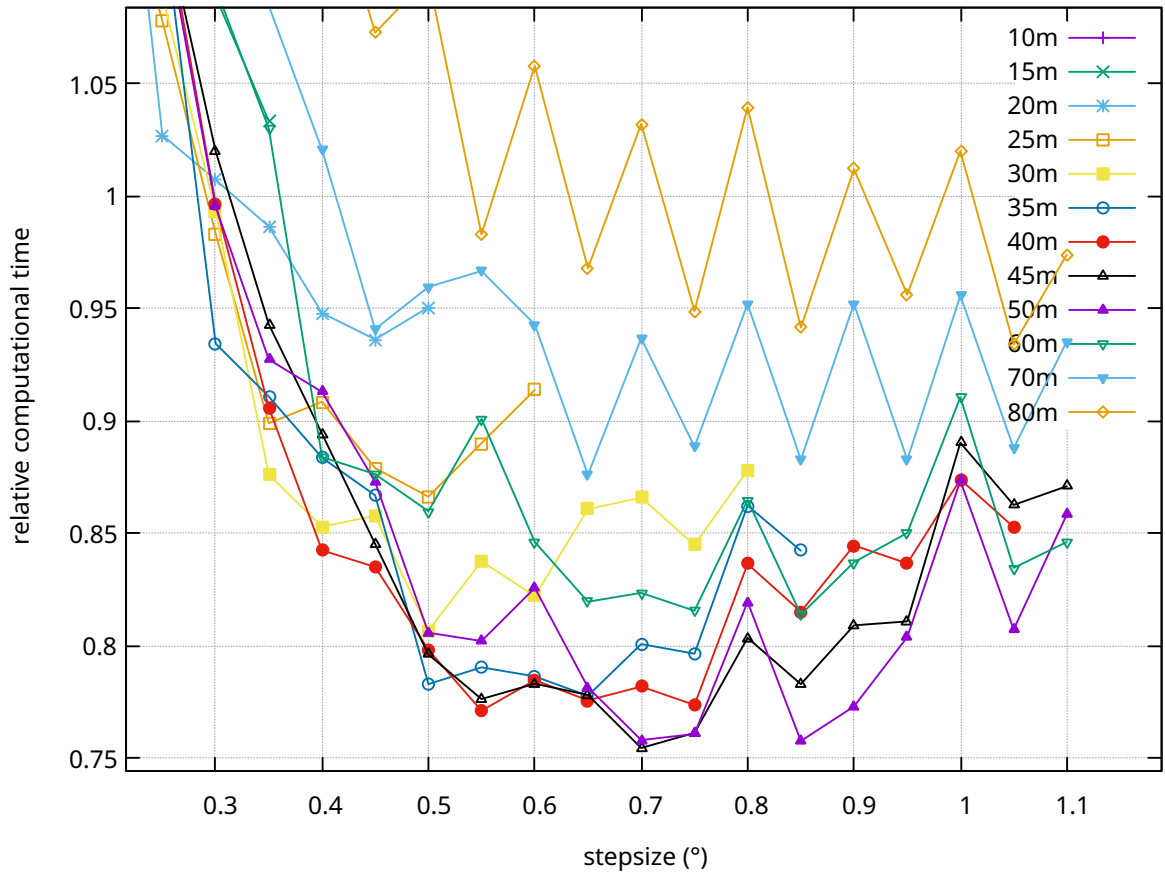ource locations with the iterative, analytic and hybrid ray tracer and comparing both the iterative to the analytic and the hybrid to the analytic we get what's shown in figures 4.11 and 4.12. Where we can clearly see that the hybrid ray tracer is more accurate, now during this simulation we also recorded the computational speed, the results are presented below:

- iterative: 0.61 computations/s

- hybrid: 0.82 computations/s

- analytic (c++ compiled version): 10289 computations/s

The hybrid ray tracer is thus 33.7% faster than the iterative ray tracer, the speed of the analytic ray tracer has a difference of the other two which might seem quite overwhelming but this can get mitigated in the future if part of the hybrid ray tracer gets re-written in a compiled language (like c++).

4.4 Expansion

Using the ray tracer as it is now defined makes us unable to calculate paths by *secondaries*. When your ice model has some kind of added complexity like a reflective bottom or a discontinuity layer (like what will be the case for our air-bourne balloon in the next chapter) you'll need to account for secondaries, these are additional rays like an extra refracted or reflected rays that get generated at this discontinuity. How the algorithm is now it will follow one ray from the source to the discontinuity and only consider, say, the reflected ray. Whilst a refracted ray generated at such a discontinuity might actually end up closer to the detector.

To implement secondaries, we'll just check if there are any found at the delta_z algorithm and if so loop over them, keeping the original ray's Δz in mind as the "global best" and looking if any of the secondary rays' Δz is lower than the "global best" if so this will be considered the new final ray which gets returned and the Δz that gets returned is the new "global best" originating from the secondary. If there is no secondary ending closer than the "original" ray than the original ray with it's Δz is returned.

5

WEATHER BALLOON

In this chapter we'll infer information about the ice the detectors reside in using radio signals coming from a weather balloon flying over the stations. We'll go about this by first finding the difference in timing between channels by fitting the balloon's sinusoidal signal to the data, and then doing a plane wave reconstruction for different indices of refraction until we find one where the reconstructed signal gets close to the balloon. This procedure will be explained more in depth over the following sections.

For us to find the timing we'll first make an educated guess, this can be done using our *hybrid minimizer* algorithm, there is one change that need to happen first to our algorithm for us to be able to simulate this: The air observers needs to be hard removed as to make a ray tracing possible from within the air into the ice and onto the detector. The modified version of the hybrid ray tracer can be found here under the branch radiopropa/BaLLooN.

Due to the way our algorithm works the time the ray took is only from when it became a secondary[1], this information is insufficient as we want the full travel time. The propagation time from the balloon to the ice t can later be added to the time of the path in the ice by measuring the length d of the drawn line from the balloon to the beginning of the recorded path, setting the speed of radio waves in air to the speed of light c and then just adding $t = d/c$ to the time found from the algorithm.

5.1 Plane Wave Reconstruction

Now having modified our ray tracer, the first problem we'll consider is plane wave reconstruction of the original position, an example path to some of the deep sensors is given in figure 5.1 The plane wave reconstruction can easily be understood using figure 5.2, the radio waves coming from the weather balloon are drawn in blue and make a certain angle with the detectors. the top detector (top box) detects the wave at a certain time t_1, the bottom detector detects it at a time t_2. In our database, after decoding the signal we'd thus see that these two detectors got

---

[1]This is a problem which also seems to hold in the iterative ray tracer, it might get fixed in the future but as secondaries weren't needed for most applications we deemed it fine for the moment
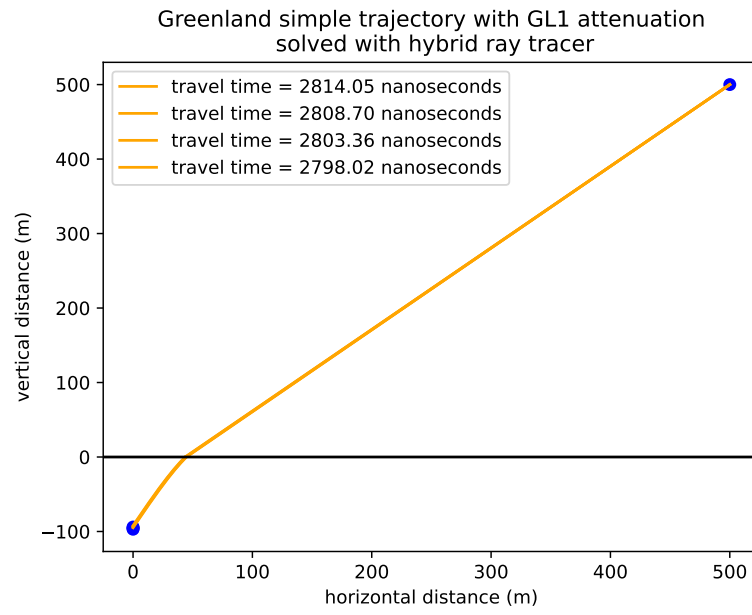
Figure 5.1: Example trajectory of rays coming from a weather balloon (blue dot top right) and going through the ice to the various detectors (blue dots bottom left)
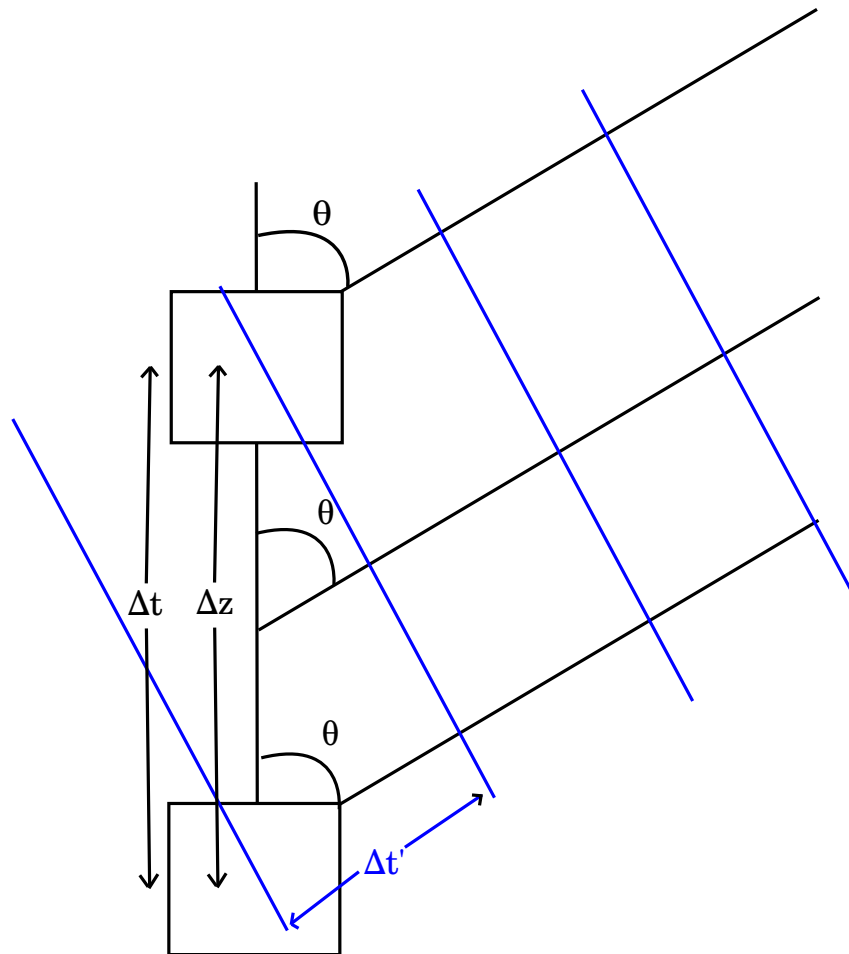

Figure 5.2: Illustration of Plane waves

a signal $\Delta t = t_1 - t_2$ seconds apart from eachother. Now ideally this is equal to the time $\Delta t'$ which is the time it took the wave to propagate that distance which we can calculate from basic trigonometry and dimensional analysis:

$$\Delta t' = \frac{m}{(m/s)} = (s/m) * m = v^{-1} * m = v^{-1} \cos\theta \Delta z \tag{5.1}$$

With $v = c/n$ the local speed of light and n the index of refraction at the depth inbetween the considered channels. Thus if we know the angle θ, the time $\Delta t'$ and the distance between the channels Δz we could infer the index of refraction at a certain depth as can be seen by reforming equation 5.1:

$$n = \frac{\Delta t'}{\Delta z} \frac{c}{\cos\theta} \tag{5.2}$$

In reality we don't know the angle a priori, we'll only have the recorded timing information Δt and the difference in channel height Δz. So to this end we'll perform a scan by choosing some index of refraction n_1 and minimizing the *correlation function*:

$$Correlation(\theta) := \Delta t - \Delta t' = \Delta t - \frac{\cos\theta \Delta z}{v} \tag{5.3}$$

If we want to use more than 2 detectors at once however (which we'll do for the phased array), we'll need to specifiy various correlation functions. E.g if we have four detectors labeled 0 to 3 we'll have to construct correlation functions between detectors 0&1, 0&2, 0&3, 1&2, 1&3 and 2&3 . As all of these correlation functions will have different sizes we'll norm them as follows:

$$Correlation_{Normed}(\theta) = \frac{Correlation(\theta)}{\int Correlation(\theta)\Delta\theta} \tag{5.4}$$

An example simulation of these correlation functions is shown in figure 5.3, notice how you can't differentiate between the correlation functions, this is only possible because of the hybrid ray tracer having that high of a precision. After this we can sum them, as shown in figure 5.4, and look at which angle it reaches it's minimum. Using this angle we can then reconstruct a ray and guess where the weather balloon is approximately, this is illustrated in figure 5.5. We can calculate the difference in distance between the weather balloon and the plane wave at the same height and label this d_1, then we can choose another index of refraction n_2 which again corresponds to a difference d_2 and so forth. At the end we take the index of refraction n_i which corresponds to the minimal distance d_i as the fitted index of refraction.

5.2 Is the goal feasible?

5.2.1 No assumptions

In the example reconstruction illustrated in figure 5.5 the difference in angle between direct to balloon and plane wave reconstruction is already quite small (0.65329617%) but as the balloon gets closer to the detector this reduces significantly. As previously layed out, our goal is to find the local index of refraction n by using the plane wave reconstruction with the recorded timing and the positional data from the weather balloon.

Now let's ask ourselves the question, within which angles should the weather balloon fly for the data to be useful? As was previously stated, the further the weather balloon is away (in the x direction) the bigger the zenith angle with the detector the less accurate the plane wave reconstruction. So which angles are acceptable? Note that not only angle but also height will eventually play a role in the accuracy, the angle however gives a good starting point. To determine this our method works as follows:
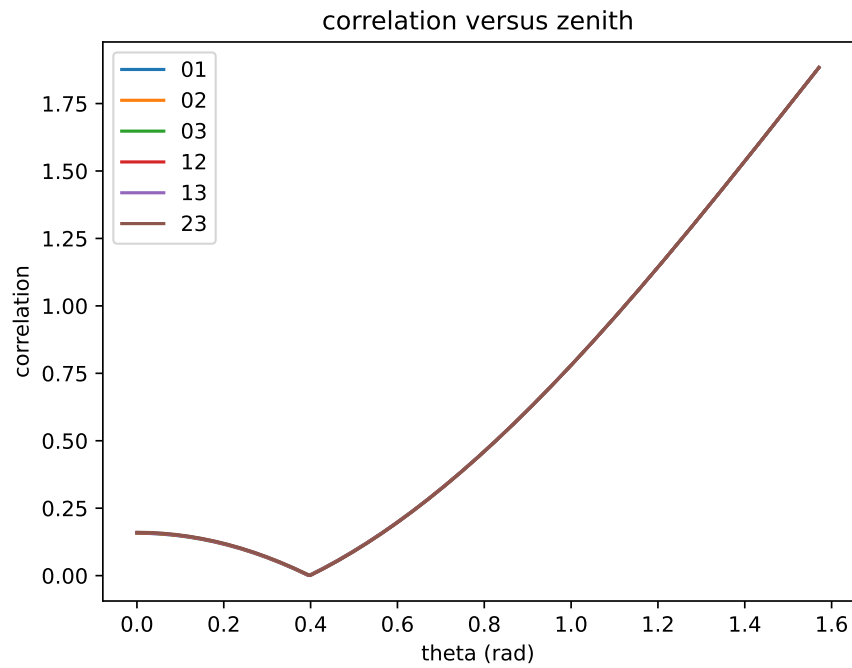
Figure 5.3: Example normed correlation functions
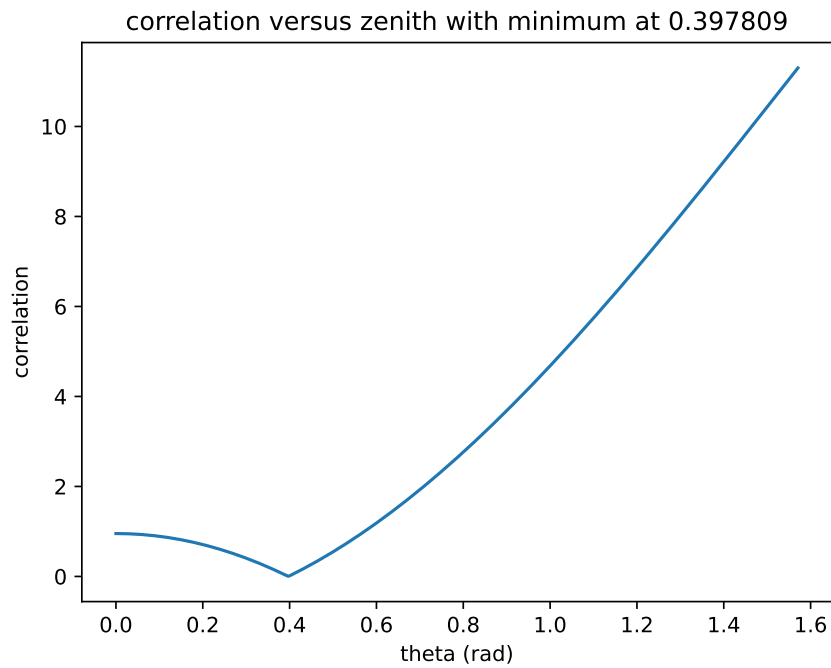
Figure 5.4: Example sum of the normed correlation functions
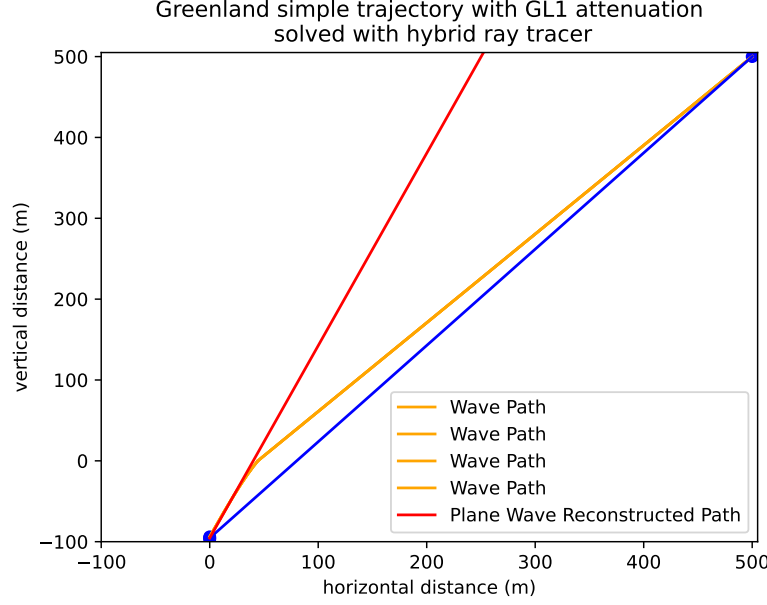
Figure 5.5: Example plane wave reconstruction of weather balloon position

We vary the position of the weather balloon in the x direction (keeping the height constant at 500m), simulate the ray path to the channels 0 to 3 and then fit n such that the difference between the balloon position and the plane wave at the same height is as small as possible. Then we compare the n we have fit to the one we know from the model at that position. We quantise the discrepency between these two indices of refraction using what we here define as the *correction factor*:

$$\varepsilon\ (\%) = \frac{n_{\text{fit}} - n_{\text{exact}}}{n_{\text{exact}}} \times 100 \tag{5.5}$$

Carrying this out[2] we get figure 5.6, i.e it gets exponentially shifted towards higher n as the balloon moves further away. If we wish our correction factor to be within 1%[3], the angle the balloon makes with the middle of channels 0 to 3 needs to be less than 10°.

Now what would our <10° policy entail? And is it even possible? Say we take a look at station Terianniaq (station 12) and see what our policy entails, it is located at 72.6000868058195° N -38.4962265332872° W in the global geographic coordinate system (longitudinal, latidude and elevation), this is quite difficult to work with directly. For this reason we first convert the station's location to local ENU coordinates (east, north, up) relative to the the 'DISC' which is a point in the base camp located at 72.58279265212887 latitude, -38.45581495328228 longitude and 3251.9489147560234 elevation.

After having converted station 12's position into ENU coordinates we get -137.67250176003688N 1727.5184983294744E, we then set our "up" to be -95.5m as this is the approximate location of the middle of channels 0-3. Now looking at e.g the Balloon path recorded on the 20th of august 2022 (figure 5.8) we see that the balloon crosses paths closely with detector 12, but how close was this encounter? To quantize this we can take a look at every balloon positional data entry (there are >12000 entries) individually and compute the angle the balloon makes with the detector by first converting the location of the balloon into ENU coordinates, then calculating the lateral distance ($\sqrt{(x-x')^2 + (y-y')^2}$) then the relative vertical distance $|z - z'|$ and then

---

[2]The code for this can be found in projects-mt/BaLLooN/simulations as plane_wave.py
[3]we want this as small as possible as to not rely on the simulation
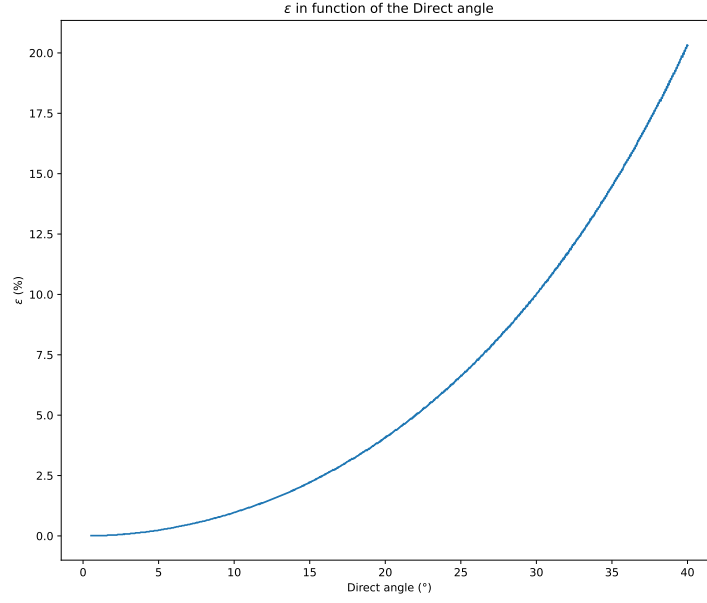
Figure 5.6: Epsilon in function of the direct angle

from those compute the angle $(\tan(\theta) = \frac{Hor.}{Vert.})$ doing this and recording when the balloon gets close enough to get below the 10° mark we get the yellow part of figure 5.8. We thus see that in this example the $< 10°$ policy is viable.

Now how much data do we have that way? We'll be looking at the data recorded over the summer of 2022, more particularly 15/06/2022 - 30/09/2022. The positional data of the weather balloons was obtained from the `ftp1.esrl.noaa.gov` website using the rno-g-sonde script of the official RNO-G github page. After looping through every weather balloon gpx file recorded in the summer of 2022 and seeing when it gets within 10 degrees from any detector we get the data shown in appendix D, even though this is quite a lot of data there's still another step that we could do to broaden the amount of usable data.

5.2.2 Refraction at the surface

Up until now we have not used the property that waves refract at the surface as we didn't want to assume anything, now say that we include refraction at the surface for our plane wave reconstruction. This would mean that we'd follow analogous steps as our previous analysis, i.e doing a plane wave reconstruction from the difference in timing and trying to fit the index of refraction, only now the plane wave abides with snell's law at the surface, going from n=1.27 to n=1.

The full algorithm thus goes as follows: We first reconstruct the plane wave launch angle θ_1 by minimizing the correlation function previously defined, we can use this angle and the height of the middle of the concerned detectors to derive the function

$$z = a_{InIce} * x + b_{InIce} \tag{5.6}$$

The outgoing zenith angle at the surface θ_2 can be calculated from snell's law:

$$n_1 \sin\theta_1 = n_2 \sin\theta_2 \implies \theta_2 = \sin^{-1}\left(\frac{n_1}{n_2}\sin\theta_1\right) \tag{5.7}$$
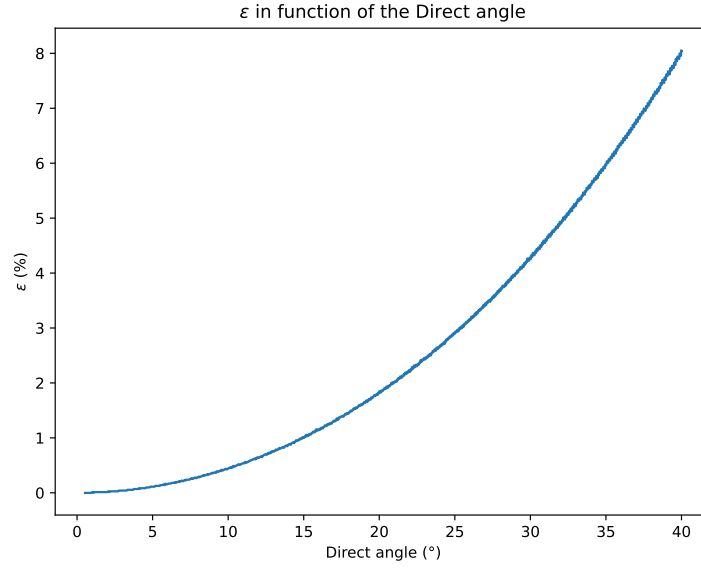
Figure 5.7: epsilon IFO possible angles from channels 0-3 with refraction at the surface

from this we know the slope of the wave path $a_{InAir} = \tan\left(\frac{\pi}{2} - \theta_2\right)$ but not the offset, but this can easily be found from equation 5.6 as

$$z = 0 = a_{InIce} * x_{End} + b_{InIce} \implies x_{End} = -\frac{b_{InIce}}{a_{InIce}} \tag{5.8}$$

and

$$z = 0 = a_{InAir} * x_{End} + b_{InAir} \implies b_{InAir} = -a_{InAir} * x_{End} \tag{5.9}$$

$$= a_{InAir} * \frac{b_{InIce}}{a_{InIce}} \tag{5.10}$$

Now that we have the function describing the "path of the plane wave"[4] in the air we can find out the horizontal position at the height of the balloon as

$$z = z_{Balloon} = a_{InAir} * x_f + b_{InAir} \implies x_f = \frac{z_{Balloon} - b_{InAir}}{a_{InAir}} \tag{5.11}$$

and iteratively loop over possible indices of refraction, minimizing $|x_{Balloon} - x_f|$.

Doing this whilst looping over possible horizontal balloon positions, we get the result shown in figure 5.7[5] As you can see we can now go up to 15° and still have a correction factor of less than 1%! The only drawback of this method is that we need to assume the index of refraction to be 1.27 at the surface of the ice, if this isn't the case in some places our predictions won't be accurate.

5.2.3   Influence of height on Epsilon

In the previous sections we've assumed the height of the weather balloon to be 500m, does changing this height influence the behaviour of ϵ in some way? The answer is, at first surprisingly, yes. As can be seen in figure 5.9 the correction factor can vary with height for the same angles. This has, as a consequence, that if we manage to get a hold on some usable data the ϵ might still

---

[4]we use double quotes as to emphasize that this is a reconstruction method and not an actual wave

[5]The code for this can be found in projects-mt/BaLLooN/simulations as plane_wave_with_snell.py
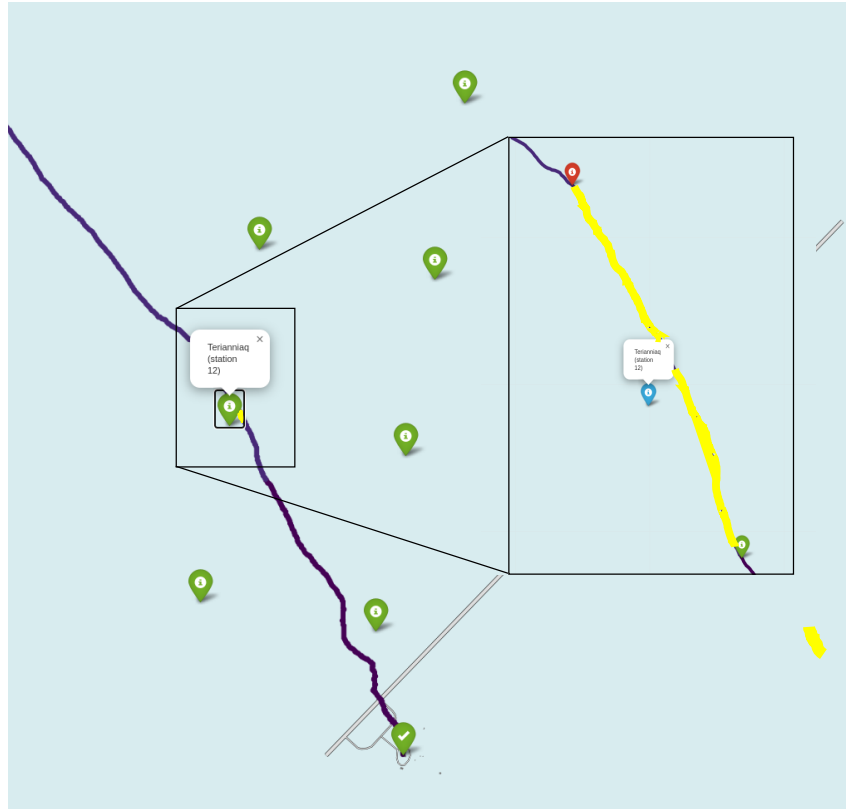
Figure 5.8: Path traced out by a weather balloon in purple released at 20/08/2022, in yellow you can see when it gets within 10° of detector 12.
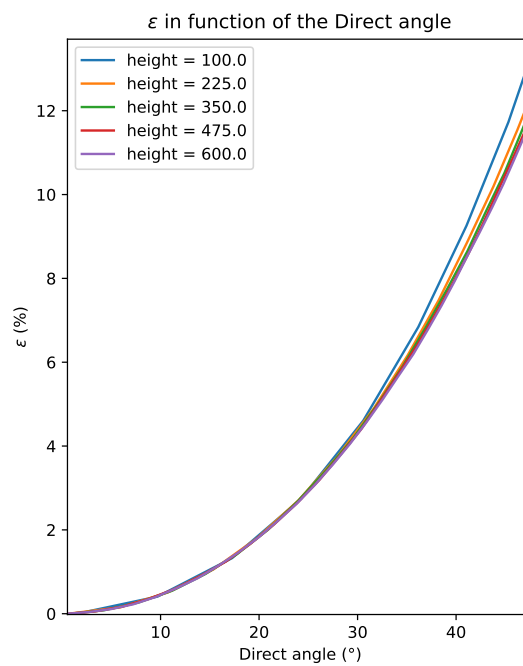
Figure 5.9: How varying the height influences ϵ over the same angles

be quite large (if the height is really large). Because of this we'll have to re-compute ϵ exactly for every measurement we do.

5.3   Fitting the index: Phased array

Now that we know our goal to be feasible, let's analyse some data. As we'll start by just analysing one event, let's take a very good one for our analysis. A close look at graph 5.6 shows that events under 5° at a height of 500m produce an ϵ of less than 0.22%, implying that if we measure n to be 1.7407 our correction factor will only be 0.0038%. A good start will thus be finding moments when the balloon gets to within a 5% angle, note that the correction factor is height dependent so we'll have to calculate it afterwards. After looping through all the balloon positional files and only outputting the $< 5°$ ones we get the timeframes shown in appendix C.

If we search in the DESY database [6] within the calculated timeframes for the particular detectors where the balloon gets close enough to, AND where the 403MHz signal coming from the Ballon is detected in the deep channels, the events of the 7th of september stands out; at 11:28:10, just before the $< 5°$ passby between 11:28:47 and 11:28:49, the balloon gets quite close to detector 23 and shows a clear 403MHz signal at the phased array, as shown for the channels 0 and 3 in figure 5.10.

Now to calculate the differences in timing for this received signal, the code we built for this is called FitN.py and stored at the repository projects-mt under BaLLooN/RealData, let's go over the full code step by step:[7]

5.3.1   Spatial data

The first part we'll need to concern ourselves with is determining the relative positions of everything. The balloon data file and the time when the event took place are given, from these two both the latitudinal and longitudinal position and the elevation of the balloon at the given time are determined. We convert these three measurements to the ENU coordinate frame (x,y,z with respect to reference) and store it in the array *BalloonPosition*. The next step is to get the location of the detector, for this we first instantiate a detector object

```
1  det = NuRadioReco.detector.detector.Detector(json_file="RNO_season_2022.json")
```

The RNO season 2022 json file can be found on the official RNO-G github under "analysistools", then we update the detector to the time of the event and get the absolute position of our detector (station 23) via the get_absolute_position module:

```
1          stationlocation = det.get_absolute_position(station_id)
```

Where at the specific station this position is doesn't matter as we'll explains shortly. Now that we have both the position of our balloon and station in ENU coordinates, let's simplify the calculations by setting the detector as the origin. i.e our new balloon position will be

$$RelBalloon = BalloonPosition - StationPosition \tag{5.12}$$

---

[6] https://rnog-monitor.zeuthen.desy.de/
[7] I advise the people who will continue this work to pull up the code side-by-side with this explanation
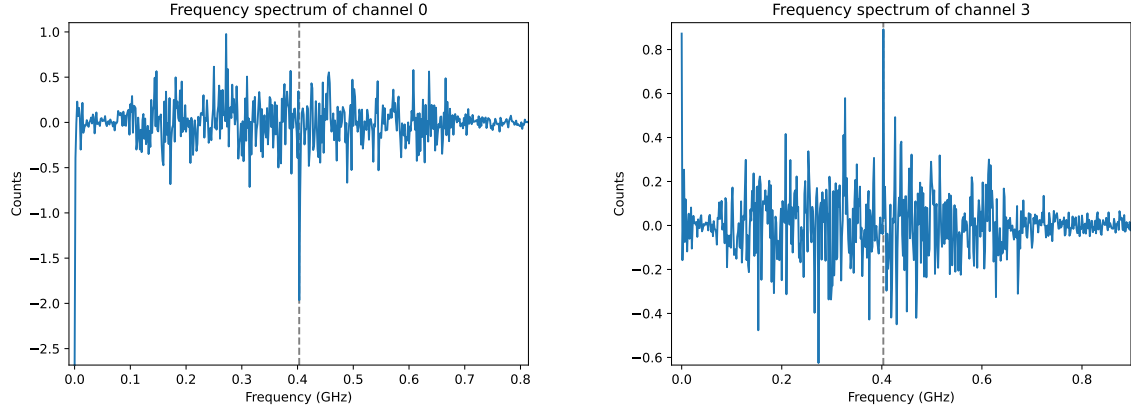
Figure 5.10: recorded frequencies on detector 23 at 2022/08/29 11:18:32

And as we might want to plot this later, due to the cylindrical symmetry of the ice model, we can rotate the coordinates to get rid of our y-axis. We can do this by defining the radius:

$$r := \sqrt{RelBalloon_x^2 + RelBalloon_y^2} \tag{5.13}$$

and just setting this equal to $Balloon_x$ and setting $Balloon_y$ to zero (equivalent to setting $\theta = 0$ in a rotation).

We don't have the position of our individual channels yet, only of the station itself, these can however easily be obtained using the get_relative_position module on our detector object, as we chose our station to be the center of the coordinate system, these relative positions are absolute positions of the channels in our frame of reference. Using these it thus doesn't matter where the position of the station was defined.

5.3.2 Signal analysis and initial guesses

Now that we have our geometry, let's analyse the data for the channels, the data for a particular channel is stored in a channel object. From this object we can get the recorded voltages with the get_trace() module, the recorded times with the get_times() module, the recorded frequencies with the get_frequencies() and the recorded spectrum corresponding to these frequencies with get_frequency_spectrum() , note that the last two do a FFT on our data. We know that the signal sent out by the weather balloon is a sine wave with a frequency of 403MHz; as the data is measured in nanoseconds the frequency is:

$$f = 403\text{MHz} = 403 \times 10^{-3}\frac{1}{\text{ns}} \tag{5.14}$$

The recorded spectrum of channels 0 and 3 is given in figure 5.10, for which we previously mentioned the spike at 0.403GHz. Now note that there are peaks both below 0.15GHz and above 0.6GHz, this is all non-physical noise as the Vpol antenna's range only goes from 0.15GHz to 0.6GHz [4] that's why we'll first pass this signal trough a virtual butterworth bandpass filter.

After passing the signal through the filter we'd also like to upsample the signal as to increase the resolution, this can be done using the resample() module and we'll upsample towards 10GHz increasing our timely accuracy from 0.3125ns to 0.1ns. After filtering and upsampling we have
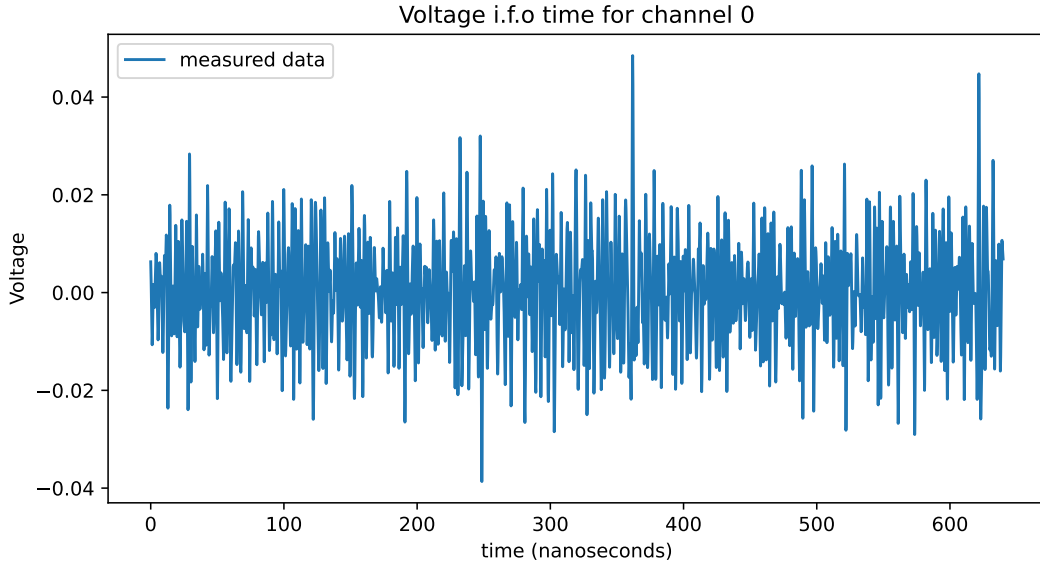
Figure 5.11: Recorded voltage i.f.o time in vpol antenna 0 after upsampling and passing through the butterworth filter

some voltage response as shown for channel 0 in figure 5.11, we wish to find a sine wave coming from the weather balloon in this signal.

To find a sine wave herein we'll radiotools' helper module get_normalized_xcorr which indirectly uses scipys signal correlation function but before using that we'll need a template to correlate with the data. The template we'll be choosing is of course the sine function but it's important to notice that the channel has a certain sampling rate, namely now after upsampling, 10GHz.

Our template sine which we'll correlate to the data will also need to have this sampling rate, meaning that it needs to be stepwise defined with spaces of 0.1ns. Next we'll also need to give the sine a certain amplitude, as we don't have a system for finding this yet let's for now assume an amplitude $A = 0.007$ (this shouldn't impact the correlation). Our template thus looks like this:

$$\mathcal{S} = A \cdot \sin(\omega t) \qquad (5.15)$$

with

$$\omega = 2\pi f \qquad (5.16)$$

Herein t is an array going from 0 to $3/f$ as to be able to fit 3 periods with intervals of 0.1ns. This template gets automatically shifted by the radiotools correlation module with steps of 0.1ns and each time the correlation with the data gets computed as is illustrated in figure 5.12. In real life the data is a bit less perfect and after doing this correlation procedure on the real data we get what's (partially) shown in figure 5.13. Herein the peaks represent the maximal correlation, if we now do the same for channel 3 we have two correlation functions, if we correlate these with eachother we'll get the difference in timing between the channels. This is easy to comprehend after taking a look at the illustration shown in figure 5.17. This cross correlation is shown in figure 5.16, the negative offset is caused by accounting for cable delay. Note that multiple peak correlations are present, to thus find out which one is the right one we'll do a simple simulation using our modified hybrid ray tracer from which we know what the approximate time difference is and search in this neighbourhood.
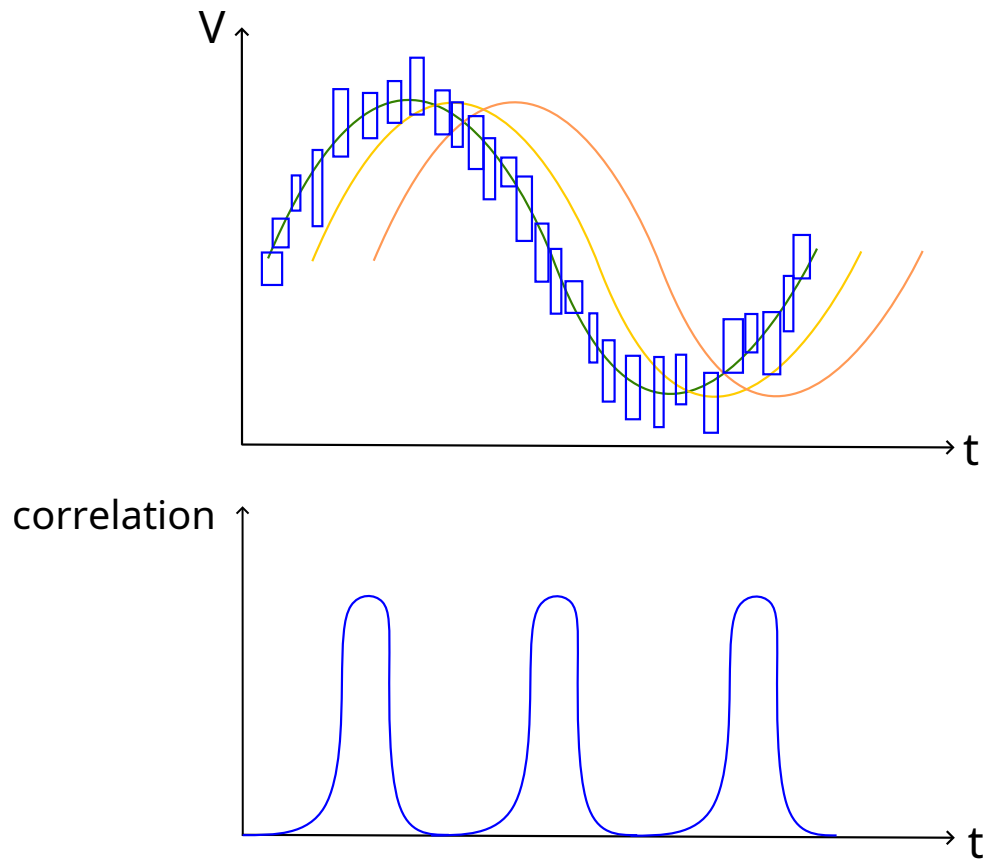
Figure 5.12: How correlating a sine with data leads to periodic peaks in the correlation function
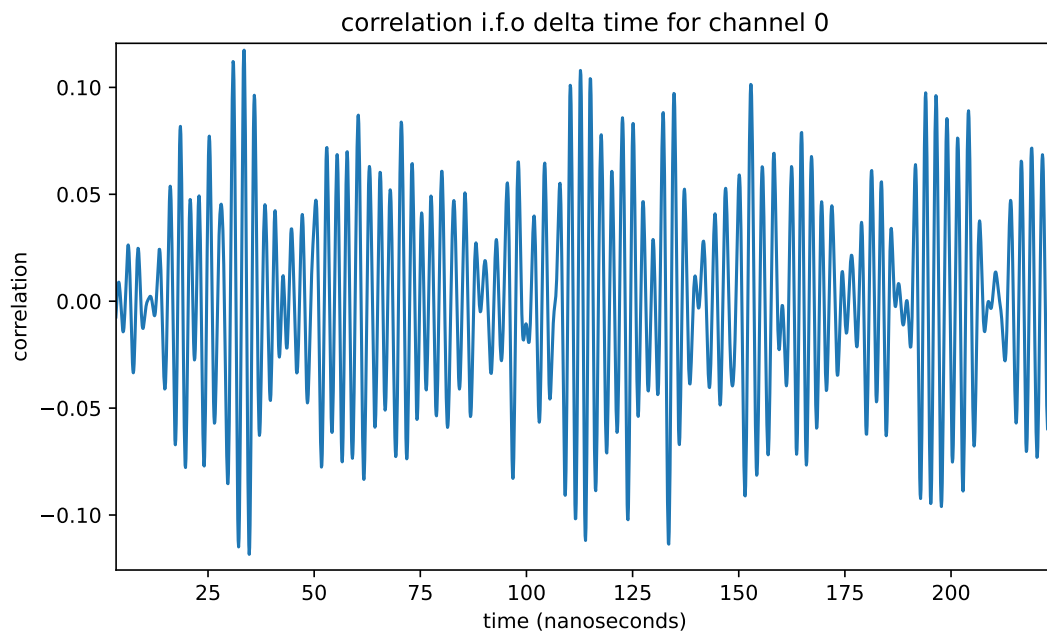
Figure 5.13: Zoom on the correlation i.f.o the time for channel 0
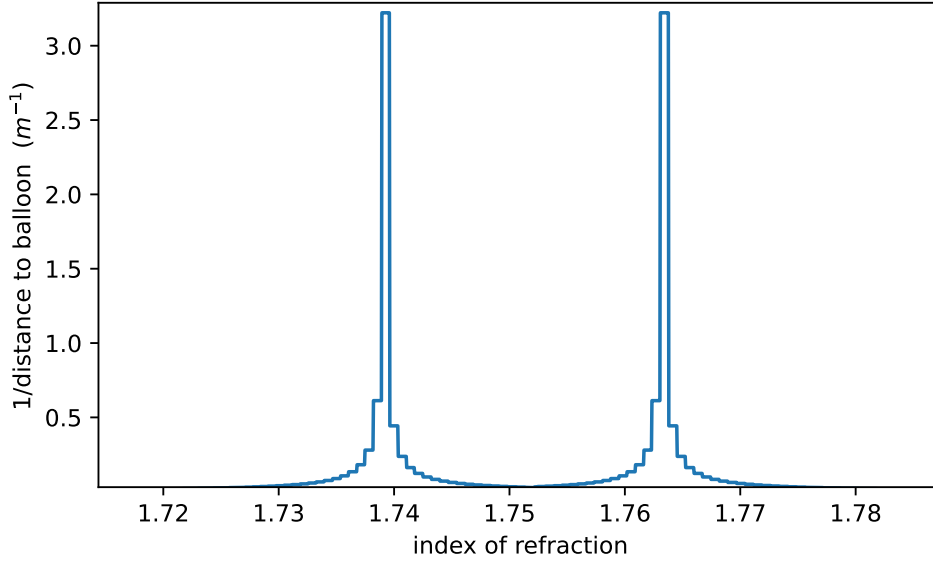
Figure 5.14: 2 possible values for n are found whilst fitting

5.3.3   Fitting n and finding it's error

Now that we have the differences in timing between the individual channels we'll loop over the local (middle of the detectors) indices of refraction, each time doing a plane wave reconstruction (also using Snell's law at the surface) and recording the distance from the reconstructed wave to the balloon at the height of the balloon. Doing this we have a list **n** and a list **differences**, if we'd select the minimal difference in the **differences** list however we'd find an incorrect n. We'll first need to rescale the indices using the correction factor ϵ as:

$$n_{corr} = \frac{n}{1 + \frac{\epsilon}{100}} \tag{5.17}$$

If we now plot these corrected indices of refraction (which we'll from now on just call "the fitted indices of refraction") on the x axis and on the y axis the inverse of the the differences we get figure 5.14. As you can see there are 2 equally probable answers, this could be caused by the plane wave landing left and equally distant right from the balloon. From looking at the exponential model and the density data however we don't expect an index of refraction as high as the second peak. We'll thus only concern ourselves with the first peak located between 1.73 and 1.75, now we want to know what is the exact peak value and it's error?

To find the peak value and it's error we'll first cut the data from 1.73 to 1.75, the value for n can be found as the middle of the peak. To find the error on this value a first idea was to gaussian to this curve of the form [20]

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} exp\left\{ -\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2} \right\} \tag{5.18}$$

the fit is shown on figure B.1, but as this fit obviously isn't that great and would underestimate the error we'll need another way to find the error. To estimate the error we think that using integration might give a good estimate, i.e we integrate over the full data and then again symmetrically outwards from the middle of the peak, if the ratio of these two is 68% then the interval this corersponds to is the one sigma error on our fit, analogously for 95%. If we carry this out we get what's shown on figure 5.15, our final fit is thus:
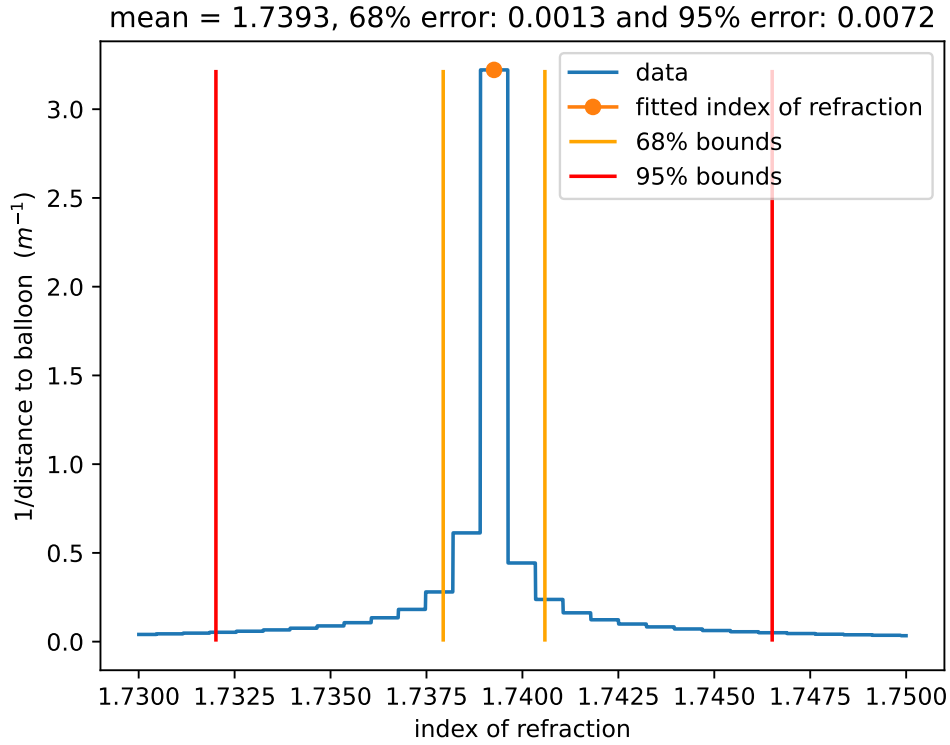
Figure 5.15: Fit to the phased array data showing various confidence intervals

Depth (m)	Station id	channels	Run:Event	$n_{exponential}$	ϵ	n_{fitted}
-93.237	23	phased array	800:1867	1.7383	4.6166%	1.7393 ± 0.0013

Here we clearly see the index of refraction from the exponential model fall within our fit.

5.4   Channels 5,6 and 7

A very interesting depth is inbetween channels 5 and 7 as, looking at the density profile shown in figure 3.2 this is where the actual density deviates the most from the theoretical models. The event we'll be using is recorded in detector 21 at the 26th of july 11:18:41 and falls within the $< 5°$ mark, doing analogous analyses as in the previous section we get figures B.2-B.4 which can be summerized in a table as shown below:

Depth (m)	Station id	channels	Run:Event	$n_{exponential}$	ϵ	n_{fit}
-48.155	21	6&7	1441:117	1.6400	-0.02%	1.6310 ± 0.001
-58.38	21	5&7	1441:117	1.6736	-0.23%	1.6688 ± 0.0006
-68.2	21	5&6	1441:117	1.6983	0.02%	1.6916 ± 0.0003
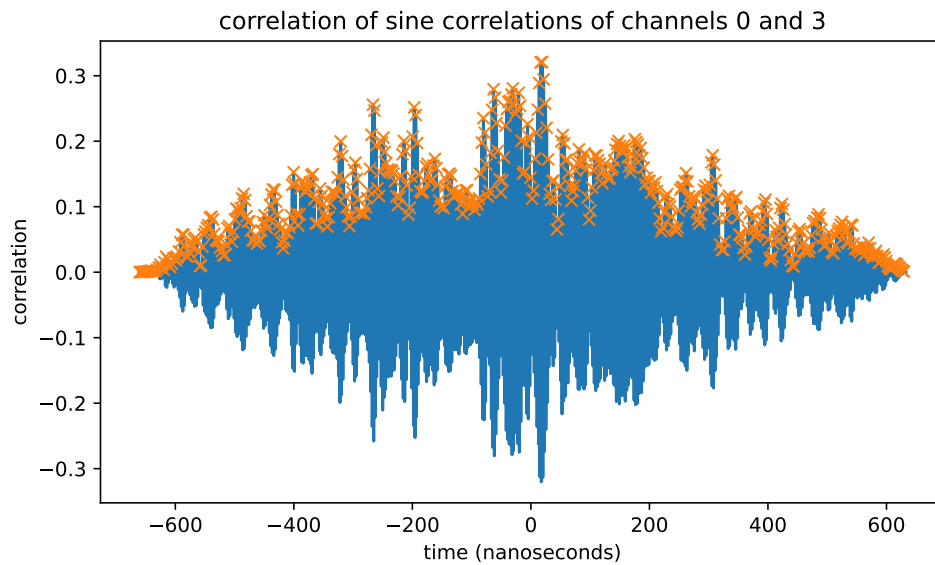
Figure 5.16: correlation of the previously, with sine, correlated dataset of channels 0 and 3, the orange marks indicate the peaks
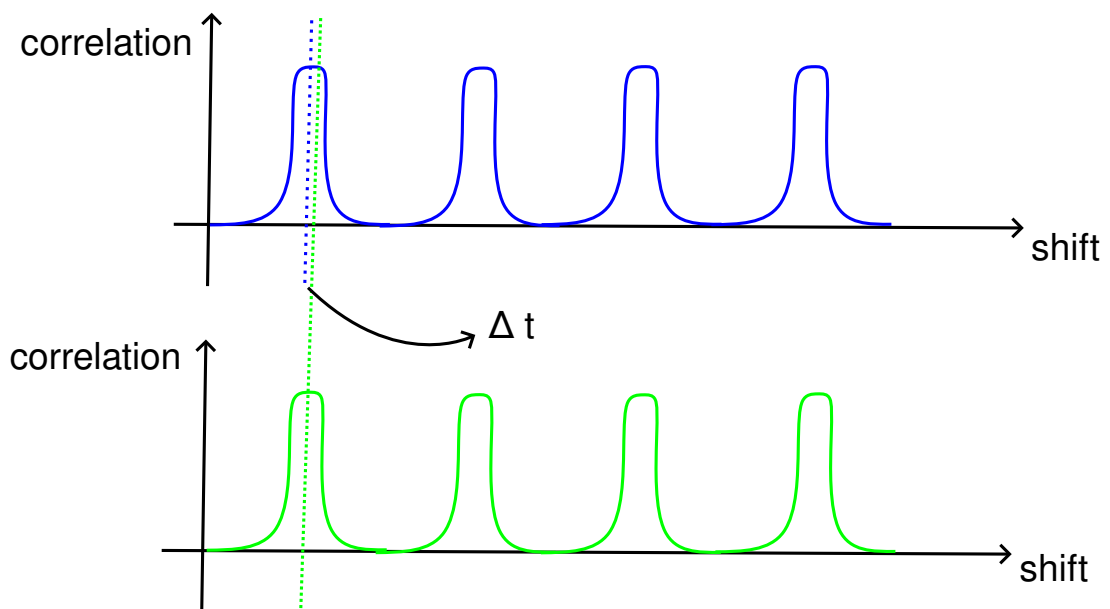
Figure 5.17: Illustration of how the different correlations with sine (above in blue being e.g channel 0 and below in green channel 1) have a difference in timing between the peaks which can be found from correlating the two
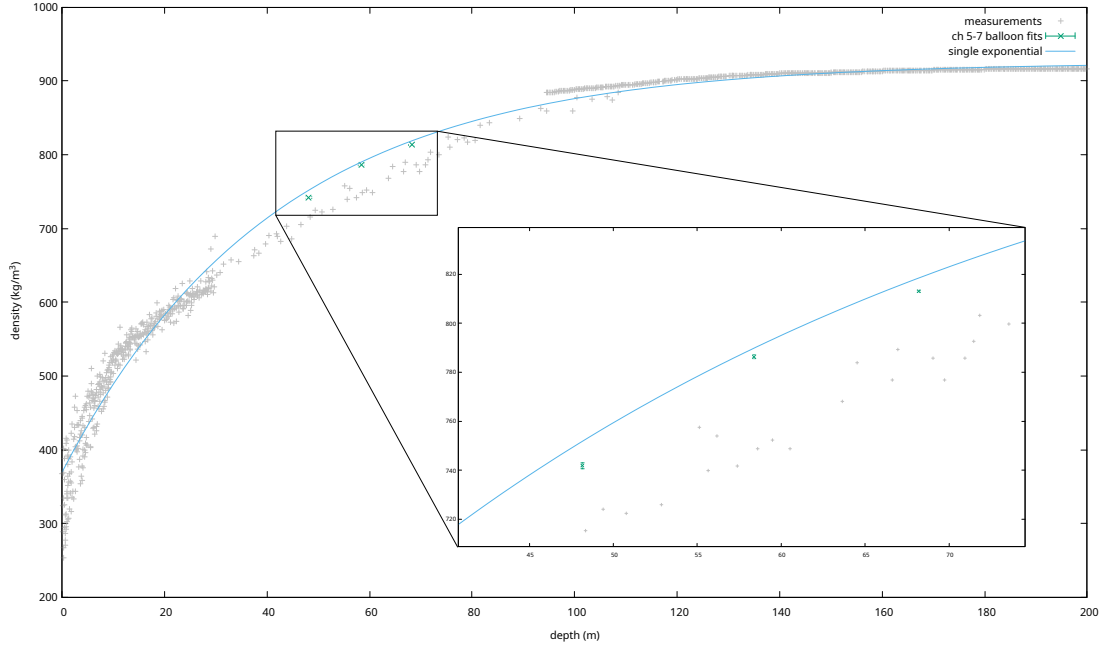
Figure 5.18: graph showing where the balloon data density is with respect to the other measurements and the exponential model

Note that all of the indices whom are found have a value below the exponential model's. Let's use Schytt's equation 3.13 to see where on the density vs depth curve these values lie:

$$n(z) \approx 1 + 0.78\rho/\rho_0 \tag{5.19}$$

$$\rho_0(n(z) - 1) \approx 0.78\rho \tag{5.20}$$

$$\frac{\rho_0}{0.78}(n(z) - 1) \approx \rho \tag{5.21}$$

$$\rho \approx 1175.641(n(z) - 1)\frac{\text{kg}}{\text{m}^3} \tag{5.22}$$

$$\tag{5.23}$$

The error propagation is also quite straightforward:

$$\delta\rho = 1175.641\delta n \tag{5.24}$$

If we use these equations the corrected indices of refraction correspond to the densities:

Depth (m)	ρ_{corr}
-48.16	741.830 ± 1.176
-58.38	786.269 ± 0.705
-68.2	813.073 ± 0.3527

If we plot this we get what's shown in figure 5.18. Note that the calculated values neither line up with the density measurements nor with the exponential model, but somewhere inbetween. This might be caused by one of the following reasons:

- The channels 5-7 are too far apart to use the plane wave reconstruction.

- Schytts law isn't fully correct

- The density measurements aren't correct

The density measurements do actually show quite a lot of spread for nearly the same values so there might be a problem there (there are also not that many recorded measurements).

As for the Schytt's equation, wheter it is a valid approximation or not will be descided when enough reliable data is available.

CONCLUSION

The hybrid ray tracer developed over the course of this thesis and reported in chapter 4 was a succes, not only is it more accurate than it's predecessors but it's also faster. The accuracy of the ray tracer also makes it perfectly suitable for very fine precision simulations as used within the Balloon radio wave measurements.

The use of weather balloons to estimate the index of refraction as was reported in 5 was also done succesfully even though it yielded quite unexpected results, not only were the index of refraction measured via this method deviating from the model but also deviating from the theoretically expected indices using the measured density in combination with Schytts equation

$$n(z) = 1 + 0.78\rho/\rho_0 \tag{5.25}$$

moreso lying somewhere inbetween. A conversion using the inverse of Schytts equation to arrive at the density makes this especially clear as shown on figure 5.18.

As there are a multitude of events (as can be found in appendices C and D but only finite time to analyse each and every one, this code is made public here as to make it easy to improve on this work. Especially interesting might be events where ϵ is negligable (the balloon is really close) at the phased array and as such is the plane wave reconstruction a near perfect representation of the actual wave.

Proposal of improved measurements

As this method seems to be a viable way of measuring the index of refraction, it might be a good idea to have a more controllable radio wave source fly closer to the detectors to make more accurate measurements, e.g an autonomous[8] drone with an antenna strapped to it. Assuming Schytt's equation to hold completely, ideally we'd like data inbetween the depths of 20-100m as this is where the biggest discrepencies might be observed between the single exponential model and the measured density data as is depicted on figure 3.2

---

[8]as to not have it need a radio controller, causing RF interference and also as autonomous GNSS positioning are always more accurate than humans

APPENDIX

$$A$$

LIST OF ABBREVIATIONS

- **AM**: **A**mplitude **M**odulated
- **AGN**: **A**ctive **G**alactic **N**ucleus
- **CMB**: **C**osmic **M**icrowave **B**ackground
- **DAQ**: **D**ata **AQ**uisition system
- **DnR**: **D**irect a**N**d **R**efracted
- **DSNB**: **D**iffuse **S**upernova **N**eutrino **B**ackground
- **FFT** **F**ast **F**ourrier **T**ransform
- **GRBs**: **G**amma-**R**ay **B**ursts
- **RADIANT**: **RA**dio **DI**gitizer and **A**uxiliary **N**eutrino **T**rigger
- **RNO-G**: **R**adio **N**eutrino **O**bservatory in **G**reenland
- **RF**: **R**adio **F**requency
- **UHE**: **U**ltra **H**igh **E**nergy
- **UHECRs**: **U**ltra **H**igh **E**nergy **C**osmic **R**ay**s**

B

EXTRA FIGURES
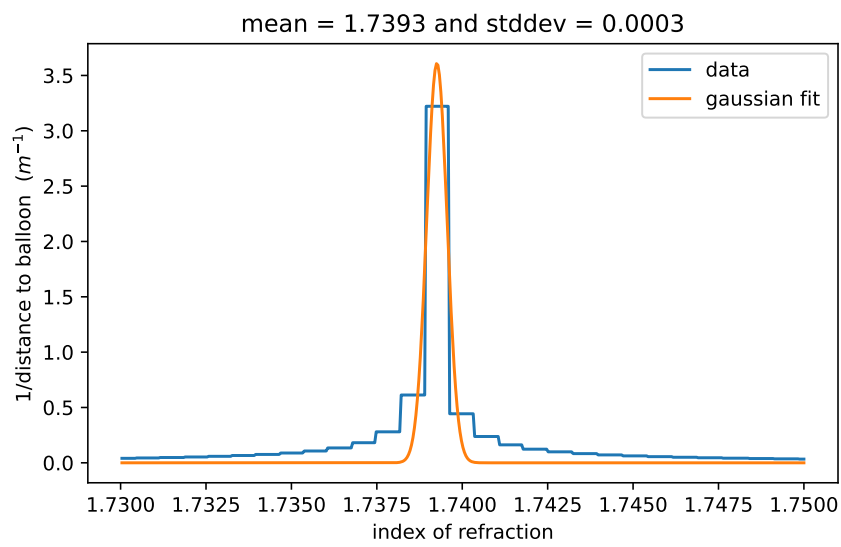
Figure B.1: gaussian fit doesn't perfectly align with the data
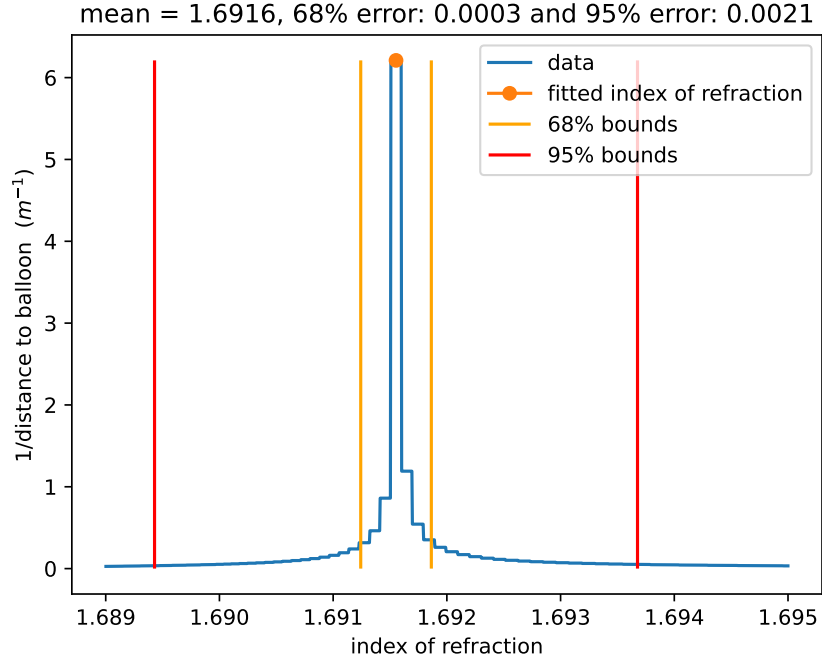
Figure B.2: Fit of channels 5 and 6
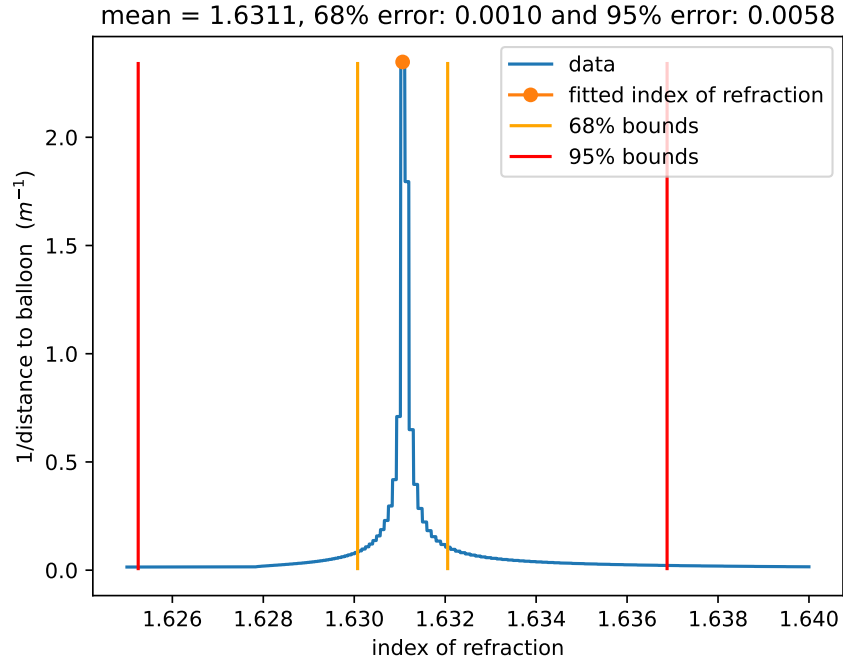
Figure B.3: Fit of channels 6 and 7
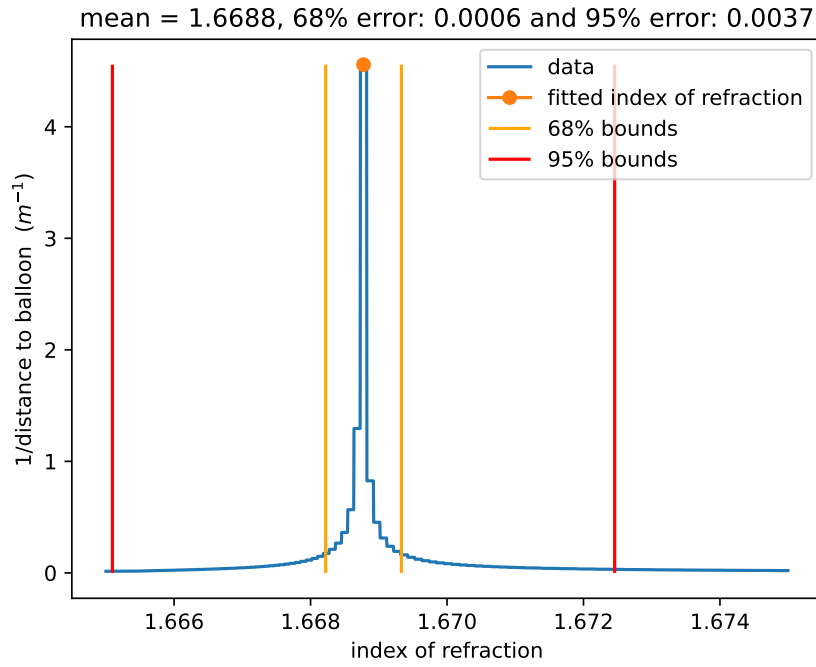
Figure B.4: Fit of channels 5 and 7

C

BALLOON PASSBYS UNDER 5°
IN THE SUMMER OF 2022

Balloon filename (gpx)	station	expected timeframe
SMT_20220907_112500.gpx	23.00	2022-09-07 11:28:47+00:00 till 2022-09-07 11:28:49+00:00
SMT_20220728_112143.gpx	12.00	2022-07-28 11:36:42+00:00 till 2022-07-28 11:37:44+00:00
SMT_20220728_112143.gpx	11.00	2022-07-28 11:34:00+00:00 till 2022-07-28 11:35:37+00:00
SMT_20220728_112143.gpx	13.00	2022-07-28 11:38:18+00:00 till 2022-07-28 11:39:13+00:00
SMT_20220723_231622.gpx	21.00	2022-07-23 23:25:19+00:00 till 2022-07-23 23:27:24+00:00
SMT_20220723_231622.gpx	22.00	2022-07-23 23:31:22+00:00 till 2022-07-23 23:33:50+00:00
SMT_20220930_231507.gpx	13.00	2022-09-30 23:19:22+00:00 till 2022-09-30 23:19:23+00:00

Balloon filename (gpx)	station	expected timeframe
SMT_20220904_111459.gpx	21.00	2022-09-04 11:17:00+00:00 till 2022-09-04 11:17:49+00:00
SMT_20220904_111459.gpx	22.00	2022-09-04 11:21:12+00:00 till 2022-09-04 11:21:29+00:00
SMT_20220726_111605.gpx	21.00	2022-07-26 11:18:31+00:00 till 2022-07-26 11:18:47+00:00
SMT_20220726_111605.gpx	23.00	2022-07-26 11:27:32+00:00 till 2022-07-26 11:28:24+00:00
SMT_20220726_111605.gpx	24.00	2022-07-26 11:34:42+00:00 till 2022-07-26 11:40:36+00:00
SMT_20220826_113003.gpx	21.00	2022-08-26 11:32:14+00:00 till 2022-08-26 11:32:30+00:00
SMT_20220826_113003.gpx	13.00	2022-08-26 11:38:18+00:00 till 2022-08-26 11:39:04+00:00
SMT_20220714_231645.gpx	11.00	2022-07-14 23:30:57+00:00 till 2022-07-14 23:31:40+00:00
SMT_20220701_111515.gpx	11.00	2022-07-01 11:22:57+00:00 till 2022-07-01 11:25:30+00:00
SMT_20220725_112511.gpx	12.00	2022-07-25 11:29:41+00:00 till 2022-07-25 11:30:12+00:00
SMT_20220703_111803.gpx	24.00	2022-07-03 11:47:29+00:00 till 2022-07-03 11:50:43+00:00
SMT_20220820_111609.gpx	12.00	2022-08-20 11:24:37+00:00 till 2022-08-20 11:25:33+00:00
SMT_20220623_231633.gpx	13.00	2022-06-23 23:29:52+00:00 till 2022-06-23 23:30:59+00:00
SMT_20220623_231633.gpx	22.00	2022-06-23 23:23:50+00:00 till 2022-06-23 23:25:39+00:00
SMT_20220703_231627.gpx	21.00	2022-07-04 00:21:36+00:00 till 2022-07-04 00:42:14+00:00
SMT_20220703_231627.gpx	12.00	2022-07-04 00:32:03+00:00 till 2022-07-04 00:45:17+00:00
SMT_20220703_231627.gpx	11.00	2022-07-04 00:32:02+00:00 till 2022-07-04 00:47:35+00:00
SMT_20220703_231627.gpx	22.00	2022-07-04 00:21:47+00:00 till 2022-07-04 00:40:30+00:00
SMT_20220703_231627.gpx	23.00	2022-07-04 00:26:31+00:00 till 2022-07-04 00:34:55+00:00
SMT_20220624_231528.gpx	24.00	2022-06-25 00:29:26+00:00 till 2022-06-25 00:35:21+00:00
SMT_20220916_231807.gpx	22.00	2022-09-16 23:21:25+00:00 till 2022-09-16 23:21:49+00:00
SMT_20220908_111450.gpx	12.00	2022-09-08 11:20:23+00:00 till 2022-09-08 11:21:08+00:00
SMT_20220908_111450.gpx	13.00	2022-09-08 11:23:08+00:00 till 2022-09-08 11:23:54+00:00
SMT_20220817_111600.gpx	21.00	2022-08-17 11:31:37+00:00 till 2022-08-17 11:33:56+00:00
SMT_20220729_112533.gpx	11.00	2022-07-29 12:26:35+00:00 till 2022-07-29 12:31:35+00:00
SMT_20220907_231621.gpx	24.00	2022-09-07 23:24:23+00:00 till 2022-09-07 23:24:51+00:00
SMT_20220630_231835.gpx	21.00	2022-07-01 00:18:16+00:00 till 2022-07-01 00:32:45+00:00
SMT_20220630_231835.gpx	12.00	2022-07-01 00:25:19+00:00 till 2022-07-01 00:38:40+00:00
SMT_20220630_231835.gpx	11.00	2022-07-01 00:24:24+00:00 till 2022-07-01 00:37:58+00:00
SMT_20220630_231835.gpx	13.00	2022-07-01 00:30:30+00:00 till 2022-07-01 00:37:06+00:00
SMT_20220630_231835.gpx	22.00	2022-07-01 00:22:24+00:00 till 2022-07-01 00:34:12+00:00
SMT_20220826_231852.gpx	23.00	2022-08-26 23:27:09+00:00 till 2022-08-26 23:27:51+00:00
SMT_20220826_231852.gpx	24.00	2022-08-26 23:29:02+00:00 till 2022-08-26 23:29:48+00:00
SMT_20220624_111844.gpx	13.00	2022-06-24 11:33:51+00:00 till 2022-06-24 11:34:42+00:00
SMT_20220624_111844.gpx	22.00	2022-06-24 11:31:46+00:00 till 2022-06-24 11:32:35+00:00
SMT_20220707_112434.gpx	21.00	2022-07-07 12:31:12+00:00 till 2022-07-07 12:48:00+00:00
SMT_20220707_112434.gpx	12.00	2022-07-07 12:37:07+00:00 till 2022-07-07 12:49:22+00:00
SMT_20220707_112434.gpx	11.00	2022-07-07 12:36:00+00:00 till 2022-07-07 12:49:22+00:00
SMT_20220707_112434.gpx	13.00	2022-07-07 12:41:38+00:00 till 2022-07-07 12:49:22+00:00
SMT_20220707_112434.gpx	22.00	2022-07-07 12:34:46+00:00 till 2022-07-07 12:49:22+00:00
SMT_20220707_112434.gpx	23.00	2022-07-07 12:41:33+00:00 till 2022-07-07 12:48:09+00:00
SMT_20220829_111459.gpx	23.00	2022-08-29 11:18:46+00:00 till 2022-08-29 11:18:58+00:00
SMT_20220829_111459.gpx	24.00	2022-08-29 11:19:52+00:00 till 2022-08-29 11:20:10+00:00
SMT_20220727_232120.gpx	21.00	2022-07-27 23:24:53+00:00 till 2022-07-27 23:25:26+00:00
SMT_20220719_231514.gpx	11.00	2022-07-19 23:23:48+00:00 till 2022-07-19 23:26:06+00:00
SMT_20220807_231525.gpx	11.00	2022-08-07 23:18:35+00:00 till 2022-08-07 23:19:02+00:00
SMT_20220725_231505.gpx	13.00	2022-07-26 00:41:45+00:00 till 2022-07-26 00:53:49+00:00
SMT_20220725_231505.gpx	22.00	2022-07-25 23:22:13+00:00 till 2022-07-25 23:23:48+00:00
SMT_20220725_231505.gpx	23.00	2022-07-25 23:26:39+00:00 till 2022-07-26 00:48:25+00:00

Balloon filename (gpx)	station	expected timeframe
SMT_20220725_231505.gpx	23.00	2022-07-25 23:26:39+00:00 till 2022-07-26 00:48:25+00:00
SMT_20220725_231505.gpx	24.00	2022-07-26 00:35:23+00:00 till 2022-07-26 00:49:53+00:00
SMT_20220928_111604.gpx	11.00	2022-09-28 11:34:13+00:00 till 2022-09-28 11:35:36+00:00
SMT_20220808_231507.gpx	11.00	2022-08-08 23:18:40+00:00 till 2022-08-08 23:19:08+00:00
SMT_20220816_232314.gpx	22.00	2022-08-16 23:34:34+00:00 till 2022-08-16 23:35:09+00:00
SMT_20220704_231605.gpx	21.00	2022-07-04 23:23:47+00:00 till 2022-07-04 23:27:33+00:00
SMT_20220630_112143.gpx	21.00	2022-06-30 12:26:56+00:00 till 2022-06-30 12:38:03+00:00
SMT_20220630_112143.gpx	12.00	2022-06-30 12:31:57+00:00 till 2022-06-30 12:44:25+00:00
SMT_20220630_112143.gpx	11.00	2022-06-30 12:31:26+00:00 till 2022-06-30 12:42:23+00:00
SMT_20220630_112143.gpx	13.00	2022-06-30 12:34:40+00:00 till 2022-06-30 12:44:06+00:00
SMT_20220630_112143.gpx	22.00	2022-06-30 12:28:56+00:00 till 2022-06-30 12:39:23+00:00
SMT_20220630_112143.gpx	23.00	2022-06-30 12:33:15+00:00 till 2022-06-30 12:37:09+00:00
SMT_20220704_112103.gpx	21.00	2022-07-04 11:49:20+00:00 till 2022-07-04 11:55:43+00:00
SMT_20220704_112103.gpx	11.00	2022-07-04 11:45:32+00:00 till 2022-07-04 11:51:31+00:00
SMT_20220704_112103.gpx	13.00	2022-07-04 12:10:16+00:00 till 2022-07-04 12:11:52+00:00
SMT_20220704_112103.gpx	22.00	2022-07-04 11:52:45+00:00 till 2022-07-04 12:02:35+00:00
SMT_20220704_112103.gpx	23.00	2022-07-04 11:58:37+00:00 till 2022-07-04 12:16:54+00:00
SMT_20220704_112103.gpx	24.00	2022-07-04 12:06:21+00:00 till 2022-07-04 12:25:28+00:00
SMT_20220720_111442.gpx	23.00	2022-07-20 11:46:26+00:00 till 2022-07-20 11:50:05+00:00
SMT_20220720_111442.gpx	24.00	2022-07-20 11:50:53+00:00 till 2022-07-20 12:03:52+00:00
SMT_20220829_231438.gpx	22.00	2022-08-29 23:16:33+00:00 till 2022-08-29 23:16:38+00:00
SMT_20220829_231438.gpx	23.00	2022-08-29 23:17:45+00:00 till 2022-08-29 23:17:58+00:00
SMT_20220921_231851.gpx	12.00	2022-09-22 10:46:18+00:00 till 2022-09-22 10:46:19+00:00
SMT_20220821_111511.gpx	21.00	2022-08-21 12:39:53+00:00 till 2022-08-21 12:56:06+00:00
SMT_20220821_111511.gpx	12.00	2022-08-21 12:51:06+00:00 till 2022-08-21 12:56:06+00:00
SMT_20220821_111511.gpx	11.00	2022-08-21 12:51:32+00:00 till 2022-08-21 12:56:06+00:00
SMT_20220821_111511.gpx	13.00	2022-08-21 12:52:27+00:00 till 2022-08-21 12:56:06+00:00
SMT_20220821_111511.gpx	22.00	2022-08-21 12:40:26+00:00 till 2022-08-21 12:56:06+00:00
SMT_20220821_111511.gpx	23.00	2022-08-21 12:43:03+00:00 till 2022-08-21 12:56:06+00:00
SMT_20220724_231539.gpx	11.00	2022-07-24 23:20:04+00:00 till 2022-07-24 23:21:11+00:00
SMT_20220827_111534.gpx	24.00	2022-08-27 11:23:00+00:00 till 2022-08-27 11:23:07+00:00
SMT_20220625_111630.gpx	21.00	2022-06-25 12:27:31+00:00 till 2022-06-25 12:35:22+00:00
SMT_20220625_111630.gpx	12.00	2022-06-25 12:30:09+00:00 till 2022-06-25 12:39:40+00:00
SMT_20220625_111630.gpx	11.00	2022-06-25 12:32:45+00:00 till 2022-06-25 12:37:21+00:00
SMT_20220625_111630.gpx	13.00	2022-06-25 12:30:42+00:00 till 2022-06-25 12:40:11+00:00
SMT_20220625_111630.gpx	22.00	2022-06-25 12:27:03+00:00 till 2022-06-25 12:37:11+00:00
SMT_20220625_111630.gpx	23.00	2022-06-25 12:28:31+00:00 till 2022-06-25 12:37:18+00:00

D

BALLOON PASSBYS UNDER 10°
IN THE SUMMER OF 2022

Balloon filename (gpx)	station	expected timeframe
SMT_20220719_231514.gpx	21	2022-07-19 23:19:46+00:00 till 2022-07-19 23:20:41+00:00
SMT_20220719_231514.gpx	11	2022-07-19 23:22:44+00:00 till 2022-07-19 23:27:19+00:00
SMT_20220831_111419.gpx	11	2022-08-31 11:16:20+00:00 till 2022-08-31 11:16:26+00:00
SMT_20220625_111630.gpx	21	2022-06-25 12:22:22+00:00 till 2022-06-25 12:40:57+00:00
SMT_20220625_111630.gpx	12	2022-06-25 12:25:55+00:00 till 2022-06-25 12:45:40+00:00
SMT_20220625_111630.gpx	11	2022-06-25 12:25:58+00:00 till 2022-06-25 12:45:09+00:00
SMT_20220625_111630.gpx	13	2022-06-25 12:26:44+00:00 till 2022-06-25 12:45:40+00:00
SMT_20220625_111630.gpx	22	2022-06-25 12:22:25+00:00 till 2022-06-25 12:42:40+00:00
SMT_20220625_111630.gpx	23	2022-06-25 12:24:10+00:00 till 2022-06-25 12:43:09+00:00
SMT_20220625_111630.gpx	24	2022-06-25 12:26:32+00:00 till 2022-06-25 12:42:24+00:00
SMT_20220826_113003.gpx	21	2022-08-26 11:32:04+00:00 till 2022-08-26 11:32:42+00:00
SMT_20220826_113003.gpx	13	2022-08-26 11:37:48+00:00 till 2022-08-26 11:39:40+00:00
SMT_20220626_112912.gpx	11	2022-06-26 11:35:52+00:00 till 2022-06-26 11:37:32+00:00
SMT_20220808_231507.gpx	11	2022-08-08 23:18:27+00:00 till 2022-08-08 23:19:24+00:00

Balloon filename (gpx)	station	expected timeframe
SMT_20220724_231539.gpx	11	2022-07-24 23:19:27+00:00 till 2022-07-24 23:21:58+00:00
SMT_20220724_231539.gpx	13	2022-07-25 00:19:35+00:00 till 2022-07-25 00:32:09+00:00
SMT_20220829_231438.gpx	22	2022-08-29 23:16:29+00:00 till 2022-08-29 23:16:42+00:00
SMT_20220829_231438.gpx	23	2022-08-29 23:17:39+00:00 till 2022-08-29 23:18:04+00:00
SMT_20220829_231438.gpx	24	2022-08-29 23:18:59+00:00 till 2022-08-29 23:19:22+00:00
SMT_20220916_231807.gpx	21	2022-09-16 23:19:09+00:00 till 2022-09-16 23:19:17+00:00
SMT_20220916_231807.gpx	22	2022-09-16 23:21:13+00:00 till 2022-09-16 23:22:03+00:00
SMT_20220816_232314.gpx	22	2022-08-16 23:33:11+00:00 till 2022-08-16 23:37:34+00:00
SMT_20220907_112500.gpx	23	2022-09-07 11:28:37+00:00 till 2022-09-07 11:28:59+00:00
SMT_20220907_112500.gpx	24	2022-09-07 11:29:35+00:00 till 2022-09-07 11:29:53+00:00
SMT_20220628_231514.gpx	12	2022-06-28 23:19:37+00:00 till 2022-06-28 23:20:55+00:00
SMT_20220630_112143.gpx	21	2022-06-30 12:21:02+00:00 till 2022-06-30 12:46:53+00:00
SMT_20220630_112143.gpx	12	2022-06-30 12:26:35+00:00 till 2022-06-30 12:48:30+00:00
SMT_20220630_112143.gpx	11	2022-06-30 12:25:33+00:00 till 2022-06-30 12:48:30+00:00
SMT_20220630_112143.gpx	13	2022-06-30 12:28:56+00:00 till 2022-06-30 12:48:30+00:00
SMT_20220630_112143.gpx	22	2022-06-30 12:23:21+00:00 till 2022-06-30 12:47:42+00:00
SMT_20220630_112143.gpx	23	2022-06-30 12:26:20+00:00 till 2022-06-30 12:47:25+00:00
SMT_20220630_112143.gpx	24	2022-06-30 12:30:34+00:00 till 2022-06-30 12:46:14+00:00
SMT_20220725_231505.gpx	21	2022-07-25 23:17:47+00:00 till 2022-07-26 00:53:39+00:00
SMT_20220725_231505.gpx	12	2022-07-26 00:38:19+00:00 till 2022-07-26 00:55:04+00:00
SMT_20220725_231505.gpx	11	2022-07-26 00:45:41+00:00 till 2022-07-26 00:55:04+00:00
SMT_20220725_231505.gpx	13	2022-07-26 00:34:56+00:00 till 2022-07-26 00:55:04+00:00
SMT_20220725_231505.gpx	22	2022-07-25 23:21:41+00:00 till 2022-07-26 00:55:04+00:00
SMT_20220725_231505.gpx	23	2022-07-25 23:25:16+00:00 till 2022-07-26 00:55:04+00:00
SMT_20220725_231505.gpx	24	2022-07-26 00:25:43+00:00 till 2022-07-26 00:55:04+00:00
SMT_20220703_111803.gpx	22	2022-07-03 11:40:24+00:00 till 2022-07-03 11:42:55+00:00
SMT_20220703_111803.gpx	23	2022-07-03 11:42:26+00:00 till 2022-07-03 11:49:33+00:00
SMT_20220703_111803.gpx	24	2022-07-03 11:45:29+00:00 till 2022-07-03 11:52:37+00:00
SMT_20220820_111609.gpx	21	2022-08-20 11:19:36+00:00 till 2022-08-20 11:21:16+00:00
SMT_20220820_111609.gpx	12	2022-08-20 11:24:02+00:00 till 2022-08-20 11:26:23+00:00
SMT_20220624_231528.gpx	12	2022-06-25 00:33:09+00:00 till 2022-06-25 00:40:51+00:00
SMT_20220624_231528.gpx	13	2022-06-25 00:28:45+00:00 till 2022-06-25 00:40:56+00:00
SMT_20220624_231528.gpx	22	2022-06-25 00:31:15+00:00 till 2022-06-25 00:34:40+00:00
SMT_20220624_231528.gpx	23	2022-06-25 00:24:52+00:00 till 2022-06-25 00:40:56+00:00
SMT_20220624_231528.gpx	24	2022-06-25 00:23:18+00:00 till 2022-06-25 00:40:56+00:00
SMT_20220701_111515.gpx	11	2022-07-01 11:21:58+00:00 till 2022-07-01 11:26:31+00:00
SMT_20220930_111518.gpx	12	2022-09-30 11:19:46+00:00 till 2022-09-30 11:20:04+00:00
SMT_20220821_111511.gpx	21	2022-08-21 12:30:15+00:00 till 2022-08-21 12:56:06+00:00
SMT_20220821_111511.gpx	12	2022-08-21 12:37:45+00:00 till 2022-08-21 12:56:06+00:00
SMT_20220821_111511.gpx	11	2022-08-21 12:37:28+00:00 till 2022-08-21 12:56:06+00:00
SMT_20220821_111511.gpx	13	2022-08-21 12:39:39+00:00 till 2022-08-21 12:56:06+00:00
SMT_20220821_111511.gpx	22	2022-08-21 12:31:58+00:00 till 2022-08-21 12:56:06+00:00
SMT_20220821_111511.gpx	23	2022-08-21 12:33:47+00:00 till 2022-08-21 12:56:06+00:00
SMT_20220821_111511.gpx	24	2022-08-21 12:39:51+00:00 till 2022-08-21 12:56:06+00:00
SMT_20220813_112235.gpx	21	2022-08-13 11:12:14+00:00 till 2022-08-13 11:12:14+00:00
SMT_20220714_231645.gpx	12	2022-07-14 23:33:22+00:00 till 2022-07-14 23:38:49+00:00
SMT_20220714_231645.gpx	11	2022-07-14 23:27:54+00:00 till 2022-07-14 23:35:34+00:00
SMT_20220714_231645.gpx	13	2022-07-14 23:37:48+00:00 till 2022-07-14 23:46:03+00:00
SMT_20220723_231622.gpx	21	2022-07-23 23:21:03+00:00 till 2022-07-23 23:30:26+00:00

Balloon filename (gpx)	station	expected timeframe
SMT_20220723_231622.gpx	21	2022-07-23 23:21:03+00:00 till 2022-07-23 23:30:26+00:00
SMT_20220723_231622.gpx	22	2022-07-23 23:30:12+00:00 till 2022-07-23 23:35:27+00:00
SMT_20220723_231622.gpx	23	2022-07-23 23:33:26+00:00 till 2022-07-23 23:37:57+00:00
SMT_20220723_231622.gpx	24	2022-07-23 23:36:33+00:00 till 2022-07-23 23:39:18+00:00
SMT_20220701_231934.gpx	22	2022-07-01 23:08:19+00:00 till 2022-07-01 23:08:19+00:00
SMT_20220701_231934.gpx	23	2022-07-01 23:08:19+00:00 till 2022-07-01 23:08:19+00:00
SMT_20220830_231921.gpx	12	2022-08-30 23:22:15+00:00 till 2022-08-30 23:22:31+00:00
SMT_20220706_231615.gpx	22	2022-07-07 00:34:24+00:00 till 2022-07-07 00:35:15+00:00
SMT_20220706_231615.gpx	23	2022-07-07 00:33:30+00:00 till 2022-07-07 00:35:15+00:00
SMT_20220706_231615.gpx	24	2022-07-07 00:33:50+00:00 till 2022-07-07 00:35:15+00:00
SMT_20220728_112143.gpx	12	2022-07-28 11:36:04+00:00 till 2022-07-28 11:38:16+00:00
SMT_20220728_112143.gpx	11	2022-07-28 11:32:06+00:00 till 2022-07-28 11:36:28+00:00
SMT_20220728_112143.gpx	13	2022-07-28 11:37:48+00:00 till 2022-07-28 11:39:49+00:00
SMT_20220716_111633.gpx	21	2022-07-16 12:05:05+00:00 till 2022-07-16 12:20:22+00:00
SMT_20220716_111633.gpx	12	2022-07-16 12:10:53+00:00 till 2022-07-16 12:24:07+00:00
SMT_20220716_111633.gpx	11	2022-07-16 12:08:23+00:00 till 2022-07-16 12:25:43+00:00
SMT_20220716_111633.gpx	22	2022-07-16 12:09:06+00:00 till 2022-07-16 12:18:35+00:00
SMT_20220715_231621.gpx	21	2022-07-16 00:08:58+00:00 till 2022-07-16 00:12:03+00:00
SMT_20220726_111605.gpx	21	2022-07-26 11:18:19+00:00 till 2022-07-26 11:19:03+00:00
SMT_20220726_111605.gpx	12	2022-07-26 12:20:14+00:00 till 2022-07-26 12:21:53+00:00
SMT_20220726_111605.gpx	13	2022-07-26 12:06:52+00:00 till 2022-07-26 12:30:56+00:00
SMT_20220726_111605.gpx	22	2022-07-26 11:21:46+00:00 till 2022-07-26 11:24:39+00:00
SMT_20220726_111605.gpx	23	2022-07-26 11:26:09+00:00 till 2022-07-26 12:22:24+00:00
SMT_20220726_111605.gpx	24	2022-07-26 11:31:17+00:00 till 2022-07-26 12:25:10+00:00
SMT_20220726_231510.gpx	11	2022-07-26 23:31:39+00:00 till 2022-07-26 23:43:04+00:00
SMT_20220724_111936.gpx	11	2022-07-24 11:24:45+00:00 till 2022-07-24 11:27:03+00:00
SMT_20220724_111936.gpx	13	2022-07-24 12:23:16+00:00 till 2022-07-24 12:33:48+00:00
SMT_20220724_111936.gpx	24	2022-07-24 12:12:44+00:00 till 2022-07-24 12:29:27+00:00
SMT_20220820_231751.gpx	13	2022-08-20 23:41:03+00:00 till 2022-08-20 23:45:01+00:00
SMT_20220820_231751.gpx	22	2022-08-20 23:39:20+00:00 till 2022-08-20 23:41:29+00:00
SMT_20220820_231751.gpx	23	2022-08-20 23:39:47+00:00 till 2022-08-20 23:42:41+00:00
SMT_20220703_231627.gpx	21	2022-07-03 23:58:58+00:00 till 2022-07-04 00:51:57+00:00
SMT_20220703_231627.gpx	12	2022-07-04 00:19:47+00:00 till 2022-07-04 00:52:09+00:00
SMT_20220703_231627.gpx	11	2022-07-04 00:19:37+00:00 till 2022-07-04 00:52:09+00:00
SMT_20220703_231627.gpx	13	2022-07-04 00:22:04+00:00 till 2022-07-04 00:52:07+00:00
SMT_20220703_231627.gpx	22	2022-07-04 00:14:09+00:00 till 2022-07-04 00:50:28+00:00
SMT_20220703_231627.gpx	23	2022-07-04 00:17:51+00:00 till 2022-07-04 00:48:29+00:00
SMT_20220703_231627.gpx	24	2022-07-04 00:23:40+00:00 till 2022-07-04 00:42:26+00:00
SMT_20220816_111548.gpx	11	2022-08-16 11:19:06+00:00 till 2022-08-16 11:20:08+00:00
SMT_20220704_231605.gpx	21	2022-07-04 23:21:14+00:00 till 2022-07-04 23:29:47+00:00
SMT_20220704_231605.gpx	22	2022-07-05 00:45:35+00:00 till 2022-07-05 00:46:21+00:00
SMT_20220704_231605.gpx	23	2022-07-05 00:44:35+00:00 till 2022-07-05 00:46:21+00:00
SMT_20220704_231605.gpx	24	2022-07-05 00:44:08+00:00 till 2022-07-05 00:46:21+00:00
SMT_20220921_231851.gpx	12	2022-09-22 10:46:18+00:00 till 2022-09-22 10:46:20+00:00
SMT_20220921_231851.gpx	13	2022-09-22 10:46:21+00:00 till 2022-09-22 10:46:22+00:00
SMT_20220630_231835.gpx	21	2022-07-01 00:11:25+00:00 till 2022-07-01 00:39:49+00:00
SMT_20220630_231835.gpx	12	2022-07-01 00:19:34+00:00 till 2022-07-01 00:45:51+00:00
SMT_20220630_231835.gpx	11	2022-07-01 00:17:35+00:00 till 2022-07-01 00:45:17+00:00

Balloon filename (gpx)	station	expected timeframe
SMT_20220630_231835.gpx	13	2022-07-01 00:23:23+00:00 till 2022-07-01 00:45:28+00:00
SMT_20220630_231835.gpx	22	2022-07-01 00:17:04+00:00 till 2022-07-01 00:40:11+00:00
SMT_20220630_231835.gpx	23	2022-07-01 00:21:11+00:00 till 2022-07-01 00:39:32+00:00
SMT_20220630_231835.gpx	24	2022-07-01 00:26:47+00:00 till 2022-07-01 00:37:45+00:00
SMT_20220807_231525.gpx	11	2022-08-07 23:18:24+00:00 till 2022-08-07 23:19:22+00:00
SMT_20220727_232120.gpx	21	2022-07-27 23:24:29+00:00 till 2022-07-27 23:25:46+00:00
SMT_20220727_232120.gpx	22	2022-07-27 23:27:22+00:00 till 2022-07-27 23:27:30+00:00
SMT_20220727_232120.gpx	23	2022-07-27 23:31:39+00:00 till 2022-07-27 23:32:39+00:00
SMT_20220727_232120.gpx	24	2022-07-27 23:33:29+00:00 till 2022-07-27 23:36:31+00:00
SMT_20220707_112434.gpx	21	2022-07-07 12:23:11+00:00 till 2022-07-07 12:49:22+00:00
SMT_20220707_112434.gpx	12	2022-07-07 12:31:06+00:00 till 2022-07-07 12:49:22+00:00
SMT_20220707_112434.gpx	11	2022-07-07 12:28:16+00:00 till 2022-07-07 12:49:22+00:00
SMT_20220707_112434.gpx	13	2022-07-07 12:34:19+00:00 till 2022-07-07 12:49:22+00:00
SMT_20220707_112434.gpx	22	2022-07-07 12:26:02+00:00 till 2022-07-07 12:49:22+00:00
SMT_20220707_112434.gpx	23	2022-07-07 12:32:01+00:00 till 2022-07-07 12:49:22+00:00
SMT_20220707_112434.gpx	24	2022-07-07 12:37:10+00:00 till 2022-07-07 12:49:22+00:00
SMT_20220617_231459.gpx	22	2022-06-17 23:17:36+00:00 till 2022-06-17 23:17:58+00:00
SMT_20220826_231852.gpx	23	2022-08-26 23:26:43+00:00 till 2022-08-26 23:28:19+00:00
SMT_20220826_231852.gpx	24	2022-08-26 23:28:41+00:00 till 2022-08-26 23:30:12+00:00
SMT_20220828_111632.gpx	22	2022-08-28 11:21:14+00:00 till 2022-08-28 11:23:09+00:00
SMT_20220828_111632.gpx	23	2022-08-28 11:24:07+00:00 till 2022-08-28 11:25:44+00:00
SMT_20220828_111632.gpx	24	2022-08-28 11:26:44+00:00 till 2022-08-28 11:28:09+00:00
SMT_20220623_231633.gpx	13	2022-06-23 23:29:11+00:00 till 2022-06-23 23:31:38+00:00
SMT_20220623_231633.gpx	22	2022-06-23 23:23:06+00:00 till 2022-06-23 23:26:51+00:00
SMT_20220807_111547.gpx	12	2022-08-07 11:22:02+00:00 till 2022-08-07 11:22:59+00:00
SMT_20220807_111547.gpx	13	2022-08-07 11:26:06+00:00 till 2022-08-07 11:27:55+00:00
SMT_20220624_111844.gpx	12	2022-06-24 11:33:01+00:00 till 2022-06-24 11:33:33+00:00
SMT_20220624_111844.gpx	13	2022-06-24 11:33:25+00:00 till 2022-06-24 11:35:11+00:00
SMT_20220624_111844.gpx	22	2022-06-24 11:31:13+00:00 till 2022-06-24 11:33:06+00:00
SMT_20220929_111540.gpx	12	2022-09-29 11:19:52+00:00 till 2022-09-29 11:20:48+00:00
SMT_20220720_111442.gpx	21	2022-07-20 11:28:20+00:00 till 2022-07-20 11:34:28+00:00
SMT_20220720_111442.gpx	22	2022-07-20 11:32:55+00:00 till 2022-07-20 11:49:29+00:00
SMT_20220720_111442.gpx	23	2022-07-20 11:37:02+00:00 till 2022-07-20 12:04:15+00:00
SMT_20220720_111442.gpx	24	2022-07-20 11:44:43+00:00 till 2022-07-20 12:12:56+00:00
SMT_20220829_111459.gpx	21	2022-08-29 11:15:53+00:00 till 2022-08-29 11:15:59+00:00
SMT_20220829_111459.gpx	23	2022-08-29 11:18:35+00:00 till 2022-08-29 11:19:05+00:00
SMT_20220829_111459.gpx	24	2022-08-29 11:19:44+00:00 till 2022-08-29 11:20:20+00:00
SMT_20220705_113009.gpx	21	2022-07-05 12:48:04+00:00 till 2022-07-05 12:53:24+00:00
SMT_20220705_113009.gpx	11	2022-07-05 12:50:13+00:00 till 2022-07-05 12:53:24+00:00
SMT_20220817_111600.gpx	21	2022-08-17 11:20:21+00:00 till 2022-08-17 11:35:09+00:00
SMT_20220817_111600.gpx	11	2022-08-17 11:27:50+00:00 till 2022-08-17 11:30:32+00:00
SMT_20220904_111459.gpx	21	2022-09-04 11:16:45+00:00 till 2022-09-04 11:18:39+00:00
SMT_20220904_111459.gpx	22	2022-09-04 11:20:46+00:00 till 2022-09-04 11:21:55+00:00
SMT_20220904_111459.gpx	23	2022-09-04 11:22:57+00:00 till 2022-09-04 11:24:53+00:00
SMT_20220904_111459.gpx	24	2022-09-04 11:24:58+00:00 till 2022-09-04 11:25:49+00:00
SMT_20220928_111604.gpx	12	2022-09-28 11:37:02+00:00 till 2022-09-28 11:38:16+00:00
SMT_20220928_111604.gpx	11	2022-09-28 11:33:16+00:00 till 2022-09-28 11:36:59+00:00
SMT_20220930_231507.gpx	13	2022-09-30 23:19:10+00:00 till 2022-09-30 23:19:35+00:00

Balloon filename (gpx)	station	expected timeframe
SMT_20220904_231730.gpx	24	2022-09-04 23:22:54+00:00 till 2022-09-04 23:23:14+00:00
SMT_20220921_111527.gpx	12	2022-09-21 11:27:31+00:00 till 2022-09-21 11:29:09+00:00
SMT_20220921_111527.gpx	11	2022-09-21 11:25:37+00:00 till 2022-09-21 11:27:09+00:00
SMT_20220921_111527.gpx	13	2022-09-21 11:29:18+00:00 till 2022-09-21 11:30:46+00:00
SMT_20220725_112511.gpx	12	2022-07-25 11:29:27+00:00 till 2022-07-25 11:30:29+00:00
SMT_20220907_231621.gpx	23	2022-09-07 23:22:59+00:00 till 2022-09-07 23:23:40+00:00
SMT_20220907_231621.gpx	24	2022-09-07 23:24:09+00:00 till 2022-09-07 23:25:07+00:00
SMT_20220827_111534.gpx	24	2022-08-27 11:22:30+00:00 till 2022-08-27 11:23:43+00:00
SMT_20220615_231713.gpx	12	2022-06-15 23:21:29+00:00 till 2022-06-15 23:22:47+00:00
SMT_20220908_111450.gpx	12	2022-09-08 11:20:02+00:00 till 2022-09-08 11:21:40+00:00
SMT_20220908_111450.gpx	13	2022-09-08 11:22:39+00:00 till 2022-09-08 11:24:21+00:00
SMT_20220616_231523.gpx	11	2022-06-16 23:17:55+00:00 till 2022-06-16 23:18:16+00:00
SMT_20220729_112533.gpx	21	2022-07-29 12:19:23+00:00 till 2022-07-29 12:33:04+00:00
SMT_20220729_112533.gpx	12	2022-07-29 12:26:18+00:00 till 2022-07-29 12:36:15+00:00
SMT_20220729_112533.gpx	11	2022-07-29 12:21:20+00:00 till 2022-07-29 12:37:29+00:00
SMT_20220704_112103.gpx	21	2022-07-04 11:46:05+00:00 till 2022-07-04 12:00:57+00:00
SMT_20220704_112103.gpx	12	2022-07-04 11:48:49+00:00 till 2022-07-04 12:35:14+00:00
SMT_20220704_112103.gpx	11	2022-07-04 11:41:27+00:00 till 2022-07-04 11:58:58+00:00
SMT_20220704_112103.gpx	13	2022-07-04 11:56:33+00:00 till 2022-07-04 12:40:36+00:00
SMT_20220704_112103.gpx	22	2022-07-04 11:49:29+00:00 till 2022-07-04 12:25:01+00:00
SMT_20220704_112103.gpx	23	2022-07-04 11:54:52+00:00 till 2022-07-04 12:33:48+00:00
SMT_20220704_112103.gpx	24	2022-07-04 11:59:40+00:00 till 2022-07-04 12:34:54+00:00
SMT_20220825_112057.gpx	12	2022-08-25 11:25:22+00:00 till 2022-08-25 11:25:43+00:00
SMT_20220712_111713.gpx	11	2022-07-12 12:29:00+00:00 till 2022-07-12 12:30:52+00:00
SMT_20220628_111612.gpx	11	2022-06-28 11:19:17+00:00 till 2022-06-28 11:19:27+00:00
SMT_20220702_231622.gpx	21	2022-07-03 00:14:37+00:00 till 2022-07-03 00:47:11+00:00
SMT_20220702_231622.gpx	12	2022-07-03 00:22:20+00:00 till 2022-07-03 00:52:30+00:00
SMT_20220702_231622.gpx	11	2022-07-03 00:29:05+00:00 till 2022-07-03 00:52:25+00:00
SMT_20220702_231622.gpx	13	2022-07-03 00:21:33+00:00 till 2022-07-03 00:52:30+00:00
SMT_20220702_231622.gpx	22	2022-07-03 00:11:26+00:00 till 2022-07-03 00:50:35+00:00
SMT_20220702_231622.gpx	23	2022-07-03 00:14:24+00:00 till 2022-07-03 00:51:28+00:00
SMT_20220702_231622.gpx	24	2022-07-03 00:16:28+00:00 till 2022-07-03 00:51:10+00:00
SMT_20220706_112237.gpx	21	2022-07-06 12:48:26+00:00 till 2022-07-06 12:48:49+00:00
SMT_20220803_232212.gpx	21	2022-08-03 23:23:53+00:00 till 2022-08-03 23:23:54+00:00
SMT_20220803_232212.gpx	22	2022-08-03 23:26:19+00:00 till 2022-08-03 23:27:55+00:00
SMT_20220803_232212.gpx	23	2022-08-03 23:29:46+00:00 till 2022-08-03 23:34:09+00:00
SMT_20220803_232212.gpx	24	2022-08-03 23:34:19+00:00 till 2022-08-03 23:38:20+00:00

BIBLIOGRAPHY

[1] Measurement of the multi-TeV neutrino interaction cross-section with IceCube using earth absorption. Nature, 551(7682):596–600, nov 2017.

[2] M. G. Aartsen, M. Ackermann, J. Adams, J. A. Aguilar, et al. Neutrino astronomy with the next generation icecube neutrino observatory, 2019.

[3] M.G. Aartsen, M. Ackermann, J. Adams, J.A. Aguilar, et al. The IceCube neutrino observatory: instrumentation and online systems. Journal of Instrumentation, 12(03):P03012–P03012, mar 2017.

[4] J.A. Aguilar, P. Allison, J.J. Beatty, et al. Design and sensitivity of the radio neutrino observatory in greenland (RNO-g). Journal of Instrumentation, 16(03):P03025, mar 2021.

[5] P. Allison, R. Bard, J. J. Beatty, D. Z. Besson, C. Bora, C.-C. Chen, C.-H. Chen, P. Chen, A. Christenson, A. Connolly, J. Davies, M. Duvernois, B. Fox, R. Gaior, P. W. Gorham, K. Hanson, J. Haugen, B. Hill, K. D. Hoffman, E. Hong, S.-Y. Hsu, L. Hu, J.-J. Huang, M.-H. A. Huang, A. Ishihara, A. Karle, J. L. Kelley, D. Kennedy, I. Kravchenko, T. Kuwabara, H. Landsman, A. Laundrie, C.-J. Li, T. C. Liu, M.-Y. Lu, L. Macchiarulo, K. Mase, T. Meures, R. Meyhandan, C. Miki, R. Morse, J. Nam, R. J. Nichol, G. Nir, A. Novikov, A. O'Murchadha, C. Pfendner, K. Ratzlaff, M. Relich, M. Richman, L. Ritter, B. Rotter, P. Sandstrom, P. Schellin, A. Shultz, D. Seckel, Y.-S. Shiao, J. Stockham, M. Stockham, J. Touart, G. S. Varner, M.-Z. Wang, S.-H. Wang, Y. Yang, S. Yoshida, and R. Young and. Performance of two askaryan radio array stations and first results in the search for ultrahigh energy neutrinos. Physical Review D, 93(8), apr 2016.

[6] G A Askaryan. Excess negative charge of an electron-photon shower and the coherent radio emission from it. Zhur. Eksptl'. i Teoret. Fiz., 41, 8 1961.

[7] S.W. Barwick, E.C. Berg, D.Z. Besson, G. Binder, W.R. Binns, D.J. Boersma, R.G. Bose, D.L. Braun, J.H. Buckley, V. Bugaev, S. Buitink, K. Dookayka, P.F. Dowkontt, T. Duffin, S. Euler, L. Gerhardt, L. Gustafsson, A. Hallgren, J.C. Hanson, M.H. Israel, J. Kiryluk, S.R. Klein, S. Kleinfelder, H. Niederhausen, M.A. Olevitch, C. Persichelli, K. Ratzlaff, B.F. Rauch, C. Reed, M. Roumi, A. Samanta, G.E. Simburger, T. Stezelberger, J. Tatar, U.I.

Uggerhoj, J. Walker, G. Yodh, and R. Young. A first search for cosmogenic neutrinos with the ARIANNA hexagonal radio array. Astroparticle Physics, 70:12–26, oct 2015.

[8] S.W. Barwick, E.C. Berg, D.Z. Besson, G. Gaswint, C. Glaser, A. Hallgren, J.C. Hanson, S.R. Klein, S. Kleinfelder, L. Köpke, I. Kravchenko, R. Lahmann, U. Latif, J. Nam, A. Nelles, C. Persichilli, P. Sandstrom, J. Tatar, and E. Unger. Observation of classically `forbidden' electromagnetic wave propagation and implications for neutrino detection. Journal of Cosmology and Astroparticle Physics, 2018(07):055–055, jul 2018.

[9] G. Bellini, L. Ludhova, G. Ranucci, and F. L. Villante. Neutrino oscillations. Advances in High Energy Physics, 2014:1–28, 2014.

[10] Federica Bradascio. Search for high-energy neutrinos from agn cores, 2019.

[11] S. Buitink, J. Bacelar, R. Braun, G. de Bruyn, H. Falcke, O. Scholten, K. Singh, B. Stappers, R. Strom, and R. al Yahyaoui. The numoon experiment: first results, 2008.

[12] Jr. Cowan, C. L., F. Reines, F. B. Harrison, H. W. Kruse, and A. D. McGuire. Detection of the Free Neutrino: A Confirmation. Science, 124(3212):103–104, July 1956.

[13] Scott Dodelson. Modern Cosmology. Academic Press, Amsterdam, 2003.

[14] K. Eguchi, S. Enomoto, K. Furuno, J. Goldman, H. Hanada, H. Ikeda, K. Ikeda, K. Inoue, et al. First results from kamland: Evidence for reactor antineutrino disappearance. Physical Review Letters, 90(2), jan 2003.

[15] S. Fukuda, Y. Fukuda, T. Hayakawa, E. Ichihara, M. Ishitsuka, Y. Itow, T. Kajita, J. Kameda, K. Kaneyuki, S. Kasuga, Kobayashi, et al. The Super-Kamiokande detector. Nuclear Instruments and Methods in Physics Research A, 501(2):418–462, April 2003.

[16] Thomas K. Gaisser, Ralph Engel, and Elisa Resconi. Cosmic Rays and Particle Physics. Cambridge University Press, 2 edition, 2016.

[17] C. Glaser, D. García-Fernández, A. Nelles, J. Alvarez-Muñiz, S. W. Barwick, D. Z. Besson, B. A. Clark, A. Connolly, C. Deaconu, K. D. de Vries, J. C. Hanson, B. Hokanson-Fasig, R. Lahmann, U. Latif, S. A. Kleinfelder, C. Persichilli, Y. Pan, C. Pfendner, I. Plaisier, D. Seckel, J. Torres, S. Toscano, N. van Eijndhoven, A. Vieregg, C. Welling, T. Winchen, and S. A. Wissel. NuRadioMC: simulating the radio emission of neutrinos from interaction to detector. European Physical Journal C, 80(2):77, January 2020.

[18] C. Glaser, D. García-Fernández, A. Nelles, J. Alvarez-Muñiz, S. W. Barwick, D. Z. Besson, B. A. Clark, A. Connolly, C. Deaconu, K. D. de Vries, J. C. Hanson, B. Hokanson-Fasig, R. Lahmann, U. Latif, S. A. Kleinfelder, C. Persichilli, Y. Pan, C. Pfendner, I. Plaisier, D. Seckel, J. Torres, S. Toscano, N. van Eijndhoven, A. Vieregg, C. Welling, T. Winchen, and S. A. Wissel. NuRadioMC: simulating the radio emission of neutrinos from interaction to detector. The European Physical Journal C, 80(2), jan 2020.

[19] Christian Glaser, Anna Nelles, Ilse Plaisier, Christoph Welling, Steven W. Barwick, Daniel García-Fernández, Geoffrey Gaswint, Robert Lahmann, and Christopher Persichilli. NuRadioReco: a reconstruction framework for radio neutrino detectors. The European Physical Journal C, 79(6), jun 2019.

[20] M. Grabe. Measurement Uncertainties in Science and Technology. Springer Berlin Heidelberg, 2005.

[21] R.A. Herman. A Treatise on Geometrical Optics. Creative Media Partners, LLC, 2019.

[22] Michael M. Herron and Chester C. Langway. Firn densification: An empirical model. Journal of Glaciology, 25(93):373–385, 1980.

[23] Nils Heyer and Christian Glaser. First-principle calculation of birefringence effects for in-ice radio detection of neutrinos. The European Physical Journal C, 83(2):124, Feb 2023.

[24] Tim Huege and Dave Besson. Radio-wave detection of ultra-high-energy neutrinos and cosmic rays. Progress of Theoretical and Experimental Physics, 2017(12), nov 2017.

[25] J.D. Jackson. Classical Electrodynamics. Wiley, 1998.

[26] Demosthenes Kazanas. Neutrinos from AGN. In Felix A. Aharonian and Heinz J. Völk, editors, High Energy Gamma-Ray Astronomy: International Symposium, volume 558 of American Institute of Physics Conference Series, pages 370–380, April 2001.

[27] I. Kravchenko, S. Hussain, D. Seckel, D. Besson, E. Fensholt, J. Ralston, J. Taylor, K. Ratzlaff, and R. Young. Updated results from the RICE experiment and future prospects for ultra-high energy neutrino detection at the south pole. Physical Review D, 85(6), mar 2012.

[28] B.R. Martin and G. Shaw. Particle Physics. Manchester Physics Series. Wiley, 2017.

[29] Tobias Melson, Hans-Thomas Janka, Robert Bollig, Florian Hanke, Andreas Marek, and Bernhard Müller. Neutrino-driven explosion of a 20 solar-mass star in three dimensions enabled by strange-quark contributions to neutrino–nucleon scattering. The Astrophysical Journal Letters, 808(2):L42, jul 2015.

[30] B. Oeyen, I. Plaisier, A. Nelles, C. Glaser, and T. Winchen. Effects of firn ice models on radio neutrino simulations using a RadioPropa ray tracer. In 37th International Cosmic Ray Conference. 12-23 July 2021. Berlin, page 1027, March 2022.

[31] H. Schoorlemmer, K. Belov, A. Romero-Wolf, D. García-Fernández, V. Bugaev, et al. Energy and flux measurements of ultra-high energy cosmic rays observed during the first anita flight. Astroparticle Physics, 77:32–43, 2016.

[32] Tobias Winchen. RadioPropa — a modular raytracer for in-matter radio propagation. EPJ Web of Conferences, 216:03002, 2019.

[33] Kane Yee. Numerical solution of initial boundary value problems involving maxwell's equations in isotropic media. IEEE Transactions on Antennas and Propagation, 14(3):302–307, 1966.