Project: A010 - Exploring the Atmosphere near Clouds - Changes to Aerosol

Candidate Number: 1014594

Supervisors: Dr Adam Povey, Prof R. G. Grainger

Abstract

The understanding of aerosol cloud interactions are crucial to future climate models. Accurately determining aerosol statistics near clouds helps determine the scale of these interactions. Algorithms were created to find the shadow of clouds in a satellite data set and Aerosol Optical Depth (AOD) was studied in the shadow of clouds. Aerosol data was looked at in cloud shadow as a potential way to understand how to correct for effects that distort aerosol retrievals near cloud edges. The AOD retrievals in shadow near the cloud edge differ from outside the shadow retrievals. The cloud shadow impacts the way AOD varies as a function of distance from cloud edge.

1 Introduction

The primary objective of this project is to study the interaction between aerosols and clouds near cloud edges. Specifically, the aim is to examine data within and without of cloud shadow in order to observe effects caused by the cloud which are unaccounted for in current models. By studying the errors in the cloud shadow the impact and nature of these errors on the near cloud aerosol properties can be ascertained. This will allow methods to be developed to correct for these errors and more accurately determine aerosol cloud interactions near the cloud boundary.

Aerosols are fine (typically less than a 1 micrometre diameter) solid or liquid particles in the air with a variety of sources, sinks and physical properties. They can be anthropogenic or natural. Some illustrative examples of aerosols are sub-saharan dust and sulfates. [9]

One of the key reasons aerosols and clouds interact is that an aerosols can act as cloud condensation nucleus (CCN). A CCN will allow a water to collect around it in typical atmospheric conditions, thus a CCN will facilitate the formation of cloud droplets. Water droplets struggle to form on their own as water evaporates more quickly when it is a smaller droplet as it has a higher curvature.[10] A water droplet will grow until the rate of evaporation and the rate of condensation are equal. The primary factor in the rate of evaporation is the surface area of the droplet, as the surface area of the droplet increases the rate of evaporation decreases, this is due to the curvature of the drop decreasing as surface area increases. The primary factor in the rate of condensation is the relative humidity as it represents the available water to be deposited on the drop. When water condenses onto a CCN it will already have reached a large size and thus will evaporate slower and it will be easier for the droplet to come into equilibrium.[9]

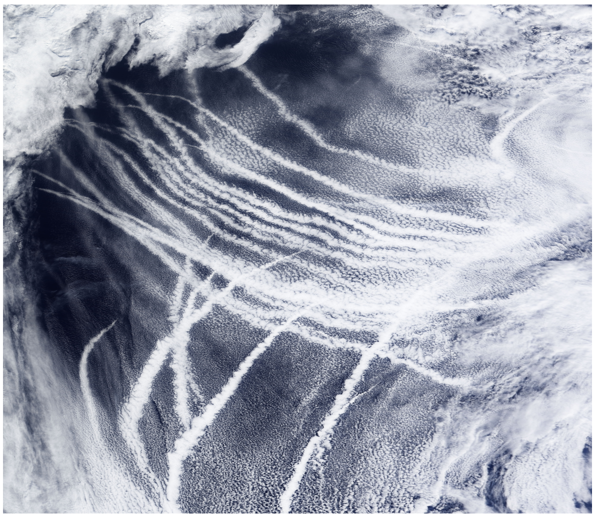

Figure 1: An illustrative image of Ship Tracks [6]

Figure 1 shows an image of a ship track which provides a great illustration of aerosol cloud interactions. Ship tracks are clouds formed in the wake of a ships path due to the aerosols produced by the ship entering the atmosphere. This provides a very clear example of aerosols acting as a CCN. [6]

Thus, aerosols that act as CCNs can affect the properties of clouds by adjusting the equilibrium of the droplets. These interactions can have profound implications on the climate. The best known demonstration of this is the Twomey effect: when larger amounts of aerosols are present in clouds more water droplets will form, however, the amount of water available will not have changed. As a result particles will be relatively smaller in size. As cloud albedo is higher than the surface albedo, cloud cover will contribute heavily to the planetary albedo. Light is reflected off the water droplets and although each drop is smaller and thus reflects less light individually in regions of high aerosol content the combined surface area of the cloud is higher creating a higher net reflectivity. Over short time scales and small distances the water in the atmosphere will not have sufficient time to respond to the Twomey effect and the local atmosphere will approximate a closed system. Thus, the Twomey effect will lead to brighter cloud cover and a higher planetary albedo. This has a net cooling effect on the Earth by impacting the Earth's radiative balance.[9] Each type of aerosol will have a different magnitude of effect due to other properties inherent to that aerosol such as its size and hygroscopicity.

It is thus important to understand large scale impacts of aerosol cloud interactions. Clouds are large and thus cannot be studied under laboratory conditions, additionally they cover too much of the planets surface to be studied representatively with an aircraft. As such one must look obliquely at changes in properties of the cloud (i.e. optical depth, effective radius) to understand the impact on the microphysical properties of the cloud (i.e. droplet size and number). The aerosol optical depth near a cloud edge is the primary property we are concerned with measuring. Aerosol optical thickness (τ) is the natural logarithm of the ratio between power of the incident power entering the aerosol (P_{inc}) and the light leaving the aerosol (P_{trans}):

$$\tau = ln(\frac{P_{inc}}{P_{trans}}) \tag{1}$$

AOD (Aerosol Optical Depth) can be used to measure the amount of aerosols in the air and has the added benefit of being easy to measure and having a long history of being measured.

The data sets used in this project are from MODIS[5][8] and AATSR[11], as processed by the ORAC aerosol retrieval algorithm.[12] These measure the radiance emitted and reflected by the Earth at approximately 1km resolution. Combining this with meteorological data and radiative transfer models allow the calculation of AOD. The location defined by the latitude and longitude where a beam from the satellite intersects with the ground.

Some other useful quantities linked to AOD are the Aerosol Index and Angström exponent. The aerosol index is a measure of how much the reflected UV radiance of the actual atmosphere differs from a pure molecular atmosphere. The Angström exponent is a measure of how the optical depth of an aerosol depends on the wavelength of the light being scattered.

The plot below shows the variation in observed AOD, Angström exponent, and Aerosol Index as a function of distance to the nearest cloud for two satellite sensors.

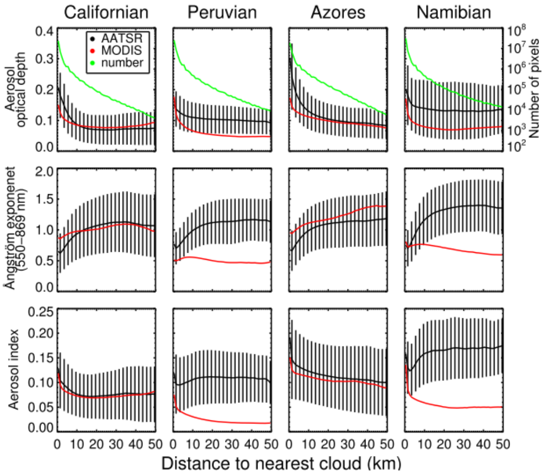

Figure 2: Reproduced from [2]

Figure 2 shows from previous work [2] that there was a clear exponential rise in aerosol optical depth as it approaches the cloud with the aerosol optical depth settling to a constant at a range of about 15 kilometers. There are many potential causes for this increase, first that there is a region between clouds and clear sky known as 'the twilight zone'. This is the region in the air near clouds which is full of forming and evaporating clouds and hydrated aerosols. In this region AOD is found to be around 13% higher in the visible spectrum and 23% higher in the near infrared spectrum. Essentially there is not a discrete boundary between the cloud and the clear sky.[4] This could help explain the regions of high aerosol optical depth near the cloud.

Another effect that could be studied further is the effect of light coming out of the side of the cloud. Light entering a cloud will not pass straight through the cloud nor will it all reflect. Some will reflect off the water droplets inside the cloud and thus will exit the cloud at all points on its surface. The aerosol retrievals make a 'plane-parallel' assumption that the physical properties of the atmosphere will only vary with depth, thus we assume that no energy can exit the side of the clouds. As light does exit the side of the clouds this assumption will have an error associated with it which could produce this 'effect'. As a significant portion of light hitting the cloud will enter from the top surface of the cloud we can assume that the light entering the cloud will come in equal amounts from all sides of the cloud.

In previous research the effect of aerosols near the cloud on aerosol cloud interactions have been estimated. It was found that by removing aerosols within 15km of the cloud edge one can reduce the cooling caused by cloud albedo by 40% and the cooling caused by cloud fraction by 70%. [2] This means that there is a clear and apparent interest in understanding aerosol properties in the near cloud region in order to create more accurate theory of cloud-aerosol interactions.

In order to produce reliable data the data sets account for the way light is reflected off of the Earth's surface. This is accomplished by using the Bi-Directional Reflectance Function (BRDF). By calculating this at a point one can find how light is scattered. The BRDF is used in retrievals in combination with the incident radiance on the surface of the planet to produce a model of how radiation is reflected across the Earth's surface. [3] This is difficult to measure and can often be wrong, for example if it rains. This error becomes more significant and large the brighter the surface is so in the cloud's shadow we should be able to retrieve more accurate aerosol data. Additionally as cloud shadows occur at quite close range we can retrieve more accurate aerosol data in the close range area we care about. Additionally the light coming out of the side of clouds will be more apparent in the data in shaded regions as it will represent a higher percentage of the radiance.

2 Methods

2.1 Cloud Shadows

In order to determine the effects of cloud shadow on measured aerosol properties we first must determine the latitude and longitude of the cloud shadow. There are 6 angles we need to locate the shadow: the measured longitude of the cloud λ_i, the measured latitude of the cloud ϕ_i, the sensor azimuth θ_{sens}, the sensor zenith ζ_{sens}, the solar azimuth θ_{sol}, and the solar zenith ζ_{sol}. Combined with the cloud top height h,x we can determine the exact position of the cloud's shadow.

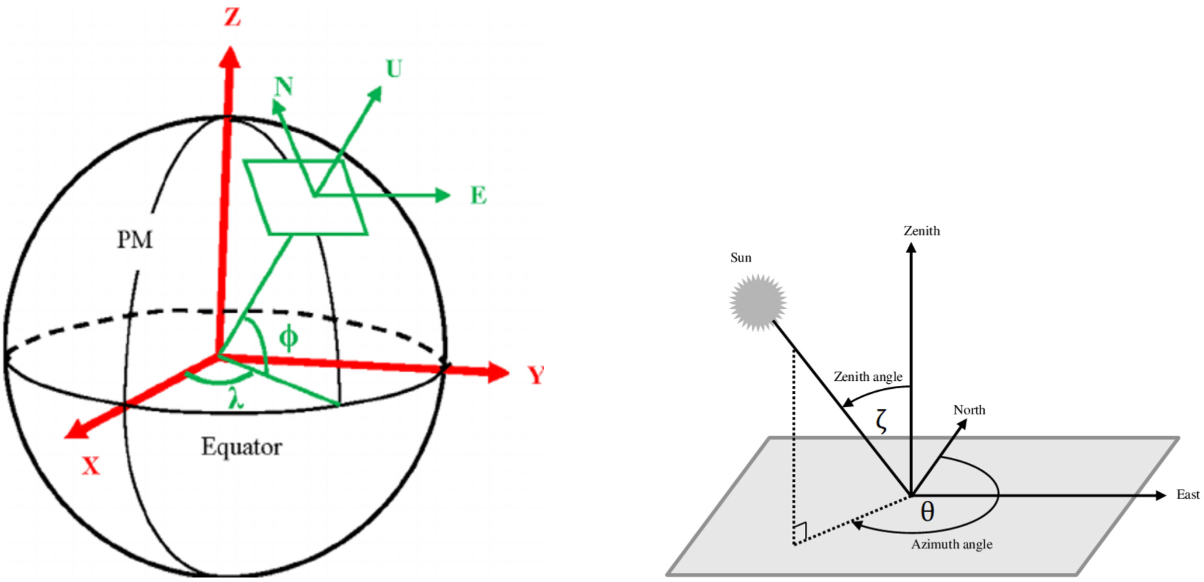

Figure 3: (a) ENU frame (green) as located in the ECEF (red) frame for the Earth[13], (b) Local example of the solar azimuth and zenith angles.[7]

The azimuth and zenith angles are defined in local East North Up (ENU) frames while latitude and longitude are defined in the Earth-centred, Earth-fixed (ECEF) rotating frame. The ENU frame is a local cartesian frame where the x, y, and z axes are defined by the east, north and up directions at that point respectively, The ECEF frame is a rotating frame where the origin is at the centre of the Earth, the z axis goes through the north pole and the x axis is defined as going through the intersection between the equator and the prime meridian. These reference frames are depicted in figure 3a. Using the azimuth and zenith angles (as shown in figure 3b for the solar case) one can determine the equation for the line connecting the cloud's recorded position (as determined by the satellite) with the satellite and the cloud's real position. This line can then be transformed into ECEF coordinates with the latitude and longitude of the cloud's false position and see where this intersects with the equation of a surface of a sphere at radius equal to the Earth's radius R_{Earth}, plus the cloud top height. In doing this we are assuming that the Earth is a sphere when it is in fact an ellipsoid. This is sensible as the cloud shadows do not propagate over a large enough distance for the difference between the shadow's using the sphere model and the ellipse model to be significantly different.

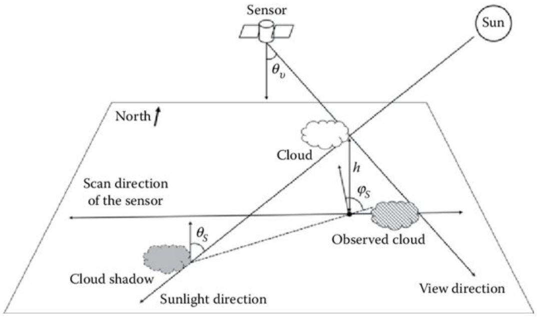

Figure 4: Relation between geometry of sun, satellite and cloud shadow.[14]

The equation for the unit vector of the ray in ECEF coordinates is given by $\boldsymbol{u}$ and the exact equation can be seen in appendix A. The equation for the ray in ECEF coordinates is given by:

$$\boldsymbol{u} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = r_{fc} + \hat{\boldsymbol{u}} d \tag{2}$$

Where d is an arbitrary distance and r_{fc} is the position of the cloud image which was found using the given latitude, longitude, and the Earth's radius. Conversions can be made between ECEF position and latitude and longitude as it is effectively a spherical polar coordinate system.

We can the input this into the equation for the surface of a sphere at the cloud top height we are looking for:

$$\boldsymbol{u}^2 = x^2 + y^2 + z^2 = (R_E + h)^2 \tag{3}$$

We can infer the latitude and longitude of the actual cloud. This process can then be done in reverse to find the position of the actual cloud as shown in figure 4.

2.2 Distance to closest clouds

The edges of clouds are also needed so that the distance to the nearest cloud for an aerosol pixel can be calculated. This is done by looping over the matrix of data and flagging cloud edges (pixels marked as clouds that are adjacent, or diagonally adjacent, to clear air). Then for each clear sky element we loop over every element of the array which is marked as a cloud edge and then we calculate the difference of the latitude and longitude of the cloud edge and the clear sky point and multiplying by the Earth's radius to convert to a distance. Over small distances we can assume the Earth's surface is locally flat so we can use Pythagoras' theorem to determine the distance between the cloud and the aerosol and loop over until we find the closest pixel.

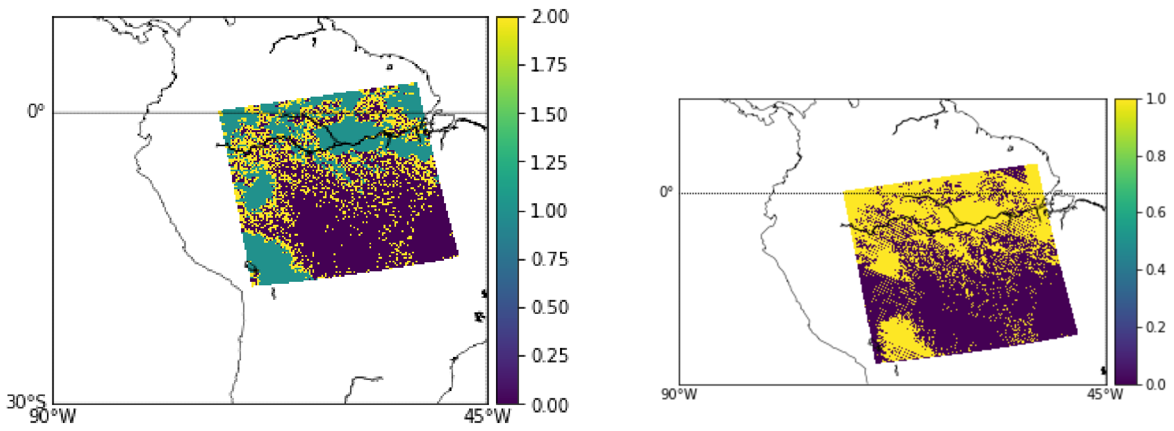

Figure 5: (a) The Edges of clouds as flagged by the written code (2 = Edge, 1 = Cloud, 0 = Clear Sky). (b) Shadows propogated from the same dataset (1 = Shadow, 0 = Clear Sky).

2.3 Satellite Datasets

Above in figure 5 are a selection of images that show the cloud edge finder and shadow finder algorithms being tested on the Aqua MODIS data set taken on June 15th 2019 from 17:40 to 17:45. The cloud edges are at the edges of the clouds as flagged by the datasets and shadows can be seen outside of the cloud. The later analysis was produced from the ORAC/AATSR data sets as the number of data points from MODIS data is not enough to form a conclusion. This is due to MODIS data being stored in a lower resolution, 5km by 5km pixels, for cloud data than ORAC data, 1km by 1km pixels, and aerosol data is stored in an even lower resolution, 10km by 10km pixels. This lack of resolution and the inability to quickly match between the pixels in cloud data sets and aerosol data sets, due to being stored in different resolutions, means that ORAC data is better for this project.

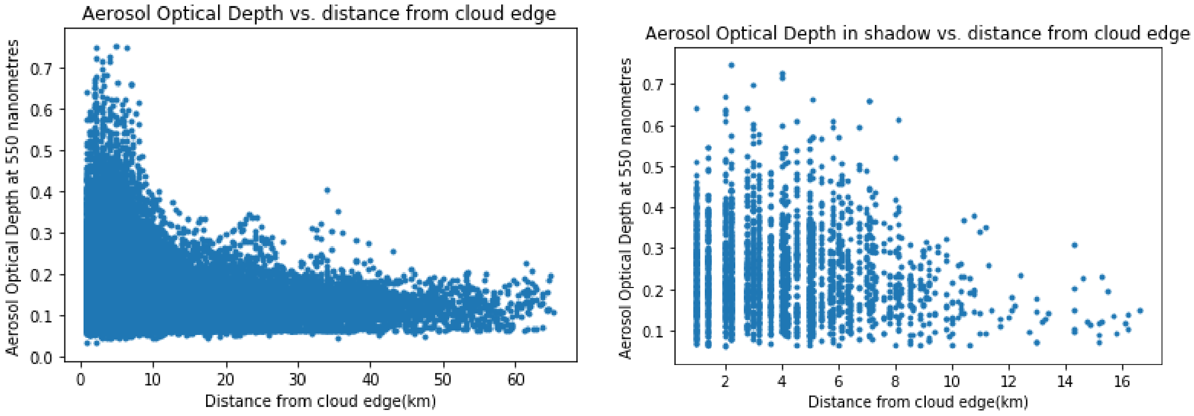

Figure 6: Scatterplots of (a) all AOD data retrieved, (b) AOD data retrieved in the cloud shadow.

3 Data

In figure 6 above and 7 below are histograms and scatter plots showing the aerosol optical depth as a function of distance from the cloud both in and outside of the shadow using ORAC/AATSR data sets. The data for these are taken from 62.3 to 68 degrees North and 137.1 to 144.4 degrees east, which is over Siberia.

There are 24,881 data points in the out of shadow histogram and 2,165 data points in the inside of shadow histogram.

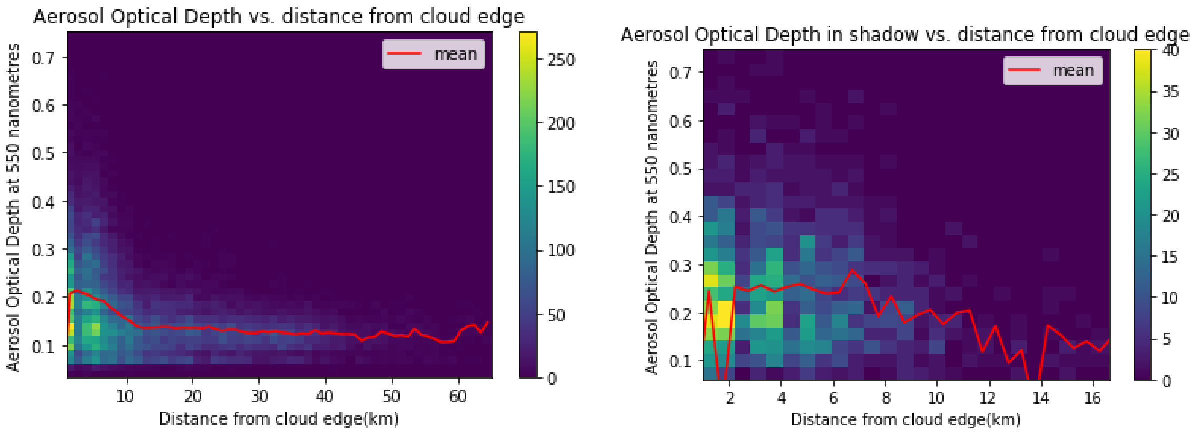

Figure 7: Histograms of (a) all AOD data retrieved, (b) AOD data retrieved in the cloud shadow.

Additionally shown in figure 8 below are similar plots for the region 1km away from (or equivalently the adjacent pixels to) the cloud shadow. There are 1,602 data points total in these plots.

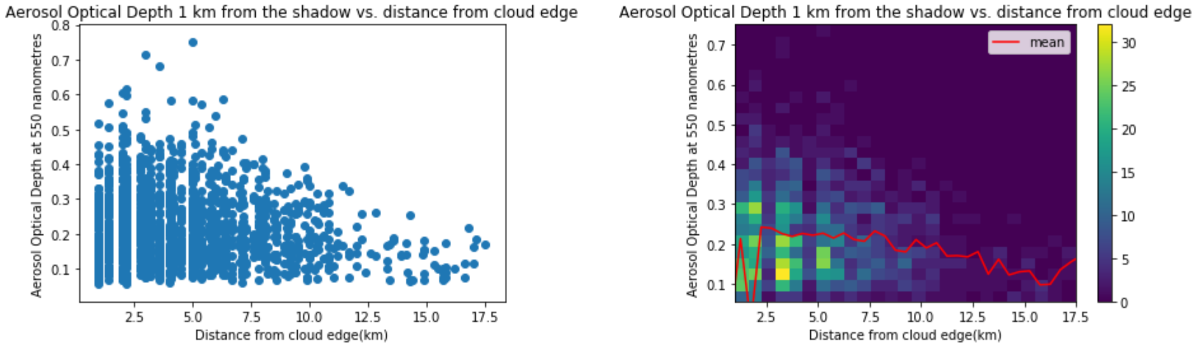

Figure 8: All AOD data retrieved adjacent to the cloud shadow represented as (a) a scatterplot, (b) a histogram.

Below in figure 9 is a rescaled version of the figure 7a to more clearly focus on the resemblance to figure 2.

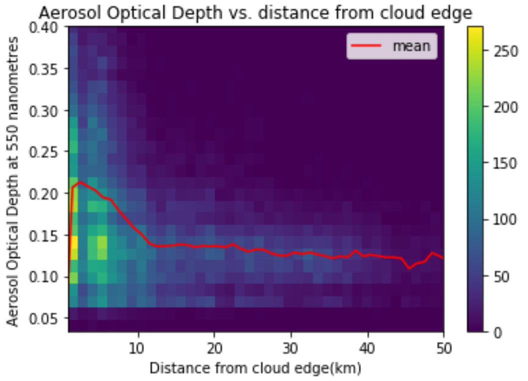

Figure 9: A rescaled version of figure 7a with distances spanning to 50 km from the cloud edge

4 Results

In the out of shadow region we have clearly reproduced the exponential relationship from the previous paper as seen in figure 9. This was done using ORAC/AATSR data and seeing figures 6 and 7 in a region of shadow and averaged is similar. However the data appears to be a lot more linear. There are fewer high AOD pixels near the cloud edge in cloud shadow as well. There are quite a few possible explanations for this.

As expected, we have removed some of the the highest outlying data points in the Aerosol Optical Depth data. However, the data points in figures 6a and 6b still span the same range of AOD and look quite similar in terms of number density as seen in figure 7. The greater linearity of the data may just be because we are looking over a smaller range (16km) than in the other graph (60km). As it appears to remove some of the outlying data points it seems that we can retrieve more accurate aerosol data in the cloud's shadow.

The lack of high AOD values at short distances to the cloud in cloud shadow provides insights into near cloud behaviour. The area near the cloud will have less incident and transmitted light present. Thus the unaccounted for light will increase P_{trans} by a proportionately greater amount compared to P_{inc} as they will both be smaller, increasing the AOD. As such our assumptions that do not account for this light will be worse and the AOD will be reduced in this region.

Figure 7a and 7b show a peak in data retrieved near the cloud edge at an AOD of about 0.1 to 0.2. The histogram seems to show a more linear distribution of points. The histogram much more strongly shows a peak at these values in the shadow as compared to the average. But, as there a lot less data points this is not completely illuminating. The peak in data points at a certain distance is due to the data set as opposed to being an inherent property of these relations. The large number of data points occurring at an AOD of 0.1 to 0.2 is a property of the relationship between aerosol and clouds.

In figure 10 below are contour plots to illustrate some of these points:

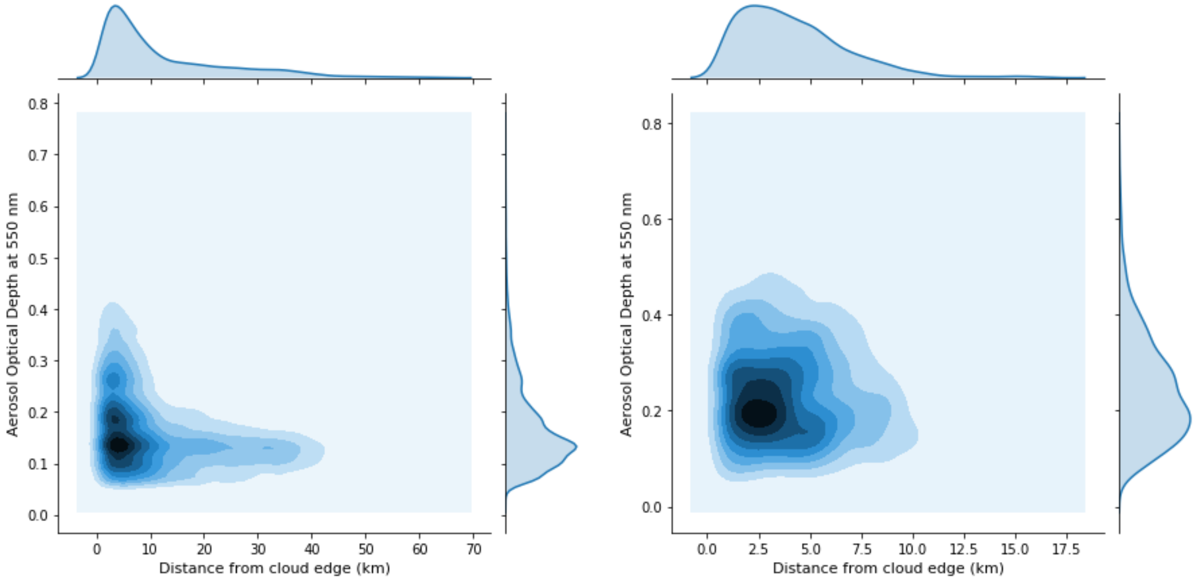

Figure 10: (a) A contour plot of the AOD data taken as a function of distance from the cloud edge, (b) The same plot in cloud shadow.

In the contour plot of the normal data it can be seen that there are two clear peaks in the number of data points. Both are at about 2.5km from the cloud edge but the smaller one occurs at an AOD of 0.2 and the larger one occurs at an AOD of about 0.1. However, only the peak at an AOD of 0.2 can be observed in the contour plot from the data in shadow. Clearly the peak at 0.2 AOD is due to the cloud shadow. This can be attributed to the incident power not being corrected for the cloud shadow in the retrievals so it remains relatively unchanged but the transmitted power will be decreased due to there being less incident light in the cloud shadow. This will cause the modal value to AOD to be smaller near the cloud edge. Thus the modal aerosol optical depth of 0.1 is more accurate.

From this and a lower maximum value of the AOD at short distances show that in the shadow near clouds the data has a higher modal value but a much smaller range. There may be rapidly condensing and evaporating clouds that are not flagged because they lie in the shade of the cloud itself and as such have less light reflecting off of them to identify them by. Thus the values for AOD at 0.2 may be clouds that should be flagged as such but are not as they lie in the cloud shadow.

However the retrievals were not adjusted to factor in the data being retrieved through the cloud shadow. This means that a variety of quantities will have an associated error. Specifically the BRDF should be corrected in this region to factor in the cloud shadow. However we can compare the data retrieved adjacent to the cloud shadows where the retrievals will be reasonably correct and will demonstrate properties of the cloud shadow due to being in an equilibrium state with the atmosphere in the cloud shadow and outwith it.

Below in figure 11 is the contour plot showing the AOD at a distance of 1km from the cloud shadow. This data looks very similar to the data presented by the data in the cloud shadow as does comparing figure 8 with figure 6b and 7b. This implies at near distances pixels in the cloud shadow or near the cloud shadow have similar properties and act similarly. Thus we can assume that the results derived from the data in the cloud shadow are good enough to derive some information from. However, there will is significant error in this as the shadow could fall into an adjacent pixel from the edges causing these effects.

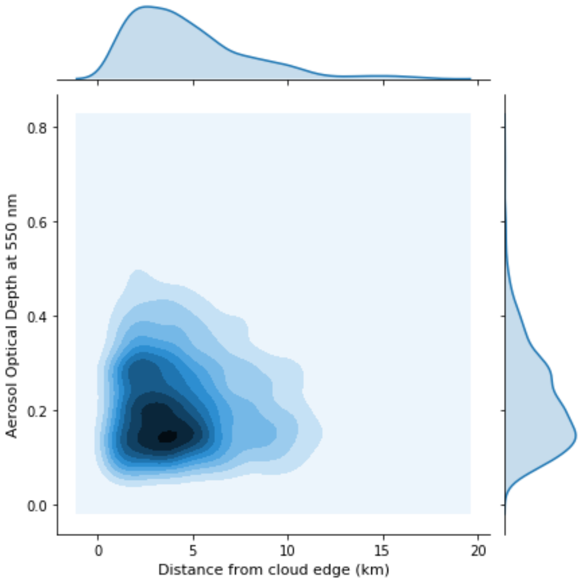

Figure 11: A contour plot of the AOD adjacent to the cloud shadow.

There are significant sources of error in these results. There is a significant uncertainty on the depth of the cloud due to the 2D model is not presented by the data sets. Any impact on AOD caused by the proximity to clouds will be amplified by the increased depth of the clouds. The cloud depth can be deduced by assuming the atmosphere is adiabatic which is not always true.

Also there could be clouds outside the swath that was scanned by the satellite that may be closer to an aerosol pixel. However, there are minimal pixels along the cloud edge so the uncertainty associated with undetected clouds will be small.

In addition, there will be an uncertainty associated with a cloud pixel propagating shadows incorrectly. As such a cloud shadow for a pixel may fall over multiple pixels. This will lead to not being completely certain about the number of pixels that are present in shadow or in sunlight.

This uncertainty will mainly come from the cloud shadow in reality falling over multiple pixels, but the algorithm assumes that the cloud shadow will only fall on one pixel alone. Additionally if the pixel is not completely covered by cloud the cloud shadow may fall in such a way that a pixel that is called shaded is in fact not shaded.

There are significant issues that were encountered with the operation of this algorithm. First of all by the nature of cloud shadows occurring near the clouds they are generated from, it was found in the region studied that no clouds were generated farther than 16km away. It was difficult to retrieve aerosol data in cloud shadows far away from the cloud. Given that the data appears to stabilise at around 15km from the cloud this was not a serious issue.

There are also issues in the processing speed of the program which makes it difficult to inspect a swath of clouds at once. The accuracy gained from using the ORAC/AATSR data set by gaining data points means that the program takes twenty five times longer as there are 25 times more pixels to run over. Thus the region considered had to be reduced in size in order to generate data at a sensible rate. This heavy discrimination of data may remove data that changes observed patterns.

5 Conclusions

The cloud shadow changes the distribution of the aerosol optical depth distribution near the cloud in particular. Thus, the properties of aerosols in shadow of clouds are useful to study in understanding aerosol cloud interactions. It would be useful in further research to discriminate between data in cloud shadow and in clear sunlight for aerosol properties. But, there is no significant change in data overall which implies that there are other factors present impacting AOD reading near clouds.

My algorithm is a reasonably effective way of providing finding the cloud shadow; however it is inefficient. As such, for further research examining data in the cloud shadow a more effective means must be created to find the cloud shadow. This could be done by measuring radiances and wavelengths of light coming from the Earth's surface to flag for cloud shadow from the data directly, thus including it in the retrievals.

The shadow over the ocean could be considered. This was discounted in the data taken as the ocean absorbs more light than the ground does so the difference between the absorbance of shadow on the ocean compared to normal ocean will be minuscule. Additionally, the reflectance of the ocean is currently believed to be well modelled. [1]

The data measured in shadows was retrieved using an aerosol retrieval algorithm. This algorithm runs on the assumption that the ground is not in shade and is instead lit by the correct amount of sunlight. This means that the data will not be completely accurate in this region.

However, the data shown in figure 8, which is the data retrieved adjacent to shadow, matches the trends shown in data retrieved in the cloud shadow. In this region the retrieval should be more accurate so it could be seen that the data should be reasonably accurate. Thus the conclusions made from data retrieved from shadows are somewhat acceptable.

As mentioned earlier the high AOD values in the near cloud region where not present in retrievals in the cloud shadow. This was due to multiple factors including higher chance of incorrect identification of the pixel as cloudless and light coming out of the side of the cloud being more dominant. Additionally, the modal value of AOD was 0.2 in the cloud shadow which helps identify the peak in data points at 0.2 AOD in all pixels as purely due to the cloud shadow and the retrieval not accounting for the decreased intensity of light reflected off the Earth's surface.

Therefore, the retrievals in the cloud shadow appears to highlight the issue of light coming out of the side of the cloud. It also appears to make problems presented by the 'twilight zone' larger. Further research should be conducted to quantify and further these observations.

References

[1] F. M. Bréon. "An analytical model for the cloud-free atmosphere/ocean system reflectance". In: *Remote Sensing of Environment* 43.2 (1993), pp. 179–192. DOI: 10.1016/0034-4257(93)90007-K.

[2] Matthew W. Christensen et al. "Unveiling aerosol-cloud interactions - Part 1: Cloud contamination in satellite products enhances the aerosol indirect forcing estimate". In: *Atmospheric Chemistry and Physics* 17.21 (2017), pp. 13151–13164. DOI: 10.5194/acp-17-13151-2017.

[3] Feng Gao et al. “MODIS bidirectional reflectance distribution function and albedo Climate Modeling Grid products and the variability of albedo for major global vegetation types”. In: *Journal of Geophysical Research: Atmospheres* 110.D1 (2005). DOI: 10.1029/2004JD005190. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2004JD005190.

[4] Ilan Koren et al. “On the twilight zone between clouds and aerosols”. In: *Geophysical Research Letters* 34.8 (2007), pp. 1–5. ISSN: 00948276. DOI: 10.1029/2007GL029253.

[5] R. Levy, C. Hsu, and et al. *MODIS Atmosphere L2 Aerosol Product.* 2015. DOI: 10.5067/MODIS/MYD04_L2.061.

[6] NASA Earth Observatory. *Ship Tracks South of Alaska.* [Online; accessed April 11, 2020]. 2009. URL: \url{https://earthobservatory.nasa.gov/images/37455/ship-tracks-south-of-alaska}.

[7] Julien Nou et al. “A new approach to the real-time assessment of the clear-sky direct normal irradiance”. In: *Applied Mathematical Modelling* 40 (Mar. 2016). DOI: 10.1016/j.apm.2016.03.022.

[8] S. Platnick et al. *MODIS Atmosphere L2 Cloud Product (06_L2).* 2015. DOI: 10.5067/MODIS/MYD06_L2.061.

[9] Steven C Sherwood et al. *Climate Science for Serving Society.* 2013, pp. 73–103. ISBN: 9789400766921. DOI: 10.1007/978-94-007-6692-1.

[10] Fredric W. Taylor. “Elementary Climate Physics”. In: Great Clarendon Street, Oxford: Oxford University Press, 2005, pp. 60–62. ISBN: 0-19-856734-0.

[11] ESA Aerosols CCI project team, G. de Leeuw, and T. Popp. *ESA Aerosol Climate Change Initiative (Aerosol CCI): Level 2 aerosol products from AATSR (ORAC Algorithm), Version 3.02.* 03/04/2020, https://catalogue.ceda.ac.uk/uuid/c41e248db8d74e22be25ce6b79e04bb6. 2016.

[12] Gareth E. Thomas et al. “Oxford-RAL Aerosol and Cloud (ORAC): aerosol retrievals from satellite radiometers”. In: *Satellite Aerosol Remote Sensing over Land.* Ed. by Alexander A. Kokhanovsky and Gerrit de Leeuw. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 193–225. ISBN: 978-3-540-69397-0. DOI: 10.1007/978-3-540-69397-0_7.

[13] Yansen Wang, Giap Huynh, and Chatt Williamson. “Integration of Google Maps/Earth with microscale meteorology models and data visualization”. In: *Computers Geosciences* 61 (Dec. 2013), 23–31. DOI: 10.1016/j.cageo.2013.07.016.

[14] Zhe Zhu et al. “Cloud and Cloud Shadow Detection for Landsat Images: The Fundamental Basis for Analyzing Landsat Time Series”. In: May 2018. DOI: 10.1201/9781315166636-1.

Bibliography

[1] B. Ervens et al. “CCN predictions using simplified assumptions of organic aerosol composition and mixing state: a synthesis from six different locations”. In: *Atmospheric Chemistry and Physics* 10.10 (2010), pp. 4795–4807. DOI: 10.5194/acp-10-4795-2010. URL: https://www.atmos-chem-phys.net/10/4795/2010/.

[2] Jane Gray Hurley. “Detection and Retrieval of Clouds from MIPAS”. In: (2008).

[3] Seoung-Soo Lee and Graham Feingold. “Precipitating cloud-system response to aerosol perturbations”. In: *Geophysical Research Letters* 37.23 (2010). DOI: 10.1029/2010GL045596. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2010GL045596. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2010GL045596.

[4] J. Liu and Z. Li. “Estimation of cloud condensation nuclei concentration from aerosol optical quantities: influential factors and uncertainties”. In: *Atmospheric Chemistry and Physics* 14.1 (2014), pp. 471–483. DOI: 10.5194/acp-14-471-2014. URL: https://www.atmos-chem-phys.net/14/471/2014/.

[5] Y. Liu et al. “Relationship between cloud radiative forcing, cloud fraction and cloud albedo, and new surface-based approach for determining cloud albedo”. In: *Atmospheric Chemistry and Physics* 11.14 (2011), pp. 7155–7170. DOI: 10.5194/acp-11-7155-2011. URL: https://www.atmos-chem-phys.net/11/7155/2011/.

[6] A. McComiskey and G. Feingold. “The scale problem in quantifying aerosol indirect effects”. In: *Atmospheric Chemistry and Physics* 12.2 (2012), pp. 1031–1049. DOI: 10.5194/acp-12-1031-2012. URL: https://www.atmos-chem-phys.net/12/1031/2012/.

[7] Bernard Pinty et al. “Theory”. In: *Journal of Geophysical Research* 105 (2000), pp. 99–112.

[8] Oliver Sus et al. “The Community Cloud retrieval for CLimate (CC4CL)-Part 1: A framework applied to multiple satellite imaging sensors”. In: *Atmospheric Measurement Techniques* 11.6 (2018), pp. 3373–3396. ISSN: 18678548. DOI: 10.5194/amt-11-3373-2018.

[9] M. Wang et al. “Aerosol indirect effects in a multi-scale aerosol-climate model PNNL-MMF”. In: *Atmospheric Chemistry and Physics* 11.11 (2011), pp. 5431–5455. DOI: 10.5194/acp-11-5431-2011. URL: https://www.atmos-chem-phys.net/11/5431/2011/.

A Equations

$$\boldsymbol{u} = \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} = \begin{pmatrix} \sin\zeta(\sin\lambda\sin\theta + \cos\lambda\sin\phi\cos\theta) - \cos\zeta\cos\lambda\cos\phi \\ \sin\zeta(\sin\lambda\sin\phi\cos\theta - \cos\lambda\sin\theta) - \sin\lambda\cos\phi\cos\zeta \\ -(\sin\theta\cos\zeta + \cos\theta\cos\phi\cos\zeta) \end{pmatrix}$$

B Code

B.1 Cloud edge flagging

clouds is the matrix containing cloud flags.

```
i = 0
j = 0
imax = np.size(clouds,1)
jmax = np.size(clouds,0)
while i ¡= (imax-1):
while j ¡= (jmax-1):
m = clouds[j][i]
if m == 0 or m == 2 or m == -1:
if i == (imax - 1):
print(’line’, j, ’of clouds done’)
if j == (jmax - 1):
j += 1
i += 1
else:
i = 0
j += 1
else:
i += 1
```

```
elif m == 1:
if i == 0:
miplus = clouds[j][(i+1)]
if j == 0:
mjplus = clouds[(j+1)][i]
mjiplus = clouds[(j+1)][(i+1)]
if miplus == 0 or mjplus == 0 or mjiplus == 0:
clouds[j][i] = 2
i += 1
else:
i +=1
elif j == (jmax-1):
mjminus = clouds[(j-1)][i]
mjminusiplus = clouds[(j-1)][(i+1)]
if miplus == 0 or mjminus == 0 or mjminusiplus == 0:
clouds[j][i] = 2
i += 1
else:
i += 1
else:
mjplus = clouds[(j+1)][i]
mjminus = clouds[(j-1)][i]
mjminusiplus = clouds[(j-1)][(i+1)]
mjiplus = clouds[(j+1)][(i+1)]
if miplus == 0 or mjplus == 0 or mjiplus == 0 or mjminusiplus == 0 or mjminus == 0:
clouds[j][i] = 2
i += 1
else:
i +=1
elif i == (imax-1):
miminus = clouds[j][(i-1)]
if j == 0:
mjplus = clouds[(j+1)][i]
mjplusiminus = clouds[(j+1)][(i-1)]
if miminus == 0 or mjplus == 0 or mjplusiminus == 0:
clouds[j][i] = 2
i = 0
j += 1
else:
i = 0
j +=1
elif j == (jmax-1):
mjminus = clouds[(j-1)][(i)]
mjiminus = clouds[(j-1)][(i-1)]
if miminus == 0 or mjminus == 0 or mjiminus == 0:
clouds[j][i] = 2
j += 1
i += 1
else:
j += 1
i += 1
```

```
else:
mjplus = clouds[(j+1)][(i)]
mjminus = clouds[(j-1)][i]
mjiminus = clouds[(j-1)][(i-1)]
mjplusiminus = clouds[(j+1)][(i-1)]
if miminus == 0 or mjplus == 0 or mjiminus == 0 or mjplusiminus == 0 or mjminus == 0:
clouds[j][i] = 2
i = 0
j += 1
else:
i = 0
j +=1
else:
miminus = clouds[j][(i-1)]
miplus = clouds[j][(i+1)]
if j == 0:
mjplus = clouds[(j+1)][i]
mjplusiminus = clouds[(j+1)][(i-1)]
mjiplus = clouds[(j+1)][(i+1)]
if miminus == 0 or miplus == 0 or mjplus == 0 or mjplusiminus == 0 or mjiplus == 0:
clouds[j][i] = 2
i += 1
else:
i +=1
elif j == (jmax-1):
mjminus = clouds[(j-1)][i]
mjiminus = clouds[(j-1)][(i-1)]
mjminusiplus = clouds[(j-1)][(i+1)]
if miminus == 0 or miplus == 0 or mjminus == 0 or mjiminus == 0 or mjminusiplus == 0:
clouds[j][i] = 2
i += 1
else:
i += 1
else:
mjplus = clouds[(j+1)][i]
mjiplus = clouds[(j+1)][(i+1)]
mjplusiminus = clouds[(j+1)][(i-1)]
mjminus = clouds[(j-1)][i]
mjiminus = clouds[(j-1)][(i-1)]
mjminusiplus = clouds[(j-1)][(i+1)]
if miminus == 0 or miplus == 0 or mjplus == 0 or mjiplus == 0 or mjplusiminus == 0 or mjminus == 0 or mjiminus == 0 or mjminusiplus == 0:
clouds[j][i] = 2
i += 1
else:
i +=1
```

B.2 Shadow Propogation

```
i=0
    j=0
    #looping over matrices
    while i ¡= (imax-1):
    while j ¡= (jmax-1):
    exist = clouds[j][i]
    h = height[j][i]
    #determining cloud existence
    if exist == 1 and h ¿ 0:
    #importing latsat, longsat, zenithsat, azimuthalsat and h = cloud top height for pixel
    latsat = latitude[j][i]
    longsat = longitude[j][i]
    zensat = zenithsat[j][i]
    azisat = azimuthsat[j][i]
    #Converting to ECEF coordinates
    pointA = wgs84.GeoPoint(latitude = latsat, longitude = longsat, z = 0, degrees = True)
    p _A = pointA.to _ecef _vector()
    cloudgroundpos = p _A.pvector.ravel().tolist()
    #Working out relevant geometric quantities
    sinlatsat = mt.sin(latsat)
    sinlongsat = mt.sin(longsat)
    coslatsat = mt.cos(latsat)
    coslongsat = mt.cos(longsat)
    sinzensat = mt.sin(zensat)
    coszensat = mt.cos(zensat)
    sinazisat = mt.sin(azisat)
    cosazisat = mt.cos(azisat)
    #Working out equation for satellite line of sight in ECEF coordinates ux = -sinzensat*(sinlongsat*sinazisat+
    uy = -sinzensat*(sinlongsat*sinlatsat*cosazisat-coslongsat*sinazisat)+sinlongsat*coslatsat*coszensat
    uz = sinlatsat*coszensat+cosazisat*coslatsat*sinzensat
    x0 = cloudgroundpos[0]
    y0 = cloudgroundpos[1]
    z0 = cloudgroundpos[2]
    #Equation for Earth's surface + height for cloud is equated to equation for satellite line of
sight and solve for intersection
    alpha = 1
    beta = 2*(ux*x0+uy*y0+uz*z0)
    gamma = -(2*Erad*h + h*h)
    #bprime = b + h
    #aprime = a + h
    #alpha = ux*ux*bprime*bprime + uy*uy*bprime*bprime + uz*uz*aprime*aprime
    #beta = 2*(ux*x0*bprime*bprime + uy*y0*bprime*bprime + uz*z0*aprime*aprime)
    #gamma = z0*z0*aprime*aprime + y0*y0*bprime*bprime + x0*x0*bprime*bprime - aprime*aprime*bprin
    radical = beta*beta-4*alpha*gamma
    sqrt = mt.sqrt(radical)
    d = (sqrt-beta)/(2*alpha)
    uxprime = ux*d
    uyprime = uy*d
    uzprime = uz*d
    u = [uxprime,uyprime,uzprime]
```

```
cloudpos = cloudgroundpos + u
#repeat for light from sun and earths surface to propogate shadow
x = cloudpos[0]
y = cloudpos[1]
z = cloudpos[2]
r = mt.sqrt(x*x+y*y+z*z)
longsun = mt.atan2(y,x)
zoverr = z/r
latsun = mt.asin(zoverr)
iprime = 0
jprime = 0
deltamin = 1000
while iprime ¡= (imax-1):
while jprime ¡= (jmax-1):
deltalat = abs((latsun-latitude[jprime][iprime]))
deltalong = abs((longsun-longitude[jprime][iprime]))
delta = deltalong + deltalat
m = min(deltamin, delta)
if m == delta:
deltamin = delta
isun = iprime
jsun = jprime
if iprime == (imax - 1):
if jprime == (jmax - 1):
jprime += 1
iprime += 1
else:
iprime = 0
jprime += 1
else:
iprime += 1
zensun = zenithsun[jsun][isun]
azisun = azimuthsun[jsun][isun]
#import zensun, azimuthalsun
sinlatsun = mt.sin(latsun)
sinlongsun = mt.sin(longsun)
coslatsun = mt.cos(latsun)
coslongsun = mt.cos(longsun)
sinzensun = mt.sin(zensun)
coszensun = mt.cos(zensun)
sinazisun = mt.sin(azisun)
cosazisun = mt.cos(azisun)
uxsun = sinzensun*(sinlongsun*sinazisun + coslongsun*sinlatsun*cosazisun) - coszensun*coslongsun*coslat
uysun = sinzensun*(sinlongsun*sinlatsun*cosazisun - coslongsun*sinazisun) - sinlongsun*coslatsun*coszensu
uzsun = -sinazisun*coszensun - cosazisun*coslatsun*sinzensun
#circle form
alphasun = 1
betasun = 2*(uxsun*x+uysun*y+uzsun*z)
gammasun = -Erad*Erad
#ellipse form
#alphasun = uxsun*uxsun*b*b + uysun*uysun*b*b + uzsun*uzsun*a*a
```

```
#betasun = 2*(uxsun*x*b*b + uysun*y*b*b + uzsun*z*a*a)
#gammasun = z*z*a*a + y*y*b*b + x*x*b*b - a*a*b*b
radicalsun = betasun*betasun-4*alphasun*gammasun
sqrtsun = mt.sqrt(radicalsun)
dshadow = (-betasun-sqrtsun)/(2*alphasun)
uxsunprime = uxsun*dshadow
uysunprime = uysun*dshadow
uzsunprime = uzsun*dshadow
usun = [uxsunprime,uysunprime,uzsunprime]
shadowpos = cloudpos + usun
xshadow = shadowpos[0]
yshadow = shadowpos[1]
zshadow = shadowpos[2]
rshadow = mt.sqrt(xshadow*xshadow + yshadow*yshadow + zshadow*zshadow)
zoverrshadow = zshadow/rshadow
latshadow = mt.asin(zoverrshadow)
longshadow = mt.atan2(yshadow,xshadow)
latshadow = mt.degrees(latshadow)
longshadow = mt.degrees(longshadow)
iprime = 0
jprime = 0
deltamin = 1000
#find best match location for shadow
while iprime ¡= (imax-1):
while jprime ¡= (jmax-1):
deltalat = abs((latshadow-latitude[jprime][iprime]))
deltalong = abs((longshadow-longitude[jprime][iprime]))
delta = deltalong + deltalat
if delta ¡ deltamin:
deltamin = delta
ishadow = iprime
jshadow = jprime
if iprime == (imax - 1) and jprime == (jmax - 1):
jprime += 1
iprime += 1
elif jprime == (jmax-1) and iprime != (imax-1):
jprime = 0
iprime += 1
else:
jprime += 1
else:
ishadow = ishadow
jshadow = jshadow
if iprime == (imax - 1) and jprime == (jmax - 1):
jprime += 1
iprime += 1
elif jprime == (jmax-1) and iprime != (imax-1):
jprime = 0
iprime += 1
else:
jprime += 1
```

```
shadows[jshadow][ishadow] = clouds[j][i]
if i == (imax - 1):
print(’line’, j, ’of shadows done’)
if j == (jmax - 1):
j += 1
i += 1
else:
i = 0
j += 1
else:
i += 1
else:
if i == (imax - 1):
print(’line’, j, ’of shadows done’)
if j == (jmax - 1):
j += 1
i += 1
else:
i = 0
j += 1
else:
i += 1
# gets rid of overlap
i = 0
j = 0
shadowsprime = np.zeros_like(clouds)
while i ¡= (imax-1):
while j ¡= (jmax-1):
fraction = clouds[j][i]
if fraction == 1:
shadowsprime[j][i] = 0
elif i == 0 or j == 0 or j == (jmax-1) or i == (imax-1):
shadowsprime[j][i] = 0
else:
shadowsprime[j][i] = shadows[j][i]
if i == (imax - 1):
print(’line’, j, ’of shadows done’)
if j == (jmax - 1):
j += 1
i += 1
else:
i = 0
j += 1
else:
i += 1
```