University of Southampton

Faculty of Engineering, Science and Mathematics

School of Electronics and Computer Science

"Can a single cloud spoil the view?": Modelling the effect of an isolated cumulus cloud on calculated surface solar irradiance

by

Robin Wilson

October 19, 2011

A dissertation submitted in partial fulfilment of the degree of
MSc Complex Systems Simulation
by examination and dissertation

Abstract

It is important to perform atmospheric correction on remotely sensed data acquired from satellite or airborne sensors before the data are used. One method of performing this correction is to use a Radiative Transfer Model (RTM) which models the physics of light passing through the atmosphere, producing output which allows this atmosphere effect to be reversed. All RTMs which are currently used for atmospheric correction assume a horizontally-homogeneous atmosphere, which is often not the case. In the UK, and other mid-latitude areas, satellite images frequently have isolated clouds in them, as it is almost impossible to obtain entirely cloud-free images. Atmospheric correction is then performed on these images using RTMs which are not capable of modelling these isolated clouds, and this may have an effect on the quality of the correction.

To determine the magnitude of this effect, a new RTM called *RTWRTM* is developed. This model allows the efficient simulation of a horizontally- and vertically-heterogeneous atmosphere using a Monte Carlo ray-tracing approach. The paths of a large number of rays are traced from the sun to the sensor through a two-dimensional grid. The rays undergo absorption and scattering as they pass through the atmosphere. Absorption effects are calculated based on coefficients from the SPCTRAL2 RTM (Bird and Riordan, 1984) and scattering effects are implemented probabilistically based on parameters from the OPAC database (Hess et al., 1998). The model is designed as an exploratory model (Murray, 2003), and therefore is not expected to produce quantitatively accurate results. Qualitatively, however, the model is found to have good skill.

Model runs show that the presence of an isolated cloud does have an effect on surface irradiance, and that this effect is still present even when the cloud is a long way from the sensor. Errors in calculated surface irradiance have been converted into their effect on typical remote sensing outputs. The average difference caused by the presence of a cloud (averaging over a number of cloud positions) is a 4% change in the Normalised Difference Vegetation Index (NDVI), and a $2.8\mathrm{g\,m^{-2}}$ change in net primary productivity (the amount of carbon fixed by plants into biomass), although maximum observed changes are higher (5% and $9\mathrm{g\,m^{-2}}$ respectively). Although these difference may seem low, they can combine over the area of a Landsat image to cause an underestimation of total net primary productivity by over 80,000 tonnes of carbon.

Acknowledgements

A great debt of gratitude is due to Olivia Wilson for a number of particularly useful discussions during the development of the model, and for marrying me during the course of this project (much as that caused a delay in submitting the project).

Ted Milton has supported my academic endeavours very enthusiastically for many years, and without him I would not have been in the position to even attempt to produce this report. His help during the model development was also much appreciated, and he helped me keep the project on track through all of the problems that I encountered.

Jo Nield, although not officially involved with me in any way, has been a constant source of advice on modelling techniques, and the use of her whiteboard has been much appreciated. I would not have had the ability to produce this model if I had not been introduced to the field of simple, cellular-automaton-based modelling in her undergraduate course *Aeolian processes and geomorphic modelling*.

Thanks are also due to my Dad and Grandad for proof-reading this document, and to Rob Hunt who provided help with some of the mathematics.

Contents

List of Figures

1 Introduction and Motivation

1.1 Atmospheric Correction in Remote Sensing

Remotely sensed data can be processed to provide information about many attributes of the Earth's surface: from carbon fluxes to urban sprawl, and has become particularly important for providing data on environmental change. Data are provided from satellite and airborne sensors in the form of *Digital Numbers*, dimensionless values which are simply the raw data collected by the CCD arrays in the sensors. Such data must be processed to provide useful data (such as carbon fluxes in tonnes of carbon, or water depths in metres) by conversion to physical units and correction for the effect of the atmosphere.

The effect of the atmosphere must be corrected for because:

- The composition of the atmosphere varies over time and space, so data from different locations or different times will have to be corrected for the effect of the atmosphere if they are to be compared.
- To compare data from satellites to data collected on the ground, the former must be processed to create 'ground-equivalent' data by removing the effect of the atmosphere.

The conversion to physical units is fairly easy, but correcting for the effect of the atmosphere is more complicated. Methods applied to *atmospherically correct* satellite images can be split into two types:

- **Empirical methods** require contemporaneous measurements on the ground and from the satellite/airborne sensor for certain locations in the image. The relationship between the satellite and ground measurements for these locations for each band of the image is then established, and this relationship is used to correct the entire image.
- **Physically-based methods** require a model of the passage of light through the atmosphere, known as a *Radiative Transfer Model (RTM)*. This model is parametrised using atmospheric data from the time of the data acquisition and used to model the passage of light from the sun, through the atmosphere to the surface, and then on the reflected path from the surface to the sensor. This model can then be inverted to allow the measured data to be converted to 'ground-equivalent' data.

The use of RTMs for atmospheric correction has significantly increased during recent years, following increases in computing power, and the increased availability of RTMs with relatively simple Graphical User Interfaces. However, questions must be asked about the accuracy of the results gained from such applications of RTMs.

1.2 The problem

One possible issue with the results of atmospheric correction using RTMs particularly affects users working with data from mid-latitude areas such as the UK. It is generally preferable to acquire remote-sensing data when there is an entirely clear sky, as then no areas of the image are obscured by cloud cover. However, in the mid-latitudes it is very rare to have an entirely clear sky, and therefore data acquisition is often undertaken on days when the sky is clear apart from one or two small *cumulus humilis* (commonly known as *fair-weather cumulus*) clouds.

When these data are processed the areas with cloud are normally masked out (so they do not appear in any final processed data product, for example a land cover map) but a RTM is used, as normal, to correct the data. This violates an assumption that nearly all operationally-used RTMs have: they assume that the sky is completely clear.

Violating this assumption may lead to the RTM producing inaccurate results, but these inaccuracies may be so small as to be insignificant. This project aims to investigate this by attempting to model the effect that small areas of cloud within a clear sky have on the intensities of the light spectrum reaching the ground. Thus, it aims to answer the question:

How much does a single cloud within a satellite image affect the surface irradiance across the image?

2 Background and Literature Review

2.1 Radiative Transfer in the atmosphere

Radiative Transfer Models attempt to model the passage of light through the Earth's atmosphere by applying mathematics based on the laws of physics. This section will explain the major processes which act upon light passing through the atmosphere and how these processes can be mathematically modelled.

2.1.1 General Overview

This discussion ignores the changes that happen to the light that leaves the sun before it reaches the top of the Earth's atmosphere (approximately 100km above the Earth's surface), and starts with light entering the top of the atmosphere. There are a number of good measurements of the intensity of sunlight across the visible and infrared spectrum at the top of the atmosphere (TOA; for example, the data from Wehrli (1985) shown in Figure 1). At this point the light from the sun approximates the standard black-body radiation curve very well, and the peak wavelength (approximately 450nm) can be calculated by Wien's law.

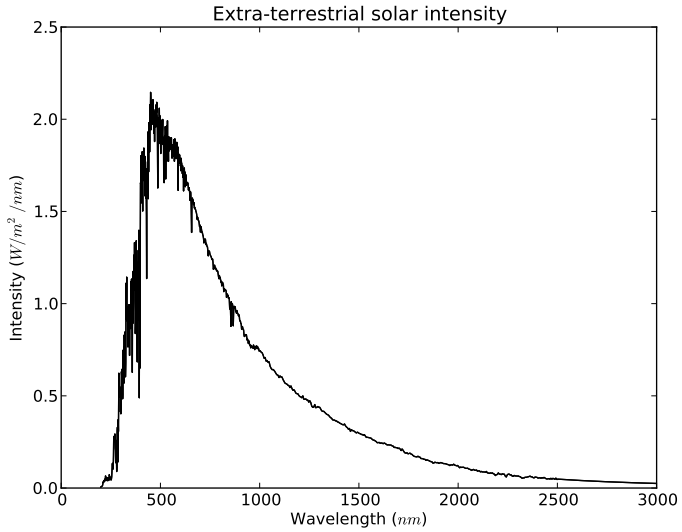


Figure 1 – Extra-terrestrial solar intensity. Data from the WRC reference spectrum (Wehrli, 1985).

As light passes through the atmosphere it interacts with a number of atmospheric constituents. These interactions principally take two forms:

- **Absorption:** This is a reduction of the intensity of the light, normally caused by a collision with a molecule. The magnitude of the reduction is usually dependent on the wavelength, and the principal atmospheric absorbers are atmospheric gases. The main gaseous constituents of the atmosphere are listed in Table 1. The proportions of all except water vapour stay fairly constant over time and space.
- **Scattering:** This is a change of direction of the light, normally caused by a collision with atmospheric molecules (such as those of the gases that make up the atmosphere) or particles suspended in the atmosphere (known as *aerosols*). These aerosols include dust, pollen, salt and soot and are unevenly distributed in the atmosphere. Light which reaches the sensor after undergoing scattering is referred to as *diffuse irradiance*, as opposed to light which has travelled in a direct beam from the sun, which is referred to as *direct irradiance*. More details about direct and diffuse irradiance, and the sky radiance distribution that results from scattering is available in Appendix C.

Importantly, the amount of absorption and scattering that occurs to the light depends on the wavelength of the light. Thus, some wavelengths of light are significantly affected by their passage through the atmosphere, and some wavelengths are barely affected at all.

Table 1 – Composition of a dry atmosphere (plus water vapour) by percentage volume

Nitrogen	78%
Oxygen	21%
Argon	0.9%
Carbon Dioxide	0.04%
Other gases	0.06%
Water vapour	∼40%

2.1.2 Absorption

Absorption occurs when photons collide with molecules in the atmosphere and transfer some of their energy to the molecule. These molecules absorb the energy and re-radiate it. This re-radiated energy can generally be ignored as it is very weak and located in the microwave or far-infrared regions of the spectrum.

The amount of absorption depends on the type of molecule (for example, water vapour is more absorbing than nitrogen). Individual collision events are rarely studied when investigating radiative transfer; instead the collective effect of a large number of molecules is calculated by relating the absorption to the concentration of absorbing molecules in a section of the atmosphere.

2.1.3 Scattering

Scattering of light in the atmosphere can be split into two types based on the mathematics which best model the effects of the scattering: Rayleigh scattering and Mie scattering.

Rayleigh scattering occurs when the diameter of the scattering particle is significantly smaller than the wavelength of the light. This is calculated based on scattering size, x, and Rayleigh scattering occurs where $x \ll 1$.

$$x = \frac{2\pi r}{\lambda}$$

In the atmosphere, this generally means that Rayleigh scattering occurs when the particles are gas molecules, such as those of gases listed in Table 1. The approximate amount of Rayleigh scattering for a given wavelength is described by the proportionality below

$$S_R \propto \frac{1}{\lambda^4}$$

Where S_R is the magnitude of Rayleigh scattering and λ is the wavelength of light. Thus, Rayleigh scattering occurs significantly more at low wavelengths (blue light) than high

wavelengths (red and infrared light).

Mie scattering occurs when the diameter of the scattering particle is roughly equal to the wavelength of the light. In the atmosphere, this means that all scattering by aerosols is Mie scattering. The mathematics behind Mie scattering is actually a full analytical solution of Maxwell's equations (albeit with a number of assumptions), so in many ways this has more physical basis than the Rayleigh approximation above. Mie scattering magnitudes are less dependent on wavelength than Rayleigh scattering magnitudes, but Mie scattering tends to affect longer wavelengths more than shorter wavelengths.

2.1.4 Absorption and Scattering combined

A number of particles in the atmosphere both absorb and scatter. For example, aerosols are known mainly for their scattering effect, but also cause absorption of energy. The balance between scattering and absorption is represented by the *single scattering albedo* which is defined as

$$SSA = \frac{ext_s}{ext_s + ext_a}$$

Where SSA is the single scattering albedo, ext_s is the extinction of light due to scattering and ext_a is the extinction of light due to absorption. The single scattering albedo varies between 0 and 1, with 1 signifying that the effect on light is entirely scattering, and 0 signifying that the effect on light is entirely absorption.

2.1.5 The fundamental equations of radiative transfer

The interactions which light undergoes as it passes through the atmosphere all result in decreasing the intensity of the light which reaches the surface. This reduction can be represented as a *transmittance*, T, which is the fraction of the incoming light which remains after the interactions have taken place. Transmittances can be calculated for all the different types of interactions which take place in the atmosphere, thus giving the following expression for the ground irradiance:

$$I_{ground} = I_{ext} T_1 T_2 T_3 \dots T_n$$

Where I_{ground} is the ground incident irradiance, I_{ext} is the top-of-atmosphere irradiance and $T_1 \dots T_n$ are the transmittances of the various types of interactions taken into account. This formula is the fundamental equation of radiative transfer, as this directly calculates the incident irradiance at ground level from top-of-atmosphere irradiance values. This calculation must be repeated for each wavelength as transmittance varies significantly with wavelength.

To use this equation we need to calculate the transmittances caused by various atmospheric interactions (for example, absorption by water vapour and scattering by aerosols). These must be between 0 and 1 and must be small when atmospheric extinction of light is high. Transmittances are generally calculated from an *optical depth* (τ) which is related to the transmittance (T) as

$$T = e^{-\tau}$$

Large optical depths produce small transmittances, which reflect a significant loss of the intensity of the light. Thus the optical depth (also known as the *optical thickness*) describes how opaque the atmosphere is to light.

To be able to use this equation it is necessary to calculate the optical depth caused by each type of atmospheric interaction (for example, absorption by ozone, scattering by aerosols). The methods by which these optical depths are calculated vary between RTMs, and are discussed in more detail in the sections below.

2.2 Types of Radiative Transfer Model

Many radiative transfer models have been written for different purposes. They can be split into a number of different types.

2.2.1 Method of calculation

- **Formula-based:** These models are based on the fundamental formulae underlying radiative transfer. Every value used in the radiative transfer is either specified by the user or calculated from basic theoretical formulae. This tends to increase the running-time of these models, but allows flexibility in the simulated parameters. For example, these models can often be run to produce outputs for any wavelength or wavelength range (some even include spectral response functions for each wavelength range).
  Examples: 6S (Vermote et al., 1997), SCIATRAN (Rozanov et al., 2005)
- **Lookup-table-based:** These models are based on tables of pre-calculated values for a number of important parameters (for example, the absorption co-efficient for ozone) which allow easy and quick calculation of final results. However, these values are normally only calculated for certain conditions: for example wavelengths may be limited to a choice from certain pre-programmed values (for example, 0.25µm, 0.30µm ...) and atmospheric conditions may be limited to a set of predefined profiles (for example, *mid-latitude maritime* or *desert*).
  Examples: SPCTRAL2 (Bird and Riordan, 1984)

2.2.2 Spatial variation in the atmosphere

- **No spatial variation:** These models treat the atmosphere as a homogeneous medium and simply use vertically- and horizontally-integrated values as parameters. For example, rather than modelling the water vapour content at varying three dimensional locations throughout the atmosphere, these models treat the water vapour content as a single value for the entire atmosphere. Although the atmosphere is treated homogeneously, provision is normally made for modelling differing path lengths for non-zenith-viewing sensors using a simple correction-factor approach.
  Examples: SPCTRAL2 (Bird and Riordan, 1984)
- **Vertical variation:** These models treat the atmosphere as horizontally homogeneous, but take into account vertical variation in atmospheric parameters. For example, these models will model water vapour content at various levels in the atmosphere. Models vary in regards to which parameters they model with full vertical profiles and which they treat as vertically-integrated as above.
  Examples: 6S (Vermote et al., 1997), SCIATRAN (Rozanov et al., 2005)
- **Fully spatially-explicit:** These models do not assume atmospheric homogeneity in any direction. They model atmospheric parameters at a number of locations both horizontally and vertically. This increases the model running time, but allows far more detailed and realistic simulations.
  Examples: O'Hirok and Gautier (1998)

2.2.3 Purpose

- **General atmospheric simulation (accurate):** These models are designed to simulate accurately the radiative transfer of light for a variety of purposes, including sensor design and calibration and the design of solar energy systems. They are designed to produce accurate results in physical units, and as such are used as *predictive models* (as defined by Murray, 2003).
  Examples: MODTRAN (Berk et al., 1999), SCIATRAN (Rozanov et al., 2005), 6S (Vermote et al., 1997)
- **General atmospheric simulation (estimates):** These models are designed to simulate the atmosphere for a variety of purposes, but are not designed to give physically-accurate results. Because of this, they are mainly used as *exploratory models* (as defined by Murray, 2003), to investigate the processes at work in the atmosphere rather than to provide accurate data.
  Examples: SPCTRAL2 (Bird and Riordan, 1984)
- **Atmospheric correction of remotely-sensed data:** These models are specifically designed to correct remotely-sensed (for example, from satellite or airborne sensors) data. They are often based on other radiative transfer models, and have the ability

to use the results from these models to correct entire images.
Examples: ATCOR (Richter, 2007)

2.3 A simple RTM: SPCTRAL2

SPCTRAL2 (Bird and Riordan, 1984) is a simple clear-sky RTM originally developed in the 1980s and designed to give quick results on the range of personal computers available at the time. A mark of its simplicity is that it is available in three versions written in C, Fortran and a Microsoft Excel spreadsheet! It uses a table of pre-calculated values (including top of atmosphere radiance and various absorption coefficients) and the fundamental equations of radiative transfer to calculate incident radiation at the ground in any of 122 specified wavelengths (ranging from 0.3-4.0µm).

The specific equation used for solving the radiative transfer problem is

$$I_\lambda = X_\lambda \; D \; T_{R\lambda} \; T_{A\lambda} \; T_{W\lambda} \; T_{O\lambda} \; T_{U\lambda}$$

where X_λ is the extra-terrestrial (top of atmosphere) irradiance, D is the earth-sun distance correction factor and $T_{R\lambda}$, $T_{A\lambda}$, $T_{W\lambda}$, $T_{O\lambda}$ and $T_{U\lambda}$ are the transmittances due to Rayleigh scattering, aerosol absorption, water vapour absorption, ozone absorption and uniformly-mixed gas absorption respectively. All parameters used for calculating these transmittances are calculated using empirical relationships, such as that below for water vapour transmittance (from Leckner, 1978), where W is the precipitable water vapour in a vertical path, $a_{w\lambda}$ is the water vapour absorption coefficient (dependent on wavelength) and M is the relative air mass.

$$T_{W\lambda} = \exp\left[\frac{-0.2385 \; a_{w\lambda} W M}{(1 + 20.07 \; a_{w\lambda} W M)^{0.45}}\right]$$

These empirical relationships are calculated for the whole height of the atmosphere, as SPCTRAL2 does not take into account any vertical or horizontal variation in atmospheric constituents.

An empirical extension to allow SPCTRAL2 to model cloudy skies was created by Bird et al. (1987). Their approach took standard SPCTRAL2 results and applied empirically-derived corrections to them to approximate the irradiance under partly- or fully-cloudy skies. However, these corrections had no theoretical basis, and produced results of low accuracy for sky types other than those used for parametrisation.

2.4 A more complicated RTM: 6S

Second Simulation of the Satellite Signal in the Solar Spectrum (6S; Vermote et al., 1997) is a model derived from 5S (Tanré et al., 1990) which is specifically designed to simulate radiative transfer as part of the processing of satellite data. Thus, it includes simulation of both the incident radiation (travelling from the sun to the ground) and the reflected radiation (reflected from the ground and travelling to the satellite sensor). This introduces an extra layer of complexity, and requires parameterisation of the ground surface reflectance. 6S allows this parameterisation to be very simple (a single albedo value) or complicated (a user-defined bi-directional reflectance distribution function).

When calculating absorption 6S takes into account oxygen, ozone, water vapour, carbon dioxide, methane and nitrous oxide. These are all treated as constant and uniformly mixed in the atmosphere apart from water vapour and ozone which are modelled based on the time and location. All absorption is calculated using random exponential band models based on the HITRAN database (Rothman et al., 2005).

Scattering is calculated as the sum of Rayleigh scattering and aerosol scattering using the Successive Orders of Scattering approach, giving results accurate to 10^{-4} in reflectance units. The aerosol content of the atmosphere can be configured to one of a number of preset configurations (such as Continental, Maritime or Urban) or manually based on a number of components (dust, oceanic, water-soluble and soot). The downward radiation field is calculated for a range of angles, allowing the necessary inputs for a full BRDF simulation of ground reflectance.

6S is provided as a set of Fortran 77 routines which can be compiled to produce an executable which can be run in a batch script. The program takes data from an input file with configuration information and produces output to another file. Unfortunately, 6S only supports calculations using a single wavelength-angle combination at a time. For example, simulating radiative transfer for light from 0.3 to 0.7µm using sensor view angles from 0° to 45° would require running 6S many times and manually editing the input file between each run.

`Py6S` is a Python module written by the author designed to alleviate this problem. It allows 6S to be scripted using the full power of the Python programming language and makes simulating the conditions above very easy. It also provides the model outputs in a form which can be easily plotted and analysed in Python (using the `matplotlib` module), R or other statistical software. An example use of Py6S is shown in Listing 1. Further details are provided in Appendix A.

Listing 1 – Example Py6S script

```
# Import NumPy so that we can use the arange command below
import numpy as np
# Import the SixS class
from SixS import SixS
```

```
# Import the helper classes for specifying 6S parameters
from SixSParams import *

# Instantiate the class
model = SixS()

# Set a couple of simple numerical parameters
model.solar_z = 50
model.view_z = 23

# Set the atmospheric profile type.
# This uses the AtmosModel class which provides a
# list of pre-specified atmospheric profile types
# for ease of use.
model.atmos_profile = AtmosModel.MIDLATITUDE_SUMMER

# Set the aerosol profile in the same way
model.aero_profile = AeroModel.MARITIME

# For each wavelength in the range 0.4 to 0.8
# stepping by 0.001 (ie. 0.400, 0.401, 0.402...)
for wavelength in np.arange(0.4, 0.8, 0.001):
        # Set the model to produce output for this wavelength
        model.wavelength = wavelength
        # Run the model
        model.run()
        # Print one of the outputs
        print "Wavelength = %f \t Direct Irradiance = %f" \
                % (wavelength, model.outputs.irradiance)
```

2.5 A state-of-the-art RTM: SCIATRAN

SCIATRAN is an advanced radiative transfer model developed by a team at the University of Bremen (Rozanov et al., 2005; Institute of Remote Sensing, 2007). It was originally designed to simulate radiative transfer for remote sensing purposes, but is also useful for general atmospheric simulation. It is particularly powerful as the model can be inverted, so as well as determining the light reaching the surface given a parametrisation of atmospheric constituents, it can also estimate the atmospheric constituents given a surface irradiance measurement. This inversion ability has been used for a number of studies including conducting sensitivity analyses of satellite trace gas retrieval methods (Vidot et al., 2010) and developing methods to calculate ice cloud properties from satellite measurements (Kokhanovsky and Nauss, 2005).

SCIATRAN is the only one of the three models described in detail here to support the

modelling of clouds, although this ability is severely limited by the requirement of horizontal homogeneity. The model supports an effectively unlimited number of atmospheric layers, each of which can be clear sky or cloud with a wide range of parametrisable aerosol and cloud types. However, each of these layers must be homogeneous across the entire horizontal range of the simulated atmosphere, thus making it impossible to simulate small scattered clouds.

Diffuse irradiance is calculated in more detail than in the models above, using an angular function to return a full angular distribution of surface irradiance. This can then be used as input to a ground BRDF function, allowing complex ground surfaces to be accurately modelled. SCIATRAN is able to simulate a number of view angles, solar angles and wavelengths with only one run of the model, which removes the need for a wrapper script like Py6S.

3 The gap: a simple, spatially-explicit RTM

None of the range of models described above is able to model an atmosphere with horizontal variation. 6S and SCIATRAN can model many vertical layers in the atmosphere, and SCIATRAN can even model many different types of cloud for each of these layers, but the atmosphere must always be horizontally homogeneous. This limitation is due to the method of calculation which these models use. The scattering of light in the atmosphere is represented in these models by calculating a single value for the diffuse irradiance, and a single value for the direct irradiance. That is, all of the extra-terrestrial irradiance is treated as one body of light, and a certain proportion of this light is assumed to have scattered in each of a number of directions (based on various scattering approximation formulae).

To allow horizontal inhomogeneity in the modelled atmosphere the model must treat light as a set of individual rays, rather than as one body. The paths of these individual rays can be calculated, based on the properties of the specific parts of the atmosphere that the ray is passing through. Thus, each point in the path of each ray could be given a different set of properties, and as such horizontal differences in atmospheric properties can be easily modelled. These models are referred to as *ray-tracing models*.

Ray-tracing techniques were originally developed for use in computer graphics systems. For example, a ray-tracing approach is commonly used for generating photo-realistic 3D models to use as part of Computer Generated Imagery in films. In general rays are generated at the position of the viewer and the path they take is traced until they hit an object in the scene. The position of any light sources is also modelled, and rays from both the viewer and the light sources undergo a number of interactions (for example, reflection, refraction and absorption) before producing a final brightness and colour value at a point in the image. This approach is described in detail in Glassner (1989).

The application of ray-tracing techniques to radiative transfer modelling is not a new idea. For example, O'Hirok and Gautier (1998) developed a complex RTM based on ray tracing within a detailed three-dimensional simulated environment. This model is fully spatially explicit, is not based upon lookup tables, and produces results with a high accuracy. However, parametrising the model is difficult as there are many parameters which are difficult to calculate from real-world datasets. Thus, in this model is found a problem which is common in simulation modelling: accurate models tend to be difficult to parametrise (as they often require a large number of parameters, each of which must be accurately specified) and without accurate parametrisation the model does not produce accurate results. Furthermore, accurate models tend to be highly computationally intensive, which limits their use for exploratory modelling.

Given the above, there appears to be a gap in the literature for a model which is spatially explicit, and therefore able to model a horizontally heterogeneous atmosphere, but is also simple enough to be easily parametrised, quick to run and easy to use as an exploratory model. The rest of this paper discusses the development of a model to fill this gap, discusses and interprets the results from this model, and presents a number of proposals for further work.

4 Model design

4.1 Overview

A new model, called RTWRTM, has been designed to fill the gap described above. It is designed as a simple Monte Carlo ray-tracing model which aims to produce insights into the behaviour of radiative transfer in the atmosphere. As such it is an exploratory model (as defined by Murray, 2003), and is not expected to produce quantitatively accurate results for use in atmospheric correction procedures.

The model simulates rays of light passing through the atmosphere, which is represented by a two-dimensional grid of cells. Each cell contains parameters describing the properties of the part of the atmosphere which it represents, thus allowing the properties of the atmosphere to be varied horizontally and vertically. It should be noted, however, that this model is not fully spatially-explicit as the grid representing the atmosphere is not three-dimensional, so the modelled atmosphere can only be varied in two axes.

RTWRTM is based upon the SPCTRAL2 model (Bird and Riordan, 1984), and can be thought of as a ray-tracing version of that model. A number of the lookup tables used by SPCTRAL2 for calculations are also used by RTWRTM, and as such the wavelengths which can be used in RTWRTM are those used by SPCTRAL2: 122 wavelengths irregularly spaced between 0.2–3µm.

The model is run by simulating the passage of many rays of light from the sun (placed at a

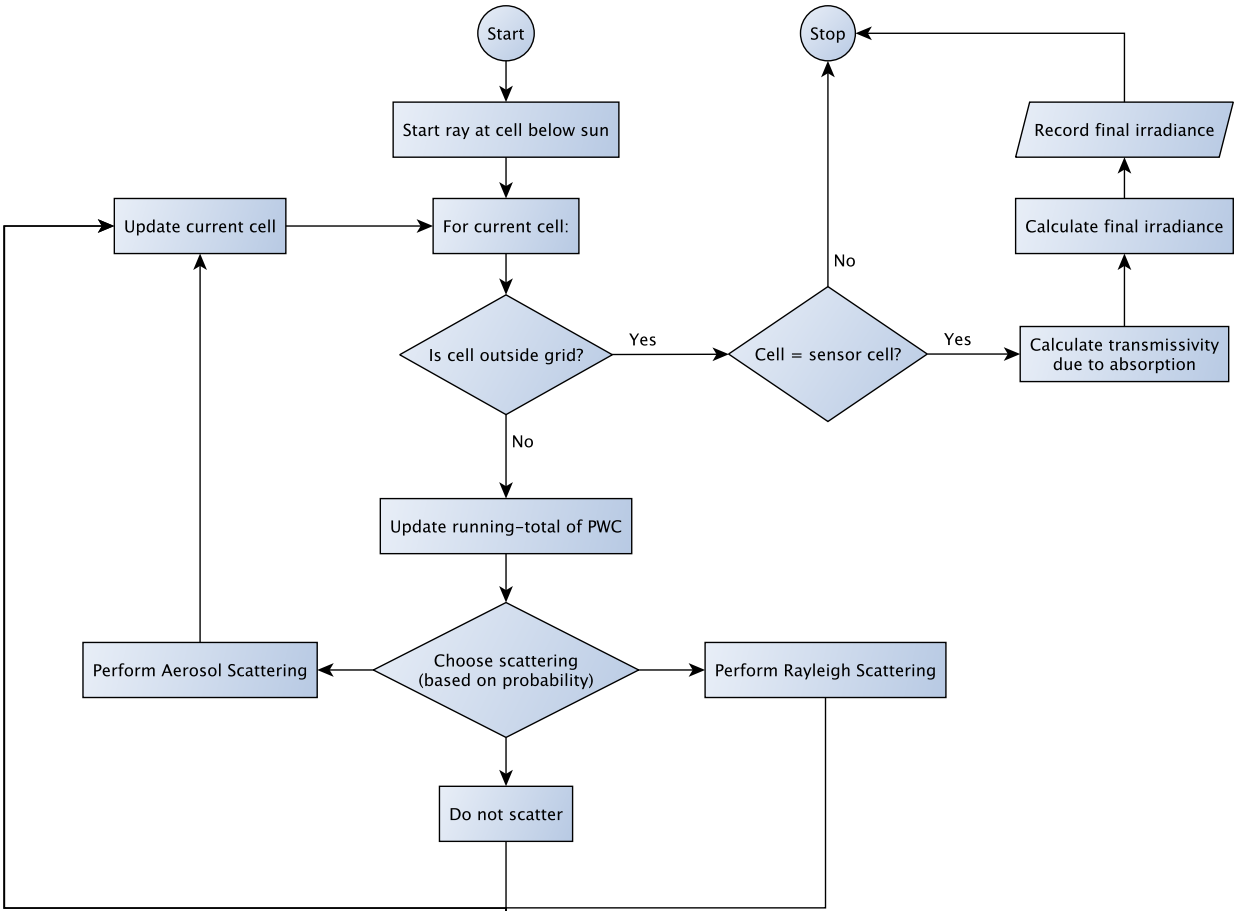


Figure 2 – Flowchart of the passage of one ray through the atmosphere

configurable location above the grid) to the sensor (placed at a configurable location below the grid). Each ray represents light of one wavelength, and starts at the sun with intensity equal to the extra-terrestrial spectrum for that wavelength (taken from the SPCTRAL2 data, from the same source as Figure 1). The intensity of this ray is reduced as it travels through cells on its path to the sensor, and the intensity upon reaching the sensor is recorded. These recorded intensities are averaged over the total number of rays and stored as the simulated ground incident radiation.

As the ray travels through the grid it is affected by scattering and absorption processes, further details of which are given below. A flowchart showing the passage of one ray through the atmosphere is shown in Figure 2.

4.2 Absorption

Absorption by three substances are taken into account in the model: ozone, water vapour and mixed gases. The latter is treated as a combination of the other gases in the atmosphere

because their distribution over time and space is uniform.

The effects of absorption are calculated for each ray of light after its entire passage through the atmosphere has been modelled. Thus the various absorption components must be summed throughout the passage of the ray and then used in various formulae once the ray has hit the sensor.

Extinction coefficients for each substance for each wavelength are used in the calculation of transmittance values, and these coefficients are taken from those used by SPCTRAL2, shown in Figure 3.

4.2.1 Absorption calculations

Water vapour absorption is calculated using the formula from Leckner (1978), which is

$$T_{w\lambda} = \exp\left(\frac{-0.2385 a_{w\lambda} W M}{(1 + 20.07 a_{w\lambda} W M)^{0.45}}\right)$$

where W is the precipitable water vapour (in cm) in a vertical path through the atmosphere, $a_{w\lambda}$ is the water vapour absorption coefficient (taken from SPCTRAL2) and M is the relative air mass. The parameterisation of W is described in the *Grid Parameterisation* section below and the value of M is described in the *Relative air mass calculation* section below.

Ozone absorption is calculated using the formula from Leckner (1978), which is

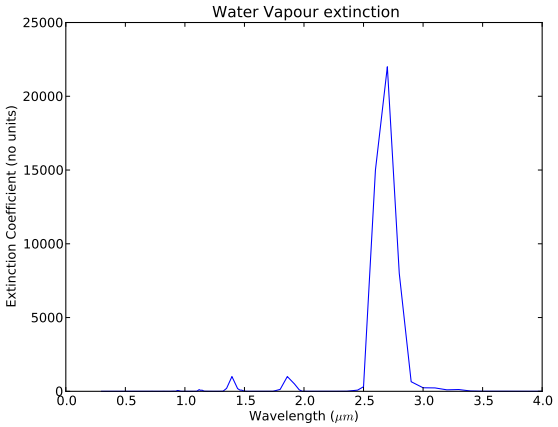
$$T_{O\lambda} = \exp(-a_{O\lambda}\, O_3\, M_O)$$

where $-aO\lambda$ is the ozone absorption coefficient (taken from SPCTRAL2), O_3 is the ozone amount and M_O is the ozone mass. Ozone amount is estimated using the method of Van Heuklon (1979), and ozone mass by the formula from Iqbal (1983), and these require inputs of latitude, longitude and date.

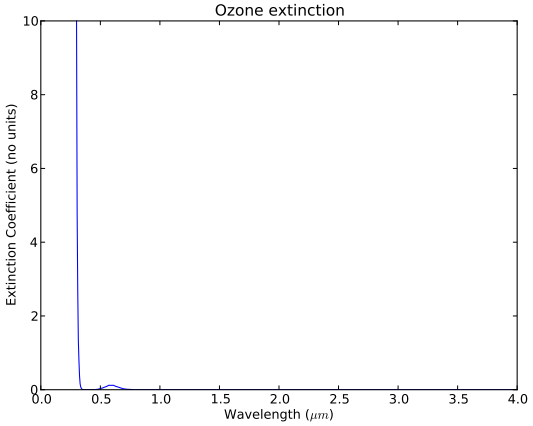
Mixed gas absorption is also calculated using the formula from Leckner (1978), which is

$$T_{U\lambda} = \exp\left[\frac{-1.41 a_{U\lambda} M}{(1 + 118.93 a_{U\lambda} M)^{0.45}}\right]$$

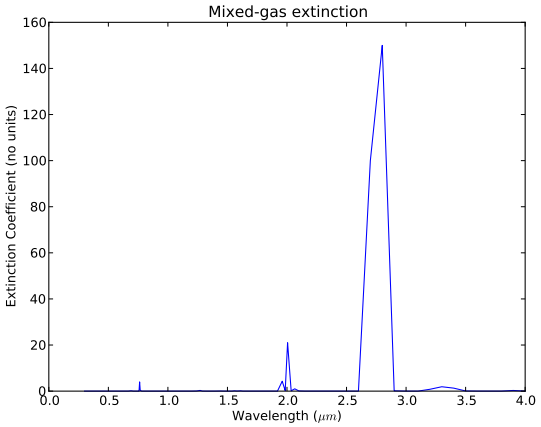
where $a_{U\lambda}$ is a combination of the absorption coefficient and the gaseous amount (taken from SPCTRAL2).



(a) Water vapour



(b) Ozone



(c) Uniformly mixed gas absorption

Figure 3 – Extinction coefficients used for absorption calculations (data from SPCTRAL2 datafiles). Note the differing vertical scales.

4.2.2 Relative air mass calculation

The formulae for calculating absorption due to water vapour and mixed gases require a value for M, the relative air mass. This value represents the distance through the atmosphere that the ray has travelled, and is calculated in SPCTRAL2 as

$$M = \left[\cos Z + 0.15(93.885 - Z)^{-1.253}\right]^{-1}$$

This gives a value which reflects the distance travelled by the ray when compared to a vertical path through the atmosphere, with a vertical path giving a value of 1.0, and a 20% longer path giving a value of 1.2. In RTWRTM M is calculated as the ratio of the number of cells the ray has passed through to the number of cells in a vertical path through the grid, giving very similar outputs.

4.3 Scattering

RTWRTM models the scattering of light probabilistically, using a number of parameters to determine both the likelihood of each ray being scattered at each cell, and the direction in which the ray should be scattered. Two types of scattering are considered: Rayleigh scattering and aerosol scattering.

Unless otherwise stated the raw data used to calculate the scattering probabilities are taken from the Optical Properties of Aerosols and Clouds (OPAC) database (Hess et al., 1998). This is a database containing a wide range of optical properties (including single-scattering albedo, phase functions and extinction coefficients) for the four fundamental aerosol types (soot, dust, water-soluble and oceanic) combined with a Fortran program which allows these components to be mixed, and the resulting aerosol parameters to be calculated. The optical properties that are wavelength-dependent are provided for selected wavelengths from 0.25–4µm, and these were interpolated to provide data for all 122 of the wavelengths used in RTWRTM.

OPAC is provided with a number of standard aerosol mixings, including maritime, urban, cumulus and stratus, many in two versions: polluted and clean. These default mixings have been used as the primary source of aerosol type parameterisations for RTWRTM.

More details on scattering are provided below, and a worked example showing the whole scattering process is provided in Appendix B.

4.3.1 Probabilities

At each cell that the ray passes through a probability of Rayleigh scattering and a probability of aerosol scattering is calculated. These probabilities are then halved and used to

decide what should take place, Rayleigh scattering, aerosol scattering or no scattering. For example, if the probabilities are as below:

$$p_R = 0.6, p_A = 0.3$$

Then the probabilities used to decide what type of scattering to perform are:

- Probability of Rayleigh scattering $= 0.5p_R = 0.3$
- Probability of aerosol scattering $= 0.5p_A = 0.15$
- Probability of no scattering $= 1 - 0.5(p_R + p_A) = 0.55$

The probability of Rayleigh scattering is calculated from the equation below

$$p_R = \frac{1}{150}\frac{1}{\lambda^4}$$

where λ is the wavelength in micrometres, and $\frac{1}{150}$ is a scaling factor used to produce realistic probabilities.

The probability of aerosol scattering is calculated as the product of the probability of scattering of the selected aerosol type-wavelength combination and the probability of scattering of the selected aerosol amount. The former is simply the single-scattering albedo for the selected aerosol type and wavelength, and the latter is calculated from an approximation of the likely probabilities of scattering for a range of aerosol amounts. Aerosol amounts are measured in number of particles per cm^3, thus allowing parameterisation from field data (for example data from Gregory et al., 1994; Yuwen et al., 1996). A graph showing these probabilities for aerosol amounts from 1–600 particles per cm^3 is shown in Figure 4.

Once the type of scattering has been chosen the direction of scattering is calculated using the phase functions below.

4.3.2 Phase functions

The *scattering angle* is defined as the difference between the incident and reflected angles for a given ray of light. Aerosol phase functions describe the likelihood of each scattering angle occurring. For aerosols, these phase functions are provided by OPAC for selected angles between 0–180°. These were interpolated to give the phase function for every angle from 0–180° and replicated to produce data for the full circle by assuming that the phase function is symmetrical. For Rayleigh scattering, the phase function can be calculated as

$$P(\theta) = \frac{3}{4}(1 + cos^2\theta)$$

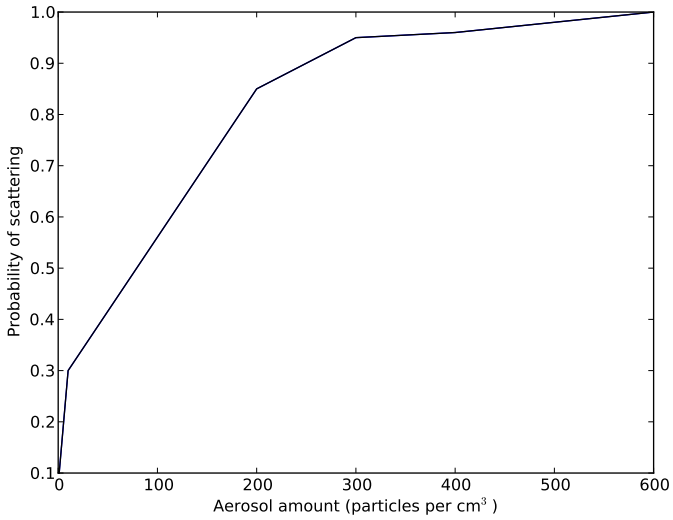


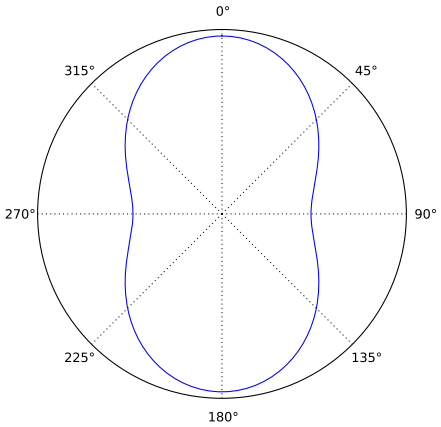
Figure 4 – Scattering probabilities for aerosol amounts from 1–600 particles per cm^3

To calculate which cell to scatter the ray to in RTWRTM the probabilities given by the phase function are used. However, the probabilities given in the phase functions are raw probabilities taken from real-world experiments, meaning that the probabilities will be much higher for a highly-reflective aerosol than for a non-reflective aerosol. As we have calculated the probability of scattering based on the aerosol type already, we need to normalise the probabilities in the phase function so that they are comparable between aerosol types. This is achieved by adjusting the function so that the integral from 0–360° is 1.0. These normalised functions are plotted in Figure 5.

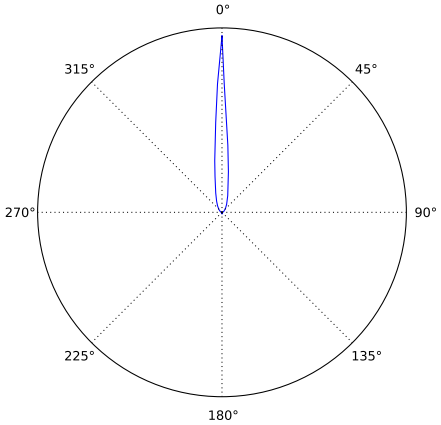
Each cell in the RTWRTM grid is surrounded by 8 cells, and we must decide which of these cells the ray should be scattered towards. Figure 6 show how the probability of scattering to each cell is calculated. The phase function is integrated using the ranges shown by the segments of the circle, producing a probability for each cell. However, it should remembered that the phase function defines the probability of each *scattering angle* occurring, and the scattering angle is the difference between the incident angle and the reflected angle. Thus, when calculating probabilities for each cell the probabilities must be shifted to ensure that 0° is at the location of the incident ray. A fully-worked e xample of this is given in Appendix B.

4.4 Clouds

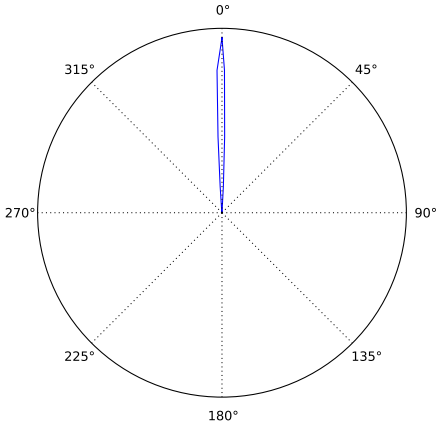
The model described above has the capability to include scattered clouds in the simulation process by varying the parametrisation of each cell in the grid, but the clouds used must be realistic to enable the model to produce valid results.



(a) Rayleigh scattering



(b) Maritime polluted aerosol scattering



(c) Cumulus continental polluted aerosol scattering

Figure 5 – Phase functions for Rayleigh scattering and two aerosol mixes. All data are normalised so that the integral is equal to 1.0

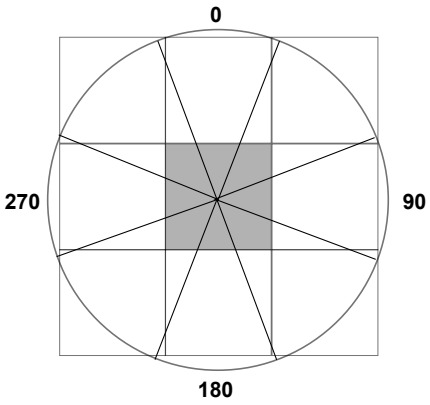


Figure 6 – Neighbourhood of the central cell (shaded) with integration regions overlaid

In particular, the simulated clouds must posses the following features:

- **Fractal appearance:** Clouds are well-known to be an example of a fractal shape found in nature (Mandelbrot, 1983). This is most apparent at the outside edge of the cloud, but the fractal appearance of edges within the cloud is also important. Indeed, Cahalan et al. (1994) found that the fractal structure within a stratocumulus cloud had more of an effect on the averaged albedo than the cloud fraction (the proportion of the total bounding area of the cloud which is fully covered by the cloud).
- **Internal variation:** Real-world clouds are inhomogeneous in all spatial dimensions, and change at rapid temporal scales. Highwood (2000) shows that this inhomogeneity has a large effect on the reflectance of the cloud.

It is clear that a simple rectangle of cloud placed in the grid will not be suitable. Furthermore, in future versions of the model it may be useful to allow the cloud to develop and change while moving across the grid, and this would be very difficult with a static cloud described as a simple geometric shape.

Cellular automata were originally developed in the 1940s by John von Neumann and have since been used to model a wide range of natural and man-made phenomena including urban land-use change (White et al., 1997), forest-fire spread (Hernández-Encinas et al., 2007) and sand dune evolution (Nield and Baas, 2008). They have proved particularly useful at simulating complex systems by using simple rules which interact to produce complex, self-organising results.

A cellular automaton model of clouds has been used to produce realistic clouds for use in RTWRTM. This model was originally developed by Nagel and Raschke (1992) to simulate clouds for scientific purposes, but was extended by Dobashi et al. (1998, 2000) for generating realistic-looking clouds for use in computer games. The model simulates the development of the cloud from a single seed, and so can be used to model the development of a cloud over time, an important ability for use with future versions of RTWRTM.

The model is based upon three binary 3D grids representing the key stages in cloud formation: humidity, activity and cloudiness. These are based upon a physical understanding of cloud formation, that clouds require humid air (represented by the humid grid), which is then uplifted (represented by the activity grid) and becomes cloud (represented by the cloud grid). To start, the humidity grid is initialised to a random state, and one or more cloud seeds are placed in the activity grid. The simulation then proceeds by following the rules listed below, executing each rule once per cell for every iteration of the model.

- A cell will become active if at least one of its neighbours is active (neighbours are taken to be all 26 surrounding cells)
- An active cell will become cloudy
- A cloudy site will stop being humid

In the current implementation the extended rules from Dobashi et al. (1998) have not been implemented, but if more dynamic clouds (including the formation and extinction of individual clouds) are required for a future version of RTWRTM these rules could be implemented very easily.

The output of the cellular automaton is a three-dimensional binary grid, containing 1 where the voxel is part of a cloud and 0 where it clear sky, as shown in Figure 7. As RTWRTM uses a two-dimensional grid whose axes represent horizontal distance along the ground (x) and vertical distance through the atmosphere (z), the three-dimensional cloud must be converted to a view of a the cloud from the size. This is done by taking the maximum along the y dimension of the cloud. The output which was used for the clouds in the model run below is shown in Figure 8.

Small, low-altitude clouds known as *cumulus humilis* are required for the model. No specific parametrisation of the cellular automaton model has been used to create these clouds; the model has simply been run for very few iterations (between 10 and 15 iterations) on a small grid to produce a cloud of the required size and complexity.

4.5 Grid parameterisation

Many of the calculations above require parameters which are stored for each cell in the grid. These parameters are the precipitable water content, the aerosol amount (in particles per cm^3) and the aerosol type (one of the types described above). At the start of the modelling process the grid must be initialised with a realistic parameterisation of the atmosphere. This parameterisation is handled separately for initialising the whole grid, and then adding clouds to the grid.

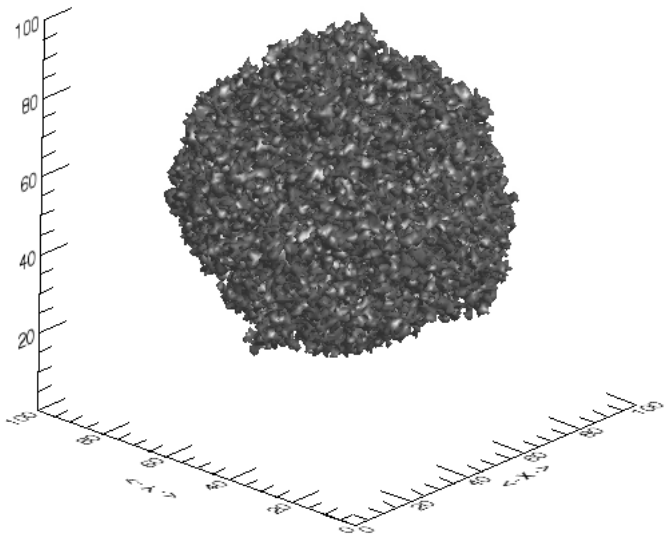


Figure 7 – An example of a 3D cloud created using the cellular automaton model, running on a 100x100x100 grid for 80 iterations. The whiter the pixel, the thicker the cloud at that point.

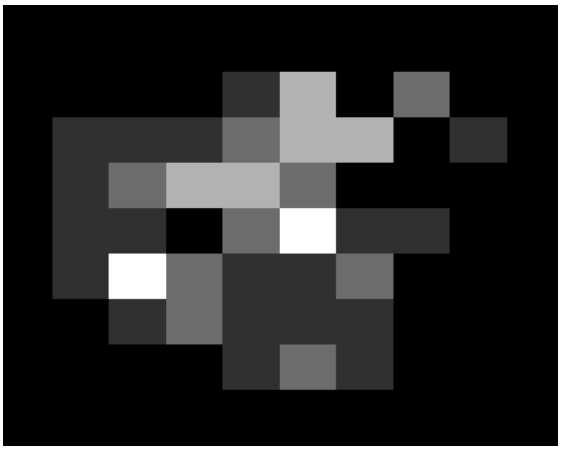


Figure 8 – The simulated cloud used in the model runs as part of this project, viewed from the side and ready to be incorporated into the model grid.

4.5.1 Whole grid

Water vapour is parametrised as the precipitable water content (PWC; in cm) for each cell. This is then summed for each cell the ray passes through, and the total is used to calculate water vapour absorption. It is clear from the literature that there is significantly more water vapour at the bottom of the atmosphere than at the top (Randel et al., 1996) and that the vertical distribution appears to be approximately exponential in shape (Ellingson et al., 1991).

Water vapour data at various altitudes acquired in a mid-latitude environment from Ellingson et al. (1991) were fitted to an exponential curve,

$$PWC = 12e^{-0.6h}$$

where h is the vertical height in the atmosphere.

When parametrising the grid the total vertical PWC should be configurable so the model can be set to model a range of environments. This is achieved by parametrising the grid so that the PWC is distributed according to the above equation and the vertical path sums to the specified PWC.

Aerosol type is easy to parametrise, as in the current implementation the entire grid is set to the maritime aerosol type. The urban aerosol type, although included in the model, is not currently used.

Aerosol amount is more difficult to parametrise. The literature contains the details of a number of real-world measurements of aerosol amount, many of which appear to contradict each other. In their atmospheric textbook, Grieken and Harrison (1998) suggest that aerosol amounts tend to range from around 1–10 particles per cm^3 in clear skies to 100–600 particles per cm^3 in clouds.

Gregory et al. (1994) found that in sub-arctic clear skies aerosol amounts could be described by a normal distribution with a mean of 3.1 and a standard deviation of 4.4. Although these values were measured in an arctic environment, they were obtained from air that was thought to have come from a mid-latitude environment. Unfortunately, no equivalent values measured in a mid-latitude environment are available. Thus, the grid is parametrised with each aerosol amount value taken as a random sample from the normal distribution described above.

4.5.2 Clouds

The cellular automaton model used to create realistic clouds described above produces a two-dimensional cloud array, with integer values showing the thickness of the cloud at each point. In a typical small cloud produced using this method the values range from 1–5.

These values must be used to set the grid parameters for the cells which are covered by cloud.

Setting the aerosol type is easy for the current implementation, as all modelled clouds are cumulus clouds, so the aerosol type is simply set to cumulus. Aerosol amounts are also parametrised very simply in the current implementation. As Grieken and Harrison (1998) suggests that clouds tend to have 100–600 particles per cm^3 and the integer values of the generated clouds tend to range from 1–5, the aerosol amount for each cell is simply set to $100c$ where c is the integer value in the generated cloud.

Data on average differences between total vertical precipitable water content in cloudy and non-cloudy skies are difficult to find. University of Wyoming Department of Atmospheric Sciences (2011) provide access to a database of measurements from weather balloons acquired at a large number of sites across the globe. These measurements include a calculated total precipitable water content for the path of the balloon (assumed to be vertical). Analysis of these PWC values suggests that they are increased by approximately 1cm when clouds are present. This approximation was used in the model by setting the PWC value of each cell in the cloud to $\frac{1}{h}$ where h is the height of the cloud (approximately 4-5 cells using the current cloud cellular automaton model).

4.6 Final irradiance calculation

Rays are immediately destroyed if they pass off the edge of the grid, except if their final cell is one of the three cells above the sensor (the cells labelled **L**, **R** and **V** in Figure 9). In this case the final intensity of the ray is calculated using the same formula as that used in SPCTRAL2,

$$I_\lambda = X_\lambda \; D \; T_{R\lambda} \; T_{A\lambda} \; T_{W\lambda} \; T_{O\lambda} \; T_{U\lambda}$$

where X_λ is the extra-terrestrial (top of atmosphere) irradiance, D is the earth-sun distance correction factor and $T_{R\lambda}$, $T_{A\lambda}$, $T_{W\lambda}$, $T_{O\lambda}$ and $T_{U\lambda}$ are the transmittances due to Rayleigh scattering, aerosol absorption, water vapour absorption, ozone absorption and uniformly-mixed gas absorption respectively. Note that this formula is the formula for direct irradiance from SPCTRAL2, and there is no explicit calculation of diffuse irradiance, because in ray-tracing models there is no differentiation between direct and diffuse irradiance as the path of each ray modelled in its entirety.

The final irradiance is then divided by the total number of rays which hit the sensor (to allow comparisons between simulations with different numbers of iterations) and stored separately depending on which cell the ray came from before it hit the sensor (also shown in Figure 9). Thus we obtain a very simple approximation of the angular distribution of irradiance at the sensor, split into irradiance from the left of the sensor, from above the sensor (vertically) and from the right of the sensor.

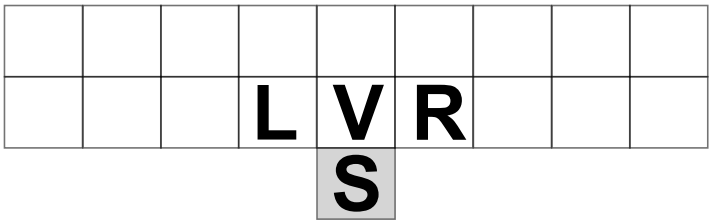


Figure 9 – The bottom of the RTWRTM grid showing the location of the sensor (**S**) and the left, right and vertical irradiance capturing cells (**L**, **R** and **V** respectively).

5 Model implementation

The model was implemented in the Interactive Data Language (IDL) programming environment (ITTVIS, 2011) in a form that can be run both on personal computers and on High Performance Computing systems such as the University of Southampton's Iridis cluster. Running a simulation of 50,000 multispectral rays (each consisting of a ray for each of 122 wavelengths) on a 20 x 20 grid takes approximately 15 minutes.

6 Results and Discussion

In this section outputs from the model described above will be shown and discussed with the aim of answering the key question raised in Section 1.2:

How much does a single cloud within a satellite image affect the surface irradiance across the image?

The model will first be assessed to ensure it is producing realistic results, then comparisons of simulations with and without clouds in various locations will be performed to answer this question.

6.1 Model skill

The *skill* of a model refers to the ability of a model to produce results which are comparable with results from other models, and which change as expected as model parameters are altered. To test the skill of this model, simulations were formed of a clear sky with two precipitable water contents: 0.5cm and 1.5cm. Simulations were also performed with 6S (using the Py6S wrapper) to allow comparisons of RTWRTM with an established model.

Figure 10 shows the results of two radiative transfer simulations using 6S and RTWRTM respectively. As described above, RTWRTM is designed as an exploratory model and therefore the results are not likely to be quantitatively comparable with those from other

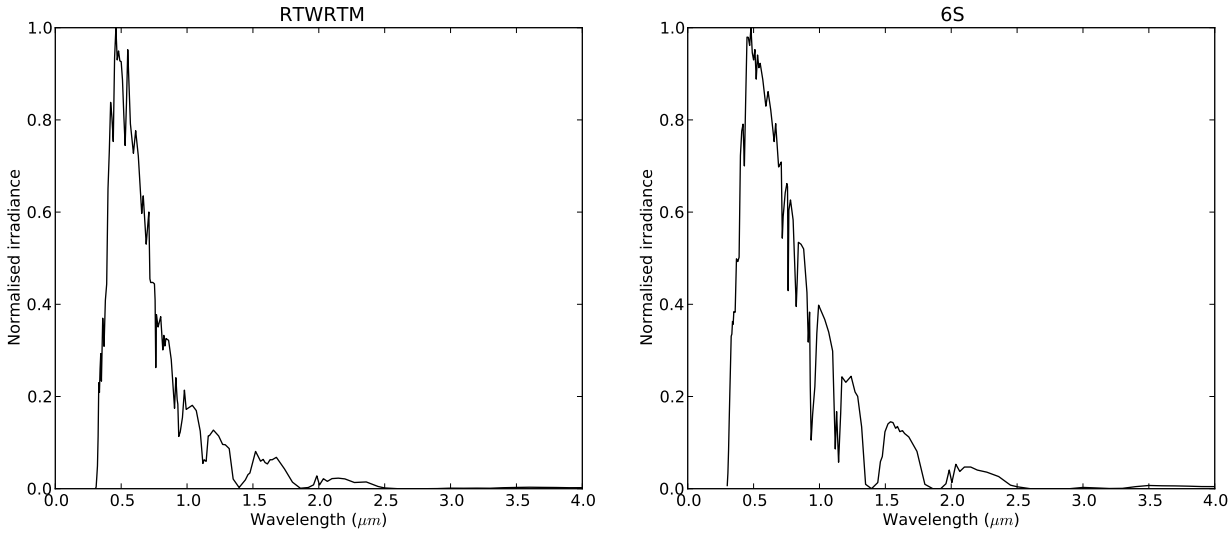


Figure 10 – Comparison of results from running RTWRTM and 6S with similar parameters. Both results have been normalised to the range 0–1 by dividing the result for each wavelength by the maximum result.

radiative transfer models. This is indeed the case, with a maximum value from the 6S output of 1700 compared to a maximum of 30 for the RTWRTM output. To enable qualitative comparison each result has been normalised to 0–1 by dividing the result for each wavelength by the maximum result.

This shows that RTWRTM produces a result which is fairly similar to that produced by 6S. The significant features of the irradiance spectrum at the Earth's surface are present such as water absorption bands at 1.2, 1.4 and 1.9µm and oxygen absorption bands at 0.63 and 0.76µm. The peak wavelength is also equal for both simulations, at 0.48µm.

However, in general the irradiance calculated by 6S seems to be higher than that calculated by RTWRTM, and thus the dips in the graph corresponding to the absorption bands are more pronounced. This is likely to be caused by differences in the parametrisation of the models. Although attempts were made to use the same parametrisation for each model to facilitate a proper comparison this was not possible as the models require different input parameters. For example, 6S bases most of its calculations on a single value of aerosol optical thickness at 550nm, whereas RTWRTM requires individual values for the aerosol amount and precipitable water content.

Assuming that the differences in outputs between RTWRTM and 6S are purely because of a difference in parametrisation, a further check of the model skill is to test the internal consistency of RTWRTM. This can be done by running two simulations with different parameters and checking that the change in the outputs is of the expected direction and magnitude. This was performed by running two simulations with precipitable water con-

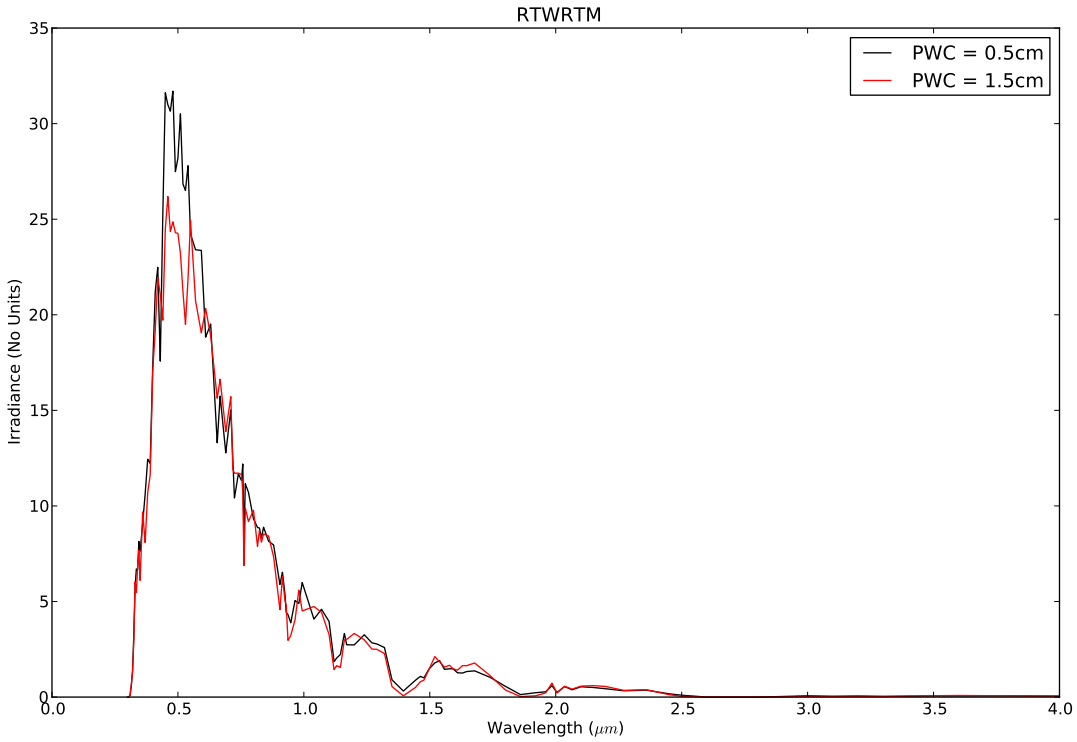


Figure 11 – Comparison of RTWRTM outputs with two different values of precipitable water content: 0.5cm and 1.5cm

tents of 0.5cm and 1.5cm respectively (see Figure 11).

The results are as expected, showing general attenuation of the light by the extra water vapour present in the atmosphere, and deeper absorption features at the water absorption wavelengths. Figure 12 shows four example paths that rays took from the sun to the sensor, showing the multiple scattering events which may occur.

From this and the comparison with 6S above, it can be concluded that the model has a good skill, and it produces results which are similar to established models and which respond to parameter changes in the expected manner.

6.2 Effect of a single cloud

6.2.1 Irradiance effects

The effect of a single cloud moving over a point is shown in Figure 13. The y-axis shows the location of the centre of the cloud and the x-axis shows the percentage difference in irradiance at the specified wavelength compared to the irradiance with no cloud. The grid

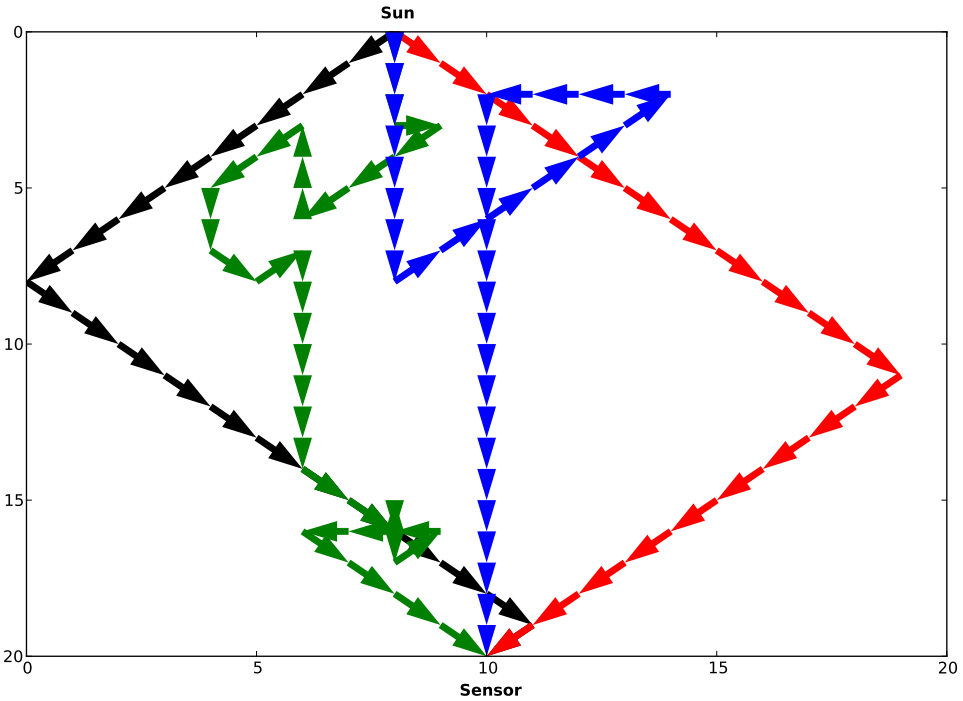


Figure 12 – Four example paths taken by light in the model from the sun to the sensor. Data taken from model runs. Note that some arrows are overlaid, and therefore not visible.

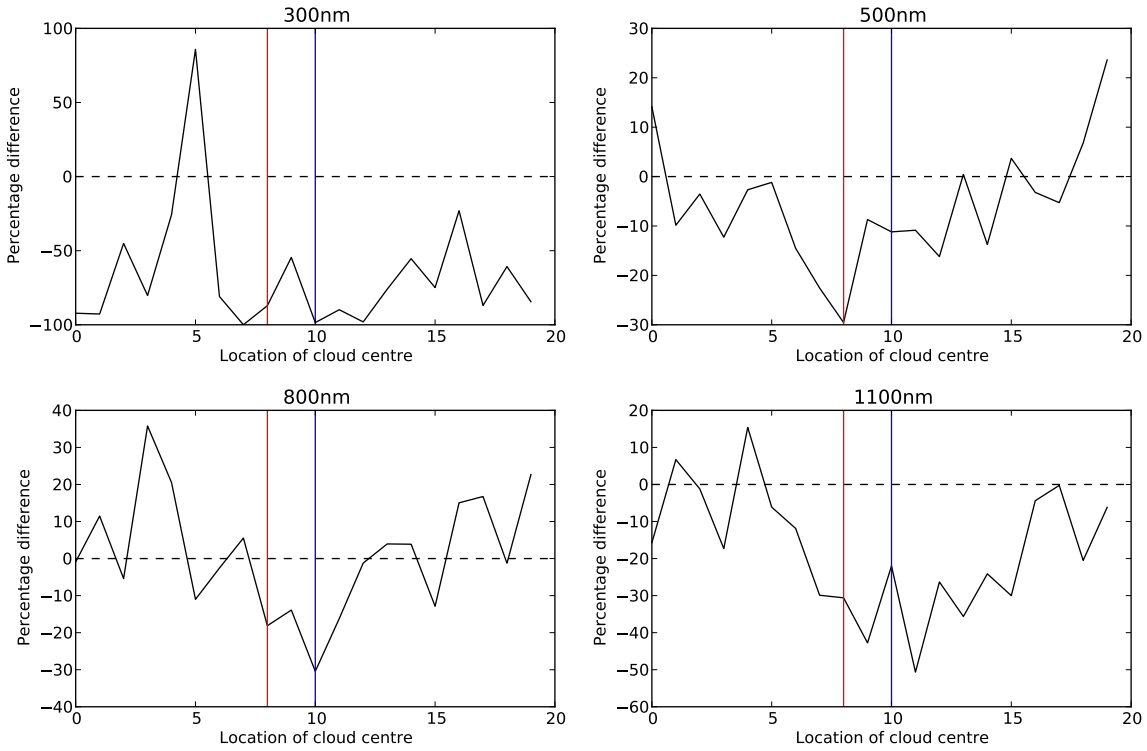


Figure 13 – Percentage difference of irradiance with a cloud passing over compared to irradiance with no cloud. Measured at four wavelengths (300nm, 500nm, 800nm and 1100nm). Sun location is indicated by the red line and sensor location by the blue line. Note that the x-axis value is the location of the centre of the cloud - the cloud extends by 2–3 grid cells either side of the centre.

was 20 x 20 cells in size, and as such simulations were run with the cloud centred at each x index from 0–20, with a constant height of 10 cells.

It is immediately apparent that there are significant differences between the irradiance with a cloud anywhere in the grid and the irradiance with no cloud. As expected, these differences are greatest at the shorter wavelengths, with a maximum absolute percentage difference of 98% at 300nm compared with 50% at 1100nm. This is because most scattering processes occur more at lower wavelengths, and the most significant effect of a cloud in an image is the increased scattering that it causes.

Generally, negative differences are found when the cloud is covering either the sun or the sensor, or both. This is particularly noticeable at 500nm where there is a 30% decrease in the irradiance when the cloud covers the sun. As the cloud is approximately five grid cells wide, the decreased irradiance values persist while parts of the cloud are covering the sun and/or sensor giving wider reductions in irradiance such as those seen between cloud centre locations of 7 and 12 in the 800nm plot.

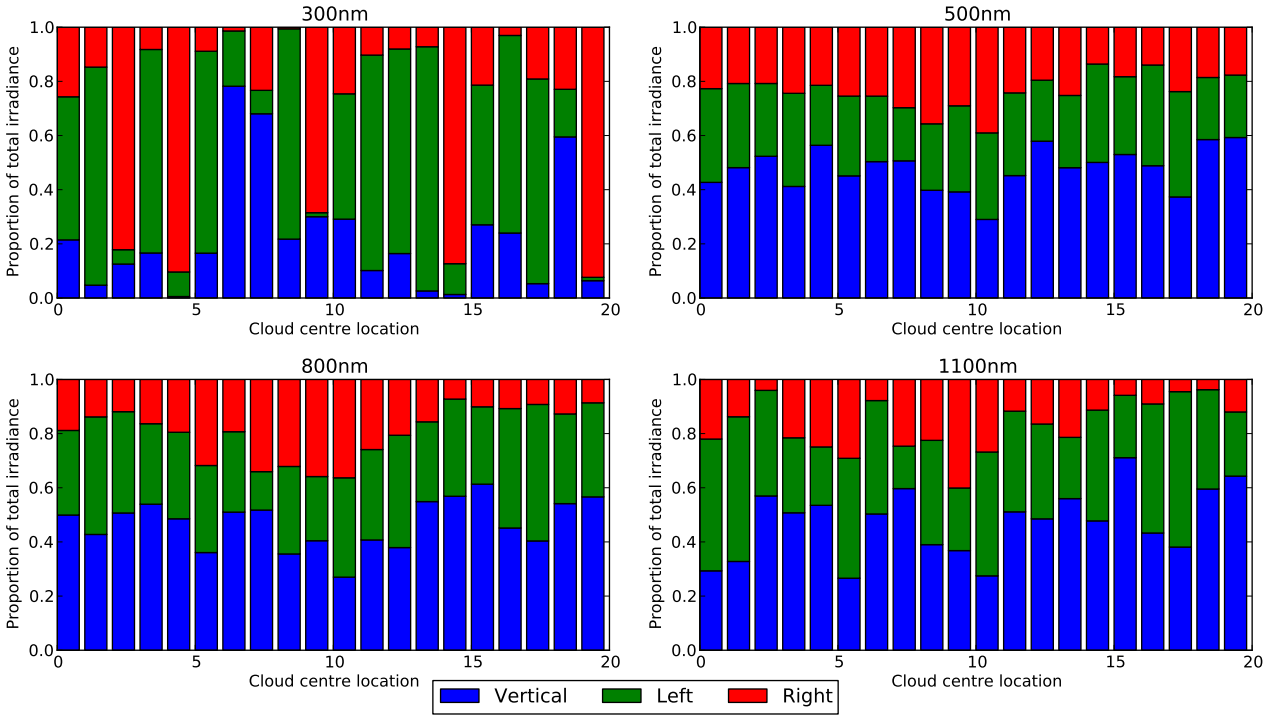


Figure 14 – Angular distribution irradiance received at the sensor, shown as the proportion of irradiance received from the left, right and vertical sides of the sensor.

A number of the plots also show a characteristic increase in irradiance values either side of the area of reduced irradiance, particularly noticeable in the 800nm plot. This is caused by the increase in scattered light reaching the sensor from a very close cloud.

Most importantly for answering the key question of this project, a cloud in the sky still has a large effect on the recorded irradiance even if the cloud is positioned a long way from the sensor. For example, even when the cloud is centred at $x = 20$ and the sensor is $x = 10$ there are still differences of up to 84%. This effect will be studied in more detail below.

6.2.2 Angular distribution of irradiance

The model records the incident light at the sensor in three separate categories depending on the angle from which it arrives, thus providing an approximation of the angular distribution of surface irradiance. Figure 14 shows the proportion of the total irradiance coming from the left of the sensor, the right of the sensor, and vertically, as a cloud passes over the sensor.

The 300nm plot shows a very variable angular distribution, probably because of both the small irradiance received at this wavelength, and the increased scattering the occurs at lower wavelengths. At 800 and 1100nm the irradiance from the right seems to decrease

as the cloud moves to the right. It might be expected that increased scattering from the cloud would cause this irradiance to increase, but it seems that the shadowing effects of the cloud are affecting the irradiance more significantly. Indeed, it may be that the scattering from the cloud is causing rays to be scattered so far that they reach the sensor from above, or even from the left.

Generally, as the cloud passes over the sensor (located at $x = 10$) the vertical irradiance reduces and the left and right irradiances increase. This is caused by the shadowing effect of the cloud, where light which would have reached the sensor from above is instead scattered, and some of it reaches the sensor from the left or right after multiple scatterings have taken place.

6.3 Effect of a single cloud at a distance

The simulations above were run with a grid size of 20 x 20, to provide a number of simulations without requiring a large amount of computational time. However, this small grid size makes it difficult to determine the effect of a cloud which is a long way away from the sensor, so a number of other simulations were performed with larger grid sizes.

The location of the cloud in each simulation is described by the *cloud location ratio*, defined as

$$\text{CLR} = \frac{\text{Horizontal distance from sensor to cloud}}{\text{Vertical height of cloud base}}$$

This is essential to allow the transfer of information gained from the model to real-world situations, as the model has no inherent scale, and therefore a cloud 20 cells away could represent a real-world distance of anything from 20m to 200km.

Simulations were then performed on a 20x200 grid with a large range of cloud distances. Differences between the irradiance spectra recorded with a cloud at a certain distance and no cloud were calculated, and the results were also processed to quantify the error which would then be produced in certain common remote sensing outputs if the effect of the cloud on the surface irradiance was not taken into account when correcting the imagery.

Spectra produced by the model were resampled to the Landsat 7 bands (Goward et al., 2001) using the Spectral Libraries feature of ENVI 4.8. The irradiances in the red and near-infrared bands (Landsat bands 3 and 4 respectively) were then converted to vegetation reflectances using the typical reflectance values of vegetation: 5% in the red band and 40% in the NIR band (personal communication, E.J. Milton) to produce simulated vegetation reflectances with no cloud, and clouds at various locations.

The Normalised Difference Vegetation Index (NDVI) was then calculated using the formula

$$\text{NDVI} = \frac{\text{NIR} - \text{Red}}{\text{NIR} + \text{Red}}$$

Although NDVI values are frequently used in remote sensing studies, they have no physical meaning. An important quantity that can be measured from satellite images is *Net Primary Productivity*. This is the amount of carbon which is fixed by plants and turned into biomass, and is very important for climate change studies. The calculated NDVI values were then converted to Net Primary Productivity (NPP) values using the formula from Ruimy et al. (1994)

$$\text{NPP} = e \times f \times S_0$$

where e is the conversion efficiency (set to 1.01 for temperate grassland), f is the absorption efficiency calculated as $-0.025 + 1.25 \times \text{NDVI}$ and S_0 is the incoming photosynthetically-active radiation (taken as the mean value given by Rossow, 1991), and the NPP is calculated in units of $\text{g}\,\text{m}^{-2}$.

NPP values calculated using such formulae provide essential data for studies investigating carbon fluxes in the biosphere, as the global NPP represents the total amount of carbon removed from the atmosphere by plants across the world. There has been much debate about the accuracy of global NPP values, as results from both models and satellite estimations range from around 40 KgC per year to over 65 KgC per year (Cramer et al., 1999). Some differences in the satellite-derived estimates may be due to errors in atmospheric correction, so it seems that comparing simulations by their difference in NPP values may be a suitable method to quantify the errors caused by ignoring clouds.

The mean differences between irradiance with a cloud at each cloud location (described by the cloud location ratio) and irradiance with no cloud are shown in Figure 15. These differences are calculated from the first four bands of the Landsat 7 resampled spectra, covering the visible and near-infrared parts of the spectrum.

This shows a decrease in irradiance when the cloud is fairly close to the sensor (for CLR < 5), but an increase when the cloud is further away. The decrease is likely to be caused by the shadowing effects of the cloud, which extend beyond the physical boundaries of the cloud itself. The increase in irradiance recorded for larger CLRs is caused by the cloud scattering light back towards the sensor, thus allowing rays which would have disappeared off the edge of the grid to be reflected back towards the sensor instead. The magnitude of the increase seems very variable, but it is interesting to note that there is no overall trend, and that the increase is still large even for CLRs of 12–14.

The NDVI differences shown in Figure 16 have the opposite shape to the irradiance differences shown in Figure 15. This means that although the irradiance overall increased, the NDVI was reduced because the irradiance increased more in the red than in the near

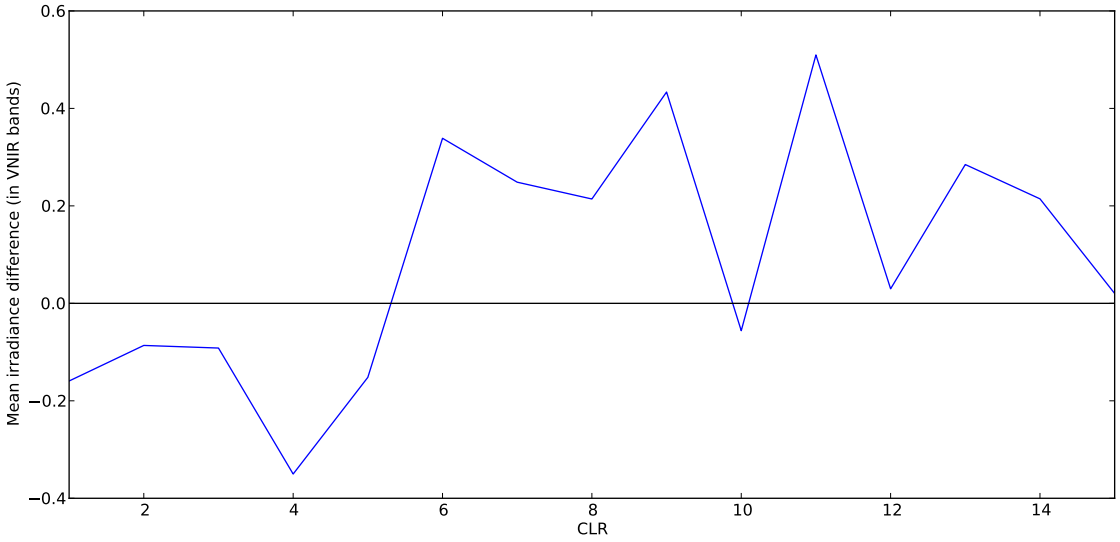


Figure 15 – Mean irradiance differences (across the VNIR bands) for cloud location ratios from 1 to 15, compared to no cloud.

infra-red. This is likely to be because scattering tends to occur more at lower wavelengths, and therefore red light is scattered more by the cloud than near-infrared light.

The differences in NDVI shown in Figure 16 cause a maximum difference of -7g m^{-2} in net primary productivity values, with a mean difference of -2.8g m^{-2}. These differences seem small, but lead to large differences when they are scaled up to the size of a satellite image. For example, if the net primary productivity of each pixel in a Landsat scene were to be reduced by the mean difference above, then there would be a difference of -89,000 tonnes of carbon in total (see Appendix D for details).

The maximum difference in NPP occurs at a CLR of 13 and there are differences in NPP even with a CLR of 17 (the largest CLR modelled in this study). The use of the CLR allows a rough comparison with real-world distances. In this case, assuming a cloud base height of 2000m significant differences in NPP would be found at a distance of 34km, or 21 miles. Thus, the model appears to suggest that an isolated *cumulus humilis* cloud present in a satellite image could affect the surface irradiance for all pixels within around 20 miles of it. It should be noted that this is only an approximation, and there are various issues with using CLR values, but it gives an impression of the significant of this effect.

6.4 Sensitivity analysis

Given the large variability of the data shown in the graphs and tables above, a sensitivity analysis was performed to assess the significance of the results produced by the model, and to inform decisions of further work.

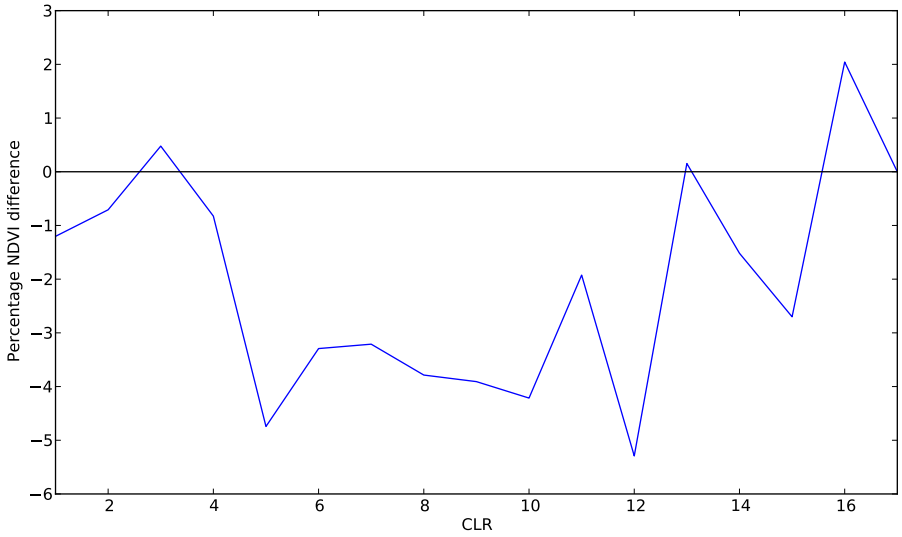


Figure 16 – Percentage differences between NDVI values calculated with a range of CLR values and NDVI values calculated for a simulation with no cloud.

The model was run four times with the same parameters, and Figure 17 shows the coefficient of variation of the irradiances received across these separate runs. The coefficient of variation accounts for the range of the values in its calculation, thus allowing the variability in different wavelengths to be compared. This shows that there is no significant differences in variability in the range of modelled wavelengths. Most wavelengths have a coefficient of variation of around 0.1, showing that irradiance variations within around 10% may be caused by the general variability of the model outputs and not be indicative of real-world change.

Some of the results shown above are within the 10% uncertainty boundary, but this does not necessarily mean that they are not significant. The differences between the four simulations seem to be roughly uniform across the wavelengths concerned, meaning that values calculated from a ratio such as NDVI (and therefore NPP) have less uncertainty, and values which are calculated as percentage differences are also less uncertain. The suggestions for further work detailed below include an investigation to understand how the model sensitivity can be reduced.

7 Conclusions

This project set out to answer the following question:

How much does a single cloud within a satellite image affect the surface irradiance across the image?

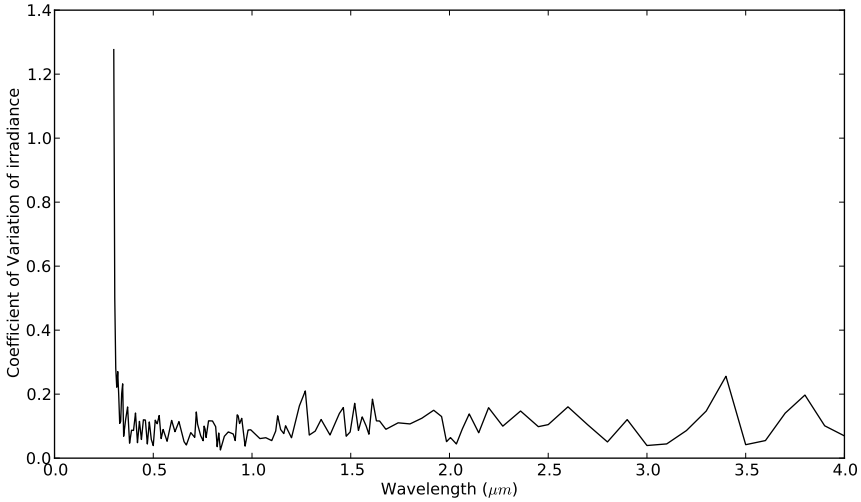


Figure 17 – Coefficiant of Variation for each wavelength for the set of four models run as part of the sensitivity analysis.

The answer appears to be that it can affect surface irradiance by enough to change NDVI values by 5%, and therefore change net primary productivity values by up to $9\mathrm{g\,m^{-1}}$, although the average changes are 4% and $2.8\mathrm{g\,m^{-1}}$ respectively. This effect seems to extend over a distance of at least twenty miles from the cloud, even if the cloud is a small isolated *cumilis* cloud. Although the effect on NPP values may seem small, when scaled to the size of a Landsat image, not correcting for the effect of a cloud at a distance may cause an underestimation of NPP by over 80,000 tonnes.

This result has the potential to significantly change the general view on the effect of clouds in the remote sensing community. At present isolated clouds are simply masked out of images and are assumed not to have affected an area any bigger than the cloud plus its shadow. The results shown in this paper suggest that this is not the case. This means that the standard radiative transfer models used for atmospheric correction in remote sensing (such as ATCOR and 6S) are not fit for purpose, as they do not take into account the effect of isolated clouds. This result will raise a number of questions for the remote sensing community, particularly those interested in the calibration and validation of remotely sensed data, and is likely to precipitate much further research.

As well as answering the main question above, this project has developed a flexible, theoretically-based exploratory model which could be used to answer many more questions. In many ways the major output of this project is the model code, which will allow scientific results far greater than those detailed above to be produced.

8 Further work

Although the results described earlier are of great interest, there are a number of aspects of the model which could be improved. These are described below.

8.1 Improvements to the core model

- **Three-dimensional representation of the atmosphere:** The biggest issue with the theoretical basis of the current model is that it represents the atmosphere as a two-dimensional grid. Using a three-dimensional grid would allow accurate angular simulation of light rays and would, for example, allow the simulation of the full angular distribution of surface irradiance as opposed to the simple left/right/vertical grouping which is used currently. However, implementing this would require the re-writing of the model core and all parameterisation would have to be done in a three-dimensional environment. For example, rather than using a simple azimuthal integration of the scattering function to provide probabilities for the eight cells surrounding the scattering point, the functions would have to be calculated for 26 cells, requiring phase functions to be specified as both azimuth and zenith probabilities.
- **Reflectance simulation:** Currently the model only simulates the ground incident solar irradiance. Although information about changes in solar irradiance are important to remote sensing scientists, the value they are really interested in is the reflected radiance measured at airborne or satellite sensors. Light which reaches these sensors has travelled through the atmosphere twice: once from the sun to the ground, where some of the light is reflected and then travels back through the atmosphere to the sensor. As the processes which occur in the atmosphere occur similarly whether light is travelling downwards or upwards, the implementation of this change would not be very difficult. The core model calculations would stay the same, and it would just involve calculating the reflected values (based on a user-specified bi-directional reflectance distribution function) leaving the surface and recording the irradiance at the location of the sensor within the atmosphere.
- **Improvement in repeatability:** The sensitivity analysis suggests that changes in output irradiance of up to 10% may be simply due to inherent variability in the model. Reducing this sensitivity to a level similar to the uncertainty associated with field spectral measurements (approximately 2–5% Milton et al., 2009) would increase the certainty associated with the conclusions drawn from the model, and allow comparison with field data. This could be done by allowing the stochasticity associated with the model to be controlled - for example allowing the aerosol amount grid to be stored and re-used rather than randomly assigning aerosol amounts to each cell for each model run.

8.2 Improvements to parametrisation

- **Quantitatively accurate outputs:** At present, the outputs from this model cannot be compared to those from other models, as the results are not quantitatively accurate. The results from other models tend to be around $1400\mathrm{W\,m^{-2}\,nm^{-1}}$, whereas the results from this model are around $30\mathrm{W\,m^{-2}\,nm^{-1}}$. The absorption coefficients are the same as those used by SPCTRAL2, so these are unlikely to be the problem. It is more likely that the other parameters such as the scattering probabilities given for each aerosol amount value, or the aerosol amount values themselves are wrong. Further study of the literature and investigation of similar models should help to produce parametrisations which will provide accurate outputs.
- **Inherent scale in the parametrisation:** The cloud location ratio was used to enable cloud distances to be compared to those in the real world. This is not ideal, as situations with similar cloud location ratios may not be comparable in terms of the radiative effect of the cloud. For example, the ratio uses the height of the cloud base as one of the parameters, but the effect of this on the resulting ground irradiance has not been investigated. Choosing a real-world size for each grid cell, in both the horizontal and vertical direction, and then parametrising the model with suitable values for the selected scale, would enable better comparison with real-world situations.

8.3 Performance improvements

- **Parallelisation:** Although the current code can be run on High Performance Computing systems such as the University of Southampton's Iridis cluster, there will not be a significant time benefit in doing so, as the current code does not take full advantage of the parallelisation capabilities of such systems. Improving the performance of the model will make it possible to run simulations with far greater numbers of rays, thus providing larger populations to make statistical inferences from, and will also allow wider ranges of parameter values to be tested within a reasonable timeframe. The most obvious way to parallelise the model is to set up the grid once, and then run a number of rays in parallel. This is a valid approach as the path of each ray is entirely independent. OpenMP is a parallel programming standard which allows loops (such as the loop calculating results for each of the rays simulated) to be easily parallelised, but IDL does not support this. Investigation as to whether extensions such as GPULIB (an IDL interface to Graphics Processing Unit programming; Messmer et al., 2008) or FastDL (an IDL interface to the MPI standard for parallel programming; Message Passing Interface Forum, 2009) could help improve performance should be performed.

8.4 Others

There are a number of more advanced ways in which the model can be extended. For example, the cloud cellular automaton model could be fully integrated with the radiative transfer model, allowing simulations to be performed recording irradiance at a point while a number of clouds develop and move above it. Furthermore, support for multiple sensors would enable the surface irradiance at a number of points spaced across an area the size of a satellite image to be measured while cloud systems develop and change overhead, giving a good idea as to how surface irradiance is likely to change over time. This data could then prove very valuable in characterising changes at ground calibration targets.

References

Berk, A., Anderson, G.P., Bernstein, L.S., Acharya, P.K., Dothe, H., Matthew, M.W., Adler-Golden, S.M., Chetwynd Jr, J.H., Richtsmeier, S.C., Pukall, B. et al., 1999, MODTRAN 4 radiative transfer modeling for atmospheric correction, in *Proceedings of SPIE-The International Society for Optical Engineering*, volume 3756, pp. 348–353

Bird, R. and Riordan, C., 1984, Simple solar spectral model for direct and diffuse irradiance on horizontal and tilted planes at the earth's surface for cloudless atmospheres, *SERI/TR-215-2436*, Solar Energy Research Inst., Golden, CO (USA)

Bird, R.E., Riordan, C.J. and Myers, D.R., 1987, Investigation of a Cloud-Cover Modification to SPCTRAL2, SERI's Simple Model for Cloudless-Sky Spectral Solar Irradiance, *SERI/TR-215-3038*, Solar Energy Research Inst., Golden, CO (USA)

Cahalan, R.F., Wiscombe, W.J., Bell, T.L. and Ridgway, W., 1994, The albedo of fractal stratocumulus clouds, *Journal of the Atmospheric Sciences*, 51(16)

Cramer, W., Kicklighter, D.W., Bondeau, A., Iii, B.M., Churkina, G., Nemry, B., Ruimy, A., Schloss, A.L. et al., 1999, Comparing global models of terrestrial net primary productivity (NPP): overview and key results, *Global Change Biology*, 5(S1), 1–15

Dobashi, Y., Kaneda, K., Yamashita, H., Okita, T. and Nishita, T., 2000, A simple, efficient method for realistic animation of clouds, in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pp. 19–28, ACM Press/Addison-Wesley Publishing Co.

Dobashi, Y., Nishita, T. and Okita, T., 1998, Animation of clouds using cellular automaton, in *Proc. of Computer Graphics and Imaging'98*, pp. 251–256

Ellingson, R., Ellis, J. and Fels, S., 1991, The intercomparison of radiation codes used in climate models- Long wave results, *Journal of Geophysical Research*, 96, 8929–8953

Glassner, A.S., 1989, *An introduction to ray tracing*, Morgan Kaufmann

Goward, S.N., Masek, J.G., Williams, D.L., Irons, J.R. and Thompson, R.J., 2001, The Landsat 7 mission:: Terrestrial research and applications for the 21st century, *Remote Sensing of Environment*, 78(1-2), 3–12

Gregory, G.L., Anderson, B.E., Barrick, J.D., Hudgins, C.H., Bagwell, D.R. and Blake, D.R., 1994, Analysis of tropospheric aerosol number density for aerosols of 0.2-to 3-μm diameter: Central and northeastern Canada, *Journal of geophysical research*, 99(D1), 1757–1762

Grieken, R.V. and Harrison, R.M., 1998, *Atmospheric Particles*, Wiley

Hernández-Encinas, A., Hernández Encinas, L., Hoya White, S., Martin del Rey, A. and Rodriguez Sanchez, G., 2007, Simulation of forest fire fronts using cellular automata, *Advances in Engineering Software*, 38(6), 372–378

Hess, M., Koepke, P. and Schult, I., 1998, Optical properties of aerosols and clouds: The software package OPAC, *Bulletin of the American Meteorological Society*, 79(5), 831–844

Highwood, E.J., 2000, Effect of cloud inhomogeneity on direct radiative forcing due to aerosols, *Journal of geophysical research*, 105(D14), 17843–17

Hooper, F.C. and Brunger, A.P., 1980, A model for the angular distribution of sky radiance, *Journal of Solar Energy Engineering*, 102, 196

Institute of Remote Sensing, 2007, User's Guide for the Software Package SCIATRAN (Radiative Transfer Model and Retrieval Algorithm) Version 2.2, *Technical report*, University of Bremen

Iqbal, M., 1983, *An introduction to solar radiation*, Academic Press

ITTVIS, 2011, Interactive Data Language (IDL), version 8.0

Kokhanovsky, A.A. and Nauss, T., 2005, Satellite-based retrieval of ice cloud properties using a semianalytical algorithm, *Journal of Geophysical Research*, 110, 0–4

Leckner, B., 1978, The spectral distribution of solar radiation at the earth's surface–elements of a model, *Solar energy*, 20(2), 143–150

Liu, B.Y.H. and Jordan, R.C., 1960, The interrelationship and characteristic distribution of direct, diffuse and total solar radiation* 1, *Solar Energy*, 4(3), 1–19

Mandelbrot, B.B., 1983, *The fractal geometry of nature*, Wh Freeman

Message Passing Interface Forum, 2009, MPI: A Message-Passing Interface Standard - Version 2.2, *Technical report*, Message Passing Interface Forum

Messmer, P., Mullowney, P.J. and Granger, B.E., 2008, GPULib: GPU computing in high-level languages, *Computing in Science & Engineering*, pp. 70–73

Milton, E.J., Schaepman, M.E., Anderson, K., Kneubühler, M. and Fox, N., 2009, Progress in field spectroscopy, *Remote Sensing of Environment*, 113, S92–S109

Murray, A.B., 2003, Contrasting the goals, strategies, and predictions associated with simplified numerical models and detailed simulations, in P. Wilcock and R. Iverson (Editors), *Predictions in Geomorphology*, pp. 151–168, AGU, Washington D.C., USA

Nagel, K. and Raschke, E., 1992, Self-organizing criticality in cloud formation?, *Physica A: Statistical Mechanics and its Applications*, 182(4), 519–531

Nield, J.M. and Baas, A.C.W., 2008, Investigating parabolic and nebkha dune formation using a cellular automaton modelling approach, *Earth Surface Processes and Landforms*, 33(5), 724–740

O'Hirok, W. and Gautier, C., 1998, A Three-Dimensional Radiative Transfer Model to Investigate the Solar Radiation within a Cloudy Atmosphere. Part I: Spatial Effects, *Journal of the Atmospheric Sciences*, 55(12), 2162–2179

Randel, D.L., Haar, V., Thomas, H., Stephens, G.L., Greenwald, T.J., Ringerud, M.A. and Combs, C.L., 1996, A new global water vapor dataset, *Bulletin of the American Meteorological Society*, 77(6), 1233–1246

Richter, R., 2007, Atmospheric/Topographic Correction for Airborne Imagery, ATCOR-4 User Guide, Version 4.2, *DLR-German Aerospace Center, Remote Sensing Data Center*

Rossow, W.B., 1991, Spatial and temporal variability of global surface solar irradiance, *Journal of Geophysical Research*, 96(C9), 16–839

Rothman, L.S., Jacquemart, D., Barbe, A., Chris Benner, D., Birk, M., Brown, L.R., Carleer, M.R., Chackerian, C. et al., 2005, The HITRAN 2004 molecular spectroscopic database, *Journal of Quantitative Spectroscopy and Radiative Transfer*, 96(2), 139–204

Rozanov, A., Rozanov, V., Buchwitz, M., Kokhanovsky, A. and Burrows, J.P., 2005, SCIATRAN 2.0-A new radiative transfer model for geophysical applications in the 175-2400 nm spectral region, *Advances in Space Research*, 36(5), 1015–1019

Ruimy, A., Saugier, B. and Dedieu, G., 1994, Methodology for the estimation of terrestrial net primary production from remotely sensed data, *Journal of Geophysical research*, 99(D3), 5263–5283

Tanré, D., Deroo, C., Duhaut, P., Herman, M., Morcrette, J.J., Perbos, J. and Deschamps, P.Y., 1990, Description of a computer code to simulate the satellite signal in the solar spectrum: the 5S code, *International Journal of Remote Sensing*, 11(4), 659–668

University of Wyoming Department of Atmospheric Sciences, 2011, Soundings Data, URL `http://weather.uwyo.edu/upperair/sounding.html`

Van Heuklon, T.K., 1979, Estimating atmospheric ozone for solar radiation models, *Solar Energy*, 22(1), 63–68

Vermote, E.F., Tanré, D., Deuze, J.L., Herman, M. and Morcette, J.J., 1997, Second simulation of the satellite signal in the solar spectrum, 6S: An overview, *IEEE Transactions on Geoscience and Remote Sensing*, 35(3), 675–686

Vidot, J., Jourdan, O., Kokhanosvky, A.A., Szczap, F., Giraud, V. and Rozanov, V.V., 2010, Retrieval of tropospheric NO2 columns from satellite measurements in presence of cirrus: A theoretical sensitivity study using SCIATRAN and prospect application for the A-Train, *Journal of Quantitative Spectroscopy and Radiative Transfer*, 111(4), 586–601

Wehrli, C.H., 1985, WRC Reference Spectrum, *Technical report*, PMOD Publication 615

White, R., Engelen, G. and Uljee, I., 1997, The use of constrained cellular automata for high-resolution modelling of urban land-use dynamics, *Environment and Planning B*, 24, 323–344

Yuwen, S., Ying, D. and Zhihui, W., 1996, The Analysis of Physical Feature of Aerosol Particles in Autumn over Shijiazhuang Area [J], *METEOROLOGICAL MONTHLY*, 2

9 Appendix A

Below are a number of pieces of documentation for Py6S.

9.1 Introduction

Py6S is a wrapper for the 6S radiative transfer model (Vermote et al., 1997). It allows programmatic creation of 6S input files, and reading of 6S output files. It is designed to hide the difficulties of creating correctly formatted input and output files and present a simple interface for parameterising, running and processing output from 6S.

WARNING: Py6S is still in early development. If it destroys anything, don't blame me.

9.2 Installation

9.2.1 Compiling 6S

Py6S is not a replacement for 6S, it is simply a *wrapper*. Therefore, it needs to have access to a working 6S executable. The Fortran source for 6S can be downloaded from `ftp://kratmos.gsfc.nasa.gov/pub/eric/6S/`. Ensure you have a Fortran compiler installed (the Fortran compiler from the GNU Compiler Collection (`http://gcc.gnu.org/`) is a good choice) and then simply run `make` which will produce an executable.

9.2.2 Helping Py6S find 6S

Py6S needs to know where to find the 6S executable. This can be done programmatically by setting the location (see below) but it is better to ensure that the 6S executable is on your system PATH. On POSIX systems (Linux, Unix, OS X) this can be done in two ways:

- Add the directory containing 6S to your `PATH` environment variable (this can normally be done through your shell initialisation files, for example `.bashrc` for the BASH shell, and rename the executable to `sixs`
- Create a symbolic link in somewhere like `/usr/bin` to the 6S executable. For example, on my Mac I can do this by typing `ln -s /usr/bin/sixs /Users/robin/6S/SixSV1.1`

Note: The executable must be named `sixs` for Py6S to find it.

9.2.3 Testing

In the directory where Py6S is installed, run `python SixS.py`. This should produce an output similar to below:

```
6S wrapper script by Robin Wilson
Using 6S located at /usr/bin/sixs
```

9.3 Usage

9.3.1 Basic Usage

The general process for using Py6S is to:

1. Instantiate the SixS class
   for example, `model = SixS()`
2. Set the parameters appropriately
   for example, `model.solar_z = 50`
   or `model.wavelength = WavelengthTypes.Wavelength(0.550, 0.600)`
3. Call the `run()` method to run the model
   for example, `model.run()`
4. Read the outputs from the `outputs` member variable
   for example, `print model.outputs.irradiance_diffuse`

See the code example in the main body of the report (p14) for a well-commented example of using 6S

9.3.2 Loading and Saving Parameters

Once the SixS object has been initialised with a number of parameters, these can be saved to a file to allow easy restoring of the values at a later time. Of course, the parameters could instead be written to a 6S format input file, but there is no way to read these values in to this wrapper due to the format of the input file. By saving using these methods the parameter values can be easily restored.

The file format in which the parameters are saved is the YAML format. This is plain-text-based, human-readable, human-editable, and parsers exist for a large number of programming languages.

To save the parameters, simply call the `save_params` class method, for example

```
model = SixS()
model.aero_soot = 0.5
model.aero_water = 0.5

SixS.save_params(model, "params_output.yml")
```

Note that the `save_params` method is called from the SixS class as opposed to the instance of the class ('model' in this case).

This will produce output like the following:

```
!!python/object:SixS.SixS
aero_dustlike: 0
aero_oceanic: 0
aero_soot: 0
aero_water: 0
aot550: 0.5
day: 14
ground_reflectance: 1.0
month: 7
sixs_path: /usr/bin/sixs
solar_a: 264
solar_z: 32
view_a: 190
view_z: 23
visibility: null
wavelength: 0.45300000000000001
```

All of the parameters are listed in alphabetical order, with human-readable names. This file can be edited by hand simply by changing the values. The order of values is not important, but it is essential to keep the top line unchanged.

To load the parameters from a previously saved (or manually created) file, simply call the `load_params` class method, for example:

```
model = SixS.load_params("params_output.yml")
```

Again, note that this method is a class method as opposed to an instance method, so is called as such.

9.4 Dealing with errors

A number of errors can occur when running 6S through Py6S. When these errors are encountered a Python exception is raised. Some errors are convered by the built-in exceptions provided by Python (such as `FileErrors` and `AttributeErrors`), but some are

more specialised. Therefore, three new exceptions have been created for Py6S:

- `ParameterError` - raised if there is an error in the 6S parameter specification (for example, an impossible combination of parameters)
- `ExecutionError` - raised if there is an error executing the 6S model (for example the executable cannot be found)
- `OutputParsingError` - raised if there is an error parsing the text output produced by 6S (for example, the output file was not produced properly)

These are defined in the `SixSExceptions` module, and must be imported if they are to be used in a `try...except` block.

All of these errors will only be raised during the `run()` method (and the methods that this calls), therefore any `try...except` block should be around the call to `run()`.

10 Appendix B

The calculation of the scattering probability for an example cell is given below. The parameters of the cell and ray are given in Table 2.

Table 2 – Parameters for the worked example of scattering calculations

Parameter	**Value**
Aerosol Type	Maritime
Aerosol Amount	8 particles per cm^3
Wavelength	350nm

The OPAC database (Hess et al., 1998) gives the single-scattering albedo for the maritime aerosol type at 350nm as 0.909. The given probability of scattering for 8 particles percm^3 is 0.255. Therefore, the aerosol scattering probability is

$$0.909 \times 0.255 = 0.231$$

The Rayleigh scattering probability can be calculated from the equation above

$$p_R = \frac{1}{150}\frac{1}{0.35^4} = 0.444$$

Therefore, the final probabilities to be used in the choice of scattering type are

- Rayleigh scattering: 0.222
- Aerosol scattering: 0.115
- No scattering: 0.663

A random number generator (generating uniformly-distributed random numbers between 0–1) is used to choose which type of scattering to perform. If we assume that Rayleigh scattering was chosen, it is necessary to calculate the direction the ray will be scattered in.

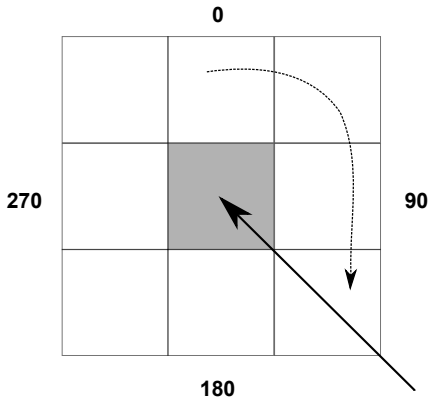
The phase function for Rayleigh scattering is shown in Figure 5(a). The function was integrated over the segments shown in Figure 6 to produce a probability for scattering to each cell, shown in Table 3.

Table 3 – Probabilities for scattering to each of the surrounding cells

0.125	0.163	0.125
0.087		0.087
0.125	0.163	0.125

Again, a random number generator is used to choose which cell to scatter to. However, the scattering angle is the difference between the angle of incidence and angle of reflection, so the probabilities must be shifted (or 'rotated') around cells to ensure that the probability for 0° is at the cell that the ray entered through. Figure 18 shows how the probabilities are rotated.

Figure 18 – The process of shifting probabilities. Incident ray shown as solid line, and direction and magnitude of shift shown by dotted line.



After rotation the probabilities are as shown in Table 4, and the random number generator can be used to calculate which cell the ray should be scattered to.

Table 4 – Probabilities for scattering to each of the surrounding cells after rotation has been performed

0.163	0.125	0.087
0.125		0.125
0.087	0.125	0.163

11 Appendix C

The total irradiance received at a hemispherical-viewing sensor consists of the direct component and the diffuse component. The direct component is that which has come directly from the solar disc, through the atmosphere, to the sensor. The diffuse component is that which has been scattered by the atmosphere, and therefore does not appear to come from the solar disc, but comes from elsewhere in the sky.

The diffuse to global ratio (that is, the ratio of the diffuse irradiance to the total irradiance) is often used as a measure of the cloudiness of the sky, as (Liu and Jordan, 1960) showed that there was a strong relationship between this ratio and the total transmittance of the atmosphere. The ratio can be measured by many widely-available instruments such as the Delta-T BF3 Sunshine Sensor.

A number of models go further than the simple diffuse:global ratio and describe the full distribution of light across the sky. Hooper and Brunger (1980) devised the Three-Component Continuous Distribution model which describes the irradiance distribution across the sky as the sum of three distributions:

- **Isotropic:** A distribution of constant brightness across the whole sky
- **Heliocentric:** A distribution focussing on the light at the solar disc, and the area of brightness around the disc
- **Horizon brightening:** A distribution focussing on the brightening of the horizon of the sky (which occurs due to greater scattering because of greater path length)

In a future three-dimensional version of RTWRTM, irradiance distributions such as those described by the Hooper and Brunger model could be simulated and the accuracy of models assessed.

12 Appendix D

Calculation of the possible difference in net primary productivity (NPP) across a Landsat scene given the differences found in the model runs for this project is as follows.

The average difference between NPP with cloud and NPP without cloud is -2.8g m^{-2}. A Landsat scene is 185km by 172km (Goward et al., 2001), which is $3.182 \times 10^{10} m^2$. If every pixel in the Landsat scene was vegetation and was affected by the mean difference then the overall difference in NPP would be -8.91×10^{10}g, or -89,000 tonnes.

Taking a more realistic example, assuming that 50% of the Landsat scene was vegetation, and that all of these vegetation pixels were affected by the mean difference, the overall difference would be -44,000 tonnes.