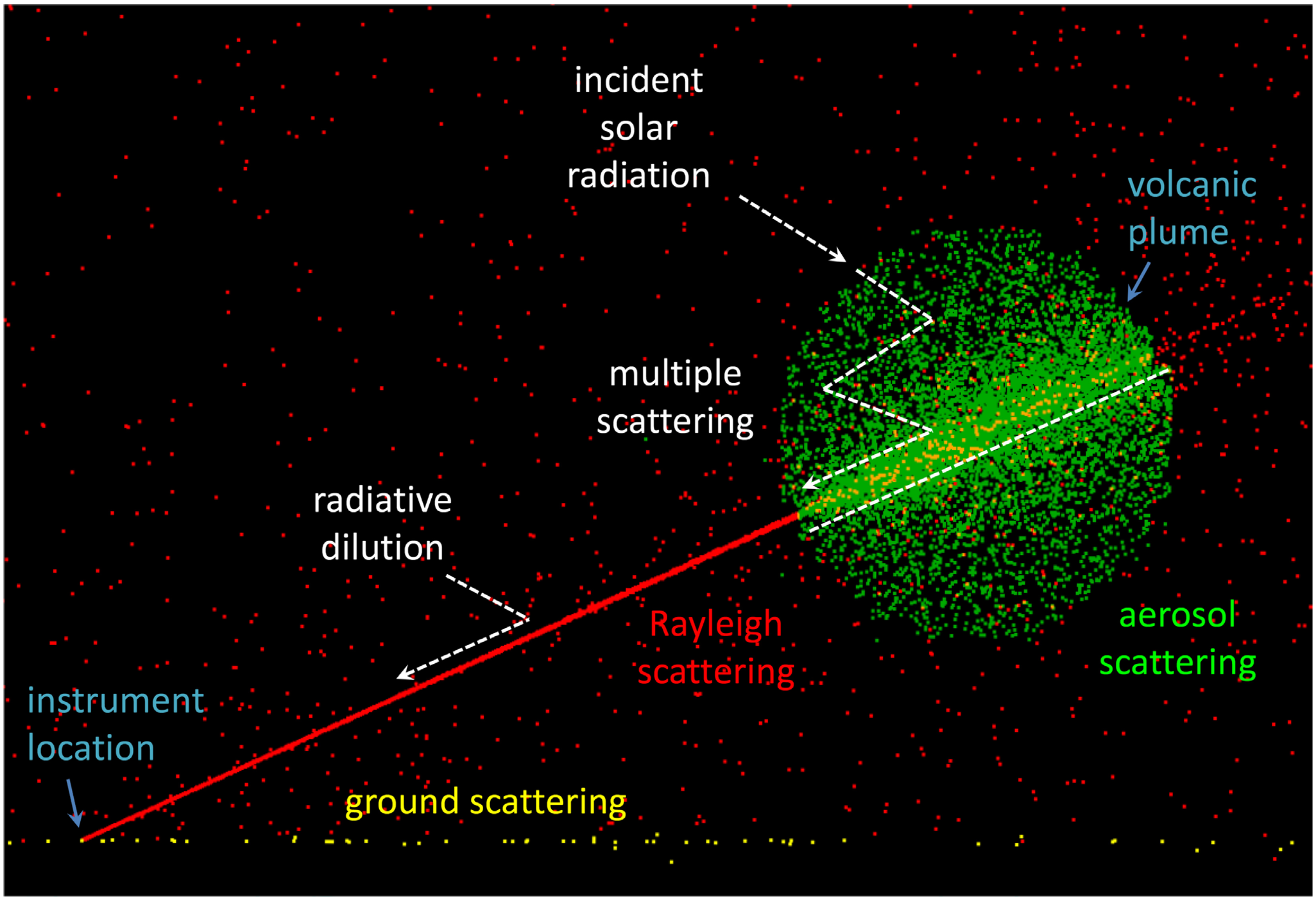
USGS
science for a changing world

An Interface for Simulating Radiative Transfer In and Around Volcanic Plumes with the Monte Carlo Radiative Transfer Model McArtim

By Christoph Kern



Open-File Report 2016–1045

U.S. Department of the Interior
U.S. Geological Survey

Cover. Graphical representation of simulated radiative transfer in and around a volcanic plume. Red dots indicate Rayleigh scattering in the atmosphere, green dots indicate scattering on aerosols in the volcanic plume, and yellow dots indicate scattering on the ground. Radiation detected by an instrument (lower left corner) has taken a variety of different paths through or around the plume.

An Interface for Simulating Radiative Transfer In and Around Volcanic Plumes with the Monte Carlo Radiative Transfer Model McArtim

By Christoph Kern

Open-File Report 2016–1045

U.S. Department of the Interior
U.S. Geological Survey

U.S. Department of the Interior
SALLY JEWELL, Secretary

U.S. Geological Survey
Suzette M. Kimball, Director

U.S. Geological Survey, Reston, Virginia: 2016

Any use of trade, firm, or product names is for descriptive purposes only and does not imply endorsement by the U.S. Government.

Suggested citation:
Kern, Christoph, 2016, An interface for simulating radiative transfer in and around volcanic plumes with the Monte Carlo radiative transfer model McArtim: U.S. Geological Survey Open-File Report 2016–1045, 18 p., http://dx.doi.org/10.3133/ofr20161045.

ISSN 2331-1258 (online)

Contents

Figures

Tables

Conversion Factors

International System of Units to U.S. customary units

Multiply	By	To obtain
	Length	
centimeter (cm)	0.3937	inch (in.)
meter (m)	3.281	foot (ft)
kilometer (km)	0.6214	mile (mi)
	Area	
square centimeter (cm^2)	0.001076	square foot (ft^2)
square centimeter (cm^2)	0.1550	square inch (ft^2)
	Volume	
cubic centimeter (cm^3)	0.06102	cubic inch (in^3)

An Interface for Simulating Radiative Transfer In and Around Volcanic Plumes with the Monte Carlo Radiative-Transfer Model McArtim

By Christoph Kern

Abstract

This report describes two software tools that, when used as front ends for the three-dimensional backward Monte Carlo atmospheric-radiative-transfer model (RTM) McArtim, facilitate the generation of lookup tables of volcanic-plume optical-transmittance characteristics in the ultraviolet/visible-spectral region. In particular, the differential optical depth and derivatives thereof (that is, weighting functions), with regard to a change in SO_2 column density or aerosol optical thickness, can be simulated for a specific measurement geometry and a representative range of plume conditions. These tables are required for the retrieval of SO_2 column density in volcanic plumes, using the simulated radiative-transfer/differential optical-absorption spectroscopic (SRT-DOAS) approach outlined by Kern and others (2012). This report, together with the software tools published online, is intended to make this sophisticated SRT-DOAS technique available to volcanologists and gas geochemists in an operational environment, without the need for an indepth treatment of the underlying principles or the low-level interface of the RTM McArtim.

Introduction

In recent years, the use of ultraviolet (UV) remote-sensing techniques for the detection and quantification of SO_2 in volcanic plumes has become increasingly widespread (see review by Kantzas and Mcgonigle, 2008). These techniques rely on solar radiation scattered in the atmosphere as a light source for gas measurements. Conventionally, the solar radiation is assumed to propagate along a straight line from the scattering point, through the plume, to the instrument. This conceptual model, however, has been shown to fail when used to measure plumes rich in SO_2 or with heavy aerosol loads, or when measuring SO_2 distribution from a distance of more than 1 to 2 kilometers (km) (Kern and others, 2009). In such situations, a more complex model must be used to take into account the radiative transfer of the scene. Kern and others (2012) recently developed a retrieval technique called simulated radiative-transfer/differential optical-absorption spectroscopy (SRT-DOAS), which integrates moderate-resolution spectroscopic measurements and radiances simulated by using the three-dimensional backward Monte Carlo atmospheric-radiative-transfer model (RTM) McArtim (Deutschmann and others, 2011). In this method, simulated radiances and weighting functions take the place of the absorption cross sections used in conventional DOAS retrievals (see Platt and Stutz, 2008, and references therein). Application of SRT-DOAS can substantially improve the accuracy of SO_2 column-density and emission-rate measurements (Kern and others, 2012). Thus, efficient generation of the requisite lookup tables permits rapid application of SRT-DOAS in an operational environment.

In this report, a front end to the RTM McArtim is described that can be used to generate lookup tables containing these radiances and weighting functions for use in operational SRT-DOAS retrievals. In particular, two software programs are described: LT-Builder (Lookup Table Builder) and RTVP (Radiative Transfer in Volcanic Plumes). Both programs are interfaces for the RTM McArtim (version 2; Deutschmann and others, 2011), and both are freely available from the VHub Web site (https://vhub.org/). The published repositories each contain an appropriate release of the RTM McArtim.

The LT-Builder software is high-level code, written in the MATLAB programming language, that facilitates the simulation of spectral radiances, optical densities, and radiance derivatives in the UV spectral region (typically 300–340 nanometers [nm]) that are evaluated when applying UV remote-sensing instrumentation to volcanic plumes. The name originates in the program's ability to batch process different volcanic-plume conditions, thus lending itself to the generation of lookup tables for use in SRT-DOAS retrievals, as described by Kern and others (2012).

The RTVP software is a low-level front end for the RTM McArtim that allows a parameterization of volcanic plumes for model initialization. RTVP is compiled C++ code that can be called from the Windows command-line interpreter. RTVP provides more flexibility than is available through LT-Builder, yet it is considerably less user friendly when run as a stand-alone program. LT-Builder uses RTVP as an interface to the RTM McArtim, as shown schematically in figure 1. Users interested in simulating radiances, optical densities, and weighting functions—for example, for use in SRT-DOAS retrievals—will likely find LT-Builder to have sufficient functionality to meet their needs and may not need to familiarize themselves with RTVP. However, documentation for using RTVP as a stand-alone program is also provided for those users who require a more flexible, albeit less intuitive, front end to the RTM McArtim.

After simulations are run, the model output files (*.rtm and *.adr) are automatically translated to text files (*.txt) that contain spectral data. The various resulting spectrum types are described in the next section, but all the text files can easily be imported into any plotting software or, if further processing is required, into the DOAS Intelligent System (DOASIS) software. Some basic postprocessing steps in DOASIS are also described in this report.

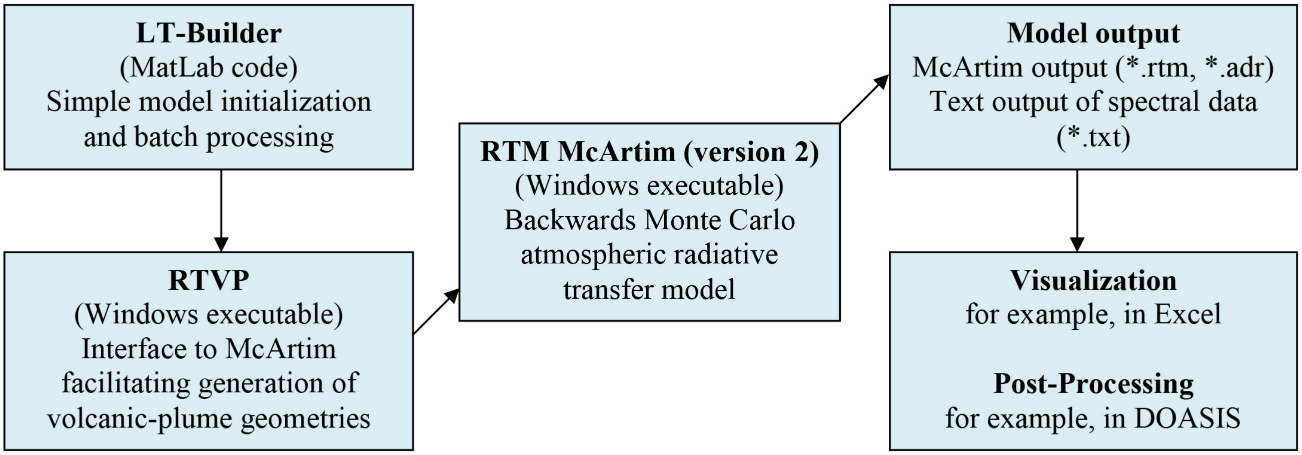


Figure 1. Diagram showing flow of information for simulating radiative transfer in and around volcanic plumes using the Lookup Table Builder (LT-Builder) and Radiative Transfer in Volcanic Plumes (RTVP) interfaces for the radiative transfer model (RTM) McArtim.

LT-Builder

The LT-Builder software was designed as a simple tool for simulating spectral radiances, optical densities, and radiance derivatives in volcanic-plume geometries for use in sophisticated spectroscopic retrievals, particularly for SRT-DOAS approaches (Kern and others, 2012). In these retrievals, simulated optical densities and derivatives with regard to SO_2 absorption and plume aerosol optical thickness (AOT) take the place of the absorption cross sections used in conventional DOAS retrievals. These more sophisticated retrievals are performed to improve the accuracy of derived column densities whenever three-dimensional radiative-transfer effects are significant (see Kern and others, 2009). To facilitate the fast evaluation of operational datasets, these retrievals require a library (lookup table) of optical densities and weighting functions simulated over a range of plume conditions. Although simulation of an entire lookup table is still time and computationally intensive, initializing the RTM McArtim for the generation of such a table is simple using LT-Builder. Here, each step of the process is explained, beginning with system requirements and installation. Examples are given, and postprocessing with the DOASIS software is explained.

System Requirements

- A 32-bit Microsoft Windows system, preferably on a multicore CPU platform
- LT-Builder software
  The MATLAB LT-Builder software is used to iterate through different atmospheric conditions and thus build the lookup table. The program is kept as simple as possible and makes only the most basic RTM settings accessible.
- MathWorks MATLAB program
  The LT-Builder software is written in the MATLAB programming language and is currently unavailable in a compiled version; therefore, the software needs the MATLAB interpreter to run. MATLAB can be purchased online from MathWorks (http://www.mathworks.com/products/matlab/); version 2008a or higher is required.
- RTVP software
  The RTVP software is a front end, written in Microsoft Visual C++, for the monochromatic RTM McArtim (version 2) that allows the simulation of spectra. RTVP was developed by the author and is included in the LT-Builder distribution package. Documentation for using RTVP as a stand-alone tool is presented in the next section.
- Microsoft Visual C++ 2010 redistributable package
  The Microsoft Visual C++ 2010 redistributable package is required to run programs compiled with Microsoft Visual C++ 2010, in which the RTVP software was written. The package can be downloaded from Microsoft’s Web site (http://www.microsoft.com/en-us/download/details.aspx?id=5555). The package is not required if Microsoft Visual C++ is already installed on the computer.
- McArtim (version 2)
  McArtim (Deutschmann and others, 2011) is a backward Monte Carlo atmospheric radiative inversion transfer model developed at the University of Heidelberg, Germany. A compiled version of McArtim (version 2, compiled for Windows 32-bit systems) is included in the LT-Builder distribution package.
- DOASIS
  DOASIS is a spectral acquisition and evaluation software developed at the University of Heidelberg, Germany. It is free of charge but requires registration. It is available for download

from the DOASIS Web site (http://www.iup.uni-heidelberg.de/institut/forschung/groups/atmosphere/software). DOASIS is not required for simulation of optical densities but is needed for building scattering weighting functions or convolving the simulated spectra with instrument line shapes of respective instruments. Several example DOASIS scripts are included in the LT-Builder distribution package.

Installation and Directory Structure

The LT-Builder software is written in the MathWorks MATLAB programming language and does not require installation on the system. It does rely on the availability of both RTVP and McArtim, which are included in the LT-Builder distribution package. To get started, unpack the zip file to the hard drive, which will automatically generate a base directory named *LT-Builder* (hereafter, the exact names of folders, files, classes, class methods and class parameters will be shown in italics. Also note the following slight differences in spelling. LT-Builder refers to the name of the software, *LT-Builder* refers to the name of the base folder, and *LTBuilder* refers to the name of a specific class. Similarly, *RTVPSettings* refers to the name of a specific class, while *RtvpSettings* refers to a handle pointing to that class.. This directory contains the MATLAB code and several subfolders. The LT-Builder code is very basic, consisting of only three files. One file, *LTBuilder.m*, defines the *LTBuilder* class. Once initialized, the class properties are then used to store the information required to build the lookup table, and the class methods can be used for setting up a directory structure and running the RTM. Another file, *RTVPSettings.m*, defines a second class that is used by the *LTBuilder* class to manipulate the settings of the RTVP program. It is automatically initialized by the *LTBuilder* class, so it can be considered a subclass.

The RTM McArtim (version 2, compiled for Windows) is included in the LT-Builder distribution package and is located in a subfolder named *McArtim*. The subfolder named *RTVP* will already contain the compiled RTVP program, which also is included in the LT-Builder distribution package. The subfolder named *DOASIS scripts* contains some example scripts that demonstrate the postprocessing of simulation results, as briefly described below. The subfolder named *Output* is the standard folder for output of the lookup tables, but this location can be changed by the user.

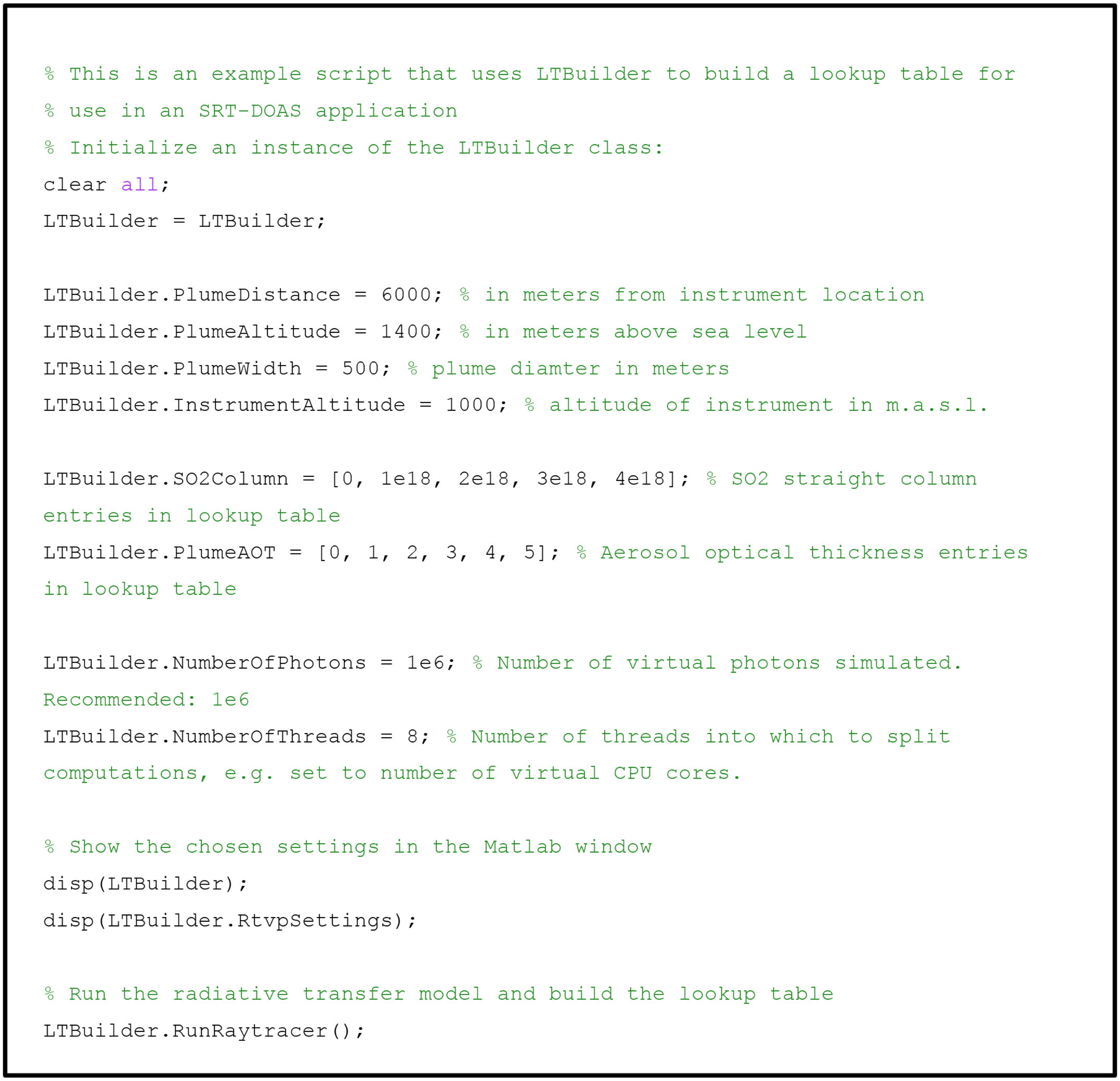
Because the *LTBuilder* class calls the RTVP front end, which was written in Microsoft Visual C++, the Microsoft Visual C++ redistribution package must be installed on the system before running LT-Builder. Also, if postprocessing of simulated spectra is needed, the DOASIS software may prove valuable. See the System Requirements section of this report for information on how to obtain these additional packages, both of which are distributed with dedicated Windows installers.

Quick Reference—How to Simulate a Spectrum or Build a Lookup Table

To simulate a spectrum or build a lookup table with the *LTBuilder* class, first define an instance of the class in MATLAB. For example, enter the following line into the MATLAB command interpreter:

```
LTBuilder = LTBuilder();
```

Next, adapt the required settings to reflect the desired plume geometry and measurement setup. Finally, start the simulations by calling the method *LTBuilder.RunRaytracer()*. This procedure is demonstrated in the script example *BuildLookupTableExample.m*, which is included in the LT-Builder distribution package (fig. 2).



```
% This is an example script that uses LTBuilder to build a lookup table for
% use in an SRT-DOAS application
% Initialize an instance of the LTBuilder class:
clear all;
LTBuilder = LTBuilder;

LTBuilder.PlumeDistance = 6000; % in meters from instrument location
LTBuilder.PlumeAltitude = 1400; % in meters above sea level
LTBuilder.PlumeWidth = 500; % plume diamter in meters
LTBuilder.InstrumentAltitude = 1000; % altitude of instrument in m.a.s.l.

LTBuilder.SO2Column = [0, 1e18, 2e18, 3e18, 4e18]; % SO2 straight column
entries in lookup table
LTBuilder.PlumeAOT = [0, 1, 2, 3, 4, 5]; % Aerosol optical thickness entries
in lookup table

LTBuilder.NumberOfPhotons = 1e6; % Number of virtual photons simulated.
Recommended: 1e6
LTBuilder.NumberOfThreads = 8; % Number of threads into which to split
computations, e.g. set to number of virtual CPU cores.

% Show the chosen settings in the Matlab window
disp(LTBuilder);
disp(LTBuilder.RtvpSettings);

% Run the radiative transfer model and build the lookup table
LTBuilder.RunRaytracer();
```

Figure 2. Example MatLab script that can be used to generate a lookup table for use in sophisticated spectroscopic retrievals. This script (BuildLookupTableExample.m) is included in the software distribution of the Lookup Table Builder (LT-Builder) software.

Running the example code in MatLab will generate a lookup table, simulating the spectral radiance, optical density, and absorption-weighting function for a specific geometry and range of plume SO_2 column densities and AOTs (defined at 300 nm). Note that the measurement and plume geometries are set to constant values, whereas the parameters named *SO2Column* and *PlumeAOT* are set to arrays of values in this example. These arrays define the individual positions of the lookup table; all combinations of SO_2 straight-column density and plume AOT will be simulated. If a single spectrum is desired, rather than an entire table, the column density and AOT can be set to a single value instead of an array.

Other than the geometry and plume settings, few parameters are needed. The *NumberOfPhotons* parameter will determine the precision of the simulations. A low number (~10^3) can be used for quick testing, but a value of 10^6 is recommended to obtain sufficient precision when building tables for use in SRT-DOAS retrievals. Note, however, that on typical desktop computers, simulation of a single lookup-table position with 10^6 photons may take several hours. The processing speed for an entire table can be improved by assigning the different table positions to separate CPU cores. In the given example, the *NumerOfThreads* parameter was set to 8, thus starting eight independent processes to take advantage of the eight virtual cores on this particular system.

Some additional settings are accessible as properties of the *LTBuilder* and *RTVPSettings* classes. In the given example, the default values were kept, but all the properties are displayed in the MATLAB window using the commands *disp(LTBuilder)* and *disp(LTBuilder.RtvpSettings)*. Note that in this syntax, *RtvpSettings* is a handle to an instance of the *RTVPSettings* class. Therefore, the capitalization is different. Finally, the command *RunRaytracer()* is called to start the simulation. This method will create a subfolder named *Threads* in the *LT-Builder* base directory and copy various instances of the RTM McArtim and the RTVP program into it. Each instance will be initialized to process certain lookup-table positions, and then each instance will be started. A command-prompt window will appear for each instance in which the current model output is shown.

After all the model instances are started, MATLAB can be closed because it is no longer required to run the simulations. As they are obtained, the lookup-table entries will appear in the specified output directory. By default, this is the subfolder *Output* in the *LT-Builder* base directory. The filenames of the lookup-table entries begin with the specified geometry, according to the following naming convention: *<PlumeDistance> <PlumeAltitude> <PlumeWidth> <SO2Column> <PlumeAOT> <InstrumentAltitude>.**

The filename extension specifies the file type for the respective geometry. Some file types are internal RTM files, but the following file types are of special interest:

<OutputFile>.meas.txt	The optical radiance simulated for a geometry looking toward the center of the plume.
<OutputFile>.ref.txt	The optical radiance simulated for a geometry looking away from or without a plume.
<OutputFile>.od.txt	The optical depth obtained by dividing the radiance simulated outside the plume by the radiance simulated for the plume center and then applying the natural logarithm.
<OutputFile>.adr.txt	The weighting function with respect to SO_2 absorption.

Visualization and Postprocessing of Lookup-Table Entries

The text files created by the LT-Builder program can be imported into any data-plotting program. The files contain two (tab-delimited) columns of ASCII data, the first of which is the wavelength, and the second of which represents the respective spectrum value. Of particular value is the ability to open the spectra directly with the DOASIS software, which can then be used to view the output, as well as to perform additional processing steps on the simulated spectra.

If, for example, the output is to be used for an SRT-DOAS evaluation, the weighting functions with regard to scattering in the plume will be required. These weighting functions can be derived from the difference quotients of the lookup-table entries once they have all been simulated. DOASIS provides a scripting functionality that allows quick building of these scattering derivatives for all lookup-table entries. *LT_Build_SDR.js*, which is a j-script file that performs this task, is included in the LT-Builder distribution package inside the subfolder named *DOASIS scripts*. A second j-script file, *LT_Convolute.js*, provides an example of how the lookup-table entries can be convolved with the instrument line shape of a given spectrometer and mapped onto its wavelength grid. This operation is also required when using the model output to evaluate actual measurement data.

Programming References

The LT-Builder code is simple and most easily understood by using the quick reference provided in the previous section. However, for the sake of completeness, all properties and methods of the *LTBuilder* and the *RTVPSettings* classes are described in this section. Users unfamiliar with object-oriented programming in MATLAB may find the online MathWorks documentation on this topic useful (http://www.mathworks.com/discovery/object-oriented-programming.html). Additional help can be obtained by pressing F1 while the cursor is on a class, method, or property name in MATLAB.

LTBuilder Class

The *LTBuilder* class contains properties that store the most important information needed to simulate a lookup table, as well as the high-level methods needed to do so. Create an instance of this class, set the properties to the appropriate values, and start the simulation. Properties and methods available to the user are listed in tables 1 and 2.

Table 1. *LTBuilder* class public properties.
[LT-Builder, Lookup Table Builder; RTVP, Radiative Transfer in Volcanic Plumes; molecules/cm^2, molecules per square centimeter]

Property	Type	Default value	Description
BaseDir	string	C:\LT-Builder\	Base directory containing the LT-Builder software.
RTMDir	string	C:\LT-Builder\McArtim\	Folder containing the radiative transfer model McArtim.
RTVPDir	string	C:\LT-Builder\RTVP\	Folder containing *RTVP.exe*.
OutputDir	string	C:\LT-Builder\Output\	Output folder for the lookup table.
SO2Column	array of floats	[0, 1e18, 2e18, 3e18, 4e18]	Array of SO_2 column densities to use in lookup-table creation, in molecules/cm^2.
PlumeAOT	array of floats	[0, 1, 2, 3, 4, 5]	Array of plume aerosol optical depths to use in lookup-table creation.
PlumeDistance	int	2,000	Distance between instrument and plume (in meters).
PlumeAltitude	int	1,000	Altitude of plume (in meters above sea level).
PlumeWidth	int	500	Width of plume (in meters).
InstrumentAltitude	int	0	Altitude of instrument (in meters above sea level).
NumberOfThreads	int	8	Number of threads into which to split the computation—for example, set to number of virtual CPU cores.
RtvpSettings	*RTVPSettings*	See table 3	*RTVPSettings* class containing settings related to the RTVP program.
NumberOfPhotons	int	10^6	Number of virtual photons to simulate.

Table 2. *LTBuilder* class public methods.
[LT-Builder, Lookup Table Builder; RTVP, Radiative Transfer in Volcanic Plumes; AOT, aerosol optical thickness]

Method	Description
LTBuilder()	Constructor of the *LTBuilder* class; also initializes several settings to default values; returns *LTBuilder*.
ThreadRollout()	Creates a folder named *Threads* in the *LTBuilder* base directory. Here, it creates two subfolders per thread—an instance of RTVP and an instance of the radiative-transfer model.
ThreadCleanup()	Removes the *Threads* folder with all of its subfolders.
RunRaytracer()	Cleans up the *Threads* folder, rolls out new instances of RTVP and the radiative-transfer model, generates batch files for batch processing of the entries in the lookup table, and then starts each batch file, each of which start an instance of the radiative-transfer model. Command-prompt windows appear, one for each process. Once all processes have been successfully started, MATLAB can be closed.
AddLTEntries()	Creates two text files in the output folder—one containing the SO_2 column densities that are simulated in the lookup table and one containing the plume AOTs. These files are later used to identify which entries exist in the lookup table.
BuildBatchFiles()	Generates a batch file for each instance of the RTVP program that successively calls the required commands to generate the lookup-table entries.
CheckBkSl(string)	Checks whether the final character in the input text string is a backslash; if not, it adds one. The method returns the string with a backslash at the end.

RTVPSettings Class

The *RTVPSettings* class is used to store settings that are written to the initialization file of the RTVP program. An instance of *RTVPSettings* is created on construction of the *LTBuilder* class. In most cases, these settings can be left unchanged. If changes are desired, be sure to call the *WriteSettings()* method to save them to disk. (Note that this method is also called operationally by *LTBuilder.ThreadRollout()*.) Changes to *RTVP_settings.ini* will be saved and used as defaults in future instances of *LTBuilder*. The interface of the *RTVPSettings* class is described in tables 3 and 4.

Table 3. *RTVPSettings* class public properties.
[RTVP, Radiative Transfer in Volcanic Plumes; AOT, aerosol optical thickness; LT-Builder, Lookup Table Builder]

Property	Type	Default value	Description
RTVPPath	string	C:\LT-Builder\RTVP\	Folder containing the RTVP program and settings files.
NumberOfPhotons	int	1e6	Number of virtual trajectories to simulate.
UseGaussianPlumeCrossSection	bool	0	0, homogeneous SO_2 and aerosol concentrations in the plume; 1,Gaussian distribution.
AngstromExponent	float	1.2	Angstrom exponent for plume aerosol scattering.
SpectralResolution	float	0.05	Spectral resolution (in nanometers) at which to simulate the lookup table.

Property	Type	Default value	Description
SpectrumStartWavelength	float	300	Lower end of the wavelength window (in nanometers) in which to simulate spectral radiances.
SpectrumEndWavelength	float	340	Upper end of the wavelength window (in nanometers) in which to simulate spectral radiances.
ExternalReferenceSpectrum	string	C:\LT-Builder\RTVP\References\null_ref_300-340nm-0.05nm.txt	Filename of a fixed external reference that contains the spectral radiance in the absence of SO_2 or aerosols. Note that the number of spectral channels must match the wavelength range of the simulation. Additional references can be simulated by specifying 0 SO_2 and AOT in *LTBuilder*, and selecting a new wavelength range.
RTMPrimaryBaseDir	string	C:\LT-Builder\McArtim\	Folder containing the radiative-transfer model McArtim.
OutputDir	string	C:\LT-Builder\Output\	Output folder for RTVP results.

Table 4. *RTVPSettings* class public methods.

[RTVP, Radiative Transfer in Volcanic Plumes]

Method	Description
RTVPSettings()	Constructor of the *RTVPSettings* class; returns an instance of the class.
ReadSettings()	Reads selected settings from the file *RTVP_settings.ini* in the folder specified by the *RTVPPath* property; settings are loaded into the properties described in the previous table.
WriteSettings()	Writes the settings stored in the *RTVPSettings* class properties to the file *RTVP_settings*.ini located in the folder specified by the *RTVPPath* property.

Note that the *RTVPSettings* class does not access all the settings and functionality that the RTVP low-level front end for the RTM McArtim provides; therefore, it is not designed as a complete interface for use of RTVP as a standalone program. Instead, the *RTVPSettings* class acts as an interface between the *LTBuilder* class and the RTVP program, making the settings needed by the *LTBuilder* class accessible.

The low-level front end RTVP for the RTM McArtim can, however, be used as a stand-alone program by calling it from the command line. Although this method should not be needed for users simply seeking to calculate spectral radiances and(or) lookup tables for use with SRT-DOAS, it may be suitable to users looking for more flexibility in their application of McArtim to volcanic-plume scenarios. Therefore, documentation of RTVP is provided in the following section.

RTVP

RTVP is a low-level front end for the RTM McArtim. Written and compiled in Microsoft Visual C++, RTVP provides a simple command-line interface for users to initialize and run McArtim in geometries relevant for the volcano-emissions remote-sensing community. However, users interested only in a simple means of simulating spectral radiances, optical densities, and weighting functions, for example for use in SRT-DOAS retrievals (as described by Kern and others, 2012) may not need to familiarize themselves with RTVP. Instead, a high-level front end specifically written for this application is available in the form of a MATLAB program (see previous section).

System Requirements

- Microsoft Windows operating system
- Microsoft Visual C++ Redistributable Package installed (available from http://www.microsoft.com/en-us/download/details.aspx?id=5555)
- Compiled RTVP package
- McArtim RTM, version 2, compiled for 32-bit Windows operating system (included in the RTVP distribution package)

Installation and Directory Structure

Because RTVP is compiled to an executable file, it does not require installation. To begin, unpack the zip file to a directory on the hard drive; the base directory will contain the file *RTVP.exe*. This executable file can be called from the Windows Command Prompt; the syntax for doing so is described in the next section. The file *RTVP_settings.ini*, also contained in the base directory, holds settings for the RTM, which are also discussed below. A folder named *McArtim* will be generated in the base directory; this folder contains two subfolders (*Raytrace1* and *Raytrace2*), each of which contains an instance of McArtim. Depending on how the RTVP program is used, one or both of these instances may be called simultaneously. The *Results* folder can be used as a target for RTM output, and the *Scenarios* folder contains an example measurement scenario, as shown in the examples described below.

Running RTVP from the Command Prompt

As mentioned above, RTVP is compiled to run from the Windows Command Prompt. Launch a new command prompt by clicking ‘Start’ and typing “cmd”. As the RTVP folder is most likely not in the system path, navigate to the folder in which the file *RTVP.exe* is located by using the commands *cd..* (back up one directory level) and *cd [dir]* (change directory). Once the RTVP base directory is active, typing *rtvp* followed by *Enter* will return syntax information regarding the use of RTVP as a front end for McArtim. The general syntax is

rtvp –[operation mode] [command line variables]

The operation mode defines the behavior of McArtim; individual operation modes are described in the next section. The command-line variables are parameters that define the geometry of the simulation and tell McArtim where to write the model output files. As described later, the required command-line variables change depending on the operation mode.

RTVP Operation Modes

The first parameter following *rtvp* on the command line represents the RTVP operation mode, a three- or four-letter identifier, preceded by a hyphen, that defines what the output of the RTM should be. The valid operation modes, along with the command-line variables required for each operation mode, are listed in table 5. The individual command-line variables are discussed in the next section.

Table 5. RTVP operation modes.

[RTVP, Radiative Transfer in Volcanic Plumes; DOAS, differential optical-absorption spectroscopy; AOT, aerosol optical thickness]

Mode identifier	Command-line variables	Description of operation mode

-scd	Wavelength(nm)
	PlumeDistance(m)
	PlumeAltitude(m.a.s.l.)
	PlumeDiameter(m)
	SO_2Column(molec/cm^2)
	PlumeAOT(300nm)
	InstrumentAltitude(m.a.s.l.)
	OutputFile
		Runs RTVP in slant column density mode. In this mode, RTVP simulates the slant column density of SO_2 that an instrument would detect for a single given wavelength and a fixed measurement scenario. A DOAS instrument cannot derive a column density from a single wavelength but instead uses a range of wavelengths. The model can simulate the column for any wavelength.

-odd	PlumeDistance(m)
	PlumeAltitude(m.a.s.l.)
	PlumeDiameter(m)
	SO_2Column(molec/cm^2)
	PlumeAOT(300nm)
	InstrumentAltitude(m.a.s.l.)
	ReferenceElevationAngle(deg)
	LineShapeFile
	OutputFile
		Runs RTVP in optical density mode using discrete wavelengths. In this mode, RTVP simulates an optical density spectrum for a given measurement scenario. Both measurement and reference spectra are simulated at discrete, equally spaced wavelengths. The spectral window and simulated wavelength resolution are set in file *RTVP_settings.ini*.

-ods	PlumeDistance(m)
	PlumeAltitude(m.a.s.l.)
	PlumeDiameter(m)
	SO_2Column(molec/cm^2)
	PlumeAOT(300nm)
	InstrumentAltitude(m.a.s.l.)
	ReferenceElevationAngle(deg)
	LineShapeFile
	WavelengthSelectionFine
	OutputFile
		Runs RTVP in optical density mode using specific wavelengths. In this mode, RTVP simulates an optical density spectrum for a given measurement scenario. Both the measurement and reference spectra are simulated for specific wavelengths given as a list in a text file. The filename of the wavelength file is passed as a command-line variable.

-inv	PlumeDistance(m)
	PlumeAltitude(m.a.s.l.)
	PlumeDiameter(m)
	InitialSO_2Column(molec/cm^2)
	InitialPlumeAOT(300nm)
	InstrumentAltitude(m.a.s.l.)
	ReferenceElevationAngle(deg)
	MeasurementSpectrum
	ReferenceSpectrum
	CalibrationFile
	LineShapeFile
	OutputFile
		Runs RTVP in inversion mode. In this mode, RTVP varies the atmospheric conditions in an attempt to minimize the discrepancy between measured and modeled optical density spectra. In addition to the measurement conditions, measured plume and reference spectra are passed to the raytracer as command-line variables. **Note: This operation mode is not yet validated and is provided only as a beta version.**

Mode identifier	Command-line variables	Description of operation mode
-chi2	PlumeDistance(m) PlumeAltitude(m.a.s.l.) PlumeDiameter(m) InstrumentAltitude(m.a.s.l.) ReferenceElevationAngle(deg) MeasurementSpectrum ReferenceSpectrum CalibrationFile LineShapeFile LowSO_2Column(molec/cm^2) HighSO_2Column(molec/cm^2) SO_2ColumnResolution(molec/cm^2) LowPlumeAOT HighPlumeAOT PlumeAOTResolution OutputFile	Generates a topography of CHI$^{\mathbf{2}}$ values depicting the difference between a single specific measured optical depth and the optical depth simulated for various plume conditions (see Kern and others, 2012). The measured optical depth is given by passing a measurement and reference spectrum, and the limits of the topography are set by passing the range and resolution of SO_2 and AOT values to consider. **Note: This operation mode is not yet validated and is provided only as a beta version.**

RTVP Command-Line Variables

RTVP has two kinds of settings; those that are passed to the program by using the command line and those that are configured in the initialization file *RTVP_settings.ini*. The settings passed by using the command line are generally those that define the plume geometry and SO_2/aerosol content, whereas those configured in the initialization file are typically not changed so often. Each program operation mode requires its own specific set of command-line variables; those that are required can be queried by typing *rtvp* at the command line or consulting table 5. Note that the correct number of variables in the correct order must always be provided for each operation mode. If the number of variables does not match the mode's requirements, the basic syntax information is displayed, and the model is not run. The various RTVP command-line variables are listed in table 6.

Table 6. RTVP command-line variables.
[AOT, aerosol optical thickness; nm, nanometers]

Variable	Description
PlumeDistance	Distance (in meters) between the instrument and the center of the volcanic plume.
PlumeAltitude	Altitude (in meters above sea level) of the volcanic plume.
PlumeDiameter	Diameter (in meters) of the volcanic plume (assumed to have a circular cross section).
InstrumentAltitude	Assumed altitude (in meters above sea level) of the remote-sensing instrument. The elevation angle of the instrument-viewing direction is chosen such that it points toward the center of the plume.
ReferenceElevationAngle	Elevation angle (in degrees above the horizon) used to simulate a reference spectrum for the measurement (unless a fixed reference is used—see settings in the initialization file).

Variable	Description
MeasurementSpectrum	Path and file name of a measured plume spectrum. Spectrum must be in the *.std ASCII format (see examples below).
ReferenceSpectrum	Path and file name of a measured Fraunhofer reference spectrum. Spectrum must be in *.std ASCII format (see examples below).
CalibrationFile	Path and file name of a calibration text file. File must contain a list of wavelengths corresponding to each channel of the spectrometer.
LineShapeFile	Path and file name of the instrument line-shape file. File must contain a single column of ASCII data describing the instrument response to a monochromatic light source, calibrated according to the calibration file above.
InitialSO_2Column	Initial SO_2 column density (in molecules per square centimeter) considered during inversion of this parameter in the *inv* operating mode.
InitialPlumeAOT	Initial plume AOT (determined at 300 nm) considered during inversion of this parameter in the inversion operating mode.
LowSO2Column	Lower limit of SO_2 column densities (in molecules per square centimeter) considered in chi2 mode.
HighSO2Column	Upper limit of SO_2 column densities (in molecules per square centimeter) considered in chi2 mode.
SO2ColumnResolution	Resolution of SO_2 column densities (in molecules per square centimeter) considered in chi2 mode.
LowPlumeAOT	Lower limit of plume AOTs (determined at 300 nm) considered in chi2 mode.
HighPlumeAOT	Upper limit of plume AOTs (determined at 300 nm) considered in chi2 mode.
PlumeAOTResolution	Resolution of plume AOTs (determined at 300 nm) considered in chi2 mode.
WavelengthSelectionFile	File containing a list of wavelengths at which the simulation will be run.
OutputFile	Path and file name for results of the simulation. Several files will be created, with variations in the file extension, indicating the type of result (see quick reference for examples).

Scenario Files

Instead of passing all geometry variables to RTVP on the command line, scenario files can be created that contain the geometry settings. Scenario files contain all of the variables described above. An example scenario file, named *example_scenario.txt*, is included in the RTVP distribution package, in the *Scenarios* subfolder. Any text editor can be used to make changes to the scenario file, but the number and order of the entries in the scenario file must be maintained (even if not all entries are used in the specific operation mode), and the last entry must be followed by a carriage return (blank line).

After the scenario parameters are adjusted by editing the scenario file, RTVP is called by using the name of the scenario file in place of the command-line variables:

rtvp –[operation mode] [scenario file]

Settings in the Initialization File

Besides the settings passed to RTVP by way of the command line, additional settings are defined in the *RTVP_settings.ini* file. The various entries in *RTVP_settings.ini* are listed in table 7. Any text editor can be used to edit this file. Note that the order of entries may not be changed while editing, and the last entry must be followed by a carriage return (blank line). Most of these entries do not need to be changed because the default values are appropriate in most cases and some parameters were introduced for debugging purposes only. Parameters that may need to be adjusted by the user are as follows:

- RTM_PRIMARY_BASEDIR—contains the path to the first instance of McArtim and must be set by the user to match the folder in which McArtim is located on the current system.
- RTM_SECONDARY_BASEDIR—contains the path to the second instance of McArtim and must be set by the user to match the folder in which McArtim is located on the current system.
- RTM_OUTPUT_DIR—contains the path to the output folder into which results will be saved; this folder must exist on the current system.
- RTM_NUMBEROFPHOTONS—sets the number of virtual trajectories simulated by the McArtim; more trajectories lead to a more precise result but take longer to simulate.
- RTM_USE_GAUSSIAN_PLUME_CROSS_SECTION—toggles the SO_2 concentration and AOT in the plume cross section between homogeneous (set to 0) or Gaussian (set to 1) distribution.
- RTM_ANGSTROM_EXPONENT—Angstrom exponent for aerosol scattering within the plume.
- RTVP_ODD_SPECTRAL_RESOLUTION—Resolution used for spectra simulated in optical sensity mode using siscrete wavelengths (ODD mode, see description of program modes in table 5).
- RTVP_ODD_INV_BEGIN_SPECTRUM_SIMULATION—lower limit of wavelength range simulated in ODD mode.
- RTVP_ODD_INV_END_SPECTRUM_SIMULATION—upper limit of wavelength range simulated in ODD mode.

Table 7. All settings configured in the *RTVP_settings.ini* initialization file (RTVP version 1.0.0).
[AOT, aerosol optical thickness; ODD, Optical Density using Discrete wavelengths; INV, inversion]

Parameter	Type	Default value	Mode	Description
RTVP_SEND_OUTPUT_TO_FILE	bool	0	all	Toggle the logfile on/off.
RTVP_OUTPUT_FILE	string	RTVP_out.txt	all	Logfile name.
RTM_PRIMARY_BASEDIR	string	C:\RTVP\McArtim\Raytrace1	all	Path to first instance of the RTM McArtim.
RTM_SECONDARY_BASEDIR	string	C:\RTVP\McArtim\Raytrace2	all	Path to second instance of the RTM McArtim.
RTM_OUTPUTDIR	string	C:\RTVP\Results\	all	Output path for results.

Parameter	Type	Default value	Mode	Description
RTM_NUMBEROFPHOTONS	int	1e6	all	Number of simulated trajectories.
RTM_SAVE_TRAJECTORIES	bool	0	all	Toggle save trajectories to disk.
RTM_NUMBER_OF_VOXALS_IN_PLUME_DIAMETER	int	21	all	Spatial resolution of the plume.
RTM_USE_GAUSSIAN_PLUME_CROSS_SECTION	bool	0	all	Toggle Gaussian distribution of SO_2 and aerosols in the plume. Default is homogeneous distribution.
RTM_ANGSTROM_EXPONENT	float	1.2	all	Angstrom exponent of aerosol scattering.
RTVP_ODD_SPECTRAL_RESOLUTION	float	0.05	odd	Spectral resolution of simulation in ODD mode.
RTVP_ODD_INV_BEGIN_SPECTRUM_SIMULATION	float	300	odd, inv	Lower limit of wavelength range (in nanometers) during spectrum simulation in ODD mode.
RTVP_ODD_INV_END_SPECTRUM_SIMULATION	float	340	odd, inv	Upper limit of wavelength range (in nanometers) during spectrum simulation in ODD mode.
RTVP_OD_INV_CHI2_CONVOLUTE_WITH_ILS	bool	0	odd, ods, chi2, inv	Toggle convolution with instrument slit function on/off.
RTVP_OD_INV_CHI2_USE_FIXED_EXTERNAL_REFERENCE_SPEC	bool	0	odd, ods, chi2, inv	Toggle use of an external reference spectrum on/off.
RTVP_OD_INV_CHI2_EXTERNAL_REFEERENCE_SPEC_FILE	string	C:\RTVP\Refs\300-340nm-0.05nm.txt	odd, ods, chi2, inv	Path and file name of external reference spectrum (number of channels must match the current simulation).
RTVP_INV_CHI2_STARTING_DAMPING_FACTOR	float	1e-3	inv	Damping factor (γ) in Levenberg Marquardt inversion (INV mode).
RTVP_INV_CHI2_DAMPING_FACTOR_MULTIPLIER	float	3	inv	Multiplier for damping factor in Levenberg Marquardt inversion.
RTVP_INV_AOD_DAMPING_ENHANCEMENT_FACTOR	float	1	inv	Damping factor enhancement for aerosol optical depth (as opposed to γ for SO_2 column density).
RTVP_INV_CHI2_USE_ONLY_SELECTED_WAVELENGTHS	bool	0	inv, chi2	Toggle selection of specific wavelengths for spectrum simulation (as opposed to discrete wavelengths as in ODD mode) on/off.

Parameter	Type	Default value	Mode	Description
RTVP_INV_CHI2_WAVELENGTH_SELECTION_FILE	string	C:\RTVP\refs\SelectedWL_SO2.txt	inv, chi2	Wavelength selection file (used only if above parameter is set to 1).
RTVP_INV_CHI2_STOP_CONDITION_IF_CHI2_SMALLER_THAN	float	1e-6	inv, chi2	Stop condition for Levenberg-Marquardt inversion of SO_2 and AOT (INV mode).
RTVP_INV_CHI2_MAX_ITERATIONS	int	20	inv, chi2	Maximum iterations in Levenberg Marquardt inversion of SO_2 and AOT.
RTVP_INV_CHI2_FIT_POLYNOMIAL_ORDER	int	3	inv, chi2	Order of additional polynomial to fit to measurement data.
RTVP_INV_ALLOW_RETROACTIVE_ADJUSTMENT_OF_POLYNOMIAL	bool	1	inv	Additionally adjusts polynomial for best after each iteration of inversion.
RTVP_INV_START_P0	float	0	inv	Initial value of fit polynomial coefficient 0.
RTVP_INV_START_P1	float	0	inv	Initial value of fit polynomial coefficient 1.
RTVP_INV_START_P2	float	0	inv	Initial value of fit polynomial coefficient 2.
RTVP_INV_START_P3	float	0	inv	Initial value of fit polynomial coefficient 3.

Examples

In this section, a few examples are given to demonstrate how RTVP can be used as a front end to McArtim in order to simulate radiative transfer in and around a volcanic plume. Several operation modes are demonstrated. As discussed above, the behavior of the McArtim will differ in dependence on the RTVP operation mode. Note that simulation times will vary in dependence on system performance.

Slant Column Density Program Mode

An example call of RTVP in Slant Column Density (SCD) mode would be

```
rtvp -scd 315 3000 1000 700 1e18 1 0 testout.dat
```

Upon pressing Enter, RTVP will start the raytracer and simulate the column density of SO_2 at 315 nm measured from a location 3 km from the plume center. In this scenario, the plume would be placed at 1 km above sea level, and its diameter would be 700 m. The SO_2 straight-column density, which is the same as the vertical-column density for a round-plume cross section like this, was chosen to be 1×10^{18} molecules/cm^2. For a homogeneous plume, this value amounts to an SO_2 concentration of 1.43×10^{13} molecules/cm^3 in the plume. The plume aerosol optical depth is 1, so a homogeneous plume would contain a scattering aerosol with an aerosol extinction coefficient of 1.43 km^{-1}. Finally, the results will be written to the ASCII file *testout.dat*.

Check the output file. In this particular example, the simulated SO_2 column density should be ~8.4×10^{17} molecules/cm^2, ~16 percent lower than the true column, owing to radiative dilution occurring between the plume and the instrument. Because McArtim is a Monte Carlo model, the results could differ slightly when the model is run a second time. A higher precision can be obtained by running the model multiple times, or by increasing the number of simulated photons in the *RTVP_settings.ini* file.

Optical Density at Discrete Wavelengths Program Mode

For the next example, set the number of photons to 1,000 by editing the *RTVP_settings.ini* file. The ODD (Optical Density using Discrete wavelengths) program mode can be used to simulate an entire optical-density spectrum:

```
rtvp -odd 3000 1000 700 1e18 1 0 90 c:\rtvp \scenarios\slit.slf testout2.dat
```

Here, RTVP will start the raytracer and simulate both the measurement spectrum (taken with the instrument pointing to the center of the plume) and the reference spectrum. In this example, the reference spectrum is specified to have been taken pointing straight up (at an elevation angle of 90°).

Several result files are produced in the output folder. The files *testout2.ref.txt* and *testout2.meas.txt* are the simulated spectra, which can be viewed by using DOASIS. Notice that the results are still quite noisy if only 1,000 photons are used for each wavelength. The files *testout2.refc.txt* and *testout2.measc.txt* are the spectra after convolution with the slit function. Finally, the file *testout2.od.txt* is the simulated optical density, which can be compared with a specific measurement.

Optical-Density at Specific Wavelengths Program Mode

The Optical-Density at Specific wavelengths (ODS) program mode is similar to ODD mode; the only difference is that a list of wavelengths is additionally passed to RTVP in ODS mode. The program then calculates the optical density for these specific wavelengths. An example call of the ODS mode is.

```
rtvp -ods 3000 1000 700 1e18 1 0 90 c:\rtvp\scenarios\slit.slf c:\rtvp\scenarios\calibration.clb
testout2.dat
```

Here, the simulation will take place at the wavelengths corresponding to the individual spectrometer channels, as defined in the file *calibration.clb*. The output format is the same as in the ODD mode.

Conclusions

The DOAS method is increasingly being applied by volcanologists around the world to measure the emission rate of SO_2 from volcanic vents. However, conventional DOAS retrievals fail to produce accurate results in situations where the measured volcanic plume is farther than about 2 km from the instrument, the plume contains high SO_2 or significant aerosol loads, or if clouds or fog are present between the instrument and the plume (Kern and others, 2009).

In such cases, measurement accuracy can be significantly improved by utilizing an advanced retrieval technique such as the SRT-DOAS method (Kern and others, 2012). Though the principles of this method have already been described in great detail, the applicability of the technique by interested researchers has been limited by the complexity of the McArtim RTM. The software programs LT-Builder and RTVP address this issue. They provide a frontend to McArtim specifically for calculating lookup tables for use by the SRT-DOAS method and simulating UV radiative transfer in volcanic plumes. By using these tools, the SRT-DOAS approach should now be more readily accessible to scientists studying volcanic gas emissions with DOAS instruments.

Acknowledgments

C. Kern would like to thank Mike Doukas, Chris Lockett, George Havach, and Jessica Dyke for their helpful reviews, which have greatly improved this report. Tim Deutschmann is gratefully acknowledged for providing the McArtim software and helpful advice on its use.

References Cited

Deutschmann, T., Beirle, S., Frieß, U., Grzegorski, M., Kern, C., Kritten, L., Platt, U., Prados-Román, C., Puķite, J., Wagner, T., Werner, B., and Pfeilsticker, K., 2011, The Monte Carlo atmospheric radiative transfer model McArtim—Introduction and validation of Jacobians and 3D features: Journal of Quantitative Spectroscopy and Radiative Transfer, v. 112, p. 1119–1137, doi: 10.1016/j.jqsrt.2010.12.009.

Kantzas, E.P., and Mcgonigle, A.J.S., 2008, Ground based ultraviolet remote sensing of volcanic gas plumes: Sensors, v. 8, p. 1559–1574.

Kern, C., Deutschmann, T., Vogel, L., Wöhrbach, M., Wagner, T., and Platt, U., 2009, Radiative transfer corrections for accurate spectroscopic measurements of volcanic gas emissions: Bulletin of Volcanology, v. 72, p. 233–247, doi: 10.1007/s00445-009-0313-7.

Kern, C., Deutschmann, T., Werner, C., Sutton, A.J., Elias, T., and Kelly, P.J., 2012, Improving the accuracy of SO_2 column densities and emission rates obtained from upward-looking UV-spectroscopic measurements of volcanic plumes by taking realistic radiative transfer into account: Journal of Geophysical Research, v. 117, no. D20, doi: 10.1029/2012JD017936.

Platt, U., and Stutz, J., 2008, Differential optical absorption spectroscopy; principles and applications: Springer-Verlag Berlin Heidelberg, 598 p.