Manuscript Details

Manuscript number	JQSRT_2019_12
Title	Matrix exponential in C/C++ version of vector radiative transfer code IPOL
Article type	Full Length Article

Abstract

We use only left eigenvectors to evaluate the matrix exponential in the method of discrete ordinates for the vector radiative transfer equation, which neglects circular polarization, in a plane-parallel atmosphere. This is contrary to a common practice of using the right eigenvectors to evaluate the matrix exponential combined with the left eigenvectors to avoid the inversion of the matrix of the right ones. Two numerical tests for Rayleigh and Aerosol scattering confirm our idea. For better explanation of our approach and for independent crosscheck of our results, we distribute an example in C/C++.

Keywords	matrix exponential; left eigenvectors; vector radiative transfer equation
Corresponding Author	Sergey Korkin
Corresponding Author's Institution	USRA GESTAR
Order of Authors	Sergey Korkin, Alexei Lyapustin
Suggested reviewers	Dmitry Efremenko, adrian doicu, Yoshifimi Ota

Submission Files Included in this PDF

File Name [File Type]

key_findings.docx [Highlights]

paper.pdf [Manuscript File]

figure1.docx [Figure]

figure2.docx [Figure]

table1.docx [Table]

table2.docx [Table]

references.docx [Supplementary Material]

To view all the submission files, including those not included in the PDF, click on the manuscript title on your EVISE Homepage, then click 'Download zip file'.

Research Data Related to this Submission

Data set	https://github.com/korkins/jqsrt_2019a

c/c++ source code

must be linked again freeware GSL library

Key findings:

- The use of the left eigenvectors, without the right ones, conveniently evaluate the matrix exponential;
- Although the use of only the left eigenvectors does not promise much gain in speed, it simplifies the development and support of the radiative transfer code, especially in C/C++;
- This approach is applicable for transfer of unpolarized or lineally polarized light;
- The known computation of eigenvalues in half-space is still applicable;
- Based on our experience and literature analysis, we believe this approach is new.

Matrix exponential in C/C++ version of vector radiative transfer code IPOL

Sergey Korkin [a,b,1], *Alexei Lyapustin* [b]

[a] USRA GESTAR, 7178 Columbia Gateway Drive, Columbia, MD, USA 21046

[b] NASA GSFC, 8800 Greenbelt Rd., Greenbelt, MD, USA 20771

Abstract:

We use only left eigenvectors to evaluate the matrix exponential in the method of discrete ordinates for the vector radiative transfer equation, which neglects circular polarization, in a plane-parallel atmosphere. This is contrary to a common practice of using the right eigenvectors to evaluate the matrix exponential combined with the left eigenvectors to avoid the inversion of the matrix of the right ones. Two numerical tests for Rayleigh and Aerosol scattering confirm our idea. For better explanation of our approach and for independent crosscheck of our results, we distribute an example in C/C++.

Key words: matrix exponential, left eigenvectors, vector radiative transfer equation.

1. Introduction

In 2015, our Fortran 90/95 radiative transfer (RT) code IPOL, that simulates multiple scattering of Intensity and POLarization of the monochromatic solar radiation in a plane-parallel atmosphere, participated in a comprehensive polarized (vector) RT codes intercomparison and confirmed high accuracy (*Emde* et al., 2015). Since then, we use IPOL to account for the effect of polarization of light in the Multi-Angle Implementation of Atmospheric Correction (MAIAC) algorithm (*Lyapustin* et al, 2018) and as a benchmark for our Successive ORDers RT code SORD (*Korkin* et al., 2017). Over time, we realized the need for further development of IPOL. This includes, but not limited to, the speed-up of the code to accelerate our research, and the translation from Fortran 90/95 into C/C++ for natural integration with the algorithm MAIAC.

[1] Corresponding author: sergey.v.korkin@nasa.gov

Similar to RT code Pstar (*Nakajima* and *Tanaka*, 1986; *Ota* et al., 2010), IPOL simultaneously uses the method of discrete ordinates (*Chandrasekhar*, 1950) and the matrix-operator method (*Plass* et al., 1973). In such RT codes, evaluation of the matrix exponential is a key and the most time consuming part. For solution to that problem, the eigendecomposition is "*likely to be most efficient for problems involving large matrices and repeated evaluation*" of the matrix exponential (*Moler* and *Van Loan*, 2003: p.20). The eigendecomposition is a widely used linear algebra technique. We refer the reader to the Wolfram MathWorld's "Eigen Decomposition"[2] or the Wikipedia's "Egendecomposition of a matrix"[3] (note different spelling) articles for in-depth reading. Later in Section 2, we use only a few relations from these articles.

In IPOL as a full vector (including circular polarization) discrete ordinates RT code, the eigendecomposition problem requires complex numbers (*Siewert*, 2000; *Spurr*, 2008, *Rozanov* et al., 2014). However, in the Earth atmosphere, the degree of circular polarization is negligibly small (*Kawata*, 1978; *Lenoble* et al., 2007: p.483). Modern space (*Deschamps* et al, 1994; *Mishchenko* et al, 2007; *Liu* and *Diner*, 2017; *Fougnie* et al, 2018; *Li* et al., 2018) and ground (*Li* et al., 2014) passive polarimetres are not designed to measure the circular component. If it is dropped from numerical simulations, the vector RT codes deal with a 3 x 3 phase matrix and the eigendecomposition problem requires only real arithmetic, similar to the scalar case (*Spurr*, 2008: Section 7.3.1.3; *Ota* et al, 2010: p.885). Our experience also indicates that it is safe to skip the circular polarized light and still get accurate results for the linear polarization components [*Q*, *U*], not to mention the total intensity *I*, regardless the atmospheric conditions (*Korkin* et al, 2017). Moreover, "*the order of the algebraic eigenvalue problem can be reduced by a factor of 2*" (*Stamnes* and *Swanson*, 1981: Abstract). This reduction tremendously speeds up the simulations.

Naturally, we decided to skip the circular polarization while translating IPOL from Fortran into C/C++. When doing that, we noticed a way to use the eigendecomposition technique in a slightly more convenient manner, than it is usually done. This paper describes all necessary details of our approach, which seems to be new. We aim to use as few equations as possible without sacrificing clarity. To achieve this goal, we *(a)* developed and distribute a C/C++ source code that

[2] http://mathworld.wolfram.com/EigenDecomposition.html (accessed 01/01/2019)
[3] https://en.wikipedia.org/wiki/Eigendecomposition_of_a_matrix (accessed 01/01/2019)

implements our approach, and *(b)* explicitly indicate equation numbers or, at worst, section or page number in references. The next Section 2 discusses theoretical background. In Section 3, we confirm our approach using two numerical examples and explain how these results can be reproduced with our C/C++ sample code. We conclude the paper with a summary.

2. Theoretical background

Solution to the RT equation using the method of discrete ordinates involves several main steps: expansion of the phase function or matrix over corresponding polynomials; the use of the *Fourier* series to decouple the dependence of the radiation field on azimuth; discretization of the scattering integral over cosine of the zenith angle μ using Gauss quadrature of an order 2*N*, where *N* is the number of ordinates per hemisphere. For the 3 components of the Stokes vector per ordinate, these steps give a system of 6*N* differential equations over optical depth τ for each *Fourier* harmonic *m*. All systems are independent and identical. Further, we skip the *Fourier* index *m*. Numerous publications discuss these steps in details – see e.g. *Lenoble* (1985), *Thomas* and *Stamnes* (1999), *Hovenier* et al. (2004).

In this paper, we immediately start with integration of the system of differential equations over τ. For simplicity, we assume a homogeneous non-emitting atmosphere of the total optical depth τ_0, and seek for a solution at 2*N* Gauss nodes. One computes the radiation field at arbitrary μ from the solution at the Gauss nodes using integration of the source function or the dummy node techniques (*Chalhoub* and *Siewert*, 2000). In equations below, we use the lower case bold letters, e.g. **α**, for a 3*N* x 3*N* matrices; "→" for column vectors of 3*N* elements; and the capital bold letters, e.g. **B**, for 6*N* x 6*N* matrices. Radiation reflected from top and transmitted through bottom of the atmosphere, $\vec{\mathbf{i}}_-(\mu<0,\,\tau=0)$ and $\vec{\mathbf{i}}_+(\mu>0,\ \tau=\tau_0)$ respectively, satisfy a system of equations (*Karp* et al, 1980: Eq.(8)):

$$-\begin{bmatrix}\vec{\mathbf{i}}_-(0)\\ \vec{\mathbf{i}}_+(0)\end{bmatrix}+\exp(\mathbf{B}\tau_0)\begin{bmatrix}\vec{\mathbf{i}}_-(\tau_0)\\ \vec{\mathbf{i}}_+(\tau_0)\end{bmatrix}=\int_0^{\tau_0}\exp\left((\mathbf{B}-\mathbf{1}/\mu_0)\tau\right)d\tau\begin{bmatrix}\vec{\mathbf{f}}_-(\mu_0)\\ \vec{\mathbf{f}}_+(\mu_0)\end{bmatrix}. \qquad (1)$$

In Eq.(1), $\vec{\mathbf{i}}_+(\mu>0,\,\tau=0)$ and $\vec{\mathbf{i}}_-(\mu<0,\,\tau=\tau_0)$ are the boundary conditions; **B** simulates multiple scattering from all Gauss directions to a given direction; **1** is a unit matrix. **B** depends on the

single scattering albedo ω_0 and a 3 x 3 scattering matrix expansion moments (*Hovenier* et al., 2004: Section 2.8). Without circular polarization, **B** has the following form (*Stamnes* and *Swanson*, 1981: Section 3)

$$\mathbf{B} = \begin{bmatrix} -\boldsymbol{\alpha} & \boldsymbol{\beta} \\ -\boldsymbol{\beta} & \boldsymbol{\alpha} \end{bmatrix} . \tag{2}$$

Note, that the sign at **β** in our Eq.(2) differs from the one in *Stamnes* and *Swanson* (1980) due to opposite definition of the positive μ direction. In the right hand side of Eq.(1), the vectors $\vec{\mathbf{f}}_-(\mu_0)$ and $\vec{\mathbf{f}}_+(\mu_0)$ come from the single scattering of the direct solar beam; μ_0 is the cosine of the solar zenith angle.

In the method of spherical harmonics, *Karp* et al. (1980: p.394) decomposed the matrix exponential using the right eigenvectors of the system matrix **B**, Eq.(2):

$$\exp(\mathbf{B}\tau) = \mathbf{U}_R \exp(\boldsymbol{\Gamma}\tau)\, \mathbf{U}_R^{-1} , \tag{3}$$

where **Γ** is a diagonal matrix of eigenvalues, known to occur in pairs with equal magnitude and different sign (*Stamnes* and *Swanson*, 1981: Eq.(12)) except for a conservative case, ω_0 =1.0, in which we are not practically interested. The columns of $\mathbf{U}_R$ are the corresponding right eigenvectors. Further, *Karp* et al. (1980: Eq.(12)) applied $\mathbf{S}\mathbf{U}_R^{-1}$ to both sides of Eq.(1)

$$-\mathbf{S}\mathbf{U}_R^{-1}\begin{bmatrix} \vec{\mathbf{i}}_-(0) \\ \vec{\mathbf{i}}_+(0) \end{bmatrix} + \mathbf{H}\mathbf{U}_R^{-1}\begin{bmatrix} \vec{\mathbf{i}}_-(\tau_0) \\ \vec{\mathbf{i}}_+(\tau_0) \end{bmatrix} = \mathbf{S}\int_0^{\tau_0} \exp\left((\boldsymbol{\Gamma} - \mathbf{1}/\mu_0)\tau\right) d\tau\, \mathbf{U}_R^{-1}\begin{bmatrix} \vec{\mathbf{f}}_-(\mu_0) \\ \vec{\mathbf{f}}_+(\mu_0) \end{bmatrix} , \tag{4}$$

where **S** is a diagonal scaling transformation matrix

$$\mathbf{S} = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \exp(-\boldsymbol{\gamma}\tau) \end{bmatrix} \tag{5}$$

that removes exponents with positive eigenvalues **γ**, and $\mathbf{H} = \mathbf{S}\exp(\boldsymbol{\Gamma}\tau_0)$. A particular form of **S** differs, depending on the sequence of eigenvalues. These same steps are used in the methods of

spherical harmonics (*Karp* et al., 1980: Eq.(8); *Lyapustin* et al, 2010: Eq.(6.15)) and discrete ordinates (*Budak* et al., 2015: Eqs.(7)-(9)).

The explicit matrix inversion, $\mathbf{U}_R^{-1}$, is avoided by using the transposed matrix of left eigenvectors, $\mathbf{U}_L^T$ (*Waterman*, 1981; *Ota* et al, 2010: Eq.(65); *Efremenko* et al, 2017: Eq.(39)), and the following relations known from linear algebra (e.g., see the Wolfram MathWorld's "Eigenvector"[4])

$$\mathbf{U}_R^{-1} = \mathbf{U}_L^T, \quad \mathbf{\Gamma}_R = \mathbf{\Gamma}_L = \mathbf{\Gamma}, \quad \mathbf{U}_L(\mathbf{B}) = \mathbf{U}_R(\mathbf{B}^T). \tag{6}$$

In Eq.(6), $\mathbf{U}_L(\mathbf{B})$ means "left eigenvalues of matrix **B**". *Efremenko* et al. (2013: Abstract) reported that the use of the left eigenvectors instead of the matrix inversion accelerates computations by about 15%.

The use of both left and right eigenvectors causes three problems. First and foremost, the left and the right eigenvectors must be properly normalized (*Efremenko* et al. 2017: Eqs.(32)-(33)). The normalization complicates the developing, debugging and support of the RT code. It also utilizes slow numerical operations, e.g. square root (although, it is of course not a bottleneck in RT codes). Second, the left and right eigenvectors are computed from the direct and transposed matrix **B**, respectively – see Eq.(6). One has to either compute the two, or use a transpose operation, which prevents from efficient storage of a square matrix in memory. Finally, the RT theory is complicated enough – the use of both right and left eigenvectors adds more complexity.

Instead of using both left and right eigenvectors, we can use only left eigenvectors and the following representation of the system matrix **B** in Eq.(1)

$$\exp(\mathbf{B}\tau) = \mathbf{U}_L^{-1} \exp(\mathbf{\Gamma}\tau)\, \mathbf{U}_L \ . \tag{7}$$

The scaling transformation $\mathbf{S}\mathbf{U}_L$ yields for Eq.(1)

[4] http://mathworld.wolfram.com/Eigenvector.html (accessed 01/01/2019)

$$-\mathbf{S}\mathbf{U}_{\mathrm{L}}\begin{bmatrix}\vec{\mathbf{i}}_{-}(0)\\ \vec{\mathbf{i}}_{+}(0)\end{bmatrix}+\mathbf{H}\mathbf{U}_{\mathrm{L}}\begin{bmatrix}\vec{\mathbf{i}}_{-}(\tau_0)\\ \vec{\mathbf{i}}_{+}(\tau_0)\end{bmatrix}=\mathbf{S}\int_0^{\tau_0}\exp\left((\boldsymbol{\Gamma}-\mathbf{1}/\mu_0)\tau\right)d\tau\,\mathbf{U}_{\mathrm{L}}\begin{bmatrix}\vec{\mathbf{f}}_{-}(\mu_0)\\ \vec{\mathbf{f}}_{+}(\mu_0)\end{bmatrix}. \tag{8}$$

Eq.(8) does not use the inverse matrix and the right eigenvectors at all and normalization of $\mathbf{U}_{\mathrm{L}}$ is not important. Given that the transpose of the system matrix does not change its symmetry

$$\mathbf{B}^{\mathrm{T}}=\begin{bmatrix}-\boldsymbol{\alpha}^{\mathrm{T}} & -\boldsymbol{\beta}^{\mathrm{T}}\\ \boldsymbol{\beta}^{\mathrm{T}} & \boldsymbol{\alpha}^{\mathrm{T}}\end{bmatrix}, \tag{9}$$

it is possible to efficiently solve the eigenproblem following *Stamnes* and *Swanson* (1981: Section 3). In particular, the matrix of the left eigenvectors is known to be symmetric (*Efremenko* et al., 2017: Eq.(37)):

$$\mathbf{U}_{\mathrm{L}}=\begin{bmatrix}\mathbf{u}_1 & \mathbf{u}_2\\ \mathbf{u}_2 & \mathbf{u}_1\end{bmatrix}, \tag{10}$$

For numerical test of our idea, we developed a C/C++ code that solves Eq.(8). In the next section, we show two examples and describe how to reproduce our results.

3. Numerical results and discussion

In this section, we show results for two numerical simulations, Rayleigh (*Emde* et al., 2015: Case A1-1) and Aerosol (*Wauben* and *Hovenier*, 1992: Model 1 – prolate spheroids). We use our full vector RT code IPOL as a benchmark. For simplicity, we consider only homogeneous atmospheres over black surface and compute solution at $2N = 16$ double Gauss nodes (*Sykes*, 1951), $N = 8$ nodes per hemisphere. In the Rayleigh scenario, we replace the true conservative case $\omega_0 = 1.0$ with a "pseudo-conservative" $\omega_0 = 0.99999999$ (eight digits) which is a widely used practice (*Stamnes* et al., 1988; *Lyapustin* et al., 2010). In the Aerosol scenario, we compute only azimuthally averaged Stokes vector (the *Fourier* harmonic $m = 0$) similar to *Lyapustin* and *Rozanov* (2010) which makes $U = 0$ in that scenario.

Tables 1 and 2 summarize the results. Table 1, left to right, defines the input parameters: name of the case; cosine of the solar zenith angle μ_0; the total optical thickness of the atmosphere τ_0; the

single scattering albedo ω_0; the total number of the phase matrix expansion moments K; the number of the Gauss nodes per hemisphere N; the peak value of the phase function $p(0^o)$; the average scattering cosine g; and a reference for each scenario. Table 2 shows numerical results generated by the C/C++ prototype of IPOL for both cases. We found perfect agreement in all the indicated digits, except for only 1 digit, marked in bold, which differs by 1 unit. Figures 1 and 2 show the comparison for the total intensity I, and the Q-component versa the zenith angle.

Independent reproducibility is often omitted nowadays (*Fomel* and *Claerbout*, 2009; *Peng*, 2011; *LeVeque* et al., 2012). In order to maintain this essential criterion of scientific research, we developed and publicly distribute a C/C++ example code that reproduces our results. It is available from GitHub[5] or by email request from the corresponding author. We link the code against the freeware C/C++ GNU Scientific Library[6], v.2.5. The package includes the following six files (five *.cpp source codes and one *.txt) zipped in `jqsrt_2019a.zip`:

`domm0.cpp` – is a subroutine that implements equations from Section 2 using the method of discrete ordinates (`dom`) for a *Fourier* harmonic `m=0`. We emphasize, `domm0` is only a prototype subroutine: it does not compute quantities at arbitrary μ, it was not yet tested for an atmosphere with many layers (although a loop over layers is included), it defines excessive arrays to compute the U-component for $m > 0$, it does not include single scattering correction or reflecting surface. The sole purpose of `domm0` is to reproduce the reported results.

The `domm0` subroutine does not compute **B**, Eq.(2), neither it computes $\mathbf{B}^T$, Eq.(9). *Stamnes* and *Swanson* (1981: Eq.(12)), formulate and solve the half-space eigenproblem using $(\boldsymbol{\alpha} - \boldsymbol{\beta})$ and $(\boldsymbol{\alpha} + \boldsymbol{\beta})$. We do the same, except for the transposed matrices: $(\boldsymbol{\alpha}^T - \boldsymbol{\beta}^T)$ and $(\boldsymbol{\alpha}^T + \boldsymbol{\beta}^T)$. For efficiency, we create all the matrices already transposed instead of explicitly call the GSL transpose function `gsl_matrix_transpose()`. Further, we note $(\boldsymbol{\alpha}^T - \boldsymbol{\beta}^T)$ uses only even orders of the expansion coefficients, while $(\boldsymbol{\alpha}^T + \boldsymbol{\beta}^T)$ – only odd. This explains the requirement for the `domm0` input: the number of the used expansion moments `nk` must be odd with a minimum value of 3 for Rayleigh scattering. Thus, the considered Aerosol case uses `nk=2*N+1=17` expansion

[5] https://github.com/korkins/jqsrt_2019a (accessed 01/01/2019)
[6] https://www.gnu.org/software/gsl/ (accessed 01/01/2019)

moments out of 34 total. For an adequate comparison, the single scattering correction was switched off in the Fortran version of IPOL.

Another benefit of our approach, Eq.(7), specifically related to the C/C++ language, is that the left eigenvectors are the row vectors. Thus the matrix of left eigenvectors, Eq.(10), is located in memory in a native C/C++ row-major order. In Fortran, $\mathbf{U}_L$ must be transposed for efficiency.

`gauszw.cpp`, `rayxk.cpp`, and `read2d.cpp` – compute the Gauss nodes, Rayleigh scattering matrix expansion moments for the first test, and read in a txt file with the Aerosol expansion moments for the second test, respectively. They are thoroughly commented and should be self-explanatory;

`test_domm0.cpp` is the main program. It creates input for both cases, calls `domm0.cpp` and other subroutines, and prints out the result on the screen;

Finally, `Xk0036_0695.txt` is an ASCII file with the expansion moments for the Aerosol case. In that file, the columns left to right are: k – the number of the expansion moment starting from 0 followed by 6 columns for expansion moments for the elements of the scattering matrix [a_1 a_2 a_3 a_4 b_1 b_2] - a_4 and b_2 are not used (*Hovenier* et al, 2004: Appendix C).

A natural question is the gain of speed. Since we do not compute the right eigenvectors at all and do not transpose matrices, we definitely anticipate a somewhat higher gain in performance than the 15% reported earlier (*Efremenko* et al, 2017: Abstrtact). But a detailed performance analysis is problematic because we have never developed a “slow” version that use both left and right eigenvectors. Neither we can use our Fortran 90/95 version of IPOL for a quick check, because it computes the full Stokes vector using complex arithmetic, do not rely on the solution of the eigenproblem in a half-space, and hence very time consuming. However, we believe the main advantage of our approach lies in simplification of theoretical background and coding.

Summary

We apply only left eigenvectors with the *Karp’s* et al. (1980) scaling transformation to evaluate the matrix exponential in a plane-parallel vector radiative transfer equation. This is contrary to a

widely used right eigenvectors approach, which either requires an explicit inversion of a matrix of the right eigenvectors (one per each optical layer) or the combined use of the right and left eigenvectors to avoid this inversion. Our approach avoids complications related to the proper normalization when both the left and right eigenvectors are used, and hence makes an RT code easier to develop and debug. It also uses the known half-space eigenproblem and hence applicable for efficient numerical simulation of unpolarized or linear polarized light scattering with the methods of discrete ordinates or spherical harmonics. Due to the left eigenvectors are essentially row vectors, the implementation of our approach in C/C++ is preferable. We demonstrate two numerical examples and offer a GNU Scientific Library-dependent C/C++ source code, https://github.com/korkins/jqsrt_2019a, for independent crosscheck of the reported results. We presented numerical results for azimuthally averaged quantities only. However, for cases with azimuthal dependence and non-zero *U*-component of the Stokes vector, the *T*-polynomials possess the same symmetry with respect to ±μ as the *Legendre* and *R*-polynomials, involved in our tests (*Hovenier* et al, 2004: Appendix B).

Acknowledgements

The research of *Alexei Lyapustin* and *Sergey Korkin* was funded by NASA Science for Terra, Aqua and SNPP (17-TASNPP17-0116; solicitation NNH17ZDA001NTASNPP).

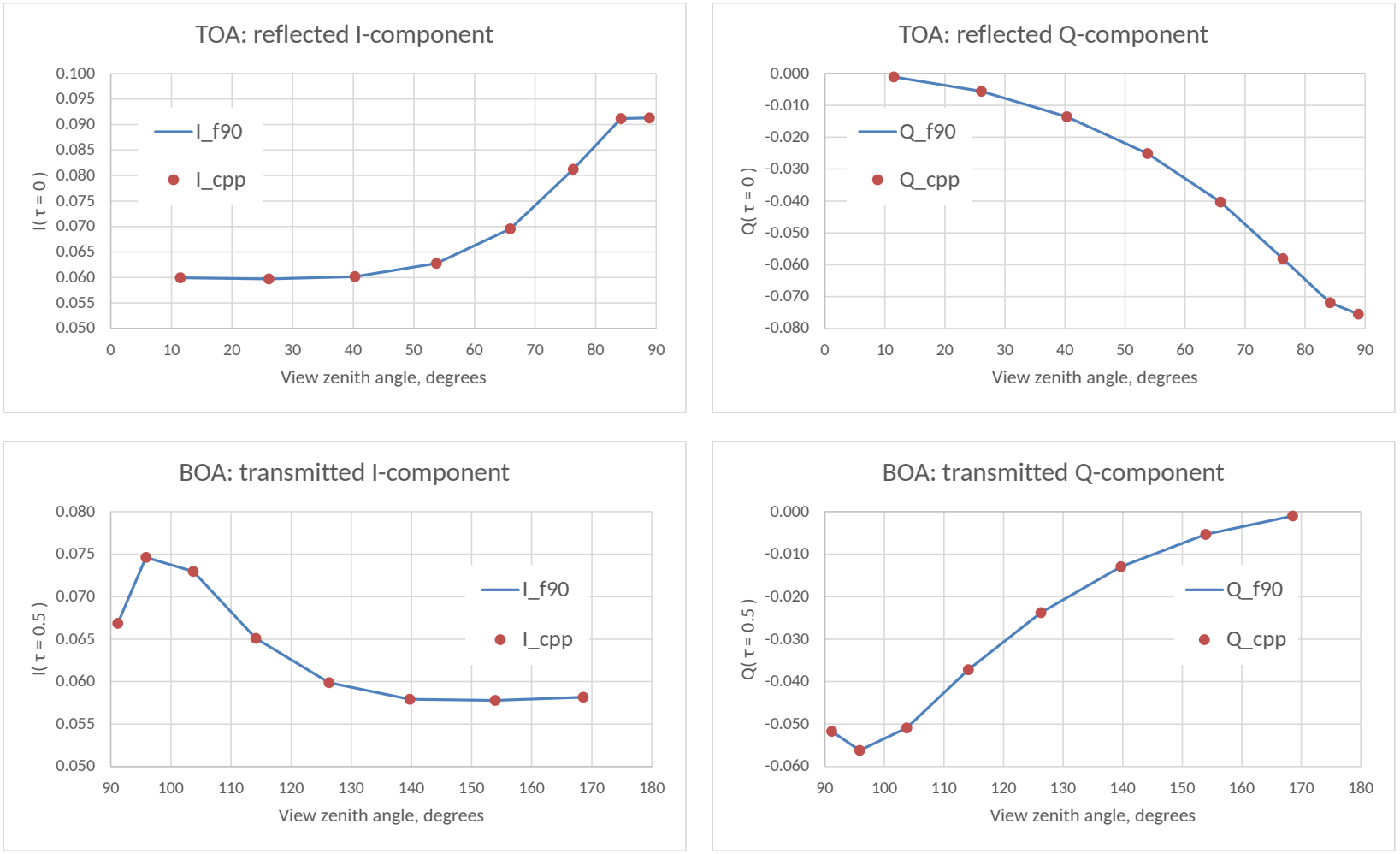

Figure 1: *I* and *Q* components on top (TOA, $\tau = 0$) and bottom (BOA, $\tau = \tau_0$) of atmosphere for the Rayleigh case at Gauss nodes. Dot markers – C/C++, blue line – Fortran. See Table 1 for input.

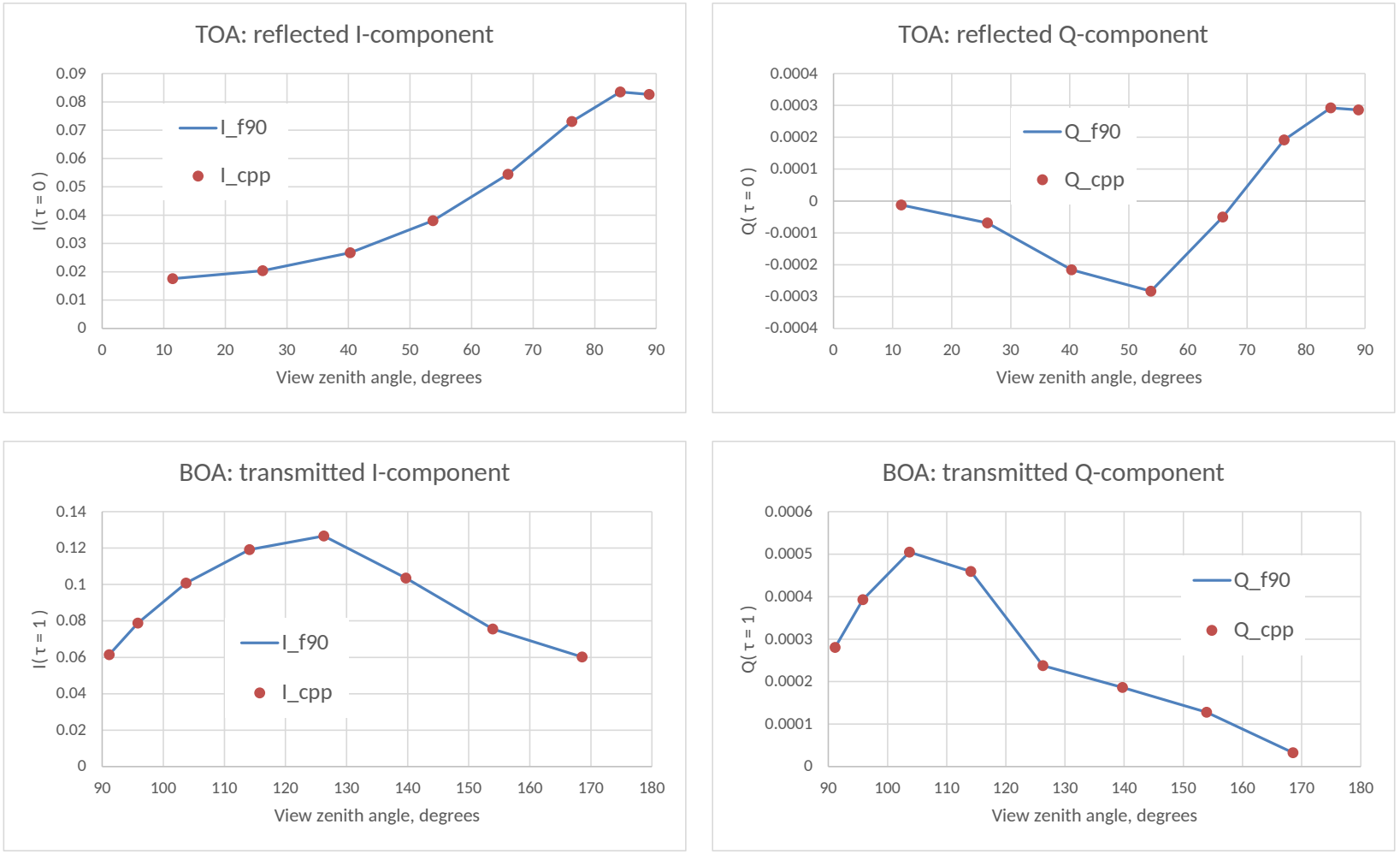

Figure 2: Same as Figure 1 except for the Aerosol case. See Table 1 for input.

Table 1: Input parameters for numerical tests. See Section 3 for details.

Case	μ_0	τ_0	ω_0	K	N	$p(0^o)$	g	Reference
Rayleigh	1	0.5	1.0	3	8	1.5	0	*Emde* et al. (2015: Case A1-1)
Aerosol	0.6	1	0.95	36	8	16	0.695	*Wauben* and *Hovenier* (1992: Model 1)

Table 2: Numerical results for the C/C++ code. All digits agree with the Fortran computations, except for the last digit in *I* (μ=0.1017), Rayleigh case, differs by one unit – marked red. Negative and positive Gauss nodes, μ, correspond to radiation at top (reflected) and bottom (transmitted) of atmosphere, respectively. See Table 1 for input.

Case:	**Rayleigh**		**Aerosol**	
μ	***I***	***Q***	***I***	***Q***
-0.0199	9.13232E-02	-7.54817E-02	8.26495E-02	2.85711E-04
-0.1017	9.11630E-02	-7.19422E-02	8.34655E-02	2.92456E-04
-0.2372	8.12421E-02	-5.80813E-02	7.30460E-02	1.91619E-04
-0.4083	6.95400E-02	-4.02927E-02	5.44124E-02	-5.04486E-05
-0.5917	6.27695E-02	-2.51031E-02	3.80289E-02	-2.82670E-04
-0.7628	6.01544E-02	-1.35367E-02	2.67002E-02	-2.15704E-04
-0.8983	5.97108E-02	-5.52871E-03	2.03950E-02	-6.85303E-05
-0.9801	5.99419E-02	-1.05280E-03	1.75805E-02	-1.22715E-05
0.0199	6.68614E-02	-5.16898E-02	6.13960E-02	2.80410E-04
0.1017	7.4613**3**E-02	-5.62263E-02	7.87672E-02	3.93080E-04
0.2372	7.29682E-02	-5.09112E-02	1.00791E-01	5.05020E-04
0.4083	6.50852E-02	-3.71982E-02	1.19207E-01	4.59117E-04
0.5917	5.98697E-02	-2.37405E-02	1.26586E-01	2.37564E-04
0.7628	5.79301E-02	-1.29607E-02	1.03563E-01	1.86095E-04
0.8983	5.77966E-02	-5.32793E-03	7.55802E-02	1.28067E-04
0.9801	5.81608E-02	-1.01767E-03	6.02075E-02	3.24825E-05

References

1. Budak V.P., Kaloshin G.A., Shagalov O.V. and Zheltov V.S, 2015: Numerical modeling of the radiative transfer in a turbid medium using the synthetic iteration, *Optic Express*, **23** (15), p.A829-A840;
2. Chalhoub E.S., Garcia R.D.M., 2000: The equivalence between two techniques of angular interpolation for the discrete-ordinates method, *Journal of Quantitative Spectroscopy and Radiative Transfer*, **64**, p.517-535;
3. Chandrasekhar S., 1950: *Radiative transfer*. London: Oxford University Press;
4. Deschamps P., Breon F., Leroy M., Podaire A., Bricaud A., Buriez J., and Seze G, 1994: The POLDER mission: instrument characteristics and scientific objectives, *IEEE Transactions on Geoscience Remote Sensing*, **32** (3), 598-615;
5. Efremenko D.S., Doicu A., Loyola D., and Trautmann T., 2013: Acceleration techniques for the discrete ordinate method, *Journal of Quantitative Spectroscopy and Radiative Transfer*, **114**, p.73–81;
6. Efremenko D.S., Garcia V.M., Garcia S.G., and Doicu A., 2017: A review of the matrix-exponential formalism in radiative transfer, *Journal of Quantitative Spectroscopy and Radiative Transfer*, **196**, p.17-45;
7. Emde C., Barlakas V., Cornet C., Evans F., Korkin S., Ota Y., C.-Labonnote L., Lyapustin A., Macke A., Mayer B., and Wendisch M., 2015: IPRT polarized radiative transfer model intercomparison project – Phase A, *Journal of Quantitative Spectroscopy and Radiative Transfer,* **164**, p.8-36.
8. Fomel S., Claerbout J.F., 2009: Reproducible research, *Computing in Science and Engineering*, **11** (5), p.5-7;
9. Fougnie B., Marbach T., Lacan A., Lang R., Schlüssel P., Poli G., Munro R., Couto A., 2018: The multi-viewing multi-channel multi-polarisation imager –Overview of the 3MI polarimetric mission for aerosol and cloud characterization, *Journal of Quantitative Spectroscopy and Radiative Transfer*, **219**, p.23-32;
10. Hovenier J.W., van der Mee C., and Domke H., 2004: *Transfer of polarized light in planetary atmospheres. Basic concepts and practical methods*, Kluwer Academic Publishers, Dordrecht, Netherlands;
11. Karp A.H., Greenstadt J., and Fillmore J.A., 1980: Radiative transfer through an arbitrary thick scattering atmosphere, *Journal of Quantitative Spectroscopy and Radiative Transfer*, **24** (5), p.391 – 406;
12. Kawata Y., 1978: Circular polarization of sunlight reflected by planetary atmospheres, *Icarus*, **33**, p.217–232.

13. Korkin S., Lyapustin A., Sinyuk A., Holben B., and Kokhanovsky A., 2017: Vector radiative transfer code SORD: performance analysis and quick start guide, *Journal of Quantitative Spectroscopy and Radiative Transfer*, **200**, p.295-310;

14. Lenoble, J. (Ed.), 1985: *Radiative Transfer in Scattering and Absorbing Atmospheres: Standard Computational Procedures*, A. Deepak Publishing, Hampton VA;

15. Lenoble J., Herman M., Deuzé J L., Lafrance B., Santer R., and Tanré, D., 2007: A successive order of scattering code for solving the vector equation of transfer in the earth's atmosphere with aerosols, *Journal of Quantitative Spectroscopy and Radiative Transfer*, **107**, p.479-507;

16. LeVeque R.J., Mitchell I.M., and Stodden V., 2012: Reproducible research for scientific computing: tools and strategies for changing the culture, Computing in Science and Engineering, **14** (13), p.13-17;

17. Li L., Li Z., Li K., Blarel L., and Wendisch M, 2014: A method to calculate Stokes parameters and angle of polarization of sky light frompolarized CIMEL sun/sky radiometers, *Journal of Quantitative Spectroscopy and Radiative Transfer,* **149**, p.334-346;

18. Li Z., Hou W., Hong J., Zheng F., Luo D., Wang J., Gu X., and Qiao Y., 2018: Directional Polarimetric Camera (DPC): Monitoring aerosol spectral optical properties over land from satellite observation, *Journal of Quantitative Spectroscopy and Radiative Transfer,* **218**, p.21-37;

19. Liu Y., Diner D., 2017: Multi-Angle Imager for Aerosols: A Satellite Investigation to Benefit Public Health, *Public Health Reports*, **132** (1), p.14-17;

20. Lyapustin A., Muldashev T., and Wang Y., 2010: Code SHARM: fast and accurate radiative transfer over spatially variable anisotropic surfaces. In: Kokhanovsky A.A. (Ed.) *Light scattering reviews 5. Single light scattering and radiative transfer*, Chichester: Springer, p.205–247;

21. Lyapustin A., Wang Y., Korkin S., and Huang D., 2018: MODIS collection 6 MAIAC algorithm, *Atmospheric Measurement Techniques*, **11**, p.5741-5765;

22. Mishchenko M., Cairns B., Kopp G., Schueler C., Fafaul D., Hansen J., Hooker R., Itchkawich T., Maring H., and Travis L., 2007: Accurate monitoring of terrestrial aerosols and total solar irradiance: Introducing the Glory mission, *Bulletin of American Meteorological Society*, **88**, p.677-691;

23. Moler C.B., van Loan C.F., 2003: Nineteen Dubious Ways to Compute the Exponential of a Matrix, Twenty-Five Years Later, *SIAM Review* 45, p.3–49;

24. Nakajima T., Tanaka M., 1986: Matrix formulations for the transfer of solar radiation in a plane-parallel scattering atmosphere, *Journal of Quantitative Spectroscopy and Radiative Transfer*, **35** (1), p.13-21;

25. Ota Y., Higurashi A., Nakajima T., Yokota T., 2010: Matrix formulations of radiative transfer including the polarization effect in a coupled atmosphere–ocean system, *Journal of Quantitative Spectroscopy and Radiative Transfer*, **111**, p.878–894;

26. Peng R.D., 2011: Reproducible research in computational science, *Science*, **334** (6060) p.1226-1227;

27. Plass G.N., Kattawar G.W., and Catchings F.E., 1973: Matrix operator theory of radiative transfer. 1: Rayleigh scattering, *Applied Optics*, **12**, p.314–329;

28. Rozanov V.V., Lyapustin A.I., 2010: Similarity of radiative transfer equation: error analysis of phase function truncation techniques, *Journal of Quantitative Spectroscopy and Radiative Transfer*, **111**, 1964-1979;

29. Rozanov V.V., Rozanov A.V., Koklhanovsky A.A., and Burrows J.P., 2014: Radiative transfer through terrestrial atmosphere and ocean: software package SCIATRAN, *Journal of Quantitative Spectroscopy and Radiative Transfer*, **133**, p.13-71;

30. Spurr R.J.D., 2008: LIDORT and VLIDORT: Linearized pseudo-spherical scalar and vector discrete ordinate radiative transfer models for use in remote sensing retrieval problems. In: Kokhanovsky A.A. (Ed.) *Light scattering reviews 3. Light Scattering and Reflection*, Chichester: Springer, p.229–275;

31. Stamnes K., Swanson R.A., 1981: A New Look at the Discrete Ordinate Method for Radiative Transfer Calculations in Anisotropically Scattering Atmospheres, *Journal of the Atmospheric Science*, **38** (2), p.387–389;

32. Stamnes K., Tsay S.C., Wiscombe W., Jayaweera K., 1988: Numerically stable algorithm for discrete-ordinate-method radiative transfer in multiple scattering and emitting layered media, *Applied Optics*, **27** (12), p.2502-2509;

33. Siewert, C. E., 2000: A discrete-ordinates solution for radiative-transfer models that include polarization effects, *Journal of Quantitative Spectroscopy and Radiative Transfer*, **64**, p.227-254;

34. Sykes J.B., 1951: Approximate integration of the equation of transfer, *Month. Not. R. Astroph. Soc.*, **111**, p.378-386;

35. Thomas G.E., Stamnes K., 1999: *Radiative transfer in the atmosphere and ocean*, Cambridge University Press.

36. Waterman P.C., 1981: Matrix-exponential description of radiative transfer, *Journal of Optical Society of America*, **71** (4), p.410-422;

37. Wauben W. M. F., de Haan J. F., and Hovenier J.W., 1993: Low orders of scattering in plane-parallel homogeneous atmosphere, Astronomy and Astrophysics, **276**, p.589-602.