Sensitivities of the MSIS–86 Thermosphere Model

H. Martin Bücker and Andre Vehreschild
Institute for Scientific Computing
RWTH Aachen University
D-52056 Aachen, Germany
E-mail: {buecker, vehreschild}@sc.rwth-aachen.de

Keywords— **MSIS–86, automatic differentiation, sensitivities, thermosphere model**

Abstract— **Atmospheric models represent the state of the environment including interactions that occur in the atmosphere and are thus useful to predict future temperature distributions and concentration densities of important species. A prominent example of an atmospheric model, specialized for the thermosphere, is the MSIS–86 model developed at NASA. In this note, we describe transforming this computer program by means of the automatic differentiation tool ADIFOR to obtain a new computer program capable of evaluating the derivatives of the output of MSIS–86 with respect to its input. These sensitivities are quantified and compared with derivatives approximated by divided differencing, demonstrating the reliability of automatic differentiation.**

I. Introduction

The Earth's atmosphere is a thin layer of air that can roughly be specified as the region from the surface of the Earth to about 1000 km altitude around the globe. Within the atmosphere, very complex chemical, thermodynamic, and fluid dynamic effects occur. Below 50 km the atmosphere is typically treated as a perfect gas, assuming homogeneity. Above 80 km the atmosphere is no longer uniform since the hydrostatic equilibrium gradually breaks down as diffusion and vertical transport become important. The major species in the upper atmosphere are N_2, O, O_2, H, and He. Atmospheric reference models predicting temperature and concentration profiles of species were first developed in the early sixties based on theoretical considerations and satellite drag data. A prominent example of an atmospheric reference model is a suite of models known as the Mass Spectrometer Incoherent Scatter (MSIS) models [11] developed at NASA's Goddard Space Flight Center.

In this note, we are interested in the sensitivities of atmospheric model predictions with respect to model parameters. We regard this work as a first step in the direction of establishing a more realistic strategy to predict the flight trajectory of a satellite. The satellite model given in [15] considers celestial gravitation and earth tides while using a constant air density in its atmospheric drag submodel. The derivatives of the MSIS model whose generation is described in this note will be used in a future work to fit real-world GPS-data with simulations of the flight trajectory.

Rather than relying on approximations computed by divided differencing, the aim is to obtain sensitivities without truncation error. To this end, a technique commonly referred to as automatic differentiation (AD) [8], [17] is employed. More precisely, the AD tool ADIFOR [4] is used to transform the MSIS–86 model into a new program capable of computing user-selected sensitivities.

The structure of this note is as follows. In Section II, the atmospheric model MSIS is sketched. The AD tool ADIFOR and its underlying technology is briefly described in Section III. Numerical experiments are carried out in Section IV.

II. The Atmosphere Models MSIS

The suite of atmosphere models called Mass Spectrometer and Incoherent Scatter (MSIS) are based on directly applying analytical solutions of simplified diffusion equations to derive concentration profiles of major species. Originally developed by Hedin and co-workers in the seventies, the MSIS models were continuously improved as new measurement data and new analysis results became available: MSIS–77 [13], MSIS–83 [10], MSIS–86 [11], MSISE–90 [12], and NRLMSISE–00 [16].

In this note, the MSIS–86 model is used as a model for the thermosphere, the sheet of air above about 120 km. This version has been extensively used for comparing model predictions with measurements; see [1], [6], [14] for instance. The MSIS model is based on extensive insitu data measurements from numerous rocket probes and satellites (OGO 6, San Marco 3, AEROS-A, AE-C, AE-D, AE-E, ESRO 4, and DE 2) as well as ground-based incoherent scatter radars (Millstone Hill, St. Santin, Arecibo, Jicamarca, and Malvern).

As input, the MSIS–86 model expects year, day of year, Universal Time, altitude (a), geodetic latitude (δ) and longitude (λ), local apparent solar time (τ), solar F10.7 flux (for previous day and three-month average), and magnetic Ap index. As output, MSIS–86 computes densities of He, O, N_2, O_2, Ar, H, and N, total mass density as well as neutral temperature and exospheric temperature (T). The source code is written in Fortran 77 and consists of about 800 lines of code.

III. The AD Tool ADIFOR

The term automatic differentiation (AD) comprises a set of techniques for mechanically transforming a given computer program into a new program for the computation of derivatives. The transformation process is based on the standard rules for differentiating elementary mathematical operations. Combining these elementary derivatives by the chain rule of differential calculus finally yields the derivatives of the whole program. AD software tools automate this transformation; see `www.autodiff.org` for a list of AD

Proceedings 20th European Conference on Modelling and Simulation
Wolfgang Borutzky, Alessandra Orsoni, Richard Zobel © ECMS, 2006
ISBN 0-9553018-0-7 / ISBN 0-9553018-1-5 (CD)

tools for various programming languages. This way, AD requires little human effort and produces derivatives without truncation error. The reader is referred to the books [8], [17] and the proceedings of AD workshops [9], [2], [7], [5] for details on this technique.

Developed in a collaborative project between the Mathematics and Computer Science Division at Argonne National Laboratory and the Center for Research on Parallel Computation at Rice University, the ADIFOR system [3], [4] implements **A**utomatic **DI**fferentiation of **FOR**tran 77. ADIFOR provides full Fortran 77 support for first-order derivatives, e.g., it supports `common` blocks, `complex` arithmetic, arbitrary `function` and `subroutine` calling sequences as well as common extensions such as `include` statements, `double complex`, and `implicit none`. Furthermore, ADIFOR reports exceptions such as the differentiation of `sqrt(x)` when `x` is zero.

Transforming MSIS–86 using ADIFOR was a straightforward process without the need for additional modifications of the source code. For the numerical experiments reported in the following section, the number of directional derivatives was set to four. We used the exception handling system to check that no elementary function is evaluated at a point where the function is defined but its derivative is not. However, for the performance measurements, we turned off the exception handling by setting ADIFOR's option `AD_EXCEPTION_FLAVOR = performance`.

IV. Numerical Experiments

In this section, numerical experiments on a SunFire E2900 with a UltraSparcIV processor (1.2 GHz) and a total of 48 GB RAM are reported. All experiments are run under Solaris 9 using the Sun Fortran compiler, version 8.1.

In the experiments, we computed the derivatives of the ten scalar output variables representing eight densities and two temperatures with respect to the four input variables altitude (a), geodetic latitude (δ), and longitude (λ), and local apparent solar time (τ). These 40 derivatives agree reasonably well with approximations based on divided differences. To give the reader a more detailed picture, we consider the derivatives of the exospheric temperature, denoted by T, with respect to $[a, \delta, \lambda, \tau]^T$.

In a first set of experiments, we are interested in comparing the AD-computed derivatives with derivatives obtained from divided differencing (DD). In all experiments, first order forward differences with a step size h are used. To demonstrate the difference in accuracy between AD and DD, we formally define

$$\Delta(T,\delta,h) \;:=\; \left| \frac{\partial T(a,\delta,\lambda,\tau)}{\partial \delta} - \frac{T(a,\delta+h,\lambda,\tau) - T(a,\delta,\lambda,\tau)}{h} \right| , \quad (1)$$

where the first term on the right-hand side is the value computed by automatic differentiation and the second term is the value obtained from divided differences. Hence, $\Delta(T, \delta, h)$ is a measure of the difference of the numerical accuracy of the partial derivatives obtained from AD and DD. The definition (1) is extended to derivatives other than $\partial T/\partial\delta$ in a straight forward fashion. Furthermore, the definition

$$\Delta_{\text{rel}}(T,\delta,h) := \Delta(T,\delta,h) \Big/ \left| \frac{\partial T(a,\delta,\lambda,\tau)}{\partial \delta} \right| \quad (2)$$

is used to denote $\Delta(T, \delta, h)$ relative to the value computed by AD.

In Figure 1, three derivatives computed via AD are compared with those obtained by DD when varying the step size h. The derivatives are evaluated at

$$a = 485.811 \text{ km}\,,\ \delta = 66.148^\circ, \quad \lambda = -211.612^\circ\ \,,\ \tau = -14.196 \text{ h}.$$

In particular, the figure shows $\Delta_{\text{rel}}(T, \delta, h)$, $\Delta_{\text{rel}}(T, \lambda, h)$ and $\Delta_{\text{rel}}(T, \tau, h)$. While decreasing the step size from 10^{-2} to smaller values, the values of $\Delta_{\text{rel}}(T, \cdot, h)$ decrease, indicating that the truncation error of DD decreases. When decreasing the step size further, cancellation error in the numerator of the DD approximations becomes dominant and the values of $\Delta_{\text{rel}}(T, \cdot, h)$ increase again. According to these experiments, the "optimal" step size for $\Delta_{\text{rel}}(T, \lambda, h)$ $\Delta_{\text{rel}}(T, \delta, h)$ and $\Delta_{\text{rel}}(T, \tau, h)$ are approximately given by 10^{-5}, 10^{-6} and 10^{-7}, respectively. The derivative $\partial T/\partial a$ is omitted in this figure because it is (exactly) zero.

In the second set of numerical experiments, we are interested in the derivatives of the density of N_2, denoted by c, with respect to $[a, \delta, \lambda, \tau]^T$ when varying the four input variables a, δ, λ, and τ. From the large set of results, we chose to concentrate on variations of a and τ and note that the corresponding variations with respect to δ and λ are all well-behaved.

In Figure 2, these derivatives are plotted for different altitudes ranging from 436 km to 536 km. The figure shows a decrease of the three derivatives $\partial c/\partial[\delta, \lambda, \tau]$ with increasing altitude, whereas the derivative $\partial c/\partial a$ is increasing with increasing altitude. These four derivatives exhibit a smooth behavior in the chosen range. Periodic behavior of the derivatives is depicted in Figure 3, where the same four derivatives are plotted ranging the local solar time from -64 h to 36 h. In each of the Figures 2 and 3, one hundred equidistant points are used on the abscissa.

The performance of the AD-generated model depends on the number of directional derivatives propagated through the code. In the experiments reported here, four directional derivatives are computed. Compared to the original MSIS–86 model, its AD-generated version uses 2.7 times more storage and the run time is increased by a factor of about 4.3. Recall that approximations by divided differencing need at least 5 evaluations of the underlying function.

V. Concluding Remarks

The MSIS–86 atmospheric model provides estimates of temperature and concentration densities of important species in the upper atmosphere. By transforming MSIS–86 using the automatic differentiation tool ADIFOR, we generate another computer model providing the derivatives of the outputs of MSIS–86 with respect to its inputs. More precisely, the derivatives of the densities as well as of the

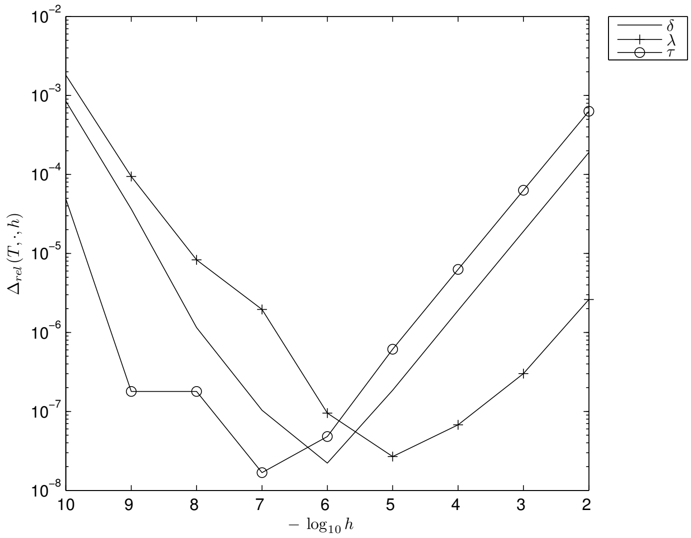

Fig. 1. Relative differences between AD and DD for $\Delta_{\text{rel}}(T, \delta, h)$, $\Delta_{\text{rel}}(T, \lambda, h)$, and $\Delta_{\text{rel}}(T, \tau, h)$

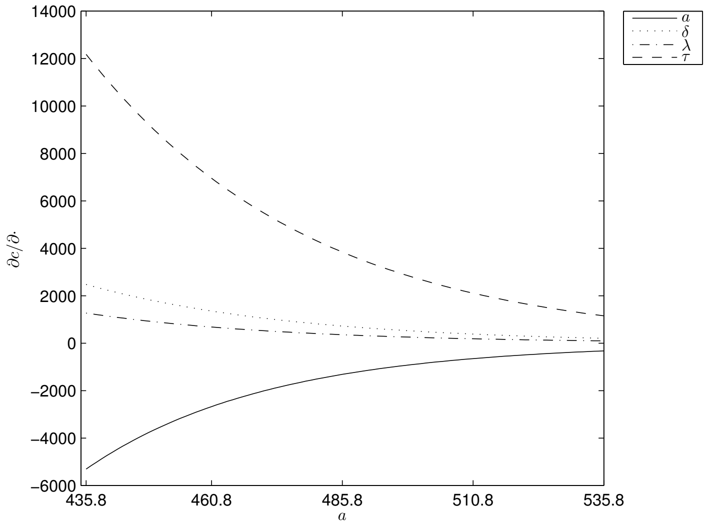

Fig. 2. Derivatives $\partial c/\partial[a, \delta, \lambda, \tau]$ for various altitudes a

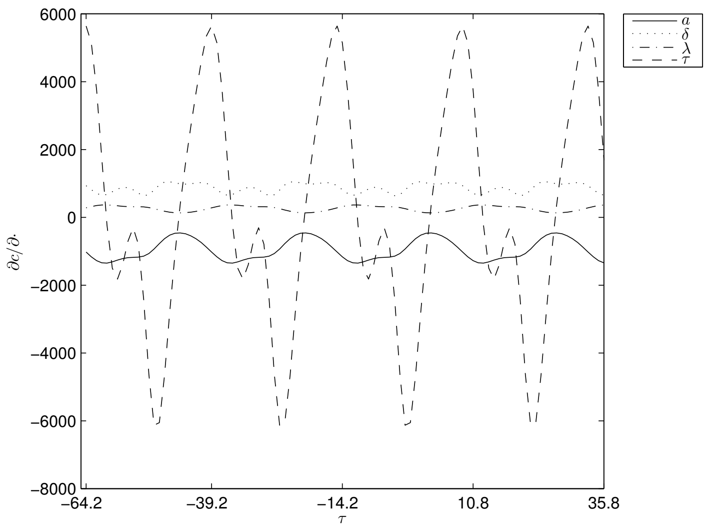

Fig. 3. Derivatives $\partial c/\partial[a, \delta, \lambda, \tau]$ for the local solar τ ranging from -64 h to 36 h

neutral temperature and exospheric temperature with respect to altitude, geodetic latitude and longitude, and local apparent solar time are computed. The crucial issue is that these derivatives are computed reliably without truncation error that would be introduced in any approach based on numerical differentiation. In the present AD-based approach, the additional cost in terms of storage and computing time is reasonably modest. Future research directions include the integration of the MSIS–86 model and its derivatives in a Kalman-filter for data fusion of simulated trajectory and GPS-data.

Acknowledgments

This work was stimulated by Marc Kalkuhl and Wolfgang Wiechert of the Department of Simulation, University of Siegen, Germany, by their interest in high precision satellite orbit simulations.

References

[1] A. C. Aikin, A. E. Hedin, D. J. Kendig, and S. Drake. Thermospheric molecular oxygen measurements using the ultraviolet spectrometer on the Solar Maximum Mission spacecraft. *Journal of Geophysical Research*, 98(A10):17,607–17,614, 1993.

[2] M. Berz, C. Bischof, G. Corliss, and A. Griewank, editors. *Computational Differentiation: Techniques, Applications, and Tools*. SIAM, Philadelphia, 1996.

[3] C. Bischof, A. Carle, G. Corliss, A. Griewank, and P. Hovland. ADIFOR: Generating derivative codes from Fortran programs. *Scientific Programming*, 1(1):11–29, 1992.

[4] C. Bischof, A. Carle, P. Khademi, and A. Mauer. ADIFOR 2.0: Automatic Differentiation of Fortran 77 Programs. *IEEE Computational Science & Engineering*, 3(3):18–32, 1996.

[5] H. M. Bücker, G. F. Corliss, P. D. Hovland, U. Naumann, and B. Norris, editors. *Automatic Differentiation: Applications, Theory, and Implementations*, volume 50 of *Lecture Notes in Computational Science and Engineering*. Springer, New York, 2005.

[6] M. J. Buonsanto and L. M. Pohlman. Climatology of neutral exospheric temperature above millstone hill. *Journal of Geophysical Research*, 103(A10):23,381–23,392, 1998.

[7] G. Corliss, C. Faure, A. Griewank, L. Hascoët, and U. Naumann, editors. *Automatic Differentiation of Algorithms: From Simulation to Optimization*. Springer, New York, 2002.

[8] A. Griewank. *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*. SIAM, Philadelphia, 2000.

[9] A. Griewank and G. Corliss. *Automatic Differentiation of Algorithms*. SIAM, Philadelphia, 1991.

[10] A. E. Hedin. A revised thermospheric model based on mass spectrometer and incoherent scatter data: MSIS–83. *Journal of Geophysical Research*, 88:10,170–10,188, 1983.

[11] A. E. Hedin. MSIS–86 thermospheric model. *Journal of Geophysical Research*, 92(A5):4649–4662, 1987.

[12] A. E. Hedin. Extension of the MSIS thermosphere model into the middle and lower atmosphere. *Journal of Geophysical Research*, 96(A2):1159–1172, 1991.

[13] A. E. Hedin, J. E. Salah, J. V. Evans, C. A. Reber, G. P. Newton, N. W. Spencer, D. C. Kayser, D. Alcayde, P. Bauer, L. Cogger, and J. P. McClure. A global thermospheric model based on mass spectrometer and incoherent scatter data MSIS 1. N_2 density and temperature. *Journal of Geophysical Research*, 82:2139–2147, 1977.

[14] A. E. Hedin and G. Thuillier. Comparison of OGO 6 measured thermospheric temperatures with the MSIS–86 empirical model. *Journal of Geophysical Research*, 93(A6):5965–5971, 1988.

[15] M. Kalkuhl, W. Wiechert, H. M. Bücker, and A. Vehreschild. High Precision Satellite Orbit Simulation: A Test Bench for Automatic Differentiation in MATLAB. In F. Hülsemann, M. Kowarschik, and U. Rüde, editors, *Proceedings of the Eighteenth Symposium on Simulation Techniques, ASIM 2005, Erlangen, September 12–15*, number 15 in Frontiers in Simulation, pages 428–433, Erlangen, 2005. SCS Publishing House.

[16] J. M. Picone, A. E. Hedin, D. P. Drob, and A. C. Aikin. NRLMSISE–00 empirical model of the atmosphere: Statistical comparisons and scientific issues. *Journal of Geophysical Research*, 107(A12):1468, 2002.

[17] L. B. Rall. *Automatic Differentiation: Techniques and Applications*, volume 120 of *Lecture Notes in Computer Science*. Springer Verlag, Berlin, 1981.

AUTHOR BIOGRAPHIES

H. MARTIN BÜCKER received the Dipl.-Ing. degree in electrical engineering, the Dipl.-Inform. degree in computer science, and the Ph.D. degree in electrical engineering from RWTH Aachen University in 1993, 1994, and 1997, respectively. He worked for the Central Institute for Applied Mathematics, Research Center Jülich, from 1993 to 1998, participating in the design of parallel algorithms for sparse matrix problems. Since 1998 he has been with the Institute for Scientific Computing, RWTH Aachen University, where he received his Habilitation in 2003. His current areas of research include high-performance computing, numerical linear algebra, and automatic differentiation.

ANDRE VEHRESCHILD received his Dipl.-Inform. in computer science from RWTH Aachen University in 2001. He is currently working as research assistant and Ph.D. student at the Institute for Scientific Computing, RWTH Aachen University. His research interests include parallel computing and automatic differentiation.