Remote Sensing

Operational Data Retrieval Method of the Satellite Sensor ILAS for Polar Stratospheric Ozone Monitoring

Contact Person — Senior Research Scientist, Tatsuya Yokota
Satellite Remote Sensing Research Team, Global Environment Division,
National Institute for Environmental Studies,
Environment Agency of Japan.

Contractor Researchers — Hiroshi. Tohno, Susumu Tahara, Yasuhiro Itoh, and Hirotomo Taniguchi
Fujitsu Facom Information Processing Corporation

Key Words — spectroscopy, data analysis, remote sensing, radiative transfer, algorithm, ILAS

1. Background

Satellite Remote Sensing Research Team is now performing a project to establish an operational data processing system (ODPS) for the satellite sensor ILAS. ILAS (Improved Limb Atmospheric Spectrometer) is designed to measure vertical profiles of high latitude stratospheric constituents such as O_3, CH_4, N_2O, HNO_3, H_2O, NO_2, and CFC-11 (from the IR channel), and vertical profiles of pressure, temperature, and aerosols (from the visible channel), in a solar occultation mode (Fig. 1), in order to provide data for ozone layer chemical and dynamic process studies. The ILAS instrument is manufactured by Matsushita Research Institute Tokyo Inc. under the direction of Environment Agency of Japan. Our team is in charge of development of the ILAS data analysis algorithms and the operational computer system. The platform satellite ADEOS (ADvanced Earth Observing Satellite) will be launched in early 1996. So, the system integration of the ILAS/ODPS will be defined in a year, and the system will be settled in two years.

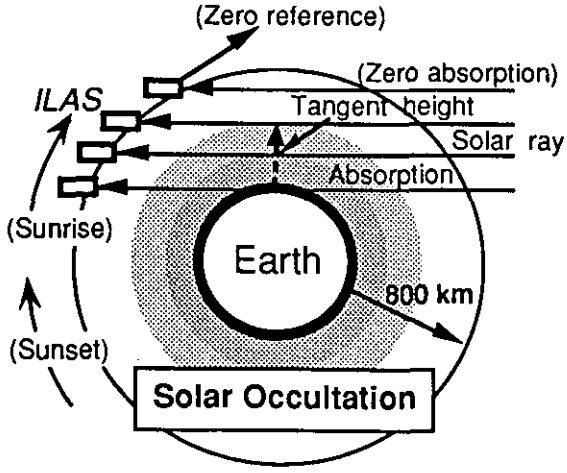

Fig. 1 Solar occultation geometry

2. Objective

We are now developing the ILAS operational software with contractor members of Fujitsu Facom Information Processing Corporation. In order to determine the operational algorithms, we are partly using the super computer SX-3 system for investigating the performance of the algorithm such as calculation speed, vectorization ratio, memory capacity, and disk file capacity.

This report describes the outline of the ILAS project and the objective of algorithm studies using the super computer in this project.

3. Outline of the ILAS[1,2)]

The platform ADEOS is developed by National Space Development Agency of Japan (NASDA) and is scheduled to be launched in early 1996 by the H-II rocket. ADEOS has a sun-synchronous sub-recurrent polar orbit with a 98.6 degree inclination. The equatorial crossing local time at descending node will be 10:15 ~ 10:45 a.m. The orbit altitude is 800 km, the period of one cycle is 101 minutes, and the recurrent period is 41 days. The design life time is 3 years.

Because the ADEOS takes a sun-synchronous polar orbit, the ILAS measurement area during the satellite sunrise and sunset (Fig. 1) are limited to the high latitude regions of both hemispheres (56~70°N & 63~88°S). As the ADEOS revolves the earth about 14 times a day, the number of sunrise and sunset events observed from the satellite is totally twenty-eight a day, which gives fourteen opportunities of measurements for each hemisphere with a longitude shift of about 25 degrees. The altitude measurement range is from 10 km (at the top of the clouds) to 60 km with 2 km resolution.

ILAS measurement gases are ozone (O_3), methane (CH_4), nitrous oxide (N_2O), nitric acid (HNO_3), water vapor (H_2O), nitrogen dioxide (NO_2), and hopefully CFC-11 using the IR channel. In the visible channel, absorption by the atmospheric A band of oxygen molecules (O_2) is measured to derive profiles of temperature, pressure, and aerosol concentration.

The profiles of pressure and temperature derived from visible channel data are used to calculate absorption cross section values in the radiative transfer for the IR channel data retrieval. This self-sufficiency is one of the ILAS's great advantages.

ILAS consists of a sun tracking unit, a telescope, an infrared sensor unit, a visible sensor unit, and an electronic circuit unit. The overview of the ILAS instrument is shown in Fig. 2.

4. Measurement data

The IR channel and the visible channel data are converted to 12 bit digital signals. These data with ILAS house keeping data and satellite orbit data are determined as "*level 0*" data, which are delivered by NASDA/EOC (Earth Observation Center). The amount of the level 0 data will be approximately 558 MB per day for 28 measurements of sunrise and sunset events a day.

The data partially extracted from level 0 data and corrected for baseline drift are "*level 1*" data, which will be about 112 MB per day.

The vertical profiles of gas mixing ratio, temperature, pressure, and aerosols retrieved from level 1 data are called "*level 2*" data. Each level 2 data is registered with location and time of the measurement. The amount of level 2 data will be about 0.5 MB per day.

Meteorological data, ephemeris data, and solar activity data are also used as the related fundamental data in the ILAS data processing.

5. Data retrieval algorithm

Data retrieval method from level 1 to level 2 data is designed to obtain vertical profiles of target species with sufficiently small error of estimation. The outline of the scheme is as follows.

In the ILAS visible channel, computer simulation data (synthetic data) of the ILAS outputs are theoretically computed by assuming vertical profiles of temperature and pressure after determining extinction caused by aerosol scattering. Then the assumed profiles are iteratively adjusted to minimize a sum of squares of the differences between the computed values and the measured values. The final profiles of pressure and temperature are the retrieved results.

This method is called "non-linear least squares (NLSQ) fit" computation technique. ILAS instrument function and all other non-linear factors involved in the measurement are also considered in this computation. Therefore the best solutions are expected to be obtained by this method.

For the IR channel data, vertical profiles of gas mixing ratio are adjusted instead of those of pressure and temperature in the visible channel case.

6. Investigation results

At the stage of the computer simulation, HITRAN database of the chemical molecules' absorption line parameters will be used as an atmospheric absorption line compilation. There are about 180,000 lines in the ILAS IR spectral region, and about 300 lines in the ILAS visible band. Therefore the stage of computer simulation for the IR channel is very CPU time-consuming. We have investigated some methods of practically rapid and precise calculation of the atmospheric cross section values such as highly vectorized line-by-line calculation by using Voigt function, table look-up method of Voigt values, and cross section table look-up method as a function of pressure and temperature (P-T table look-up). The investigation results of the rapid Voigt calculation codes are reported as a separate paper (3) in this book.

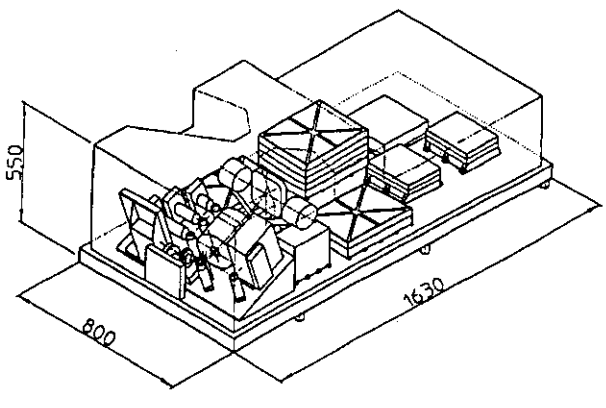

Fig. 2 Overview of the ILAS instrument

Table 1 and 2 show rough estimation results of the investigation of operational computing feasibility. In order to process one measurement data set, it must be done within 50 minutes because ILAS will take 28 measurements a day. According to the result shown in Table 1, we cannot adopt a line-by-line method under this constraint. Even in case of using P-T table look-up method, we will entirely occupy the SX-3 system resources only for this project. However, if we use cluster multi-CPU workstations and adopt P-T table look-up method, the ODPS will process ILAS data in operationally sufficient time (cf. Table 2). Therefore we are now planning to establish the ILAS/ODPS by using a cluster computer system which is composed of multiple workstations instead of using the SX-3 system. We may use the SX-3 system not operationally but for the program tuning, for line-by-line calculation to produce P-T table data, and for researching algorithm improvement.

The ILAS/ODPS will be established under these investigations by using the SX-3 super computer system.

Table 1 Operational computing feasibility in case of using the super computer SX-3 system

		P-T Table look-up	line-by-line for Voigt
IR Detector Elements		44	44
Mem-ory	Cross section calc.	289 MB	117 MB
	Mass-path calc.	5.0 GB	5.0 GB
Disk Capa-city	P-T coeff. table	7.4 GB	-
	Cross section File	5.0 GB	5.0 GB
	Total	12.4 GB	5.0 GB
Calc. Time	CPU Time	31.6 sec*	6.8 days*
	P-T Table Input	378.4 sec*	-
	Cross section Output	259.6 sec*	257.4 sec*
	Total	669.6 sec*	6.8 days*

* for one measurement data set

Table 2 Operational computing feasibility in case of using cluster multi-CPU workstations (S-4/670MP)

		P-T Table look-up	line-by-line for Voigt
Number of workstations		6	44
Elements per workstation		8	1
Mem-ory	Cross section calc.	805 MB**	117 MB**
	Mass-path calc.	936 MB**	117 MB**
Disk Capa-city	P-T coeff. table	1376 MB**	-
	Cross section File	936 MB**	117 MB**
	Total	2312 MB**	117 MB**
Calc. Time	CPU Time	180.8 sec*	8655 sec*
	P-T Table Input	137.6 sec*	-
	Cross section Output	93.6 sec*	12 sec*
	Total	412.0 sec*	8667 sec*

* for one measurement data set, ** per workstation

References

1. Y. Sasano et al., "ILAS and RIS for ADEOS," in *Proceedings of SPIE*, vol. 1490, pp. 233-242, (1991).

2. T. Yokota et al., "Data System of the Improved Limb Atmospheric Spectrometer (ILAS) on ADEOS for Polar Stratospheric Ozone Monitoring," in *Proceedings of IGARSS'93*, IEEE-93CH3294-6, vol. 3, pp. 1081-1083, (1993).

3. M. Suzuki, "Comparison of fast calculation codes for Voigt function," in this book.

Comparison of fast calculation codes for Voigt function

Contact Person Makoto Suzuki
Global Environment Research Division
National Institute for Environmental Studies
Environment Agency of Japan
Keywords Voigt function, complex error function, radiative transfer, remote sensing, ILAS

1. Background and Objectives

Computation of molecular spectral line shape is an important problem for spectroscopy and atmospheric radiative transfer.[1,2] Especially recent requirement for the accurate retrieval of high resolution atmospheric remote-sensing data[3] needs fast and accurate line-by-line atmospheric radiative transfer code. To calculate the line shape the effects of both pressure broadening (Lorentian line shape) and Doppler broadening have to been taking account. The convolution of these profiles gives the Voigt line shape

$$\gamma_V(\tilde{\nu}) = \gamma_L \otimes \gamma_D = \int_{-\infty}^{+\infty} \gamma_L(\tilde{\nu}-\tilde{\nu}')\gamma_D(\tilde{\nu}')\, d\tilde{\nu}', \qquad (1)$$

where ν is the wavenumber (or frequency) and γ_L, γ_D, and γ_V are normalized line shape functions for Lorentian, Doppler, and Voigt, respectively.

This convolution integral has to be evaluated numerically, which takes significant computation time. The accuracy of numerical solution is obtained by an expense of computation time, because the accuracy is closely related to the terms of series expansion, continued fraction, rational approximation or Gauss-Hermite integration. Therefore there is strong needs for faster algorithms with better accuracy.

The Environment Agency is now developing a satellite sensor, "Improved Limb infrared Atmospheric Spectrometer (ILAS)" for ozone layer monitoring and research[4], which will be launched on Feb. 1996. The ground data segment for ILAS instruments will be established at NIES, where accurate atmospheric radiative transfer code is necessary.

In the present work, several computational procedures for Voigt function accurate more than 4 digits are compared concerning their computational speed. Recently, Schriefer[5] concluded that two algorithms developed by Humlicek[6], and by Hui et al.[7] are adequate for vector super computer based upon a benchmark result using Cray Y-MP2. The author found that Hui's algorithm shows better performance than others that includes Humlicek-82 on SX-3. This benchmark difference on CRAY and SX-3 are briefly discussed both on Algorithmic detail and computer hardware architecture. An estimate of Voigt line-by-line calculation using SX-3 for the ILAS data system is also shown.

2. Definition of Voigt Function

Voigt profile $\gamma_V(x,y)$ and Voigt function $K(x,y)$ is defined by

$$\gamma_V(x,y) = \frac{1}{\nu_D}\sqrt{\frac{\ln 2}{\pi}}\, K(x,y). \qquad (2)$$

and

$$K(x,y) = \frac{k_\nu}{k_0} = \frac{y}{\pi}\int_{-\infty}^{+\infty} \frac{\exp(-t^2)}{(x-t)^2 + y^2}\, dt, \qquad (3)$$

where the k_0 is the dimensional constant of the reduced absorption coefficient and S is integrated line strength.[1,2,5] The dimensionless variables x, y are defined using the distance from line center, and the ratio of Lorentian and Doppler half widths γ_L and γ_D:

$$\begin{aligned} k_0 &= \frac{S}{\gamma_D}\sqrt{\frac{\ln 2}{\pi}} \\ x &= \sqrt{\ln 2}\,\frac{\tilde{\nu}-\tilde{\nu}_0}{\gamma_D}, \\ y &= \sqrt{\ln 2}\,\frac{\gamma_L}{\gamma_D} \end{aligned} \qquad (4)$$

$$S = \int_{-\infty}^{+\infty} k_\nu\, d\nu.$$

Pressure broadening of molecular transition is proportional to the pressure p with typical values $\gamma_L \approx 0.1\, p$ [cm^{-1}/atm]. The Doppler half width is proportional to the line center wavenumber ν_0 and a function of temperature T and molecular mass m:

$$\gamma_D = \tilde{\nu}_0\sqrt{\frac{2\ln 2\, kT}{mc^2}},$$

where k and c are the Boltzmann constant and the speed of light. For atmospheric molecules of average mass, the Doppler half width $\gamma_D \approx 6\cdot10^{-8}\,\nu_0\, T^{0.5}$.

For vanishing arguments x or y, $K(0, y) = \exp(y^2)[1 - \mathrm{erf}(y)]$ and $K(x, 0) = \exp(-x^2)$ respectively, where erf is the error function.

The Voigt function can be represented as the real part $K(x, y) = \mathrm{Re}[W(z)]$ of the complex function

$$W(z)=\frac{i}{\pi}\int_{-\infty}^{+\infty}\frac{e^{-t^2}}{z-t}\,dt. \qquad (5)$$

The imaginary part of this function, is written as

$$L(x,y)=\mathrm{Im}[W(z)]=\frac{1}{\pi}\int_{-\infty}^{+\infty}\frac{(x-t)e^{-t^2}}{(x-t)^2+y^2}\,dt. \qquad (6)$$

The function W is closely related to the complex error function (probability function) according to

$$w(z)=\begin{cases} W(z) & y>0, \\ W(z)+2\,e^{-z^2} & y<0, \end{cases} \qquad (7)$$

where w(z) is defined by

$$w(z)=e^{-z^2}\left(1+\frac{2i}{\sqrt{\pi}}\int_0^z e^{t^2}dt\right) = e^{-z^2}\left[1-\mathrm{erf}(-iz)\right]=e^{-z^2}\,\mathrm{erfc}(-iz). \qquad (8)$$

Further mathematical properties and relationship of the Voigt function and complex error function are described by Armstrong[1,2], Gautschi[8], Faddeyeva and Terent'ev[9].

3. Computational Algorithms

Table 1 lists the computational methods compared in this study. These algorithms are characterized as fast computation and relatively good accuracy. They are in general truncated approximation of series expansion, continued fraction expansions, rational approximations, Gauss-Hermite integration or combinations of them.

The Gauss-Hermite integration (eq. 9a) gives good approximation for increasing |x| and y with decreasing number of terms[1,2,8,9], therefore it is applied in several methods [Drayson[10], Humlicek[5], and Pierluissi[11]] listed in Table 1. However, at small y (and |x|), the pole on the y-axis makes the Gauss-Hermite integration inappropriate[1,2,9].

$$\frac{1}{\sqrt{\pi}}\int_{-\infty}^{+\infty}\frac{e^{-t^2}dt}{z-t}=\frac{1}{\sqrt{\pi}}\lim_{n\to\infty}\sum_{k=1}^{n}\frac{H_k^{(n)}}{z-x_k^{(n)}} \qquad (9a)$$

$$=\cfrac{1}{z-\cfrac{1/2}{z-\cfrac{1}{z-\cfrac{3/2}{z-\cfrac{2}{z-\cdots}}}}} \qquad (\Re z\neq 0). \qquad (9b)$$

Continued fraction (eq. 9b) also gives good approximation for the region $y > 0$.[9, 12] Although it requires significant computation time, Drayson[10] used the continued fraction for the region $y > 1$ if Gauss-Hermite integration in 2 terms approximation is not applicable. The number of continued fraction terms required for Drayson's method is varied from 9 near x=0 and y=1 to 4 at the border to Gauss-Hermite integration region.

For small y, Armstrong[1], and Drayson[2] uses the asymptotic expansion derived by Hummer[13], and Faddeyeva and Terent'ev[9]. The function $w(z)$ can be written

$$w(z)=e^{-z^2}+\frac{2i}{\sqrt{\pi}}F(z) \qquad (10)$$

where F(z) is called Dawson's integral,[14]

$$F(z)=e^{-z^2}\int_0^z e^{t^2}dt. \qquad (11)$$

This function is real on the real axis; Dawson's $F(x)$. The coefficients d_n of the Taylor series about some point x on the x-axis for y-direction satisfy the recurrence relations,[9]

$$\begin{aligned} d_0 &= F(x) \\ d_1 &= 1-2xd_0 \\ d_n &= -\frac{2}{n}(xd_{n-1}+d_{n-2}) \quad n\geq 2. \end{aligned} \qquad (12)$$

The Voigt function, real part of w(z), is therefore given by the real part of $\exp(-z^2)$ added to the imaginary part of the Taylor expansion for $F(z)$:

$$\begin{aligned} Im(F(z)) &= d_1y-d_3y^3+d_5y^5-\cdots \\ K(x,y) &= Re(exp(-z^2))+Im(F(z)) \end{aligned} \qquad (13)$$

where $z=x+iy$.

As the initial value to calculate recursion eq.(12), Armstrong[1] used the asymptotic expansion of F(x) for the region $x > 5$ and $y \leq 1.8/(x+1)$,

$$F(x)\cong\frac{1}{2x}+\frac{1}{2^2x^3}+\frac{1\cdot 3}{2^3x^5}+\frac{1\cdot 3\cdot 5}{2^4x^7}+\cdots. \qquad (14)$$

For the region $x \leq 5$, Armstrong[1] and Drayson[10] uses the Chebyshev polynomial expansion obtained by Hummer.[15]

Hui et al.[7] and Humlicek[6,16] use rational approximation of complex error function, $w(z)$,

$$w(z)\cong\frac{a_0+a_1z+a_2z^2+a_3z^3+\cdots}{b_0+b_1z+b_2z^2+b_3z^3+\cdots}. \qquad (15)$$

In expansions of 5 to 7 terms determined by minimax approach[17], rational approximation of these function gives fast and accurate alternative against Taylor series expansion used by Armstrong[1,2] or by Drayson[10] for the region of small x and y.

Among the algorithms listed in Table 1, those by Gautschi[18] [which is available as a library function in IMSL[19]] and by Armstrong[1] have been proving to be highly accurate.[5] Thus they are also used as references for accuracy of other methods.

4. Benchmark Results and Discussion

Table 2 list the benchmark results of algorithms developed by Armstrong[1], Drayson[10], Hui et al.[7], Humlicek[6], Gautschi[8,19], and Pierluissi et al.[11], where

the results on IBM 3090 and Cray Y-MP2 were measured by Schriefer[5] and other value on Sun and NEC SX-3/14 were measured by the author. The data were taken for two regions, (1) for small y: $0 < y \le 1$, and (2) for medium y: $1 < y \le 10$. In these two regions, 50 points in constant spacing for y coordinate and 1000 points, $0 \le x < 10 \cdot x_{1/2}(y)$, as a total of 50,000 points in x-y plane were calculated for the benchmark. The Voigt half width for given y, $x_{1/2}(y)$, is calculated by the definition by Olivero and Longbothum,

$$x = (y + \sqrt{\ln 2})\, R\left(\frac{y - \sqrt{\ln 2}}{y + \sqrt{\ln 2}}\right),$$

$$R(d) = 1 - 0.1821 \cdot (1 - d^2) \qquad (16)$$
$$+ (0.023665\, e^{0.6d} + 0.00418\, e^{-1.9d})\, \sin \pi d.$$

The value for Sun 4 and IBM 3090 show the typical characteristics of the algorithms on scalar computers. Indeed there is no difference in feature between Sun and IBM 3090, for both (1) two regions of y, and (2) among algorithms compared. The author therefore conclude that the results on Table 1 is generally applicable to any scalar-type machine. Drayson[10], Hui[6], Humlicek[7], and Pierluissi[11] showed comparable speed each other over both small ($0 \le y < 1$) and medium y ($1 \le y < 10$) regions, which shows these algorithms were already well tuned for scalar computers. And considering various mathematical method used, it is almost impressive that the four method showed comparable speed. In the case of radiative transfer calculation, it is frequent to evaluate the cross-section over wide spectral range in constant frequency interval. The results of optimized code (Drayson's and Hui et al.'s) for the case of a given y (fixed) and an array of x (contains the value in constant interval) are also listed in Table 2 (below the results of original codes). The performance increase around 10-20% were observed for these scalar computers.

On the vector computers(Cray Y-MP2, NEC SX-3/14, and IBM 3090 vector option), the speed of algorithms shows much differences compared to the scalar machines. Hui's simple function couldn't be inline-coded on Cray thus it took almost 1450 ms. On the contrary, it is well inline-coded and show fastest results on NEC SX-3/14, 2.53 ms. Original form of Drayson's and Pierluissi et al.'s method do not show good vector performance. While constant interval Drayson's code marked significant speed increase, it is still slower than Hui's code. The Pierluissi's method may be coded in constant interval form, in such a case it is certain to show significant performance increase as in the case of Drayson's code. The Humlicek-82 code by Schriefer[5,6] which showed best performance on Cray Y-MP2, also show good results on NEC-SX3. The constant interval approach again showed good speed improvement on Humlicek-Schriefer's code (7.04 to 2.78 ms, and 4.23 to 2.35 ms). The IMSL-CERFE function which uses gautschi's algorithm gave slowest results on both Cray and NEC, because it is not vecterized at all as an IMSL library function.

Table 3 shows detail comparison of benchmark results for Humlicek-82, constant interval Humlicek-82 and Hui's code. The last column shows dummy results to prepare the array for x. The original Humlicek-82 code appears well vecterized (99.615 and 99.430% for different regions of y), but it takes more vector instructions than Hui's and constant interval Humlicek code. This clearly shows the cost of BLOCK-IF statement in the Humlicek-82 code, which disappear in the constant interval version. The two codes, the constant interval Humlicek's and the Hui's show the comparable speed. In the condition tested here (1000 points for x), the vector length (VLEN) of constant interval method is far below the optimum length, 250, of SX-3. Hui's code shows 2.5 GFlops when 10,000 points for x is calculated, and constant interval Humlicek shows about 50% performance increase under the same condition.

Comparing Hui's and constant interval Humlicek code (which is developed by the author), both code shows comparable speed on SX-3(Table 2 and 3) and accuracy[5]. Because the simplicity of Hui's code (see Appendix) and it is applicable any step of x, the author concludes the Hui's code is better than the Humlicek's code. But it should be noted that at large x and y the rational approximation used by Hui et al. shows offset[5] around $+10^{-8}$.

It is not clear why the benchmark on Cray Y-MP2 and NEC SX-3 shows slightly different tendency for Humlicek's and Hui et al.'s codes is not clear. It should be attributed to the number and the length of vector pipelines.

The ILAS data system will require the massive line-by-line computing capability, where 28 observations per a day, some 180,000 lines, resolution of 0.002 cm^{-1} or better at least ± 25 cm^{-1} calculation range per line, and 50 or more layers covering 5 to 60 km. The estimated CPU time if using SX-3 (assuming 1.5 ms saturated CPU time for 10,000 points calculation) for processing one ILAS observation (in each 50 minites) is,

$$\frac{1.5\ ms}{1.0E4} \cdot \frac{50\ cm^{-1}}{0.002\ cm^{-1}} \cdot 1.8E3 \cdot 50$$
$$= 33{,}750\ sec, \qquad (17)$$
$$= 562.5\ min.$$

This CPU time is longer than the actual interval of observation, 50 min. Consequently it is inappropriate

to utilize the Voigt line-shape algorithms compared in this paper for the ILAS data processing. It is therefore necessary to use some sort of interpolation or table reference approach, the latter is indeed now under development for the ILAS data system.

Acknowledgment

The author express special thanks to Dr. Tadao Aoki (Meteorological Research Institute, Japan Meteorological Agency) and Prof. Teruyuki Nakajima (Center for Climate System Research, University of Tokyo) for the present work.

References

1. B.H. Armstrong, *JQSRT* 7,61 (1967).
2. B.H. Armstrong and R.W. Nicholls, "Emission, Absorption and Transfer of Radiation in Heated Atmosphere", Pergamon Press, Oxford (1972).
3. R. H. Norton and C. P. Rinsland, *Appl. Opt.* 30, 389 (1991).
4. M. Suzuki, A. Matsuzaki, T. Yokota, Y. Sasano T. Ishigaki, A. Kimura, N. Araki, J. Japan IR Sci. & Tech 1(2), 1(1991).
5. F. Schreifer, *JQSRT* 48, 743 (1992).
6. J. Humlicek, *JQSRT* 27, 437 (1982).
7. A. K. Hui, B. H. Armstrong, and A. A. Wray, *JQSRT* **19**, 509 (1978).
8. W. Gautschi, "Error Function and Fresnel Integrals", M. Abramowitz and I. A. Stegun eds.,"Handbook of Mathematical Functions", NBS, AMS55, New York (1964), Chapter 7.
9. V. N. Faddeyeva and N. M. Terent'ev, "Tables of Values of the Function w(z) For Complex Argument, Pergamon Press, New York (1961) [Translation; original published in 1954).]
10. S. R. Drayson, *JQSRT* **16**, 611 (1976).
11. J. H. Pierluissi, P. C. Vanderwood, and R. B. Gomez, *JQSRT* **18**, 555 (1977).
12. B. D. Freid and S. D. Conte, "The Plasma Dispersion Function", Academic Press, New York (1961).
13. D. G. Hummer, *Mem. R. Astr. Soc.* **70**, 1 (1965).
14. H. G. Dawson, *Lond. Math. Soc. Proc. S1*, **29**, 519 (1898).
15. D. G. Hummer, *Maths. Comput.* **18**, 317 (1964).
16. J. Humlicek, *JQSRT* **21**, 309 (1979).
17. A. Curtis and M. R. Osborne, *Computer J.* **9**, 286 (1967).
18. W. Gautschi, *Comm. ACM* **12**, 635 (1969); *SIAM J. Num. Anal.* **7**, 187(1970).
19. IMSL SFUN/LIBRARY User's Manual (version 2.1), (1989)
20. J. J. Olivero and R. L. Longbothum, *JQSRT* **17**, 233 (1977).

Table 1. Overview of different algorithms

Author	Region	Method	Stated Accuracy
Armstrong	$y < 1, x < 4$ or $y<1.8/(x+1), x > 4$ $1 < y < 2.5, x \leq 4$ $y > 2.5, x < 4$ or $y > 1.8/(x+1), x > 4$	Hummer-Faddeyeva/Terent'ev power series 20-term Gauss-H. integration 20-term Gauss-Hetmite integration	1-2 digits in 6th significant figure
Drayson	$y < 1, x < 5 - 0.8\ y$ $y > 1, x < 1.85(3.6-y)$ else $y < 11 - 0.6875\ x$ else	Taylor and Chebyshev expansion continued fraction 4-point Gauss-Herimite integration 2-point Gauss-Herimite integration	10^{-4} relative error
Gautschi+	all x, y	truncated Taylor expansion Gauss-Hermite quadrature for large |z|	10 digit accuracy
Hui et al.	all x, y	5th or 6th order rational approximation	$<10^{-6}$ relative error 5 or 6 digit accuracy
Humlicek (1982)	$\|x\| + y \geq 15$ $5.5 \leq \|x\| + y < 15$ $\|x\| + y < 5.5$ and $y \geq 0.195\ \|x\| - 0.176$	rational approximation for 3 domains	10^{-4} relative error
Pierluissi *et al.*	$0 \leq x < 3, 0 \leq y < 1.8$ $3 \leq x < 5, 1.8 \leq y < 5$ $x \geq 5, y \geq 5$	series expansion 6-point Gauss-Herimite integration 4-point Gauss-Herimite integration	peak deviation $\leq 1.4 \cdot 10^{-3}$ RMS deviation $\leq 1.4 \cdot 10^{-5}$

+Implemented as CERFE and ZERFE for IMSL library available on SX-3 and other machines.

Table 2. Computing times for 1000 mesh points in $0 < x < 10 \cdot x_{1/2}(y)$ and 50 y cases (50000 calculations).

	Function call CPU Time (msec)							
	Sun 4/670MP*		IBM 3090+		CRAY Y-MP2+		NEC SX-3/14	
Algorithms	0<y<1	1<y<10	0<y<1	1<y<10	0<y<1	1<y<10	0<y<1	1<y<10
Armstrong	913	678	1407	1070	2042	1555	1762	178.4
Drayson	295	183	428	227	1612	1427	187.4	92.8
(const. interval)	(235)	(152)					(7.58)	(6.69)
Hui, p = 5	363	362	432	430	1457	1455	2.53	2.55
(subroutine)			(83)	(83)	(18)	(17)		
Humlicek82	467	300	480	310	1542	1431	7.04	4.23
(const. interval)	(375)	(272)	357**	165**	27**	9**	(2.78)	(2.35)
IMSL-CERFE	na.	na.	5763	3282	4167	2675	5053	2974
Pierluissi	263	143	610	247	1603	1427	814.4	73.7

+Taken from reference 12. *Super SPARC 40 MHz, 1 MBytes Cache. **The value for subroutine.

Compiler options, for Sun, Sun FORTRAN ver. 2.1.1: f77 -O4 -libmil -cg92 -Bstatic -r8 and for SX-3: f77sx -A -pi.

Table 3. Vecterization benchmark results. (For 1000 mesh points in $0 < x < 10 \cdot x_{1/2}(y)$ and 50 y cases)

	Humlicek82		Humlicek82 const. interval		Hui et al.		dummy
	0<y<1	1<y<10	0<y<1	1<y<10	0<y<1	1<y<10	
User Time (msec)	7.038	4.226	2.775	2.349	2.531	2.547	1.197
Vector Time (msec)	5.475	3.040	1.442	1.063	1.542	1.544	0.249
Inst. Count	140317	86783	80489	66961	61095	61094	35916
V. Inst. Count	74654	35654	21864	12894	26869	26870	2067
V. Element Count	16989137	8915335	3976931	2136361	6676113	6676163	476013
FLOP Count	7942843	4228660	2162906	1065362	3956128	3956178	56128
MOPS	2423.17	2121.59	1454.29	932.69	2651.10	2634.97	426.12
MFLOPS	1128.53	1000.56	779.44	453.63	1562.98	1553.47	46.91
VLEN	227.572	250.051	181.894	165.686	248.469	248.462	230.292
V. Op. Ratio (%)	99.615	99.430	98.547	97.532	99.490	99.490	93.361

cf. The column for dummy shows the results without function or subroutine call , which is included in the other value.

Appendix: Sample code, Hui et al's 5-th order rational approximation[7].

```
      REAL FUNCTION HUI(X,Y)
      COMPLEX Z,ZH,F
      PARAMETER (A5=0.56418958297228, B5=8.2863279156)
      PARAMETER (A4=4.67506018267650, B4=33.5501020941)
      PARAMETER (A3=18.64649990312317,B3=80.6459493922)
      PARAMETER (A2=43.16280063072749,B2=118.6763981260)
      PARAMETER (A1=57.90331938807185,B1=99.9290005933)
      PARAMETER (A0=37.24429446739879,B0=37.2442945086)
      Z = CMPLX(X,Y)
      ZH = CMPLX(IMAG(Z),-REAL(Z))
      F = (((((A5*ZH+A4)*ZH+A3)*ZH+A2)*ZH+A1)*ZH+A0)
     &       /((((((ZH+B5)*ZH+B4)*ZH+B3)*ZH+B2)*ZH+B1)*ZH+B0)
      HUI = REAL(F)
      RETURN
      END
```

CLASSIFICATION OF WETLAND VEGETATION BY TEXTURE ANALYSIS METHODS USING ERS-1 AND JERS-1 IMAGES

Yoshiki Yamagata and Yoshifumi Yasuoka

National Institute for Environmental Studies
Social and Environmental Systems Division
16-2 Onogawa, Tsukuba, Ibaraki 305, JAPAN

Keywords Wetland, Remote Sensing, Synthetic Aparture Radar, Texture

Abstract - Images obtained by the C-band (5.3GHz, 5.7cm) VV-polarized ERS-1 SAR and the L-band (1.275GHz, 23.5cm) HH-polarized JERS-1 SAR were analyzed for the classification of wetland vegetation types. Both scenes were obtained when the wetland vegetation was at the maximum biomass stage. We applied texture analysis method using the cooccurrence matrix to classify the vegetation of the Kushiro mire. The result of the classification was evaluated comparing with the result using the SPOT data. As a result, it is shown that several texture features of SAR images are useful for the wetland vegetation classification. We especially demonstrate that the bog and the fen vegetation in the wetland can be segregated using JERS-1 image, and also the swamp forest can be delineated from the fen vegetation using ERS-1 image.

INTRODUCTION

In recent years, people began to realize the importance of wetlands in terms of nature conservation. However the vegetation investigation inside wetlands is very difficult and it is hoped that the wetland vegetation can be monitored using remote sensing. At present, our research group is under the project of wetland monitoring using airborne MSS data and various satellite images. The biggest problem there is the cloud cover that prevents us from monitoring wetlands during the vegetation growing seasons. Recently, SAR sensors have been launched and available for observation of the ground through the cloud cover, it is an urgent subject of research to investigate the applicability of SAR sensor for wetland monitoring.

In July 1991, European Space Agency launched ERS-1 and in February 1992 , National Space Development Agency of Japan launched JERS-1. Both satellites carry SAR sensor on board. SAR sensor is an active all-weather type sensor that emits microwave by itself and observe the reflection from the ground. Although not all is clarified about the interaction between the microwave scattering intensity and the ground conditions yet, the scattering intensity is known to be affected by the surface roughness. Microwave is especially greatly scattered by the structure objects, which have the same scale as the wavelength. In this study, using the ERS-1 and JERS1-1 SAR images, we analyzed the relationships between SAR backscatter intensity and the wetland vegetation types.

In the conventional remote sensing using optical sensors, vegetation classifications are generally conducted by the maximum likelihood supervised classification method and the vegetation condition is evaluated using the numerical operation between bands (ex. vegetation index). However, a SAR image has only one band and the image is uniformly contaminated by so called speckle noises, so it is difficult to apply the conventional pixel by pixel method directly to the SAR image analysis. As a first step of the SAR image analysis for the wetland monitoring, we have investigated the texture analysis method using the cooccurrence matrix [1,2] and evaluated the results comparing with the wetland vegetation classification using a SPOT image.

STUDY AREA AND ANALYZED IMAGES

Study area is the Kushiro mire which is located at east part of Hokkaido island of Japan (Figure 1). This mire is formed by the peat accumulation during the last 4000 years and the peat layer depth is from 1 to 4m. This mire is composed of three types of wetland category, bog, fen and swamp. Bog area is mainly covered by moss (Sphagnum spp.), the fen is covered by reed (Phragmites communis) and sedge (Eriophorum vaginatam), and the swamp is covered by alder (Alnus japonica).

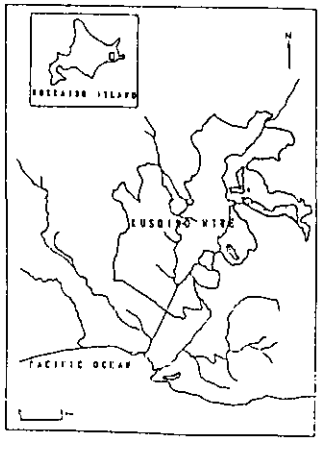

Figure 1. Study area for the wetland vegetation classification using SAR image texture analysis.

We used ERS-1 and JERS-1 images acquired around when the mire vegetation attained maximum biomass level. Table 1 shows the specification of each sensor and the acquisition condition.

Table 1. Comparison between ERS1 and JERS-1 image

Item	ERS-1	JERS-1
Frequency	5300MHz (c band)	1275MHz (L band)
Wavelength	5.7cm	23.5cm
Polarization	VV	HH
Incident angle	23deg	39deg
Resolution	30m	18m
Swath width	100km	75km
Recurrent day	35day	44day
Acquisition date	26 Aug.	12 Sept.

COOCCURRENCE MATRIX METHOD

In order to analyze the texture of SAR images, we have applied the cooccurrence matrix method. In this method, firstly the original image is subdivided into small windows and level sliced to digital values with smaller levels. Secondly, for each window all pixel with level j that is (m,n) pixels distance far from the pixel with level i is counted and divided by the total number of the combination. This value P(i,j), which represents the probability of cooccurrence of levels in the window, is defined as the element of cooccurrence matrix. (m,n is the adjustable parameters depending on frequency of the texture. For the fine texture image, small value such as m=1,n=1 is used) . Texture feature of the small window is then calculated using the several parameters which characterize the cooccurrence matrix. We used next parameters as to extract texture features from SAR images.

CONTRAST

$$F=\sum_{k=0}^{LVL-1} k^2\left\{\sum_{i=1}^{LVL}\sum_{\substack{j=1 \\ |i-j|=k}}^{LVL} P(i,j)\right\} \quad (1)$$

VARIANCE

$$F=\sum_{i=1}^{LVL}\sum_{j=1}^{LVL}(i-\mu x)^2 P(i,j) \quad (2)$$

INEVERSE DIFFERENCE MOMENT

$$F=\sum_{i=1}^{LVL}\sum_{j=1}^{LVL}\frac{1}{1+(i-j)^2}P(i,j) \quad (3)$$

SUM AVERAGE

$$F=\sum_{k=2}^{2\times LVL} kP_{x+y}(k) \quad (4)$$

SUM VARIANCE

$$F=\sum_{k=2}^{2\times LVL}(k-\text{Sum Average})^2 P_{x+y}(k) \quad (5)$$

ENTROPY

$$F=-\sum_{i=1}^{LVL}\sum_{j=1}^{LVL} P(i,j)\log P(i,j) \quad (6)$$

RESULT OF TEXTURE ANALYSIS

Texture analyses were performed for ERS-1 and JERS-1 SAR images. In this analysis, original images were subdivided in to 8 x 8 windows, and the range of 3σ of digital value were level sliced into 10 levels.

By comparing the images of all texture feature parameters of cooccurrence matrix method applied to JERS-1 and ERS-1 images, Sum Average, Sum Variance and Sum Entropy were selected as suitable parameters to enhance the texture features of wetland vegetation types. Figure 2 shows the result of the texture analysis using these parameters.

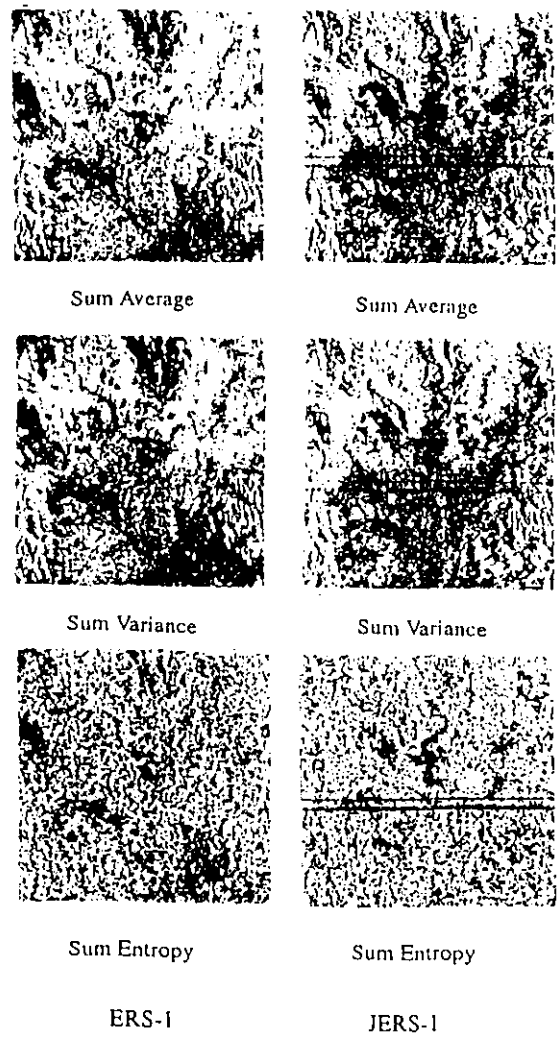

Figure 2. Texture features of ERS-1 and JERS-1 SAR image

To evaluate the vegetation classification performance of the SAR image texture analysis, we also conducted a precise classification using the SPOT image acquired on 29 August 1991. ERS-1, JERS-1 and SPOT images were all geometrically corrected each other to allow direct comparisons. Delineated vegetation classes were bog, fen (sedge dominant), fen (reed dominant), forested swamp, water, forest and others. The training area for each land use categories were selected by the ground truth and the classification was performed using maximum likelihood method.

By comparing the result of the texture analysis of SAR images with the actual vegetation distribution known from the SPOT image classification, it has been shown:
a) Wetland vegetation classes are more distinct in JERS-1 image.
b) In ERS-1 image, there are large low backscatter intensity regions that do not correspond to the apparent vegetation type nor ground condition.
c) Bog area and water inundated area is distinct in JERS-1 image.
d) Difference between fen and swamp were better distinguishable in ERS-1 image.
e) All types of wetland vegetation can not be classified at once using only SAR images.

CONCLUSION

SAR is an active type microwave sensor that can monitor the wetland vegetation during the growing seasons with cloudcover. SAR image is always stained by speckle noises, so it is difficult to apply the conventional supervised classification method to delineate the wetland vegetation types. We have investigated the texture analysis using the cooccurrence matrix for wetland vegetation classification of the Kushiro mire. We compare the results with the classification using SPOT image. The result shows that the texture feature parameters such as Sum Variance, Sum Average, Sum Entropy are effective for wetland vegetation segregation. Using JERS-1 SAR image we could delineate the bog and water inundated area. In future, it is necessary to develop the much faster supervised texture classification method which is applicable to full scene images .

ACKNOWLEDGMENTS

This study was supported by the joint project between National Institute for Environmental Studies and National Space Development Agency titled "The Study on the environmental change analysis method in the climate succession region."

REFERENCES

[1] R.W.Conners and C.H.Harlow, "A theoretical comparison of texture algorithms." IEEE Trans., vol. PAMI-2, no.3, pp.204-222 (1980)
[2] R.M. Haralick, "Textural features for image classification." IEEE Trans., vol.SMC-3, Nov., pp 610-621 (1973)