

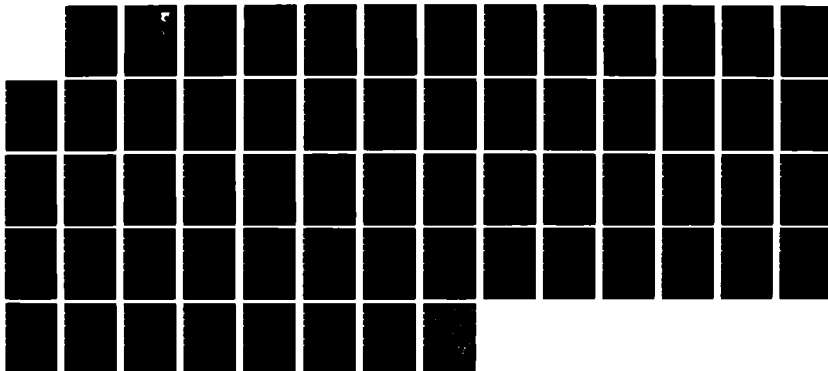
AD-A194 283

EVALUATION OF ATMOSPHERIC EFFECTS FOR OPERATIONAL
TACTICAL DECISION AID (U) PENNSYLVANIA STATE UNIV
UNIVERSITY PARK DEPT OF ELECTRICAL EN K TOMIYAMA
89 FEB 88 AFWAL-TR-87-1190 F49620-85-C-0013 F/G 4/1

1/1

UNCLASSIFIED

NL



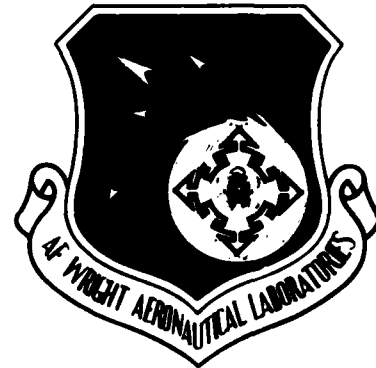


AFWAL-TR-87-1190

DTIC FILE COPY

2

AD-A194 205



**EVALUATION OF ATMOSPHERIC EFFECTS
FOR OPERATIONAL TACTICAL DECISION AID**

Dr. Ken Tomiyama
The Department of Electrical Engineering
Pennsylvania State University
University Park, Pennsylvania 16802

DTIC
ELECTE
APR 15 1988
S D

February 1988

Final Report for Period January 1987 to August 1987

Approved for public release; distribution is unlimited.

DESTRUCTION NOTICE - Destroy by any
method that will prevent disclosure
of contents or reconstruction of
the document.

AVIONICS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6543

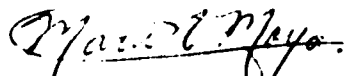
88 4 14 028

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention they may in any way be related thereto.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.



MARIO N. MOYA, Project Engineer
EO Sensors Eval/Analysis Group
Electro-Optics Branch



JAMES J. STEWART, Chief, AFWAL/AARI-3
EO Sensors Eval/Analysis Group
Electro-Optics Branch

FOR THE COMMANDER



GALE D. URBAN, Chief, AFWAL/AARI
Electro-Optics Branch

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization, please notify AFWAL/AARI-3, Wright-Patterson AFB OH 45433-6543 to help maintain a current mailing list.

Copies of this report should not be returned unless return is required by security consideration, contractual obligations, or notice on a specific document.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified		1b. RESTRICTIVE MARKINGS	
2a. SECURITY CLASSIFICATION AUTHORITY		3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.	
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE			
4. PERFORMING ORGANIZATION REPORT NUMBER(S)		5. MONITORING ORGANIZATION REPORT NUMBER(S) AFWAL-TR-87-1190	
6a. NAME OF PERFORMING ORGANIZATION Pennsylvania State University	6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Avionics Laboratory (AFWAL/AARI) AF Wright Aeronautical Laboratories	
6c. ADDRESS (City, State and ZIP Code) University Park, PA 16802		7b. ADDRESS (City, State and ZIP Code) Wright-Patterson AFB, OH 45433-6543	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION AFOSR	8b. OFFICE SYMBOL (If applicable) XOT	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER Contract No. F49620-85-C-0013	
8c. ADDRESS (City, State and ZIP Code) Solling Rd Washington, D.C. 20332		10. SOURCE OF FUNDING NOS.	
11. TITLE (Include Security Classification) Evaluation of Atmospheric Effects for Operational Tactical Decision Aid		PROGRAM ELEMENT NO 61102F	PROJECT NO 2305
12. PERSONAL AUTHOR(S) Ken Tomiyama		TASK NO. D5	WORK UNIT NO
13a. TYPE OF REPORT Final Report	13b. TIME COVERED FROM 1/87 TO 8/87	14. DATE OF REPORT (Yr., Mo., Day) 88/02/09	15. PAGE COUNT 59
16. SUPPLEMENTARY NOTATION			
17. COSATI CODES		18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB GR	
		On the other side.	
19. ABSTRACT (Continue on reverse if necessary and identify by block number) The Tactical Decision Aid (TDA) is an integrated target/atmosphere/sensor model that is used to estimate target acquisition ranges for infrared sensors. It employs an extensive 8000-plus line computer code, LOWTRAN-6, to evaluate the atmospheric extinction of infrared signals for various climatological conditions. The Operational Tactical Decision Aid (OTDA) is a simplified version of the TDA housed on an HP-41CX, a hand-held computer, and is intended for field use. Since LOWTRAN-6 is too voluminous to be employed for the OTDA, precomputed extinction data tables are currently in use. Manual input of data from the tables to the OTDA is cumbersome and is prone to erroneous readings. Therefore, compact atmospheric extinction models were developed for various types of atmospheric extinction which are significant for the TDA application. The models were developed based on the LOWTRAN-6 computation and were verified (over)			
20. DISTRIBUTION STATEMENT (ABSTRACT)		21. ABSTRACT SECURITY CLASSIFICATION Unclassified	
22a. NAME OF REPORT NUMBERING AGENCY		22b. TELEPHONE NUMBER (Include Area Code) (513) 255-9609	22c. OFFICE SYMBOL AFWAL/AARI-3

19. through extensive accuracy analysis. For automated generation of a data base for accuracy analysis, an interactive driver for LOWTRAN-6, called DGU, was developed. The program can create input decks for LOWTRAN-6 from interactive sessions, run LOWTRAN-6, and post process the LOWTRAN-6 generated data. Finally, the obtained models were integrated into a program, called CTRAN, and coded onto a VAX computer in FORTRAN and in Reverse Polish Notation for the HP-41CX.

10. LOWTRAN, Tactical Decision Aid, Atmospheric Absorption, Transmittance Modeling, Aerosol.

ACKNOWLEDGEMENTS

I would like to thank Roger L. Cranos and Don L. Tomlinson, both of the Avionics Laboratory, the Air Force Wright Aeronautical Laboratory (AFWAL/AARI), for providing me with the opportunity to pursue this research. I would like to mention that this report could not be materialized without the help from Mario N. Moya. Valuable help in UNIX and the C programming language from Jeff D. Sweet, also at AFWAL/AARI is acknowledged. I would also like to thank Dennis W. Richardson for his help throughout this project.

Approved For	
NAME	CRAN
DATE	TAB
BY	
A-1	



TABLE OF CONTENTS

<u>Section</u>	<u>page</u>
1. Introduction	1
1.1 Background	1
1.2 Project Objectives	2
1.3 Summary of Assumptions	3
2. Atmospheric Extinction Models	4
2.1 Introduction	4
2.2 Molecular Resonant Absorptions	5
2.2.1 Water Vapor Absorption	7
2.2.2 Uniformly-Mixed Gasses	9
2.2.3 Ozone	10
2.3 Water Vapor Continuum Absorption	10
2.4 Aerosol Extinction	12
2.5 Rain Model	14
2.6 Summary of Model Equations	14
3. Accuracy Evaluation	17
4. Model Upgrading	20
5. Model Implementation	23
6. Conclusions and Directions for Future Study	26
6.1 Conclusions	26
6.2 Directions for Future Study	27
REFERENCES	29
Appendix A. Data Generation Utility (DGU) Program	30
Appendix B. CTRAN User's Manual	38
Appendix C. CTRAN Programs	44

LIST OF TABLES

<u>Table</u>	<u>page</u>
1. <u>Preliminary</u> Extinction Models for the OTDA	15
2. Input Variables for Extinction Models	16
3. Accuracy Evaluation of Extinction Models	19
4. Upgraded Aerosol Extinction Model	22
5. Error Analysis of CTRAN	24
5-a. Molecular and Continuum Extinctions	24
5-b. RURAL Aerosol Extinctions	24
5-c. OCEAN Aerosol Extinction	24
5-d. URBAN Aerosol Extinction	24
5-e. TROPOSPHERIC Aerosol Extinction	25
5-f. FOG1 Aerosol Extinction	25
5-g. FOG2 Aerosol Extinction	25
A-1. Listing of DGU	31
A-2. Listing of a Typical Output of DGU (Partial)	37
C-1. Listing of the FORTRAN Version of CTRAN	45
C-2. Listing of the HP-41 Version of CTRAN	48
C-2.a. PTRAN - Interactive Version	48
C-2.b. CTRAN - Batch Version	51

SUMMARY

The Tactical Decision Aid (TDA) is an integrated target/atmosphere/sensor model that is used to estimate target acquisition ranges for infrared sensors. It employs an extensive 8000-plus line computer code, LOWTRAN-6, to evaluate the atmospheric extinction of infrared signals for various climatological conditions. The Operational Tactical Decision Aid (OTDA) is a simplified version of the TDA housed on an HP-41CX, a hand-held computer, and is intended for field use. Since LOWTRAN-6 is too voluminous to be employed for the OTDA, precomputed extinction data tables are currently in use. Manual input of data from the tables to the OTDA is cumbersome and is prone to erroneous readings. Therefore, compact atmospheric extinction models were developed for various types of atmospheric extinction which are significant for the TDA application. The models were developed based on the LOWTRAN-6 computation and were verified through extensive accuracy analysis. For automated generation of a data base for accuracy analysis, an interactive driver for LOWTRAN-6, called DGU, was developed. The program can create input decks for LOWTRAN-6 from interactive sessions, run LOWTRAN-6, and post process the LOWTRAN-6 generated data. Finally, the obtained models were integrated into a program, called CTRAN, and coded onto a VAX computer in FORTRAN and in Reverse Polish Notation for the HP-41CX.

1. Introduction

1.1 Background

The Tactical Decision Aid (TDA) is an integrated target/atmosphere/sensor model that is used to estimate target acquisition ranges for infrared sensors. It employs LOWTRAN-6 [1], [2], an extensive 8000-plus line computer code, to evaluate the atmospheric extinction of infrared signals for various climatological conditions. The Operational Tactical Decision Aid (OTDA) is a simplified version of the TDA, and is intended for field use. It is housed on an HP-41CX, a hand-held computer. Because LOWTRAN-6 is too voluminous to be employed in the OTDA, precomputed extinction data tables are currently in use. However, it is inconvenient to carry the printed tables to the field, and the process of manual data input from the tables to the OTDA is prone to erroneous readings. It is preferred to have an extinction computation program as a part of the OTDA. Therefore, development of a compact atmospheric extinction computation code for the HP-41CX was initiated [3]. The program will replace the transmittance tables, and automate the extinction evaluation process of the OTDA.

As an initial step towards this goal, compact atmospheric extinction models for various extinction mechanisms were developed based on the LOWTRAN-6 models [3]. First, various components of the atmospheric extinction computation in LOWTRAN-6 were studied in detail, and the extinction mechanisms which are active over the wavelength interval of interest, 8 - 12 (μm), were

identified. Then, simple analytical expressions were selected to model these active extinction mechanisms. Optimal values for model parameters were obtained by minimizing the differences between the LOWTRAN-6 computations and model predictions using parameter optimization techniques.

1.2 Project Objectives

The preliminary models in [3] were in good agreement with LOWTRAN-6 results. However, these models were not extensively tested for various combinations of climatological conditions which are typical of the TDA application. Furthermore, accuracies associated with some aerosol models may not be adequate in demanding applications. Critical testing of the developed models, including possible modifications, should be rendered before they can be coded to replace the extinction tables now in use.

To accommodate the general objective of developing an atmospheric extinction computation program for the OTDA, we have set the following specific objectives:

- (1) Develop a computer program to interactively generate extinction data for various extinction mechanisms using LOWTRAN-6, and to perform an error analysis of a given model. The program will be written as general as possible to facilitate its use for future error analysis.
- (2) Perform an exhaustive error analysis of the extinction models in [3].

- (3) Modify the models to obtain better accuracy, if warranted.
- (4) Develop an extinction computation program for the HP-41CX based on the fully tested models.
- (5) Test the program developed in (4) against climatological conditions typical to the TDA application.

Before starting the summary of previous work, some underlying assumptions which are in effect will be stated.

1.3 Summary of Assumptions

Some basic assumptions were made to focus our modeling effort onto the OTDA applications. It was assumed that the quantity to be modeled is an average transmittance over 830 - 1250 (cm^{-1}), corresponding to 8 - 12 (μm), band which is the primary spectral region of sensitivity for infrared sensors considered in the OTDA. Optical paths between the sensors and targets are considered horizontal and are located below 2 (km) altitude. The altitude of 300 (m) above sea level, which is the altitude of the sensor test cite at AFWAL/AARI, was designated as a standard height.

Finally, all simplified models should be consistent with LOWTRAN-6. Thus, the models will be derived from the LOWTRAN-6 computation.

2. Atmospheric Extinction Models

Although the derivation of various extinction models from the LOWTRAN-6 computation is reported in [3], it is revised and repeated here for review and for completeness of this report. The results of extended error analysis and model upgrading will be given in the next chapter.

2.1 Introduction

Infrared radiation passing through the atmosphere loses its intensity as a result of interactions with atmospheric constituents. A quantity which characterizes this process is the atmospheric extinction in terms of the extinction coefficient k , or the atmospheric transmission in terms of the transmittance t . The transmittance t is defined as the ratio of the emitted and received infrared radiation intensities $I(\text{emitted})$ and $I(\text{received})$ as,

$$t = \frac{I(\text{received})}{I(\text{emitted})}, \quad (1)$$

and the extinction coefficient k is related to the transmittance t by

$$t = \exp(-k). \quad (2)$$

The extinction coefficient k includes contributions from two extinction mechanisms; absorption, and scattering. Each of these, in turn, consists of various individual contributions. The absorption includes molecular resonant absorptions, molecular continuum absorptions, aerosol (including fog) absorptions, and

rain absorption. The scattering includes molecular scattering and aerosol scattering.

LOWTRAN-6 computes a LOW resolution TRANsmittance called the band transmittance. It is a degraded (or band) transmittance obtained by averaging the monochromatic transmittance over a small wavenumber interval using a triangular weighting function. LOWTRAN-6 adopts the basic assumption of superposition, where the total extinction is the sum of individual contributions. Equivalently, the total transmittance is assumed to be the product of transmittances corresponding to individual sources. This assumption enables us to deal with various extinction mechanisms listed above separately.

As LOWTRAN-6 suggests, some of the extinction mechanisms are inactive in the wavenumber region of interest, 830 - 1250 (cm^{-1}). As a result, we only need to consider the extinction due to the following: water vapor, uniformly-mixed gasses, ozone, water vapor continuum, aerosol, and rain.

As we discussed above, the requirement of the OTDA is the evaluation of the average transmittance over the 8 - 12 (μm) wavelength band. Therefore, the extinction models will be developed to represent the relationships between the average transmittance and various climatological conditions including the optical path length.

2.2 Molecular Resonant Absorptions

In LOWTRAN-6, various sources are considered for molecular resonant absorption including the three absorbers of concern; water vapor, uniformly-mixed gasses, and ozone. In the evalua-

tion of the absorption due to these three absorbers, two intermediate quantities, called an equivalent absorber amount U and a modified equivalent absorber amount x , are utilized in conjunction with two empirical transmittance models. Both the formulation of the modified equivalent absorber amount x and the computation formula for transmittance depend on the absorber, and are discussed later for each absorber.

Two empirical models, one for ozone and another for both water vapor and uniformly-mixed gasses, are stored as 67 pairs of numbers which represent transmittance t versus modified equivalent absorber amount x . The variation of the absorption with respect to the wavenumber are specified through sets of spectral parameters $C(v)$ which appear within x . The values of $C(v)$ are stored at 5 (cm^{-1}) interval over wavenumber regions of significant absorption, called the absorption bands, for each absorber. LOWTRAN-6 computes the equivalent absorber amount U first and then the modified equivalent absorber amount x using the pressure, temperature, and wavenumber dependencies specific to each absorber. Finally, the transmittance is computed using the linear interpolation of empirical transmittance functions.

Transmittance profiles for those three absorbers over the 830 - 1250 (cm^{-1}) band were generated at 5 (cm^{-1}) intervals for various combinations of values for atmospheric variables using LOWTRAN-6. Then the resulting profiles are averaged and stored together with atmospheric variables into a data base. Analytical expressions based on the LOWTRAN-6 computations are developed to model the relationship of the average transmittance versus other

variables. An optimal set of model parameters are found using linearization of model equations and the linear least square estimation technique.

2.2.1 Water Vapor Absorption

The transmittance expression used in LOWTRAN-6 for a horizontal path with homogeneous meteorological conditions of pressure P (mbar), temperature T (K), relative humidity RH (%), and path length R (km) at wavenumber ν (cm^{-1}) is as follows.

$$t = f(x), \quad (3-a)$$

$$x = C(\nu) P N^a T N^b U, \quad (3-b)$$

$$P N = P/P_0, \quad T N = T_0/T, \quad (3-c)$$

$$U = 0.1 \text{ WH } R, \quad (3-d)$$

$$\text{WH} = 0.01 \text{ RH } F(T_0/T), \quad (3-e)$$

where $f(\cdot)$, a , b , $P N$, $T N$, P_0 , T_0 , WH , and $F(\cdot)$ are the empirical transmittance function, absorber parameters ($a=0.9$, $b=0.45$), normalized pressure, normalized temperature, standard pressure (1013.25 mbar), standard temperature (273.15 K), water vapor density (g/m^3), and an empirical function for saturated water vapor density (g/m^3) at temperature T , respectively.

In earlier efforts on modeling of the molecular resonant absorption [4], [5], the following analytical expression, called the double exponential function, was found to have excellent agreement with the LOWTRAN-6 empirical transmittance function.

$$t = \exp(-10^{a_0 + a_1 x}), \quad (4-a)$$

$$x = \log C(v) + n \log(PN) + m \log(TN) + \log(U), \quad (4-b)$$

where a_0 , a_1 , n , and m are model parameters to be selected optimally.

This function was chosen as our model since it agrees excellently with the band transmittance which is a weighted average of transmittances and, therefore, is very similar to the average transmittance considered here. It is noted that the spectral parameter $C(v)$ in this equation may be eliminated in our model since only an averaged transmittance is to be modeled. As a result, the model can be simplified to

$$t = \exp(-10^{a_0 + a_1 \log(PN) + a_2 \log(TN) + a_3 \log(U)}), \quad (5)$$

or

$$t = \exp(-A_0 PN^{a_1} TN^{a_2} U^{a_3}), \quad (6)$$

where a_0 , a_1 , a_2 , a_3 , and $A_0 = 10^{a_0}$ are the adjustable model parameters.

For the optimal determination of the model parameters, we take the double logarithm of Eq. (5). This linearizes the model in terms of the unknown parameters.

$$\log(-\ln(t)) = a_0 + a_1 \log(PN) + a_2 \log(TN) + a_3 \log(U). \quad (7)$$

Linear regression techniques can then be utilized to obtain the optimal parameter values. Specifically, we take the difference

between the two sides of Eq. (7), square the difference, sum the squared differences, and minimize the sum with respect to the parameters. The minimization can be achieved by setting the partial derivatives of the sum of squared differences with respect to the unknown parameters to be zero. This process gives rise to a linear equation of the form $Ax=b$, commonly known as the normal equation, where A , x , and b are the symmetric coefficient matrix, unknown parameter vector, and the known vector, respectively. This type of equation can be solved by any linear equation solver.

2.2.2 Uniformly-Mixed Gasses

The absorber in question here is a mixture of various atmospheric gaseous molecules whose density profiles are relatively unperturbed, except for the pressure and temperature dependencies. Therefore, the corresponding absorber amount is a function of the pressure, temperature, and the path length only. Basically, the transmittance expression for this absorber is the same as that for the water vapor given in Eq. (3). The only difference is that the pressure and temperature dependencies within the absorber amount U can be integrated into these appearing in x . This leads to the following LOWTRAN-6 model.

$$t = f(x), \quad (8-a)$$

$$x = C(v) P N^a T N^b U, \quad (8-b)$$

$$U = R, \quad (8-c)$$

where the absorber parameters a and b have values 1.75 and 1.375, respectively.

Thus, similar to the water vapor case, an appropriate model is given by, Eq. (5) or (6) with the expression for U being replaced by the path length R as in Eq. (8-c).

2.2.3 Ozone

The transmittance expression for ozone is the same as that for the water vapor, except the absorber parameter values, $a = 0.4$ and $b = 0.2$, and the expression for the absorber amount U which is given by

$$U = 46.667 W_O R, \quad (9)$$

where W_O is the ozone density in g/m^3 . Therefore, the appropriate model expression is again given by Eqs. (5) or (6) together with the absorber amount expression in Eq. (9).

2.3 Water Vapor Continuum Absorption

The LOWTRAN-6 expression for the water vapor continuum absorption consists of self- and foreign-components. The expression for a homogeneous path is given by,

$$t = \exp(-v \tanh(hcv/2kT) [R_s C_s + R_f C_f] W_H R), \quad (10)$$

where $hc/k = 1.43879 \text{ (K/cm}^{-1}\text{)}$, R_s and R_f are self (water vapor versus total air at standard condition) and foreign (all other molecular species) number density ratios, and C_s and C_f ($1/(\text{cm}^{-1}\text{mol/cm}^2)$) are wavenumber dependent parameters for self- and foreign-components, respectively.

The temperature dependence of the self-component C_s is taken into account through the linear interpolation from two values, C_{s1} at 296 (K) and C_{s2} at 260 (K) if the temperature is between

these two, or by setting at one of the two if the temperature is outside of the range. This can be expressed conveniently using a factor K_p defined by,

$$K_p = \begin{cases} 1 & , & T < 260, \\ (296 - T) / (296 - 260), & 260 < T < 296, \\ 0 & , & 296 < T, \end{cases} \quad (11)$$

as

$$C_s = (1 - K_p) C_{s1} + K_p C_{s2}. \quad (12)$$

The parameters C_{s1} , C_{s2} , and C_f are stored in LOWTRAN-6 at 10 (cm^{-1}) wavenumber intervals over regions where the water vapor continuum absorption is non-trivial.

Now, various quantities in Eq. (10) are investigated. The number density ratio R_s is linearly dependent on the water vapor concentration W_H . The sum of the two densities, water vapor and all others, is linearly dependent on the product $PN*TN$ since it is the air density. Therefore, our model needs to carry two linear dependencies on $PN*TN$ and W_H in an additive fashion. The wavenumber dependent coefficient in Eq. (10), on the other hand, can be imbedded into C_s and C_f . Combining all of these observations to modify the expression in Eq. (10), we obtain,

$$t = e^{-\{[q(C_{s1}' + C_f') + K_p(C_{s2}' - C_{s1}')] W_H + r C_f' PN*TN\} W_H R} \quad (13)$$

where q and r are wavenumber independent constants, and C_{s1}' , C_{s2}' , and C_f' are scaled wavenumber dependent parameters.

For our model, the averages of C_{s1}' , C_{s2}' , and C_f' over 830

- 1250 (cm^{-1}) region are computed from the LOWTRAN-6 data, and these parameters in Eq. (13) are replaced with respective averages. Then the expression is simplified by combining the constants yielding,

$$t = \exp\{-C_0 [PN \, TN + (C_1 \, Kp + C_2) \, WH] \, WH \, R\}, \quad (14)$$

where C_0 , C_1 , and C_2 are the final model parameters.

2.4 Aerosol Extinction

Aerosols are active over 830 - 1250 (cm^{-1}) wavenumber region in both absorption and scattering. Since LOWTRAN-6 has models for the extinction, we will consider the modeling of the extinction instead of the absorption and scattering individually.

The transmittance due to aerosols is given by an exponential law,

$$t = \exp\{-X \, H \, R\}, \quad (15)$$

where X is the aerosol extinction profile which is dependent on the type of aerosol, the relative humidity RH and the wavelength. H is the aerosol density profile which represents the visibility and the altitude dependencies.

We first consider the aerosol extinction profile X . There are 10 aerosol types used in LOWTRAN-6, some are relative humidity dependent and some are not. Due to our assumption that the application of the OTDA is limited to horizontal paths below 2 (km) altitude, we only need to consider four humidity dependent aerosols; RURAL, URBAN, MARITIME, TROPOSPHERE, and two humidity independent ones; FOG1 and FOG2. LOWTRAN-6 stores four extinc-

tion profiles X , corresponding to the relative humidities (RH) of 0, 70, 80, and 99 (%), for each humidity dependent aerosol and one each for FOG1 and FOG2. These profiles are first averaged to eliminate wavelength dependence. Then the humidity dependencies in four aerosols are modeled using the following empirical relationship which was suggested in [6], based on the observation by Hanel [7],

$$X(RH) = c_1 (1 - RH/100)^{c_2}, \quad (16)$$

where c_1 and c_2 are model parameters. Noting that this relationship represents a straight line in log-log scale, optimal values for these parameters were obtained using the linear regression technique. This reduced the set of four profiles for each aerosol type to only two numbers. The same model is also used for the two humidity independent models, FOG1 and FOG2, by setting c_2 to 0 to eliminate humidity dependence.

Next, the visibility (VIS) dependent aerosol density profiles H are studied. In the first 2 (km) height, it is represented by three empirical functions of the visibility at 0, and 1 (km) altitudes. First, these profiles are fitted by the inverse relationship,

$$H(VIS) = d_1 VIS^{-1} + d_2, \quad (17)$$

which is used in LOWTRAN-6 for interpolation of the $H(.)$. Then, using the assumption that the typical altitude at which the OTDA is used is 300 (m) above sea level, the weighted average of the two profiles at 0 and 1 (km) heights is adopted as our model.

Finally, Eqs. (15), (16) and (17) are combined to form the

following aerosol model,

$$t = \exp\{-(\text{VIS}^{-1} + d_2') c_1' (1 - \text{RH}/100)^{c_2} R\}, \quad (18)$$

where d_1 is imbedded into c_1' by factoring it out to reduce the number of parameters. We note that d_2' is independent of the aerosol type.

2.5 Rain Model

Since the rain extinction model used in LOWTRAN-6 is a simple analytic function of the rain rate RR (mm/hr) and the range R (km), we can adopt it with a slight modification. After combining some parameters to minimize the number of constants, the model becomes as follows,

$$t = \exp\{-0.3647 RR^{0.63} R\}. \quad (19)$$

2.6 Summary of Model Equations

The model equations derived in the previous subsections are summarized in Table 1 together with the obtained optimum parameter values. Table 2 lists the input variables together with the definitions and default values where applicable.

Table 1. Preliminary Extinction Models for the OTDA**(1) Molecular resonant absorption**

$$t = \exp\{-A_0 (P/P_0)^{a_1} (T_0/T)^{a_2} U^{a_3}\}$$

Absorber	A ₀	a ₁	a ₂	a ₃
Water Vapor	0.0850	0.4981	0.2989	0.5582
	U = 0.1 WH R, WH = 0.01 RH F(T ₀ /T)			
Uniformly-mixed Gasses	0.0118	1.0792	0.8488	0.6178
	U = R			
Ozone	0.0076	0.3091	0.1541	0.7498
	U = 46.667 WO R			

$$F(s) = s \exp(18.9766 - 14.9595 s - 2.43882 s^2)$$

$$WO = 6.0E-05 \text{ (g/m}^3\text{)}.$$

(2) Water Vapor Continuum Absorption

$$t = \exp\{-C_0 [(P/P_0)(T_0/T) + (C_1 K_p + C_2) WH] WH U\}$$

$$C_0 = 1.655E-03 \quad C_1 = 0.5693 \quad C_2 = 0.5437$$

$$K_p = \begin{cases} \frac{1}{(296 - T)/(296 - 260)}, & T < 260 \\ 0, & 260 < T < 296 \\ , & 296 < T \end{cases}$$

(3) Aerosol Extinction

$$t = \exp\{-(VIS^{-1} + d_2') c_1' (1 - RH/100)^{c_2} R\}$$

#	Model	c ₁ '	c ₂	Default VIS(km)
1	RURAL	0.3670	-0.02877	23
2	URBAN	0.3119	-0.08499	5
3	OCEAN	0.4013	-0.3417	23
4	TROPOSPHERIC	0.08054	-0.04621	50
5	FOG1	4.487	0	0.2
6	FOG2	1.309	0	0.5

$$d_2' = -0.005183 \text{ (independent of aerosol type)}$$

(4) Rain Extinction

$$t = \exp\{-0.3647 RR^{0.63} R\}$$

Table 2. Input Variables for Extinction Models

Variable	Notation (Units)	Default
Pressure	P (mbar)	None
Temperature	T (°C)	None
Relative Humidity	RH (%)	None
Visibility	VIS (km)	*
Aerosol Model	IHAZE (integer)	0 (No Aerosol)
Rain Rate	RR (mm/h)	0.0
Range	R (km)	None

* See Table 1 for model dependent default values.

3. Accuracy Evaluation

A preliminary accuracy evaluation was done at AFWAL/AARI-3 while the principal investigator was visiting the laboratory. The obtained results were reported in [3]. For more comprehensive accuracy study, a data base was generated using LOWTRAN-6 according to the following scheme:

- a. For the molecular and water vapor continuum extinction, eight pressure values are used. Maximum and minimum pressure values of the 0 and 1 (km) layers of six atmosphere models in LOWTRAN-6 are first chosen. Then, six more intermediate values, evenly distributed between the two extremes, are selected. They are: 1018.0, 999.4, 980.8, 962.2, 943.6, 925.0, 906.4, and 887.8 (mbar).
- b. Eight temperature values are chosen for temperature dependent models similarly to a. The values are 35.0, 27.14, 19.29, 11.43, 3.57, -4.26, -12.14, and -20.0 (C).
- c. Eight humidity levels; 99, 95, 90, 85, 75, 50, 30 and 10 (percent) are used for humidity dependent models.
- d. Eight visibility values are used for aerosols. The values are chosen at and about the default visibility values for each aerosol model. Therefore, the actual values are dependent on the aerosol model.
- e. Eight range values were used where the values were chosen so that the resulting transmittances are as well-distributed in [0, 1] interval. For aerosol models, this implies that the range value is comparable to the default visibility values.

Although the numbers of variations in each input parameter is small, combinations of the above set of variations can be very large. To automate the process of data generation, an interactive computer program, called DGU (Data Generation Utility), was developed. It was written mostly in the C programming language. It uses a FORTRAN routine for post processing of LOWTRAN-6 generated data. (The listing of the DGU program is given in Appendix A.) Enough prompts are included in the program so that the user will need minimal effort in generating input data for LOWTRAN-6.

The program takes advantage of the UNIX operating system in creating input decks for LOWTRAN-6 automatically from the interactive inputs. It also runs LOWTRAN-6 for each set of input deck and post-processes the output data to compute the averaged transmittances. DGU repeatedly performs the combined task; generation of input deck, running LOWTRAN-6, and post-processing output data file, rather than creating all the input deck and then running LOWTRAN-6. This is because the latter method creates impractically large data files. The adopted method reduces the entire output of one LOWTRAN-6 run to a single line. Thus, the storage requirement is immensely reduced.

LOWTRAN-6 was run over 6000 times to generate data for all possible input value combinations. Outputs produced from this extensive computation were utilized for accuracy evaluation and stored for later usage as well. Results of accuracy evaluations are summarized in Table 3, in which the column MODEL 1 represents RMS errors of the developed models.

Table 3. Accuracy Evaluation of Extinction Models

TYPE OF EXTINCTION	MODEL 1		MODEL 1A	
	R.M.S. Error	# of +/- Err.	R.M.S. Error	# of +/- Err.
Water Vapor	0.0123	1834/2262		
Uniformly-Mixed Gasses	0.0007	2008/2088		
Ozone	0.0015	592/3504		
Water Vapor Continuum	0.0064	416/3680		
Aerosol	0.0097	217/295	0.0089	234/278
RURAL	0.0170	219/293	0.0164	236/276
OCEAN	0.0108	199/313	0.0089	231/281
URBAN	0.0063	265/247	0.0040	295/217
TROPOSPHERIC	0.0013	13/26*	0.0006	36/28
FOG1	0.0261	0/49*	0.0179	11/53
FOG2				

* Data points which resulted in zero transmittance by LOWTRAN computation are ignored.

4. Model Upgrading

Table 3 shows that the accuracy of most of the models are acceptably high, but some are not. Aerosol extinction models exhibit higher RMS errors. Especially, the RMS errors associated with RURAL, URBAN, and OCEAN models may be excessive for demanding applications. Note also that the numbers of positive (+) and negative (-) errors differ rather large in some models. This is not desirable since it indicates there exists a non-zero bias in the model. The accuracy of biased models can easily be improved by eliminating the biases. Based on these observations, we decided to upgrade all the aerosol models.

Model improvements start from the identification of causes of the observed error. In developing the aerosol extinction models, each intermediate quantity was separately modelled from the LOWTRAN-6 expressions and then combined. If better intermediate models are developed, then the total model accuracy should be improved. This approach leads us to, for example, quadratic models for the relative humidity dependent quantity X .

On the other hand, the component-wise modelling approach may be optimal for each individual component, but the combination of the individual components need not be optimal as a whole. This prompted the approach of obtaining optimal parameter values by directly minimizing the difference between model prediction and the LOWTRAN-6 computation. Keeping the model formulation derived in Section 2.4, namely Eq.(18), unchanged, the difference was minimized in terms of three parameters $c1'$, $c2$, and $d2'$ simul-

taneously. Since the model equation, Eq.(18), is nonlinear in the unknown parameters, the method of choice here is nonlinear optimization.

The difference between the LOWTRAN-6 and model transmittances are squared and summed to form a sum of squared errors which is to be minimized. A general approach for this minimization is to set the derivative of the sum of squared errors to be zero. However, if the model equation is nonlinear and complicated, derivation of derivative expressions is cumbersome and susceptible to error in analytical manipulation. Therefore, a subroutine called ZXSSQ of the IMSL library was adopted for this task because it directly minimizes the sum of squares without utilizing derivatives [8].

Table 4 summarizes the new optimal parameter values for aerosol models. Note that the parameter c_3 (formerly d_2') is no longer independent of the aerosol type. In Table 3, RMS errors corresponding to the new parameter set are listed in the MODEL 1A column. The new set of model parameters produces uniformly better modelling accuracies with the same number of parameters for each aerosol type. We concluded that the obtained models have sufficient accuracy to be adopted in the OTDA.

Table 4. Upgraded Aerosol Extinction Model

$$t = \exp\{ -c_1 (1 - RH/100)^{c_2} (VIS^{-1} + c_3) R \}$$

#	Model	c_1	c_2	c_3
1	RURAL	0.3250795	-0.0698031	-0.0062501
2	OCEAN	0.407210	-0.346923	-0.0073294
3	URBAN	0.377737	-0.010870	-0.0055683
4	TROPOSPHERIC	0.084286	-0.018607	-0.0055992
5	FOG1	4.475481	0	0.0051780
6	FOG2	1.082089	0	0.2241563

5. Model Implementation

The models listed in Table 1 with the upgraded parameters given in Table 4 are now ready to be integrated into a transmittance computation code. First, a FORTRAN program, called CTRAN (Compact TRANsmittance code), was developed. Outputs from CTRAN were compiled for climatological conditions which were used for data generation. As a final check of the validity of the models, the CTRAN transmittance values were compared with the average LOWTRAN-6 total transmittance computed within LOWTRAN-6. This is done using an error evaluation code called ILMAP (Interactive Lowtran Model Analysis Program). Tables 5-a through 5-g summarize the obtained error statistics. The overall result was an RMS error of 0.00463 in the total transmittance for 6,272 cases. This shows an excellent accuracy of the adopted models, considering the simplicity of them.

The models are then implemented on the HP-41CX. The programming language of the HP-41CX, which is based on the Reverse Polish notation, has been carefully studied for its use and capabilities. We developed two versions; one utilizes interactive inputs, and the other uses stored input values. The developed program is thoroughly tested by computing the transmittance values for the input combinations used in accuracy test data generation, and then by comparing it with the results from the FORTRAN version. A user's manual for the HP-41CX version of CTRAN is prepared and is listed in Appendix B. The listing of CTRAN, both the FORTRAN and HP-41CX versions, is in Appendix C.

Table 5. Error Analysis of CTRAN**5-a. Molecular and Continuum Extinctions**

MECHANISM	AVG % ERROR	RMS ERROR	POS/NEG
H2O	9.14734E-01	1.22954E-02	1834/2262
CO2	4.98500E-02	7.38009E-04	2008/2088
O3	9.97693E-02	1.45760E-03	592/3504
CONTINUUM	6.25526E+00	6.35660E-03	416/3680
TOTAL	5.85113E+00	4.58491E-03	558/3538

5-b. RURAL Aerosol Extinctions

MECHANISM	AVG % ERROR	RMS ERROR	POS/NEG
AEROSOL	3.42939E+00	8.38769E-03	234/278
H2O	3.34638E+00	2.75920E-02	176/336
CO2	4.86877E-02	6.78719E-04	256/256
O3	9.99364E-02	1.48416E-03	64/448
CONTINUUM	2.41252E+01	7.82923E-03	96/416
TOTAL	1.72796E+01	4.67787E-03	227/285

5-c. OCEAN Aerosol Extinction

MECHANISM	AVG % ERROR	RMS ERROR	POS/NEG
AEROSOL	7.27404E+00	1.64423E-02	236/276
H2O	3.34638E+00	2.75920E-02	176/336
CO2	4.86877E-02	6.78719E-04	256/256
O3	9.99364E-02	1.48416E-03	64/448
CONTINUUM	2.41252E+01	7.82923E-03	96/416
TOTAL	1.70245E+01	4.00414E-03	193/319

5-d. URBAN Aerosol Extinction

MECHANISM	AVG % ERROR	RMS ERROR	POS/NEG
AEROSOL	2.56816E+00	8.87016E-03	231/281
H2O	2.74367E+00	2.34464E-02	184/328
CO2	3.34411E-02	3.71156E-04	320/192
O3	6.56806E-02	8.48168E-04	128/384
CONTINUUM	2.47998E+01	7.94757E-03	80/432
TOTAL	1.85094E+01	3.74464E-03	220/292

Table 5. Error Analysis of CTRAN (contd.)**5-e. TROPOSPHERIC Aerosol Extinction**

MECHANISM	AVG % ERROR	RMS ERROR	POS/NEG
AEROSOL	3.22787E-01	3.98972E-03	295/217
H2O	1.00691E+01	5.51320E-02	56/456
CO2	3.43206E-01	4.85434E-03	128/384
O3	6.63520E-01	9.08007E-03	64/448
CONTINUUM	2.81415E+01	6.68107E-03	240/272
TOTAL	2.56393E+01	5.71252E-03	266/246

5-f. FOG1 Aerosol Extinction

MECHANISM	AVG % ERROR	RMS ERROR	POS/NEG
AEROSOL	4.07248E+00	6.15750E-04	36/28
H2O	1.63007E-01	1.79514E-03	40/24
CO2	6.69631E-02	6.69892E-04	0/64
O3	8.30899E-02	8.56288E-04	0/64
CONTINUUM	5.64428E-01	4.60394E-03	0/64
TOTAL	3.08853E+00	4.85773E-04	48/16

5-g. FOG2 Aerosol Extinction

MECHANISM	AVG % ERROR	RMS ERROR	POS/NEG
AEROSOL	5.49332E+00	1.79535E-02	11/53
H2O	3.04797E-01	2.74605E-03	64/0
CO2	5.73236E-02	5.86308E-04	0/64
O3	1.06208E-01	1.06103E-03	0/64
CONTINUUM	2.79759E+00	1.13680E-02	0/64
TOTAL	7.33842E+00	8.88518E-03	7/57

6. Conclusions and Directions for Future Study

6.1 Conclusions

Simple models for atmospheric extinctions due to various atmospheric absorption mechanisms were developed for the Operational Tactical Decision Aid. These models were for three molecular resonant absorptions due to the water vapor, uniformly-mixed gasses, and ozone, for water vapor continuum absorption, for aerosol extinction, and for rain extinction. All of those absorption mechanisms are active in the 8 - 12 (μm) band which is the primary band of sensitivity for infrared sensors.

Set of preliminary models were developed from LOWTRAN-6 by carefully simplifying the structures of the LOWTRAN-6 models. The preliminary models were extensively compared with LOWTRAN-6 computations for various combinations of input values. An interactive program, DGU, was written to automate the data generation for accuracy testing. It generates LOWTRAN-6 input deck sets for all possible combinations of given input values, runs LOWTRAN-6, and post-processes the LOWTRAN-6 generated data files.

Acceptable agreements were found in most models with the exceptions of the aerosol models. As a result, aerosol models were upgraded using the direct minimization of the discrepancies between the model predicted transmittances and the LOWTRAN-6 results.

The final models with their optimal parameter values were integrated into a compact transmittance computation code, called

CTTRAN, in the FORTRAN language for mainframe computers. CTRAN was then coded onto an HP-41CX using the HP-41CX programming language. The program was thoroughly tested against the data set used for accuracy testing. The HP-41CX version of CTRAN is now ready to be employed in the OTDA.

6.2 Directions for Future Study

The models obtained in this project are very simple. All of them are simple analytical functions with a small number of parameters. This was necessary because of the severe restriction of the computational capability of the HP-41CX hand-held computer. It is well understood that the portability is of utmost importance for the OTDA, and that this leads to the employment of HP-41CX computer. However, the availability of powerful but truly portable microcomputers is rapidly increasing. Newer machines are smaller, faster, of higher memory capacity, and longer operation time between rechargings. The choice of the HP-41CX for OTDA may need to be reconsidered. The capability of the HP-41CX can hardly be compared with that of a microcomputer. Since the portability of the two do not differ significantly, availability of high level languages alone may be a sufficient advantage of microcomputers. Also, because of the large memory capacity, a microcomputer can store the entire OTDA in a cluster of programs and data files. Therefore, it is highly recommended that a portable microcomputer instead of an HP-41CX be used for the OTDA.

The advantage of higher memory capacity of microcomputers can be exploited to employ more accurate but complicated atmos-

pheric extinction models to the OTDA. It is even conceivable to add another set of models for 3 - 5 (μm) spectral band, which is also used by some infrared detectors. Furthermore, it should be possible to implement a program which generates simplified transmittance profiles for the two spectral regions. This point is currently under investigation.

REFERENCES

- [1] Kneizys, F.X., E.P. Shettle, W.O. Gallery, J.H. Chetwynd, Jr., L.W. Abreu, J.E.A. Selby, R.W. Fenn, and R.A. McClatchey 1980. "Atmospheric Transmittance/Radiance: Computer Code LOWTRAN 5," Report AFGL-TR-80-0067, Air Force Geophysics Laboratory, Hanscom Air Force Base, Mass.
- [2] Kneizys, F.X., E.P. Shettle, W.O. Gallery, J.H. Chetwynd, Jr., L.W. Abreu, J.E.A. Selby, S.A. Clough, and R.W. Fenn 1983. "Atmospheric Transmittance/Radiance: Computer Code LOWTRAN 6," Report AFGL-TR-83-0187, Air Force Geophysics Laboratory, Hanscom Air Force Base, Mass.
- [3] Tomiyama, K. 1986. "Atmospheric Modeling for Operational Tactical Decision Aid," Final Report, Contract Number F49620-85-C-0013, Air Force Office of Scientific Research.
- [4] Gruenzel, R.R. 1978. "Mathematical Expressions for Molecular Absorption in LOWTRAN 3B," Applied Optics, 17, pp 2591-2593.
- [5] Pierluissi, J.H., K. Tomiyama, and R.G. Gomez 1979. "Analysis of LOWTRAN Transmission Functions," Applied Optics, 18, pp 1607-1612.
- [6] Shettle, E.P., and R.W. Fenn 1979. "Models for the Aerosols of the Lower Atmosphere and the Effects of Humidity Variations on Their Optical Properties," Report AFGL-TR-79-0214, Air Force Geophysics Laboratory, Hanscom Air Force Base, Mass.
- [7] Hanel, G. 1976. "The Properties of Atmospheric Aerosol Particles as Functions of the Relative Humidity at Thermodynamic Equilibrium with the Surrounding Moist Air," Advances in Geophysics, Ed. by H.E. Landsberg and J. Van Mieghem, Academic Press, New York, 19, pp 73-188.
- [8] IMSL 1984. IMSL User's Manual, V.4, Chapter Z, pp ZXSSQ1-ZXSSQ7.

Appendix A. Data Generation Utility (DGU) Program

The DGU program is listed in this appendix. DGU consists of two parts; DGU.C, a C-language main program, and AVERAGE.F, a FORTRAN program. DGU.C generates a LOWTRAN-6 input deck, executes LOWTRAN-6, calls AVERAGE to post-process the output of LOWTRAN-6, and repeats this process until all combinations of input variables have been exhausted. A part of a typical output of DGU is also listed here.¹

¹ A computer tape which contains DGU and the FORTRAN version of CTRAN was prepared and was submitted separately to the effort focal point.

Table A-1. Listing of DGU**-- DGU.C in C Language --**

```

#include <stdio.h>
#include <ctype.h>

#define STRING "lowdata"          /* defines shell program to be
                                   executed by the system call */

/*      #undef    goes to screen      */
/*      #define   creates files       */
#define FILE_MODE    /* create and write files      */

float hum[9] = {0.0, 99.0, 95.0, 90.0, 85.0, 70.0, 50.0,
                30.0, 10.0};
float temp[9] = {0.0, 35.00, 27.14, 19.29, 11.43, 3.57,
                 -4.29, -12.14, -20.00};
float pr[9] = {0.0, 1018.0, 999.4, 980.8, 962.2, 943.6,
               925.0, 906.4, 887.8};

main()
(
    int i1, i2, i3, i4, i5;

    FILE *fp;
    char name2[40];

    float max_t, min_t, max_p, min_p, step;
    float range[9], vis[9];
    char buff[20], buff1[20], string1, name1[20];
    int i, j, irange, itemp, ipress, ihum, ivis, ihaze, itotal;

    system("clear");
    printf("\n\n\n\n\n\n\n");
    sprintf(buff, "banner 'ILMAP/DGU'\n\n\n\n\n");
    system(buff);
    printf(" Interactive LOWtran Model Analysis Program and Data Generation Utilit
y\n");
    printf("\n\n\n\n\n\n\n\n\n");
    printf("                      Press <return> to begin the data generation\n");
    getchar();
    system("clear");

    printf("You must first choose the type of AEROSOL computation");
    printf(" that you wish to perform. The choices are:\n\n");
    printf("0. ----- NO AEROSOL COMPUTATION ----- \n");
    printf("1. RURAL Model           Default VISIBILITY = 23.0 KM\n");
    printf("2. RURAL Model           Default VISIBILITY = 5.0 KM\n");
    printf("3. OCEAN Model           Default VISIBILITY = 23.0 KM\n");
    printf("4. URBAN Model           Default VISIBILITY = 5.0 KM\n");
    printf("5. TROPOSPHERIC Model    Default VISIBILITY = 50.0 KM\n");
    printf("6. FOG1 Model (Advection) Default VISIBILITY = 0.2 KM\n");

```

-- DGU.C in C Language (contd.) --

```

printf("7. FOG2 Model (Radiation)   Default VISIBILITY = 0.5 KM\n");
printf("\n");
printf("Type 0,1,2,3,4,5,6 or 7 followed by <return>\n\n");
scanf("%d", &ihaze);
getchar();
if (ihaze == 0) sprintf(name1, "molabs.data");
if (ihaze == 1) sprintf(name1, "rural.data");
if (ihaze == 2) sprintf(name1, "rural.data");
if (ihaze == 3) sprintf(name1, "ocean.data");
if (ihaze == 7) (ihaze = 9; sprintf(name1, "fog2.data")););
if (ihaze == 6) (ihaze = 8; sprintf(name1, "fog1.data")););
if (ihaze == 5) (ihaze = 6; sprintf(name1, "tropo.data")););
if (ihaze == 4) (ihaze = 5; sprintf(name1, "urban.data")););
printf("\n\n");
printf("IHAZE = %d\n\n", ihaze);
printf("The file which contains the output data will be (%s", name1);
printf(")\n\n\n\n\n");
printf("                                Press <return> to continue ... \n");
getchar();
system("clear");
printf("Enter the desired minimum TEMPERATURE (C), followed by <return>\n\n");
scanf("%10f", &min_t);
getchar();
printf("\n\n");
printf("Enter the desired maximum TEMPERATURE (C), followed by <return>\n\n");
scanf("%10f", &max_t);
getchar();
printf("\n\n");
printf("Maximum TEMPERATURE is %10.3f\n\nMinimum TEMPERATURE is %10.3f\n\n",
      max_t, min_t);
if (max_t == min_t) {
    printf("\n\n");
    printf("You have chosen the same values for maximum and minimum \n");
    printf("TEMPERATURE. One value will be used \n\n");
    itemp = 1;
} else {
    printf("How many total TEMPERATURE points do you wish to use?\n\n");
    printf("Example: 3 means maximum, minimum and the");
    printf(" midpoint between the two.\n\n");
    printf("Enter the total number of TEMPERATURE points followed by <return>.");
    printf(" It MUST be\n ");
    printf("two or more but no more than 8.\n\n");
    scanf("%d", &itemp);
    getchar();
    step = (max_t - min_t) / (itemp - 1);
    printf("\n\n");
    printf("TEMPERATURE values :   (C)\n\n");
    for (j=1; j<=itemp; j++) {
        temp[j] = min_t + (j - 1.0)*step;
        printf("%d:      %7.2f\n", j, temp[j]);
    }
}

```

-- DGU.C in C Language (contd.) --

```

printf("\n\n");
printf("                                Press <return> to continue ...\n\n");
getchar();
system("clear");
printf("Enter the desired minimum PRESSURE (mbar), followed by <return>\n\n");
scanf("%10f", &min_p);
getchar();
printf("\n\n");
printf("Enter the desired maximum PRESSURE (mbar), followed by <return>\n\n");
scanf("%10f", &max_p);
getchar();
printf("\n\n");
printf("Maximum PRESSURE is %10.3f\n\nMinimum PRESSURE is %10.3f\n\n",
      max_p, min_p);
if (max_p == min_p) {
    printf("\n\n");
    printf("You have chosen the same values for maximum and minimum \n");
    printf("PRESSURE. One value will be used \n\n");
    ipress = 1;
} else {
    printf("How many total PRESSURE points do you wish to use?\n\n");
    printf("Enter the total number followed by <return>.");
    printf(" It MUST be\n ");
    printf("two or more but no more than 3 \n\n");
    scanf("%d", &ipress);
    getchar();
    step = (max_p-min_p)/(ipress-1);
    printf("\n\n");
    printf("PRESSURE values : (mbar)\n\n");
    for (j=1; j<=ipress; j++) {
        pr[j] = min_p + (j - 1.0)*step;
        printf("%d:      %7.2f\n", j, pr[j]);
    }
}
printf("\n\n");
printf("                                Press <return> to continue ...\n\n");
getchar();

printf("Enter the number of VISIBILITY values you wish to use.\n\n");
printf("It should be no more than eight and you must choose\n");
printf("at least one. Enter number followed by <return>.\n\n");
scanf("%d", &ivis);
getchar();
printf("\n\n");
printf("Enter the %d desired VISIBILITY (km) values each followed by <return>\n\n", ivis);
printf("\n\n");

for (j=1; j<=ivis; j++) {
    scanf("%10f", &vis[j]);
    getchar();
}
printf("\n\n");

```

-- DGU.C in C Language (contd.) --

```

printf("VISIBILITY values : (km)\n\n" );
for (j=1; j<=ivis; j++)
    printf("%d. %7.3f\n", j, vis[j]);
printf("\n\n");
printf("                                Press <return> to continue ...\n\n");
getchar();
system("clear");

printf("Do you want to change the default HUMIDITY values? (y,n) \n\n");
scanf("%c", &string1);
printf("\n\n");
getchar();
if (tolower(string1) == 'y') (
    printf("\n\n");
    printf("The current values for HUMIDITY are as follows :\n\n\n");
    printf("HUMIDITY values : (%%)\n\n" );
    for (i=1; i<=8; i++)
        printf("%d. %7.3f\n", i, hum[i]);
    printf("\n\n");
    printf("How many HUMIDITY value(s) do you wish to use?\n\n");
    printf("You should choose no more than 8 and at least 1\n\n");
    printf("Enter number followed by <return>\n\n");
    scanf("%d", &ihum);
    printf("\n\n");
    printf("Enter the %d HUMIDITY (%%) value(s) each followed by <return>.\n\n",
    ihum);
    for (i=1; i<=ihum; i++) (
        scanf("%10f", &hum[i]);
        getchar(); )
    ) else (
        ihum = 8;
    )
    printf("\n\n");
    printf("HUMIDITY values : (%%)\n\n");
    for (j=1; j<=ihum; j++)
        printf("%d. %7.3f\n", j, hum[j]);
    printf("\n\n");

    printf("                                Press <return> to continue ...\n\n");
    getchar();

    system("clear");
    printf("Enter the number of RANGE values you wish to use.\n\n");
    printf("It should be no more than eight and you must choose\n");
    printf("at least one. Enter number followed by <return>.\n\n");
    scanf("%d", &irange);
    printf("\n\n");
    printf("Enter the %d desired RANGE (km) values each followed by <return>\n\n",
    irange);
    printf("\n\n");

    for (j=1; j<=irange; j++) (

```

-- DGU.C in C Language (contd.) --

```

        scanf("%10f", &range[j]);
        getchar(); }
printf("\n\n");
printf("RANGE values : (km)\n\n" );
for (j=1; j<=irange; j++)
    printf("%d. %7.3f\n", j, range[j]);
printf("\n\n");
printf("                                Press <return> to continue ...\n\n");
getchar();

j = 1;

itotal = ipress*itemp*ihum*irange*ivis;

for(i1=1;i1<=ipress;i1++)
    for(i2=1;i2<=itemp;i2++)
        for(i3=1;i3<=ihum;i3++)
            for(i4=1;i4<=irange;i4++)
                for(i5=1;i5<=ivis;i5++)
                    (
#ifdef FILE_MODE
                        sprintf (name2, "input.card", j++);
                        fp = fopen(name2, "w");
                        if (fp == NULL)
                            exit(1);
                        /*
                        printf ("Creating file %s\n", name2); */

fprintf(fp,
"      0      1      0      0      0      0      0      0      0.000      0.000\n");
fprintf(fp,
"      %d      0      0      0      0      0x10.3f      0.000      0.000      0.000\n",
        ihaze, vis[i5]);
fprintf(fp,
"      0.300x10.3fx10.3f 0.0x5.1f 0.0E+00 6.0E-05x10.3f\n",
/*123456789          12345          12345678901234567890 */
        pr[i1], temp[i2], hum[i3], range[i4]);
fprintf(fp, "      830.000 1250.000      5.000\n");
fprintf(fp, "      0\n");
/*
        printf ("Closing File %s\n", name2); */
        printf ("COMPUTATION NUMBER %5d of %6d\n",j-1,itotal);
        fclose(fp);
        system (STRING);
#else
        printf("%d:\t %10.4f\t%10.4f\t%5.1f\t%10.4f\t%10.4f\n",
            j++, pr[i1], temp[i2], hum[i3], range[i4], vis[i5]);
#endif

        ) /* end for          */

sprintf(buff1,"mv test.data %s",name1);
system(buff1);
printf("The data for ilmap is located in file %s\n", name1);
)

```

-- AVERAGE.F in FORTRAN used by DGU.C --

```

dimension h2o(101),co2(101),o3(101),cont(101),aero(101)
open(12,file='temp1',status='old',access='sequential',form=
.'formatted')
rewind 12
open(13,file='temp2',status='old',access='sequential',form=
.'formatted')
rewind 13
open(14,file='temp3',status='new',access='sequential',form=
.'formatted')
rewind 14
open(15,file='output.data',status='old',access='sequential',
.form='formatted')
rewind 15
npts=85
read(15,96) trans
96 format(t25,f6.4)
read(12,97) junk
97 format(i5)
read(12,98) ihaze,i,j,k,l,m,vis,c1,c2,rnrt
98 format(6i5,4f10.3)
read(12,99) h1,p,t,dp,h,wh,wo,r
99 format(3f10.3,2f5.1,2e10.3,f10.3)
read(13,101) (h2o(i),co2(i),o3(i),cont(i),aero(i),i=1,npts)
101 format(t2,f7.4,t10,f7.4,t18,f7.4,t28,f7.4,t38,f7.4)
sum1=0.0
sum2=0.0
sum3=0.0
sum4=0.0
sum5=0.0
sum6=0.0
do 12 i=2,84
sum1=sum1+h2o(i)
sum2=sum2+co2(i)
sum3=sum3+o3(i)
sum4=sum4+cont(i)
sum5=sum5+aero(i)
12 continue
sum1=sum1+0.5*(h2o(1)+h2o(84))
sum2=sum2+0.5*(co2(1)+co2(84))
sum3=sum3+0.5*(o3(1)+o3(84))
sum4=sum4+0.5*(cont(1)+cont(84))
sum5=sum5+0.5*(aero(1)+aero(84))
sum1=sum1/84.0
sum2=sum2/84.0
sum3=sum3/84.0
sum4=sum4/84.0
sum5=sum5/84.0
write(14,19) ihaze,vis,p,t,h,r,sum1,sum2,sum3,sum4,sum5,trans
19 format(i2,f7.3,f8.2,f7.2,f5.1,f7.3,6f7.4)
endfile 14
stop
end

```

Table A-2. Listing of a Typical Output of DGU (Partial)

The values are, from left to right, IHAZE, visibility, pressure, temperature, relative humidity, range, and transmittances due to water vapor, uniformly-mixed gasses, ozone, water vapor continuum, aerosol, and the total transmittance. This data set has IHAZE = 9. This implies that the FOG2 aerosol model is used.

9	.200	1018.00	35.00	99.0	.200	.9296	.9968	.9987	.7519	.3071	.2152
9	.300	1018.00	35.00	99.0	.200	.9296	.9968	.9987	.7519	.4518	.3158
9	.400	1018.00	35.00	99.0	.200	.9296	.9968	.9987	.7519	.5495	.3835
9	.500	1018.00	35.00	99.0	.200	.9296	.9968	.9987	.7519	.6186	.4314
9	.600	1018.00	35.00	99.0	.200	.9296	.9968	.9987	.7519	.6697	.4667
9	.700	1018.00	35.00	99.0	.200	.9296	.9968	.9987	.7519	.7089	.4938
9	.850	1018.00	35.00	99.0	.200	.9296	.9968	.9987	.7519	.7530	.5243
9	1.000	1018.00	35.00	99.0	.200	.9296	.9968	.9987	.7519	.7856	.5468
9	.200	1018.00	35.00	99.0	.300	.9118	.9957	.9980	.6535	.1745	.1043
9	.300	1018.00	35.00	99.0	.300	.9118	.9957	.9980	.6535	.3072	.1830
9	.400	1018.00	35.00	99.0	.300	.9118	.9957	.9980	.6535	.4101	.2438
9	.500	1018.00	35.00	99.0	.300	.9118	.9957	.9980	.6535	.4886	.2902
9	.600	1018.00	35.00	99.0	.300	.9118	.9957	.9980	.6535	.5497	.3261
9	.700	1018.00	35.00	99.0	.300	.9118	.9957	.9980	.6535	.5981	.3546
9	.850	1018.00	35.00	99.0	.300	.9118	.9957	.9980	.6535	.6544	.3877
9	1.000	1018.00	35.00	99.0	.300	.9118	.9957	.9980	.6535	.6971	.4128
9	.200	1018.00	35.00	99.0	.400	.8969	.9947	.9973	.5687	.1006	.0514
9	.300	1018.00	35.00	99.0	.400	.8969	.9947	.9973	.5687	.2104	.1072
9	.400	1018.00	35.00	99.0	.400	.8969	.9947	.9973	.5687	.3073	.1562
9	.500	1018.00	35.00	99.0	.400	.8969	.9947	.9973	.5687	.3870	.1965
9	.600	1018.00	35.00	99.0	.400	.8969	.9947	.9973	.5687	.4521	.2292
9	.700	1018.00	35.00	99.0	.400	.8969	.9947	.9973	.5687	.5055	.2561
9	.850	1018.00	35.00	99.0	.400	.8969	.9947	.9973	.5687	.5693	.2882
9	1.000	1018.00	35.00	99.0	.400	.8969	.9947	.9973	.5687	.6190	.3131
9	.200	1018.00	35.00	99.0	.500	.8841	.9938	.9967	.4957	.0587	.0257
9	.300	1018.00	35.00	99.0	.500	.8841	.9938	.9967	.4957	.1451	.0633
9	.400	1018.00	35.00	99.0	.500	.8841	.9938	.9967	.4957	.2312	.1007
9	.500	1018.00	35.00	99.0	.500	.8841	.9938	.9967	.4957	.3074	.1337
9	.600	1018.00	35.00	99.0	.500	.8841	.9938	.9967	.4957	.3725	.1619
9	.700	1018.00	35.00	99.0	.500	.8841	.9938	.9967	.4957	.4277	.1857
9	.850	1018.00	35.00	99.0	.500	.8841	.9938	.9967	.4957	.4957	.2151
9	1.000	1018.00	35.00	99.0	.500	.8841	.9938	.9967	.4957	.5500	.2384
9	.200	1018.00	35.00	99.0	.600	.8727	.9929	.9961	.4326	.0347	.0130
9	.300	1018.00	35.00	99.0	.600	.8727	.9929	.9961	.4326	.1006	.0377
9	.400	1018.00	35.00	99.0	.600	.8727	.9929	.9961	.4326	.1746	.0653
9	.500	1018.00	35.00	99.0	.600	.8727	.9929	.9961	.4326	.2448	.0914
9	.600	1018.00	35.00	99.0	.600	.8727	.9929	.9961	.4326	.3075	.1147
9	.700	1018.00	35.00	99.0	.600	.8727	.9929	.9961	.4326	.3625	.1352

Appendix B. CTRAN User's Manual

CTRAN/PTRAN manual which explains how to use the two HP-41CX versions of CTRAN, one interactive and the other batch, is listed here.

PTRAN / CTRAN USER'S MANUAL

PTRAN and CTRAN are programs which run on the HP-41CX handheld computer. Both of these programs compute the average transmittance of the atmosphere in the 8- to 12-micron region of the infrared spectrum. The full model analysis and development was discussed in this report. This manual is designed to instruct the user in the operation of PTRAN and CTRAN. The programs themselves are stored on a single mini cassette tape. The user must have access to Hewlett-Packard's Digital Cassette Drive for program operation. The general procedure to run a stored program and to use a mini data cassette tape are explained in the respective HP manuals, and are not discussed here.

The program CTRAN (Compact low-resolution TRANsmittance computation code) is in modular form suitable for batch computation while PTRAN is in interactive form. The first time user may want to use PTRAN until he/she becomes familiar with the data storage and recall locations. For the remainder of this manual, CTRAN will be used to mean either program.

Both CTRAN and PTRAN are smaller versions of parent LOWTRAN-6. LOWTRAN-6 is an exhaustive computation code which requires on the average 35 input variables. Since the CTRAN computation is performed on a very small part of the spectrum, only certain atmospheric conditions play a part in the transmittance calculation. We have shown that the number of relevant input variables to CTRAN is 7. Those variables are Pressure, Temperature, IHAZE (used for choosing the desired aerosol model), Relative Humidity,

Rain Rate, Visibility and Range. The variable names appear as the concatenation of the capital letters in their names. For instance, the rain rate variable is RNRT, etc ...

The HP-41CX stores data and programs in registers in its internal memory. The data registers are numbered from 00-99. CTRAN only uses registers 00-18 for input variable storage, intermediate computations storage, data holding and output transmittance storage. The following is a break-down of the individual register contents:

REGISTER	STORES	REGISTER	STORES
00	VIS	09	TH20
01	P	10	TCO2
02	T	11	TO3
03	RH	12	TCONT
04	RNRT	13	TRAIN
05	R	14	TTOTAL
06	PBAR	15	WH
07	TBAR	16, 17, & 18	temp
08	TAERO		

where

VIS	= Visibility in km
P	= Pressure in mbar
T	= Temperature in C
RH	= Relative humidity in %
RNRT	= Rain rate in mm/hr
R	= Range in km

PBAR	= P/P0 with P0=1013.25 mbar
TBAR	= T0/T with T0=273.15 C
TAERO	= Aerosol transmittance
TH2O	= Water Vapor absorption transmittance
TCO2	= Uniformly-mixed gasses absorption transmittance
TO3	= Ozone absorption transmittance
TCONT	= Water Vapor continuum absorption transmittance
TRAIN	= Rain absorption transmittance
TTOTAL	= Total transmittance
WH	= A value computed for the continuum calculation
temp	= temporary storage of intermediate results

The aerosol extinction model is chosen through the value of IHAZE. The definition of IHAZE is as follows:

0	= No aerosol computation (TAERO=1.0000)
1	= Ocean model
2	= Rural model
3	= Tropospheric model
4	= Urban model
5	= Fog1 (Advection) model
6	= Fog2 (Radiation) model

The user specifies the value of IHAZE by setting a user flag.

To run CTRAN, the six input variables need to be stored in the respective memory locations, and the value of IHAZE needs to be given as a set flag. On the other hand, PTRAN can be run by simply following the prompts. The detailed instructions for running each program are given next.

(1) RUNNING PTRAN

The following steps are necessary to run PTRAN.

1. Load PTRAN into main memory by doing the following:
 - (a) Store the string "PTRAN" in the alpha register
 - (b) execute the program READP
2. Clear all the user flags! This is most important, as only one user flag must be set during the execution of PTRAN or erroneous results could occur. The user flags are flags 00-07. After pressing CF (Clear flag), type 00. Press CF, followed by 01, and so on to 07.
3. Execute PTRAN
4. After each prompt, enter the desired value followed by the R/S key.
5. The program will run automatically after each input has been typed in. The program is running when the arrows move across the HP-41CX display.
6. The total transmittance will be displayed. The total transmittance is also stored in register 14.
7. Use the chart and RCL function to view other results.

For example, if you want to look at the continuum transmittance, type RCL and 12 when RCL prompts you. Also, if you want to save some results from the computation, you can store them somewhere outside the register range of PTRAN, i.e., store it in register 19 or whatever you like after register 18. Also, if you want to run PTRAN twice and compare the total transmittances, you may recall (RCL) register 14 and store (STO) it in a register that is

higher than 18. Then run PTRAN the second time and the total transmittance for the second computation will now be in register 14. Two results can be compared by recalling them from the respective memory registers.

(2) RUNNING CTRAN

1. Load CTRAN into main memory by doing the following:
 - (a) Put the string "CTRAN" into the alpha register
 - (b) Execute the program READP
2. Use the charts to store the desired input values in the corresponding registers. For example, if you want to have the range as 10.0 (km), store 10.0 in register 05.
3. Clear all the user flags! (See Instruction 2 of PTRAN.)
4. Having cleared all the user flags 00-07 in step 3 you must now set the flag which corresponds to the desired aerosol model. Use the aerosol IHAZE chart to find the correct flag to set. For example, if the tropospheric model is desired, set flag 03 by typing SF and 03 when SF prompts you.
5. The total transmittance will be displayed and the other values may be displayed by the same method as described in step 7 and what follows of the PTRAN instructions.

If you have any questions or suggestions about the operation of PTRAN or CTRAN please contact:

Dennis W. Richardson
201 Electrical Engineering East
University Park, PA 16802
(814) 863-3211

Appendix C. CTRAN Programs

The CTRAN; the FORTRAN version, and the two HP-41CX versions, are listed in this appendix.

Table C-1. Listing of the FORTRAN Version of CTRAN

```

PROGRAM CTRAN
c
c   FORTRAN version
c
c   This program computes the average transmittance over
c   8 - 12 (micron) band for a given set of inputs.
c   The inputs are:
c       pressure           (mbar)
c       temperature        (C)
c       relative humidity  (%)
c       visibility         (km)
c       rain rate          (mm/Hr)
c       and
c       path length        (km)
c
c   For questions and comments, please contact
c       Ken Tomiyama
c       121 Electrical Engineering East
c       Penn State University
c       University Park, PA 16802
c       (814) 865-7667
c
c   dimension c1(9), c2(9), c3(9)
c   data c1/0.3250795, 0.3250795, 0.0, 0.407210, 0.377737,
1      0.084286, 0.0, 4.475481, 1.082089/
c   data c2/-0.0698031, -0.0698031, 0.0, -0.346923, -0.010870,
1      -0.018607, 0.0, 0.0, 0.0/
c   data c3/-0.0062501, -0.0062501, 0.0, -0.0073294, -0.0055683,
1      -0.0055992, 0.0, 0.0051780, 0.2241563/
c
c   print *, '***** CTRAN (FORTRAN version) *****'
c   print *, ' '
c   print *, 'This program computes the average transmittance over'
c   print *, '8 - 12 (micron) spectral region.'
c   print *, ' '
c   print *, 'Start of Input Session'
c   print *, ' '
c   print *, 'Do you want intermediate results (0=no)?'
c       read(5,*) iout
c       if (iout .eq. 0) then
c           print *, 'No intermediate result printed (iout=0).'
c       else
c           print *, 'Intermediate results are stored in int.dat file.'
c           open(unit=8, file='int.dat', status='unknown')
c       endif
10 continue
c   print *, ' '
c   print *, 'aerosol types (ihaze) are identical to LOWTRAN-6 a'
c   print *, '      no aerosol      (ihaze=0)'
c   print *, '      rural          (ihaze=1 or 2)'
c   print *, '      ocean            (ihaze=4)'

```

Table C-1. Listing of CTRAN program (contd.)

```

print *, '      urban      (ihaze=5)'
print *, '      tropospheric (ihaze=6)'
print *, '      fog1      (ihaze=8)'
print *, '      fog2      (ihaze=9)'
print *, '      (ihaze = 3 and 7 are not used and reset to 0.)'
print *, ' '
print *, 'aerosol type =?'
print *, '[ihaze < 0 or ihaze > 9 to stop]'
  read(5,*) ihaze
  if ((ihaze .lt. 0) .or. (ihaze .gt. 9)) stop
print *, 'visibility (km) =?'
  read(5,*) vis
print *, 'pressure (mbar) =?'
  read(5,*) pres
  pbar = pres / 1013.25
print *, 'temperature (C) =?'
  read(5,*) tempc
  tempk = tempc + 273.15
  tbar = 273.15 / tempk
print *, 'relative humidity (%) =?'
  read(5,*) rhump
  rhum = rhump / 100.0
print *, 'rain rate (mm/Hr) =?'
  read(5,*) rr
print *, 'path length (km) =?'
  read(5,*) range

c
c-----start of transmittance computation-----
c
c  aerosol
c
  taero = 1.0000
  if (ihaze .eq. 0) go to 20
  if (ihaze .le. 6) then
    xx1 = c1(ihaze) * (1.0 - rhum)**c2(ihaze)
  else
    xx1 = c1(ihaze)
  endif
  zhaze = (1.0 / vis) + c3(ihaze)
  taero = exp(-xx1 * zhaze * range)
20 continue

c
c  water vapor
c
  wh = rhum * tbar * exp(18.9766 - 14.9596*tbar - 2.43882*tbar**2)
  uh2o = 0.1 * wh * range
  eh2o = 0.08500 * pbar**0.4981 * tbar**0.2989 * uh2o**0.5582
  th2o = exp(-eh2o)

c
c  uniformly-mixed gasses
c

```

Table C-1. Listing of CTRAN program (contd.)

```

eco2 = 0.01164 * pbar**1.0792 * tbar**0.8488 * range**0.6178
tco2 = exp(-eco2)
c
c ozone
c
eo3 = 0.007561 * pbar**0.3091 * tbar**0.1541 * range**0.7498
to3 = exp(-eo3)
c
c water vapor continuum
c
a3 = 0.0
if (tempk .lt. 296.0) a3 = (296.0 - tempk) / 36.0
if (tempk .lt. 260.0) a3 = 1.0
econt = 1.655e-03 * (pbar * tbar + (0.5693 * a3 + 0.5437) * wh)
1      * wh * range
tcont = exp(-econt)
c
c rain
c
erain = 0.3647 * rr**0.63 * range
train = exp(-erain)
c
c total transmittance
c
trans = taero * th2o * tco2 * to3 * tcont * train
c
write(6,200) trans
200 format(5x,'TOTAL TRANSMITTANCE =',f7.4)
c
c-----returning point-----
c
if (iout .eq. 0) go to 10
write(8,100) ihaze, vis, pres, tempc, rhump, rr, range,
1      taero, th2o, tco2, to3, tcont, trans
100 format(i3,f6.2,f7.1,f6.1,2f5.1,f6.2,6f7.4)
go to 10
end

```

Table C-2. Listing of the HP-41CX Version of CTRAN**C-2.a. PTRAN - Interactive Version**

001♦LBL "PTRAN"	046 FS? 03	091 RCL 05
002 2.0	047 GTO "TROPO"	092 *
003 "IHAZE = ?"	048 FS? 04	093 E^X
004 PROMPT	049 GTO "URBAN"	094 STO 08
005 Y^X	050 FS? 05	095 GTO "H2O"
006 X<>F	051 GTO "FOG1"	096♦LBL "TROPO"
007 "VIS = ?"	052 FS? 06	097 RCL 14
008 PROMPT	053 GTO "FOG2"	098 -0.0186073
009 STO 00	054♦LBL "NONE"	099 Y^X
010 "P = ?"	055 1.0000	100 0.0842857
011 PROMPT	056 STO 08	101 *
012 STO 01	057 GTO "H2O"	102 STO 15
013 1013.25	058♦LBL "OCEAN"	103 RCL 00
014 /	059 RCL 14	104 1/X
015 STO 06	060 -0.3469229	105 -0.0055992
016 273.15	061 Y^X	106 +
017 "T = ?"	062 0.4072999	107 RCL 15
018 PROMPT	063 *	108 *
019 STO 02	064 STO 15	109 CHS
020 +	065 RCL 00	110 RCL 05
021 STO 18	066 1/X	111 *
022 273.15	067 -0.0073296	112 E^X
023 RCL 18	068 +	113 STO 08
024 /	069 RCL 15	114 GTO "H2O"
025 STO 07	070 *	115♦LBL "URBAN"
026 "RH = ?"	071 CHS	116 RCL 14
027 PROMPT	072 RCL 05	117 -0.0108702
028 STO 03	073 *	118 Y^X
029 -100.00	074 E^X	119 0.3777368
030 /	075 STO 08	120 *
031 1.0000	076 GTO "H2O"	121 STO 15
032 +	077♦LBL "RURAL"	122 RCL 00
033 STO 14	078 RCL 14	123 1/X
034 "RNRT = ?"	079 -0.0698031	124 -0.0055683
035 PROMPT	080 Y^X	125 +
036 STO 04	081 0.3250795	126 RCL 15
037 "RANGE = ?"	082 *	127 *
038 PROMPT	083 STO 15	128 CHS
039 STO 05	084 RCL 00	129 RCL 05
040 FS? 00	085 1/X	130 *
041 GTO "NONE"	086 -0.0062501	131 E^X
042 FS? 01	087 +	132 STO 08
043 GTO "OCEAN"	088 RCL 15	133 GTO "H2O"
044 FS? 02	089 *	134♦LBL "FOG1"
045 GTO "RURAL"	090 CHS	135 RCL 00

C-2.a. PTRAN - Interactive Version (contd.)

136 1/X	181 STO 15	226♦LBL "O3"
137 0.0051780	182 RCL 05	227 RCL 05
138 +	183 *	228 0.7498
139 4.4754801	184 0.100	229 Y`X
140 *	185 *	230 STO 16
141 CHS	186 0.5582	231 RCL 07
142 RCL 05	187 Y`X	232 0.1541
143 *	188 STO 16	233 Y`X
144 E`X	189 RCL 07	234 STO 17
145 STO 08	190 0.2989	235 RCL 06
146 GTO "H2O"	191 Y`X	236 0.3091
147♦LBL "FOG2"	192 STO 17	237 Y`X
148 RCL 00	193 RCL 06	238 0.0076
149 1/X	194 0.4981	239 *
150 0.2241563	195 Y`X	240 RCL 16
151 +	196 0.0850	241 *
152 1.0820891	197 *	242 RCL 17
153 *	198 RCL 16	243 *
154 CHS	199 *	244 CHS
155 RCL 05	200 RCL 17	245 E`X
156 *	201 *	246 STO 11
157 E`X	202 CHS	247♦LBL "CONT"
158 STO 08	203 E`X	248 260.0
159♦LBL "H2O"	204 STO 09	249 RCL 18
160 RCL 03	205♦LBL "CO2"	250 X<=Y?
161 0.01	206 RCL 05	251 GTO A
162 *	207 0.6178	252 296.0
163 RCL 07	208 Y`X	253 X<=Y?
164 *	209 STO 16	254 GTO B
165 STO 15	210 RCL 07	255 296.0
166 RCL 07	211 0.8488	256 RCL 18
167 X`2	212 Y`X	257 -
168 -2.43882	213 STO 17	258 36.0
169 *	214 RCL 06	259 /
170 STO 16	215 1.0792	260 STO 16
171 -14.9595	216 Y`X	261 GTO C
172 RCL 07	217 0.01164	262♦LBL A
173 *	218 *	263 1.000
174 RCL 16	219 RCL 16	264 STO 16
175 +	220 *	265 GTO C
176 18.9766	221 RCL 17	266♦LBL B
177 +	222 *	267 0.000
178 E`X	223 CHS	268 STO 16
179 RCL 15	224 E`X	269 GTO C
180 *	225 STO 10	270♦LBL C

C-2.a. PTRAN - Interactive Version (contd.)

271 0.5693	290 CHS	309 STO 13
272 RCL 16	291 E ⁻ X	310♦LBL E
273 *	292 STO 12	311 RCL 08
274 0.5437	293 0.0	312 RCL 09
275 +	294 RCL 04	313 *
276 RCL 15	295 X=Y?	314 RCL 10
277 *	296 GTO D	315 *
278 STO 16	297 RCL 04	316 RCL 11
279 RCL 07	298 0.63	317 *
280 RCL 06	299 Y ⁻ X	318 RCL 12
281 *	300 RCL 05	319 *
282 RCL 16	301 *	320 RCL 13
283 +	302 -0.3647	321 *
284 RCL 15	303 *	322 STO 14
285 *	304 E ⁻ X	323 CLA
286 RCL 05	305 STO 13	324 "TOTAL ="
287 *	306 GTO E	325 ARCL X
288 1.655 E-03	307♦LBL D	326 AVIEW
289 *	308 1.000	327 .END.

C-2.b. CTRAN - Batch Version

001 LBL "CTRAN"	046 1/X	091 *
002 RCL 01	047 -0.0073296	092 E`X
003 1013.25	048 +	093 STO 08
004 /	049 RCL 15	094 GTO "H2O"
005 STO 06	050 *	095 LBL "URBAN"
006 273.15	051 CHS	096 RCL 14
007 RCL 02	052 RCL 05	097 -0.0108702
008 +	053 *	098 Y`X
009 STO 18	054 E`X	099 0.3777368
010 273.15	055 STO 08	100 *
011 RCL 18	056 GTO "H2O"	101 STO 15
012 /	057 LBL "RURAL"	102 RCL 00
013 STO 07	058 RCL 14	103 1/X
014 RCL 03	059 -0.0698031	104 -0.0055683
015 -100.00	060 Y`X	105 +
016 /	061 0.3250795	106 RCL 15
017 1.0000	062 *	107 *
018 +	063 STO 15	108 CHS
019 STO 14	064 RCL 00	109 RCL 05
020 FS? 00	065 1/X	110 *
021 GTO "NONE"	066 -0.0062501	111 E`X
022 FS? 01	067 +	112 STO 08
023 GTO "OCEAN"	068 RCL 15	113 GTO "H2O"
024 FS? 02	069 *	114 LBL "FOG1"
025 GTO "RURAL"	070 CHS	115 RCL 00
026 FS? 03	071 RCL 05	116 1/X
027 GTO "TROPO"	072 *	117 0.0051780
028 FS? 04	073 E`X	118 +
029 GTO "URBAN"	074 STO 08	119 4.4754801
030 FS? 05	075 GTO "H2O"	120 *
031 GTO "FOG1"	076 LBL "TROPO"	121 CHS
032 FS? 06	077 RCL 14	122 RCL 05
033 GTO "FOG2"	078 -0.0186073	123 *
034 LBL "NONE"	079 Y`X	124 E`X
035 1.0000	080 0.0842857	125 STO 08
036 STO 08	081 *	126 GTO "H2O"
037 GTO "H2O"	082 STO 15	127 LBL "FOG2"
038 LBL "OCEAN"	083 RCL 00	128 RCL 00
039 RCL 14	084 1/X	129 1/X
040 -0.3469229	085 -0.0055992	130 0.2241563
041 Y`X	086 +	131 +
042 0.4072999	087 RCL 15	132 1.0820891
043 *	088 *	133 *
044 STO 15	089 CHS	134 CHS
045 RCL 00	090 RCL 05	135 RCL 05

C-2.b. CTRAN - Batch Version (contd.)

136 *	181 *	226 STO 11
137 E`X	182 CHS	227 LBL "CONT"
138 STO 08	183 E`X	228 260.0
139 LBL "H2O"	184 STO 09	229 RCL 18
140 RCL 03	185 LBL "CO2"	230 X<=Y?
141 0.01	186 RCL 05	231 GTO A
142 *	187 0.6178	232 296.0
143 RCL 07	188 Y`X	233 X<=Y?
144 *	189 STO 16	234 GTO B
145 STO 15	190 RCL 07	235 296.0
146 RCL 07	191 0.8488	236 RCL 18
147 X`2	192 Y`X	237 -
148 -2.43882	193 STO 17	238 36.0
149 *	194 RCL 06	239 /
150 STO 16	195 1.0792	240 STO 16
151 -14.9595	196 Y`X	241 GTO C
152 RCL 07	197 0.01164	242 LBL A
153 *	198 *	243 1.000
154 RCL 16	199 RCL 16	244 STO 16
155 +	200 *	245 GTO C
156 18.9766	201 RCL 17	246 LBL B
157 +	202 *	247 0.000
158 E`X	203 CHS	248 STO 16
159 RCL 15	204 E`X	249 GTO C
160 *	205 STO 10	250 LBL C
161 STO 15	206 LBL "O3"	251 0.5693
162 RCL 05	207 RCL 05	252 RCL 16
163 *	208 0.7498	253 *
164 0.100	209 Y`X	254 0.5437
165 *	210 STO 16	255 +
166 0.5582	211 RCL 07	256 RCL 15
167 Y`X	212 0.1541	257 *
168 STO 16	213 Y`X	258 STO 16
169 RCL 07	214 STO 17	259 RCL 07
170 0.2989	215 RCL 06	260 RCL 06
171 Y`X	216 0.3091	261 *
172 STO 17	217 Y`X	262 RCL 16
173 RCL 06	218 0.0076	263 +
174 0.4981	219 *	264 RCL 15
175 Y`X	220 RCL 16	265 *
176 0.0850	221 *	266 RCL 05
177 *	222 RCL 17	267 *
178 RCL 16	223 *	268 1.655 E-03
179 *	224 CHS	269 *
180 RCL 17	225 E`X	270 CHS

C-2.b. CTRAN - Batch Version (contd.)

271 E`X	284 E`X	297 *
272 STO 12	285 STO 13	298 RCL 12
273 0.0	286 GTO E	299 *
274 RCL 04	287 LBL D	300 RCL 13
275 X=Y?	288 1.000	301 *
276 GTO D	289 STO 13	302 STO 14
277 RCL 04	290 LBL E	303 CLA
278 0.63	291 RCL 08	304 "TOTAL ="
279 Y`X	292 RCL 09	305 ARCL X
280 RCL 05	293 *	306 AVIEW
281 *	294 RCL 10	307 .END.
282 -0.3647	295 *	
283 *	296 RCL 11	

END
DATED
FILM
8-88
Dtric