Effective Date: November 06, 2013
Revision C

GSFC JPSS CMO
November 8, 2013
Released

**Joint Polar Satellite System (JPSS) Ground Project**
**Code 474**
**474-00074**

# Joint Polar Satellite System (JPSS) Operational Algorithm Description (OAD) Document for VIIRS Cloud Optical Properties (COP) Intermediate Product (IP) Software

## For Public Release



Goddard Space Flight Center
Greenbelt, Maryland

National Aeronautics and
Space Administration

# Joint Polar Satellite System (JPSS) Operational Algorithm Description (OAD) Document for VIIRS Cloud Optical Properties (COP) Intermediate Product (IP) Software
# JPSS Electronic Signature Page

**Prepared By:**

Bonnie Reed
JPSS Data Products and Algorithms EDR Lead
(Electronic Approvals available online at (https://jpssmis.gsfc.nasa.gov/mainmenu_dsp.cfm)

**Approved By:**

Eric Gottshall
DPA Manager
(Electronic Approvals available online at (https://jpssmis.gsfc.nasa.gov/mainmenu_dsp.cfm)

**Goddard Space Flight Center**
**Greenbelt, Maryland**

# Preface

This document is under JPSS Ground Algorithm ERB configuration control. Once this document is approved, JPSS approved changes are handled in accordance with Class I and Class II change control requirements as described in the JPSS Configuration Management Procedures, and changes to this document shall be made by complete revision.

Any questions should be addressed to:

JPSS Configuration Management Office
NASA/GSFC
Code 474
Greenbelt, MD 20771

# Change History Log

| Revision | Effective Date | **Description of Changes** (Reference the CCR & CCB/ERB Approve Date) |
|---|---|---|
| Original | 06/03/2011 | This version incorporates **474-CCR-11-0090** which converts D39298, Operational Algorithm Description (OAD) Document for VIIRS Cloud Optical Properties (COP) Software, Rev A, dated 10/29/2008 to a JPSS document, Rev -. This was approved by the JPSS Ground Algorithm ERB on June 3, 2011. |
| Revision A | 01/18/2012 | **474-CCR-11-0273:** This version baselines 474-00074, Joint Polar Satellite System (JPSS) Operational Algorithm Description (OAD) Document for VIIRS Cloud Optical Properties (COP) Intermediate Product (IP) Software, for the Mx 6 IDPS release. This CCR was approved by the JPSS Algorithm ERB on January 18, 2012. |
| Revision B | 05/14/2013 | **474-CCR-13-0948:** This version authorizes 474-00074, JPSS OAD Document for VIIRS COP IP Software, for the Mx 7.0 IDPS release. Includes **ECR-ALG-0037** which contains Raytheon PCR029954; OAD: PRO: SW and DOC Investigation: COP OAD and CODE (Tables 1 & 2). Includes Raytheon PCR032720; 474-CCR-13-0916/ECR-ALG-0037: Update applicable OAD filenames/template/Rev/etc. for Mx7 Release. |
| Revision C | 11/06/2013 | **474-CCR-13-1288:** This version authorizes 474-00074, JPSS OAD Document for VIIRS COP IP Software, for the Mx 8.0 IDPS release. Includes administrative changes authorized by interoffice memo and Raytheon PCR033833; OAD: Update COP IP for NOGAPS to NAVGEM (FNMOC) change, in Table 37. |
| | | |
| | | |

# NATIONAL POLAR-ORBITING OPERATIONAL ENVIRONMENTAL SATELLITE SYSTEM (NPOESS)

# OPERATIONAL ALGORITHM DESCRIPTION DOCUMENT FOR VIIRS CLOUD OPTICAL PROPERTIES (COP) SOFTWARE

## SDRL No. S141
## SYSTEM SPECIFICATION SS22-0096

**RAYTHEON COMPANY**
**INTELLIGENCE AND INFORMATION SYSTEMS (IIS)**
**NPOESS PROGRAM**
**Omaha, NEBRASKA**

IAW DFAR 252.227-7036, Raytheon hereby declares that, to the best of its knowledge and belief, the technical data delivered under Subcontract No. 7600002744 is complete, accurate, and complies with all requirements of the Subcontract.

TITLE: NATIONAL POLAR-ORBITING OPERATIONAL ENVIRONMENTAL SATELLITE SYSTEM (NPOESS) OPERATIONAL ALGORITHM DESCRIPTION DOCUMENT FOR VIIRS CLOUD OPTICAL PROPERTIES (COP) SOFTWARE

APPROVAL SIGNATURES:

________________________________________

Stephen E. Ellefson Date
ING/PRO Lead

________________________________________

Gabriela A. Ostler Date
Mission Assurance and Enterprise Effectiveness (MAEE)

Northrop Grumman Space & Mission Systems Corp.
**Space Technology**
One Space Park
Redondo Beach, CA 90278

**Engineering & Manufacturing Development (EMD) Phase**
**Acquisition & Operations Contract**

**CAGE NO. 11982**

# Operational Algorithm Description Document for the VIIRS Cloud Optical Properties (COP) Software

**Document Date: Nov 11, 2011**

**Document Number: D39298**
**Revision: B11**

**PREPARED BY:**

| | |
|---|---|
| Eric Wong *Date*<br>*AM&S COP Lead* | Paul D. Siebels *Date*<br>*IDPS PRO SW Manager* |

**ELECTRONIC APPROVAL SIGNATURES:**

| | |
|---|---|
| Roy Tsugawa *Date*<br>*A&DP Lead & ACCB Chair* | Stephen E. Ellefson *Date*<br>*IDPS Processing SI Lead* |
| Bob Hughes *Date*<br>*A&DP Deputy & ARB Chair* | |

Prepared by
**Northrop Grumman Space Technology**
One Space Park
Redondo Beach, CA 90278

Prepared for
**Department of the Air Force**
NPOESS Integrated Program Office
C/O SMC/CIK
2420 Vela Way, Suite 1467-A8
Los Angeles AFB, CA 90245-4659

Under
**Contract No. F04701-02-C-0502**

This document has been identified per the NPOESS Common Data Format Control Book – External Volume 5 Metadata, D34862-05, Appendix B as a document to be provided to the NOAA Comprehensive Large Array-data Stewardship System (CLASS) via the delivery of NPOESS Document Release Packages to CLASS.

| | |
|---|---|
| Northrop Grumman Space & Mission Systems Corp.<br>**Space Technology**<br>One Space Park<br>Redondo Beach, CA 90278 | NORTHROP GRUMMAN<br>Raytheon |

## Revision/Change Record

**Document Number D39298**

| Revision | Document Date | Revision/Change Description | Pages Affected |
|---|---|---|---|
| --- | 8-21-04 | Initial Release. | All |
| A1 | 10-30-06 | 7Jun05Discussion of quality flags per SPCR ALG 629.<br>22Sep05 - Reflects Raytheon-Omaha's initial edits for Science To Operational (Sci2Ops) Code Conversion Process, adding Raytheon coversheet, title/signature page, etc.<br>17Mar06 – Added new Unit Test prior to CCPR scheduled for 27Mar06.<br>27Mar06 – Made edits to correct copyright year on coversheet, correct numbering and List of Tables for Glossary and Acronym tables, renamed Tables 10 thru 22 to include the Function or Routine name, remove all extra blank rows from bottom of tables, add OAD to acronym list, replaced section 1.3.3 Source Code and Test Data References with a 4 column table containing more information.<br>17Apr06 – Within Section 1.3.3 Source Code and Test Data References inserted information relating to NGST original dropped COP documentation for the Unit Test and science source code. Added Raytheon-Omaha electronic signatures in preparation for official delivery to the prime.<br>18Apr06 – Updated the vobs/paths within Section 1.3.3 Source Code and Test Data References table, far right column. Changed all 2-digit section headers so they are left-aligned and not indented .13 inch. Did spacing/format edits plus changed all text so it's left aligned to the margin and not both left and right.<br>30Oct06 – changed per DDPR for COP FO changes for 20060424_ISTN_VIIRS_NGST_4.2 drop. Added M14 and M16 for inputs. Code changes per SPCR ALG00001054. | All |
| A2 | 5-16-07 | 01Dec06 – 2.2.1 Inputs, Bright to Brightness and ster to sr to match the SI units, microns to micrometers, heights described to geopotential heights, Updated TOC, Inserted Table 9, CM Struct input, corrected revision editions.<br>04Apr07 – Modified Figure 4 & Figure 5 according to the current version of IDPS code, added new routine input tables for ComputeClrRadNight_Extra and ComputeClrRadDay_Extra, updated the table numbers, updated source code to reflect on 1.5 build of IDPS COP code instead of 1.4.<br>16May07 – Included Track Changes, and made changes to the document according to NP-EMD.2006.510.0019. | All |
| A3 | 5-21-07 | Delivered to NGST. | All |

Northrop Grumman Space & Mission Systems Corp.
**Space Technology**
One Space Park
Redondo Beach, CA 90278

## Revision/Change Record

**Document Number** **D39298**

| Revision | Document Date | Revision/Change Description | Pages Affected |
|---|---|---|---|
| A4 | 6-16-08 | Accepted tracked changes for fresh baseline version. | All |
| A5 | 6-19-08 | Reformatted to new OAD template. Updated in response to NGST comments from last delivery. TMs NP-EMD.2005.510.0078, NP-EMD.2006.510.0058, NP-EMD.2007.510.0030 have been incorporated into COP. Delivered to NGST. | All |
| A6 | 7-16-08 | Implemented TM NP-EMD.2008.510.0007. Modified cloud optical properties algorithm to be unit consistent with SDR, read the new cloud mask and renamed ConvertSdrUnits to CreateTempSdrUnits | 1.3.2, 2.1.2 |
| A7 | 9-4-08 | Updated Graceful Degradation. | 17,18 |
| A8 | 10-1-08 | Prepared for TIM/ACCB. | All |
| A | 10-29-08 | ECR A-093. Incorporated interim changes and addressed TIM comments. | All |
| B1 | 5-27-09 | Implemented TM NP-EMD.2008.510.0067, Rev A. Update Cloud Optical Properties with System Spec Rev N and fix software bugs | Sec. 2.1.1.2 |
| B2 | 8-13-09 | Updated due to PCR019987 | Table 11 |
| B3 | 11-04-09 | Updated for SDRL | All |
| B4 | 01-07-10 | Fixed Table 10 based on SDRL review comments from A&DP | 7 |
| B5 | 06-21-10 | Implemented TM-EMD.2010.510.0027, Upgrade of Cloud Optical Properties Algorithm in clear sky radiance modeling and NP-EMD.2010.510.0047, Fix software bugs in Cloud Optical Properties Code | All |
| B6 | 07-08-10 | Prepared for SDRL | All |
| B7 | 08-27-10 | Updated product ranges | Table 12 |
| B8 | 09-14-10 | Changed RetrieveSfcAlbedoEmiss and CreateTempSdrUnits method signatures. | Tables 17 & 19 |
| B9 | 09-14-10 | ECR1061/PCR024068 Updated COT ranges | Table 12 |
| B10 | 10-12-10 | Updated due to document convergence to include TM 2010.510.0011, 2010.510.0012, 2010.510.0013, & 2010.510.0014, | All |
| B11 | 11-11-11 | Updated for PCRs PCR027775, PCR027776, PCR028270 & PCR028553 | All |

## Table of Contents

1.0 INTRODUCTION .......... 1
1.1 Objective .......... 1
1.2 Scope .......... 1
1.3 References .......... 1
1.3.1 Document References .......... 1
1.3.2 Source Code References .......... 3
2.0 ALGORITHM OVERVIEW .......... 5
2.1 VIIRS Cloud Optical Properties (COP) Description .......... 5
2.1.1 Interfaces .......... 6
2.1.1.1 Inputs .......... 6
2.1.2.1.1 Requirements for Input .......... 8
2.1.2.2 Outputs .......... 8
2.1.2.2.1 Requirements for Output .......... 10
2.1.3 Algorithm Processing .......... 11
2.1.3.1 Main Module 1 - getAveClrSkyRad( ) .......... 13
2.1.3.2 DetermineCopProcessPath( ) .......... 14
2.1.3.3 CreateTempSdrUnits( ) .......... 15
2.1.3.4 RetrieveSfcType( ) .......... 15
2.1.3.5 RetrieveSfcAlbedoEmiss( ) .......... 15
2.1.3.6 ComputeCLR() .......... 15
2.1.3.7 Calculate_CLR() .......... 16
2.1.3.8 pfaast() .......... 16
2.1.3.9 tran_modis042() .......... 17
2.1.3.10 modrad() .......... 17
2.1.3.11 Main Module 2 - DetermineCotEps( ) .......... 18
2.1.3.11.1 Usage .......... 19
2.1.3.12 SdrData_Rescale() .......... 19
2.1.3.13 DetermineCotEpsCttWater( ) .......... 19
2.1.3.14 RetrieveDayWaterCotEps( ) .......... 20
2.1.3.15 ComputeNightWaterCotEpsCtt( ) .......... 20
2.1.3.16 extract_water_lut_values() .......... 21
2.1.3.17 ChooseDayWaterCotEps() .......... 21
2.1.3.18 DetermineCotEpsCttIce( ) .......... 21
2.1.3.19 RetrieveDayIceCotEps( ) .......... 22

2.1.3.20 extract_water_lut_values() ... 22
2.1.3.21 ChooseDayIceCotEps() ... 23
2.1.3.22 MergeWaterIceCotEpsCtt( ) ... 23
2.1.4 Graceful Degradation ... 23
2.1.4.1 Graceful Degradation Inputs ... 23
2.1.4.2 Graceful Degradation Processing ... 24
2.1.4.3 Graceful Degradation Outputs ... 24
2.1.5 Exception Handling ... 24
2.1.6 Data Quality Monitoring ... 24
2.1.7 Computational Precision Requirements ... 24
2.1.8 Algorithm Support Considerations ... 25
2.1.9 Assumptions and Limitations ... 25
2.1.9.1 Assumptions ... 25
2.1.9.2 Limitations ... 25
3.0 GLOSSARY/ACRONYM LIST ... 26
3.1 Glossary ... 26
3.2 Acronyms ... 29
4.0 OPEN ISSUES ... 30

## List of Figures

Figure 1. A High-Level COP Processing Chain ........ 5
Figure 2. Flow Chart of COP/EPS with Other EDRs ........ 11
Figure 3. Two-band Reflectances for Cirrus Clouds ........ 12
Figure 4. Flow of getAveClrSkyRad Algorithm ........ 14
Figure 5. Flow Chart of DetermineCotEps Algorithm ........ 19

## List of Tables

Table 1. Reference Documents ........ 1
Table 2. Source Code References ........ 3
Table 3. VIIRS SDR Data Items ........ 6
Table 4. Ancillary Data Items ........ 6
Table 5. VIIRS Ice Cloud LUT Data Structure ........ 6
Table 6. VIIRS Water Cloud LUT Data Structure ........ 6
Table 7. VIIRS COP Surface Type LUT Data Structure ........ 7
Table 8. VIIRS COP Transmittance LUT Data Structure ........ 7
Table 9. VIIRS COP IR Band Spectural LUT Data Structure ........ 7
Table 10. VIIRS COP PFAAST Regression LUT Data Structure ........ 7
Table 11. VIIRS CM Data Structure ........ 8
Table 12. COP Output Data Specifications ........ 8
Table 13. COP Quality Flag Specifications ........ 9
Table 14. CTT Quality Flag Specifications ........ 9
Table 15. Input Argument Variables Function getAveClrSkyRad( ) ........ 13
Table 16. Input Argument Variables DetermineCopProcessPath( ) ........ 14
Table 17. Input Argument Variables CreateTempSdrUnits( ) ........ 15
Table 18. Input Argument Variables RetrieveSfcType( ) ........ 15
Table 19. Input Argument Variables RetrieveSfcAlbedoEmiss( ) ........ 15
Table 20. Input Argument Variables Function ComputeCLR( ) ........ 16
Table 21. Input Argument Variables Function Calculate_CLR( ) ........ 16
Table 22. Input Argument Variables Function pfaast( ) ........ 17
Table 23. Input Argument Variables Function tran_modis042( ) ........ 17
Table 24. Input Argument Variables Function modrad( ) ........ 17
Table 25. Input Argument Variables Function DetermineCotEps( ) ........ 18
Table 26. Input Argument Variables SdrData_Rescale( ) ........ 19
Table 27. Input Argument Variables DetermineCotEpsCttWater( ) ........ 20

Table 28. Input Argument Variables RetrieveDayWaterCotEps( ) .......... 20
Table 29. Input Argument Variables ComputeNightWaterCotEpsCtt( ) .......... 20
Table 30. Input Argument Variables extract_water_lut_values ( ) .......... 21
Table 31. Input Argument Variables ChooseDayWaterCotEps ( ) .......... 21
Table 32. Input Argument Variables DetermineCotEpsCttIce( ) .......... 22
Table 33. Input Argument Variables RetrieveDayIceCotEps( ) .......... 22
Table 34. Input Argument Variables extract_water_lut_values ( ) .......... 22
Table 35. Input Argument Variables ChooseDayIceCotEps ( ) .......... 23
Table 36. Input Argument Variables MergeWaterIceCotEpsCtt( ) .......... 23
Table 37. Graceful Degradation .......... 24
Table 38. Glossary .......... 26
Table 39. Acronyms .......... 29
Table 40. TBXs .......... 30

# 1.0 INTRODUCTION

## 1.1 Objective

The purpose of the Operational Algorithm Description (OAD) document is to express, in computer-science terms, the remote sensing algorithms that produce the National Polar-Orbiting Operational Environmental Satellite System (NPOESS) end-user data products. These products are individually known as Raw Data Records (RDRs), Temperature Data Records (TDRs), Sensor Data Records (SDRs) and Environmental Data Records (EDRs). In addition, any Intermediate Products (IPs) produced in the process are also described in the OAD.

The science basis of an algorithm is described in a corresponding Algorithm Theoretical Basis Document (ATBD). The OAD provides a software description of that science as implemented in the operational ground system -- the Data Processing Element (DPE).

The purpose of an OAD is two-fold:

1. Provide initial implementation design guidance to the operational software developer.
2. Capture the “as-built” operational implementation of the algorithm reflecting any changes needed to meet operational performance/design requirements.

An individual OAD document describes one or more algorithms used in the production of one or more data products. There is a general, but not strict, one-to-one correspondence between OAD and ATBD documents.

This particular document describes operational software implementation for the Visible/Infrared Imager/Radiometer Suite (VIIRS) Cloud Optical Properties (COP).

## 1.2 Scope

The scope of this document is limited to the description of the core operational algorithm(s) required to create the VIIRS Cloud Optical Properties (COP) software. The theoretical basis for this algorithm is described in Section 3.3 of the Cloud Effective Particle Size and Cloud Optical Thickness Algorithm Theoretical Basis Document ATBD, 474-00042.

## 1.3 References

The primary software detailed design documents listed here include science software documents, NPOESS program documents, plus source code and test data references.

### 1.3.1 Document References

The science and system engineering documents relevant to the algorithms described in this OAD are listed in Table 1.

**Table 1. Reference Documents**

| Document Title | Document Number/Revision | Revision Date |
|---|---|---|
| Cloud Effective Particle Size and Cloud Optical Thickness Algorithm Theoretical Basis Document ATBD | 474-00042 | Latest |
| VIIRS Cloud Mask (VCM) Algorithm Theoretical Basis Document ATBD | 474-00033 | Latest |

| Document Title | Document Number/Revision | Revision Date |
|---|---|---|
| JPSS Environmental Data Record (EDR) Production Report (PR) for NPP | 474-00012 | Latest |
| JPSS Environmental Data Record (EDR) Interdependency Report (IR) for NPP | 474-00007 | Latest |
| NPP Mission Data Format Control Book and App A (MDFCB) | 429-05-02-42_MDFCB | Latest |
| JPSS Common Data Format Control Book - External - – Block 1.2.2 (All Volumes) | 474-00001-01-B0122 CDFCB-X Vol I<br>474-00001-02-B0122 CDFCB-X Vol II<br>474-00001-03-B0122 CDFCB-X Vol III<br>474-00001-04-01-B0122 CDFCB-X Vol IV Part 1<br>474-00001-04-02-B0122 CDFCB-X Vol IV Part 2<br>474-00001-04-03-B0122 CDFCB-X Vol IV Part 3<br>474-00001-04-04-B0122 CDFCB-X Vol IV Part 4<br>474-00001-05-B0122 CDFCB-X Vol V<br>474-00001-06-B0122 CDFCB-X Vol VI<br>474-00001-08-B0122 CDFCB-X Vol VIII | Latest |
| JPSS Common Data Format Control Book - External - Block 1.2.3 (All Volumes) | 474-00001-01-B0123 CDFCB-X Vol I<br>474-00001-02-B0123 CDFCB-X Vol II<br>474-00001-03-B0123 CDFCB-X Vol III<br>474-00001-04-01-B0123 CDFCB-X Vol IV Part 1<br>474-00001-04-02-B0123 CDFCB-X Vol IV Part 2<br>474-00001-04-03-B0123 CDFCB-X Vol IV Part 3<br>474-00001-04-04-B0123 CDFCB-X Vol IV Part 4<br>474-00001-05-B0123 CDFCB-X Vol V<br>474-00001-06-B0123 CDFCB-X Vol VI<br>474-00001-08-B0123 CDFCB-X Vol VIII | Latest |
| NPP Command and Telemetry (C&T) Handbook | D568423 Rev. C | 30 Sep 2008 |
| VIIRS Cloud Optical Properties Unit Level Detailed Design | Y0010870 Ver. 5 Rev. 9 | Jul 2004 |
| VIIRS Cloud Module Level Interface Control Document | Y3278 Ver. 5 Rev. 9 | Oct 2004 |
| VIIRS Cloud Module-level Software Architecture | Y2472 Ver. 5 Rev. 12 | Jan 2005 |
| VIIRS Cloud Module Data Dictionary | Y0010871 Ver. 5 Rev. 11 | Jan 2005 |
| JPSS CGS Data Processor Inter-subsystem Interface Control Document (DPIS ICD) Vol I – IV | IC60917-IDP-002 | Latest |
| Joint Polar Satellite System (JPSS) Program Lexicon | 470-00041 | Latest |
| NGST/SE technical memo – MS Engineering Memo_COP OAD Update | NP-EMD.2005.510.0078 Rev. --- | 07 Jul 2005 |

| Document Title | Document Number/Revision | Revision Date |
|---|---|---|
| NGST/SE technical memo – NPP_VIIRS_COP_OAD_Blockiness Correction Updates | NP-EMD.2006.510.0019 Rev. --- | 23 Mar 2006 |
| NGST/SE technical memo – NPP_COP_CODE_FIX_FILL_VALUE_STRIP | NP-EMD.2006.510.0058 | 07 Aug 2006 |
| NGST/SE technical memo – NPP_VIIRS_COP_array_out_of_bound | NP-EMD.2007.510.0030 | 25 Apr 2007 |
| NGST/SE technical memo – NPP_VIIRS_COP_Unit_Consistent | NP-EMD.2008.510.0007 Rev. --- | 29 Jan 2008 |
| NGAS/AM&S technical memo – Update Cloud Optical Properties Code to be consistent with System Spec Rev N and fix software bugs | NP-EMD.2008.510.0067 Rev. A | 04 Dec 2008 |
| NGST/SE technical memo – NPP_VIIRS_COP_Skip_FILLs_in Profiles | NP-EMD.2009.510.0073 | 03 Dec 2009 |
| NGST/SE technical memo – Upgrade of Cloud Optical Properties Algorithm in clear sky radiance modeling | NP-EMD.2010.510.0027 | 24 Mar 2010 |
| NGST/SE technical memo – Fix software bugs in Cloud Optical Properties Code | NP-EMD.2010.510.0047 | 19 May 2010 |
| NGST/SE technical memos:<br>LUT_OAD_Drop History_Corrections<br>LUT_Format_Corrections<br>PC_OAD_Last_Drop_Corrections<br>PC_Format_Corrections | <br>NPOESS GJM-2010.510.0011<br>NPOESS GJM-2010.510.0012<br>NPOESS GJM-2010.510.0013<br>NPOESS GJM-2010.510.0014 | <br>21 Sep 2010<br>21 Sep 2010<br>22 Sep 2010<br>22 Sep 2010 |
| NGST/SE technical memo – Description of OAD and CDFCB updates resulting from Ice cloud reflectance LUT dimension change | NPP.2011.220.0006 | 24 Aug 2011 |
| NGST/SE technical memo – Modify COP for changes needed by the NPP cloud ATDR 20 for CEPS | NPP.2011.220.0001 (also titled as ATDR20) | 20 Jun 2011 |

### 1.3.2 Source Code References

The science and operational code and associated documentation relevant to the algorithms described in this OAD are listed in Table 2.

**Table 2. Source Code References**

| Reference Title | Reference Tag/Revision | Revision Date |
|---|---|---|
| VIIRS COP science-grade software (original reference source | VIIRS_Drop1_Documentation | 31 Mar 2003 |
| VIIRS Science Algorithms 2.7 Delivery to IDPS | ISTN_VIIRS_NGST_2.7 | 31 Aug 2004 |
| NGST/SE technical memo – NPP_VIIRS_COP_OAD_Blockiness Correction Updates | NP-EMD.2006.510.0019 Rev. ---<br>(OAD Rev A2) | 23 Mar 2006 |
| VIIRS COP science-grade software | ISTN_VIIRS_NGST_4.2 | 12 Apr 2006 |
| NGST/SE technical memo – NPP_COP_CODE_FIX_FILL_VALUE_STRIP | NP-EMD.2006.510.0058 Rev. --- | 07 Aug 2006 |
| VIIRS "Drop 3.5.1" science algorithm packages delivered to the IDPS IPT in support of IDPS Science to Ops conversion. TEST DATA ONLY – no source code | ISTN_VIIRS_NGST_3.5.1_DATA | 21 Dec 2006 |
| VIIRS Science Algorithms 2.7 Delivery to IDPS | ISTN_VIIRS_NGST_3.5.3_DATA (ECRs | 02 Apr 2008 |

| Reference Title | Reference Tag/Revision | Revision Date |
|---|---|---|
| | A140 & A141) | |
| VIIRS COP operational software | B1.5.x.1 (OAD Rev A5) | Jul 2008 |
| NGST/SE technical memo – NPP_VIIRS_COP_array_out_of_bound | NP-EMD.2007.510.0030 Rev. --- | 25 Apr 2007 |
| NGST/SE technical memo – NPP_VIIRS_COP_Unit_Consistent | NP-EMD.2008.510.0007 Rev. --- | 29 Jan 2008 |
| VIIRS COP science-grade software, includes Update Cloud Optical Properties Code to be consistent with System Spec Rev N and fix software bugs, TM NP-EMD.2008.510. 0067, Rev A | ISTN_VIIRS_NGST_3.5.7 (ECR-A187) | 28 Jan 2009 |
| VIIRS COP operational-grade software | Build Post-X-I (OAD Rev B1) | 27 May 2009 |
| PCR019987 | Sensor Characterization Build (SC-2)(OAD Rev B2) | 13 Aug 2009 |
| SDRL | (OAD Rev B3) | 04 Nov 2009 |
| VIIRS COP science-grade software, includes Upgrade of Cloud Optical Properties Algorithm in clear sky radiance modeling, TM NP-EMD.2010.510.0027 | ISTN_VIIRS_NGST_3.5.8 (ECR-A284) | 21 Apr 2010 |
| VIIRS COP operational-grade software, includes PCRs 23540, 23764, 22836, 23437, 23440, 23439, 23969, 23841; and TMs NP-EMD.2010.510.0047 & NP-EMD.2009.510.0073 | Sensor Characterization Build (SC-11) (OAD Rev B5) | 21 Jun 2010 |
| SDRL | (OAD Rev B6) | 08 Jul 2010 |
| ECR1061/PCR024068 update output field ranges | (OAD Rev B7) | 17 Aug 2010 |
| PCR024479 update cloud edr Output Scaling Limits | (OAD Rev B9) | 14 Sep 2010 |
| Convergence Update (No code updates) | (OAD Rev B10) | 12 Oct 2010 |
| VIIRS COP science-grade software, includes TM \ NP-EMD.2010.510.0105 | ISTN_VIIRS_NGST_3.5.9 Not implemented, superseded by 4.26 | 05 Jan 2011 |
| VIIRS COP science-grade software, includes TM 2011.220.0001 | ISTN_VIIRS_NGST_4.26 (ECR-ALG-0026) | 29 Jun 2011 |
| VIIRS COP science-grade software, includes TM NPP.2011.220.0006 | ISTN_VIIRS_NGST_4.26.1 (ECR-ALG-0029) | 24 Aug 2011 |
| VIIRS COP operational-grade software, includes PCRs PCR027775 (code-includes TM 2011.220.0001), PCR028269 (Code), PCR027776 (OAD) & PCR028270 (OAD) | Build Maintenance Release 1.5.6-K (OAD Rev B11) | 11 Nov 2011 |
| OAD transitioned to JPSS Program – this table is no longer updated. | | |

## 2.0 ALGORITHM OVERVIEW

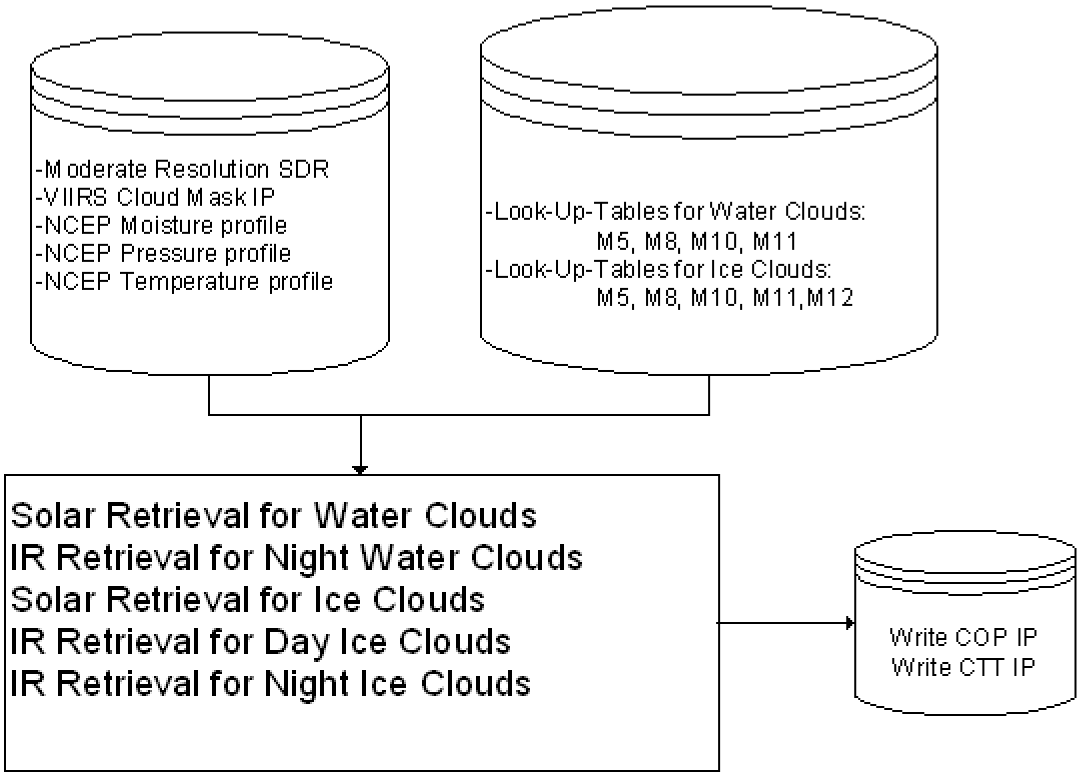


**Figure 1. A High-Level COP Processing Chain**

A high-level flow diagram of processing to determine the Cloud Optical Thickness (COT) and Effective Particle Size (EPS) parameters is provided in Figure 1. Input parameters required by these algorithms include VIIRS Sensor Data Record (SDR), VIIRS Cloud Mask (VCM) and ancillary input data from NCEP. The overall processing begins with the detection of cloud-contaminated pixels and the determination of their associated phase via the VCM program. For each cloud phase, the retrieval algorithms contain two basic approaches: solar and infrared (IR). The solar approach uses the reflectances of the 1.61μm channel and, either 0.672μm (non-snow/ice surface) or 1.24μm (snow/ice surface) channels for daytime retrieval of COT and EPS. It determines these cloud parameters by matching measured reflectances with those from the comprehensive radiance look-up tables (LUTs), which are pre-computed reflectances from the UCLA LBLE Radiative Transfer Model (RTM) for a wide range of scenarios. The IR approach utilizes various combinations of radiances of the VIIRS 3.7, 8.55, 10.7625 and 12.013 μm channels to infer cloud temperature and IR emissivity, from which the COT and EPS (except for daytime ice clouds in which the solar retrieved COT and EPS are used) are determined on the basis of the theory of radiative transfer and parameterizations.

### 2.1 VIIRS Cloud Optical Properties (COP) Description

The COP retrieval algorithm and the theoretical basis are described in detail in the Cloud Effective Particle Size and Cloud Optical Thickness Algorithm Theoretical Basis Document ATBD, 474-00042. The ATBD had mentioned the use of VIIRS surface albedo and surface type Intermediate Products (IPs). Instead, the code has adopted the use of surface type specified in VCM. Surface albedo and emissivity is then determined from the Surface Type LUT.

### 2.1.1 Interfaces

The code reads in the required input files including VIIRS moderate resolution SDR, VCM, ancillary data, and the LUTs for water and ice clouds. Furthermore, data items from ING also include the specification of configurable threshold values.

#### 2.1.1.1 Inputs

The details of the required input parameters for the retrieval of COT and EPS come from VIIRS and non-VIIRS sources and are summarized in Tables 3 through 9. The VIIRS data either come directly from the VIIRS processing stream or are generated from the RTM.

**Table 3. VIIRS SDR Data Items**

| Input Data | Description/Source | Units |
|---|---|---|
| Radiance for bands M12, M14, M15 and M16 | Radiance/VIIRS MOD SDR | W/(m $^2$ .sr.µm) |
| Reflectance for band M5, M8, M10, and M11 | Reflectance/VIIRS MOD SDR | Unitless |
| Brightness Temperature for bands M12, M14, M15, M16 | Brightness temperature/ VIIRS MOD SDR | Deg. Kelvin |
| Sen Zen Angle | Sensor zenith angle/ VIIRS MOD Geolocation | Radians |
| Sen Azi Angle | Sensor azimuth angle/ VIIRS MOD Geolocation | Radians |
| Sol Zen Angle | 2.1.2 Solar zenith angle/ VIIRS MOD Geolocation | Radians |
| Sol Azi Angle | Solar Azimuth angle/ VIIRS MOD Geolocation | Radians |

**Table 4. Ancillary Data Items**

| Input Data | Description/Source | Units |
|---|---|---|
| Moisture | Water vapor mixing ratio profile | gram/kgram |
| Temperature | Temperature profile | Deg. Kelvin |
| Surface Skin Temperature | Surface Skin Temperature profiile | Deg. Kelvin |

**Table 5. VIIRS Ice Cloud LUT Data Structure**

| Input | Description/Source | Units |
|---|---|---|
| Sol_zen_bins | Solar Zenith index bins | Radians |
| Sen_zen_bins | Sensor Zenith index bins | Radians |
| Rel_az_bins | Relative Azimuth index bins | Radians |
| Sfc_albedo_bins | Surface Albedo index bins | None |
| Sfc_emiss_bins | Surface Emissivity index bins | None |
| Eps_index | Effective Particle Size indices | None |
| Eps_bins | Effective Particle Size index bins | Micrometers |
| Cot_bins | Cloud Optical Thickness index bins | Tau |
| Reflectance_M5 | Reflectance of VIIRS band M5 | None |
| Reflectance_M8 | Reflectance of VIIRS band M8 | None |
| Reflectance_M10 | Reflectance of VIIRS band M10 | None |
| Reflectance_M11 | Reflectance of VIIRS band M11 | None |

**Table 6. VIIRS Water Cloud LUT Data Structure**

| Input | Description/Source | Units |
|---|---|---|
| Sol_zen_bins | Solar Zenith index bins | Radians |

| Input | Description/Source | Units |
|---|---|---|
| Sen_zen_bins | Sensor Zenith index bins | Radians |
| Rel_az_bins | Relative Azimuth index bins | Radians |
| Sfc_albedo_bins | Surface Albedo index bins | None |
| Sfc_emiss_bins | Surface Emissivity index bins | None |
| Eps_index | Effective Particle Size indices | None |
| Eps_bins | Effective Particle Size index bins | Micrometers |
| Cot_bins | Cloud Optical Thickness index bins | Tau |
| Reflectance_M5 | Reflectance of VIIRS band M5 | None |
| Reflectance_M8 | Reflectance of VIIRS band M8 | None |
| Reflectance_M10 | Reflectance of VIIRS band M10 | None |
| Reflectance_M11 | Reflectance of VIIRS band M11 | None |

**Table 7. VIIRS COP Surface Type LUT Data Structure**

| Input | Description/Source | Units |
|---|---|---|
| Albedo | Surface Albedo | None |
| Emissivity | Surface Emissivity | None |

**Table 8. VIIRS COP Transmittance LUT Data Structure**

| Input | Description/Source | Units |
|---|---|---|
| Altitude | Altitude profile | Meters |
| Trans_ref | None | None |
| transdT_ref | None | None |
| Transdq_ref | None | None |
| T_ref | None | None |
| Du_ref | None | None |

**Table 9. VIIRS COP IR Band Spectural LUT Data Structure**

| Input | Description/Source | Units |
|---|---|---|
| csw_band | Central Wave Numbers | Inverse centimeters |
| tsc_band | Temperature Correction Slaps | None |
| tci_band | Temperature Correction Intercepts | Deg. Kelvin |

**Table 10. VIIRS COP PFAAST Regression LUT Data Structure**

| Input | Description/Source | Units |
|---|---|---|
| pstd | Pressure | Milibars |
| tstd | Temperature | Deg. Kelvin |
| wstd | Moisture | gm/kg |
| ostd | Ozone | ppmv |
| coefd37 | None | None |
| coefo37 | None | None |
| coefs37 | None | None |
| coefl37 | None | None |
| coefc37 | None | None |
| coefd84 | None | None |
| coefo84 | None | None |
| coefs84 | None | None |
| coefl84 | None | None |
| coefc84 | None | None |
| coefd107 | None | None |
| coefo107 | None | None |
| coefs107 | None | None |
| coefl107 | None | None |
| coefc107 | None | None |
| coefd12 | None | None |
| coefo12 | None | None |
| coefs12 | None | None |

| Input | Description/Source | Units |
|---|---|---|
| coefl12 | None | None |
| coefc12 | None | None |

**Table 11. VIIRS CM Data Structure**

| Input | Description/Source | Units |
|---|---|---|
| cloud_confidence | Cloud Confidence Level | None |
| cloud_phase | Cloud Phase | None |
| snow_ice_sfc | None | None |
| cm_quality | None | None |
| sun_glint | None | None |
| non_cloud_obstruction | None | None |
| land_water | None | None |
| forest | None | None |

The VCM is a VIIRS IP and is described in the VIIRS Cloud Mask (VCM) Algorithm Theoretical Basis Document ATBD, 474-00033. The surface albedo and surface type both are set up in the COP source code rather than directly imported from files.

#### 2.1.2.1.1 Requirements for Input

The VIIRS moderate resolution SDR and VCM must be generated before COP is processed.

Ancillary atmospheric profiles, i.e. moisture, surface skin temperature, and temperature profiles must be generated before COP is processed. In addition, pre-calculated reflectance LUTs for ice and water clouds must be stored in the proper path/directories before COP is processed.

### 2.1.2.2 Outputs

EPS and optical depth are output for all clouds and illumination conditions. Additionally, cloud temperature is retrieved whenever the IR algorithms are used, which includes ice clouds (day and night) and water clouds at night. During daytime, water cloud temperature is not retrieved from these algorithms. It is instead calculated in the Cloud Top Parameters (CTP) unit. Table 12 describes the COP output data specifications. Table 13 describes the COP quality flag specifications. Table 14 describes the CTT quality flag specifications.

**Table 12. COP Output Data Specifications**

| Output | Type | Description/Source | Units/Valid Range |
|---|---|---|---|
| COT | Float32 | Cloud Optical Thickness | Unitless / 0.0 minimum |
| EPS | Float32 | Effective Particle Size | microns / 0.0 to 124.0 |
| CTT | Float32 | Cloud Top Temperature | Kelvin 180 - 260 |
| COP Quality Flag 0 | UInt8 | COP Quality Flag 0 | Bit-wise flags |
| COP Quality Flag1 | UInt8 | COP Quality Flag 1 | Bit-wise flags |
| COP Quality Flag2 | UInt8 | COP Quality Flag 2 | Bit-wise flags |

| Output | Type | Description/Source | Units/Valid Range |
|---|---|---|---|
| CTT Quality Flag | UInt8 | CTT Quality Flag | Bit-wise flags |

**Table 13. COP Quality Flag Specifications**

| Quality Flag – Byte 0 | Bit Field | Description/Source | Note |
|---|---|---|---|
| COP_QF_FLAG_SET | 0 | Overall pixel level flag | 1: cop quality flag is set;<br>0: not |
| COP_QF_COT_ICE_OUT_OF_BOUNDS | 1 | Ice cot out of bound<br>Day: (0.1-30)<br>Night: (0.1-30) | 1: out of bound;<br>0: not |
| COP_QF_COT_WATER_OUT_OF_BOUNDS | 2 | Water cot out of bound<br>Day: (0.1-30)<br>Night: (0.1-30) | 1: out of bound;<br>0: not |
| COP_QF_EPS_ICE_OUT_OF_BOUNDS | 3 | ice eps out of bound (1-50 micrometers) | 1: out of bound;<br>0: not |
| COP_QF_EPS_WATER_OUT_OF_BOUNDS | 4 | Water eps out of bound (1-50 micrometers) | 1: out of bound;<br>0: not |
| cloud phase | 5 - 7 | Cloud phase | 0: COP_PHASE_NOT_EXE<br>1: COP_PHASE_CIRRUS<br>2: COP_PHASE_OPQ_ICE<br>3: COP_PHASE_WATER<br>4: COP_PHASE_MIXED<br>5: COP_PHASE_MUL_LYR |
| **Quality Flag – Byte 1** | **Bit Field** | **Description/Source** | **Note** |
| COP_QF_DAY_WATER_CONVERGENCE | 0 | Iteration convergence | 1: convergent;<br>0: not |
| COP_QF_DAY_ICE_CONVERGENECE | 1 | Iteration convergence | 1: convergent;<br>0: not |
| COP_QF_WATER_COT_LT1_DAY | 2 | water cot < 1 at daytime | 1: yes;<br>0: not |
| COP_QF_ICE_COT_LT1_DAY | 3 | ice cot < 1 at daytime | 1: yes;<br>0: not |
| COP_QF_WATER_COT_LT1_NIGHT | 4 | water cot < 1 at nighttime | 1: yes;<br>0: not |
| COP_QF_ICE_COT_LT1_NIGHT | 5 | ice cot < 1 at nighttime | 1: yes;<br>0: not |
| COP_QF_EXCLUDE_SUN_GLINT | 6 | sun glint region | 1: in sun glint;<br>0: not |
| COP_QF_CLOUD_CONFIDENCE | 7 | Probably/confidently cloudy | 1: yes;<br>0: not |
| **Quality Flag – Byte 2** | **Bit Field** | **Description/Source** | **Note** |
| COP_QF_DEGRADED_ICE_COT_GT_TEN | 0 | Degraded conditions for ice cloud cot > 10 | 1: yes;<br>0: not |
| Bad SDR | 1 - 2 | Bad SDR flag data | 0:good<br>1:bad<br>2:no calibration |
| Unused | 3 - 7 | Not used | |

**Table 14. CTT Quality Flag Specifications**

| Quality Flag – Byte 0 | Bit Field | Description/Source | Note |
|---|---|---|---|
| CTT_QF_WATER_CTT_OUT_OF_BOUNDS | 0 | Water ctt out of bound | 1: out of bound;<br>0: not |

| Quality Flag – Byte 0 | Bit Field | Description/Source | Note |
|---|---|---|---|
| CTT_QF_ICE_CTT_OUT_OF_BOUNDS | 1 | Ice ctt out of bound | 1: out of bound;<br>0: not |
| CTT_QF_NIGHT_WATER_CONVERGENCE | 2 | IR ice ctt convergence for night water | 1: convergence;<br>0: not |
| CTT_QF_NIGHT_ICE_CONVERGENCE | 3 | IR ice ctt convergence for night ice | 1: convergence;<br>0: not |
| CTT_QF_DAY_ICE_CONVERGENCE | 4 | IR ice ctt convergence for day ice | 1: convergence;<br>0: not |
| Unused | 5 - 7 | Not used | |

#### 2.1.2.2.1 Requirements for Output

COP IP products are to be used for all the cloud algorithms; namely CTP, Cloud Cover/layers (CCL) and Cloud Base Height (CBH). In addition, the output products are used by snow ice. These IP products include COT, Cloud EPS (radius) and Cloud Top Temperature (CTT) (except for daytime water clouds).

**Figure 2. Flow Chart of COP/EPS with Other EDRs**

### 2.1.3 Algorithm Processing

COP can be thought of as two sets of algorithms. The solar algorithms are used for day time retrievals both for water and ice clouds. These algorithms use a set of pre-computed reflectance LUTs of the visible and Near-IR bands, i.e. M5, M8 and M10 bands. The simultaneous determination of COT and EPS involves two steps. First, a subset of the LUTs at M5 and M10 are determined by using a multi-dimensional interpolation scheme, at specified solar zenith angle, viewing zenith angle, relative azimuth angle and surface albedo. Second, COP is determined by minimizing the error with respect to the sensor measured reflectances. The original ATBD proposed the use of a 3-band approach including the M11 (2.25 micron) band, in retrieving COP but the software code has adopted instead a 2-band approach. Without considering the 2.25 micron band, the algorithm can be illustrated in Figure 3 below. Figure 3 displays the correlation between the VIIRS 0.672 and 1.61 mm reflectances (top) and between the 0.672 and 2.25 mm reflectances (bottom) for Cirrus clouds in US std atmosphere.

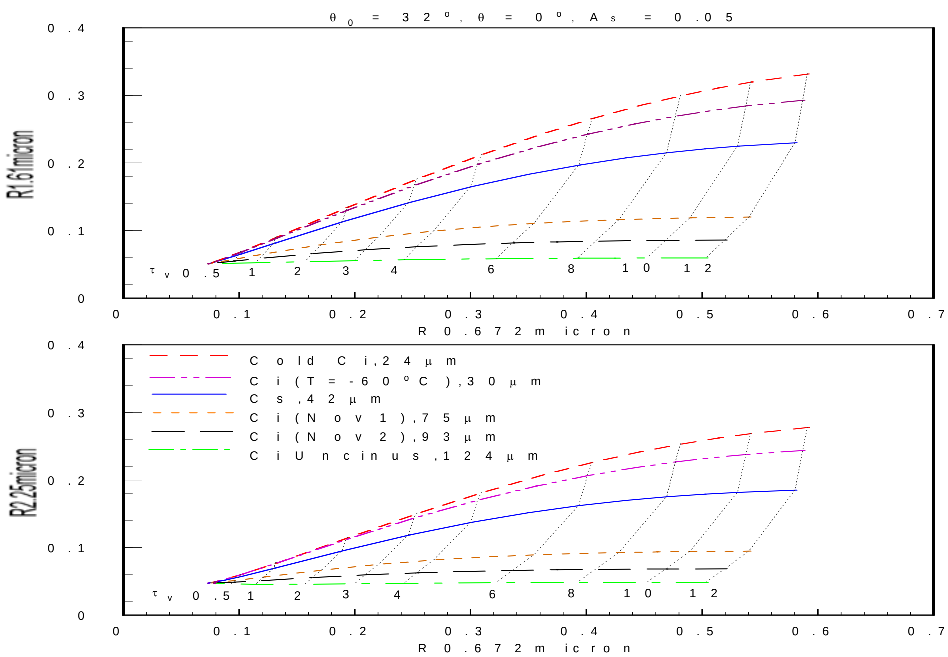


**Figure 3. Two-band Reflectances for Cirrus Clouds**

The three IR algorithms are used for night time water, night time ice, and day time ice cloud retrievals. For night time ice cloud, the IR algorithm first employs M14 and M16 to retrieve the Cloud Top Temperature. This follows with the calculation of emissivity using the M15 band radiance. Lastly, COT and EPS are calculated using parameterization equations. For day time ice clouds, however, COT and EPS are provided from the Solar ice cloud algorithm. Thus, the IR algorithm for daytime ice cloud is reduced to the retrieval of Cloud Top Temperatures which again use the M14 and M16 bands as are used for night time ice cloud. For night time water cloud, the transfer equations for the M12 and M15 bands are used to solve for CTT. Coupled with the parameterization equations for COT and EPS (in terms of CTT) the required COT and EPS are calculated by inversion of these equations. The transfer equations for all the bands are functionally similar. For example, for M12 and M15:

$$R_{M12} = (1 - e_{M12})\, R_{aM12} + e_{M12}\, B_{M12}(T_c)$$
$$R_{M15} = (1 - e_{M15})\, R_{aM15} + e_{M15}\, B_{M15}(T_c)$$

where RM12 , RaM12 , eM12 , and BM12(Tc) are the sensor measured radiance, clear sky radiance of scene, cloud emissivity and band averaged Planck function at Cloud Top Temperature.

The processing chain of the COP algorithm first performs the clear sky radiance (getAveClrSkyRad) calculations for daytime ice clouds, and nighttime ice and water clouds, then follows with the main COP processing function (DetermineCotEps).

#### 2.1.3.1 Main Module 1 - getAveClrSkyRad( )

**Returns Type:** Int32

Table 13 describes the input argument variables for getAveClrSkyRad.

**Table 15. Input Argument Variables Function getAveClrSkyRad( )**

| Name | Type | Initialization | Description |
|---|---|---|---|
| *aDataPtr | ViirsCopDataType | None | Pointer to data items |
| *mask | CM_STRUCT | None | Pointer to cloud mask |
| *work | WORK_STRUCT | None | Pointer to needed working data |
| *sdr | SDR_STRUCT | None | Pointer to SDR data |
| *anc | ANC_STRUCT | None | Pointer to ancillary data |
| *clr | CLR_STRUCT | None | Pointer to Clear sky radiance data |
| *sfc_emiss | SFC_EMISS_STRUCT | None | Pointer to Emissivity Data |

For the entire granule, the logic flow is as follows:

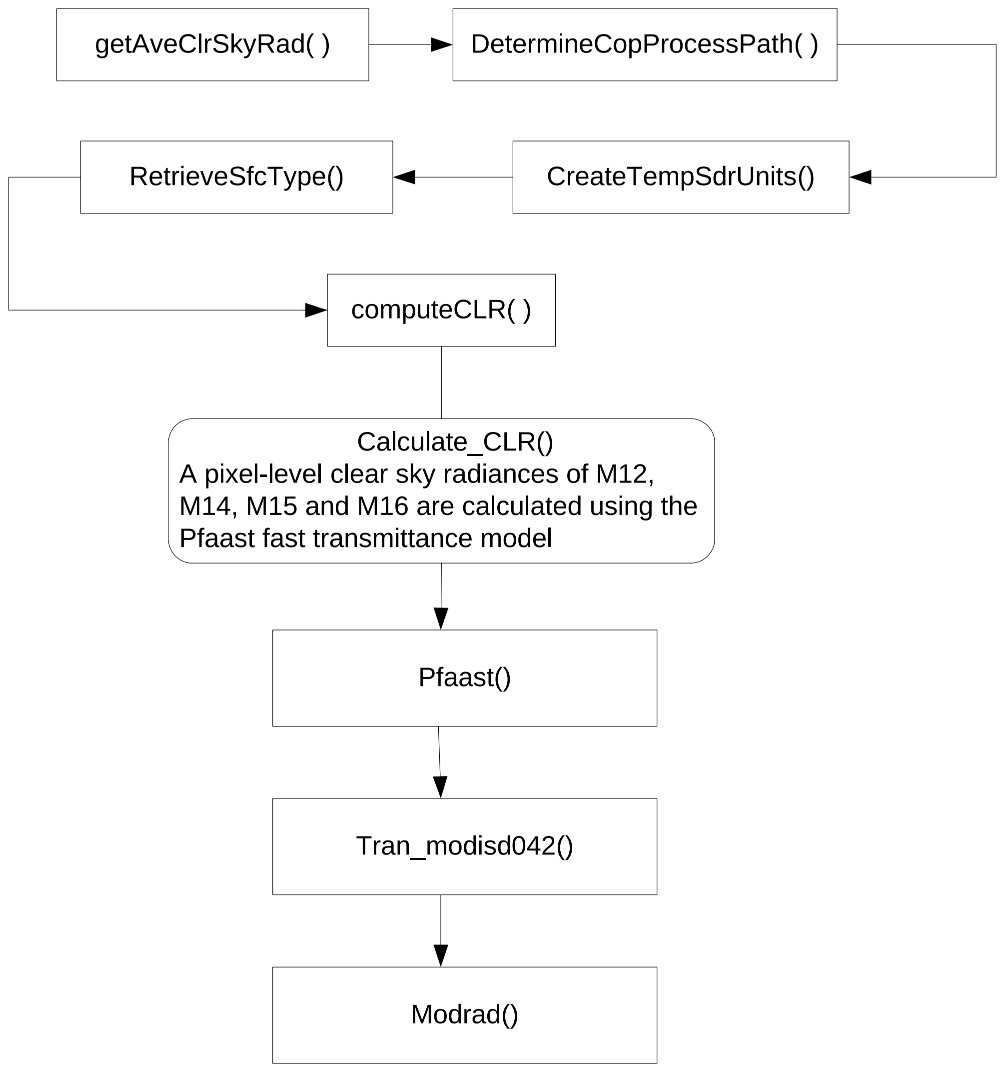


**Figure 4. Flow of getAveClrSkyRad Algorithm**

The functions DetermineCopProcessPath( ), CreateTempSdrUnits( ), and RetrieveSfcType( ) are discussed in the next section. Here, only the functions used by computeCLR( ) are discussed.

#### 2.1.3.2 DetermineCopProcessPath( )

This function determines the processing path for each pixel, based on the cloud confidence (cloudy or clear) and the cloud phase (ice, mixed, or water). Table 16 describes the input argument variables for DetermineCopProcessPath.

**Returns Type:** void

**Table 16. Input Argument Variables DetermineCopProcessPath( )**

| Name | Type | Description |
|---|---|---|

| Name | Type | Description |
|---|---|---|
| *mask | CM_STRUCT | Pointer to Cloud mask data |
| *work | WORK_STRUCT | Pointer to needed Working data |

### 2.1.3.3 CreateTempSdrUnits( )

**Returns Type:** void

**Table 17. Input Argument Variables CreateTempSdrUnits( )**

| Name | Type | Description |
|---|---|---|
| *sdr | SDR_STRUCT | Pointer to VIIRS SDR Data |
| *work | WORK_STRUCT | Pointer to Needed Work data |
| *geo | viirs_SDR_MOD_Fgeoloc_type | Pointer to VIIRS GEO Data |
| *thresholds | IngMsdCoefficients_ViirsCopStruct | Pointer to COP Coeffficient Data |

This function copies existing SDRs in to a temporary structure where the values can be modified as needed by Cloud Optical Properties algorithm. Table 17 describes the input argument variables for CreateTempSdrUnits().

### 2.1.3.4 RetrieveSfcType( )

**Returns Type:** Int32

**Table 18. Input Argument Variables RetrieveSfcType( )**

| Name | Type | Description |
|---|---|---|
| *anc | ANC_STRUCT | Pointer to Ancillary data |
| *mask | CM_STRUCT | Pointer to Cloud Mask data |

This function sets the surface type data from Cloud Mask. The surface type is described in Table 11, VIIRS CM data structure. This function will populate data inside the ANC_STRUCT for sfc_type and will contain information regarding surface type obtained from Cloud Mask input.

### 2.1.3.5 RetrieveSfcAlbedoEmiss( )

**Returns Type:** void

**Table 19. Input Argument Variables RetrieveSfcAlbedoEmiss( )**

| Name | Type | Description |
|---|---|---|
| *anc | ANC_STRUCT | Ancillary data structure (i) |
| *work | WORK_STRUCT | Work structure (i) |
| *aDataPtr | ViirsCopDataType | Config structure (i) |
| *sfc_albedo | SFC_ALBEDO_STRUCT | Surface Albedo structure (o) |
| *sfc_emiss | SFC_EMISS_STRUCT | Surface Emissivity structure (o) |

This function sets the surface albedo and surface emissivity for every daytime cloudy pixel after surface type is retrieved from VCM. The surface type is described in Table 11, VIIRS CM data structure. The ANC_STRUCT contains information regarding surface type obtained from Cloud Mask Input. Table 19 describes the input argument variables for RetrieveSfcAlbedoEmiss.

### 2.1.3.6 ComputeCLR()

This function sets up the loop to compute clear sky radiance calculated by pfaast().

**Return Type:** Int32

**Table 20. Input Argument Variables Function ComputeCLR( )**

| Name | Type | Initialization | Description |
|---|---|---|---|
| *aDataPtr | ViirsCopDataType | None | Pointer to data items |
| *work | WORK_STRUCT | None | Pointer to store needed working data |
| *sdr | SDR_STRUCT | None | Pointer to store SDR data |
| *anc | ANC_STRUCT | None | Pointer to store ancillary data |
| *mask | CM_STRUCT | None | Pointer to store cloud mask data |
| *clr | CLR_STRUCT | None | Pointer to store Clear sky radiance data |
| *sfc_emiss | SFC_EMISS_STRUCT | None | Pointer to store Emissivity Data |

### 2.1.3.7 Calculate_CLR()

This function sets up the required input of the surface emissivity and sensor zenith angle of the pixel for which clear radiances of M12, M14, M15 and M16 are computed. This function also sets up the ancillary data; temperature, moisture and pressure profiles to be used for pfaast(). Table 21 describes the input argument variables for Calculate_CLR.

**Return Type:** Int32

**Table 21. Input Argument Variables Function Calculate_CLR( )**

| Name | Type | Initialization | Description |
|---|---|---|---|
| *aDataPtr | ViirsCopDataType | None | Pointer to data items |
| *sdr | SDR_STRUCT | None | Pointer to store SDR data |
| *anc | ANC_STRUCT | None | Pointer to store ancillary data |
| *work | WORK_STRUCT | None | Pointer to store needed working data |
| *sfc_emiss | SFC_EMISS_STRUCT | None | Pointer to store emissivity data |
| *pix_locs | Int32 | None | Pointer to Pixel Location data |
| *ClrRad_M12 | Float32 | None | Pointer to store ClrRad_M12 |
| *ClrRad_M14 | Float32 | None | Pointer to store ClrRad_M14 |
| *ClrRad_M15 | Float32 | None | Pointer to store ClrRad_M15 |
| *ClrRad_M16 | Float32 | None | Pointer to store ClrRad_M16 |
| *path_rad37 | Float32 | None | Pointer to store path_rad 37 Value |
| *path_rad84 | Float32 | None | Pointer to store path_rad 84 Value |
| *path_rad107 | Float32 | None | Pointer to store path_rad 107 Value |
| *path_rad12 | Float32 | None | Pointer to store path_rad 12 Value |
| *tau_abv_cld37 | Float32 | None | Pointer to store tau_abv_cld37 value |
| *tau_abv_cld84 | Float32 | None | Pointer to store tau_abv_cld84 value |
| *tau_abv_cld107 | Float32 | None | Pointer to store tau_abv_cld107 value |
| *tau_abv_cld12 | Float32 | None | Pointer to store tau_abv_cld12 value |
| *pwater | Float32 | None | Pointer to store precip water value |

### 2.1.3.8 pfaast()

This Fortran function sets up the atmospheric temperature profile from space-down and interpolate the temperature to the preset 42 pressure levels as required for the pfaast fast transmittance operations. Table 22 describes the input argument variables for pfaast.

**Return Type:** void

**Table 22. Input Argument Variables Function pfaast( )**

| Name | Type | Initialization | Description |
|---|---|---|---|
| *temp_profile | Float32 | None | Pointer to temperature profile array |
| *mois_profile | Float32 | None | Pointer to moisture profile array |
| *pres_profile | Float32 | None | Pointer to pressure profile array |
| &avg_SenZen | Float32 | None | Reference value of average SenZen |
| &skin_temp | Float32 | None | Reference value to Surface Skin temperature |
| &BT_M15 | Float32 | None | Reference value to BT M15 |
| (Float32*)&esfc | Float32 | None | Pointer to esfc structure |
| *ClrRad_M12 | Float32 | None | Pointer to store ClrRad_M12 |
| *ClrRad_M14 | Float32 | None | Pointer to store ClrRad_M14 |
| *ClrRad_M15 | Float32 | None | Pointer to store ClrRad_M15 |
| *ClrRad_M16 | Float32 | None | Pointer to store ClrRad_M16 |
| *path_rad37 | Float32 | None | Pointer to store path_rad 37 Value |
| *path_rad84 | Float32 | None | Pointer to store path_rad 84 Value |
| *path_rad107 | Float32 | None | Pointer to store path_rad 107 Value |
| *path_rad12 | Float32 | None | Pointer to store path_rad 12 Value |
| *tau_abv_cld37 | Float32 | None | Pointer to store tau_abv_cld37 value |
| *tau_abv_cld84 | Float32 | None | Pointer to store tau_abv_cld84 value |
| *tau_abv_cld107 | Float32 | None | Pointer to store tau_abv_cld107 value |
| *tau_abv_cld12 | Float32 | None | Pointer to store tau_abv_cld12 value |
| *pwater | Float32 | None | Pointer to store precip water value |
| aDataPtr->pfaast_lut | Float32 | None | Pointer to PFAAST-LUT |
| aDataPtr->ir_lut | Float32 | None | Pointer to IR-LUT |

### 2.1.3.9 tran_modis042()

This Fortran function computes the level-to-space transmittances for M12, M14, M15 and M16 bands. There are 42 pressure levels for which the transmittances are calculated. Table 23 describes the input argument variables for tran_modis042.

**Return Type:** Float32

**Table 23. Input Argument Variables Function tran_modis042( )**

| Name | Type | Initialization | Description |
|---|---|---|---|
| Temp | Float32 | Read Temperature profile | Store ancillary data |
| Wvmr | Float32 | Read Moisture profile | Store ancillary data |
| Ozmr | Float32 | Read ozone profile | Store ancillary data |
| Theta | Float32 | Read SDR file | Store sensor zenith angle from SDR data |

### 2.1.3.10 modrad()

This Fortran function computes the clear sky radiances for M12, M14, M15 and M16 bands. It also computes the portion of the clear radiance from contribution of the water vapor emission above the cloud top. Table 24 describes the input argument variables for modrad.

**Return Type:** Float32

**Table 24. Input Argument Variables Function modrad( )**

| Name | Type | Initialization | Description |
|---|---|---|---|
| Tskin | Float32 | Read Surface temeperature profile | Store ancillary data |

| Name | Type | Initialization | Description |
|---|---|---|---|
| Esfc | Float32 | Read surface emissivity | Store ancillary data |
| Temp | Float32 | Read Temperature profile | Store ancillary data |
| Tau | Float32 | Calculated by tran_modis042 | Store data |
| Band | Int32 | Set to 1 to 4 | 1-4 corresponding to M12, M14, M15 and M16 |
| Nl | Int32 | Set to 42 | Corresponding to 42 pressure levels |
| lev_cld | Int32 | Calculate in pfaast function | Estimated pressure level at which the cloud is located |

#### 2.1.3.11 Main Module 2 - DetermineCotEps( )

Following the calculation of clear sky radiances is the main cloud optical processing function. This function, DetermineCotEps, is the main controlling function. It makes the necessary calls to read the cloud mask, determine the processing path, compute the surface albedo, determine the COT, EPS, and CTT for both water and ice clouds (CTT not determined for daytime water clouds), and finally produce the output IP files. Table 25 describes the input argument variables for DetermineCotEps.

**Returns Type:** int

**Table 25. Input Argument Variables Function DetermineCotEps( )**

| Name | Type | Initialization | Description |
|---|---|---|---|
| *aDataPtr | ViirsCopDataType | Read configuration file | Store configuration info |
| *mask | CM_STRUCT | Read CM file | Store cloud mask |
| *sdr | SDR_STRUCT | Read SDR file | Store SDR data |
| *aux | ANC_STRUCT | Read ancillary files | Store ancillary data |
| *work | WORK_STRUCT | Working info | Store needed working data |
| *cop_ips | IP_STRUCT | None | Store data needed for IP output |
| *water_cop | IP_STRUCT | None | Store data needed for IP output |
| *ice_cop | IP_STRUCT | None | Store data needed for IP output |
| *sfc_albedo | ALBEDO_STRUCT | None | Store surface Albedo data |
| *sfc_emiss | SFC_EMISS_STRUCT | None | Store surface emissivity data |
| *clr | CLR_STRUCT | None | Clear sky radiance structure |
| *precalc_day_water | PRECALC_DAY_WATER | None | Store water cloud LUT |
| *precalc_day_ice | PRECALC_DAY_ICE | None | Store ice cloud LUT |
| *aStopPtr | Int32 | None | A pointer to indicate success or fail |

The data structures are defined in a header file called cop.h. See this header file for detailed information.

The Figure 5 flow chart illustrates the processing chain of the main module at pixel level.

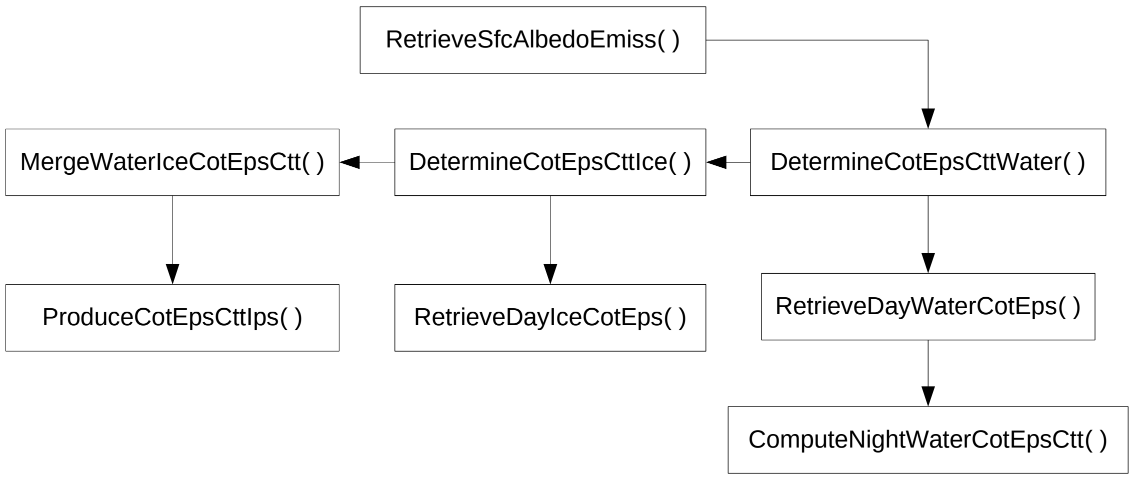


**Figure 5. Flow Chart of DetermineCotEps Algorithm**

#### 2.1.3.11.1 Usage

This function returns an integer type of status (PRO_FAIL or PRO_SUCCESS). The memory of argument variables in Table 11 have to be allocated before calling the function. Some variables need to be initialized as indicated in Table 11. The usage is straightforward. The calculation for night ice cloud is done by a subroutine called by the RetrieveDayIceCotEps() routine.

### 2.1.3.12 SdrData_Rescale()

**This function rescales SDR radiance values for M12, M14, M15 and M16 bands. This function uses the values calculated in pfaast() for path_emis_rad and tran_abv_cld to rescale SDR radiances.**Return Type: **Int32**

**Table 26. Input Argument Variables SdrData_Rescale( )**

| Name | Type | Description |
|---|---|---|
| *mask | CM_STRUCT | Pointer to Cloud mask data |
| *sdr | SDR_STRUCT | Pointer to SDR data |
| *work | WORK_STRUCT | Pointer to needed work data |
| *clr | CLR_STRUCT | Pointer to Clear Sky Radiance Data |
| *geo | viirs_SDR_MOD_Fgeoloc_type | Poiner to VIIRS GEO Data |
| *thresholds | IngMsdCoefficients_ViirsCopStruct | Pointer to COP Coefficient Data |

### 2.1.3.13 DetermineCotEpsCttWater( )

**Returns Type:** int

**Table 27. Input Argument Variables DetermineCotEpsCttWater( )**

| Name | Type | Description |
|---|---|---|
| *work | WORK_STRUCT | Work structure (i) |
| *aDataPtr | ViirsCopDataType | Structure of pointers to data |
| *sdr | SDR_STRUCT | VIIRS SDR Data structure (i) |
| *mask | CM_STRUCT | Cloud mask (i) |
| *water_cop | IP_STRUCT | IP output data (o) |
| *sfc_albedo | SFC_ALBEDO_STRUCT | Surface albedo structure |
| *sfc_emiss | SFC_EMISS_STRUCT | Surface Emissivity structure (o) |
| *clr | CLR_STRUCT | Clear Sky Radiance |
| *precalc_day_water | PRECALC_DAY_WATER | Store water cloud LUT |

This function retrieves the daytime water cloud COT and EPS from the water cloud LUT, computes the nighttime water cloud COT, EPS, and CTT, merges their results and stores them in the output IP_STRUCT. Table 27 describes the input argument variables for DetermineCotEpsCttWater.

### 2.1.3.14 RetrieveDayWaterCotEps( )

This function retrieves the daytime water cloud COT and EPS from the water cloud LUT and stores the results in the output IP_STRUCT water_cop. Table 28 describes the input argument variables for RetrieveDayWaterCotEps.

**Returns Type:** int

**Table 28. Input Argument Variables RetrieveDayWaterCotEps( )**

| Name | Type | Description |
|---|---|---|
| *config | IngMsdCoefficients_ViirsCopStruct | Config structure (i) |
| *work | WORK_STRUCT | Work structure (i) |
| *sdr | SDR_STRUCT | VIIRS SDR Data structure (i) |
| *mask | CM_STRUCT | Cloud mask (i) |
| *water_cop | IP_STRUCT | IP output data (o) |
| *sfc_albedo | SFC_ALBEDO_STRUCT | Surface albedo structure |
| *sfc_emiss | SFC_EMISS_STRUCT | Surface Emissivity structure (o) |
| *precalc_day_water | PRECALC_DAY_WATER | Store water cloud LUT |
| *selected_bins | ViirsCopBinsSelectedType | Values selected from LUT |
| *chosen_water_refl | ViirsCopInterpWaterReflType | Interpolated values from the Water LUT |

### 2.1.3.15 ComputeNightWaterCotEpsCtt( )

This function computes the nighttime water cloud COT and EPS. The results are stored in the IP_STRUCT water_cop. Table 29 describes the input argument variables for ComputeNightWaterCotEpsCtt.

**Returns Type:** int

**Table 29. Input Argument Variables ComputeNightWaterCotEpsCtt( )**

| Name | Type | Description |
|---|---|---|
| *aDataPtr | ViirsCopDataType | Structure of pointers to data |
| *work | WORK_STRUCT | Work structure (i) |
| *sdr | SDR_STRUCT | VIIRS SDR Data structure (i) |
| *water_cop | IP_STRUCT | IP output data (o) |
| *clr | CLR_STRUCT | Clear Sky Radiance |

### 2.1.3.16 extract_water_lut_values()

This function reads water cloud LUT files and stores the data in a multi-dimensional array. Based on the input parameters, certain values are stored in the output structure.

**Returns Type:** int

**Table 30. Input Argument Variables extract_water_lut_values ( )**

| Name | Type | Description |
|---|---|---|
| *solzen | int | Solar Zenith Angle |
| *senzen | int | Sensor Zenith Angle |
| *relaz | int | Relative Azimuth Angle |
| *alb_M5 | int | Albedo for band M5 |
| *alb_M8 | int | Albedo for band M8 |
| *alb_M10 | int | Albedo for band M10 |
| *emiss_M5 | int | Emissivity for band M5 |
| *emiss_M8 | int | Emissivity for band M8 |
| *emiss_M10 | int | Emissivity for band M10 |
| *sfc_albedo | SFC_ALBEDO_STRUCT | Pointer to Surface Albedo data |
| *sfc_emiss | SFC_EMISS_STRUCT | Pointer to Surface Emissivity data |
| *precalc_day_water | PRECALC_DAY_WATER | Pointer to pre-calculated day water data |
| *work | WORK_STRUCT | Pointer to needed work data |
| *sdr | SDR_STRUCT | Pointer to VIIRS SDR data |
| *config | IngMsdCoefficients_ViirsCopStruct | Pointer to configuration information |
| *mask | CM_STRUCT | Pointer to cloud mask data |
| *chosen_water_refl | ViirsCopInterpWaterReflType | Poitner to Interpolated Water Reflectance |

### 2.1.3.17 ChooseDayWaterCotEps()

This function computes the square sum of the residuals of the measured reflectances and the pre-calculated reflectances for both sets of values retrieved from the LUT and sets the day water COT and EPS values equal to the smallest.

**Returns Type:** Int32

**Table 31. Input Argument Variables ChooseDayWaterCotEps ( )**

| Name | Type | Description |
|---|---|---|
| *precalc_day_water | PRECALC_DAY_WATER | Pointer to precalcuated ay water values |
| *sdr | SDR_STRUCT | Pointer to VIIRS SDR data |
| *water_cop | IP_STRUCT | Pointer to a structure for storing day time output values |
| *work | WORK_STRUCT | Pointer to needed work data |
| *config | IngMsdCoefficients_ViirsCopStruct | Poitner to configuration information |

### 2.1.3.18 DetermineCotEpsCttIce( )

This function retrieves the daytime ice cloud COT and EPS from the ice cloud LUT, computes the nighttime ice cloud COT, EPS, and CTT, merges their results and stores them in the output IP_STRUCT. Table 32 describes the input argument variables for DetermineCotEpsCttIce.

**Returns Type:** int

**Table 32. Input Argument Variables DetermineCotEpsCttIce( )**

| Name | Type | Description |
|---|---|---|
| *work | WORK_STRUCT | Work structure (i) |
| *aDataPtr | ViirsCopDataType | Config structure (i) |
| *sdr | SDR_STRUCT | VIIRS SDR Data structure (i) |
| *mask | CM_STRUCT | Cloud mask (i) |
| *ice_cop | IP_STRUCT | IP output data (o) |
| *sfc_albedo | SFC_ALBEDO_STRUCT | Surface albedo structure |
| *sfc_emiss | SFC_EMISS_STRUCT | Surface Emissivity structure (o) |
| *clr | CLR_STRUCT | Clear Sky Radiance |
| *precalc_day_ice | PRECALC_DAY_ICE | Store ice cloud LUT |

### 2.1.3.19 RetrieveDayIceCotEps( )

This function retrieves the daytime ice cloud COT and EPS from the ice cloud LUT and stores the results in the output IP_STRUCT ice_cop. Table 33 describes the input argument variables for RetrieveDayIceCotEps.

**Returns Type:** int

**Table 33. Input Argument Variables RetrieveDayIceCotEps( )**

| Name | Type | Description |
|---|---|---|
| *aDataPtr | ViirsCopDataType | Config structure (i) |
| *work | WORK_STRUCT | Work structure (i) |
| *sdr | SDR_STRUCT | VIIRS SDR Data structure (i) |
| *mask | CM_STRUCT | Cloud mask (i) |
| *ice_cop | IP_STRUCT | IP output data (o) |
| *sfc_albedo | SFC_ALBEDO_STRUCT | Surface albedo structure |
| *sfc_emiss | SFC_EMISS_STRUCT | Surface Emissivity structure (o) |
| *precalc_day_ice | PRECALC_DAY_ICE | Store ice cloud LUT |
| *clr | CLR_STRUCT | Clear Sky Radiance |

### 2.1.3.20 extract_water_lut_values()

This function reads ice cloud LUT files and stores the data in a multi-dimensional array. Based on the input parameters, certain values are stored in the output structure.

**Returns Type:** int

**Table 34. Input Argument Variables extract_water_lut_values ( )**

| Name | Type | Description |
|---|---|---|
| *solzen | int | Solar Zenith Angle |
| *senzen | int | Sensor Zenith Angle |
| *relaz | int | Relative Azimuth Angle |
| *alb_M5 | int | Albedo for band M5 |
| *alb_M8 | int | Albedo for band M8 |
| *alb_M10 | int | Albedo for band M10 |
| *alb_M12 | int | Albedo for band M12 |
| *emiss_M5 | int | Emissivity for band M5 |
| *emiss_M8 | int | Emissivity for band M8 |
| *emiss_M10 | int | Emissivity for band M10 |
| *emiss_M12 | int | Emissivity for band M12 |

| Name | Type | Description |
|---|---|---|
| *sfc_albedo | SFC_ALBEDO_STRUCT | Pointer to Surface Albedo data |
| *sfc_emiss | SFC_EMISS_STRUCT | Pointer to Surface Emissivity data |
| *precalc_day_water | PRECALC_DAY_ICE | Pointer to pre-calculated day ice data |
| *work | WORK_STRUCT | Pointer to needed work data |
| *sdr | SDR_STRUCT | Pointer to VIIRS SDR data |
| *config | IngMsdCoefficients_ViirsCopStruct | Pointer to configuration information |
| *mask | CM_STRUCT | Pointer to cloud mask data |
| *chosen_ice_refl | ViirsCopInterpIceReflType | Poitner to Interpolated Ice Reflectance |

#### 2.1.3.21 ChooseDayIceCotEps()

This function computes the square sum of the residules of the measured reflectances and the pre-calculated reflectances for both sets of values retrieved from the LUT and sets the day ice COT and EPS values equal to the smallest.

**Returns Type:** Int32

**Table 35. Input Argument Variables ChooseDayIceCotEps ( )**

| Name | Type | Description |
|---|---|---|
| *aDataPtr | ViirsCopDataType | Pointer to configuration information |
| *precalc_day_ice | PRECALC_DAY_ICE | Pointer to pre-calculated day ice values |
| *sdr | SDR_STRUCT | Pointer to VIIRS SDR struct |
| *ice_cop | IP_STRUCT | Pointer to Ice Cloud COT, EPS and CTT |
| *work | WORK_STRUCT | Pointer to needed work data |
| *mask | CM_STRUCT | Pointer to cloud mask data |
| *clr | CLR_STRUCT | Pointer to Clear Sky Radiance values |

#### 2.1.3.22 MergeWaterIceCotEpsCtt( )

**Returns Type:** int

**Table 36. Input Argument Variables MergeWaterIceCotEpsCtt( )**

| Name | Type | Description |
|---|---|---|
| *aDataPtr | ViirsCopDataType | Config structure (i) |
| *work | WORK_STRUCT | Work structure (i) |
| *sdr | SDR_STRUCT | VIIRS SDR Data structure (i) |
| *water_cop | IP_STRUCT | IP output data (o) |
| *ice_cop | IP_STRUCT | IP output data (o) |
| *cop_ips | IP_STRUCT | Store merged water/ice property |
| *mask | CM_STRUCT | Cloud mask (i) |

This function merges out the COT field, the EPS field, and the INWCTT field from individual structures to the one output data item. Table 36 describes the input argument variables for MergeWaterIceCotEpsCtt.

### 2.1.4 Graceful Degradation

#### 2.1.4.1 Graceful Degradation Inputs

There is one case where input graceful degradation is indicated in the VIIRS Cloud Optical Properties.

1. An input retrieved for the algorithm had its N_Graceful_Degradation metadata field set to YES (propagation).

Table 37 details the instances of this case. Note that the shaded cells indicate that the graceful degradation was done upstream at product production.

**Table 37. Graceful Degradation**

| Input Data Description | Baseline Data Source | Primary Backup Data Source | Secondary Backup Data Source | Tertiary Backup Data Source | Graceful Degradation Done Upstream |
|---|---|---|---|---|---|
| Atmospheric Temperature Profile | VIIRS_GD_11.4.3 NCEP | VIIRS_GD_11.4.3 NCEP (Extended Forecast) | N/A | N/A | Yes |
| Corresponding Pressure Levels | VIIRS_GD_11.4.1 NCEP | VIIRS_GD_11.4.1 NCEP (Extended Forecast) | N/A | N/A | Yes |
| Atmospheric Moisture Profile | VIIRS_GD_11.4.2 NCEP | VIIRS_GD_11.4.2 NCEP (Extended Forecast) | N/A | N/A | Yes |
| Surface Skin Temperature Profile | VIIRS-ANC-Temp-Skin-Mod-Gran (NCEP-ANC-Int) | VIIRS-ANC-Temp-Skin-Mod-Gran (NCEP-ANC-Int) Extended Forecast (FNMOC-ANC-Int) (+EF) | VIIRS-ANC-Temp-Skin-Mod-Gran Climatology | N/A | Yes |

### 2.1.4.2 Graceful Degradation Processing

None.

### 2.1.4.3 Graceful Degradation Outputs

None.

### 2.1.5 Exception Handling

The COP algorithm initially attempts to process all pixels that are not bow-tie deleted. The VIIRS Cloud Mask (VCM) output is used to further control which pixels are processed. Invalid SDR pixel values are also not processed.

### 2.1.6 Data Quality Monitoring

The COP algorithm produces an IP and therefore quality assessment and diagnostics are not done in this algorithm. Any Data Quality Notification (DQN) produced for the Cloud algorithms are handled by the respective cloud EDR processes.

### 2.1.7 Computational Precision Requirements

The COP algorithm does calculations with both 32-bit floats and 64-bit doubles to produce output products using 32-bit floats.

The surface emissivity and surface albedo values are retrieved from pre-computed average values for certain surface types. If more accurate emissivity and albedo values are required, they could be computed or a more detailed breakdown of emissivity and albedo per surface type could be developed instead of using the few surface types that the cloud mask provides.

### 2.1.8 Algorithm Support Considerations

- DMS should be up and running. All the data (primary or secondary) needed for the VCM calculations must be available in the DMS for the successful completion of the process.
- INF must be up and running, so the process can retrieve the task from INF and also send messages to INF upon successful completion or failure to complete the process.
- A C ++ compiler is necessary to compile the COP source code.
- The PRO libraries are available.
- The imake files can create the Makefile used to compile the COP algorithm.

### 2.1.9 Assumptions and Limitations

#### 2.1.9.1 Assumptions

There are several assumptions involved in the theoretical development of retrieval algorithms for cloud effective particle size and optical depth:

- For the retrieval of ice cloud parameters, representative ice crystal size distributions are based on *in situ* observations obtained during field experiments over mid-latitude areas. Ice crystals are assumed to be randomly oriented, and the ice crystal habits considered are solid columns and plates.
- For the retrieval of water cloud parameters, a gamma function is assumed to be the typical droplet size distribution, with water droplets assumed to be spherical in shape.
- Radiative transfer within the cloud is assumed to be plane-parallel.

#### 2.1.9.2 Limitations

There are also a few limitations of the current algorithms:

- The solar and IR algorithms are applicable to a single-layer cirrus or water cloud. Use of these algorithms with multi-layer clouds will result in larger error than those for single-layer clouds.
- These algorithms do not adequately address mixed phase clouds which occur frequently around the globe. In that, the algorithm treats mixed phase clouds as water clouds.
- The IR algorithms do not address situations in which the clear radiance is less than the cloudy radiance, which is prevalent during polar winter.
- The solar and IR algorithms do not adequately address overlap clouds.
- The IR algorithms do not adequately address Polar winter (black stratus), when the cloudy radiances would exceed the clear sky radiances.

# 3.0 GLOSSARY/ACRONYM LIST

## 3.1 Glossary

Table 38 contains terms most applicable for this OAD.

**Table 38. Glossary**

| Term | Description |
|---|---|
| Algorithm | A formula or set of steps for solving a particular problem. Algorithms can be expressed in any language, from natural languages like English to mathematical expressions to programming languages like FORTRAN. On NPOESS, an algorithm consists of:<br>1. A theoretical description (i.e., science/mathematical basis)<br>2. A computer implementation description (i.e., method of solution)<br>3. A computer implementation (i.e., code) |
| Algorithm Configuration Control Board (ACCB) | Interdisciplinary team of scientific and engineering personnel responsible for the approval and disposition of algorithm acceptance, verification, development and testing transitions. Chaired by the Algorithm Implementation Process Lead, members include representatives from IWPTB, Systems Engineering & Integration IPT, System Test IPT, and IDPS IPT. |
| Algorithm Verification | Science-grade software delivered by an algorithm provider is verified for compliance with data quality and timeliness requirements by Algorithm Team science personnel. This activity is nominally performed at the IWPTB facility. Delivered code is executed on compatible IWPTB computing platforms. Minor hosting modifications may be made to allow code execution. Optionally, verification may be performed at the Algorithm Provider's facility if warranted due to technical, schedule or cost considerations. |
| Ancillary Data | Any data which is not produced by the NPOESS System, but which is acquired from external providers and used by the NPOESS system in the production of NPOESS data products. |
| Auxiliary Data | Auxiliary Data is defined as data, other than data included in the sensor application packets, which is produced internally by the NPOESS system, and used to produce the NPOESS deliverable data products. |
| EDR Algorithm | Scientific description and corresponding software and test data necessary to produce one or more environmental data records. The scientific computational basis for the production of each data record is described in an ATBD. At a minimum, implemented software is science-grade and includes test data demonstrating data quality compliance. |
| Environmental Data Record (EDR) | *[IORD Definition]*<br>Data record produced when an algorithm is used to convert Raw Data Records (RDRs) to geophysical parameters (including ancillary parameters, e.g., cloud clear radiation, etc.).<br>*[Supplementary Definition]*<br>An Environmental Data Record (EDR) represents the state of the environment, and the related information needed to access and understand the record. Specifically, it is a set of related data items that describe one or more related estimated environmental parameters over a limited time-space range. The parameters are located by time and Earth coordinates. EDRs may have been resampled if they are created from multiple data sources with different sampling patterns. An EDR is created from one or more NPOESS SDRs or EDRs, plus ancillary environmental data provided by others. EDR metadata contains references to its processing history, spatial and temporal coverage, and quality. |
| Model Validation | The process of determining the degree to which a model is an accurate representation of the real-world from the perspective of the intended uses of the model. [Ref.: DoDD 5000.59-DoD Modeling and Simulation Management] |
| Model Verification | The process of determining that a model implementation accurately represents the developer's conceptual description and specifications. [Ref.: DoDD 5000.59-DoD Modeling and Simulation Management] |
| Operational Code | Verified science-grade software, delivered by an algorithm provider and verified by IWPTB, is developed into operational-grade code by the IDPS IPT. |
| Operational-Grade Software | Code that produces data records compliant with the System Specification requirements for data quality and IDPS timeliness and operational infrastructure. The software is modular relative to the IDPS infrastructure and compliant with IDPS application programming interfaces (APIs) as specified for TDR/SDR or EDR code. |

| Term | Description |
|---|---|
| Raw Data Record (RDR) | *[IORD Definition]*<br>Full resolution digital sensor data, time referenced and earth located, with absolute radiometric and geometric calibration coefficients appended, but not applied, to the data. Aggregates (sums or weighted averages) of detector samples are considered to be full resolution data if the aggregation is normally performed to meet resolution and other requirements. Sensor data shall be unprocessed with the following exceptions: time delay and integration (TDI), detector array non-uniformity correction (i.e., offset and responsivity equalization), and data compression are allowed. Lossy data compression is allowed only if the total measurement error is dominated by error sources other than the data compression algorithm. All calibration data will be retained and communicated to the ground without lossy compression.<br>*[Supplementary Definition]*<br>A Raw Data Record (RDR) is a logical grouping of raw data output by a sensor, and related information needed to process the record into an SDR or TDR. Specifically, it is a set of unmodified raw data (mission and housekeeping) produced by a sensor suite, one sensor, or a reasonable subset of a sensor (e.g., channel or channel group), over a specified, limited time range. Along with the sensor data, the RDR includes auxiliary data from other portions of NPOESS (space or ground) needed to recreate the sensor measurement, to correct the measurement for known distortions, and to locate the measurement in time and space, through subsequent processing. Metadata is associated with the sensor and auxiliary data to permit its effective use. |
| Retrieval Algorithm | A science-based algorithm used to 'retrieve' a set of environmental/geophysical parameters (EDR) from calibrated and geolocated sensor data (SDR). Synonym for EDR processing. |
| Science Algorithm | The theoretical description and a corresponding software implementation needed to produce an NPP/NPOESS data product (TDR, SDR or EDR). The former is described in an ATBD. The latter is typically developed for a research setting and characterized as "science-grade". |
| Science Algorithm Provider | Organization responsible for development and/or delivery of TDR/SDR or EDR algorithms associated with a given sensor. |
| Science-Grade Software | Code that produces data records in accordance with the science algorithm data quality requirements. This code, typically, has no software requirements for implementation language, targeted operating system, modularity, input and output data format or any other design discipline or assumed infrastructure. |
| SDR/TDR Algorithm | Scientific description and corresponding software and test data necessary to produce a Temperature Data Record and/or Sensor Data Record given a sensor's Raw Data Record. The scientific computational basis for the production of each data record is described in an Algorithm Theoretical Basis Document (ATBD). At a minimum, implemented software is science-grade and includes test data demonstrating data quality compliance. |
| Sensor Data Record (SDR) | *[IORD Definition]*<br>Data record produced when an algorithm is used to convert Raw Data Records (RDRs) to calibrated brightness temperatures with associated ephemeris data. The existence of the SDRs provides reversible data tracking back from the EDRs to the Raw data.<br>*[Supplementary Definition]*<br>A Sensor Data Record (SDR) is the recreated input to a sensor, and the related information needed to access and understand the record. Specifically, it is a set of incident flux estimates made by a sensor, over a limited time interval, with annotations that permit its effective use. The environmental flux estimates at the sensor aperture are corrected for sensor effects. The estimates are reported in physically meaningful units, usually in terms of an angular or spatial and temporal distribution at the sensor location, as a function of spectrum, polarization, or delay, and always at full resolution. When meaningful, the flux is also associated with the point on the Earth geoid from which it apparently originated. Also, when meaningful, the sensor flux is converted to an equivalent top-of-atmosphere (TOA) brightness. The associated metadata includes a record of the processing and sources from which the SDR was created, and other information needed to understand the data. |

| Term | Description |
|---|---|
| Temperature Data Record (TDR) | *[IORD Definition]*<br>Temperature Data Records (TDRs) are geolocated, antenna temperatures with all relevant calibration data counts and ephemeris data to revert from T-sub-a into counts.<br>*[Supplementary Definition]*<br>A Temperature Data Record (TDR) is the brightness temperature value measured by a microwave sensor, and the related information needed to access and understand the record. Specifically, it is a set of the corrected radiometric measurements made by an imaging microwave sensor, over a limited time range, with annotation that permits its effective use. A TDR is a partially-processed variant of an SDR. Instead of reporting the estimated microwave flux from a specified direction, it reports the observed antenna brightness temperature in that direction. |

## 3.2 Acronyms

Table 39 contains terms most applicable for this OAD.

**Table 39. Acronyms**

| Acronym | Description |
|---|---|
| AM&S | Algorithms, Models & Simulations |
| API | Application Programming Interfaces |
| ARP | Application Related Product |
| CCL | Cloud Cover/Layers |
| CDFCB-X | Common Data Format Control Book - External |
| COP | Cloud Optical Properties |
| COT | Cloud Optical Thickness |
| CTT | Cloud Top Temperature |
| DMS | Data Management Subsystem |
| DPIS ICD | Data Processor Inter-subsystem Interface Control Document |
| DQTT | Data Quality Test Table |
| EPS | Effective Particle Size |
| IIS | Intelligence and Information Systems |
| INF | Infrastructure |
| ING | Ingest |
| IP | Intermediate Product |
| LUT | Look-Up Table |
| MDFCB | Mission Data Format Control Book |
| PRO | Processing |
| QF | Quality Flag |
| RTM | Radiative Transfer Model |
| SDR | Sensor Data Records |
| SI | Software Item or International System of Units |
| TBD | To Be Determined |
| TBR | To Be Resolved |
| TOA | Top of the Atmosphere |
| VCM | VIIRS Cloud Mask |

# 4.0 OPEN ISSUES

**Table 40. TBXs**

| TBX ID | Title/Description | Resolution Date |
|---|---|---|
| None | | |
| | | |