**Global Precipitation Mission (GPM)**

**Ground Validation System**

**GPM Validation Network Volume Matching Script Descriptions and Execution**

**November, 2016**

Goddard Space Flight Center

Greenbelt, Maryland 20771



**Document History**

| Document Version | Date | Changes |
| --- | --- | --- |
| Draft | November, 2016 | Initial draft |

# Table of Contents

1. Introduction 4

2. Script Summaries – Baseline Matchups 5

2.1 do\_DPR\_GeoMatch.sh 5

2.2 do\_DPRGMI\_GeoMatch.sh 9

2.2.1 do\_DPRGMI\_GeoMatch\_from\_ControlFiles.sh 12

2.3 do\_GMI\_GeoMatch.sh 15

2.4 do\_GR\_HS\_MS\_NS\_GeoMatch.sh 18

2.5 do\_DPR2GR\_GeoMatch.sh 21

3. Script Summaries - Non-routine and legacy matchup scripts 24

3.1 do\_DPR\_GeoMatch\_NPOL\_MD\_rules.sh 24

3.2 do\_DPRtoGMI\_OrbitMatch.sh 28

3.3 do\_PR\_GeoMatch.sh 30

3.4 doTMIGeoMatch4NewRainCases.sh 33

# Introduction

The Global Precipitation Measurement (GPM) Validation Network (VN) performs volume matches of data from the satellite-borne TRMM Precipitation Radar (PR), GPM Dual-frequency Precipitation Radar (DPR) and GPM, TRMM and constellation satellite Microwave Imagers (GMI, TMI, etc.) to coincident ground-based scanning weather radar (GR) observations for rain events occurring at the ground radar sites.

As part of the data ingest process of PR and DPR data, the VN software pre-analyzes the satellite radar products over the ground radars within the satellite data swath to determine the areal extent of precipitation of each rain type within 100 km range of the GR site, and stores this information as raw metadata in the VN PostGRESQL database ‘gpmgv’, in the ‘event\_meta\_numeric’ table. This part happens automatically with no manual intervention required. A sequence of SQL queries in the file rainCases100kmAddNewEvents.sql, which is run under the ‘***psql***’ database utility called at the beginning of most volume matching scripts, analyzes the raw metadata and determines the number of points within 100 km of the radar that indicate ‘rain certain’ in the PR or DPR data, the percent of the 100-km range limit area covered by the satellite radar swath, and the percentages of this area indicating stratiform and convective rain type, and stores these values with identifying information in the database table ‘rainy100inside100’ for new rain events not yet tabulated. The satellite/site specific rain events tallied in this table define and limit the data to be processed in most of the volume matching scripts, such that only data for rainy site overpasses are processed. Completed matchup data files and their identifying metadata are cataloged in the database in the ‘geo\_match\_product’ table. In general, the queries in the scripts will exclude those rainy site overpass events where the corresponding matchup output file is already tabulated in this table.

The volume matching scripts are the connection between the ‘gpmgv’ database, which contains information on site rain events and the satellite and ground radar data files associated with these events, and the IDL procedures that perform the volume matching of the satellite and ground radar data. The scripts communicate the information from the database to the IDL matchup procedures in the form of a ‘control file’, a text file that lists the satellite and associated ground radar data files to be paired, along with metadata about the events (satellite ID, product type, product algorithm version, scan type, and orbit number; satellite overflight time; ground radar site ID, latitude, longitude, and elevation, etc.). By convention, the control files are for rainy overpass events occurring on a single calendar date (UTC), and are specific to the type of satellite data product type to be processed. The daily control files constitute the “what” part of the input to the IDL volume matching procedures. The “how” and “where” parts are contained in a separate IDL ‘batch’ command file that configures the variable parameters that control the matchup algorithm and execute the IDL volume matching procedure specific to the satellite data type to be matched.

# Script Summaries – Baseline Matchups

The following tables describe the function and primary inputs and outputs of each volume matching script used in routine GPM Validation Network operations.

## do\_DPR\_GeoMatch.sh

In practice, do\_DPR\_GeoMatch.sh, which only processes one scan type for one DPR product type at a time, has been superseded by the more efficient matchup procedures provided by the scripts do\_ GR\_HS\_MS\_NS\_GeoMatch.sh and do\_DPR2GR\_GeoMatch.sh. Refer to Sections 2.4 and 2.5, below, for details.

|  |  |
| --- | --- |
| Main Script Name | do\_DPR\_GeoMatch.sh |
| Satellite Products Matched | 2ADPR, 2AKa, or 2AKu. Only one product type per run, as specified in input options to script. |
| Scan Type(s) Matched | HS,MS, or NS (2ADPR); HS or MS (2AKa); NS only (2AKu). Only one scan type per run, as specified in input option to script. |
| Control Files | DPR\_files\_sites4geoMatch.tag.txt, where ‘tag’ is a combination of product type, scan type, product version, and YYMMDD (the date of the matchup events), e.g., 2AKu.NS.V04A.161025 |
| Child Script Called | do\_DPR\_geo\_matchup4date.sh, processes one date (one control file) at a time. Calls IDL to run the batch file commands, and tallies its success or failure in the database. |
| IDL Batch File Called | polar2dpr.bat, sets up matchup parameters and executes the polar2dpr procedure |
| Output matchup files | **GRtoDPR.rad.yymmdd.oo.vv.pp.ss.n\_n.uu.nc.gz**, where:  **GRtoDPR**: Literal value, is the same regardless of whether the 2A product subtype is DPR, Ku, or Ka.  **rad**: ID of ground radar, e.g. KAMX  **yymmdd**: date of event  **oo**: GPM orbit number  **vv**: GPM product version, e.g., V04A  **pp**: 2A product subtype, e.g., Ku  **ss**: DPR scan type, e.g., NS  **n\_n**: Matchup file version, e.g. 1\_21 for version 1.21  **uu**: *Optional* user-supplied addition to the filename  **nc**: Literal value indicating a netCDF file type  **gz**: Literal value, present if the file is gzip compressed |
| Log Files | do\_DPR\_GeoMatch.yymmdd.log (main script log, where yymmd here is the current calendar date when the script was started)  do\_DPR\_geo\_matchup4date.YYMMDD.log (child script log, where YYMMDD here is the date of the matchup events configured in the file) |

DESCRIPTION

-----------

Wrapper to do DPR-GR NetCDF geometric matchups for DPR 2A-[DPR|Ka|Ku] files

already received and cataloged, for cases meeting predefined criteria. This

script drives volume matches between GPM DPR and ground radar (GR) data for

a single user-specified DPR scan type to produce the baseline "GRtoDPR"

matchup netCDF files. This script queries the 'gpmgv' database to find rainy

site overpass events between a specified start and end date and assembles a

series of date-specific control files to run matchups for those dates. For

each date, calls the child script do\_DPR\_geo\_matchup4date.sh, which in turn

invokes IDL to generate the GRtoDPR volume match netCDF files for that day's

rainy overpass events as listed in the daily control file. Ancillary output

from the script is a series of 'control files', one per day in the range of

dates to be processed, listing DPR and ground radar data files to be processed

for rainy site overpass events for that calendar date, as well as metadata

parameters related to the DPR and GR data and the site overpass events.

Volume matches are done for only one DPR 2A file type (2A-DPR, 2A-Ka, or

2A-Ku) at a time, and for only one scan type in that 2A data type. The

2A-DPR contains 3 scan types (HS, MS, NS), the 2A-Ka contains 2 (HS, MS),

and the 2A-Ku only contains the NS scan type.

The script has logic to compute the start and end dates over which to attempt

volume match runs. The end date is the current calendar day, and the start

date is 30 day prior to the current date. These computed values exist to

support routine (cron-scheduled) runs of the script. The computed values

are overridden in practice by specifying override values for the variables

'startDate' and 'endDate' in the main script itself, and these values must be

updated each time the script is to be (re)run manually.

Only those site overpasses within the user-specified date range which are

identified as 'rainy' will be configured in the daily control files to be run.

Event criteria are as defined in the table "rainy100inside100" in the "gpmgv"

database, whose contents are updated by an SQL query command file run in this

script as a default option. Event definition includes cases where the DPR

indicates "rain certain" at 100 or more gridpoints within 100 km of the radar

within the 4km gridded 2A-DPR product. See the SQL command file

${BIN\_DIR}/'rainCases100kmAddNewEvents.sql'.

SYNOPSIS

--------

do\_DPR\_GeoMatch.sh [OPTION]...

OPTIONS/ARGUMENTS:

-----------------

-i INSTRUMENT\_ID Override default INSTRUMENT\_ID (DPR) to the specified

INSTRUMENT\_ID (Ka or Ku). This determines which type of

2A product will be processed in the matchups. STRING type.

-v PPS\_VERSION Override default PPS\_VERSION to the specified PPS\_VERSION.

This determines which version of the 2A product will be

processed in the matchups. STRING type.

-p PARAMETER\_SET Override default PARAMETER\_SET to the specified PARAMETER\_SET.

This tracks which “version” of polar2dpr.bat file was used in

processing matchups, when changes are made to the batch file.

This value does not actually control anything, but it gets

written to the geo\_match\_product table in the gpmgv database

as a descriptive attribute. It's up to the user to keep

track of its use and meaning in relation to the configured

parameters in the polar2dpr.bat file. INTEGER type.

-w SWATH Override the default DPR scan type (NS) to another

scan type (either MS or HS), as applicable to the current

INSTRUMENT\_ID. See the COMBO variable in the script.

-m GEO\_MATCH\_VERSION Override default GEO\_MATCH\_VERSION to the specified

GEO\_MATCH\_VERSION. This only changes if the IDL code that

produces the output netCDF file now produces a new or

different version of the matchup file. If the value of

GEO\_MATCH\_VERSION is not the same as the version encoded in

the output netCDF filename an error is noted. FLOAT type.

-k If specified, skip the step of updating the rain events in

the rainy100inside100 table. Takes no argument value.

-f If specified, then instruct matchup programs to create

and overwrite any existing matchups for a date even if

the database says they have been run already (see NOTE).

Takes no argument value.

NOTE: When running dates that might have already had DPR-GR matchup sets

run, the called script will skip these dates, as the 'appstatus' table

will say that the date has already been done. Delete the entries

from this table where app\_id='geo\_match\_IISS', where II is the value

of $INSTRUMENT and SS is the value of $SWATH, either for the date(s)

to be run, or for all dates. EXCEPTION: If script is called with the

-f option (e.g., "do\_DPR\_GeoMatch.sh -f"), then the status of prior

runs for the set of dates configured in the script will be ignored and

the matchups will be re-run, possibly overwriting the existing files.

EXAMPLES

--------

1) Run the script with all defaults. Will run 2ADPR, version V04A, for NS scan type; will execute the SQL command to update the rainy100inside100 database table, and will skip any dates that have already been run. The range of dates that will be attempted is defined in the script by the startDate and endDate variables, which must be edited to reflect the desired data range. From the command line, enter:

**do\_DPR\_GeoMatch.sh**

2) Run the script to do matchups of the 2AKa HS scan type for the version V03D products, skip the SQL command to update the rainy100inside100 database table, and overwrite matchups for any dates that have already been run. The range of dates that will be attempted is defined in the script by the startDate and endDate variables, which must be edited to reflect the desired data range. From the command line, enter:

**do\_DPR\_GeoMatch.sh –i Ka –w HS –v V03D –k -f**

3) Clean out the entries in the appstatus table to allow a re-run of the 2A-DPR matchups for the NS scan type without having to do the –f option to force the run. From the command line, enter:

**[morris@ds1-gpmgv ~]$ psql gpmgv**

**psql (8.4.20)**

**Type "help" for help.**

to begin a psql session. At the psql prompt (gpmgv=>), enter and run the command:

**gpmgv=> delete from appstatus where app\_id = 'geo\_match\_DPRNS';**

DELETE 593

to delete the entire desired set of entries (593 rows were deleted in this example, as shown by the status line output by psql).

To delete the appstatus entries to re-run a specific range of dates (for example, matching the range of dates defined by startDate 20160201 and endDate 20160228 in the script), enter and run the command:

**gpmgv=> delete from appstatus where datestamp between ‘160201’ and ‘160228’ and app\_id = 'geo\_match\_DPRNS';**

Enter \q at the psql prompt to exit the ‘psql’ session and return to the shell prompt.

**gpmgv=> \q**

**[morris@ds1-gpmgv ~]$**

## do\_DPRGMI\_GeoMatch.sh

By design, the IDL procedure called by this script performs volume matches of the ground radar to all scan types (MS and NS) and instruments (Ka and Ku) contained in the 2BDPRGMI product, and stores all the data subtypes in a single output netCDF file. Thus, for a given 2BDPRGMI version and a given range of dates, it is only necessary to make one run of the do\_DPRGMI\_GeoMatch.sh script.

|  |  |
| --- | --- |
| Main Script Name | do\_DPRGMI\_GeoMatch.sh |
| Satellite Products Matched | 2BDPRGMI “Combined” product only. |
| Scan Type(s) Matched | MS and NS. Both scan types are matched up between the GR and the DPRGMI product, and both the Ka and Ku frequencies are matched up for the MS scan type, and all the scan types and frequencies are included in the output netCDF matchup file. |
| Control Files | COMB\_files\_sites4geoMatch.YYMMDD.txt, where YYMMDD is the date of the matchup events configured in the file, e.g., 161025. These control files are written to a product-version-specific subdirectory created in the scratch directory to support their identification and use by other scripts. |
| Child Script Called | do\_DPRGMI\_geo\_matchup4date.sh, processes one date (one control file) at a time. Calls IDL to run the batch file commands, and tallies its success or failure in the database. |
| IDL Batch File Called | polar2dprgmi.bat, sets up matchup parameters and executes the polar2dprgmi procedure |
| Output matchup files | **GRtoDPRGMI.rad.yymmdd.oo.vv.n\_n.uu.nc.gz**, where:  **GRtoDPRGMI**: Literal value.  **rad**: ID of ground radar, e.g. KAMX  **yymmdd**: date of event  **oo**: GPM orbit number  **vv**: GPM product version, e.g., V04A  **n\_n**: Matchup file version, e.g. 1\_21 for version 1.21  **uu**: *Optional* user-supplied addition to the filename  **nc**: Literal value indicating a netCDF file type  **gz**: Literal value, present if the file is gzip compressed |
| Log Files | doDPRGMIGeoMatch4NewRainCases.vv.yymmdd.log (main script log, where vv is as defined above, and yymmd here is the current calendar date when the script was started)  do\_DPRGMI\_geo\_matchup4date.YYMMDD.log (child script log, where YYMMDD here is the date of the matchup events configured in the file) |

DESCRIPTION

-----------

Wrapper to do DPRGMI-GR NetCDF geometric matchups for GPM 2B-DPRGMI files

already received and cataloged, for cases meeting predefined criteria. This

script drives volume matches between DPRGMI and ground radar (GR) data for

both the MS and NS scan type and Ka and Ku instruments to produce the baseline

"GRtoDPRGMI" matchup netCDF files. Queries the 'gpmgv' database to find rainy

site overpass events between a specified start and end date and assembles a

series of date-specific control files to run matchups for those dates. For

each date, calls the child script do\_DPRGMI\_geo\_matchup4date.sh, which in turn

invokes IDL to generate the GRtoDPRGMI volume match netCDF files for that day's

rainy overpass events as listed in the daily control file. Ancillary output

from the script is a series of 'control files', one per day in the range of

dates to process, listing DPRGMI and ground radar data files to be processed

for rainy site overpass events for that calendar date, as well as metadata

parameters related to the DPRGMI and GR data and the site overpass events.

Volume matches are done for MS and NS scan types for Ku data, and MS only for

the Ka data. Data for all scan types and frequencies are contained in the

GRtoDPRGMI netCDF matchup data files. These files are not split out by scan

type and frequency as the 2A-DPR/Ka/Ku matchups are.

The script has logic to compute the start and end dates over which to attempt

volume match runs. The end date is the current calendar day, and the start

date is 30 day prior to the current date. These computed values exist to

support routine (cron-scheduled) runs of the script. The computed values

are overridden in practice by specifying override values for the variables

'startDate' and 'endDate' in the main script itself, and these values must be

updated each time the script is to be (re)run manually.

Only those site overpasses within the user-specified date range which are

identified as 'rainy' will be configured in the daily control files to be run.

Event criteria are as defined in the table "rainy100inside100" in the "gpmgv"

database, whose contents are updated by an SQL query command file run in this

script as a default option. Event definition includes cases where the DPR

indicates "rain certain" at 100 or more gridpoints within 100 km of the radar

within the 4km gridded 2A-DPR product. See the SQL command file

${BIN\_DIR}/'rainCases100kmAddNewEvents.sql'.

SYNOPSIS

--------

do\_DPRGMI\_GeoMatch.sh [OPTION]...

OPTIONS/ARGUMENTS:

-----------------

-v PPS\_VERSION Override default PPS\_VERSION to the specified PPS\_VERSION.

This determines which version of the 2B product will be

processed in the matchups. STRING type.

-p PARAMETER\_SET Override default PARAMETER\_SET to the specified PARAMETER\_SET

This tracks which version of IDL batch file was used in

processing matchups, when changes are made to the batch file.

This value does not actually control anything, but it gets

written to the geo\_match\_product table in the gpmgv database

as a descriptive attribute. It's up to the user to keep track

of its use and meaning. INTEGER type.

-m GEO\_MATCH\_VERSION Override default GEO\_MATCH\_VERSION to the specified

GEO\_MATCH\_VERSION. This only changes if the IDL code that

produces the output netCDF file now produces a new or

different version of the matchup file. If the value of

GEO\_MATCH\_VERSION is not the same as the version encoded in

the output netCDF filename a fatal error occurs. FLOAT type.

-k If specified, skip the step of updating the rain events in

the rainy100inside100 table. Takes no argument value.

Option is disabled, but internally set to always skip the

updates under the assumption that updates were already

done by a prerequisite run of do\_GR\_HS\_MS\_NS\_GeoMatch.sh.

-f If specified, then instruct matchup programs to create

and overwrite any existing matchups for a date even if

the database says they have been run already (see NOTE).

Takes no argument value.

-r If specified, then configure to run matchups to RHI

ground radar scans instead of the default PPI scans.

This capability is not yet supported, so this option

has no effect on the output. Script will gracefully

exit if it is specified.

-c If specified, then configure to just create the control

files for the date range and defer running the volume

matches in IDL. This helps support the mode where two

different machines might work simultaneously, running

volume matches for different dates. See the related

script: do\_DPRGMI\_GeoMatch\_from\_ControlFiles.sh

NOTE: When running dates that might have already had DPRGMI-GR matchup sets

run, the called script will skip these dates, as the 'appstatus' table

will say that the date has already been done. Delete the entries

from this table where app\_id='geo\_match\_COMB', either for the date(s)

to be run, or for all dates. EXCEPTION: If script is called with the

-f option (e.g., "do\_DPRGMI\_GeoMatch.sh -f"), then the status of prior

runs for the set of dates configured in the script will be ignored and

the matchups will be re-run, possibly overwriting the existing files.

EXAMPLES

--------

1) Run the script with all defaults. Will run 2BDPRGMI, version V04A. Will execute the SQL command to update the rainy100inside100 database table, and will skip any dates that have already been run. The range of dates that will be attempted is defined in the script by the startDate and endDate variables, which must be edited to reflect the desired data range.

**do\_DPRGMI\_GeoMatch.sh**

2) Run the script to do matchups of the version V03D products, skip the SQL command to update the rainy100inside100 database table, and overwrite matchups for any dates that have already been run. The range of dates that will be attempted is defined in the script by the startDate and endDate variables, which must be edited to reflect the desired data range.

**do\_DPRGMI\_GeoMatch.sh -v V03D -k -f**

3) Run the script with all defaults, but including the –c option, which will instruct the script to produce the daily control files for the range of dates with rain events, but will skip the call to the child script do\_DPRGMI\_geo\_matchup4date.sh, which actually runs the matchups. The range of dates that will be attempted is defined in the script by the startDate and endDate variables, which must be edited to reflect the desired data range.

**do\_DPRGMI\_GeoMatch.sh -c**

### do\_DPRGMI\_GeoMatch\_from\_ControlFiles.sh

There is a variation on the use of the do\_DPRGMI\_GeoMatch.sh script that divides its functions up into two scripts: the first produces a series of daily control files but does not call the child script to do matchups, and a second script that identifies a series of DPRGMI matchup files produced by the first and calls the child script to run the matchups defined in the daily control files. The first function is accomplished by calling do\_DPRGMI\_GeoMatch.sh (Section 2.2) with the ‘-c’ option specified on the command line. The second function is accomplished by running this script, do\_DPRGMI\_GeoMatch\_from\_ControlFiles.sh, configured to look in the product-version-specific subdirectory to locate the version and dates of control files previously created by the do\_DPRGMI\_GeoMatch.sh script, and then calls the child script do\_DPRGMI\_geo\_matchup4date.sh to run the GR-DPRGMI matchups in the same manner as do\_DPRGMI\_GeoMatch.sh.

The main difference is that, unlike do\_DPRGMI\_GeoMatch.sh, the internal function catalog\_to\_db in this script does not access the PostGRESQL database ‘gpmgv’, it only prepares the output netCDF file cataloging metadata intended for the database and writes it to a text file for later, manual loading to the database. In that way, this script can successfully run and produce matchups on a separate machine that does not have access to the database. *For this mode of operation to succeed without errors will require the addition of an option to the shared child script* do\_DPRGMI\_geo\_matchup4date.sh *to instruct it to also not interact with the database.*

|  |  |
| --- | --- |
| Main Script Name | do\_DPRGMI\_GeoMatch\_from\_ControlFiles.sh |
| Satellite Products Matched | 2BDPRGMI “Combined” product only. |
| Scan Type(s) Matched | MS and NS. Both scan types are matched up between the GR and the DPRGMI product, and both the Ka and Ku frequencies are matched up for the MS scan type, and all the scan types and frequencies are included in the output netCDF matchup file. |
| Control Files | COMB\_files\_sites4geoMatch.YYMMDD.txt, where YYMMDD is the date of the matchup events configured in the file, e.g., 161025. These control files are located in a product-version-specific subdirectory in the scratch directory to support their identification and use by this script. |
| Child Script Called | do\_DPRGMI\_geo\_matchup4date.sh, processes one date (one control file) at a time. Calls IDL to run the batch file commands, and tallies its success or failure in the database. |
| IDL Batch File Called | polar2dprgmi.bat, sets up matchup parameters and executes the polar2dprgmi procedure |
| Output matchup files | **GRtoDPRGMI.rad.yymmdd.oo.vv.n\_n.uu.nc.gz**, where:  **GRtoDPRGMI**: Literal value.  **rad**: ID of ground radar, e.g. KAMX  **yymmdd**: date of event  **oo**: GPM orbit number  **vv**: GPM product version, e.g., V04A  **n\_n**: Matchup file version, e.g. 1\_21 for version 1.21  **uu**: *Optional* user-supplied addition to the filename  **nc**: Literal value indicating a netCDF file type  **gz**: Literal value, present if the file is gzip compressed |
| Log Files | doDPRGMIGeoMatch4NewRainCases.vv.yymmdd.log (main script log, where vv is as defined above, and yymmd here is the current calendar date when the script was started).  do\_DPRGMI\_geo\_matchup4date.YYMMDD.log (child script log, where YYMMDD here is the date of the matchup events configured in the file) |

DESCRIPTION

-----------

When the script do\_DPRGMI\_GeoMatch.sh is run with the '-c' option, it will

query the gpmgv database for dates/times of rainy DPR or GR events and write

daily control files used as input to do the DPRGMI-GR geometry matching for

events with data, but will stop short of doing the actual matchups. This

script will look at a series of these control files for a range of dates

internally defined by 'dateStart' and 'dateEnd', locate the matching control

files for the $SAT\_ID/$ALGORITHM/$PPS\_VERSION and date combination, and

for each such control file, will call do\_DPRGMI\_geo\_matchup4date.sh which

then runs IDL to do the GR/DPRGMI matchups. This mode allows more than one

IDL matchup process to run at a time, on the same host or a different host,

against a set of non-overlapping dates. This is useful to speed up

reprocessing a complete set of matchups for a new version of GPM or ground

radar data.

Completed geometry match files are cataloged in a delimited text file that

can be loaded into the 'gpmgv' database table 'geo\_match\_product', present

on ds1-gpmgv. This file is written to in append mode in the function

catalog\_to\_db() contained in this script file, so it always grows until it is

manually deleted, hopefully following a successful manual loading to the

database. Unlike in other scripts, catalog\_to\_db() only prepares the data to

load to the database, and does not do the actual loading step. This allows

this script to be run on a 2nd machine that does not have database access.

See SQL\_BIN2 in the catalog\_to\_db() documentation.

SYNOPSIS

--------

do\_DPRGMI\_GeoMatch\_from\_ControlFiles.sh [OPTION]...

OPTIONS/ARGUMENTS:

-----------------

-v PPS\_VERSION Override default PPS\_VERSION to the specified PPS\_VERSION.

This determines which version of the 2B product will be

processed in the matchups. STRING type.

-p PARAMETER\_SET Override default PARAMETER\_SET to the specified PARAMETER\_SET

This tracks which version of IDL batch file was used in

processing matchups, when changes are made to the batch file.

This value does not actually control anything, but it gets

written to the geo\_match\_product table in the gpmgv database

as a descriptive attribute. It is up to the user to keep

track of its use and meaning. INTEGER type.

-m GEO\_MATCH\_VERSION Override default GEO\_MATCH\_VERSION to the specified

GEO\_MATCH\_VERSION. This only changes if the IDL code that

produces the output netCDF file now produces a new or

different version of the matchup file. If the value of

GEO\_MATCH\_VERSION is not the same as the version encoded in

the output netCDF filename a fatal error occurs. FLOAT type.

-f If specified, then instruct matchup programs to create

and overwrite any existing matchups for a date even if

the database says they have been run already (see NOTE).

Takes no argument value.

-r If specified, then configure to run matchups to RHI

ground radar scans instead of the default PPI scans.

This capability is not yet supported, so this option

has no effect on the output. Script will gracefully

exit if it is specified.

## do\_GMI\_GeoMatch.sh

|  |  |
| --- | --- |
| Main Script Name | do\_GMI\_GeoMatch.sh |
| Satellite Products Matched | 2AGPROF, with 1CRXCAL where available. Data from GPM, TRMM, or any constellation satellite may be used, despite the fact that the scripts, control files, and IDL programs are named according to the GPM GMI instrument. In practice, only GPM GMI and TRMM TMI have matching ground radar data. Satellite and instrument IDs are specified as options to script; defaults are GPM and GMI. |
| Scan Type(s) Matched | Not applicable, only the conical scan is made, and footprints are spatially matched for all variables. |
| Control Files | GMI\_files\_sites4geoMatch.YYMMDD.txt, where YYMMDD is the date of the matchup events configured in the file, e.g., 161025 |
| Child Script Called | do\_GMI\_geo\_matchup4date.sh, processes one date (one control file) at a time. Calls IDL to run the batch file commands, and tallies its success or failure in the database. |
| IDL Batch File Called | polar2gmi.bat, sets up matchup parameters and executes the polar2gmi procedure |
| Output matchup files | **GRtoGPROF.sat.mi.rad.yymmdd.oo.vv.n\_n.uu.nc.gz**, where:  **GRtoGPROF**: Literal value.  **sat**: ID of satellite, e.g., GPM, METOPA  **mi**: ID of microwave instrument, e.g., GMI, MHS, SSMIS  **rad**: ID of ground radar, e.g. KAMX  **yymmdd**: date of event  **oo**: GPM orbit number  **vv**: GPM product version, e.g., V04A  **n\_n**: Matchup file version, e.g. 1\_21 for version 1.21  **uu**: *Optional* user-supplied addition to the filename  **nc**: Literal value indicating a netCDF file type  **gz**: Literal value, present if the file is gzip compressed |
| Log Files | do\_satmi\_GeoMatch.vv.yymmdd.log (main script log, where sat, mi, and vv are as defined above, and yymmd here is the current calendar date when the script was started). sat and mi are merged in the file name, e.g.: do\_GPMGMI\_GeoMatch.V03B.160414.log  do\_GMI\_geo\_matchup4date.YYMMDD.log (child script log, where YYMMDD here is the date of the matchup events configured in the file). Currently always labeled as GMI, regardless of the satellite whose data are being processed. |

DESCRIPTION

-----------

Wrapper to do GPROF-GR NetCDF geometric matchups for 2A-GPROF and 1C-R-XCAL

files in the GV data system, for cases meeting predefined criteria. This

script drives volume matches between GPROF and ground radar (GR) data to

produce the baseline "GRtoGPROF" matchup netCDF files. Queries the 'gpmgv'

database to find rainy site overpass events between a specified start and end

date and assembles a series of date-specific control files to run matchups for

those dates. By default, the GPM GMI 2A-GPROF and 1C-R-XCAL data are used in

the volume matching, but this script and its called procedures have the

capability to do matchups against these product types for any constellation

satellite, since the file formats are the same. In practice this is not

feasible, since the matching ground radar data typically are not available

except for the GPM and TRMM satellites.

For each date, calls the child script do\_GMI\_geo\_matchup4date.sh, which

invokes IDL to generate the GRtoGPROF volume match netCDF files for that day's

rainy overpass events as listed in the daily control file. Ancillary output

from the script is a series of 'control files', one per day in the range of

dates to be processed, listing 2A-GPROF, 1C-R-XCAL (if available), and ground

radar data files to be processed for rainy site overpass events for that

calendar date, as well as metadata parameters related to the GPROF and GR

data and the site overpass events.

Volume matches are done using the 2A-GPROF data products, and will also

include equivalent blackbody temperature data from the matching (i.e., same

satellite, orbit, orbit subset, and version) 1C-R-XCAL product, if available.

Completed geometry match files are cataloged in the 'gpmgv' database table

'geo\_match\_product'.

The script has logic to compute the start and end dates over which to attempt

volume match runs. The end date is the current calendar day, and the start

date is 30 day prior to the current date. These computed values exist to

support routine (cron-scheduled) runs of the script. The computed values

are overridden in practice by specifying override values for the variables

'startDate' and 'endDate' in the main script itself, and these values must be

updated each time the script is to be (re)run manually.

Only those site overpasses within the user-specified date range which are

identified as 'rainy' will be configured in the daily control files to be run.

Event criteria are as defined in the table "rainy100inside100" in the "gpmgv"

database, whose contents are updated by an SQL query command file run in this

script as a default option. Event definition includes cases where the DPR

indicates "rain certain" at 100 or more gridpoints within 100 km of the radar

within the 4km gridded 2A-DPR product. See the SQL command file

${BIN\_DIR}/'rainCases100kmAddNewEvents.sql'.

SYNOPSIS

--------

do\_GMI\_GeoMatch.sh [OPTION]...

OPTIONS/ARGUMENTS:

-----------------

-s SAT\_ID Override default SAT\_ID (GPM) to the specified SAT\_ID.

This, along with INSTRUMENT\_ID, determines which type

of 2AGPROF product will be processed in the matchups.

STRING type.

-i INSTRUMENT\_ID Override default INSTRUMENT\_ID (GMI) to the specified

INSTRUMENT\_ID that pertains to the specified SAT\_ID.

This determines which type of 2AGPROF product will be

processed in the matchups. STRING type.

-v PPS\_VERSION Override default PPS\_VERSION to the specified PPS\_VERSION.

This determines which version of the 2A/1C products will be

processed in the matchups. STRING type.

-p PARAMETER\_SET Override default PARAMETER\_SET to the specified PARAMETER\_SET

This tracks which version of IDL batch file was used in

processing matchups, when changes are made to the batch file.

This value does not actually control anything, but it gets

written to the geo\_match\_product table in the gpmgv database

as a descriptive attribute. It's up to the user to keep track

of its use and meaning. INTEGER type.

-m GEO\_MATCH\_VERSION Override default GEO\_MATCH\_VERSION to the specified

GEO\_MATCH\_VERSION. This only changes if the IDL code that

produces the output netCDF file now produces a new or

different version of the matchup file. If the value of

GEO\_MATCH\_VERSION is not the same as the version encoded in

the output netCDF filename a fatal error occurs. FLOAT type.

-k If specified, skip the step of updating the rain events in

the rainy100inside100 table. Takes no argument value.

-f If specified, then instruct matchup programs to create

and overwrite any existing matchups for a date even if

the database says they have been run already (see NOTE).

Takes no argument value.

NOTE: When running dates that might have already had GPROF-GR matchup sets

run, the called script will skip these dates, as the 'appstatus' table

will say that the date has already been done. Delete the entries

from this table where app\_id='geo\_match\_gmi', either for the date(s)

to be run, or for all dates. EXCEPTION: If script is called with the

-f option (e.g., "do\_GMI\_GeoMatch.sh -f"), then the status of prior

runs for the set of dates configured in the script will be ignored and

the matchups will be re-run, possibly overwriting the existing files.

## do\_GR\_HS\_MS\_NS\_GeoMatch.sh

To run matchups between DPR and GR for all available DPR 2A products (2ADPR, 2AKa, 2AKu) and scan types (HS, MS, NS) would require 6 separate runs of the do\_DPR\_GeoMatch.sh script, since only one product and scan type is handled at a time by this legacy script and IDL procedure. The GR data files would need to be copied, uncompressed, open, and read each time the script is run, and this is a slow process now that the GR data files are so large. To alleviate some of this processing time, do\_ GR\_HS\_MS\_NS\_GeoMatch.sh and its called IDL procedure reads the 2ADPR product which contains all 3 DPR scan types, reads the GR data once and volume matches the GR fields to each of the 3 DPR scan types, and saves the volume-matched GR data only, for all 3 scan types, in an intermediate site/version/orbit-specific netCDF file. The volume-matched GR data in these intermediate files can then be merged with the product- and scan-specific DPR data, volume-matched to the GR data in the follow-on script do\_DPR2GR\_GeoMatch.sh, run separately at a later time to produce the baseline GRtoDPR matchup netCDF files. All the spatial information needed to volume-match the DPR data for each scan type to the GR data is contained in the intermediate netCDF files, alleviating the need to re-read the GR data files in the merging process. The assumption here is that the HS and MS (NS) scan locations are the same between the 2ADPR and 2AKa (2AKu) products for a given orbit and version.

|  |  |
| --- | --- |
| Main Script Name | do\_ GR\_HS\_MS\_NS\_GeoMatch.sh |
| Satellite Products Matched | 2ADPR only. Performs GR-only volume matching to DPR footprint locations for all of the 3 DPR scan types. A run of this script should be followed up with a sequence of runs of do\_DPR2GR\_GeoMatch.sh for the same range of dates. Product- and scan-type-specific 2ADPR, 2AKa, or 2AKu volume matching to the GR data is performed by do\_DPR2GR\_GeoMatch.sh and merged with previously volume-matched GR data output by this script to produce the baseline GRtoDPR netCDF matchup files containing both volume-matched DPR and GR data. |
| Scan Type(s) Matched | HS,MS, and NS. GR data are matched to all three scan types per run. |
| Control Files | DPR\_files\_sites4geoMatch.tag.txt, where ‘tag’ is a combination of product type, scan type (hard-coded as ‘All3’), product version, and YYMMDD (the date of the matchup events), e.g., 2AKu.All3.V04A.161025 |
| Child Script Called | do\_GR\_HS\_MS\_NS\_geo\_matchup4date.sh, processes one date (one control file) at a time. Calls IDL to run the batch file commands, and tallies its success or failure in the database. |
| IDL Batch File Called | polar2dpr\_hs\_ms\_ns.bat, sets up matchup parameters and executes the polar2dpr\_hs\_ms\_ns procedure |
| Output matchup files | **GRtoDPR\_HS\_MS\_NS.rad.yymmdd.oo.vv.n\_n.uu.nc.gz**, where:  **GRtoDPR\_HS\_MS\_NS**: Literal value.  **rad**: ID of ground radar, e.g. KAMX  **yymmdd**: date of event  **oo**: GPM orbit number  **vv**: GPM product version, e.g., V04A  **n\_n**: Matchup file version, e.g. 1\_0 for version 1.0  **uu**: *Optional* user-supplied addition to the filename  **nc**: Literal value indicating a netCDF file type  **gz**: Literal value, present if the file is gzip compressed |
| Log Files | do\_GR\_HS\_MS\_NS\_GeoMatch.yymmdd.log (main script log, where yymmd here is the current calendar date when the script was started)  do\_GR\_HS\_MS\_NS\_geo\_matchup4date.YYMMDD.log (child script log, where YYMMDD here is the date of the matchup events configured in the file) |

DESCRIPTION

-----------

Wrapper to do GR geometric matchups to HS, MS, and NS scans in 2A-DPR files

already received and cataloged, for cases meeting predefined criteria. This

script drives volume matches of GR data to DPR footprint locations for all

of the DPR scan types. The DPR-matched GR data are written to event-specific

"GRtoDPR\_HS\_MS\_NS" netCDF files that hold the GR matchups for all three types

of DPR scans in a single file. These matchup files are specific to GR site,

GPM orbit, PPS processing version of the input 2A-DPR data, and matchup file

definition version (currently 1.0). Output of this script and its called

processes consists of a set of intermediate netCDF files containing ground

radar data only, volume-matched to all three (HS, MS, NS) DPR scan types.

Ancillary output is a series of 'control files', one per day in the range of

dates to be processed, listing DPR and ground radar data files to be processed

for rainy site overpass events for that calendar date, as well as metadata

parameters related to the DPR and GR data and the site overpass events.

A run of this script should be followed up with one or more runs of

do\_DPR2GR\_GeoMatch.sh for the same range of dates to complete the creation of

baseline GRtoDPR netCDF files containing both volume-matched DPR and GR data.

The script has logic to compute the start and end dates over which to attempt

volume match runs. The end date is the current calendar day, and the start

date is 30 day prior to the current date. These computed values exist to

support routine (cron-scheduled) runs of the script. The computed values

are overridden in practice by specifying override values for the variables

'startDate' and 'endDate' in the main script itself, and these values must be

updated each time the script is to be (re)run manually.

Only those site overpasses within the user-specified date range which are

identified as 'rainy' will be configured in the daily control files to be run.

Event criteria are as defined in the table "rainy100inside100" in the "gpmgv"

database, whose contents are updated by an SQL query command file run in this

script as a default option. Event definition includes cases where the DPR

indicates "rain certain" at 100 or more gridpoints within 100 km of the radar

within the 4km gridded 2A-DPR product. See the SQL command file

${BIN\_DIR}/'rainCases100kmAddNewEvents.sql'.

SYNOPSIS

--------

do\_GR\_HS\_MS\_NS\_GeoMatch.sh [OPTION]...

OPTIONS/ARGUMENTS:

-----------------

-v PPS\_VERSION Override default PPS\_VERSION to the specified PPS\_VERSION.

This determines which version of the 2ADPR product will be

processed in the matchups. STRING type.

-p PARAMETER\_SET Override default PARAMETER\_SET to the specified PARAMETER\_SET

This tracks which version of IDL batch file was used in

processing matchups, when changes are made to the batch file.

This value does not actually control anything, but it gets

written to the geo\_match\_product table in the gpmgv database

as a descriptive attribute. It's up to the user to keep track

of its use and meaning. INTEGER type.

-m GEO\_MATCH\_VERSION Override default GEO\_MATCH\_VERSION to the specified

GEO\_MATCH\_VERSION. This only changes if the IDL code that

produces the output netCDF file now produces a new or

different version of the matchup file. If the value of

GEO\_MATCH\_VERSION is not the same as the version encoded in

the output netCDF filename a fatal error occurs. FLOAT type.

-k If specified, skip the step of updating the rain events in

the rainy100inside100 table. Takes no argument value.

-f If specified, then instruct matchup programs to create

and overwrite any existing matchups for a date even if

the database says they have been run already (see NOTE).

Takes no argument value.

NOTE: When running dates that have already had GR->DPR matchup sets run,

the child script do\_GR\_HS\_MS\_NS\_geo\_matchup4date.sh will skip

processing for these dates, as its query of the 'appstatus' table

will say that the date has already been done. Delete the entries

from this table where app\_id='geo\_match\_GRx3', either for the date(s)

to be run, or for all dates. EXCEPTION: If script is called with the

-f option (e.g., "do\_GR\_HS\_MS\_NS\_GeoMatch.sh -f"), then the status of

prior runs for the set of dates configured in the script is ignored and

the matchups will be re-run, possibly overwriting the existing files.

## do\_DPR2GR\_GeoMatch.sh

This script is a required follow-on to the do\_ GR\_HS\_MS\_NS\_GeoMatch.sh script described in the preceding section. Once do\_ GR\_HS\_MS\_NS\_GeoMatch.sh has completed for a given range of dates, do\_DPR2GR\_GeoMatch.sh is run for each desired 2A product subtype (DPR, Ka, or Ku) to compete the final-output GRtoDPR matchup files for every scan type present within that product type. This requires editing the startDate and endDate values in the do\_DPR2GR\_GeoMatch.sh script to match the date range used in the run of do\_ GR\_HS\_MS\_NS\_GeoMatch.sh.

|  |  |
| --- | --- |
| Main Script Name | do\_DPR2GR\_GeoMatch.sh |
| Satellite Products Matched | 2ADPR (default), 2AKa, or 2AKu. Only one 2A product subtype is processed per run, either for the default ‘DPR’ subtype or as specified in ‘-i’ input option to script, e.g.:  do\_DPR2GR\_GeoMatch.sh –i Ku |
| Scan Type(s) Matched | HS,MS, and NS (2ADPR); HS and MS (2AKa); NS (2AKu). Called IDL procedure automatically performs a separate matchup for each scan type that applies to the selected 2A satellite product subtype, resulting in separate output matchup files with differing ‘ss’ filename components. |
| Control Files | DPR\_files\_sites4geoMatch.tag.txt, where ‘tag’ is a combination of product type, scan type (hard-coded as ‘All’), product version, and YYMMDD (the date of the matchup events), e.g., 2AKu.All.V04A.161025 |
| Child Script Called | do\_DPR2GR\_geo\_matchup4date.sh, processes one date (one control file) at a time. Calls IDL to run the batch file commands, and tallies its success or failure in the database. |
| IDL Batch File Called | dpr2gr\_prematch.bat, sets up matchup parameters and executes the dpr2gr\_prematch procedure |
| Output matchup files | **GRtoDPR.rad.yymmdd.oo.vv.pp.ss.n\_n.uu.nc.gz**, where filename components are the same as defined in Section 2.1, for do\_DPR\_GeoMatch.sh. |
| Log Files | do\_DPR2GR\_GeoMatch.yymmdd.log (main script log, where yymmd here is the current calendar date when the script was started)  do\_DPR2GR\_geo\_matchup4date.YYMMDD.log (child script log, where YYMMDD here is the date of the matchup events configured in the file) |

DESCRIPTION

-----------

Wrapper to do DPR-GV NetCDF geometric matchups for DPR 2A-[DPR|Ka|Ku] files

already received and cataloged, for cases meeting predefined criteria. This

script drives volume matches of DPR data to GR data that has already been

volume matched to DPR for all of the DPR scan types. The previously matched

GR data are contained in "GRtoDPR\_HS\_MS\_NS" netCDF files, which are read and

merged with the volume matched DPR data to produce the baseline "GRtoDPR"

matchup netCDF files. This script queries the 'gpmgv' database to find rainy

site overpass events between a specified start and end date and assembles a

series of date-specific control files to run matchups for those dates. For

each date, calls the child script do\_DPR2GR\_geo\_matchup4date.sh, which in turn

invokes IDL to generate the DPRtoGR volume match netCDF files for that day's

rainy overpass events as listed in the daily control file. Ancillary output

from the script is a series of 'control files', one per day in the range of

dates to be processed, listing DPR and ground radar data files to be processed

for rainy site overpass events for that calendar date, as well as metadata

parameters related to the DPR and GR data and the site overpass events.

The script has logic to compute the start and end dates over which to attempt

volume match runs. The end date is the current calendar day, and the start

date is 30 day prior to the current date. These computed values exist to

support routine (cron-scheduled) runs of the script. The computed values

are overridden in practice by specifying override values for the variables

'startDate' and 'endDate' in the main script itself, and these values must be

updated each time the script is to be (re)run manually.

Only those site overpasses within the user-specified date range which are

identified as 'rainy' will be configured in the daily control files to be run.

Event criteria are as defined in the table "rainy100inside100" in the "gpmgv"

database, whose contents are updated by an SQL query command file run in this

script as a default option. Event definition includes cases where the DPR

indicates "rain certain" at 100 or more gridpoints within 100 km of the radar

within the 4km gridded 2A-DPR product. See the SQL command file

${BIN\_DIR}/'rainCases100kmAddNewEvents.sql'.

SYNOPSIS

--------

do\_DPR2GR\_GeoMatch.sh [OPTION]...

OPTIONS/ARGUMENTS:

-----------------

-i INSTRUMENT\_ID Override default INSTRUMENT\_ID (DPR) to the specified

INSTRUMENT\_ID (Ka or Ku). This determines which type of

2A product will be processed in the matchups. STRING type.

-v PPS\_VERSION Override default PPS\_VERSION to the specified PPS\_VERSION.

This determines which version of the 2A product will be

processed in the matchups. STRING type.

-p PARAMETER\_SET Override default PARAMETER\_SET to the specified PARAMETER\_SET

This tracks which version of IDL batch file was used in

processing matchups, when changes are made to the batch file.

This value does not actually control anything, but it gets

written to the geo\_match\_product table in the gpmgv database

as a descriptive attribute. It's up to the user to keep track

of its use and meaning. INTEGER type.

-m GEO\_MATCH\_VERSION Override default GEO\_MATCH\_VERSION to the specified

GEO\_MATCH\_VERSION. This only changes if the IDL code that

produces the output netCDF file now produces a new or

different version of the matchup file. If the value of

GEO\_MATCH\_VERSION is not the same as the version encoded in

the output netCDF filename a fatal error occurs. FLOAT type.

-f If specified, then instruct matchup programs to create

and overwrite any existing matchups for a date even if

the database says they have been run already (see NOTE).

Takes no argument value.

NOTE: When running dates that might have already had DPR-GV matchup sets

run, the child script do\_DPR2GR\_geo\_matchup4date.sh will skip

processing for these dates, as a check of the 'appstatus' table

will say that the date has already been done. Delete the entries

from this table where app\_id='geo\_matchIIAll', where II is the value

of $INSTRUMENT and the rest of the characters (geo\_match and All) are

fixed and literal values, for the date(s) to be run, or for all dates.

EXAMPLE SQL: DELETE FROM appstatus WHERE app\_id='geo\_matchKuAll';

EXCEPTION: If script is called with the -f option

(e.g., "do\_DPR2GR\_GeoMatch.sh -f"), then the status of prior runs

for the set of dates configured in the script will be ignored and

the matchups will be re-run, possibly overwriting the existing files.

The SWATH variable has the fixed value "All" in this script, as the

IDL procedure dpr2gr\_prematch.pro automatically runs matchup for all

the scan types present in the 2A product type being processed.

PRECONDITIONS

-------------

Since this script relies on the existence of previously-volume-matched ground

radar data in "GRtoDPR\_HS\_MS\_NS" netCDF files, do\_GR\_HS\_MS\_NS\_GeoMatch.sh

must already have been configured and successfully run over the start and

end date range configured in this script. Neither this script nor its child

script do\_DPR2GR\_geo\_matchup4date.sh check for the pre-existence of these

GRtoDPR\_HS\_MS\_NS netCDF files.

# Script Summaries - Non-routine and legacy matchup scripts

## do\_DPR\_GeoMatch\_NPOL\_MD\_rules.sh

This script is a variation of the script do\_DPR\_GeoMatch.sh described in Section 2.1, but processes GPM DPR matchups only for the NPOL ground radar. It contains additional features to deal with the mobility and RHI scan capabilities of the NPOL radar. Ground radar data files for the NPOL radar are all named, stored in the file system, and cataloged in the database under the site ID "NPOL", regardless of where the radar is installed and operating. Satellite overpass and rain event criteria for the NPOL radar are provided in the PPS CT (coincidence Table) files and cataloged in the ‘gpmgv’ database under location-specific site IDs depending on where the NPOL radar has been installed and operated; for example, "NPOL\_MD" for the Wallops installation or “NPOL\_WA” for the OLYMPEX field campaign.

The database VIEW (a predefined set of queries that acts like a virtual database table) named ‘collate\_npol\_md\_1cuf’ attempts to automatically reconcile the naming difference between the variable, satellite-related site ID and the fixed ID ‘NPOL’ used for the ground radar files/directories and their database entries. This view can fail to associate the correct NPOL ground radar file to an NPOL rain event listed in the database for one of two reasons: either the NPOL radar scan time of the desired GR scan type (PPI or RHI) is outside the time range considered to be “coincident” between the satellite overpass and the GR volume, or some other disconnect between the database entries exists that causes the VIEW to not return a valid row of data or to identify an NPOL volume different from the best one to be matched to the satellite data. In normal NPOL operations there may be multiple volume scans taken in a mix of PPI and RHI mode for a given GPM overflight.

To allow the user to manually reconcile the NPOL ground radar file to be used for the event, the main script pauses as each date’s control file is prepared and brings the control file up in an editor to allow the user to make changes or corrections to the NPOL radar file partial pathname (or the missing-file placeholder value “no\_1CUF\_file”) listed in the control file before proceeding with the volume matching procedures.

A simple argument (-r) to the script configures it to attempt DPR-GR matchups using NPOL volume scans taken in RHI scan mode. NPOL data files taken in RHI mode have a “.rhi” component to the file name to distinguish them from files for volume scans taken in PPI scan mode. At the IDL level a different ‘.bat’ command file and different IDL program are called when processing matchups for RHI scans.

|  |  |
| --- | --- |
| Main Script Name | do\_DPR\_GeoMatch\_NPOL\_MD\_rules.sh |
| Satellite Products Matched | 2ADPR, 2AKa, or 2AKu. Only one product type per run, as specified in input options to script. |
| Scan Type(s) Matched | HS,MS, or NS (2ADPR); HS or MS (2AKa); NS only (2AKu). Only one scan type per run, as specified in input option to script. |
| Control Files | DPR\_files\_sites4geoMatch.tag.txt, where ‘tag’ is a combination of product type, scan type, product version, and YYMMDD (the date of the matchup events), e.g., 2AKu.NS.V04A.161025 |
| Child Script Called | do\_DPR\_geo\_matchup4date.sh, processes one date (one control file) at a time. Calls IDL to run the batch file commands, and tallies its success or failure in the database. |
| IDL Batch File Called | In default (PPI) matchup mode: polar2dpr.bat, sets up matchup parameters and executes the polar2dpr procedure.  In RHI matchup mode: : rhi2dpr.bat, sets up matchup parameters and executes the rhi2dpr procedure. |
| Output matchup files | **GRtoDPR.rad.yymmdd.oo.vv.pp.ss.n\_n.uu.nc.gz**, where:  **GRtoDPR**: Literal value, is the same regardless of whether the 2A product subtype is DPR, Ku, or Ka.  **rad**: ID of ground radar, e.g. KAMX  **yymmdd**: date of event  **oo**: GPM orbit number  **vv**: GPM product version, e.g., V04A  **pp**: 2A product subtype, e.g., Ku  **ss**: DPR scan type, e.g., NS  **n\_n**: Matchup file version, e.g. 1\_21 for version 1.21  **uu**: *Optional* user-supplied addition to the filename  **nc**: Literal value indicating a netCDF file type  **gz**: Literal value, present if the file is gzip compressed |
| Log Files | do\_DPR\_GeoMatch.yymmdd.log (main script log, where yymmd here is the current calendar date when the script was started). Same log file name convention as for do\_DPR\_GeoMatch.sh  do\_DPR\_geo\_matchup4date.YYMMDD.log (child script log, where YYMMDD here is the date of the matchup events configured in the file) |

DESCRIPTION

-----------

Wrapper to do DPR-GR geometric matchups for 2ADPR/2AKa/2AKu and NPOL 1CUF

files already received and cataloged, for cases meeting predefined criteria.

This version of the script uses the 'gpmgv' database VIEW collate\_npol\_md\_1cuf

to find the matching 1CUF ground radar data file for the event. These files

are all cataloged under the site ID "NPOL", while the overpass and rain event

criteria are cataloged under the site ID "NPOL\_MD". The view attempts to

account for the site ID disconnect between the different database tables

involoved in generating the control files, but does not always succeed in

finding the correct matching NPOL 1CUF file for the event. To handle this,

this "special" script brings up each new control file in the editor 'gedit'

and waits while the user makes any required manual changes to the control file

to replace missing or incorrect NPOL 1CUF file placeholders with the correct

matching 1CUF filenames. Once editing is complete and the editor is dismissed

the script will allow the user to choose either to proceed with the matchups

for this control file or skip the matchup step and go to the next date.

Criteria are as defined in the query which is run to update the table

"rainy100inside100" in the "gpmgv" database. Includes cases where the DPR

indicates "rain certain" at 100 or more gridpoints within 100 km of the NPOL

radar within a 4km gridded 2A-Ku product. See the SQL command file

rainCases100kmAddNewEvents.sql'.

SYNOPSIS

--------

do\_DPR\_GeoMatch\_NPOL\_MD\_rules.sh [OPTION]...

OPTIONS/ARGUMENTS:

-----------------

-i INSTRUMENT\_ID Override default INSTRUMENT\_ID (DPR) to the specified

INSTRUMENT\_ID (Ka or Ku). This determines which type of

2A product will be processed in the matchups. STRING type.

-v PPS\_VERSION Override default PPS\_VERSION to the specified PPS\_VERSION.

This determines which version of the 2A product will be

processed in the matchups. STRING type.

-p PARAMETER\_SET Override default PARAMETER\_SET to the specified PARAMETER\_SET

This tracks which version of IDL batch file was used in

processing matchups, when changes are made to the batch file.

This value does not actually control anything, but it gets

written to the geo\_match\_product table in the gpmgv database

as a descriptive attribute. It's up to the user to keep track

of its use and meaning. INTEGER type.

-w SWATH Override the default DPR scan type (NS) to another

scan type (either MS or HS), as applicable to the current

INSTRUMENT\_ID. See the COMBO variable in the script.

-m GEO\_MATCH\_VERSION Override default GEO\_MATCH\_VERSION to the specified

GEO\_MATCH\_VERSION. This only changes if the IDL code that

produces the output netCDF file now produces a new or

different version of the matchup file. If the value of

GEO\_MATCH\_VERSION is not the same as the version encoded in

the output netCDF filename an error is noted. FLOAT type.

-k If specified, skip the step of updating the rain events in

the rainy100inside100 table. Takes no argument value.

-f If specified, then instruct matchup programs to create

uncataloged matchups for rain events on a date, even if

the 'appstatus' table in the database indicates the date's

matchups for the type have been run already (see NOTE).

Takes no argument value.

-r If specified, then query the database for GR files with

RHI scans and instruct child script to call a version of

matchup programs to generate RHI volume matches to the

DPR data, rather than the default PPI matchups.

Takes no argument value.

-n Like -f option, but instructs the queries that build the

control file to include all rain events even if the

database table 'geo\_match\_product' indicates that a

matching output netCDF file for the event already exists

(event-by-event override, versus date-by-date override).

Takes no argument value.

NOTE: When running dates that might have already had DPR-GR matchup sets

run, the called script will skip these dates, as the 'appstatus' table

will say that the date has already been done. Delete the entries

from this table where app\_id='geo\_match\_IISS', where II is the value

of $INSTRUMENT and SS is the value of $SWATH, either for the date(s)

to be run, or for all dates. EXCEPTION: If script is called with the

-f option (e.g., "do\_DPR\_GeoMatch\_NPOL\_MD\_rules.sh -f"), then the status

of prior runs for the dates configured in the script will be ignored and

the matchups will be re-run, possibly overwriting the existing files.

- By default, processes matchups for the site ID 'NPOL\_MD' only.

This cannot be changed unless a different VIEW than

collate\_npol\_md\_1cuf is coded in SQL and defined in the ‘gpmgv’

database.

## do\_DPRtoGMI\_OrbitMatch.sh

This is an incomplete script, still in development. It is intended to produce a listing of 2A-GPROF GMI files for orbit subsets that have matching rain events at overflown ground radar sites as input to the IDL **dpr2gmi\_driver** procedure, which in turn calls the IDL function **dpr2gmi** to do the individual matchups for each orbit subset product (file). The IDL **dpr2gmi** function performs spatial matching between surface rain rate estimates in the GMI GPROF product and the near-surface rain rate estimates in the matching 2A-[DPR/Ka/Ku] and (optionally) the 2B-DPRGMI products along the GPM orbit track, and (optionally) stores all the matched rainrate data in an output netCDF file.

While the ground radar product itself is not involved in the spatial matching process, this script still queries the database for rainy overpass events at the GR sites to assure that there is a minimum coverage of observed precipitation in the DPR swath for a given GPM orbit. Only one of the DPR products (2A-DPR, 2A-Ku, or 2A-Ka) and one of the scan types (HS, MS, or NS) is matched to the GMI rain rate data in a run of **dpr2gmi\_driver**.

For now, **dpr2gmi\_driver** is run directly in IDL, and the necessary configuration parameters are manually input to **dpr2gmi\_driver**. Its output is considered to be an experimental product and is not routinely produced for the Validation Network data set. However, this script can be used to produce a control file

|  |  |
| --- | --- |
| Main Script Name | do\_DPRtoGMI\_OrbitMatch.sh |
| Satellite Products Matched | 2ADPR, 2AKa, or 2AKu. Only one product type per run, as specified in input options to script. Optionally includes the matching 2BDPRGMI “Combined” product in the matchups (identified and handled internally by the IDL code). |
| Scan Type(s) Matched | HS,MS, or NS (2ADPR); HS or MS (2AKa); NS only (2AKu). Only one scan type per run, as specified in input option to script. If HS, then the 2BDPRGMI product is excluded from the matchup by the IDL matchup procedure. |
| Control Files | GMI\_orbit\_files\_sites4geoMatch.YYMMDD.txt, where YYMMDD is the date of the matchup events configured in the file, e.g., 161025 |
| Child Script Called | TBD. |
| IDL Batch File Called | TBD. The IDL procedure to be called within the future batch file is **dpr2gmi\_driver.** |
| Output matchup files | **DPRtoGMI\_NN.yyyymmdd.oo.vv.UU\_pp\_ss\_VV\_DD.dkm.nc.gz**, where:  **DPRtoGMI\_NN**: Literal value.  **yyyymmdd**: date of event  **oo**: GPM orbit number  **vv**: GMI-specific product version, e.g., V03B  **UU**: Orbit subset label, e.g., CONUS  **pp**: 2A product subtype, e.g., Ku  **ss**: DPR scan type, e.g., NS  **VV**: 2A-[DPR/Ka/Ku]-specific product version, e.g., V04A  **DD.dkm**: Averaging radius of DPR samples around the GMI footprint center point, where **DD.d** is the radial distance in km in the floating point format nn.n, and **km** is a literal value.  **nc**: Literal value indicating a netCDF file type  **gz**: Literal value, present if the file is gzip compressed |
| Log Files | doDPRtoGMI\_OrbitMatch.vv.yymmdd.log (main script log, where vv is as defined above, and yymmd here is the current calendar date when the script was started)  TBD.log (child script log, TBD) |

DESCRIPTION

-----------

**TBD.**

The following sections describe the function and primary inputs and outputs of the legacy (TRMM-specific) volume matching scripts used in the prototype and GPM-era Validation Network operations.

## do\_PR\_GeoMatch.sh

The legacy script do\_PR\_GeoMatch.sh processes matchups between the TRMM PR and ground radars using multiple PR products in TRMM HDF4 files. Unlike the all-in-one GPM DPR products, TRMM PR variables of interest are distributed among multiple product types, e.g.: 1C21 (measured Z, land/ocean flag, surface clutter blanking), 2A23 (bright band/QC fields), and 2A25 (corrected Z, rain type, 3-D and surface rain rate, BB height). If the matching 2B31 file is available, it is processed to include the combined PR/TMI surface rain rate in the matchups. The 1C21 and 2A25 products are mandatory, while matchups are allowed to proceed if the matching 2A23 or 2B31 files are not available.

There is no new file format in the GPM era for the PR data files, although the file naming convention was changed to meet the GPM filename convention beginning on 3/1/2014. There is about a 2-week overlap between files in the old naming convention (located under /data/gpmgv/prsubsets) in ‘flat’ product-specific directories, and those with the new naming convention and PPS-like directory structures (located under /data/gpmgv/orbit\_subset/TRMM).

|  |  |
| --- | --- |
| Main Script Name | do\_PR\_GeoMatch.sh |
| Satellite Products Matched | 1C21, 2A23, 2A25, and 2B31. Only the 1C21 and 2A25 are mandatory. |
| Scan Type(s) Matched | Not applicable, PR does not have multiple scans. |
| Control Files | PR\_files\_sites4geoMatch.yymmdd.txt, where yymmdd is the date of the matchup events. |
| Child Script Called | do\_geo\_matchup4date.sh, processes one date (one control file) at a time. Calls IDL to run the batch file commands, and tallies its success or failure in the database. |
| IDL Batch File Called | polar2pr.bat, sets up matchup parameters and executes the polar2pr procedure |
| Output matchup files | **GRtoPR.rad.yymmdd.oo.vv.n\_n.uu.nc.gz**, where:  **GRtoPR**: Literal value.  **rad**: ID of ground radar, e.g. KAMX  **yymmdd**: date of event  **oo**: TRMM orbit number  **vv**: TRMM product version (6 or 7 for the original TRMM product naming convention; V06 or V07 for TRMM products generated in the GPM era.)  **n\_n**: Matchup file version, e.g. 1\_21 for version 1.21  **uu**: *Optional* user-supplied addition to the filename  **nc**: Literal value indicating a netCDF file type  **gz**: Literal value, present if the file is gzip compressed |
| Log Files | doGeoMatch4NewRainCases.SUBSET.yymmdd.log (main script log, where yymmd here is the current calendar date when the script was started, and SUBSET is the orbit subset ID of the PR data files used in the matchup.)  do\_geo\_matchup4date.YYMMDD.log (child script log, where YYMMDD here is the date of the matchup events configured in the file) |

DESCRIPTION

-----------

Wrapper to do PR-GR NetCDF geometric matchups for 1C21/2A25/2B31/1CUF files

already received and cataloged, for cases meeting predefined criteria.

Criteria are as defined in the query that is run to update the table

"rainy100inside100" in the "gpmgv" database. Includes cases where the PR

indicates "rain certain" at 100 or more gridpoints within 100 km of the radar

within the 4km gridded 2A-25 product. See the SQL command file

'rainCases100kmAddNewEvents.sql'.

SYNOPSIS

--------

do\_PR\_GeoMatch.sh [OPTION]...

OPTIONS/ARGUMENTS:

-----------------

-v PPS\_VERSION Override default PPS\_VERSION to the specified PPS\_VERSION.

This determines which version of the TRMM products will be

processed in the matchups. STRING type.

-p PARAMETER\_SET Override default PARAMETER\_SET to the specified PARAMETER\_SET

This tracks which version of IDL batch file was used in

processing matchups, when changes are made to the batch file.

This value does not actually control anything, but it gets

written to the geo\_match\_product table in the gpmgv database

as a descriptive attribute. It's up to the user to keep track

of its use and meaning. INTEGER type.

-u SUBSET Override default (CONUS) orbit subset of data to process to the

specified subset. This also controls which ground radar sites

will be included in the processing.

NOTE: When running dates that might have already had PR-GR matchup sets

run, the called script will skip these dates, as the 'appstatus' table

will say that the date has already been done. Delete the entries

from this table where app\_id='geo\_match\_PR', either for the date(s) to

be run, or for all dates.

## doTMIGeoMatch4NewRainCases.sh

The legacy script doTMIGeoMatch4NewRainCases.sh processes matchups between the TRMM TMI and ground radars using pre-GPM-era TRMM 2A-12 HDF4 files. Starting with the GPM era, TRMM TMI products were produced and/or reprocessed in the 2A-GPROF HDF5 format, whose matchups are handled by the do\_GMI\_GeoMatch.sh script (Section 2.3) with options specified to instruct TRMM TMI products to be processed. There is about a 2-week overlap between files in the old naming convention (located under /data/gpmgv/prsubsets/2A12) in a ‘flat’ directory, and those with the new naming convention and PPS-like directory structures (located under /data/gpmgv/orbit\_subset/TRMM).

|  |  |
| --- | --- |
| Main Script Name | doTMIGeoMatch4NewRainCases.sh |
| Satellite Products Matched | 2A12 |
| Scan Type(s) Matched | Not applicable, TMI does not have multiple scans. |
| Control Files | TMI\_files\_sites4geoMatch.yymmdd.txt, where yymmdd is the date of the matchup events. |
| Child Script Called | do\_TMI\_geo\_matchup4date.sh, processes one date (one control file) at a time. Calls IDL to run the batch file commands, and tallies its success or failure in the database. |
| IDL Batch File Called | polar2tmi.bat, sets up matchup parameters and executes the polar2tmi procedure |
| Output matchup files | **GRtoTMI.rad.yymmdd.oo.vv.n\_n.uu.nc.gz**, where:  **GRtoPR**: Literal value.  **rad**: ID of ground radar, e.g. KAMX  **yymmdd**: date of event  **oo**: TRMM orbit number  **vv**: TRMM product version (6 or 7 for the original TRMM product naming convention)  **n\_n**: Matchup file version, e.g. 1\_21 for version 1.21  **uu**: *Optional* user-supplied addition to the filename  **nc**: Literal value indicating a netCDF file type  **gz**: Literal value, present if the file is gzip compressed |
| Log Files | doTMIGeoMatch4NewRainCases.Vvv.yymmdd.log (main script log, where yymmd here is the current calendar date when the script was started, vv is the version of TMI data files used in the matchup (6 or 7), and all other characters are fixed, literal values.  do\_TMI\_geo\_matchup4date.YYMMDD.log (child script log, where YYMMDD here is the date of the matchup events configured in the file) |

DESCRIPTION

-----------

Query gpmgv database for dates/times of rainy PR or GR events and assemble

command string to do the TMI-GR geometry matching for events with data.

Completed geometry match files are cataloged in the 'gpmgv' database table

'geo\_match\_products'.

SYNOPSIS

--------

doTMIGeoMatch4NewRainCases.sh

OPTIONS/ARGUMENTS:

-----------------

None. All reconfiguration is done by editing values of variables internal to the script.

NOTE: When running dates that might have already had TMI-GR matchup sets

run, the called script will skip these dates, as the 'appstatus' table

will say that the date has already been done. Delete the entries

from this table where app\_id='geo\_match\_tmi', either for the date(s) to

be run, or for all dates.