Document NWPSAF-KN-TV-008
Version 2.1
February 2017

PenWP Test Plan and Test Report

Anton Verhoef, Jur Vogelzang, Jeroen Verspeek and Ad Stoffelen

KNMI, De Bilt, the Netherlands

PenWP Test Plan and Test Report

KNMI, De Bilt, the Netherlands

This documentation was developed within the context of the EUMETSAT Satellite Application Facility on Numerical Weather Prediction (NWP SAF), under the Cooperation Agreement dated 29 June, 2011, between EUMETSAT and the Met Office, UK, by one or more partners within the NWP SAF. The partners in the NWP SAF are the Met Office, ECMWF, KNMI and Météo France.

Change record			
Version	**Date**	**Author / changed by**	**Remarks**
1.9	May 2015	Anton Verhoef	First version for PenWP beta release
2.0	Oct 2015	Anton Verhoef	Version for first public PenWP release
2.0.01	Nov 2015	Anton Verhoef	Modified according to DRI comments
2.1	Feb 2017	Anton Verhoef	Version for PenWP 2.1

Contents

1 Introduction

This document is the test plan and test report for the Pencil beam Wind Processor (PenWP) software package. It is set up according to the guidelines of the NWP SAF; see the NWP SAF Development Procedures for Software Deliverables. Parts of the PenWP developments are in fact genscat developments. The tests for genscat modules are also included in this document. Part of the test plan is a traceability matrix to show how requirements as described in the Product Specification [2] are related to the tests in this document.

Most of the module tests described in this document have been developed and performed for OWDP (the OSCAT Wind Data Processor), AWDP (the ASCAT Wind Data Processor) and SDP (the SeaWinds Data Processor) a large part of the code in genscat is shared between PenWP and other NWP SAF wind processors. For this new PenWP version, all module tests have been repeated.

1.1 Aims and scope

The Pencil Beam Wind Processor (PenWP) is a software package written mainly in Fortran 90 with some parts in C for handling data from the SeaWinds (on QuikSCAT or ADEOS-II), OSCAT (on Oceansat-2 or ScatSat-1), HSCAT (on HY-2A) and RapidScat (on the International Space Station) scatterometer instruments. Details of these instruments can be found in [4] and [5], respectively, and on several web sites, see e.g. information on the NASA and ISRO web sites. PenWP is intended to be a generic wind processor for Ku band pencil beam scatterometer data. It will be adapted to handle data from future instruments like the OSCAT successor ScatSAT (from ISRO) once they become available.

PenWP generates surface winds based on pencil beam radar backscatter data. It allows performing the ambiguity removal with the Two-dimensional Variational Ambiguity Removal (2DVAR) method and it supports the Multiple Solution Scheme (MSS). The output of PenWP consists of wind vectors which represent surface winds within the ground swath of the scatterometer. Input of PenWP is Normalized Radar Cross Section (NRCS, σ^0) data. These data may be near real-time. The input files of PenWP are in BUFR. Conversion programs are included in the package to convert Hierarchical Data Format (HDF5) data from various instruments to BUFR. Output is written using the SeaWinds BUFR template or the KNMI BUFR template with generic wind section.

Depending on the grid spacing of the BUFR product, PenWP will process the data on 25 km, 50 km or 100 km grid spacing. The SeaWinds/RapidScat HDF5 to BUFR converter can create BUFR data on 25, 50 or 100 km grid spacing by averaging the backscatter data in the level 2a input file to the requested gridding. The OSCAT HDF5 to BUFR converter will create BUFR data with the same grid spacing as the level 2a input file. This can be 50 km when using the level 2a input data from the Indian Space Research Organisation (ISRO), or 25 km when using level 1b input data from ISRO in combination with a separate OSCAT level 1b to level 2a converter which is also included in the software package and which is based on software provided by NOAA. HY-2A input data are currently available on 25 km grid spacing but can also be averaged to a 50 km product.

Apart from the scatterometer input data, PenWP needs Numerical Weather Prediction (NWP) model winds as a first guess for the Ambiguity Removal step. These data need to be provided in GRIB edition 1 or 2.

1.2 Development of PenWP

PenWP is developed within the Numerical Weather Prediction Satellite Application Facility (NWP SAF) and Ocean and Sea Ice Satellite Application Facility (OSI SAF) projects as code which can be run in an operational setting. The coding is mainly in Fortran 90 with some parts in C and has followed the procedures specified for the NWP SAF. Special attention has been paid on robustness and readability. PenWP may be run on every modern Unix or Linux machine. In principle, PenWP can also be run on a Windows machine if a Linux environment like the Windows Installer for Ubuntu (Wubi) is installed. Details on the PenWP package can be found in [1], [2] and [3].

1.3 Testing PenWP, traceability matrix

This section describes the Test Plan of the PenWP deliverable. Tests have been carried out in all stages of the development of PenWP. The inversion module is not tested for the PenWP package, because such a test has already been made for the QuikSCAT Data Processor (QDP) development. PenWP contains several methods for Ambiguity Removal within module *ambrem* and its sub modules. Only modules needed for the KNMI 2DVAR scheme for Ambiguity Removal are tested within this project.

Compilation is done on several platforms (operating systems) and with different Fortran 90 compilers. The integration and validation tests were done on both a Linux work station and a Linux server environment.

Section 2 contains the tests for a number of individual modules. In general, modules are tested with the associated test programs that are located in the folder containing the module under consideration. The output of the test programs is always the standard output (screen) which may be redirected to any test log file or to some output files which are stored in the associated folders. Section 3 describes the PenWP integration test. A test folder containing some sample data is provided with PenWP and some of the resulting wind fields from these data are shown. Section 4 discusses the validation tests. PenWP has been compared with ECMWF model winds in the scope of this report, buoy validations are or will be performed in the scope of the OSI SAF. Section 4 also contains a technical check of the ice screening algorithm. Section 5 describes the portability tests. It contains an overview of platform/operating systems and Fortran and C compilers for which PenWP is supported. Finally, section 6 is devoted to testing the user documentation.

The table below is the traceability matrix. It shows the requirements in the Product Specification [2], how they are tested and where in this report these tests are described.

Requirement	Section of PS	Testing method	Test plan reference (section)	Comment
PenWP generates surface winds	2.1, 3.5, 3.7	Process L2A file in penwp/test folder and inspect output	3.1	
PenWP generates BUFR output in NOAA format and in KNMI format	2.1, 3.1	Process L2A file in penwp/test folder and inspect output	3.1	
PenWP generates output in the same WVC spacing as the input data	2.2, 3.2	Process L2A file in penwp/test folder and inspect output	3.1	
PenWP output contains latitude, longitude and other parameters	2.2	Process L2A file in penwp/test folder and inspect output	3.1	
PenWP can use either L2A HDF5 data or BUFR data as input (HDF5 after conversion to BUFR)	2.3	Process L2A HDF5 data in penwp/test folder and subsequently reprocess BUFR output	3.1	
PenWP reads GRIB data containing LSM, SST and forecast winds	2.3	Process L2A file in penwp/test folder and check that a consistent wind field is obtained	3.1	
PenWP will compile and run on different Linux and Unix platforms	2.4	Compile and run PenWP on different platforms	5	
L2A backscatter slices are averaged correctly, unusable backscatter data are rejected.	3.2	Process a few orbits of L2A data and compare output winds to ECMWF background.	4.1	When averaging is not done well, a noisy or inconsistent wind field is obtained. This is reflected in the statistics of scatterometer winds vs. ECMWF.
Atmospheric attenuations are computed and stored in output	3.3	Process L2A file in penwp/test folder and inspect output	3.1	Atmospheric attenuations should be in the order of 0.2 to 0.3 dB
WVCs with high MLEs must be rejected by Quality Control	3.4	Process L2A file in penwp/test folder and check if QC flag is set for high MLE values	3.1	
Bayesian ice screening is implemented	3.6	Process a few orbits of L2A data and inspect ice maps	4.2	
A product monitoring flag is implemented	3.8	Not tested since there are no data with anomalous instrument performance available	-	
PenWP can process data within reasonable CPU time.	3.9	Process L2A file in penwp/test folder and check processing time.	3.1	

Table 1.1 Traceability matrix.

1.4 Test folders

The Test folder of the PenWP software package is located in subdirectory `penwp/tests`. This subdirectory contains several input files for PenWP that are discussed in more detail in section 3. The scripts for executing these tests are located in directory `penwp/execs`. It is recommended to use these scripts (or a modified version) also for normal PenWP operation, as the environment variables needed by PenWP are set in these scripts.

As stated before, most test programs are located in the same directory as the module to be tested. See section 2 for detailed information.

1.5 Conventions

Names of physical quantities (e.g., wind speed components *u* and *v*), modules (e.g. *BufrMod*), subroutines and identifiers are printed italic.

Names of directories and subdirectories (e.g. `penwp/src`), files (e.g. `penwp.F90`), and commands (e.g. `penwp -f input`) are printed in Courier. Software systems in general are addressed using the normal font (e.g. PenWP, genscat).

Hyperlinks are printed in blue and underlined (e.g. http://www.knmi.nl/scatterometer/).

2 Module tests

In this section the various tests to individual modules within PenWP are presented. The tests are listed alphabetically in the module name. Table 2.1 gives an overview of the modules tested, their location and the name of the associated test programs.

Module tests have been included in PenWP if the following conditions were satisfied:

1. The test does not require additional software.
2. The output of the test program is self-explanatory enough to judge the outcome of the test.

Module name	**Location**	**Test program**
BFGSMod	`genscat/support/BFGS`	*Test_BFGS*
BufrMod	`genscat/support/bufr`	*test_modules*
convert	`genscat/support/convert`	*test_convert*
CostFunction	`genscat/ambrem/twodvar`	*Test_SOS*
StrucFunc	`genscat/ambrem/twodvar`	*Test_SOS*
DateTimeMod	`genscat/support/datetime`	*TestDateTimeMod*
ErrorHandler	`genscat/support/ErrorHandler`	*TestErrorHandler*
gribio_module	`genscat/support/grib`	*test_read_GRIB1*, *test_read_GRIB2*, *test_read_GRIB3*
HDF5Mod	`genscat/support/hdf5`	*TestHDF5*
LunManager	`genscat/support/file`	*TestLunManager*
numerics	`genscat/support/num`	*test_numerics*
SingletonFFT	`genscat/support/singletonfft`	*TestSingleton*
SortMod	`genscat/support/sort`	*SortModTest*

Table 2.1 Overview of module tests.

2.1 Module *BFGSMod*

Directory `genscat/support/BFGS` contains program `Test_BFGS`. This program tests the minimization routine LBFGS and its associated routines in module *BFGSMod*. The routines in *BFGSMod* are slightly modified versions of the freeware routine `LBFGS` and its subroutines. `LBFGS` was written by J. Nocedal, see [6].

Program `Test_BFGS` finds the minimum of the function

$$f(x) = \sum_{i=1}^{100000} (x - i)^4$$

The minimum is the point (1, 2, …, 100000). The search starts at the origin. The typical output is shown in table 2.2.

```
Program Test_BFGS testing routine LBFGS

 Behavour of cost function:
 Iter         Cost
 -----------------
    0  0.20001E+25
    1  0.19527E+25
    2  0.17724E+25
…
   84  0.29492E-15
   85  0.95608E-16
   86  0.30995E-16

 Routine LBFGS completed succesfully
   Number of iterations                 :      87
   Dimension of problem                 :  100000
   Number of corrections in BFGS update :       5
   Cost function at start               :  0.20001D+25
   Cost function at end                 :  0.30995D-16
   Precision required                   :  0.10D-19
   Norm of final X                      :  0.18258D+08
   Norm of final G                      :  0.97625D-13
   Minimum and Maximum error in solution :  0.000003  0.000005
   Time needed                          :  0.460  seconds
 Program Test_BFGS completed succesfully.
```

Table 2.2 Output of program Test_BFGS.

2.2 Module *BufrMod*

Directory `genscat/support/bufr` contains program *test_modules*. This program is compiled and called automatically by the genscat make system, since it is needed to translate the ASCII BUFR tables to binary form. It will also read in a small BUFR test file, decode it, encode the data again and write them to an output BUFR file. Hence, the program can be used to check the BUFR library. Table 2.3 shows the output generated by *test_modules*. The program can be invoked by calling the shell script *run_test_modules*, which sets the environment variable `$BUFR_TABLES` and calls *test_modules*.

```
nr of BUFR messages in this file is:     1
                   ECMWF

      BUFR DECODING SOFTWARE VERSION -  403



 Your path for bufr tables is :
 ./bufr_tables/
BUFR TABLES TO BE LOADED  B0000000000210000001.TXT,D0000000000210000001.TXT
 tbd%nelements =           44
 pos_lat =           25
 pos_lon =           26
 latitude  range:    -3.630000        1.260000
 longitude range:     2.850000        7.690000
                   ECMWF

      BUFR ENCODING SOFTWARE VERSION -  7.2
            1 April  2007.



 Your path for bufr tables is :
 ./bufr_tables/
BUFR TABLES TO BE LOADED  B0000000000210000001.TXT,D0000000000210000001.TXT
```

Table 2.3 Output of program *test_modules.*

2.3 Module *convert*

Directory `genscat/support/convert` contains module *convert.F90*, a number of routines for the conversion of meteorological and geographical quantities. Its associated test program is *test_convert*, and part of its output is listed in table 2.4. Program *test_convert* produces quite a lot of output.

It starts with checking some conversions between different wind vector representations and transformations between different geographical coordinate systems, followed by a check of the transformation from orbit angles (p,a,rot(z)) to three-dimensional position (x,y,z).

Only the results for p = 0° and 90° are (partly) shown in table 2.4; those for p = 10°, 45°, and 70° are omitted. Program *test_convert* ends with some trigonometric calculations on a sphere.

```
===========================================
 u =     5.000000      v =    -7.000000
 uv_to_speed, uv_to_dir ====> sp =     8.602325      dir =     324.4623
 ===========================================
 sp =     8.602325      dir =     324.4623
 speeddir_to_u, speeddir_to_v ====> u =     5.000002      v =    -6.999999
 ===========================================
 met2uv: sp =     10.00000      dir =     135.0000
 met2uv: ====> u =    -7.071068      v =     7.071068
 uv2met: u =    -7.071068      v =     7.071068
 uv2met: ====> sp =     10.00000      dir =     135.0000
 ===========================================
 lat,lon =     55.00000        5.000000
 latlon2xyz: ====> x,y,z =    0.5713938       4.9990479E-02   0.8191521
 x,y,z =    0.5713938       4.9990479E-02   0.8191521
 xyz2latlon: ====>lat,lon =     55.00000        5.000000
 ===========================================
         p          a      rot_z          x          y          z         a1     rot_z1         a2     rot_z2
   0.00000  -90.00000    0.00000    0.00000    0.00000   -1.00000  -90.00000  106.16298  270.00000    0.00000
   0.00000  -90.00000   15.00000    0.00000    0.00000   -1.00000  -90.00000  105.59795  270.00000    9.72975
   0.00000  -90.00000   30.00000    0.00000    0.00000   -1.00000  -90.00000  103.95005  270.00000   27.91061
   0.00000  -90.00000   45.00000    0.00000    0.00000   -1.00000  -90.00000  101.35209  270.00000   43.81981
   0.00000  -90.00000   60.00000    0.00000    0.00000   -1.00000  -90.00000   98.00070  270.00000   59.32336
   0.00000  -10.00000    0.00000    0.98481    0.00000   -0.17365  -10.00000    0.00000  190.00000  180.00000
   0.00000  -10.00000   15.00000    0.95125    0.25489   -0.17365  -10.00000   15.00000  190.00000 -164.99998
   0.00000  -10.00000   30.00000    0.85287    0.49240   -0.17365  -10.00000   30.00000  190.00000 -149.99998
…
  90.00000   45.00000   30.00000    0.25882    0.96593    0.00000   74.99999    0.00000  105.00000    0.00000
  90.00000   45.00000   45.00000    0.00000    1.00000    0.00000   90.00000    0.00000   90.00000    0.00000
  90.00000   45.00000   60.00000   -0.25882    0.96593    0.00000   74.99999    0.00000  105.00000    0.00000
  90.00000   90.00000    0.00000    0.00000    1.00000    0.00000   90.00000    0.00000   90.00000    0.00000
  90.00000   90.00000   15.00000   -0.25882    0.96593    0.00000   74.99999    0.00000  105.00000    0.00000
  90.00000   90.00000   30.00000   -0.50000    0.86603    0.00000   59.99999    0.00000  120.00000    0.00000
  90.00000   90.00000   45.00000   -0.70711    0.70711    0.00000   45.00000    0.00000  135.00000    0.00000
  90.00000   90.00000   60.00000   -0.86603    0.50000    0.00000   30.00000    0.00000  149.99998    0.00000
 ===========================================
 latlon1 =     5.000000        5.000000     latlon2 =     6.000000
    5.000000
 angle distance =     1.000000
 km distance    =     111.3188
 latlon1 =     55.00000        5.000000     latlon2 =     56.00000
    5.000000
 angle distance =     1.000000
 km distance    =     111.3188
 latlon1 =     85.00000        5.000000     latlon2 =     86.00000
    5.000000
 angle distance =     1.000000
 km distance    =     111.3188
 ===========================================
 latlon1 =     5.000000        5.000000     latlon2 =     5.000000
    6.000000
 angle distance =    0.9961947
 km distance    =     110.8952
```

```
latlon1 =     55.00000        5.000000     latlon2 =     55.00000
   6.000000
angle distance =    0.5735765
km distance    =     63.84987
latlon1 =     85.00000        5.000000     latlon2 =     85.00000
   6.000000
angle distance =    8.7155804E-02
km distance    =     9.702084
===========================================

Test WVC_Orientation
  WVC1 coordinates (Lam1,Phi1) =   -115.2000       -18.61000
  WVC2 coordinates (Lam2,Phi2) =   -123.6500       -17.52000
  WVC1 orientation Alfa1 =    173.5995        (Should equal 173.5994720)
  WVC2 orientation Alfa2 =    170.9747        (Should equal 170.9747467)
===========================================
```

Table 2.4 Output of program *test_convert*

2.4 Modules *CostFunction* and *StrucFunc*

Module *CostFunc.F90* in directory `genscat/ambrem/twodvar` contains the cost function definition of the 2DVAR method. Module *StrucFunc* in the same directory contains the error covariance model of the background field. Large parts of these modules are tested in the single observation solution test implemented in program *Test_SOS*. Table 2.5 lists its output.

The main idea behind this test is that the 2DVAR analysis increment can be calculated analytically in case of one single observation with unit probability. Starting with zero background increment and an observation increment (t_o,l_o) on the 2DVAR grid at the position with indices (1,1), the initial total cost function equals

$$J_t^{init} = \frac{t_o^2 + l_o^2}{\varepsilon^2}$$

where ε stands for the standard deviation of the observation error, which is set to 1.8 in *Test_SOS*. The 2DVAR problem now reduces to a simple optimal interpolation problem. If the standard deviation of the background error is set to the same value as that of the observation error, the final solution has $J_t^{fin} = J_o^{fin} + J_b^{fin} = ½\ J_t^{init}$ with $J_b^{fin} = J_o^{fin}$. This allows construction of the final solution and its gradient, see [7] for more detailed information and a complete description of the 2DVAR method.

Program *Test_SOS* reads the observation increment and the structure function parameters from an input file with default name *Test_SOS.inp*, see below. There are two modes for calculating the Helmholz transformation coefficients, controlled by the variable *Mode* in routine *Set_HelmholzCoefficients* in module *CostFunc.F90*. Mode is a character variable of length 2. Its default value is '*JV*' which stands for sampled continuum (the other value is '*HB*' which stands for periodic boundary conditions but these do not reproduce the correct scaling, see [7] for more details). The program copies the structure function parameters into the *SF*-struct, and the observation increments in the *TwoDvarObs*-struct. The structure function parameters are printed by routine *PrnStrucFuncPars*.

The error covariances are calculated numerically in module *StrucFunc*. For Gaussian structure functions, they can also be calculated analytically. The two methods are compared and the relative precision is printed. In table 2.5 it is 0.00345 for the stream function ψ and 0.0 for the velocity potential χ, since the latter quantity is identically zero in this example. The precision of the covariances depends on the correlation lengths R_ψ and R_χ.

The total cost function and its gradient is evaluated by routine *JoScat* in module *CostFunction*. From this the cost function components and gradients at the final solution are calculated and checked against their analytical value. The (absolute) precision is printed. Finally, *Test_SOS* checks the packing and unpacking routines of the control vector in both directions.

As stated before, program *Test_SOS* reads its input from an input file. The name (and path) of that file must be given as command line argument of *Test_SOS*. When omitted, the program assumes `Test_SOS.inp` as input file. Table 2.6 gives the structure and contents of the input file, which is in free format. The last decimals of the output values may depend on machine precision.

```
================================================================
PROGRAM Test_SOS - Single Observation Soluton Check
================================================================


Input read from file        : Test_SOS.inp
Helmholz coefficients type  : JV
2DVAR:
2DVAR: Parameters inside the StructFunc module:
2DVAR:  Grid size in position domain      :    100000.0       m
2DVAR:  Grid dimensions                   :          32  by          32
2DVAR:  Free edge size                    :           5  points
2DVAR:  Structure function type           : Gaus
2DVAR:  Northern hemisphere:
2DVAR:    Error standard deviation in psi :    1.800000      m/s
2DVAR:    Error standard deviation in chi :    1.800000      m/s
2DVAR:    Rotation/divergence ratio       :    1.000000
2DVAR:    Range parameter for psi         :    300000.0
2DVAR:    Range parameter for chi         :    300000.0
2DVAR:  Tropics:
2DVAR:    Error standard deviation in psi :    2.000000      m/s
2DVAR:    Error standard deviation in chi :    2.000000      m/s
2DVAR:    Rotation/divergence ratio       :   0.1000000
2DVAR:    Range parameter for psi         :    300000.0
2DVAR:    Range parameter for chi         :    300000.0
2DVAR:  Southern hemisphere:
2DVAR:    Error standard deviation in psi :   -1.000000      m/s
2DVAR:    Error standard deviation in chi :    76.00000      m/s
2DVAR:    Rotation/divergence ratio       :    0.000000
2DVAR:    Range parameter for psi         :    1.800000
2DVAR:    Range parameter for chi         :    1.800000

CheckCovMat - checking precision of Covariances
  Relative precision in covariances of psi:   3.3184644E-04
  Relative precision in covariances of chi:   2.7596165E-04

Number of observations      :           1
Number of control variables :        2046

Obs2dvar after initialization:
 i  j  Namb    u    v          Jo          gu          gv
-------------------------------------------------------------
 1  1     1  1.0  0.0  0.00000E+00  0.00000E+00  0.00000E+00

The gradient velocity fields duo and dvo (nonzero components only):
 i  j         duo         dvo
-------------------------------

The cost function of the solution:
  Observation part :    0.000000
  Background part  :    0.000000            precision    0.000000

The background velocity field:
```

```
  u(1,1)         :    0.000000
  Expected value :   0.5000000                 precision   0.5000000
  v(1,1)         :    0.000000
  Expected value :    0.000000                 precision    0.000000

Check background cost function
  Direct calculation from psi and chi    :    0.000000
  Calculation by Jb from control vector :    0.000000            precision     0.000000

Check observation cost function
  Expected value                         :    0.000000
  Calculation by Jo from control vector :    0.000000            precision     0.000000
  Precision in gradients better than    1.9753901E-10

Check packing/unpacking:
  Precision in packing/unpacking of xi    0.000000
  Precision in packing/unpacking of psi    0.000000
  Precision in packing/unpacking of chi    0.000000

Program Test_SOS completed.
================================================================
```

Table 2.5 Output of the single observation solution test.

Record	Item nr.	Name	Meaning
1	1	u0_ini	Initial observation increment in transversal direction (m/s)
1	2	v0_ini	Initial observation increment in longitudinal direction (m/s)
2	1	lparameter	Logical parameter indicating if 2DVAR parameters should be read from file
3	1	TDVParameterFile	Name of 2DVAR parameter file

Table 2.6 Input file for *Test_SOS*.

2.5 Module *DateTimeMod*

Module *DateTimeMod.F90* in directory `genscat/support/datetime` contains general purpose date and time help functions. These are tested by program *TestDateTimeMod*, the output of which is listed in table 2.7.

```
 time-tests
time: 14:22:03.70
time_real        = 51723.70
time_real + 77.2 = 51800.90
time: 14:23:20.90
 time2 is valid
 time1 =
time: 14:22:03.70
 time2 =
time: 14:23:20.90
 time 1 .ne. time2
 date-tests
date: 15-12-1999
 date_int =      19991215
 date_int + 1 =      19991216
date: 16-12-1999
 date2 is valid
 date1 =
date: 15-12-1999
 date2 =
date: 16-12-1999
 date 1 .ne. date2
 date-stepping-tests
 ERROR: The date      21000101  is outside the range
 19000101...20991231, this is not implemented at this time
 ERROR: Julian routines differ from my own routines
date: 31-12-2099
```

```
 next_date_int =    2147483647
date: 01-01-2100
 next_julian_date_int =      21000101
 all OK
 before:
time: 23:59:57.70
date: 31-12-1999
after incrementing by:  5.22 seconds
time: 00:00:02.92
date: 01-01-2000
 valid time
 test of function date2string: 19991231
 test of function date2string_sep: 1999-12-31
 test of function time2string: 235957
 test of function time2string_sep: 23:59:57
 before convert_to_derived_datetime:
date: 28-02-2005
time: 52:00:00.00
 after convert_to_derived_datetime:
date: 02-03-2005
time: 04:00:00.00
 Current date and time:
date: 16-04-2015
time: 09:20:37.13
```

Table 2.7 Output of program *TestDateTimeMod*.

2.6 Module *ErrorHandler*

Module *ErrorHandler.F90* in directory `genscat/support/ErrorHandler` contains routines for handling errors during program execution. The module is tested by program *TestErrorHandler*, the output of which is listed in table 2.8.

```
The Error Handler program_abort routine is set to
return after each error,
in order to try and resume the program...
testing: report_error
an error was reported from within subroutine: dummy_module_name1
error while allocating memory
testing: program_abort (with abort_on_error = .false.)
an error was reported from within subroutine: dummy_module_name2
error while allocating memory
==> trying to resume the program ...
The Error Handler program_abort routine is set to
abort on first error...
testing: program_abort (with abort_on_error = .true.)
an error was reported from within subroutine: dummy_module_name2
error while allocating memory
```

Table 2.8 Output of program *TestErrorHandler*.

2.7 Module *gribio_module*

Module *gribio_module.F90* in directory `genscat/support/grib` contains routines for reading and decoding GRIB files. The module is tested by programs *test_read_GRIB1*, *test_read_GRIB2* and *test_read_GRIB3*, the output of which is listed in tables 2.9 to 2.11. The test programs read in two small GRIB files (`testfile.grib` in GRIB edition 1 format and `testfile.grib2` in GRIB edition 2 format) present in this directory and print some of their contents to the standard output. The environment variable `$GRIB_DEFINITION_PATH` needs to be set and has to point to the directory containing GRIB definition tables. These are available in `(...)/genscat/support/grib/definitions`.

```
open GRIB edition 1 file
file name = ./testfile.grib
date of grib field =            20031111
time of grib field =                  24
derived date of grib field =    20031112
derived time of grib field =           0

    lat    lon      10u      10v    speed
  54.00   4.00   -4.576    8.006    9.221
  54.00   4.50   -5.143    7.764    9.313
  54.00   5.00   -5.034    7.520    9.050
  54.00   5.50   -4.925    7.276    8.786
  54.50   4.00   -4.849    8.455    9.747
  54.50   4.50   -5.139    8.315    9.775
  54.50   5.00   -5.200    8.426    9.902
  54.50   5.50   -5.261    8.537   10.028
  55.00   4.00   -5.267    8.577   10.065
  55.00   4.50   -5.398    8.454   10.031
  55.00   5.00   -5.416    8.620   10.180
  55.00   5.50   -5.434    8.786   10.330
  55.50   4.00   -5.686    8.699   10.392
  55.50   4.50   -5.657    8.594   10.289
  55.50   5.00   -5.632    8.814   10.459
  55.50   5.50   -5.606    9.034   10.632

open GRIB edition 2 file
file name = ./testfile.grib2
date of grib field =            20031111
time of grib field =                  24

End of tests
```

Table 2.9 Output of program *test_read_GRIB1*.

```
retrieve grib field par_id_t
lat of first gridpoint =   89.142
lat step               =   -1.121
number of lat points   =      160
lon of first gridpoint =    0.000
lon step               =    1.125
number of lon points   =      320

    i    j  field(i,j)
   80  160     302.663
   80  161     302.445
   80  162     302.148
   80  163     301.560
   81  160     301.999
   81  161     302.298
   81  162     301.808
   81  163     301.708
   82  160     302.056
   82  161     302.117
   82  162     301.490
   82  163     301.888
   83  160     302.214
   83  161     302.001
   83  162     301.796
   83  163     302.361
```

Table 2.10 Output of program *test_read_GRIB2*.

```
retrieve grib field par_id_10u
date of grib field =    20031111
time of grib field =          24
 WARNING: latitude dimension of field is too small to contain
 WARNING: the read data; truncating the array !!!!!
 original:  nr_lat_points =           160
 truncated: nr_lat_points =            50
 WARNING: longitude dimension of field is too small to contain
 WARNING: the read data; truncating the array !!!!!
 original:  nr_lon_points =           320
 truncated: nr_lon_points =            50

    i    j  field(i,j)
   48   48      -0.414
   48   49       0.477
   48   50      -0.111
   49   48       3.330
   49   49       2.899
   49   50       3.252
   50   48       3.503
   50   49       2.408
   50   50       3.212
```

Table 2.11 Output of program *test_read_GRIB3*.

2.8 Module HDF5Mod

Module *HDF5Mod.F90* in directory `genscat/support/hdf5` contains routines for reading and writing HDF5 files. It is tested by program *TestHDF5*, the output of which is listed in table 2.12. The test program reads in a small HDF5 file called `deflate.h5` and displays some of its contents. After that, it creates a file called `testfile.h5` and writes some data into it. Its contents can be checked e.g. with the command line utility `h5dump`.

```
Successfully opened file deflate.h5 with f_id     67108864
Successfully opened dataset //Dataset1 with d_id    335544320
Successfully closed dataset with d_id    335544320
Successfully opened group / with g_id    134217728
Successfully opened dataset Dataset1 with d_id    335544321
Number of datapoints of dataset    335544321  is        20000
First data values are:
      0       1       2       3       4       0       1       2       3       4
      0       1       2       3       4       0       1       2       3       4
Successfully closed dataset with d_id    335544321
Successfully closed group with g_id    134217728
Successfully closed file with f_id     67108864

End of file reading tests in TestHDF5

Successfully opened file testfile.h5 with f_id     67108865
Successfully created group Group1 with g_id    134217729
Successfully wrote Attribute Attribute1 in group    134217729
Successfully wrote Dataset Dataset1_int_1d in group    134217729
Successfully wrote Dataset Dataset2_int_2d in group    134217729
Successfully wrote Dataset Dataset3_float_1d in group    134217729
Successfully wrote Dataset Dataset4_float_2d in group    134217729
Successfully wrote Dataset Dataset5_string_1d in group    134217729
Successfully closed group with g_id    134217729
Successfully closed file with f_id     67108865

End of file writing tests in TestHDF5

A HDF5 file called testfile.h5 was created
You can check its contents e.g. using the h5dump utility

End of TestHDF5
```

Table 2.12 Output of program *TestHDF5*.

2.9 Module *LunManager*

Module *LunManager.F90* in directory `genscat/support/file` contains routines for file unit management. It is tested by program *TestLunManager*, the output of which is listed in table 2.13.

```
Starting fileunit test program
 ===== lun_manager ======
 fileunit:            31  was not in use !!!
 free_lun returns without freeing any fileunit
 fileunit:            88  was not in the range that is handled
 by this module ! (           30  -           39 )
 free_lun returns without freeing any fileunit
 fileunit:            88  was not in the range that is handled
 by this module ! (           30  -           39 )
 enable_lun returns without enabling any fileunit
 fileunit:            88  was not in the range that is handled
 by this module ! (           30  -           39 )
 disable_lun returns without disabling any fileunit
 fileunit:            21  was not in the range that is handled
 by this module ! (           30  -           39 )
 disable_lun returns without disabling any fileunit
 unit:            31  is used?:   F
 unit:            31  is used?:   T
 start of inspect_luns
  lun             0  is open
  lun             0  has a name: stderr
  lun             5  is open
  lun             5  has a name: stdin
  lun             6  is open
  lun             6  has a name: stdout
  lun            31  is open
  lun            31  has a name: TestLunManager.F90
 end of inspect_luns
 fileunit:            31  is still in use !
 disabling it is only possible if it is not used !
 disable_lun returns without disabling any fileunit
 fileunit:            30  is in use
 fileunit:            31  is in use
 fileunit:            32  is still available
 fileunit:            33  is still available
 fileunit:            34  is still available
 fileunit:            35  is still available
 fileunit:            36  is still available
 fileunit:            37  is still available
 fileunit:            38  is still available
 fileunit:            39  is still available
 fileunit:            21  was not in the range that is handled
 by this module ! (           30  -           39 )
 enable_lun returns without enabling any fileunit
 fileunit:            22  was not in the range that is handled
 by this module ! (           30  -           39 )
 enable_lun returns without enabling any fileunit
```

Table 2.13 Output of program *TestLunManager*.

2.10 Module *Numerics*

Module *numerics.F90* in directory `genscat/support/num` contains routines for checking and handling numerical issues like variable sizes and ranges. These are tested by program *test_numerics*, the output of which is listed in Table 2.14.

```
 Starting numerics test program
 ===== representation tests ======
 REALACC(6)
 r4: digits                 24
 r4: epsilon       1.1920929E-07
 r4: huge          3.4028235E+38
 r4: minexponent         -125
 r4: maxexponent          128
 r4: precision              6
 r4: radix                  2
 r4: range                 37
 r4: tiny          1.1754944E-38
 ENDREALACC
 REALACC(12)
 r8: digits                 53
 r8: epsilon       2.2204460492503131E-016
 r8: huge          1.7976931348623167E+308
 r8: minexponent        -1021
 r8: maxexponent         1024
 r8: precision             15
 r8: radix                  2
 r8: range                307
 r8: tiny          2.2250738585072010E-308
 ENDREALACC
 ===== numerics tests ======
 int1 =    127
 int2 =    32767
 int4 =    2147483647
 int8 =       9223372036854775807
 huge(int1) =          127
 huge(int2) =        32767
 huge(int4) =    2147483647
 huge(int8) =       9223372036854775807
 REALACC(6)  r4   =    1.7000000E+38  ENDREALACC
 REALACC(12) r8   =    1.7000000000000000E+038  ENDREALACC
 ===== check variable sizes  ======
 Variable sizes are as expected
 ===== detect and print variable sizes ======
var_type nr_of_words range precision
       i           4     9
     i1_           1     2
     i2_           2     4
     i4_           4     9
     i8_           8    18
      dr           4    37         6
      s_           4    37         6
      l_           4    37         6
      r_           4    37         6
     r4_           4    37         6
     r8_           8   307        15
 ===== dB conversion test ======
 REALACC(6)
 input test number:           1.2300001E-04
 converted to dB:             -39.10095
 converted back to a real:    1.2299998E-04
 ENDREALACC
 ===== done ======
```

Table 2.14 Output of program *test_mumerics*.

2.11 Module *SingletonFFT*

Module *SingletonFFT* in directory `genscat/support/singletonfft` contains routines for Fast Fourier Transforms. The associated test program is *TestSingleton*. Part of its output is shown in table 2.15.

```
========================================================================
 PROGRAM TestSingleton
 Test of SingletonFFT routines by comparing with analytical FT
 ========================================================================
```

```
Spreading times grid size in dimension 1:   0.1000000      (should be ~ 0.1)
Spreading times grid size in dimension 2:   0.1000000      (should be ~ 0.1)
=========================================================================

 1D             F O R W A R D          B A C K W A R D
           P r e c i s i o n         P r e c i s i o n
 N1        Real          Imag         Real          Imag
-----------------------------------------------------
 32  0.83631E-06  0.10286E-04  0.11921E-06  0.69247E-07
 34  0.61329E-06  0.78932E-05  0.11921E-06  0.11285E-07
 36  0.94782E-06  0.12215E-04  0.11921E-06  0.11036E-06
 38  0.27877E-06  0.20358E-05  0.17881E-06  0.22604E-07
 40  0.83631E-06  0.12143E-04  0.11921E-06  0.54017E-07
 42  0.44603E-06  0.56252E-05  0.77824E-07  0.92940E-07
 44  0.12900E-06  0.27819E-06  0.17881E-06  0.14948E-06
 46  0.94782E-06  0.13554E-04  0.35763E-06  0.34905E-07
 48  0.94782E-06  0.14143E-04  0.23842E-06  0.12666E-06
 50  0.50178E-06  0.66967E-05  0.17881E-06  0.10431E-06
=========================================================================

 2D                F O R W A R D   F F T         B A C K W A R D   F F T
               P r e c i s i o n               P r e c i s i o n
 N1  N2          Real          Imag   Time          Real          Imag   Time
-------------------------------------------------------------------------
 32  32  0.11995E-05  0.20572E-04 0.0000  0.17881E-06  0.10663E-06 0.0000
 32  34  0.10952E-05  0.18179E-04 0.0001  0.11921E-06  0.63061E-07 0.0000
 32  36  0.12516E-05  0.22501E-04 0.0000  0.11921E-06  0.11339E-06 0.0000
 32  38  0.88658E-06  0.82503E-05 0.0001  0.17881E-06  0.66826E-07 0.0001
 32  40  0.12516E-05  0.22430E-04 0.0000  0.17881E-06  0.95745E-07 0.0000
 32  42  0.99089E-06  0.15911E-04 0.0000  0.11921E-06  0.12151E-06 0.0000
 32  44  0.88658E-06  0.10286E-04 0.0001  0.29802E-06  0.17938E-06 0.0001
 32  46  0.11473E-05  0.23840E-04 0.0001  0.35763E-06  0.63112E-07 0.0001
 32  48  0.12516E-05  0.24430E-04 0.0000  0.27816E-06  0.12973E-06 0.0000
 32  50  0.10430E-05  0.16983E-04 0.0000  0.17881E-06  0.11206E-06 0.0000
 34  32  0.11473E-05  0.18179E-04 0.0001  0.11921E-06  0.78046E-07 0.0001
...
 48  50  0.10952E-05  0.20840E-04 0.0001  0.30120E-06  0.12803E-06 0.0001
 50  32  0.99089E-06  0.16983E-04 0.0000  0.17881E-06  0.11192E-06 0.0000
 50  34  0.83443E-06  0.14590E-04 0.0001  0.17881E-06  0.10692E-06 0.0001
 50  36  0.10430E-05  0.18912E-04 0.0001  0.23842E-06  0.11300E-06 0.0001
 50  38  0.46937E-06  0.47101E-05 0.0001  0.17881E-06  0.10619E-06 0.0001
 50  40  0.93873E-06  0.18840E-04 0.0001  0.35763E-06  0.11030E-06 0.0001
 50  42  0.62582E-06  0.12322E-04 0.0001  0.29802E-06  0.11184E-06 0.0001
 50  44  0.46937E-06  0.66967E-05 0.0001  0.29802E-06  0.14250E-06 0.0001
 50  46  0.99089E-06  0.20251E-04 0.0001  0.23842E-06  0.10202E-06 0.0001
 50  48  0.10430E-05  0.20840E-04 0.0001  0.29802E-06  0.15117E-06 0.0001
 50  50  0.57367E-06  0.13393E-04 0.0001  0.35763E-06  0.11255E-06 0.0001
=========================================================================

Program TestSingleton: Resume
Worst case accuracies

               F O R W A R D          B A C K W A R D
             Real          Imag         Real          Imag
-----------------------------------------------------
1D     0.94782E-06  0.14143E-04  0.35763E-06  0.14948E-06
2D     0.13559E-05  0.28287E-04  0.77486E-06  0.28650E-06

Program TestSingleton: Normal termination.
=========================================================================
```

Table 2.15 Output of program *TestSingleton*

2.12 Module *SortMod*

Module *SortMod* in directory `genscat/support/sort` contains two routines for sorting the wind vector solutions found in the inversion step to their probability. The associated test program is *SortModTest*. Its output is shown in table 2.16.

```
Test program for the SortMod module
Unsorted array
10.0  9.0  8.0  7.0  6.0  5.0  4.0  3.0  2.0  1.0
After GetSortIndex
 1.0  2.0  3.0  4.0  5.0  6.0  7.0  8.0  9.0 10.0
Sorted array, after SortWithIndex
 1.0  2.0  3.0  4.0  5.0  6.0  7.0  8.0  9.0 10.0
```

Table 2.16 Output of program *SortModTest*

3 PenWP integration test

Directory `penwp/tests` contains one HDF5 file for testing the PenWP executable. File `S1L2A2012006_12113_12114_2.h5.gz` contains (gzipped) OSCAT level 2a data from 6 January 2012, 13:51 to 14:03 UTC with 50 km cell spacing, as obtained from ISRO. The files `ECMWF*.grib` contain the necessary NWP data (SST, land-sea mask and wind forecasts) to perform the NWP collocation step.

The user can test the proper functioning of PenWP using the files in the `penwp/tests` directory. To do this, first create a small file containing a list of NWP files:

```
ls -1 ECMWF_* > nwpflist
```

Note that the '-1' contains the number '1' and not the character 'l'. Then, gunzip the HDF5 file:

```
gunzip -c S1L2A2012006_12113_12114_2.h5.gz >
S1L2A2012006_12113_12114_2.h5
```

Set the $BUFR_TABLES environment variable:

```
export BUFR_TABLES=../../genscat/support/bufr/bufr_tables/
```

for Korn shell or Bourne shell or

```
setenv BUFR_TABLES ../../genscat/support/bufr/bufr_tables/
```

for C shell. Convert the level 2a input file to BUFR:

```
../execs/oscat_hdf2bufr –allswath -f S1L2A2012006_12113_12114_2.h5
-o oscat.bufr
```

Then run PenWP:

```
../execs/penwp_run -f oscat.bufr -nwpfl nwpflist -mss -mon –noc
–genericws 4
```

The result should be two OSCAT level 2 files in BUFR format, called `oscat_20120106_135109_ocsat2_12113_o_500_ovw_l2.bufr` and `oscat_20120106_135109_ocsat2_12113_o_500_ovw_l2.bufr.genws`.

The first file is in NOAA BUFR format and the second file is in KNMI format with generic wind section.

3.1 OSCAT test data

Figure 3.1 shows the global coverage of the OSCAT test run on 50 km. The colours show the magnitude of the wind speed as indicated by the legend. Figure 3.2 shows detailed wind vector plots over the Atlantic west of Africa, with 50 km cell spacing. In the detail plot, a magenta marker on top of the wind arrow denotes land presence. Orange wind arrows indicate that the Variational

Quality Control flag is set, i.e. the Wind Vector Cell is spatially inconsistent. An orange dot means that the KNMI Quality Control Flag is set.

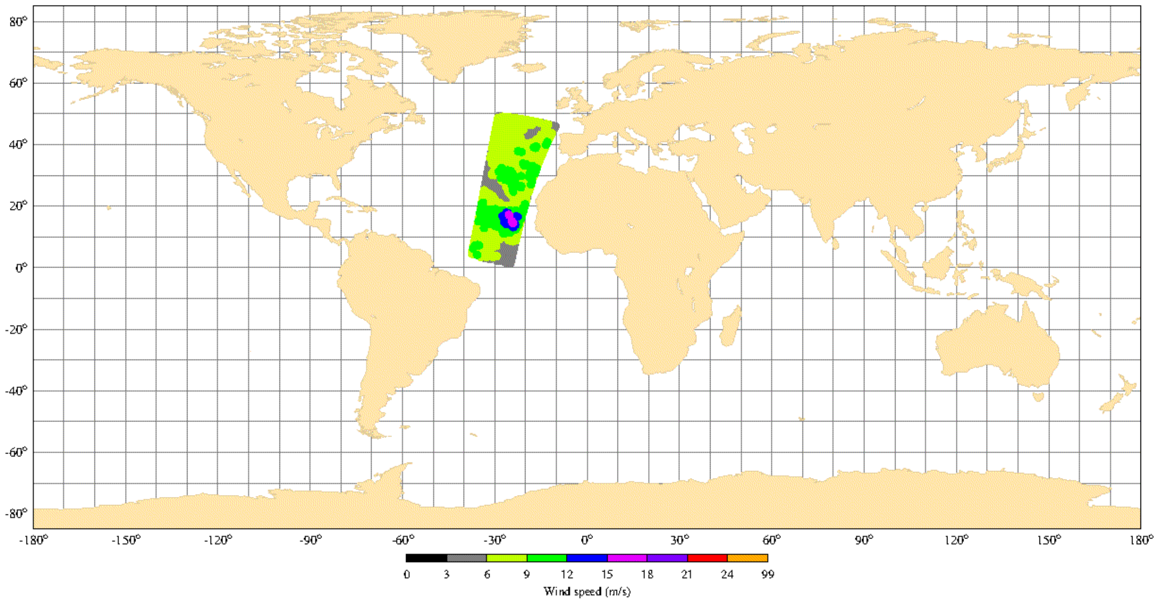

Figure 3.1 Global coverage of the OSCAT test run. Wind speed results for the 50 km product are shown.

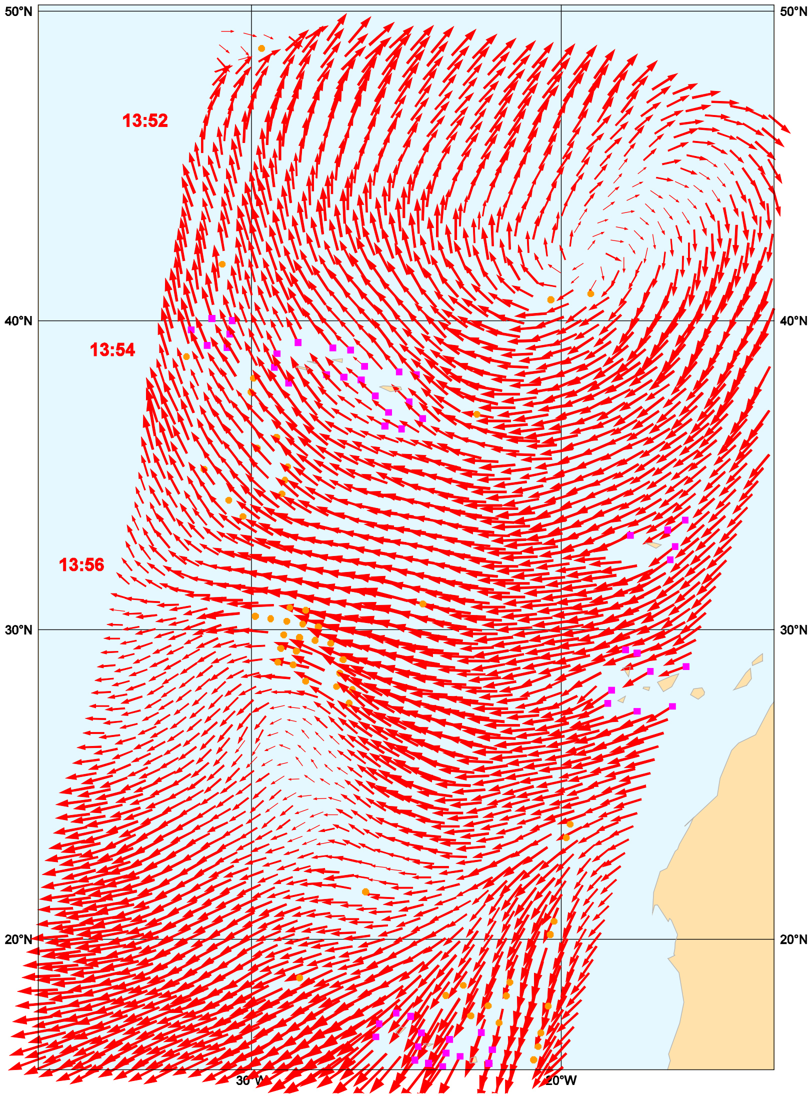

Figure 3.2 Detail plot of the OSCAT test run. Wind vectors for the 50 km product are shown.

Table 3.1 shows one decoded Wind Vector Cell of the resulting output file in NOAA BUFR format and table 3.2 the same WVC in KNMI BUFR format with generic wind section.

1	SATELLITE IDENTIFIER	421.0000	CODE TABLE 1007
2	DIRECTION OF MOTION OF MO	195.0000	DEGREE TRUE
3	SATELLITE SENSOR INDICATO	MISSING	CODE TABLE 2048
4	WIND SCATTEROMETER GEOPHY	9.0000	CODE TABLE 21119
5	SOFTWARE IDENTIFICATION (	1903.0000	NUMERIC
6	CROSS TRACK RESOLUTION	50000.0000	M
7	ALONG TRACK RESOLUTION	50000.0000	M
8	ORBIT NUMBER	12113.0000	NUMERIC
9	YEAR	2012.0000	YEAR
10	MONTH	1.0000	MONTH
11	DAY	6.0000	DAY
12	HOUR	13.0000	HOUR
13	MINUTE	51.0000	MINUTE
14	SECOND	9.0000	SECOND
15	LATITUDE (COARSE ACCURACY	46.1700	DEGREE
16	LONGITUDE (COARSE ACCURAC	-11.4400	DEGREE
17	TIME DIFFERENCE QUALIFIER	5.0000	CODE TABLE 8025
18	SECOND	0.0000	SECOND

```
     19 ALONG TRACK ROW NUMBER                 MISSING NUMERIC
     20 CROSS-TRACK CELL NUMBER                 6.0000 NUMERIC
     21 SEAWINDS WIND VECTOR CELL               0.0000 FLAG TABLE 21109
     22 MODEL WIND DIRECTION AT 1             314.0900 DEGREE TRUE
     23 MODEL WIND SPEED AT 10M                 4.4500 M/S
     24 NUMBER OF VECTOR AMBIGUIT               2.0000 NUMERIC
     25 INDEX OF SELECTED WIND VE               1.0000 NUMERIC
     26 TOTAL NUMBER OF SIGMA-0 M               4.0000 NUMERIC
     27 PROBABILITY OF RAIN                    MISSING NUMERIC
     28 SEAWINDS NOF* RAIN INDEX               13.0000 NUMERIC
     29 INTENSITY OF PRECIPITATIO              MISSING KG/(M**2)S
     30 ATTENUATION CORRECTION OF              MISSING dB
     31 WIND SPEED AT 10 M                      4.5900 M/S
     32 FORMAL UNCERTAINTY IN WIN              MISSING M/S
     33 WIND DIRECTION AT 10 M                327.5000 DEGREE TRUE
     34 FORMAL UNCERTAINTY IN WIN               2.0200 DEGREE TRUE
     35 LIKELIHOOD COMPUTED FOR S               0.6080 NUMERIC
     36 WIND SPEED AT 10 M                      4.1000 M/S
     37 FORMAL UNCERTAINTY IN WIN              MISSING M/S
     38 WIND DIRECTION AT 10 M                170.0000 DEGREE TRUE
     39 FORMAL UNCERTAINTY IN WIN               2.6300 DEGREE TRUE
     40 LIKELIHOOD COMPUTED FOR S               0.3920 NUMERIC
     41 WIND SPEED AT 10 M                     MISSING M/S
     42 FORMAL UNCERTAINTY IN WIN              MISSING M/S
     43 WIND DIRECTION AT 10 M                 MISSING DEGREE TRUE
     44 FORMAL UNCERTAINTY IN WIN              MISSING DEGREE TRUE
     45 LIKELIHOOD COMPUTED FOR S              MISSING NUMERIC
     46 WIND SPEED AT 10 M                     MISSING M/S
     47 FORMAL UNCERTAINTY IN WIN              MISSING M/S
     48 WIND DIRECTION AT 10 M                 MISSING DEGREE TRUE
     49 FORMAL UNCERTAINTY IN WIN              MISSING DEGREE TRUE
     50 LIKELIHOOD COMPUTED FOR S              MISSING NUMERIC
     51 ANTENNA POLARISATION                    0.0000 CODE TABLE 2104
     52 TOTAL NUMBER (WITH RESPEC              58.0000 NUMERIC
     53 BRIGHTNESS TEMPERATURE                124.9000 K
     54 STANDARD DEVIATION BRIGHT               9.1000 K
     55 ANTENNA POLARISATION                    1.0000 CODE TABLE 2104
     56 TOTAL NUMBER (WITH RESPEC              42.0000 NUMERIC
     57 BRIGHTNESS TEMPERATURE                198.0000 K
     58 STANDARD DEVIATION BRIGHT              12.8000 K
     59 NUMBER OF INNER-BEAM SIGM               1.0000 NUMERIC
     60 LATITUDE (COARSE ACCURACY              46.1700 DEGREE
     61 LONGITUDE (COARSE ACCURAC             -11.4200 DEGREE
     62 ATTENUATION CORRECTION ON               0.1700 dB
     63 RADAR LOOK ANGLE                      133.0400 DEGREE
     64 RADAR INCIDENCE ANGLE                  48.9600 DEGREE
     65 ANTENNA POLARISATION                    0.0000 CODE TABLE 2104
     66 SEAWINDS NORMALIZED RADAR             -31.7800 dB
     67 KP VARIANCE COEFFICIENT (               1.0040 NUMERIC
     68 KP VARIANCE COEFFICIENT (               0.0000 NUMERIC
     69 KP VARIANCE COEFFICENT (G             -86.2750 dB
     70 SEAWINDS SIGMA-0 QUALITY                0.0000 FLAG TABLE 21115
     71 SEAWINDS SIGMA-0 MODE                   0.0000 FLAG TABLE 21116
     72 SEAWINDS LAND/ICE SURFACE               0.0000 FLAG TABLE 8018
     73 SIGMA-0 VARIANCE QUALITY               MISSING NUMERIC
     74 NUMBER OF OUTER-BEAM SIGM               1.0000 NUMERIC
     75 LATITUDE (COARSE ACCURACY              46.1900 DEGREE
     76 LONGITUDE (COARSE ACCURAC             -11.4400 DEGREE
     77 ATTENUATION CORRECTION ON               0.1700 dB
     78 RADAR LOOK ANGLE                      154.6300 DEGREE
     79 RADAR INCIDENCE ANGLE                  57.9000 DEGREE
     80 ANTENNA POLARISATION                    1.0000 CODE TABLE 2104
     81 SEAWINDS NORMALIZED RADAR             -26.2300 dB
     82 KP VARIANCE COEFFICIENT (               1.0040 NUMERIC
     83 KP VARIANCE COEFFICIENT (               0.0000 NUMERIC
     84 KP VARIANCE COEFFICENT (G             -77.6000 dB
     85 SEAWINDS SIGMA-0 QUALITY                0.0000 FLAG TABLE 21115
     86 SEAWINDS SIGMA-0 MODE                8192.0000 FLAG TABLE 21116
     87 SEAWINDS LAND/ICE SURFACE               0.0000 FLAG TABLE 8018
     88 SIGMA-0 VARIANCE QUALITY               MISSING NUMERIC
     89 NUMBER OF INNER-BEAM SIGM               1.0000 NUMERIC
     90 LATITUDE (COARSE ACCURACY              46.1700 DEGREE
     91 LONGITUDE (COARSE ACCURAC             -11.4400 DEGREE
     92 ATTENUATION CORRECTION ON               0.1700 dB
```

```
 93 RADAR LOOK ANGLE                     76.6300 DEGREE
 94 RADAR INCIDENCE ANGLE                48.9600 DEGREE
 95 ANTENNA POLARISATION                  0.0000 CODE TABLE 2104
 96 SEAWINDS NORMALIZED RADAR           -33.8500 dB
 97 KP VARIANCE COEFFICIENT (             1.0040 NUMERIC
 98 KP VARIANCE COEFFICIENT (             0.0000 NUMERIC
 99 KP VARIANCE COEFFICENT (G           -86.2560 dB
100 SEAWINDS SIGMA-0 QUALITY              0.0000 FLAG TABLE 21115
101 SEAWINDS SIGMA-0 MODE              4096.0000 FLAG TABLE 21116
102 SEAWINDS LAND/ICE SURFACE             0.0000 FLAG TABLE 8018
103 SIGMA-0 VARIANCE QUALITY             MISSING NUMERIC
104 NUMBER OF OUTER-BEAM SIGM             1.0000 NUMERIC
105 LATITUDE (COARSE ACCURACY            46.1700 DEGREE
106 LONGITUDE (COARSE ACCURAC           -11.4500 DEGREE
107 ATTENUATION CORRECTION ON             0.1700 dB
108 RADAR LOOK ANGLE                     55.5400 DEGREE
109 RADAR INCIDENCE ANGLE                57.9000 DEGREE
110 ANTENNA POLARISATION                  1.0000 CODE TABLE 2104
111 SEAWINDS NORMALIZED RADAR           -32.0100 dB
112 KP VARIANCE COEFFICIENT (             1.0040 NUMERIC
113 KP VARIANCE COEFFICIENT (             0.0000 NUMERIC
114 KP VARIANCE COEFFICENT (G           -78.2850 dB
115 SEAWINDS SIGMA-0 QUALITY              0.0000 FLAG TABLE 21115
116 SEAWINDS SIGMA-0 MODE             12288.0000 FLAG TABLE 21116
117 SEAWINDS LAND/ICE SURFACE             0.0000 FLAG TABLE 8018
118 SIGMA-0 VARIANCE QUALITY             MISSING NUMERIC
```

Table 3.1 Wind Vector Cell in NOAA BUFR format

```
  1 SATELLITE IDENTIFIER                421.0000 CODE TABLE 1007
  2 DIRECTION OF MOTION OF MO           195.0000 DEGREE TRUE
  3 SATELLITE SENSOR INDICATO            MISSING CODE TABLE 2048
  4 WIND SCATTEROMETER GEOPHY             9.0000 CODE TABLE 21119
  5 SOFTWARE IDENTIFICATION (            13.0000 NUMERIC
  6 CROSS TRACK RESOLUTION            50000.0000 M
  7 ALONG TRACK RESOLUTION            50000.0000 M
  8 ORBIT NUMBER                      12113.0000 NUMERIC
  9 YEAR                               2012.0000 YEAR
 10 MONTH                                 1.0000 MONTH
 11 DAY                                   6.0000 DAY
 12 HOUR                                 13.0000 HOUR
 13 MINUTE                               51.0000 MINUTE
 14 SECOND                                9.0000 SECOND
 15 LATITUDE (COARSE ACCURACY            46.1700 DEGREE
 16 LONGITUDE (COARSE ACCURAC           -11.4400 DEGREE
 17 TIME DIFFERENCE QUALIFIER             5.0000 CODE TABLE 8025
 18 SECOND                                0.0000 SECOND
 19 ALONG TRACK ROW NUMBER               MISSING NUMERIC
 20 CROSS-TRACK CELL NUMBER               6.0000 NUMERIC
 21 TOTAL NUMBER OF SIGMA-0 M             4.0000 NUMERIC
 22 PROBABILITY OF RAIN                  MISSING NUMERIC
 23 SEAWINDS NOF* RAIN INDEX             13.0000 NUMERIC
 24 INTENSITY OF PRECIPITATIO            MISSING KG/(M**2)S
 25 ATTENUATION CORRECTION OF            MISSING dB
 26 ANTENNA POLARISATION                  0.0000 CODE TABLE 2104
 27 TOTAL NUMBER (WITH RESPEC            58.0000 NUMERIC
 28 BRIGHTNESS TEMPERATURE              124.9000 K
 29 STANDARD DEVIATION BRIGHT             9.1000 K
 30 ANTENNA POLARISATION                  1.0000 CODE TABLE 2104
 31 TOTAL NUMBER (WITH RESPEC            42.0000 NUMERIC
 32 BRIGHTNESS TEMPERATURE              198.0000 K
 33 STANDARD DEVIATION BRIGHT            12.8000 K
 34 NUMBER OF INNER-BEAM SIGM             1.0000 NUMERIC
 35 LATITUDE (COARSE ACCURACY            46.1700 DEGREE
 36 LONGITUDE (COARSE ACCURAC           -11.4200 DEGREE
 37 ATTENUATION CORRECTION ON             0.1700 dB
 38 RADAR LOOK ANGLE                    133.0400 DEGREE
 39 RADAR INCIDENCE ANGLE                48.9600 DEGREE
 40 ANTENNA POLARISATION                  0.0000 CODE TABLE 2104
 41 SEAWINDS NORMALIZED RADAR           -31.7800 dB
 42 KP VARIANCE COEFFICIENT (             1.0040 NUMERIC
 43 KP VARIANCE COEFFICIENT (             0.0000 NUMERIC
```

```
  44 KP VARIANCE COEFFICENT (G           -86.2750 dB
  45 SEAWINDS SIGMA-0 QUALITY              0.0000 FLAG TABLE 21115
  46 SEAWINDS SIGMA-0 MODE                 0.0000 FLAG TABLE 21116
  47 SEAWINDS LAND/ICE SURFACE             0.0000 FLAG TABLE 8018
  48 SIGMA-0 VARIANCE QUALITY             MISSING NUMERIC
  49 NUMBER OF OUTER-BEAM SIGM             1.0000 NUMERIC
  50 LATITUDE (COARSE ACCURACY            46.1900 DEGREE
  51 LONGITUDE (COARSE ACCURAC           -11.4400 DEGREE
  52 ATTENUATION CORRECTION ON             0.1700 dB
  53 RADAR LOOK ANGLE                    154.6300 DEGREE
  54 RADAR INCIDENCE ANGLE                57.9000 DEGREE
  55 ANTENNA POLARISATION                  1.0000 CODE TABLE 2104
  56 SEAWINDS NORMALIZED RADAR           -26.2300 dB
  57 KP VARIANCE COEFFICIENT (             1.0040 NUMERIC
  58 KP VARIANCE COEFFICIENT (             0.0000 NUMERIC
  59 KP VARIANCE COEFFICENT (G           -77.6000 dB
  60 SEAWINDS SIGMA-0 QUALITY              0.0000 FLAG TABLE 21115
  61 SEAWINDS SIGMA-0 MODE              8192.0000 FLAG TABLE 21116
  62 SEAWINDS LAND/ICE SURFACE             0.0000 FLAG TABLE 8018
  63 SIGMA-0 VARIANCE QUALITY             MISSING NUMERIC
  64 NUMBER OF INNER-BEAM SIGM             1.0000 NUMERIC
  65 LATITUDE (COARSE ACCURACY            46.1700 DEGREE
  66 LONGITUDE (COARSE ACCURAC           -11.4400 DEGREE
  67 ATTENUATION CORRECTION ON             0.1700 dB
  68 RADAR LOOK ANGLE                     76.6300 DEGREE
  69 RADAR INCIDENCE ANGLE                48.9600 DEGREE
  70 ANTENNA POLARISATION                  0.0000 CODE TABLE 2104
  71 SEAWINDS NORMALIZED RADAR           -33.8500 dB
  72 KP VARIANCE COEFFICIENT (             1.0040 NUMERIC
  73 KP VARIANCE COEFFICIENT (             0.0000 NUMERIC
  74 KP VARIANCE COEFFICENT (G           -86.2560 dB
  75 SEAWINDS SIGMA-0 QUALITY              0.0000 FLAG TABLE 21115
  76 SEAWINDS SIGMA-0 MODE              4096.0000 FLAG TABLE 21116
  77 SEAWINDS LAND/ICE SURFACE             0.0000 FLAG TABLE 8018
  78 SIGMA-0 VARIANCE QUALITY             MISSING NUMERIC
  79 NUMBER OF OUTER-BEAM SIGM             1.0000 NUMERIC
  80 LATITUDE (COARSE ACCURACY            46.1700 DEGREE
  81 LONGITUDE (COARSE ACCURAC           -11.4500 DEGREE
  82 ATTENUATION CORRECTION ON             0.1700 dB
  83 RADAR LOOK ANGLE                     55.5400 DEGREE
  84 RADAR INCIDENCE ANGLE                57.9000 DEGREE
  85 ANTENNA POLARISATION                  1.0000 CODE TABLE 2104
  86 SEAWINDS NORMALIZED RADAR           -32.0100 dB
  87 KP VARIANCE COEFFICIENT (             1.0040 NUMERIC
  88 KP VARIANCE COEFFICIENT (             0.0000 NUMERIC
  89 KP VARIANCE COEFFICENT (G           -78.2850 dB
  90 SEAWINDS SIGMA-0 QUALITY              0.0000 FLAG TABLE 21115
  91 SEAWINDS SIGMA-0 MODE             12288.0000 FLAG TABLE 21116
  92 SEAWINDS LAND/ICE SURFACE             0.0000 FLAG TABLE 8018
  93 SIGMA-0 VARIANCE QUALITY             MISSING NUMERIC
  94 SOFTWARE IDENTIFICATION (          1903.0000 NUMERIC
  95 GENERATING APPLICATION               91.0000 CODE TABLE 1032
  96 MODEL WIND SPEED AT 10M               4.4500 M/S
  97 MODEL WIND DIRECTION AT 1           314.0900 DEGREE TRUE
  98 ICE PROBABILITY                      MISSING NUMERIC
  99 ICE AGE ("A" PARAMETER)              MISSING dB
 100 WIND VECTOR CELL QUALITY              0.0000 FLAG TABLE 21155
 101 NUMBER OF VECTOR AMBIGUIT             2.0000 NUMERIC
 102 INDEX OF SELECTED WIND VE             1.0000 NUMERIC
 103 DELAYED DESCRIPTOR REPLIC             4.0000 NUMERIC
 104 WIND SPEED AT 10 M                    4.5900 M/S
 105 WIND DIRECTION AT 10 M              327.5000 DEGREE TRUE
 106 BACKSCATTER DISTANCE                 -2.0000 NUMERIC
 107 LIKELIHOOD COMPUTED FOR S            -0.2160 NUMERIC
 108 WIND SPEED AT 10 M                    4.1000 M/S
 109 WIND DIRECTION AT 10 M              170.0000 DEGREE TRUE
 110 BACKSCATTER DISTANCE                 -2.6000 NUMERIC
 111 LIKELIHOOD COMPUTED FOR S            -0.4070 NUMERIC
 112 WIND SPEED AT 10 M                   MISSING M/S
 113 WIND DIRECTION AT 10 M               MISSING DEGREE TRUE
 114 BACKSCATTER DISTANCE                 MISSING NUMERIC
 115 LIKELIHOOD COMPUTED FOR S            MISSING NUMERIC
 116 WIND SPEED AT 10 M                   MISSING M/S
 117 WIND DIRECTION AT 10 M               MISSING DEGREE TRUE
```

118	BACKSCATTER DISTANCE	MISSING	NUMERIC
119	LIKELIHOOD COMPUTED FOR S	MISSING	NUMERIC

Table 3.2 Wind Vector Cell in KNMI BUFR format with generic wind section

From the plots and tables in this section it is clear that:

- Output can be provided in two BUFR formats.
- The Wind Vector Cell spacing is 50 km, see fields 6 and 7 in the BUFR outputs and figure 3.2.
- The output contains latitude, longitude, time, orbit and node numbers, NWP background wind vector, WVC quality flag, and information on the radar backscatter including σ^0 and K_p data.
- A consistent wind field is obtained which proves that both HDF5 and GRIB data are read successfully.
- The atmospheric attenuations are present in the BUFR output (fields 62, 77, 92 and 107 in the NOAA BUFR format).

The test was re-run with the BUFR output file as input and this results in a new output file with the same wind information. Hence, it is clear that PenWP accepts BUFR data as input as well as HDF5.

Table 3.3 shows what happens when the MLE value exceeds the threshold for Quality Control. The MLE of the fourth wind solution (the selected one by ambiguity removal) is contained in field 49 and has a value of 94.38. This is above the threshold value of 14.4 corresponding to wind speeds close to 9 m/s. The Wind Vector Cell Quality (field 21) has an integer value of 1028, i.e., Fortran bits 10 and 2 are set, corresponding to the flags for KNMI Quality Control and Rain.

21	SEAWINDS WIND VECTOR CELL	1028.0000	FLAG TABLE 21109
22	MODEL WIND DIRECTION AT 1	148.4800	DEGREE TRUE
23	MODEL WIND SPEED AT 10M	6.2700	M/S
24	NUMBER OF VECTOR AMBIGUIT	4.0000	NUMERIC
25	INDEX OF SELECTED WIND VE	4.0000	NUMERIC
26	TOTAL NUMBER OF SIGMA-0 M	4.0000	NUMERIC
27	PROBABILITY OF RAIN	MISSING	NUMERIC
28	SEAWINDS NOF* RAIN INDEX	13.0000	NUMERIC
29	INTENSITY OF PRECIPITATIO	MISSING	KG/(M**2)S
30	ATTENUATION CORRECTION OF	MISSING	dB
31	WIND SPEED AT 10 M	7.7900	M/S
32	FORMAL UNCERTAINTY IN WIN	MISSING	M/S
33	WIND DIRECTION AT 10 M	290.0000	DEGREE TRUE
34	FORMAL UNCERTAINTY IN WIN	14.9900	DEGREE TRUE
35	LIKELIHOOD COMPUTED FOR S	1.0000	NUMERIC
36	WIND SPEED AT 10 M	8.9600	M/S
37	FORMAL UNCERTAINTY IN WIN	MISSING	M/S
38	WIND DIRECTION AT 10 M	112.5000	DEGREE TRUE
39	FORMAL UNCERTAINTY IN WIN	48.4600	DEGREE TRUE
40	LIKELIHOOD COMPUTED FOR S	0.0000	NUMERIC
41	WIND SPEED AT 10 M	7.9600	M/S
42	FORMAL UNCERTAINTY IN WIN	MISSING	M/S
43	WIND DIRECTION AT 10 M	2.5000	DEGREE TRUE
44	FORMAL UNCERTAINTY IN WIN	89.1800	DEGREE TRUE
45	LIKELIHOOD COMPUTED FOR S	0.0000	NUMERIC
46	WIND SPEED AT 10 M	8.5300	M/S
47	FORMAL UNCERTAINTY IN WIN	MISSING	M/S
48	WIND DIRECTION AT 10 M	147.5000	DEGREE TRUE
49	FORMAL UNCERTAINTY IN WIN	94.3800	DEGREE TRUE
50	LIKELIHOOD COMPUTED FOR S	0.0000	NUMERIC

Table 3.3 Part of Wind Vector Cell in NOAA BUFR format, rejected by Quality Control

The processing of the test file (1/8th of a full orbit) takes ~3 seconds on a Linux workstation with an Intel Xeon quad core CPU at 3.20GHz and 8 GB of memory. Hence the OSCAT wind processing can be done easily in near-real time on an affordable computer system.

4 Validation tests

There are several methods to validate scatterometer winds. Scatterometer winds are routinely compared with NWP data and in situ buoy winds in the OSI SAF project. See http://www.knmi.nl/scatterometer/osisaf/ for more information. In the scope of this Test Report, we show the results of a validation study of PenWP winds versus model wind forecasts from the ECMWF model. The correct implementation of the ice screening algorithm is demonstrated in section 4.2.

4.1 PenWP winds versus ECMWF winds

We compared the OSCAT winds from PenWP with ECMWF forecast winds from the operational model (+3 to +21 hours forecasts from the 00 UTC and 12 UTC runs). The OSCAT data are level 2a data version 1.3 from ISRO from 9 and 10 February 2012 (28 orbits), reprocessed with PenWP.

Figure 4.1 shows the collocations of the OSCAT and ECMWF winds. Contoured histograms are shown for wind speed, wind direction and *u* and *v* wind components and after rejection of Quality Controlled (KNMI QC flagged) wind vectors. Note that the ECMWF winds are real 10m winds, whereas the scatterometer winds are equivalent neutral 10m winds, which are on average 0.2 m/s higher. In the wind direction plots, only those wind vectors where the model wind speed is at least 4 m/s are taken into account. The bin sizes for the histograms are 0.5 m/s for wind speed, *u* and *v*, and 2.5° for wind direction.

From the contour plots it is clear that biases are generally low. We obtain wind component standard deviations of 1.33 in *u* and 1.28 in *v* directions. This is comparable to the values we found for SeaWinds in the past: approximately 1.33 for *u* and *v* for the 25-km product and approximately 1.5 for both components for the 100-km product in the same period of the year. We expect that the OSCAT results can be improved by applying better calibration on the backscatter data. This is subject to further study in the NWP SAF and OSI SAF projects.

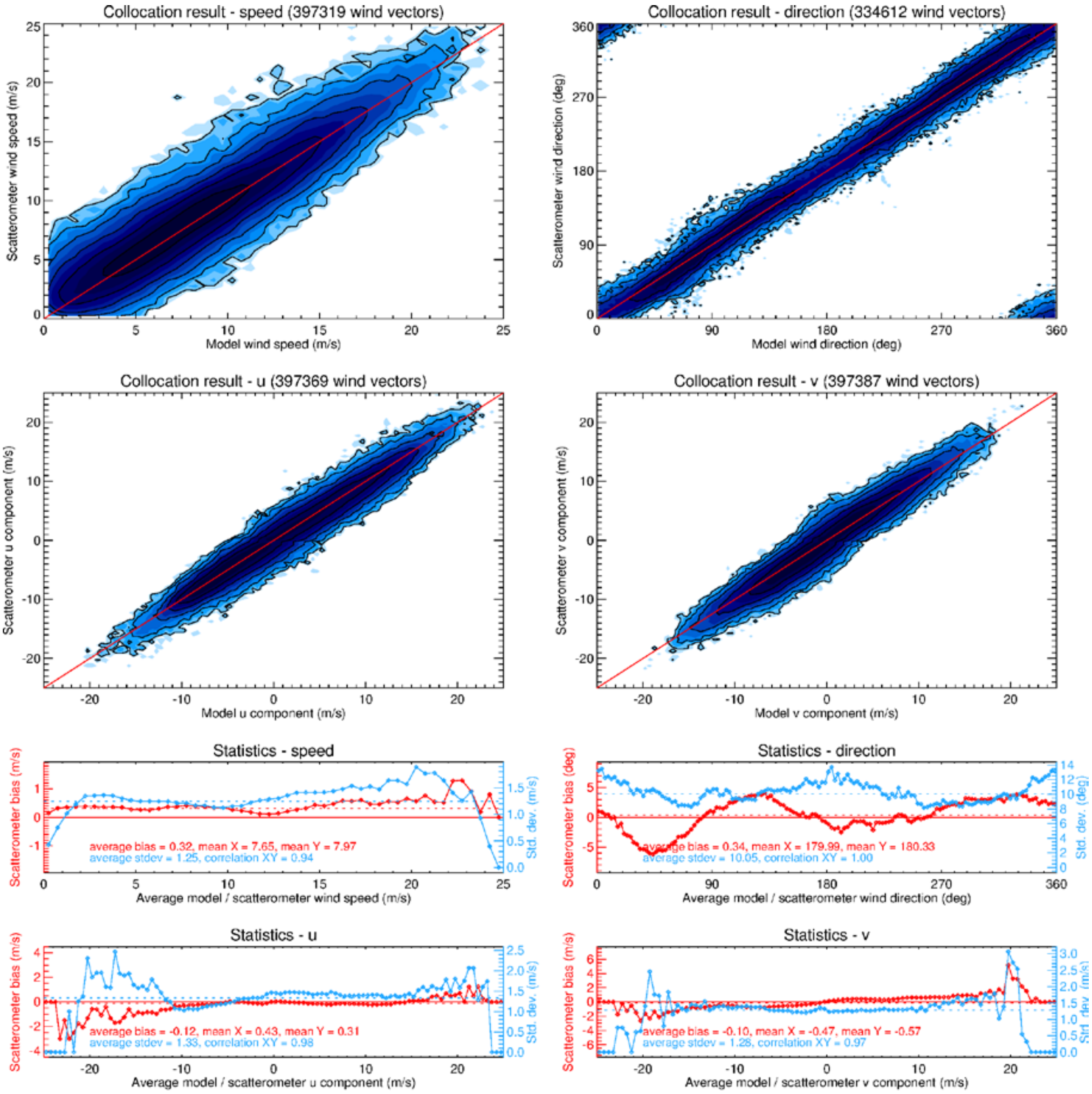

Figure 4.1 Collocation results of Oceansat-2 winds from PenWP and ECMWF forecast winds. Biases and standard deviations in bottom plots are in m/s for wind speed and components, in degrees for wind direction.

4.2 Ice screening test

Figure 4.2 shows the ice maps for North and South poles after processing two days of data. The test data are the same as in the previous section, i.e., 9 and 10 February 2012. Ice maps of the North Pole and South Pole are provided. The blue parts in the maps indicate open water; the black parts correspond to land areas or areas not visited within these two days. The gray scale is a measure of the ice *A*-parameter (albedo). Multi year ice has in general a higher albedo than first year ice, so lighter areas correspond to older ice. In the scope of this report we did not verify the ice extent in detail with other measurements. More information about the ice screening algorithm can be found in [8].

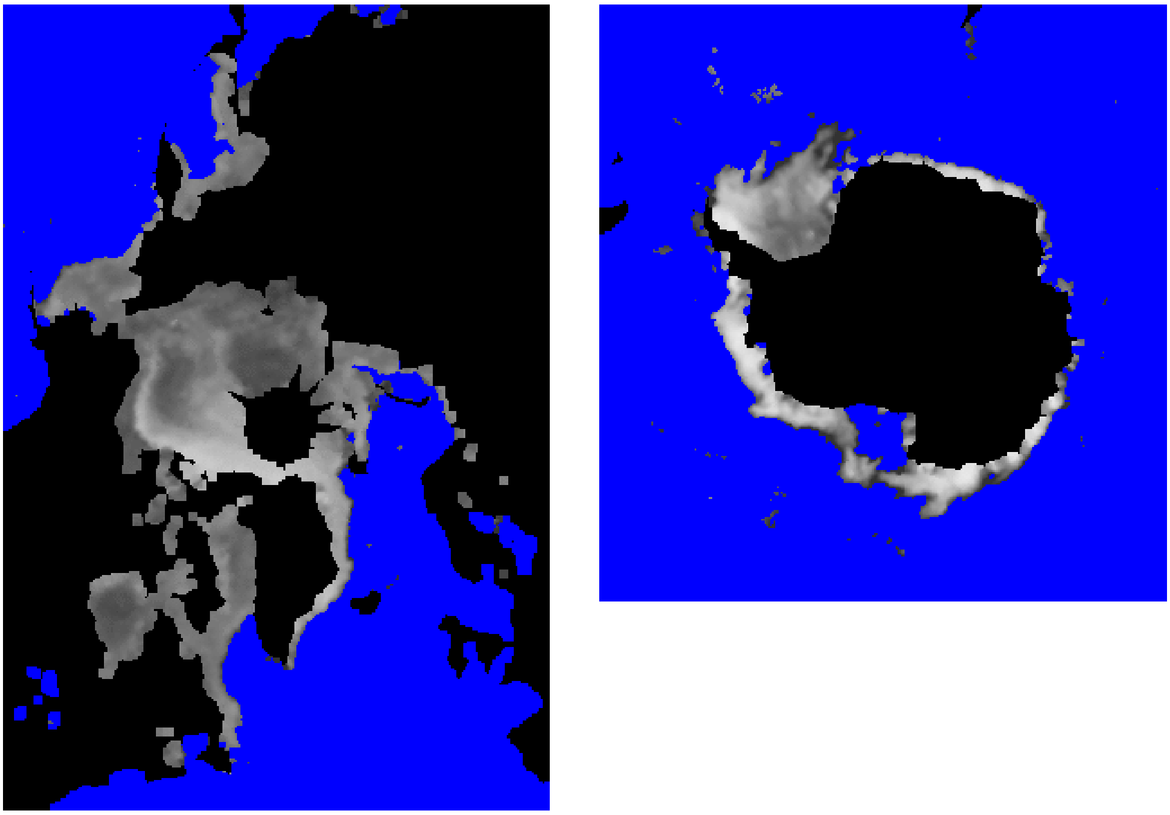

Figure 4.3 PenWP ice maps for North Pole (left) and South Pole (right).

4.3 NBEC test

From version 2.1 onwards PenWP offers the possibility to use Numerical Background Error Correlations (NBECs) in 2DVAR. These lead to a better analysis and hence better agreement with buoy winds and lower setting frequencies of the KNMI QC and VarQC flags, but at the cost of increased processing time. A detailed account is given in [9].

5 Portability tests

The PenWP software package inherits its portability by using strict Fortran 90 code (with a few low level routines for reading and writing binary in C). PenWP is delivered with a complete make system. The Makeoptions include file of genscat takes care of the different settings needed under various platforms. This Makeoptions file is also used for the SeaWinds scatterometer wind processor SDP, The ASCAT wind processor AWDP and the OSCAT scatterometer wind processor OWDP.

The default platform for development is a Linux work station. Different Fortran 90 compilers were used to compile both genscat and PenWP. Table 5.1 provides an overview of the platforms and compilers on which PenWP was tested successfully.

Platform	**Operating system**	**Fortran compiler**
Intel-based workstation	Fedora Linux v.19 3.14.27-100.fc19.x86_64	GNU g95 v0.94 Portland f90 v11.10-0 gfortran v4.8.3-7 Intel Fortran v12.1.4
Linux cluster	Redhat Linux v5.11 2.6.18-404.el5	Portland f90 v11.8-0
Apple MacBook	MacOS X Darwin	GNU gfortran

Table 5.1 Supported platforms and compilers for PenWP.

6 User documentation tests

The user documentation (readme files within the software package and the PenWP user documents, [1], [2], [3]) have been provided to beta testers for review. The beta tester's comments have been implemented in the user documentation. User feedback on the documentation will also be implemented in future versions of the documentation.

References

[1] Verhoef, A., Vogelzang, J., Verspeek, J. and Stoffelen, A., 2017, *PenWP User Manual and Reference Guide*, Report NWPSAF-KN-UD-009, EUMETSAT.

[2] Verhoef, A., Vogelzang, J., Verspeek, J. and Stoffelen, A., 2017, *PenWP Product Specification*, Report NWPSAF-KN-DS-002, EUMETSAT.

[3] Verhoef, A., Vogelzang, J., Verspeek, J. and Stoffelen, A., 2017, *PenWP Top Level Design*, Report NWPSAF-KN-DS-001, EUMETSAT.

[4] Freilich, M.H., 1999, *The SeaWinds Algorithm Theoretical Basis Document*, NASA, (available on http://eospso.gsfc.nasa.gov/sites/default/files/atbd/atbd-sws-01.pdf).

[5] Padia, K, 2010, *Oceansat-2 Scatterometer algorithms for sigma-0, processing and products format*, Version 1.1, April 2010, ISRO.

[6] Liu, D.C., and Nocedal, J., 1989 *On the limited memory BFGS method for large scale optimization methods*, Mathematical Programming, 45, pp. 503-528.

[7] Vogelzang, J., 2007, *Two dimensional variational ambiguity removal (2DVAR)*. Report NWPSAF-KN-TR-004, EUMETSAT. (Available on http://www.knmi.nl/scatterometer/publications/).

[8] Belmonte Rivas, M. and Stoffelen, A, 2011, *New Bayesian algorithm for sea ice detection with QuikSCAT*, IEEE Transactions on Geoscience and Remote Sensing, 49, 6, 1894-1901, doi:10.1109/TGRS.2010.2101608.

[9] Vogelzang, J. and Stoffelen, A, 2016, *ASCAT-derived empirical background error correlations in Ku-band scatterometer wind ambiguity removalQuikSCAT*, Report NWPSAF-KN-TR-25, EUMETSAT. (Available on http://www.knmi.nl/scatterometer/publications/).

Appendix A: Acronyms

Name	Description
ASCAT	Advanced SCATterometer on Metop
AWDP	ASCAT Wind Data Processor
BUFR	Binary Universal Form for the Representation of data
C-band	Radar wavelength at about 5 cm
ECMWF	European Centre for Medium-range Weather Forecasts
EUMETSAT	European Organization for the Exploitation of Meteorological Satellites
genscat	generic scatterometer software routines
KNMI	Koninklijk Nederlands Meteorologisch Instituut (Royal Netherlands Meteorological Institute)
Ku-band	Radar wavelength at about 2 cm
L1b	Level 1b product
LSM	Land Sea Mask
Metop	Meteorological Operational Satellite
MLE	Maximum Likelihood Estimator
MSS	Multiple Solution Scheme
NRCS	Normalized Radar Cross-Section (σ^0)
NWP	Numerical Weather Prediction
OSI	Ocean and Sea Ice
OSCAT	Scatterometer onboard of the Indian Oceansat-2 and ScatSat-1 satellites
OWDP	OSCAT Wind Data Processor
PenWP	Pencil beam Wind Processor
QC	Quality Control
SAF	Satellite Application Facility
SDP	SeaWinds Data Processor
SST	Sea Surface Temperature
WVC	Wind Vector Cell, also called node or cell

Table A.1 List of acronyms.