Implementation of HIRLAM physics in ALADIN

Bent H. Sass

Danish Meteorological Institute

Draft version, September 2005.

Contents

1. Introduction

The first HIRLAM project was established in August 1985. Over the years significant efforts have been spent on the development of physical parameterizations for the HIRLAM model (see section 2). In recent years schemes have been improved and developed with a view to perform at increasing model resolution.

However, several aspects of a true meso-γ scale model physics have not been developed. From a traditional operational point of view this has not been a serious problem in the past since the computer power to run meso-γ scale models for sufficiently large model areas has not been available. However, in the future it is considered necessary to develop all aspects of a meso-γ scale forecasting system. Such a system must include non-hydrostatic dynamics which has not been developed as a HIRLAM core activity. In order to promote progress on the meso-γ scale it has been decided by the HIRLAM community to seek close collaboration with the ALADIN and Météo-France modelling communities which have available a well tested non-hydrostatic model and on-going ambitious plans to develop and implement adequate physics and data-assimilation for a true meso-γ scale forecasting system.

The planned operational meso-γ scale system at Météo-France is named AROME which includes the advanced model physics of the meso-NH model community. These physics are however computationally very expensive. As a consequence it is considered relevant to implement the HIRLAM physics in the common coding environment since some potential of this package has been shown for a very high model resolution. In addition, it is interesting to assess both the potential and the limitations of different physics packages.

The present document provides relevant information related to the implementation of HIRLAM physical parameterizations in IFS/ALADIN. In section 2 a very brief summary is given on HIRLAM physics with references to more detailed information. Section 3 explains some strategic considerations related to the implementation of selected HIRLAM physics. Special implementation issues to be considered for the future are mentioned in section 4. Concluding remarks are given in section 5. The Appendix mentions information on the new subroutines and the modified code.

2. What is HIRLAM physics ?

2.1. Evolution of HIRLAM physics

It has been decided at an early stage of the HIRLAM project to develop and maintain a physical parameterization package which has gone through a significant evolution over the years. The first physics were developed from the assumption that some aspects of accuracy could be relaxed in shortrange forecasting, mainly with regard to radiation computations. An extremely parameterized scheme was designed which could be called every time step. Other physics components (surface scheme, turbulence, condensation and convection) were essentially early schemes of the European Centre (ECMWF).

In the 1990s a need for more advanced schemes occured aiming at a "local weather" oriented forecasting. A highly parameterized radiation scheme was maintained, but improved in many aspects (Savijärvi, 1990; Sass et al., 1994; Undén et al., 2002). More prognostic hydrometeors were aimed at in order to increase the realism of cloud related processes. At first the ideas of Sundqvist were tested using "total cloud condensate" as a new prognostic variable (Sundqvist, 1988; Sundqvist et al., 1989). It turned out that progress was difficult to obtain due to a strong interaction between the physics and the model dynamics which were Eulerian at that time causing "grid point storms" and sometimes numerical instability. Improved methods for coupling physics and dynamics were worked on. For the dynamics part a semi-Lagrangian scheme was improved with better coupling properties than the Eulerian scheme. Different methods to couple physics and dynamics tendencies along a semi-Lagrangian trajectory have been studied. Simultaneously, improved numerics in physics have been developed, e.g. counteracting drastic "on-off" switches in condensation processes. A scheme STRACO (Soft TRAnsition COndensation) paying attention to these features became HIRLAM reference scheme for condensation, cloud and precipitation. This scheme is still essentially based on the microphysics of Sundqvist. A parallel activity resulted in the Rasch-Kristjansson scheme (Rasch and Kristjansson, 1998; Undén et al., 2002) using a prognostic treatment of cloud cover whereas a pseudo-prognostic treatment is used in the STRACO scheme.

Convection parameterization has been an area of continued research , and again two alternative solutions have become available. The convection scheme of STRACO has been improved e.g., with a better cloud ascent model where the entrainment formulation is essential. Moreover triggering and built in scale dependent formulations automatically tends to make convection more inactive at very high model resolution near the grid size of cloud resolving models. The convective closure is still based on humidity convergence principles. A description of the scheme is available (Sass, 2002).

Alternatively, The Rasch-Kristjansson scheme is used together with a version of the Kain-Fritsch convection scheme. This scheme is based on a mass-flux CAPE closure. Also triggering and the cloud ascent model with model resolution aspects are important features of the scheme (Undén et al., 2002).

Both schemes for clouds and condensation have been developed further in recent years, e.g. developing the cloud cover formulation further. For the future, prognostic ice and 3-D precipitation species are being considered for the precipitation schemes.

The link between the convection schemes and the turbulence formulation is rather obvious. The HIRLAM turbulence scheme has been selected and further developed since the later 1990s. It is a special version of the CBR scheme (Cuxart et al., 2000; Undén et al., 2002) using turbulent kinetic energy as a prognostic variable. Specific features of the scheme have been developed, e.g. the mixing length formulation.

The surface scheme used is a special version of ISBA (Integrated Soil Biosphere Atmosphere) parameterization, (Undén et al., 2002). e.g. using a multi-tyle approach. Also the surface flux formulation differs from that developed and used at Météo-France.

In recent years when adequate versions of the hydrostatic semi-Lagrangian grid point model dynamics have become available the HIRLAM physics provide a numerically very stable package which performs well in many aspects.

2.2. First HIRLAM physics implementation choices for the ALADIN model system

When implementing code in a large environment it is common experience that a step-wise implementation has advantages from a technical point of view when compared to the alternative to implement a large package. On the other hand a physics package has some parts which are strongly linked, e.g. the turbulence scheme and the convection parameterization. As regards the surface conditions both HIRLAM and ALADIN use the ISBA scheme in some form and there is an ongoing activity to 'externalise' the surface computations in the future IFS/ALADIN code releases.

As a consequence it has been decided to first implement a coherent but limited physics package in the sense that only the HIRLAM reference versions of the main physics components are implemented (radiation + clouds, condensation, convection, precipitation + turbulence). Alternative options as mentioned above can be implemented later rather easily from a formal coding point of view (see the Appendix). Since the future computations of surface processes will be externalised it may be argued that it is premature to think of implementing any specific HIRLAM surface scheme.

3. Implementation strategy of HIRLAM physics code in IFS/ALADIN

The main strategy chosen can be expressed by the following items:

a) Adapt to existing coding practice in IFS/ALADIN whenever possible.

This implies that the physics are coded in Fortran-90 almost exclusively. A recoding of subroutines to fulfil this practice has taken place since the HIRLAM forecasting system so far used Fortran-77. The individual subroutines are structured like a typical and comparable IFS routine. For example , this means that the interface, purpose, method, author +date, is followed by declarations, interface include blocks and code with explaining comments. Modules are used in a similar way as in IFS/ALADIN. Also the "DR_HOOK" -routine is used twice (at the top and bottom respectively) as is current practice in IFS/ALADIN.

b) Adapt to existing code structures when interfacing HIRLAM physics

So far it has only been necessary to implement subroutines into the directory **ARP** with its sub-directories (see the Appendix). The physics routines with the parameterizations are in "phys_dmn". The HIRLAM constants used in the new physics are in "module". The initialization of the constants is done in the routines of "setup". Function statements related to HIRLAM water vapor saturation computations are in the "function" directory. A new namelist "namhir" has been included into the directory "namelist" The 'uppermost branch' of the code structure related to ARPEGE/ALADIN physics is that associated with the following subroutine calls:

CPG − − − > MF_PHYS − − − > APLPAR − − − > "SUBROUTINES"

The CPG -routine manages gridpoint computations. The physics call sequence is then activated by MF_PHYS and APLPAR. The latter routine calls the different ALADIN physics routines needed in a time step.

When other physics configurations are called in the IFS world , e.g. 'simplied physics' a corresponding physics calling routine APLPARS is activated. In order to adapt to the same philosophy a new routine APLPARH has been created to call the HIRLAM physics (in reality ALADIN routines plus substituted HIRLAM routines).

c) Adapt to the reference system of equations of the new Arome forecasting system.

This is a natural demand since the future code is to be based on this set of equations. It follows as a consequence that various fluxes have to be computed as output from the parameterizations.

Currently there are 6 fluxes associated with convection (PDIFCQ, PDIFCQL, PDIFCQN, PDIFCS, PFCCQL, PFCCQN). The first four of these describe the fluxes of specific humidity, cloud condensate, ice and enthalpy, respectively. These should not involve latent heating proceseses. The last two terms are convective fluxes for the convective condensation of water and ice, respectively. The condensation fluxes of water and ice for stratiform condensation are PFCSQL and PFCSQN respectively. The precipitation fluxes of rain and snow, computed separately for convective and stratiform processes, are respectively PFPLCL, PFPLCN, PFPLSL, and PFPLSN. These are computed as complete 3-D flux fields.

The determination of these fluxes have caused some additional computations. Originally the code produces tendencies of the prognostic variables, and enthalpy is not a prognostic variable in the HIRLAM system. It is possible that slight inconsistencies of the coded enthalpy exist with respect to assumptions elsewhere in the model code.

For radiation the solar and thermal fluxes (PFRSO and PFRTH respectively) are already computed in HIRLAM so no additional complications are involved.

Even though the strategy is followed to produce all the fluxes demanded by the reference system of equations there are no doubt (other) slight inconsistencies between the HIRLAM physics computations and the reference system of equations. These are not known to be serious, but should be investigated later.

d) An initial implementation of HIRLAM physics should involve only a limited part of all available components.

This strategy has already been mentioned. For the surface parameterization an externalised surface physics computations are under preparation in a separate project which makes it unnatural to go for any separate HIRLAM scheme at this stage. For the atmospheric physics it seems natural to limit the implementation as much as possible in order to avoid a large validation effort. On the other hand a coherent physics package should be aimed for as argued below in the next paragraph.

3.1. First experiences with the ALADIN code

The implementation and test of HIRLAM physics in ALADIN has not been as straight forward as might be expected. From the start there has been a quite limited knowledge of the ARPEGE/ALADIN code. It turns out to be not at all trivial to experiment with namelist changes in the existing large coding environment. There seems to be a rather sparse documentation on which parts of the namelists that can be changed safely and which are not easily changed without significant consequences. It seems also that some diagnostic printouts are still lacking in order to tell users if they try impossible namelist combinations or telling what to change to be able to run a meaningful experiment in a given context. As an example of namelist problems it may be stated that experimentation with different settings for the GFL-variable "turbulent kinetic energy" (TKE) may require additional code changes in some subroutines (e.g., in sugfl.F90 and sudyn.F90). If this is not done the TKE variable may not be used at all. The required additional source code modifications are not obvious for the user and seem also against the principle that namelist parameters should be changeable without source code modifications.

Secondly, when implementing HIRLAM physics a recoding from Fortran 77 to Fortran 90 has been involved. Thirdly, some special fluxes which were not originally present in the HIRLAM code had to be computed, as mentioned above, in order to adapt to the presumed implementation of the system of equations.

In addition, some noise and instabilities have been detected when implementing a HIRLAM cloud microphysics scheme together with the original ALADIN turbulence scheme using only water vapor as moisture variable. There is reason to believe that this finding is linked to the lack of coherent physics. This perhaps somewhat surprising behaviour has also been found recently in a similar excersise at Météo-France (Bouyssel et al., 2004). Diffusing moist conserved variables or cloud condensates in addition to specific humidity was found to cure instabilities in their physics combination (Lopez microphysics)

As a consequence this has called for implementing the HIRLAM turbulence scheme to get a more coherent physics package. The HIRLAM turbulence scheme uses 'turbulent kinetic energy' (TKE) as a prognostic variable. Again the implementation of this scheme implies additional work because the default IFS system is not yet fully prepared for TKE as regards ALADIN type of physics implementation.

4. Special coding issues

It is relevant to take note of technical issues related to the coding. Some potential coding weaknesses have been identified. These are briefly mentioned below. Some additional information is available in connection with the description of the individual subroutines in the Appendix.

- Nomenclature for constants: Due to the very many modules in IFS defining constants in global memory it needs to be considered if there is any potental risk that different constants with the same name can lead to unintended global memory overwrite problems or naming confusion. A possible naming convention for HIRLAM constants could be to start all HIRLAM module constants with letter "h" to reduce any such problems if the exist.

Although the HIRLAM fortran routines have been recoded from Fortran77 to Fortan90 in the spirit of IFS coding practice there are some remaining question marks. The importance of following Fortran 90 coding strictly, e.g. as suggested from different type of compiler warnings during compilation, could be considered. Some examples:

- Compilers may suggest that certain input parameters should start with a particular letter.
- Some variables may be declared, but not used in that subroutine.
- Some Fortran 77 features may remain in the code. These features could be easily updated to corresponding recommended Fortran 90 code if considered sufficiently important.

5. Concluding remarks

The present document has given a brief summary on issues related to the implementation of HIRLAM physics in ALADIN. The work has turned out to be more difficult than first anticipated for several reasons which have been mentioned in this document. Although the main blocks of the HIRLAM physics have been implemented in the ARPEGE/ALADIN coding environment a full test and evaluation has not yet been made. Furthermore some adjustments will be needed in the coming IFS/ALADIN cycles before a phasing and implementation in regular code cycles can take place.

6. APPENDIX: Modified and new code

The HIRLAM code is currently (September 2005) implemented to ARPEGE/ALADIN Cycle 29t2. Adjustments caused by code changes in later reference cycles are inherent. The code elements involved in the current HIRLAM implementation are listed below with brief relevant comments. New components are in bold while modified existing components are in italics. Subroutines being a part of turbulence parameterization or related to the treatment of TKE have been written with square brackets.

6.1. Direcory arp/namelist

namhir.h
This is the basic HIRLAM namelist. It is currently very simple in the sense that it contains a logical switch LHIR which may be true or false depending on the use of HIRLAM physics or not. The second namelist parameter LHIRPRT may be used to activate printing inside HIRLAM routines. The third array NHIROPT is an integer array defining options in the sense that each integer in the array corresponds to a physical process. The first number concerns radiation, the next one condensation, the third one turbulence, etc. The integer value assigned points to a given scheme implemented for that process. For example, NHIROPT(3)=1 would normally mean the first (perhaps the only one) turbulence scheme implemented. This method for treating options in a broad sense is very powerfull because implementing a new method for some process does not change the number of namelist parameters, but only the integer numbers assigned.

namfa.h
A name associated with the magnitude of total humidity fluctuations (YFAQVA) is used in the context of a pseudoprognostic field which is included when the HIRLAM condensation is fully implemented.

namgfl.h
Namelist value "YQVA_NL" for pseudo-grognostic field "YQVA" (amplitude of total humidity fluctuations) is introduced in existing namelist NAMGFL

6.2. Direcory arp/module

yomhir.F90
Declaration of the contents of new module YOMHIR with basic HIRLAM namelist parameters defining potential switches for different HIRLAM options.

yomhir1.F90
Declare a number of basic physics constants used in HIRLAM physics.

yomhir2.F90
Declare constants used in connection with HIRLAM condensation schemes.

yomhir3.F90
Declare constants and functions used in connection with HIRLAM radiation scheme.

[**yomhir4.F90**]
Declare constants used in connection with the HIRLAM turbulence scheme.

6.3. Direcory arp/setup

sudim1.F90
Include HIRLAM features in the setup of basic GFL-attributes.

sudyn.F90
Define attributes for new variable "YQVA" with regard to dynamics (e.g. no advection)

sufa.F90
Include small changes due to new field "YQVA" using call to YSUFAD()

sugfl.F90
Include standard calls to DEFINE_GFL_COMP in connction with new pseudo-prognostic field "YQVA"

suhir.F90
Initialize default values for HIRLAM namelist NAMHIR.

suhir1.F90
Initialize basic constants used in HIRLAM physics corresponding to contents of module YOMHIR1

suhir2.F90
Initialize HIRLAM constants used in condensation programs (contents of YOMHIR2)

suhir3.F90
Initialize constants and functions used in HIRLAM radiation scheme (module YOMHIR3)

[**suhir4.F90**]
Initialize constants used in the HIRLAM turbulence scheme (module YOMHIR4)

suphir.F90
Organises calls to the specific HIRLAM initialization routines mentioned above (suhir.F90, suhir1.F90, suhir2.F90, suhir3.F90 and suhir4.F90)

suphy.F90

Top routine for initializing physics constants in modules. After calling subroutines initializing Météo-France - and ECMWF physics it makes a call to suphir.F90 arranging the initialization of HIRLAM constants as mentioned above.

6.4. Direcory arp/function

eshir.**h**
Define HIRLAM functions related to saturation vapor pressure (over water and ice and a mixture)

6.5. Direcory arp/phys_dmn

aconds.**F90**
An interface type of routine preparing for adequate input arguments for the STRACO condensation, convection and precipitation scheme (conds.F90) which is callled as a subroutine inside aconds.F90. The natural input to conds.F90 is in the form of tendencies. For specific humidity the input tendency as a sum from dynamics and turbulence is already provided in aplparh.F90 in a similar way as the current method inherited from aplpar.F90 (Eulerian tendency estimate). For other moisture variables the dynamical tendencies as input to aconds.F90 are currently set to zero. Non-zero values would be relevant in the case of a full implementation of sequential physics. However, tendencies from turbulence may be taken into account in the subroutine condfc.F90 called from aconds.F90 for the case of sequential physics. The output fluxes of moisture and enthalpy from turbulence may be converted to the relevant tendencies that are added to the dynamics tendencies being zero or not.

After the call to the conds.F90 routine a conversion of a number of tendencies to fluxes are done by calling a specific subroutine "tend2flx" for each output flux of asked processes in the output of aconds.F90. The code to compute the total tendencies of temperature, specific humidity, cloud water and cloud ice due to condensation, convection and precipitation is kept in order to be compatible with the old HIRLAM code.

aplparh.**F90**
This routine is the HIRLAM counterpart of aplpar.F90 At present it calls all other ALADIN physics that are not implemented in the HIRLAM part, that is, everything related to the surface including surface fluxes and boundary layer height. Also the gravity wave drag scheme is called. The first HIRLAM physics routine is the radiation parameterization, then the turbulence scheme based on TKE, and last the condensation, convection and precipitation scheme.

[**avcbr**.**F90**]
The HIRLAM outer part of the turbulence scheme (comparable to aconds.F90 in the sense that it contains first preparations of the right form for some input parameters to the turbulence scheme (cbr.F90). After the turbulence scheme has been called , the output tendencies are converted to appropriate output fluxes (again calling the conversion routine tend2flx.F90 for each flux)

cloudcv.F90
The cloud cover routine treating both convective and stratiform cloud cover is based on probability density functions (asymmetric for convection, symmetric for stratiform condensation).

cloudhir.F90
A simplified cloud cover routine is retained for use in the first few time steps in case of missing cloud condensate variables.

condcv.F90
The convection routine where both convective transports with and without condsensation, evaporation or sublimation are computed (6 terms).

condfc.F90
Routine called from aconds.F90 computes input tendencies of temperature, cloud water and cloud ice to the master routine conds.F90 of the STRACO scheme. Both sequential and parallel type of forcing to condensation routines can be handled. This feature may later easily be moved outside the condensation routines.

conds.F90
The master routine for STRACO scheme calling the different subroutines associated with convection, cloud cover, stratiform condensation and precipitation release. Subroutines: nocondcv.F90(routine making it possible to skip call to the convection scheme), condcv.F90 (the convection scheme) , qcampli.F90 (pseudoprognostic treatment of the amplitude of the total humidity fluctuations in the grid box, cloudcv.F90 (cloud cover), condst.F90 (stratiform condensation), prevap.F90 (precipitation release including evaporation/sublimation of precipitation).

condst.F90
Stratiform condensation routine including subgrid scale precipitation.

cpchet.F90
Diagnostic routine involved in computation of statistics for absolute values of physical tendencies. This routine has been modified because it calls subroutine cptend.F90 which computes physical tendencies (also for turbulent kinetic energy which is the new variable included).

hirlamrad.F90
Interface routine preparing input to the HIRLAM radiation scheme radia.F90 . The most important output from radia.F90 is the flux profiles PFRTH (net thermal flux density) and PFRSO (net solar radiation).

Currently hirlamrad.F90 calls the initialization of constants of YOMHIR3 if it is the first time step. Also the simplified cloud cover routine cloudhir.F90 is called for the first 2 time steps.

initaplpar.F90
Initialization routine extended to iniclude initial zero assessment of the flux profile of TKE.

mf_phys.F90

This routine calling different "aplpar"-routines depending on the configuration has become increasingly complex in cycle 29. The present update distinguishing beteen calls to aplpar.F90 or aplparh.F90 depending on the new switch LHIR has contributed further to the complexity. The present update involves also TKE flux and tendency passed to cptend.F90 (see below). There are also a substantial number of calls to subroutine cputqy.F90 (also before the present update). An increased modularisation of mf_phys.F90 seems desirable.

nocondcv.F90
Routine called from conds.F90 to enable that the convection scheme is not active, but only the resolved scale precipitation.

prevap.F90
The precipitation release routine treating both convective and stratiform precipitation. The precipitation fluxes PFPLCL, PFPLCN, PFPLSL, and PFPLSN (convective rain- and snow fluxes, and stratiform rain- and snow fluxes respectively) are output. Also the diabatic flux terms related to phase changes between vapor and liquid or solid are updated. Freezing is treated as evaporation followed by sublimation. Melting is similarly described as sublimation to vapor followed by condensation. This practice should be followed in the model system.

qcampli.F90
Routine which computes an amplitude of variation of total specific humidity within a grid box. This is a pseudoprognostic field.

radia.F90
The HIRLAM radiation scheme producing net flux frofiles of both solar radiation and thermal radiation (called from aradia.F90).

suhirad.F90
Subroutine computing constants of the radiation scheme and special functions used in the scheme.

tend2flx.F90
A simple conversion routine between tendencies and fluxes. This routine is called a number of times due to the different conversions needed in the model physics.

[**tridiag.F90**]
A simple tridiagonal solver routine called for each parameter treated in the HIRLAM turbulence scheme.

[**vcbr.F90**]
The HIRLAM turbulence scheme (called by interface routine avcbr.F90).

6.6. Directory arp/adiab

[*cptend.F90*]
Routine called from mf_phys.F90 to compute tendencies of the model dependent variables as a result of the fluxes due to different processes. This routine has been extended to contain a tendency computation for TKE (not in the CY29 default version) This

tendency only receives a contribution from the turbulence scheme as a source of TKE tendency.

[*cpg.F90*]
This top routine in the call tree to the grid point physics has been modified only marginally to pass information on TKE to mf_phys.F90 and to the diagnostics routine cpg_dia.F90 as mentioned below.

[*cpg_dia.F90*]
TKE-information has been passed from cpg.F90 to cpg_dia.F90. However, the diagnostics inside cpg_dia.F90 and its subroutines have not been completed in view of the preliminary code tested.

[*cputqy.F90*]
Routine which updates prognostic variables due to physics tendencies. It is used in the ALADIN physics part which does not yet include TKE time stepping in Cycle 29. The present update compensates for this limitation and has included TKE updating.

References

Bouyssel, F., Bouteloup, Y., and Marquet, P. (2004). Towards an operational implementation of lopez's prognostic large scale cloud and precipitation scheme in arpege/aladin nwp models. *HIRLAM Workshop report*, pages 56–60.

Cuxart, J., Bougeault, P., and Redelsperger, J.-L. (2000). A turbulence scheme allowing for meso-scale and large-eddy simulations. *Quart. J. Roy. Meteorol. Soc.*, 126:1–30.

Rasch, P. J. and Kristjansson, J. E. (1998). A comparison of the CCM3 model climate using diagnosed and predicted condensate parameterizations. *J. Climate*, 11:1587–1614.

Sass, B. H. (2002). A research version of the STRACO cloud scheme. DMI technical report no 02-10, Danish Meteorological Institute.

Sass, B. H., Rontu, L., and Räisänen, P. (1994). HIRLAM-2 radiation scheme: Documentation and tests. *HIRLAM Tech. Report*, 16:1–43.

Savijärvi, H. (1990). Fast radiation parameterization schemes for mesoscale and short-range forecast models. *J. Appl. Meteor.*, 29:437–447.

Sundqvist, H. (1988). Parameterization of condensation and associated clouds in models for weather prediction and general circulation simulation. *Kluwer Academic*, 1:433–461.

Sundqvist, H., Berge, E., and Kristjánsson, J. E. (1989). Condensation and cloud parameterization studies with a mesoscale numerical weather prediction model. *Mon. Weather Rev.*, 117:1641–1657.

Undén, P., Rontu, L., Calvo, J., Cats, G., Cuxart, J., Eerola, K., Fortelius, C., Garcia-Moya, J. A., Gustafsson, N., Jones, C., Järvenoja, S., Järvinen, H., Lynch, P., McDonald, A., McGrath, R., Navascues, B., Odegaard, V., Rodriguez, E., Rummukainen, M., Rõõm, R., Sattler, K., Savijärvi, H., Sass, B. H., Schreur, B. W., The, H., and Tijm, S. (2002). The HIRLAM-5 scientific documentation. HIRLAM-5 project, december 2002, SMHI Norrköping, Sweden.