1 About Icepack1.1.0

Icepack v.1.1.0 is the thermodynamics column model component of CICE6.0. In this branch, the Icepack code was modify to initialise the model with observation data, to force Icepack with GEM atmospheric forcing and to increase the number of diagnostics. Here is a brief summary of Icepack1.1.0 (for more information, visite the consortium documentation) and the implementations.

1.1 organisation

The code in icepack is organised as follows :

- 1. Scripts, namelists and executables for setting up, compiling and running the model are in the root directory (i.e. Icepack-1.1.0/)
- 2. Codes for the machine environment are located in Icepack-1.1.0/configuration/scripts/machine.
- 3. The driver files are located in Icepack-1.1.0/configuration/drivers. These are scripts ruling the time-stepping scheme, calling different physics modules, etc. The main program script is "icedrv_RUNmod.F90".
- 4. The model physics is located in Icepack-1.1.0/columnphysics.

The time-stepping scheme in CICE is written such that it first solves the equations, then change the time-step and get the new forcing. (at mcgill, the loop is written such that we change step, get the new forcing and then solve for the new solution. It is the exact same, just written differently)

1.2 Compiling

A machine environment arxt42 was added and should work when on the EC network. Alternatively, the test machine can be used. The netcdf, mpi, intel, and librnm need to be mounted.

Setting up an Icepack simulation:

>> ./icepack.setup -c path/to/RUNdirectory -m arxt42

Compiling and running icepack:

>> cd path/to/RUNdirectory and modify the namelist (icepack_in) if needed

>> ./icepack.build

>> ./icepack.run

2 Initialisation

Experiment specifications are set in the namelist "icepack_in" and "icepack.settings" in the run directory indicated in the model setup. "icepack.settings" sets the number of ice and snow layers, ice categories, and vertical columns. The number of columns should be set to 5 to include a column initiated from obsevations. "icepack_in" sets the physics parameterization options (tracers, forcing, thermodynamic schemes, etc).

The model runs 5 different columns with different initial ice conditions (stated in ...). The output diagnositcs for each columns are located in RUNdirectory/logs :

- column 1: initialised ice free (i = 1 in code);
- column 2: initialised uniform thickness, no snow(i = 2);
- column 3: initialised with full ITD and snow (i = 3);
- column 4: initialised with SIMBA data for snow, ice and internal temperature (i = 4);
- column 5: land (i = 5).

2.1 forcing

The forcing data is prepared during the initialisation. The model then runs using the prepared forcing time series.

The routines ruling the forcing are coded in "icedrv_init_forcing.f90". A number of atmospheric and ocean forcing options are available in the namelist.

For the ocean forcing, forcing data or standard values can be used for SSS, uocn, vocn, etc. Here, we use the standard values. The option "oceanmixed_layer" can be used to compute the fluxes at the ice-ocean interface.

2.2 Thermodynamics

The thermodynamics computations is divided in two component:

2.2.1 Therm1: Thermodynamic changes computations

The mass balance and thermodynamics in the column is computed in "icepack-therm-vertical", and the different thickness change contribution are defined. Note that in icepack, the internal temperature profile is not a tracer. At each time step, the intial temperature profile is computed from the enthalpy (a tracer), before solving for the new internal layer profile. The enthalpy is then updated from the solution. Hence, the internal temperature is not a global variable.

2.2.2 Therm2: ITD updates

After soving for the new thermodynamic profiles, the ITD is updated for the new ice thickness. In each category, the new layer boundaries are computed from the mean ice growth. The area and volume of ice transfered between categories is computed by integrating the ITD function between the original and the new boundaries. Once mass and area is transferred, the original boundaries are restored. For more information, see section "transport in thickness space" in the icepack documentation.

At the left boundary (h = 0), the growth of ice in open water is used to determine the amount of frazil than is formed in each time step. If the energy balance is negative (melt), then ice of thickness lower than the melt volum is assume to be lost. The corresponding area is used to decrease the ice concentration.

The lateral melt contribution is computed after the the ITD computation, as a function of the ice concentration and mean floe size.

Note : if the model is ran with only one ice category, then the total ice melt is splited 50-50 between loss of area and thickness. This was commented to have only a thickness reduction in land-fast ice.

2.3 Diagnostics

All diagnostics from the model are output in a single text file. The diagnostics can be sorted using the executable “timeseries.csh”, and plots and re-write the diagnostics as time-series, also stored in “exp_time_series.txt”. Note that diagnostics as still merged in a single column.

There is one diagnostic file for each column. These are located in Icepack_run_RUNdirectory/logs :

- Column 1: ice_diag.icefree (i = 2):
- Column 2: ice_diag.slab (i = 4):
- Column 3: ice_diag.full_ITD (i = 3):
- Column 4: ice_diag.landfast (i = 4):
- Column 5: ice_diag.land (i = 5):

3 Branch implementations

In this local version of Icepack1.1.0 (the thermodynamic column physics of CICE6), the following options were implemented:

- **GEM Forcing option:** Icepack can be forced with atm fluxes derived from the GEM model used within CONCEPTS.

- **Initial condition from observations:** Icepack can be initialized with observed ice thickness, snow thickness and internal temperature from SIMBA buoy data (provided by the Canadian Ice Service).

- **Diagnostics:** A number of variables were added in the diagnostics, such as the internal temperature profiles, ITD functions or thermodynamic growth and melt contributions.

Here is a summary of these implementations.

3.1 GEM Forcing option

The GEM atm forcing is added as a forcing option (in "icedrv_forcing.F90") by setting the atmospheric forcing to "GEM" in the namelist ("icepack_in"). The geographic position of atm forcing is determined by the buoy data used to initialize the model, but is hard-coded to Nain if the model is not initialised using observations. The GEM data is loaded from standard files, located in the forcing path specified in the namelist. The rpn files must additionally be listed in a file named "GEM_atm_forcing.txt" with the forcing data. This file is specified in the GEM_data subroutine, in the Module "icedrv_forcing.F90", and a python script "make_GEM_atm_forcing_file.py" is available to create this list.

All the forcing files are loaded at the initialisation of the model: this is the slowest part of the model, and should be improved. Note also that the GEM data shifts from being hourly (before 2011) to 3-hourly (from 2011 on). This is handled within the new scripts that are manipulating the GEM data into atm fluxes. I.e., if the model time step is shorter than 3h, interpolation in time is made using a simple linear interpolation scheme.

The scripts dealing with the GEM data are the following 5 modules that were added to configuration/driver :

- **gemdrv_get_forcing.F90** : calling driving the package, and 4 scripts that are derived from CONCEPTS module,

- **gemdrv_load_forcing.F90** : preparing the arrays in which the data will be written,

- **gemdrv_read_rpn.F90** : reading the standard files and loading the data,
- **gemdrv_sbc_rpntls.F90** : module containing "timestep-to-date" functions, date string manipulations and time interpolation functions,
- **gemdrv_prepare_forcing.F90**: this is where data manipulations are made to transform the raw data into surface fluxes input in Icepack.

3.2 Initiation from SIMBA observations

In Icepack, different columns with hardcoded initial conditions are simultaneously solved. Here, we added a column which consist in a slab of land-fast ice initialised with SIMBA data, including the initial internal temperature profile.

Note that in the original Icepack model code, the internal ice temperature is computed only to update the enthalpy. As such, a subroutine is added to compute the initial enthalpy associated to the observed temperature profile.

The SIMBA observations are saved in .csv files specified in the namelist. The data is loaded in Icepack and organised in the model layer profile in a new module "icedrv_init_SIMBA.F90".

3.3 Diagnostics

- **Internal temperature and salinity**: To include the internal temperature and salinity profiles as diagnostic, and new subroutine was created to re-compute the temperatures from the enthalpy tracer.
- **ITD variables**: The parameters needed to reconstruct the ITD function g are now included in the diagnostics. A linear ITD profile for g is assumed in each category. The ITD in each category is then defined by for parameter: H_L, H_R (defining the thickness associated to the left and right category bin), g_0, g_1 (determining the line defining the ITD within each normalised thickness category bin, i.e. $g = g_1 h + g_0$, with 0¡h¡1).
- **Thermo contributions**: The thermodynamics of the ice is computed in the file

“icepack_therm_vertical.F90”, and the contributions of the different terms (top, lateral and bottom growth/melt, frazil, snow-ice, etc) are output in the diagnostics. New variables (recognisable by the suffix “_cumul”) a created to store the cumulative contribution of the different terms . The growth terms are in units of m/timestep, and in (m) for the cumulative contribution variables.

- **Albedo**: The albedo of ice, snow and melt ponds, for each thickness category, are include in the list of diagnostics.

- **Category ice area, volume,thickness**: In the original Icepack code, only the mean thickness, area and volume are included in the diagnostics. We added these values for each thickness category.