

# **MODELO DE DERIVA DE TÉMPANOS**

## ***Iceberg drift forecast for NAVAREA VI***

### **REVISIÓN DE LAS CARPETAS Y ARCHIVOS MÁS IMPORTANTES PARA LAS CORRIDAS**

Corrida: LU y VIE a la tarde, luego de la carga del grupo de glaciología

Modelo: NAIS (iceberg Drift Forecasting System) + Python

Que usa el modelo: COPERNICUS+WW3+GFS

- GFS (winds and temperature)
- WW3 (waves and ocean data)
- Copernicus (ocean currents and SST)

Área de cobertura: NAVAREA VI

#### **Fuentes de error**

- 1) todos los témpanos tienen el mismo tamaño= 10 km?
- 2) no se consideran nuevos témpanos que se generan por desprendimiento o fractura de otros existentes
- 3) completa con color amarillo los cuadrados que se quedan sin témpanos. Es solo en el gráfico para que no queden agujeros blancos

Ejemplo de carta pronosticada

<http://www.hidro.gob.ar/smara/MTempanos.asp?op=1>

**Master:** es la carpeta con el modelo original. No tocar jaj

**RC\_Python\_Iceberg\_Drift\_Model-III\_size: Versión de Facu** (viene de ice island, son témpanos más grandes pues está adaptado al HS)

**icebergshn:** Archivo con librerías

**Source:** código fuente  
**system**

**environment.py y config.py**= modificaciones en el código fuente para la implementación de Copernicus

**test:** archivos con los los datos bajados de glaciología con los datos de los iceberg y archivos para cambiar los parámetros en las corridas.

**1-berg-no-ensemble**=para corridas individuales y pruebas

**BergModel.in**= aca se pueden modificar los parámetros del modelo. Además se importante por que desde aca se cambia la ruta si se mueve todo de máquina

**Berg.in**= aca esta la lista de icebergs con posición, tamaño, témpanos especiales, y la fecha-hora de la corrida

**Berg.out**= es el archivo de salida, con todos los icebergs con fecha-horario y las trayectorias pronosticadas. No están todas las salidas del mismo témpano en orden, sino que hay que buscarlo por fecha. La última columna indica el tamaño del témpano, en largo.

Start= año/mes/dia/hora/min/seg

target=fin de la corrida

melt on= prende el derretimiento

**shapefile\_py**: archivos con códigos de Facu

**main.py**= código principal de Facu

Crea un string con la fecha del día y lo emplea para descargar desde el ftp del SHN (con la librería *urllib*) el shapefile de la carta de témpanos más reciente, cuya información se lee con la librería *geopandas*. Se estima la latitud y longitud de los centroides de cada una de las celdas de la carta de témpanos, y se guarda la referencia asociada a la misma en un archivo separado (**referencias.dat**) para su posterior lectura. Se agregan también los témpanos singulares que se informan en el *NAVAREA VI*. Para ello, dentro del **main.py** se llama a otro programa, **navareas\_bergs.py**

Finalmente recopila la información de todos los puntos a correr, y se plasman sobre el archivo **Berg.in**.

**Berg.in**=archivo que usa el modelo para reconocer los parámetros de entrada que necesita.

**berg\_out\_plot** = código que genera los gráficos, osea las cartas. Lee el archivo **berg.out** (la salida del modelo NAIS), y el archivo **referencias.dat**, para reconstruir la deriva de témpanos pronosticada, armar las cartas y representarlas gráficamente. Ejecuta una reclasificación de la carta en base a la deriva de los témpanos y aplica un suavizado en zonas donde es necesario. Además, incluye la representación de la deriva de los témpanos pronosticados en el *NAVAREA VI*.

**navareas\_bergs.py**= lee el *NAVAREA VI* subido en el ftp del SHN, interpreta cuales son los témpanos mencionados, devuelve sus nombres, latitudes, longitudes, y dimensiones. Este programa también realiza un mantenimiento del ftp para mantenerlo en orden y sin acumulación de archivos ya leídos. Va al servidor del servicio para buscar témpanos grandes. Genera el archivo **tempanos\_singulares.txt**.

ftp: **sihn.ddns.net**

user = 'Meteorologia', passwd= 'Meteorologia-2019'

**testita.py**= creo que lee **tempanos\_singulares.txt** **revisar mejor**

## Como hacer una corrida individual con el modelo original

- Limpiar **Berg.out**, para que se copien las nuevas salidas.

- Se corre desde una terminal como el Command Prompt de Windows. Tiene que estar el **environment** en **icebergshn**.

Para activarlo:

**conda activate icebergshn**

- Ir a disco D  
(icebergshn) **C:\Users\Usuario>D:**

### **Para correr BergModel.in**

Primero modificar la fecha de START Y TARGET. No se puede poner una fecha más atrás de 7 días o menos porque va a saltar error (no van a estar los archivos en las bases de datos para descargar). La lista de témpanos tiene que tener la fecha de START en la tercera columna. Si no es el caso por que fueron descargados témpanos de fechas anteriores, borrar todo. Para hacer una prueba dejar uno solo y poner la fecha de START.

**Correr una prueba:** esta prueba usa cualquier témpano si tiene bien las fechas de START y TARGET. Solamente corre el modelo original, no descarga las cartas de glaciología ni tampoco grafica cartas. Da un pronóstico horario. La salida se va a ver en BergOut

- Ir a la carpeta "**system**" que está dentro de **source**:  
**cd**  
**D:\Programacion\python\tempanos\NRC\_Python\_Iceberg\_Drift\_Model-IIL\_size\source\system**
- Luego poner:  
**python main.py**  
**D:\Programacion\python\tempanos\NRC\_Python\_Iceberg\_Drift\_Model-IIL\_size\tests\1-berg-no-ensemble\BergModel.in**

**cd**  
**C:\Proyectos\tempanos\NRC\_Python\_Iceberg\_Drift\_Model-IIL\_size\source\system**

**python main.py**  
**D:\Programacion\python\tempanos\NRC\_Python\_Iceberg\_Drift\_Model-IIL\_size\source\system**

## **Correr una prueba usando bajando los témpanos de glaciología: código Facu**

De glacio se necesitan dos cosas:

-Los témpanos grandes de la NAVAREA se publican en la página todos los días radioavisos nauticos (<http://www.hidro.gov.ar/nautica/cnv.asp>). A mí me llegan por mail, los tengo guardados. Estos datos deben copiarse en el archivo

**tempanos\_singulares.txt** que está en **shapefile\_py** para la corrida correspondiente de ese día. Hay que hacerlo a mano, si no se actualiza el gráfico va a tener témpanos viejos. Hay que borrar y copiar los nuevos a mano.

-También se necesitan los archivos que están, subidos al ftp (<ftp://sihn.ddns.net/>) . El archivo **main.py** de Facu los baja automáticamente y los copia en la carpeta. Tiene la información de la densidad de iceberg de las cartas de glaciología para el día correspondiente.

Para acceder a los datos de glacio ver [Cuestiones a considerar con el Dto de Glaciología](#)

Bien, ahora podemos correr el modelo como antes pero vamos a obtener un output basado en las cartas de glacio. Tener en cuenta que no podemos hacer corridas para días que estén a más de una semana atrás (fallan los datos de Copernicus o Hycom), y además los archivos de ftp se van actualizando y desaparecen los archivos viejos. Tener en cuenta que la fecha de aviso en la NAVAREA tiene que corresponder con el día de la carta (la NAVAREA es diaria y las cartas salen dos veces por semana).

Primero hay que activar el environment usando **conda activate icebergshn** y después ir al disco D poniendo a secas D: (**C:\Users\Usuario>D:**)

Hay que correr el archivo **main.py** de Facundo. Para eso ir a la carpeta **shapefile\_py** en en la consola y de ahí correr **main.py**. Hay que poner bien las fechas en **Berg.in** que está en la carpeta **test**, no puede ser una fecha anterior a siete días.

Pasos:

- **conda activate icebergshn**

-**D:**

-Limpiar **Berg.out**

-Ir a **Berg.in** y modificar la fecha de START Y TARGET. Borrar todos los témpanos que estén en la lista pues esa lista se creará con los datos de glacio.

-ir a la carpeta **shapefile\_py**

**cd**

**D:\Programacion\python\tempanos\NRC\_Python\_Iceberg\_Drift\_Model-IIL\_size\shapefile\_py**

-correr en python el script main de Facu

**python main.py**

**D:\Programacion\python\tempanos\NRC\_Python\_Iceberg\_Drift\_Model-IIL\_size\tests\1-berg-no-ensemble\BergModel.in**

resumen

Correr

**conda activate icebergshn**

**D:**

**cd**

**D:\Programacion\python\tempanos\NRC\_Python\_Iceberg\_Drift\_Model-IIL\_size\shapefile\_py**

correr

```
python main.py
D:\Programacion\python\tempanos\NRC_Python_Iceberg_Drift_Model-III_size\
tests\1-berg-no-ensemble\BergModel.in
```

```
conda activate icebergshn
```

```
cd
Proyectos\tempanos\NRC_Python_Iceberg_Drift_Model-III_size\shapefile_py
```

```
python main.py
C:\Proyectos\tempanos\NRC_Python_Iceberg_Drift_Model-III_size\tests\1-berg
-no-ensemble\BergModel.in
```

## Para graficar con una prueba manual

Primero preparar **Berg.in** y obtener un **Berg.out** con el main.py de Facundo que baja los archivos de la página del servicio con la última carta de témpanos (leer [Correr una prueba usando bajando los témpanos de glaciología: código Facu](#))

Hay que correr **berg\_out\_plot** por partes, hay que correrlo desde el código en visual studio. Poner en la consola **CODE** después de activar el environment y se abren todos los códigos en el Visual Studio.

Para correr hay que correr hasta la línea 20 (dejar afuera lo que dice: #Solo para pruebas, ahora está en línea 21).

Para correr usar **F9**.

Luego se corre a partir de: #%% Separamos de bergout los datos de témpanos singulares, hasta # EN CASO DE PRUEBAS (en línea 73). Si queremos un día específico hay que activar **fecha\_dtime = datetime.date(2022,9,12)** y cambiar la fecha. Si se corre automático activar **#fecha\_dtime = datetime.datetime.today()**. Luego correr hasta el final.

### Como graficar

```
python berg_out_plot.py
D:\Programacion\python\tempanos\NRC_Python_Iceberg_Drift_Model-III_size\
tests\1-berg-no-ensemble\BergModel.in (para prueba no me funciona)
```

## Copernicus:

```
model_name = "GLOBAL_ANALYSIS_FORECAST_PHY_001_024"
```

Cómo acceder manualmente:

Ir al ftp usando la cuenta de Facu (ver archivo **copernicus\_cred.txt**) desde el Explorer:

[ftp://nrt.cmems-du.eu/Core/GLOBAL\\_ANALYSIS\\_FORECAST\\_PHY\\_001\\_024/global-analysis-forecast-phy-001-024/2022/](ftp://nrt.cmems-du.eu/Core/GLOBAL_ANALYSIS_FORECAST_PHY_001_024/global-analysis-forecast-phy-001-024/2022/)

user: fmartinez

pass: QB52Zxe?aU\*TTpD

Ir a la página de copernicus y bajarlo (ahi esta la info de los datos)

[https://resources.marine.copernicus.eu/product-detail/GLOBAL\\_ANALYSIS\\_FORECAST\\_PHY\\_001\\_024/INFORMATION](https://resources.marine.copernicus.eu/product-detail/GLOBAL_ANALYSIS_FORECAST_PHY_001_024/INFORMATION)

La forma automática es: (cambiar la fecha y los límites geográficos)

[https://docs.google.com/document/d/1t9LAnPqG6HqUQMroxwerVCcy\\_E-tLzV3hwMq\\_9iflIQ/edit#](https://docs.google.com/document/d/1t9LAnPqG6HqUQMroxwerVCcy_E-tLzV3hwMq_9iflIQ/edit#)

En el código de **enviromeny.py** se debe agregar adaptar lo siguiente (línea 1056)

```
python -m motuclient --motu https://nrt.cmems-du.eu/motu-web/Motu --service-id
GLOBAL_ANALYSIS_FORECAST_PHY_001_024-TDS --product-id
global-analysis-forecast-phy-001-024 --longitude-min -110 --longitude-max 10 --latitude-min
-80 --latitude-max -30 --date-min "2021-10-07 12:00:00" --date-max "2021-10-16 12:00:00"
--depth-min 0.494 --depth-max 65.8073 --variable thetao --variable uo --variable vo --out-dir
<OUTPUT_DIRECTORY> --out-name <OUTPUT_FILENAME> --user <USERNAME> --pwd
<PASSWORD>
```

## **CAMBIOS EN LA FORMA DE BAJAR LOS DATOS DE COPERNICUS:**

Para bajar los datos de copernicus ver el siguiente archivo en la carpeta

'NRC\_Python\_Iceberg\_Drift\_Model-IIL\_size'

*cambios\_bajar\_datos\_copernicus.txt*

-Cambio el nombre del producto

-Cambio en la forma de agrupar las variables, ahora en vez de estar en el mismo data set, se encuentran divididos en distintos data set

-Por lo tanto cambia la forma de usar motuclient

ahora: se divide en dos dataset

Un dataset para theta

```
run_params1 = f'-m motuclient --motu {config.COP_URL} --service-id {model_name}
--product-id cmems_mod_glo_phy-thetao_anfc_0.083deg_PT6H-i --longitude-min {lon_min}
--longitude-max {lon_max} --latitude-min {lat_min} --latitude-max {lat_max} --date-min
{date_start} --date-max {date_end} --depth-min {depth_min} --depth-max {depth_max}
--variable thetao --out-dir {OUTPUT_DIRECTORY} --out-name {OUTPUT_FILENAME}
--user {USERNAME} --pwd {PASSWORD}'
os.system(str(config.PYTHONEXE + " " + run_params1))
```

Un dataset para uo y vo, pero como es muy grande divido la descarga en 2, una para cada variable

```
run_params2 = f'-m motuclient --motu {config.COP_URL} --service-id {model_name}  
--product-id cmems_mod_glo_phy-cur_anfc_0.083deg_PT6H-i --longitude-min {lon_min}  
--longitude-max {lon_max} --latitude-min {lat_min} --latitude-max {lat_max} --date-min  
{date_start} --date-max {date_end} --depth-min {depth_min} --depth-max {depth_max}  
--variable uo --out-dir {OUTPUT_DIRECTORY} --out-name {OUTPUT_FILENAME} --user  
{USERNAME} --pwd {PASSWORD}'
```

```
os.system(str(config.PYTHONEXE + " " + run_params2))
```

```
run_params3 = f'-m motuclient --motu {config.COP_URL} --service-id {model_name}  
--product-id cmems_mod_glo_phy-cur_anfc_0.083deg_PT6H-i --longitude-min {lon_min}  
--longitude-max {lon_max} --latitude-min {lat_min} --latitude-max {lat_max} --date-min  
{date_start} --date-max {date_end} --depth-min {depth_min} --depth-max {depth_max}  
--variable vo --out-dir {OUTPUT_DIRECTORY} --out-name {OUTPUT_FILENAME} --user  
{USERNAME} --pwd {PASSWORD}'
```

```
os.system(str(config.PYTHONEXE + " " + run_params3))
```

Obviamente ahora la descarga tarda mas por que se dividio en tres, pero bueno es mas facil descargar subset de dataset por periodo/coordenadas de esta manera.

En la pagina recomiendan pasar el sistema de descarga a FTP o OPeNDAP.

## Cuestiones a considerar con el Dto de Glaciología

### Cartas de témpanos y los témpanos singulares de glaciología

Hay que bajarlos regularmente para hacer comparaciones

Antes:

<ftp://sihn.ddns.net/> Usuario: Meteorologia

Contraseña: Meteorologia-2019

Ahora:

Usar el geoportal desde Filezilla- <ftp://geoportal.ddns.net/>

Usuario: meteorologia

Contraseña: Meteorologia-2019

**Pedir que suban los témpanos singulares por que no están desde junio**

hay que ver como actualizar esto en el main.py

Acá está el informe que se sube de NAVAREA VI donde se informan los témpanos grandes de más de 10 millas náuticas.

Radioavisos nauticos

<http://www.hidro.gov.ar/nautica/cnv.asp>

Pero los van a subir al ftp del shn

Aca se suben algunas cartas del glaciología

<https://www.polarview.aq/antarctic>

### **Datos importantes**

Los datos satelitales (ópticos y de radar) se usan para encontrar la posición de los iceberg grandes (se calcula el punto medio) y se informan en la página al día siguiente.

Leer el siguiente tutorial para entender cómo se hacen las cartas de glaciología y que significan los colores:

<https://drive.google.com/drive/folders/186oJtmPsFqB2wyr0mhFOabUps1CEwtKU>

Es decir que cuando sale un informe del SHN se basa en datos satelitales del día anterior. La posición de los iceberg mayores a 2 millas náuticas se informan de manera diaria y pueden servir para la validación del modelo (la ubicación es del día anterior).

Otra cosa importante es que durante el seguimiento, algunos iceberg no se vean en las imágenes por lo que se repite la posición de la última fecha donde se vio. Es decir que habría que identificar cuales son los iceberg que no tienen actualizada la posición por que eso es fuente importante de error en los que son mayores a dos millas náuticas. Si el modelo corriera todos los días este error disminuye porque puede que un iceberg no se vea un día pero al siguiente se vuelva a divisar. En todo caso el input del modelo tiene error si la ubicación está desactualizada. Quizás habría que ver si los datos oceanográficos asimilados de WW+GFS+Copernicus se deben tomar de las corridas del día anterior.

Por otro lado, para los iceberg que tienen menos de dos millas, no se dan las posiciones individualizadas sino que se cuentan cuantos iceberg hay en cada rectángulo y se le asigna un color dependiendo de la densidad de icebergs. No se siguen sino que se cuenta la existencia de más o menos témpanos. Para estos témpanos solo se hace el recuento dos veces por semana por lo que no sería tan importante si un témpano no es visible en el radar pues es una estimación de la densidad aproximadamente (y muy a ojo). Es decir el input no estaría tan mal porque no es un seguimiento individualizado (no se sigue un témpano en particular), sino más bien es un promedio de varios témpanos.

### **Importante para revisar**

Ver que hace el scrip de python, es decir como calcula las posiciones de los témpanos chicos (donde la carta dice riesgo de témpanos). Las posiciones no son reales sino son el centroide.

Ver de cuanto es la grilla del modelo.

## **SERVER**

ssh facundo@sihn.ddns.net -p 5501

pass: dni



una vez adentro

ssh modelos-olas@192.168.1.117

pass: fTDvSFaEST63yfcF

(Para mantener organizado el server solo trabajen en la carpeta home del usuario)

**Acceso servidor olas:**

ssh [facundo@sihn.ddns.net](mailto:facundo@sihn.ddns.net) -p 5501

pass: 37987578

**Acceso servidor témpanos:**

ssh modelos-olas@192.168.1.117

pass: fTDvSFaEST63yfcF

**Modo administrador:**

sudo su

pass: fTDvSFaEST63yfcF