

Tutorial 04 LiveCROCO: Simulación con Mareas

1. Propósito

En este tutorial revisaremos como realizar una simulación del dominio de Benguela incluyendo el efecto de la marea en los bordes, los archivos importantes para este tipo de simulaciones, y la validación de esta.

2. Creando el directorio de trabajo MAREAS

El ejemplo más sencillo de CROCO es la configuración llamada BENGUELA_LR que corresponde a un dominio de la zona de surgencia de Benguela de baja resolución (de ahí el LR : *Low Resolution*). Esta configuración es la que viene por defecto en el código de CROCO y lo que haremos es similar a lo descrito en Penven et al. (2001).

El primer paso es editar el archivo `create_config.bash` con las instrucciones para crear un nuevo directorio de trabajo que llamaremos **MAREAS**

```
1 cd Desktop
2 nano create_config.bash
```

Ahora tiene que modificar esta sección para el nombre de su configuración

```
1 #
2 # Name of the configuration directory defined by the user
3 #
4 MY_CONFIG_NAME='MAREAS'
5 #
6 #
```

y después ejecute la instrucción

```
1 ./create_config.bash
```

3. Configuración REGIONAL con MAREAS

Para este tutorial agregaremos la opción de mareas a la física del modelo.

3.1. MAREAS

Nos cambiamos al directorio **MAREAS**

```
1 cd MAREAS
```

En el archivo `cppdefs.h` activamos las mareas modificando el archivo a

```
1 /* Open Boundary Conditions */
2 # define TIDES
```

activar esta opción tiene mas consecuencias, como se puede ver más abajo en el código

```
/* Open Boundary Conditions */
# ifdef TIDES
# define SSH_TIDES
```

```

# define UV_TIDES
# define POT_TIDES
# undef TIDES_MAS
# ifndef UV_TIDES
# define OBC_REDUCED_PHYSICS
# endif
# define TIDERAMP
# endif
# define OBC_M2CHARACT
# undef OBC_M2ORLANSKI
# define OBC_M3ORLANSKI
# define OBC_TORLANSKI
# undef OBC_M2SPECIFIED
# undef OBC_M3SPECIFIED
# undef OBC_TSPECIFIED

```

Para comenzar, destacaremos la opción

```
1 # define TIDERAMP
```

esta implica que la onda de marea será atenuada por los primeros tres días, con el objetivo de evitar inestabilidades numéricas en la simulación.

4. Mareas - param.h

Dentro del *param.h* el definir la opción **TIDES** implica que

```

!-----
! Tides
!-----
!
#if defined SSH_TIDES || defined UV_TIDES
    integer Ntides           ! Number of tides
                           ! ===== == =====
# if defined IGW
    parameter (Ntides=1)
# else
    parameter (Ntides=10)
# endif
#endif
#endif

```

es decir que nuestro modelo usará 10 componentes de marea. Estas están definidos en la sección 5 de **croco-tools_param.m**

```

1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2 %
3 % 5 - Parameters for tidal forcing
4 %
5 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
6 %
7 % TPXO file name (TPX06 or TPX07)
8 %
9 tidename=[DATADIR,'TPX07/TPX07.nc'];
10 %
11 % Self-Attraction and Loading GOT99.2 file name
12 %
13 sal_tides=1;
14 salname=[DATADIR,'GOT99.2/GOT99_SAL.nc'];
15 %
16 % Number of tides component to process
17 %
18 Ntides=10;
19 %
20 % Chose order from the rank in the TPXO file :
21 % "M2 S2 N2 K2 K1 O1 P1 Q1 Mf Mm"
22 % " 1 2 3 4 5 6 7 8 9 10"
23 %
24 tidalrank=[1 2 3 4 5 6 7 8 9 10];
25 %
26 % Compare with tidegauge observations
27 %
28 lon0 = 18.37; % Example:
29 lat0 = -33.91; % Cape Town location
30 Z0 = 1; % Mean depth of tide gauge
31 %
32 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

Es decir, para el forzamiento de mareas estaremos usando la solución globales TPX07 (Egbert y Erofeeva, 2002).

La opción `sal.tides=1` implica una corrección del 5% de la amplitud, con una desfase de 180° con la marea local o aún más en zonas costeras.

```

1 https://www.esr.org/data-products/antarctic\_tg\_database/ocean-tide-and-ocean-tide-loading/

```

5. Compilando CROCO

A continuación compilamos el modelo

```

1 ./jobcomp

```

6. Los archivos de entrada

Los archivos de entrada que leerá el ejecutable `croco` los crearemos con la herramienta `CROCO_TOOLS`. Esta depende principalmente de dos archivos, el `oct.start.m` y el `crocotools.param.m`

6.1. MAREAS - Usando Octave

Para crear los archivos de entrada usando Octave, primero tenemos que cargar el programa usando

```
1 octave-cli
```

Dentro de Octave las instrucciones a usar, desde el directorio de trabajo **MAREAS** son:

```
1 oct_start
2 make_grid
3 make_forcing
4 make_clim
```

en este caso, agregaremos la instrucción

```
1 make_tides
```

y en el terminal veremos

```
1 mkdir: cannot create directory '/home/livecroco/Desktop/MAREAS/CROCO_FILES/':
2 File exists
3 Start date for nodal correction : 1-Jan-2005
4 Reading CROCO grid parameters ...
5 Tidal components : M2 S2 N2 K2 K1 O1 P1 Q1 Mf Mm
6 Processing tide : 1 of 10
7   ssh...
8 Getting ssh_r for time index 1
9 Getting ssh_i for time index 1
10  u...
11 Getting u_r for time index 1
12 Getting u_i for time index 1
13  v...
14 Getting v_r for time index 1
15 Getting v_i for time index 1
16 Convert to tidal ellipse parameters...
17 Process equilibrium tidal potential...
18 Process tidal loading and self-attraction potential...
19 Get total tidal potential...
```

Obteniendo gráficos de la amplitud y fase de las componentes usadas, por ejemplo para K1 Figs. 1 y 2

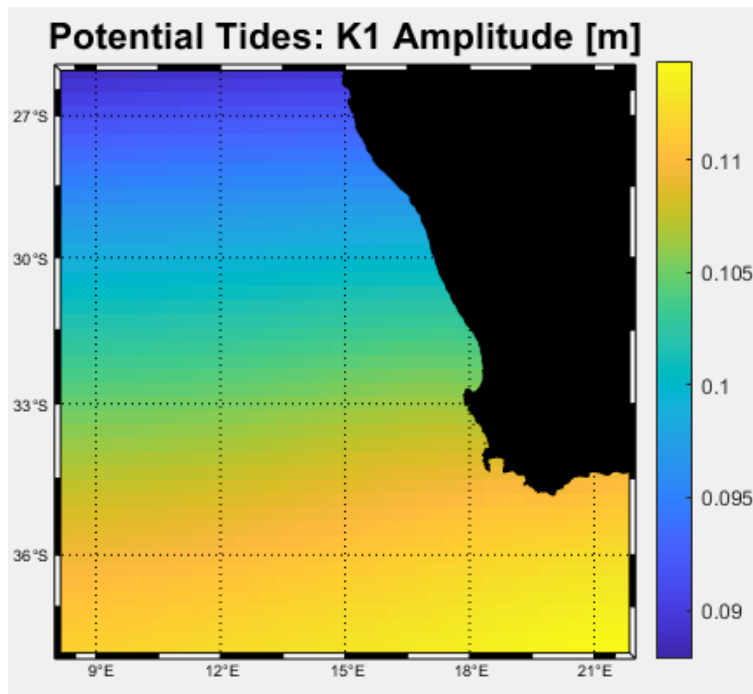


Figura 1: Amplitud de la componente K1

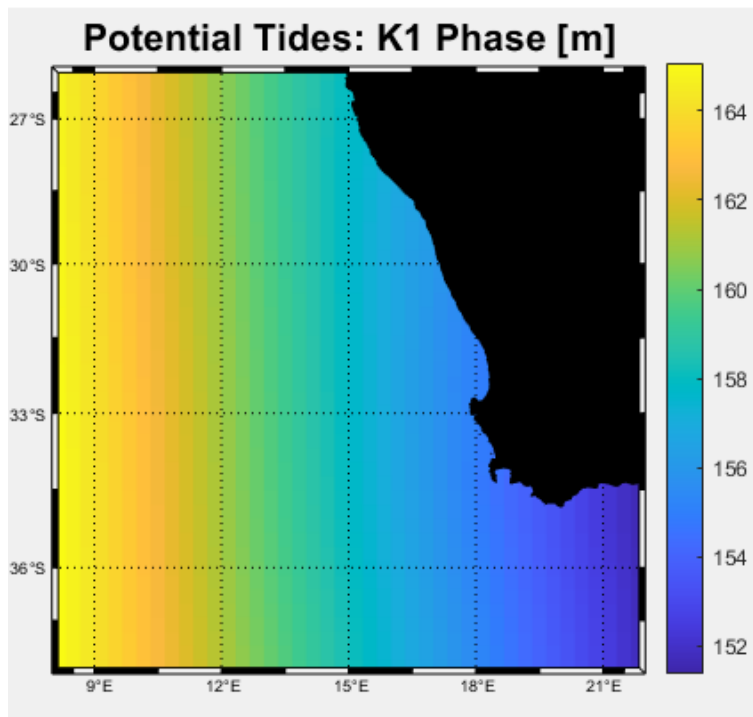


Figura 2: Fase de la componente K1

y una serie de tiempo dentro del dominio (Fig. 3)

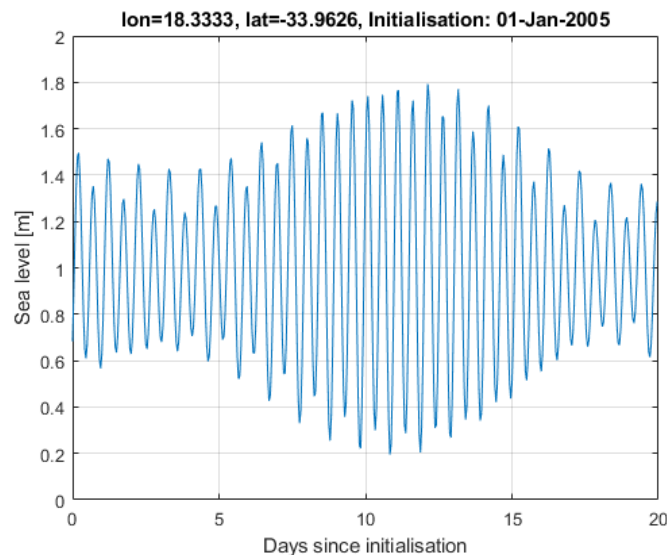


Figura 3: Serie de tiempo en un punto del dominio

Esto agrega las variables asociadas a la marea en el archivo **croco_frc.nc**

Los archivos que obtenga deben ser iguales a los que se encuentran en

```
1 http://mosa.dgeo.udec.cl/LiveCROCO/Tutorial04/ArchivosIniciales/
```

si tuvo problemas con esta etapa, copie esos archivos en el directorio **CROCO_FILES** para avanzar a la siguiente sección.

7. MAREAS - croco.in

Para el análisis de las mareas en nuestro modelo es importante modificar el **croco.in** para grabar las salidas del archivo **HIS** cada 1 hora. Como el paso de tiempo de esta configuración es de 3600 segundos, grabaremos las salida entonces cada 1 paso de tiempo

```
1 history: LDEFHIS, NWRT, NRPFHIS / filename
2         T      1      0
3         CROCO_FILES/croco_his.nc
```

Una vez hecho esto, lanzamos la simulación

```
1 ./croco croco.in
```

Una vez que la simulación termine exitosamente, encontraremos en el directorio **CROCO_FILES** los siguientes archivos de salida

```
1 croco_avg.nc
2 croco_his.nc
3 croco_rst.nc
```

donde el archivo de nuestro interés será el **croco_his.nc**. Sin embargo, como usamos la opción **TIDERAMP** en el **cppdefs.h** el nivel del mar de los primeros tres días es irreal.

Para corregir esto edite de nuevo el **cppdefs.h** y configure

```
1 # undef TIDERAMP
```

luego recompile el modelo con

```
1 ./jobcomp
```

reemplace el archivo **INI**

```
1 cp croco_rst.nc croco_ini.nc
```

y lance de nuevo la simulación

```
1 ./croco croco.in
```

8. Validación de la simulación de mareas

8.1. Análisis de la simulación

Analizaremos nuestra simulación y observaciones usando el programa **t_tide** (Pawlowicz et al., 2002). Una versión de **t_tide** viene ya incluida en **CROCO_TOOLS**. Para esto usaremos el siguiente código

```
1 https://raw.githubusercontent.com/AndresSepulveda/CROCO\_plots/master/t\_tide/oct\_analyze\_t\_tide.m
```

que fue adaptado de

```
1 https://isabeljalonrojas.com/analisis-de-armonicos-de-marea-astronomica-con-t\_tide/
```

8.2. Análisis de observaciones

Primero obtendremos mediciones para su zona del Centro de Variación del Nivel del Mar de la Universidad de Hawaii

<https://uhslc.soest.hawaii.edu/>

o en el sitio web antiguo

<http://uhslc.soest.hawaii.edu/data/>

Por ejemplo, usamos los datos horarios de Simon's Town, Sud Africa (latitud -34.18300 , longitud 18.43300) válidos entre 1959-04-24 y 2020-08-31.

```
1 wget http://uhslc.soest.hawaii.edu/data/nc/fdh/OS_UH-FDH221_20170628_R.nc
```

El archivo **analyze_t_tide.m** está configurado para usar sólo 720h, pues es el mismo periodo que tendremos disponible del modelo numérico.

```
1 octave-cli  
2 >> oct_analyze_t_tide
```

que debería entregar

```
1 Add the paths of the different toolboxes
2 Points used: 721 of 721
3 percent of var residual after lsqfit/var original: 2.72 %
4 Phases at central time
5 Using nonlinear bootstrapped error estimates
6 Generating prediction without nodal corrections, SNR is 2.000000
7 percent of var residual after synthesis/var original: 2.82 %
8 Points used: 721 of 721
9 percent of var residual after lsqfit/var original: 2.72 %
10 Phases at central time
11 Using nonlinear bootstrapped error estimates
12 Generating prediction without nodal corrections, SNR is 2.000000
13 percent of var residual after synthesis/var original: 2.82 %
14
15 TOTAL =
16
17 4×30 char array
18
19 'M2 0.0805 0.502 250.504'
20 'S2 0.0833 0.295 215.455'
21 'N2 0.079 0.093 36.242'
22 'K1 0.0418 0.054 310.811'
23
24 Nombre Frecuencia Amplitud Fase
25 Factor de Forma (F)
26
27 F =
28
29 0.0941
30
31 Marea Semidiurna
32 >>
```

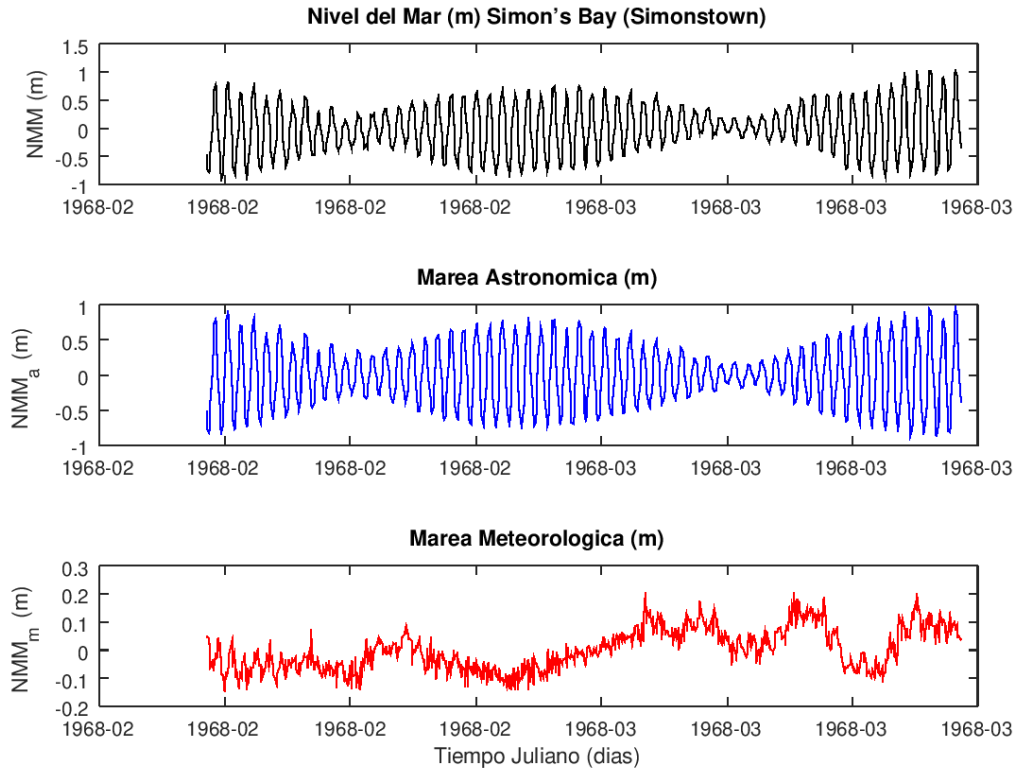



Figura 4: Serie de tiempo para Simon's Town, África de Sur. Observaciones.

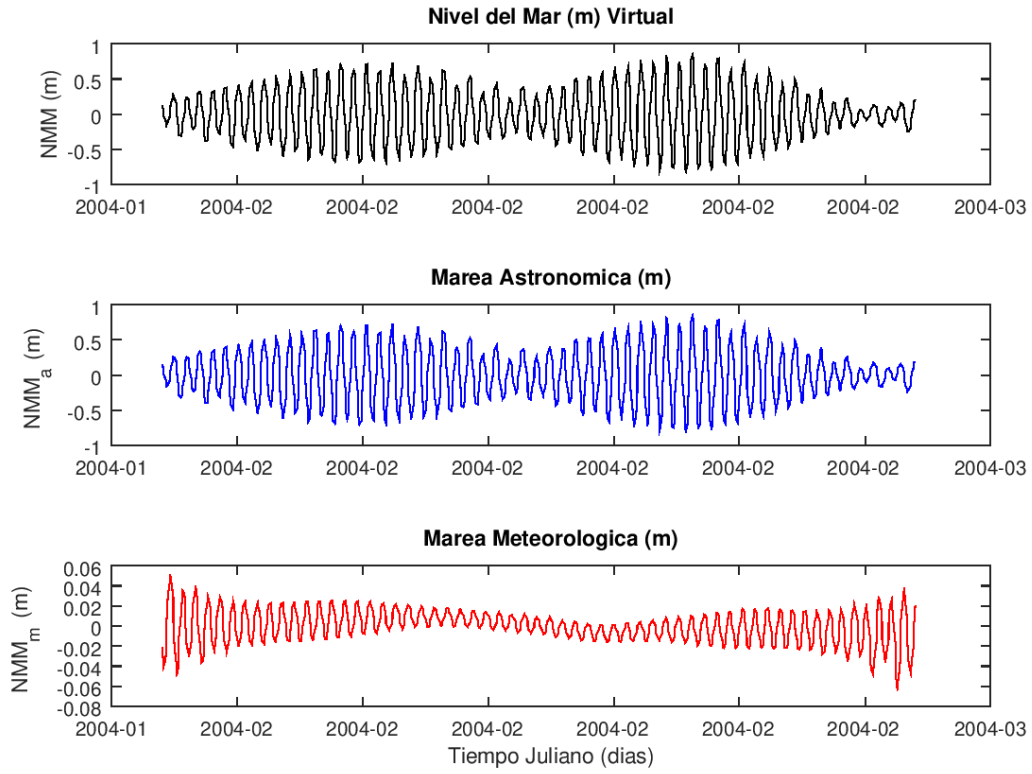


Figura 6: Serie de tiempo para Simon's Town, África de Sur. Modelo

observaciones, incluyendo la incerteza respectiva. Use a lo más 2 cifras significativas. Si analiza varias estaciones puede hacer un gráfico como la Figura 3 de Artal et al. (2019).

Si le interesan los problemas inversos puede intentar usar el código de Leffler & Jay (2009) ó Codiga (2011), que contienen una estimación de parámetros más robusta. Otros códigos (NS_TIDE) permiten comparar armónicos no estacionarios (Matte et al. 2013).

9. Trabajo Avanzado

Realice una simulación para un dominio de su interés y analice la señal de marea.

10. Conclusión

En este tutorial aprendió mas detalles de los archivos **cppdefs.h**, **param.h**, y **croco.in**, y las modificaciones que hay que hacer para hacer una simulación de mareas, así como aspectos de la validación de los resultados.

Para más información:

Andrés Sepúlveda (asepulveda@dgeo.udec.cl)

Contribuciones de:

Marcela Contreras

Bruno Herrera

Scarlett Moraga

Mauro Santiago

Si le sirvió este tutorial mande una postal a:

Dr. Andrés Sepúlveda

Departamento de Geofísica

Casilla 160-C

Correo 3

Concepción

Chile

11. Referencias

- Artal, O., Pizarro, O., & Sepúlveda, H. H. (2019). The impact of spring-neap tidal-stream cycles in tidal energy assessments in the Chilean Inland Sea. *Renewable Energy*, 139, 496-506.
- Codiga, D.L., 2011. Unified Tidal Analysis and Prediction Using the UTide Octave Functions. Technical Report 2011-01. Graduate School of Oceanography, University of Rhode Island, Narragansett, RI. 59pp. <https://github.com/wesleybowman/UTide>
- Egbert, Gary D., and Svetlana Y. Erofeeva. "Efficient inverse modeling of barotropic ocean tides." *Journal of Atmospheric and Oceanic Technology* 19.2 (2002): 183-204.
- Leffler, K. E., & Jay, D. A. (2009). Enhancing tidal harmonic analysis: Robust (hybrid L1/L2) solutions. *Continental Shelf Research*, 29(1), 78-88.
- Matte, P., Jay, D.A., Zaron, E.D., 2013. Adaptation of classical tidal harmonic analysis to nonstationary tides, with application to river tides. *J. Atmos. Ocean. Technol.* 30, 569{589. doi:10.1175/JTECH-D-12-00016.
- Pawlowicz, R., Beardsley, B., & Lentz, S. (2002). Classical tidal harmonic analysis including error estimates in Octave using T_TIDE. *Computers & Geosciences*, 28(8), 929-937.
- Penven, P., Roy, C., Brundrit, G. B., De Verdière, A. C., Fréon, P., Johnson, A. S., Lutjeharms J. R. E. & Shillington, F. A. (2001). A regional hydrodynamic model of upwelling in the Southern Benguela. *South African Journal of Science*, 97(11-12), 472-475.