

Manual de wind3D

Tabla de contenidos

Introducción.....	1
wind3D.....	2
Lenguaje de programación.....	2
Entorno operativo.....	2
Listado de ficheros.....	2
Diagrama de flujo.....	3
Instalación.....	3
Uso.....	3
Fichero de configuración.....	4
Fichero de malla.....	4
Fichero de viento.....	4
Fichero estación.....	5
Fichero de topografía.....	5
Ficheros de salida.....	5
wind3D_netcdf_converter.py.....	6
Lenguaje de programación.....	6
Entorno operativo.....	6
Listado de ficheros.....	6
Diagrama de flujo.....	6
Uso.....	6
Fichero de salida.....	6
wind3D_wizard.py.....	7
Lenguaje de programación.....	7
Entorno operativo.....	7
Listado de ficheros.....	7
Diagrama de flujo.....	7
Fichero de salida.....	7
Referencias.....	8

Introducción

Este documento es un breve manual que describe el uso del programa **wind3D** y los formatos de los ficheros que se usan en sus entradas de datos y salidas de resultados.

El código **wind3D** es un código de simulación de campos de viento 3D sobre orografía irregular con un modelo adaptativo de masa consistente.

El código puede usar datos de los modelos predictivos WRF y HARMONIE, así como datos de estaciones de medida experimentales

El código está compuesto por tres módulos.

wind3D es el código en C que se ejecuta desde la línea de comandos y leyendo un fichero de configuración con todos los datos necesarios hace la simulación del campo de viento.

wind3D_netcdf_converter.py es un código en python, por lo tanto código interpretado (no hay ejecutable) que sirve para convertir los ficheros netCDF de HARMONIE y WRF en los ficheros de viento que usa el programa Wind3D.

wind3D_wizard .py es un código en python, por lo tanto código interpretado (no hay ejecutable), que ofrece una interfaz gráfica para crear o modificar el fichero de configuración, convertir los ficheros netCDF y ejecutar el programa Wind3D.

El código ha sido el resultado de la investigación en este campo. Los artículos de investigación que sientan las bases del programa y dónde se muestran el uso del programa se pueden ver en la sección referencias en este mismo documento.

wind3D

Lenguaje de programación

El lenguaje de programación de este programa es C (estándar 99). Se ha compilado con el compilador de GNU.

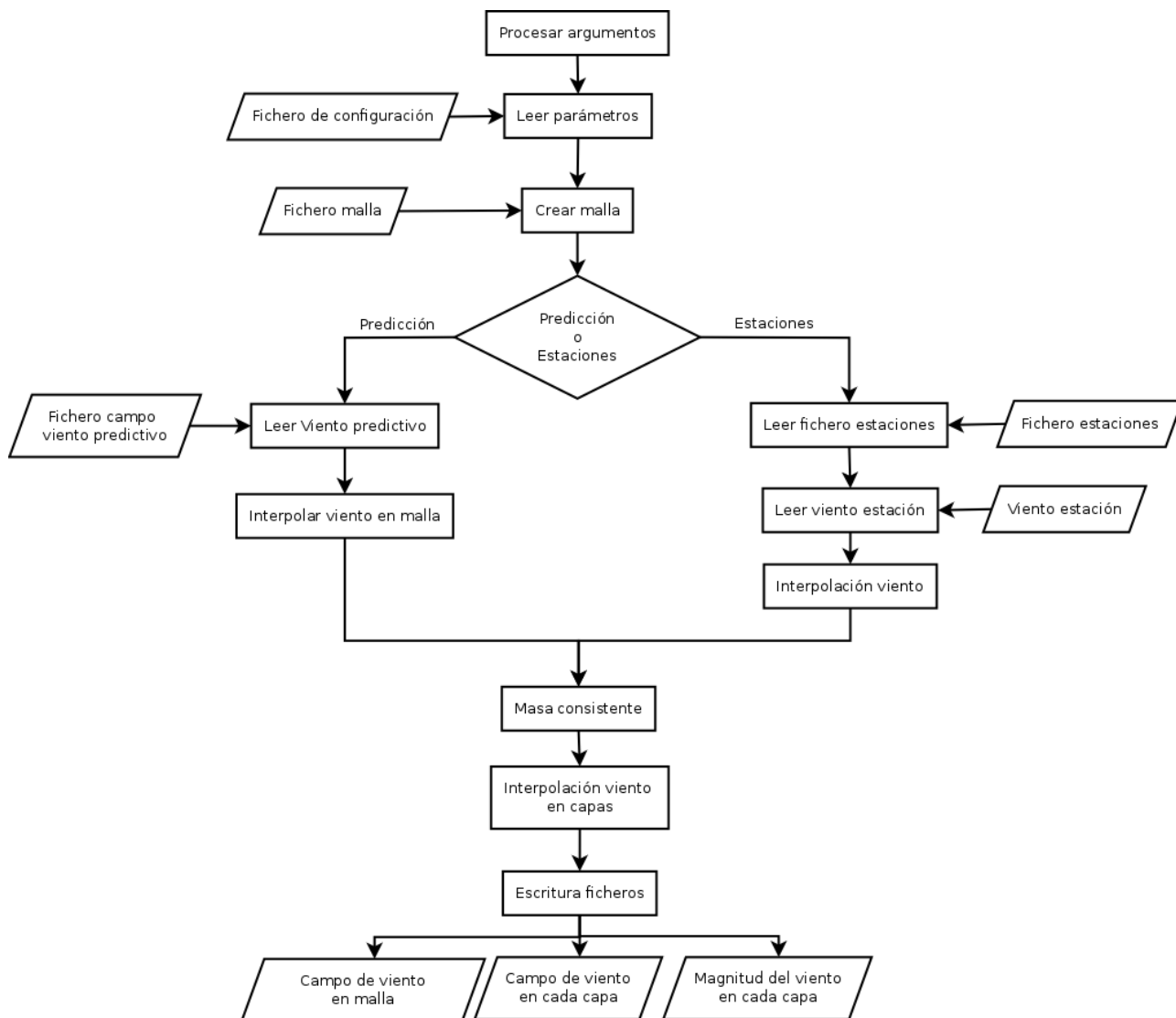
Entorno operativo

El programa funciona en el sistema operativo GNU/Linux y se ejecuta desde la línea de comandos.

Listado de ficheros

ana.c ana.h avs.c avs.h calculo.c calculo.h caras_ext.c caras_ext.h
ensamblaje.c ensamblaje.h funciones.c funciones.h genetico.h grad.c grad.h
interpolacion.c interpolacion.h lectura.c lectura.h Makefile topografia.c
topografia.h viento.c viento.h wind3D.c

Diagrama de flujo



Instalación

Para compilar el programa hace falta tener instalado el compilador de gnu gcc, y la herramienta make.

Desde el directorio de las fuentes se puede invocar a la herramienta make y compilar el programa.

Para que sea accesible desde cualquier directorio hacer que el ejecutable wind3D esté en el \$PATH.

Uso

wind3D -c fichero_configuración -f tipo

tipo será:

- “forecast” si se usan datos predictivos
- “stations” si se usan datos de estaciones meteorológicas.

El fichero de configuración contiene los parámetros necesarios para correr el programa.

Fichero de configuración

El formato del fichero de configuración es el siguiente:

- Puede contener líneas en blanco
- Las líneas que empiezan con el carácter # son comentarios
- Una línea sólo puede contener una asignación
- La asignación será del tipo *nombre_parámetro = valor_parámetro*

Los parámetros que deben constar son:

- **ALPHA**. Valor usado en la ecuación de masa consistente. Equivale a los parámetros α_1^2 y α_2^2 de la ecuación (3) del artículo [4]
- **FICH_VIENTO**. Ruta al fichero de viento
- **FICH_MALLA**. Ruta al fichero de malla
- **EPSILON**. Valor usado para interpolar horizontalmente los valores de las estaciones. Es el usado en la ecuación (9) del artículo [4]
- **GAMMA1** y **GAMMA2**. Valores usados en la interpolación vertical de los valores de las estaciones. Equivalen a γ y γ' usados en las ecuaciones (14) y (15) respectivamente del artículo [4]
- **CTE_VON_KARMANN**. Valor de la constante de Von Karmann. Es adimensional y su valor típico es 0.4
- **LATITUD**. Latitud de la zona de estudio, expresada en grados (ver ecuaciones (14) y (15) del artículo [4])
- **PASQUILL**. Valor de la estabilidad de Pasquill con la siguiente relación: -3 = Clase A, -2 = Clase B, -1 = Clase C, 0 = Clase D, 1 = Clase E, 2 = Clase F, 3 = Clase G
- **V_GEOSTROFICO_X**, **V_GEOSTROFICO_Y** y **V_GEOSTROFICO_Z**. Valores x, y, z de la velocidad del viento geostrófico
- **FICH_TOPO**. Ruta al fichero de topografía

Cabe notar que en el caso de un campo de viento proveniente de un modelo predictivo sólo se usarán los valores ALPHA, FICH_VIENTO y FICH_MALLA, pero todos los parámetros deben constar en el fichero, aunque no tengan definido ningún valor.

Fichero de malla

El formato del fichero de malla será el siguiente:

```
Número_nodos Número_elementos  
lista de coordenadas  
lista de conectividades
```

Igualmente que en el caso del fichero de configuración, las líneas que comienzan con el carácter # se ignoran, por lo que pueden usarse para comentarios.

Fichero de viento

El formato del fichero de viento dependerá del origen de los datos: viento predictivo o viento de estaciones.

Viento predictivo

El formato de ficheros de viento procedentes de un modelo predictivo (WRF o HARMONIE) será el siguiente:

```
nx ny nz  
X Y Z vx vy vz
```

donde nx es el número de nodos a lo largo del eje X, ny el número de nodos a lo largo del eje Y y nz el número de capas.

El orden de los valores X Y Z vx vy vz es importante, primero se debe mover por nx, luego por ny y finalmente por nz.

La utilidad wind3D_netcdf_converter.py convierte los ficheros netCDF de HARMONIE o WRF a este formato de fichero.

Viento de estaciones

El formato del fichero de viento de las estaciones será el siguiente

```
numero_estaciones  
ruta_fichero_estacion
```

Fichero estación

El formato de los ficheros de estación será:

```
X Y
```

```
módulo ángulo
```

dónde “X” e “Y” es la situación de la estación, “módulo” el módulo del viento en la estación, y “ángulo” el ángulo del viento en la estación.

Fichero de topografía

El fichero de topografía sirve para tener los valores de rugosidad del suelo.

Es un fichero tipo modelo digital del terreno

El formato es:

```
xmin
```

```
xmax
```

```
ymin
```

```
ymax
```

```
deltax
```

```
deltay
```

```
X Y Z rugosidad
```

Ficheros de salida

El programa wind3D escribe 3 tipos de ficheros de salida. Todos son ficheros inp que se pueden consultar con AVS o paraview.

- wind_field.inp Es el campo de viento en los tetraedros de la malla de entrada
- viento_capa_vect_h.inp. Es el campo de viento en la capa de altura h con el valor en cada nodo
- viento_capa_esc_h.inp. Es la magnitud del viento en la capa de altura h con la magnitud en cada nodo

wind3D_netcdf_converter.py

Lenguaje de programación

El lenguaje de programación es python. Se necesitan las librerías easygui (<http://easygui.sourceforge.net>) y netCDF (<http://gfsuite.noaa.gov/developer/netCDFPythonInterface.html>)

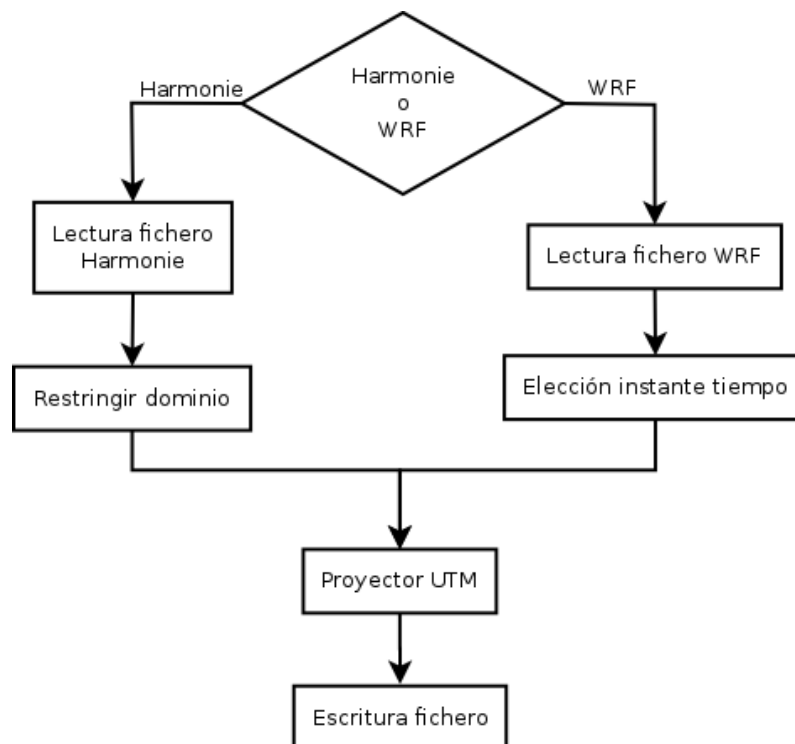
Entorno operativo

Funciona en el sistema operativo GNU/Linux y usa una interfaz gráfica tipo TK.

Listado de ficheros

harmonie_preprocessor.py wrf_preprocessor.py wind3d_netcdf_converter.py

Diagrama de flujo



Uso

El uso del programa es un asistente mediante ventanas. Se deben seguir las indicaciones de la pantalla.

Fichero de salida

El fichero de salida, será un fichero válido para usarse como fichero de viento en el programa wind3D

wind3D_wizard.py

Lenguaje de programación

El lenguaje de programación es python. Se necesitan las librerías easygui (<http://easygui.sourceforge.net>) y netCDF (<http://gfsuite.noaa.gov/developer/netCDFPythonInterface.html>)

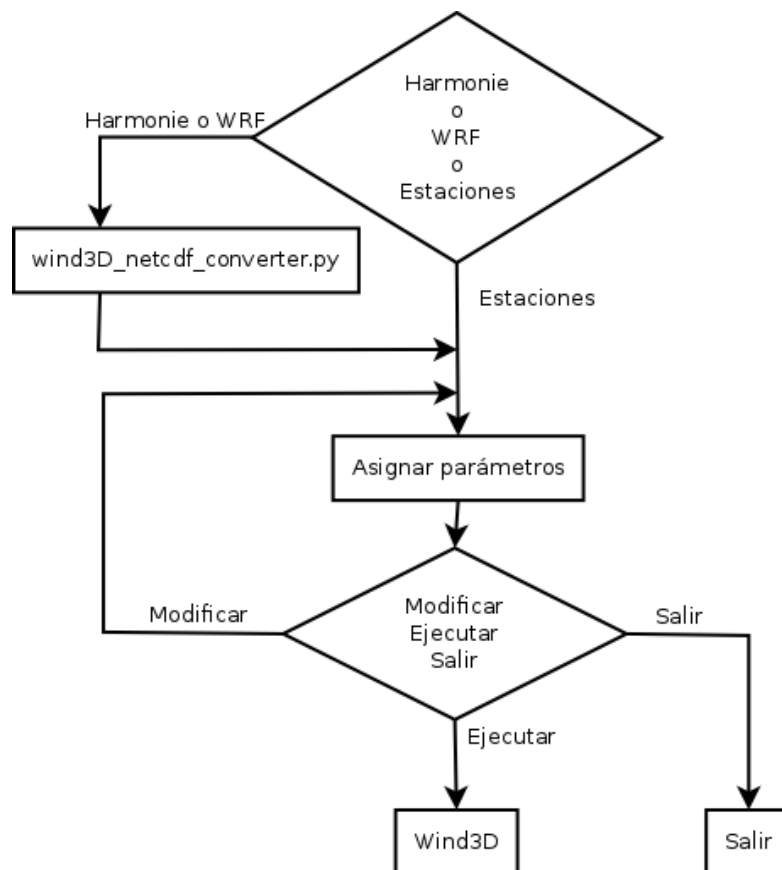
Entorno operativo

El programa funciona en el sistema operativo GNU/Linux y usa una interfaz gráfica tipo TK.

Listado de ficheros

harmonie_preprocessor.py wrf_preprocessor.py wind3d_wizard.py

Diagrama de flujo



Fichero de salida

Este programa genera el fichero de configuración para el código Wind3D y opcionalmente ejecuta el programa.

Referencias

- [1] G. Winter, R. Montenegro, G. Montero, J. Estany. Sobre modelización numérica de distribución de velocidades de campos de viento en parques eólicos y aplicaciones. *Revista Dinámica* 3, 81-82 (1988).
- [2] G. Winter, G. Montero, L. Ferragut, R. Montenegro. Adaptive strategies using standard and mixed finite elements for wind field adjustment. *Solar Energy* 54, 1, 49-56 (1995).
- [3] G. Montero, R. Montenegro, J.M. Escobar. A 3-D diagnostic model for wind field adjustment. *Journal of Wind Engineering & Industrial Aerodynamics* 74-76, 249-261 (1998).
- [4] G. Montero, E. Rodríguez, R. Montenegro, J.M. Escobar, J.M. González-Yuste. Genetic algorithms for an improved parameter estimation with local refinement of tetrahedral meshes in a wind model. *Advances in Engineering Software* 36, 3-10 (2005).
- [5] R. Montenegro, G. Montero, J.M. Escobar, E. Rodríguez, J.M. González-Yuste. Wind field simulation using adaptive tetrahedral meshes. *Innovation in Civil and Structural Engineering Computing*, Cap. 8, 159-185, Saxe-Coburg Publications, Stirling, Scotland (2005).
- [6] R. Montenegro, G. Montero, E. Rodríguez, J.M. Escobar, J.M. González-Yuste. Applications of genetic algorithms in realistic wind field. *Knowledge-Driven Computing, Knowledge Engineering and Intelligent Computations, Studies in Computational Intelligence*, 102, 165-182, Springer-Verlag, Berlin (2008).
- [7] L. Ferragut, R. Montenegro, G. Montero, E. Rodríguez, M.I. Asensio, J.M. Escobar. Comparison between 2.5-D and 3-D realistic models for wind field adjustment. *Journal of Wind Engineering & Industrial Aerodynamics* 98, 548–558 (2010).