Procesamiento paralelo de los pronósticos meteorológicos del modelo WRF mediante NCL

Guerrero-Higueras, A.M.[1], García-Ortega, E.[2], Matellán-Olivera, V.[3], Sánchez, J.L.[4]

Resumen— **La predicción meteorológica es un problema clásico de la programación paralela. Muchos de los modelos matemáticos que se utilizan en meteorología, para pronosticar el comportamiento de la atmósfera, tienen implementaciones preparadas para ejecutarse en entornos de cálculo paralelo.**

La información obtenida después de ejecutar un modelo matemático de predicción es necesario procesarla para poder visualizar los resultados. Sin embargo, no es fácil encontrar herramientas que permitan el procesado en paralelo de estas salidas.

El presente trabajo plantea la posibilidad de dar un paso más y mejorar el rendimiento, utilizando el paradigma paralelo para procesar las salidas de un modelo de predicción numérica, en concreto, el modelo WRF (Weather Research and Forecasting).

Palabras clave— **Meteorología, predicción numérica, WRF, MM5, netCDF, NCL.**

I. Introducción

La predicción meteorológica consiste en la determinación anticipada de los valores de una serie de variables meteorológicas que afectan a una determinada región: temperatura, presión, humedad, nubosidad, precipitación, etc.

En la actualidad las predicciones se realizan utilizando modelos de predicción numérica que emplean las ecuaciones de las leyes físicas que describen el comportamiento de la atmósfera. La resolución de las mismas es complicada puesto que se trata de ecuaciones para las que no existe una solución matemática exacta. Por esta razón, es necesario recurrir a las llamadas técnicas de modelización numérica.

Formalmente, la utilización de los modelos de predicción se basa en resolver de forma numérica, mediante aproximaciones, un conjunto de ecuaciones diferenciales referidas a la conservación de la masa, la energía y el momento lineal en la atmósfera. Partiendo de unas condiciones iniciales[1] en un determinado instante de tiempo (t), el modelo, o más bien sus ecuaciones, obtienen los valores de las variables en un tiempo posterior (t + $\triangle$t). Para ello es preciso hacer uso de las ecuaciones en derivadas parciales que traducen las leyes generales de laGP física que rigen la dinámica de la atmósfera terrestre.

La atmósfera es un fluido en movimiento sometido a procesos de intercambio de energía. En consecuencia, se necesita describir adecuadamente el movimiento de las partículas de aire sometidas a un conjunto de fuerzas e introducir el balance energético, una vez que se tienen en cuenta tanto los intercambios que tienen lugar en el medio atmosférico, como entre la atmósfera y la superficie terrestre. Para describir su comportamiento se utilizan ecuaciones generales de la mecánica de fluidos y de la termodinámica. Además, el movimiento de las partículas de fluido atmosférico debe ser estudiado teniendo en cuenta que nuestro planeta es un sistema de referencia no inercial. Por lo tanto, para que se verifiquen las leyes de la mecánica clásica, es necesario introducir algunos términos denominados fuerzas ficticias.

El movimiento de una partícula de fluido atmosférico viene dada por la ecuación 1.

$$\frac{d\vec{V}}{dt} = \sum_i \frac{\vec{F_i}}{m} = -\frac{1}{\rho}\vec{\nabla}p + \vec{g} - 2\vec{\Omega} \times \vec{V} + \vec{F_r} \qquad (1)$$

El término de la izquierda representa la fuerza por unidad de masa que actúa sobre una burbuja de aire, medida desde un sistema de referencia no inercial (la Tierra). Los términos de la derecha señalan las fuerzas por unidad de masa que actúan sobre dicha burbuja: la fuerza debida al gradiente de presión, resultante de la fuerza gravitatoria y centrífuga, la fuerza de Coriolis, que junto a la anterior, son los dos términos ficticios de la ecuación, y el término asociado a las fuerzas de rozamiento. Para resolver esta ecuación es necesario introducir una serie de simplificaciones en función de las escalas espacio-temporales utilizadas en cada caso.

Desde el punto de vista termodinámico hay que tener en cuenta los intercambios de calor asociados a los cambios de fase del agua (vapor, líquido y estado sólido) que afectan a la formación y desarrollo de las nubes y a los intercambios de energía con la superficie de la Tierra, los océanos y la troposfera.

Otro factor a tener en cuenta es el hecho de que la atmósfera es un fluido gaseoso que sufre variaciones de densidad, como consecuencia de la dinámica a la que se encuentra sometida.

Las ecuaciones matemáticas que describen estos procesos no son lineales y, por tanto, las soluciones

[1]Grupo de Física de la Atmósfera, Univ. de León, e-mail: `am.guerrero@unileon.es`.

[2]Grupo de Física de la Atmósfera, Univ. de León, e-mail: `eduardo.garcia@unileon.es`.

[3]Dpto. de Ingenierías Mecánica, Informática y Aeroespacial, Univ. de León, e-mail: `vicente.matellan@unileon.es`.

[4]Grupo de Física de la atmósfera, Univ. de León, e-mail: `jl.sanchez@unileon.es`.

[1]Fijadas a partir de la salida de un modelo de mayor cobertura espacial, como GFS (Global Forecast Model), ejecutado por NOAA (National Oceanic and Atmospheric Administration), o como ECMWF (European Centre for Medium-Range Weather Forecasts).

no se pueden obtener de forma analítica. Para resolverlas es preciso recurrir al cálculo numérico, que proporciona una solución aproximada.

El primer paso consiste en discretizar el área geográfica objeto de análisis. Para ello se define una rejilla tridimensional que la abarque.

A continuación se resuelven las ecuaciones, utilizando cálculo numérico, para obtener los valores de las variables en cada punto de la rejilla. Casi todos los modelos de predicción numérica tienen implementaciones que permiten hacerlo en entornos de cálculo paralelo con MPI [1]. Incluso hay proyectos para abordar el problema utilizando GPUs [2].

Por último hay que procesar las salidas del modelo para generar gráficas que permitan analizar los valores de las variables de manera visual. Esto puede ser bastante costoso en tiempo si se quieren procesar muchas variables. Sin embargo, no hay soluciones *ad hoc* que acometan este problema mediante cálculo paralelo.

Hay técnicas que permiten paralelizar problemas secuenciales [3], muy utilizadas en bioinformática, pero hay que adaptarlas a las peculiaridades del procesado de las salidas de un modelo de predicción.

Se pretende demostrar que el procesado paralelo de las salidas mejora el rendimiento global. Para ello, primero se explican las peculiaridades de los modelos mesoescalares, haciendo hincapié en el modelo WRF [4] [5]. Seguidamente se presenta el la solución secuencial clásica, para después proponer la alternativa paralela. Por último, tras comparar los resultados de las pruebas realizadas, se recogen las conclusiones.

II. Modelos mesoescalares

Se entiende por modelo mesoescalar aquel que permite estudiar y pronosticar fenómenos meteorológicos mesoescalares, es decir, aquellos que ocurren a una escala espacial relativamente pequeña. Históricamente, los primeros modelos eran hidrostáticos pues consideraban despreciables los movimientos en la escala vertical frente a los movimientos en la horizontal. Esto ocurre con los modelos que simulan a escala sinóptica, válidos para establecer el comportamiento de las masas de aire, frentes polares, etc. En estos casos se mantiene la condición hidrostática y la presión está completamente determinada por las masas de aire extendidas, quedando inhibidos los movimientos del aire en la vertical en las hipótesis iniciales de partida. Sin embargo, cuando la escala de la rejilla del modelo se reduce, es posible incorporar movimientos en la vertical, y por tanto, la hipótesis hidrostática deja de verificarse.

Por tanto hay dos grandes tipos de modelos numéricos: hidrostáticos, como HIRLAM[2], y no hidrostáticos, como Méso NH [6], MM5 [7] [8] o WRF.

Los modelos numéricos mesoescalares de área limitada resultan muy útiles para el estudio de fenómenos con precipitaciones intensas y severas.

El Grupo de Física de la Atmósfera (GFA) de la Universidad de León trabaja desde hace diez años con modelos mesoescalares [9] [10] [11]. Inicialmente se utilizó Méso NH, más tarde MM5 y WRF. Este último se ejecuta de manera operativa[3] en el *cluster* de la Fundación Centro de Supercomputación de Castilla y León[4] (FCSCL).

Los modelos de simulación mesoescalares hacen uso de distintos esquemas de parametrizaciones de cúmulos, microfísica, capa límite, etc., para dar cuenta de aquellos fenómenos cuya escala espacial es inferior a la resolución del modelo.

III. El modelo mesoescalar WRF

WRF es un modelo de simulación de la atmósfera de área limitada, no hidrostático, sensible a las características del terreno y diseñado para predecir la circulación atmosférica a escala sinóptica, mesoescalar y regional.

Ha sido desarrollado en colaboración por NOAA, NCAR (National Centers for Atmospheric Research) y otras organizaciones.

La implementación del modelo está preparada para trabajar en entornos de cálculo paralelo con memoria compartida, mediante OpenMP, y de memoria distribuida, mediante MPI. Además, el modelo tiene la capacidad de combinar ambas tecnologías. También se está trabajando en implementaciones sobre GPUs.

A. Las salidas de WRF. El formato netCDF

Las salidas que proporciona WRF contienen los valores, para un horizonte de tiempo determinado, de una serie de variables meteorológicas. Los ficheros que componen estas salidas cumplen con las especificaciones de un formato conocido como netCDF (network Common Data Form) [12] [13].

La información en formato netCDF tiene las siguientes características:

- *Autodescrita.* Un fichero netCDF incluye información sobre los datos que contiene.
- *Multiplataforma.*
- *Escalable.* Un pequeño subconjunto de datos que forman parte de un conjunto más grande de datos es accesible de una manera eficiente.
- *Agregable.* Es posible añadir datos a un fichero netCDF bien construido sin redefinir su estructura.
- *Intercambiable.* Un escritor y múltiples lectores pueden acceder de manera simultánea al mismo fichero netCDF. Incluso existen implementaciones MPI para acceso paralelo [14].
- *Archivable.* Todas las versiones de netCDF garantizan la compatibilidad con las anteriores.

[2] Utilizado por la Agencia Estatal de Meteorología. `http://hirlam.org/`.

[3] Las salidas pueden consultarse en `http://gfa.unileon.es/prediccion_wrf/`.

[4] `http://www.fcsc.es`.

```
netcdf wrfout_d01_2011-12-22_00:00:00 {
        dimensions:
                Time = UNLIMITED ; // (17 currently)
                west_east = 122 ;
                south_north = 111 ;
                bottom_top = 34 ;
                ...
        variables:
                float T(Time, bottom_top, south_north, west_east) ;
                        T:description = "perturbation potential temperature (theta-t0)" ;
                        T:units = "K" ;
                float RAINC(Time, south_north, west_east) ;
                        RAINC:description = "ACCUMULATED TOTAL CUMULUS PRECIPITATION" ;
                        RAINC:units = "mm" ;
                ...
        // global attributes:
                        :START_DATE = "2011-12-22_00:00:00" ;
                        ...
        data:
                T =     -7.835083, -7.864655, -7.891296, -7.904358, -7.911102, -7.931854,
                        -7.968079, -8.019257, -8.089844, -8.18219,  -8.287872, -8.430267, ... ;
                RAINC = 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                        0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ... ;
                ...
}
```

Fig. 1. Representación en modo texto de un fichero netCDF.

A.1 Estructura de un fichero netCDF

Los ficheros netCDF, habitualmente con extensión `.nc`, son de tipo binario. Para poder analizarlos y editarlos se pueden representar en modo texto.

Es posible guardar esta representación en formato de texto en un fichero, habitualmente con extensión `.cdl`, generar un netCDF binario a partir de este y viceversa, utilizando a los comandos `ncdump` y `ncgen`[5].

Los datos en formato netCDF se almacenan en forma vectorial. Cada vector almacena los datos de una variable concreta y tiene unas dimensiones y atributos acordes a los datos. La estructura básica del fichero netCDF, en su representación en modo texto, tiene 4 partes, como se puede ver en el ejemplo de la figura 1.

- Dimensiones: Especificación de las diferentes dimensiones que pueden tener las variables contenidas en él. Como por ejemplo: latitud, longitud, altura, tiempo, etc.
- Variables: Definición de las diferentes variables incluídas en el fichero, especificando sus dimensiones y atributos.
- Atributos globales: Lista de atributos aplicables a todas las variables del fichero.
- Datos: Los datos propiamente dichos.

B. Herramientas de post-proceso

Hay un buen número de herramientas disponibles para visualizar la información de modelo WRF. Entre las más conocidas destacan GrADS, RIP4, Vis5D, VAPOR y NCL.

El GFA utiliza, históricamente, GrADS [15] para procesar las salidas de WRF, como antes hizo con las salidas de MM5. Sin embargo en la actualidad se está trabajando con NCL (NCAR Command Language) [16] [17] [18] debido a su mayor versatilidad y a que dispone de funciones específicas para trabajar con las salidas de WRF.

IV. Procesado secuencial

La mayoría de herramientas que permiten visualizar las salidas del modelo WRF no están diseñados para el procesamiento en paralelo. Esto, a priori, no es un problema ya que el coste de ejecutar un script para visualizar el contenido de una variable meteorológica no es elevado. El problema aparece cuando se necesita visualizar el valor de múltiples variables, en múltiples instantes de tiempo.

Para hacer esto, un script de post-procesado, tiene que recorrer todo el fichero netCDF. Aunque el acceso a este tipo de ficheros está muy optimizado, la cantidad de información que contienen es muy grande. Por ejemplo, las salidas de una predicción del GFA ocupan en torno a los 7,0 GB.

La figura 2 muestra parte de un script en GrADS[6] utilizado por el GFA para generar gráficas de temperatura y viento en superficie. Se recorre todo el fichero netCDF iterando entre los distintos intervalos de tiempo (*frames*). En 48 horas de predicción con intervalos de 6 horas hay 8 *frames* de tiempo. En cada uno de estos *frames*, se genera una gráfica de temperatura y viento en superficie.

El tiempo empleado en la generación de gráficas que permiten visualizar toda la información del modelo, es considerablemente grande respecto al tiempo de ejecución del propio modelo. La tabla I muestra un listado de tiempos de ejecución de algunas predicciones operativas que ejecuta el GFA, separando por un lado la ejecución paralela del modelo, y por otro el procesado secuencial de las salidas producidas por el mismo. Como puede observarse, el tiempo que lleva

[5] Incluidos en las librerías netCDF.

[6] La versión completa del script se puede encontrar en `http://gfa.unileon.es/toolbox/postproc.gs`.

```
inputPath = '/home/gfa_1/gfa_1_2/output/'
outputPath = '/home/gfa_1/gfa_1_2/output/'

filename = inputPath%'outputd01.ctl'
'open '%filename
i = 1
while (i <= 8)
   t2mw10m(1, i, outputPath%'dom1_snh')
   i = i + 1
endwhile
'close 1'
'quit'

function t2mw10m(domain, iteration, outputFile)
   domainData(domain)
   'set mpt 1 1'
   'set mpt 2 1'
   'set mpt 3 1'
   'clear'
   'set t '%iteration + 1
   vel()
   saveFile(domain, iteration, outputFile,
      'Viento a 10m y Temperatura a 2m (°C)', 0)
return
...
```

Fig. 2. Script GrADS de post-procesado de las salidas de WRF.

procesar las salidas es considerablemente grande respecto al tiempo de ejecución del modelo. Sobre todo si se tiene en cuenta que para ejecutar el modelo, se utilizan 128 nodos del *cluster* de cálculo paralelo de la FCSCL.

TABLA I
TIEMPO DE EJECUCIÓN DEL MODELO Y DEL POST-PROCESADO SECUENCIAL DE LAS SALIDAS.

Simulación operativa	t_{modelo}	$t_{post-proceso}$
14/05/2012 12:00 UTC	0:51:01	0:17:10
15/05/2012 0:00 UTC	0:51:06	0:19:07
15/05/2012 12:00 UTC	0:50:24	0:18:11
16/05/2012 0:00 UTC	0:49:38	0:17:20
16/05/2012 12:00 UTC	0:48:20	0:18:29
17/05/2012 0:00 UTC7	0:49:05	0:16:59
17/05/2012 12:00 UTC	0:50:33	0:18:34
18/05/2012 0:00 UTC	0:49:53	0:18:26
18/05/2012 12:00 UTC	0:49:16	0:17:41
19/05/2012 0:00 UTC	0:50:08	0:19:24

V. PROCESAMIENTO EN PARALELO

A la vista de lo comentado anteriormente, existe un punto de mejora importante en el procesado de las salidas de un pronóstico meteorológico.

Descomponiendo el problema se obtiene una primera aproximación paralela a la solución. En lugar de utilizar un único script para procesar la información de múltiples variables meteorológicas en múltiples instantes de tiempo, se puede ejecutar, en varios nodos a la vez, scripts que procesen la información de una variable en un *frame* de tiempo.

A. Fase I

El primer punto que es preciso abordar es la adaptación de los scripts de post-proceso clásicos para poder utilizarlos en la aproximación paralela.

Con un enfoque secuencial, se lee el fichero completo y se generan las gráficas una a una. Ahora es preciso modificar los scripts para que acepten una serie de parámetros, que permitan acceder a una determinada sección dentro de un fichero netCDF. Es recomendable utilizar un lenguaje que permita esto de una manera sencilla, como NCL.

Los parámetros que es necesario pasar a los scripts de post-proceso incluyen:

- La ruta del fichero netCDF que contiene la información.
- El *frame* de tiempo.
- El nivel de presión cuando se trate de variables que tengan valores en la vertical.
- El nombre del fichero de salida que contendrá la gráfica y formato del mismo, normalmente `png`.

La figura 3 muestra parte de un script NCL[7] para generar una gráfica de precipitación similar a la que muestra la figura 4.

```
begin
  ; Check arguments
  if (.not. isvar("inputFile")) then
    print("Input file not defined")
    exit
  end if
  if (.not. isvar("time")) then
    time = 0 ; Valor de tiempo por defecto
  end if
  ...
  ; Open the WRF ARW input file
  a = addfile(inputFile, "r")
  ; Open workstation
  wks = gsn_open_wks(outputFileType,
                     outputFileName)
  ...
  ; Get the variables from the WRF input file
  rain = wrf_user_getvar(a, "RAINC", time)
  ...
  ; Create graph
  contour_rain = wrf_contour(a, wks, rain, opts)
  ; Plot graph
  plot = wrf_map_overlays(a, wks, contour_rain,
                          pltres, mpres)
end
```

Fig. 3. Script NCL para calcular la precipitación.

Después de adaptar los scripts es necesario construir un *array* de trabajos para ejecutar en un *cluster* de cálculo paralelo.

FCSCL utiliza el gestor de colas SGE [19]. La figura 5 muestra parte de una implementación en *python*[8] de un *array* de trabajos para SGE. La idea es construir una lista de elementos. Cada elemento tiene la información necesaria para ejecutar los scripts de post-proceso: script, fichero de en-

[7]Todos los scripts NCL que utiliza el GFA consultarse en `http://gfa.unileon.es/toolbox`

[8]La versión completa puede consultarse en `http://gfa.unileon.es/toolbox/postproc.py`.

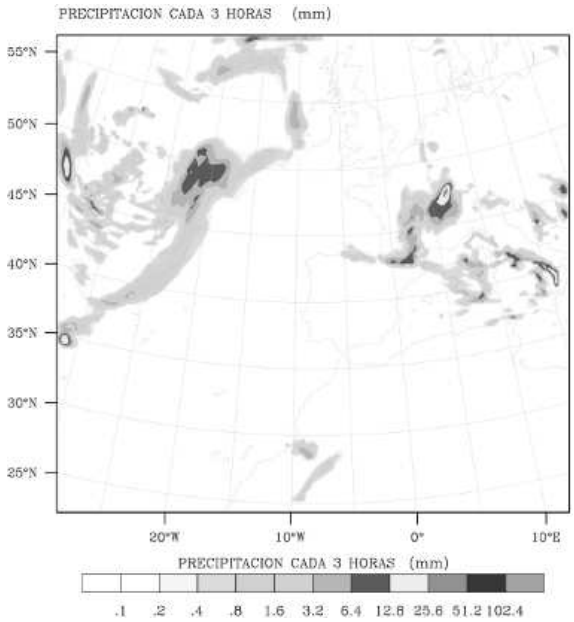

Fig. 4. Gráfica de precipitación generada con NCL.

trada, *frame* de tiempo, etc. En función del índice que el gestor de colas asigne a un nodo determinado, este tomará los parámetros de la lista. Utilizando el mismo número de nodos, que elementos tenga la lista, cada nodo genera únicamente una gráfica.

En la tabla II puede consultarse el tiempo de ejecución del procesamiento paralelo de algunas predicciones operativas del GFA ejecutadas en el *cluster* de la FCSCL. Si se comparan estos números con los de la ejecución secuencial de la tabla I se puede observar claramente la mejora.

TABLA II
TIEMPO DE EJECUCIÓN DEL POST-PROCESADO PARALELO.

Simulación operativa	$t_{post-proceso}$
14/05/2012 12:00 UTC	0:01:34
15/05/2012 0:00 UTC	0:01:27
15/05/2012 12:00 UTC	0:01:21
16/05/2012 0:00 UTC	0:01:37
16/05/2012 12:00 UTC	0:01:39
17/05/2012 0:00 UTC7	0:01:41
17/05/2012 12:00 UTC	0:01:38
18/05/2012 0:00 UTC	0:01:41
18/05/2012 12:00 UTC	0:01:22
19/05/2012 0:00 UTC	0:01:29

B. Fase II

Licheros netCDF están diseñados, y optimizados, para que múltiples lectores puedan leer de un único fichero. Sin embargo, parece claro que el acceso será más rápido si cada ejecución de los scripts de post-proceso accede a un fichero netCDF diferente.

WRF dispone de muchos parámetros que es posible modificar, editando el fichero de configuración `namelist.input`, y que permiten alterar su comportamiento. Entre ellos hay uno, denominado `frames_per_outfile`, que permite decidir cuantos *frames* de tiempo se añaden en cada fichero de salida. El valor por defecto para este parámetro es 1000. Esto quiere decir que hay 1000 *frames* de tiempo por cada fichero de salida. O lo que a efectos prácticos viene a ser lo mismo, que todos los frames de tiempo de la predicción están en un único fichero de salida.

Si se mosidifica el valor por defecto y see introduce el valor 1, WRF solamente guarda un *frame* de tiempo en cada fichero de salida que genera.

Con este nuevo enfoque no es necesario cambiar los scripts de post-proceso, solamente los parámetros con los que son invocados.

La tabla III muestra el tiempo de ejecución de una serie de predicciones operativas con post-procesado paralelo sobre ficheros independientes. Comparando estos números con los que aparecen en la tabla II, es posible comprobar la ganancia de este nuevo enfoque.

TABLA III
TIEMPO DE EJECUCIÓN DEL POST-PROCESADO PARALELO SOBRE FICHEROS INDEPENDIENTES.

Simulación operativa	$t_{post-proceso}$
14/05/2012 12:00 UTC	0:01:11
15/05/2012 0:00 UTC	0:01:03
15/05/2012 12:00 UTC	0:01:01
16/05/2012 0:00 UTC	0:00:58
16/05/2012 12:00 UTC	0:00:59
17/05/2012 0:00 UTC7	0:01:08
17/05/2012 12:00 UTC	0:01:02
18/05/2012 0:00 UTC	0:01:07
18/05/2012 12:00 UTC	0:00:56
19/05/2012 0:00 UTC	0:00:59

VI. CONCLUSIONES

El procesado en paralelo de las salidas del modelo de predicción numérica WRF es un problema complejo que obliga replantear el proceso completo.

Por un lado, hay que revisar la ejecución del propio modelo, cambiando algunas opciones de configuración. Por otro, hay que re-implementar el procesado de las salidas, para adaptarlo al paradigma de la programación paralela.

La tabla IV muestra los tiempos medios de ejecución del procesado de las salidas del modelo siguiendo los tres métodos utilizados:

1. Procesado secuencial de las salidas.
2. Procesado en paralelo de las salidas.
3. Procesado en paralelo sobre ficheros de salida independientes.

TABLA IV
TIEMPOS MEDIOS DE EJECUCIÓN DEL PROCESADO DE LAS SALIDAS DE WRF.

Ejecución secuencial	Ejecución paralela	Ejecución paralela sobre ficheros independientes
0:18:08	0:01:33	0:01:02

```
joblist = [] ; Definición de la lista de trabajos

# Añadir trabajo a la lista
joblist.append({'inputfile'  : './wrfout_d01.nc'   ,  # Fichero de entrada
                'outputfile' : './dom1_t2m-w10m_00',  # Fichero de salida
                'script'     : 't2_w10.ncl'        ,  # Temperatura y viento en superficie
                'args'       : '\'time=0\''        }) # frame=0

# Añadir trabajo a la lista
joblist.append({'inputfile'  : './wrfout_d01.nc'                ,   # Fichero de entrada
                'outputfile' : './dom1_tmp-gpt_850hPa_00'       ,   # Fichero de salida
                'script'     : 'tc_geopt.ncl'                   ,   # Temperatura y geopotencial
                'args'       : '\'time=0\' \'levels=(/850./)\'' })  # Frame=0, nivel de presión=850hPa

...

i = int(os.environ['SGE_TASK_ID']) - 1 # Índice dentro del array de trabajos
runNclCommand(joblist[i]['inputfile'], joblist[i]['outputfile'], joblist[i]['script'],
              args = joblist[i]['args'])
```

Fig. 5. Script *python* para ejecutar un *array* de trabajos en SGE.

Para calcular los tiempos medios, en cada caso, se han utilizado los datos de las tablas I, II y III respectivamente.

Si se añade a los tiempos medios de la tabla IV, el tiempo medio de ejecución del modelo (0:49:56), calculado en base a los datos de la tabla I, se obtiene el tiempo global de ejecución que se muestra en la tabla V.

TABLA V

TIEMPOS MEDIOS DE EJECUCIÓN GLOBAL.

Ejecución secuencial	Ejecución paralela	Ejecución paralela sobre ficheros independientes
1:08:04	0:51:30	0:50:58

La reducción del tiempo de procesado de las salidas con el método 2, respecto al método 1, es de un $92,64\%$. A la vista de estos datos, se observa que la hipótesis inicial se cumple. El procesado de las salidas de WRF en paralelo mejora el rendimiento global.

La reducción de tiempo con el método 3, respecto al método 2, es de un $23,34\%$. Sin embargo, respecto al tiempo global, apenas tiene influencia. De todas maneras es recomendable introducir este cambio, ya que solamente supone modificar el fichero de configuración `namelist.input`.

AGRADECIMIENTOS

El presente trabajo ha sido financiado mediante el proyecto Plataforma MicroMeteo.com (IPT-310000-2010-22).

Todas las pruebas y ejecuciones operativas que realiza el GFA se realizan en el *cluster* de cálculo paralelo de la FCSCL.

Los autores agradecen su colaboración a Jesús Lorenzana (FCSCL).

REFERENCIAS

[1] *ARW Version 3 Modeling System User's Guide*, National Center for Atmospheric Research (NCAR), Mesoscale & Microscale Meteorology Division, 2010.

[2] Michalakes, J. and M. Vachharajani, *GPU Acceleration of Numerical Weather Prediction*, Parallel Processing Letters, Vol. 18, No. 4, pp. 531-548, World Scientific, 2008.

[3] D. Ramet, J. Lago, J. Falgueras, O. Trelles, *Mr-Cirrus: Implementación de Map-Reduce bajo MPI para la ejecución paralela de programas secuenciales* Actas de las XXII Jornadas de Paralelismo JP2011, ISBN: 978-84-694-1791-1, 2011.

[4] W.C. Skamarock, J.B. Klemp, J. Dudhia, D.O. Gill, D.M. Barker, W. Wang, J.G. Powers, *A description of the Advanced Research WRF Version 2*, NCAR Tech., Note NCAR/TN-468+STR, 2005.

[5] *The Weather Research and Forecasting (WRF) Model website*, `http://www.wrf-model.org/`.

[6] Lafore, J. P., et al. *The Meso-NH Atmospheric Simulation System. Part I: Adiabatic formulation and control simulations*, Ann. Geophys., vol. 16, pp. 90-109, 1998.

[7] Dudhia, J., *A nonhydrostatic version of the Penn State-NCAR mesoscale model: validation tests and simulation of an Atlantic cyclone and cold front*, Mon. Weather Rev., vol. 121, pp. 1493-1513, 1993.

[8] Grell, G., Dudhia, J., Stauffer, D., 1995. *A description of the fifth-generation Penn State / NCAR Mesoscale Model (MM5)*, NCAR Tech., Note NCAR/TN-398+STR 117pp, 1995.

[9] García-Ortega, E., Fita, L., Romero, R., López, L., Ramis, C., Sánchez, J. L., *Numerical simulation and sensitivity study of a severe hailstorm in northeast Spain*, Atmospheric Research, vol. 83, pp. 225-241, 2007.

[10] García-Ortega, E., López, L., Sánnchez, J. L., *Diagnosis and sensitivity study of two severe storm events in the Southeastern Andes*, Atmospheric Research, vol. 93, pp. 161-178, 2009.

[11] García-Ortega, E., Merino, A., López, L., Sánchez, J. L., *Role of mesoscale factors at the onset of deep convection on hailstorm days and their relation to the synoptic patterns*, Atmospheric Research, in press.

[12] *Network Common Data Form (netCDF) website*, `https://www.unidata.ucar.edu/netcdf`.

[13] Ed Hartnett, *The NetCDF Tutorial*, Unidata Program Center, 2011

[14] *Parallel I/O with NetCDF*, Unidata NetCDF Workshop, 2010. `http://www.unidata.ucar.edu/software/netcdf/workshops/2010/pnetcdf/index.html`.

[15] *GrADS User's Guide*, `http://www.iges.org/grads/gadoc/users.html`.

[16] *CISL's NCAR Comman Language (NCL) website*, `http://www.ncl.ucar.edu/`.

[17] *NCAR Command Language (NCL). Mini-Language Reference Manual*, National Center for Atmospheric Research (NCAR), 2011.

[18] *NCAR Command Language (NCL). Mini Graphics Manual*, National Center for Atmospheric Research (NCAR), 2011.

[19] *SunTM ONE Grid Engine Administration and User's Guide*, Sun Microsystems, 2002.