Análisis de rendimiento y profiling del modelo WRF en un clúster HPC

Ronald M. Gualán S., Lizandro Solano-Quinde
Departamento de Ingeniería Eléctrica, Electrónica y Telecomunicaciones, Universidad de Cuenca.
Autor para correspondencia: lizandro.solano@ucuenca.edu.ec
Fecha de recepción: 21 de septiembre de 2014 - Fecha de aceptación: 17 de Octubre de 2014

RESUMEN
El modelo de investigación y pronóstico climático (WRF) es un sistema completamente funcional de modelado que permite realizar investigación atmosférica y predicción meteorológica. WRF fue desarrollado con énfasis en la eficiencia, portabilidad, facilidad de mantenimiento, escalabilidad y productividad, lo que ha permitido que sea implementado con éxito en una amplia variedad de equipos HPC. Por esta razón, el tamaño de los problemas a los que WRF da soporte ha incrementado, por lo que el entendimiento de la dependencia del WRF con los diversos elementos de clúster, como la CPU, interconexiones y librerías, son cruciales para permitir predicciones eficientes y de alta productividad. En este contexto, el presente manuscrito estudia la escalabilidad de WRF en un equipo HPC, tomando en consideración tres parámetros: número de CPUs y nodos, comunicaciones y librerías. Para esto, dos benchmarks son llevados a cabo sobre un clúster de alto rendimiento dotado de una red GigaEthernet, los cuales permiten establecer la relación entre escalabilidad y los tres parámetros estudiados, y particularmente demuestran la sensibilidad del WRF a la comunicación inter-nodo. Dicho factor es esencial para mantener la escalabilidad y el aumento de la productividad al añadir nodos en el clúster.
Palabras clave: WRF, HPC, escalamiento paralelo, benchmark, rendimiento, profiling.

ABSTRACT
The Weather Research and Forecast (WRF) model is a fully functional modeling system that supports atmospheric research and weather prediction. WRF was developed with emphasis on efficiency, portability, maintainability, scalability and productivity, allowing it to be successfully implemented in a wide variety of HPC equipment. Therefore, the size of the problems supported by WRF has increased, so the understanding of the WRF's dependence on the various elements of the cluster, such as CPU, networking and libraries are crucial to enable efficient forecasting and high productivity. In this context, this manuscript examines WRF scalability in HPC equipment, taking into account three parameters: number of CPUs and nodes, communications and libraries. Two benchmarks carried out on a cluster of high performance provided with a GigaEthernet network, allow to establish the relationship between scalability and the three parameters studied, particularly WRF demonstrates sensitivity to inter-node communication. This factor is essential for maintaining the scalability and increasing productivity by adding nodes to the cluster.
Keywords: WRF, HPC, parallel scaling, benchmark, performance, profiling.

1. INTRODUCCIÓN

Michalakes *et al.* (2001) afirma que la simulación numérica de la atmósfera para el pronóstico climático es una de las primeras aplicaciones de computación de alto rendimiento y, en términos de impacto y relevancia para el público, sigue siendo uno de los más importantes en la actualidad. A pesar de los continuos incrementos en la potencia de cálculo hasta la llegada de los sistemas actuales de computación en la Tera-escala, la necesidad de computación continúa: las resoluciones se hacen

más finas, los dominios se hacen más grandes, las escalas de tiempo se hacen más largas, y la complejidad de los modelos y sistemas de asimilación sigue creciendo. Por esta razón, es de vital importancia utilizar los sistemas de alto rendimiento de manera eficiente, para lo cual se requiere procesos meticulosos de ingeniería de software, entendimiento del hardware de la plataforma de computación y una puesta a punto de los modelos para obtener porcentajes razonables del rendimiento teórico máximo.

El presente manuscrito está organizado de la siguiente forma: la Sección 2 hace una breve descripción del modelo WRF, la Sección 3 presenta los dos benchmarks planteados junto con los resultados obtenidos y finalmente la Sección 4 contiene las conclusiones.

El modelo de investigación y pronóstico climático WRF

El modelo WRF, por sus siglas en inglés "Weather Research and Forecasting", es un sistema de predicción climática numérico de mesoescala de próxima generación diseñado para servir tanto a las necesidades de investigación atmosférica, como a las de predicción operativa. Cuenta con dos núcleos dinámicos, un sistema de asimilación de datos y una arquitectura de software que permite la utilización de computación paralela y la extensibilidad del sistema. El modelo sirve a una amplia gama de aplicaciones meteorológicas a través de escalas que van desde metros hasta miles de kilómetros. El esfuerzo para desarrollar el WRF comenzó en la última parte de la década de 1990 y surge por la colaboración entre el Centro Nacional de Investigación Atmosférica (NCAR), la Administración Nacional Oceánica y Atmosférica (representado por los Centros Nacionales de Predicción Ambiental (NCEP) y el (en aquel entonces) Laboratorio de Sistemas de Pronóstico (FSL)), la Agencia Climática de la Fuerza Aérea (AFWA), el Laboratorio de Investigación Naval, la Universidad de Oklahoma, y la Administración Federal de Aviación (FAA) (WRF, n.d.).

La mejor forma de entender el funcionamiento del modelo WRF es revisando los módulos que lo conforman. En la Figura 1 se presenta el diagrama de flujo de las principales rutinas usadas en el modelo WRF, donde se distingue cuatro grupos: (1) Fuentes de datos externas, que engloba las diversas fuentes de datos que pueden ser empleadas para simulaciones climáticas. (2) WPS (WRF Pre-processing System), el cual es una colección de programas de pre-procesamiento que agrupa los programas *Geogrid, Ungrib* y *Metgrid*. *Geogrid* permite mapear información de proyección (latitud, longitud, parámetros de Coriolis, factores de escala de mapas, etc.), e información topográfica (elevación, vegetación y categorías de suelo, etc.). Mientras que, *Ungrib* y *Metgrid* pre-procesan campos 3D (viento horizontal, temperatura, altura geopotencial, humedad relativa), campos 2D (presión a nivel del suelo, temperatura superficial, etc.) y campos de superficie sensibles al tiempo. (3) El tercer grupo de rutinas es el Modelo WRF, que agrupa las rutinas centrales que permiten la ejecución de casos de tipo ideal y real. Se destaca el programa *Real* que toma los datos de salida del WPS y los transforma a un formato utilizable por el modelo WRF, y el programa *Wrf* (ARW model) que generar la simulación numérica del clima sobre el área deseada, por lo que se convierte en el programa con mayores requerimientos de computación. (4) Finalmente están los programas de post-procesamiento y visualización (NCAR, 2012) que actúan sobre los resultados del modelamiento.

1.1. Plataforma Avanzada de Software del WRF

La plataforma de software del WRF (WSF, WRF Software Framework) (ver Fig. 2) proporciona la infraestructura que permite el uso eficiente de una gran variedad de sistemas HPC, arquitecturas que continúan evolucionando a medida que nos adentramos en la computación de Petaescala y más allá. La arquitectura tiene capacidad para múltiples solucionadores dinámicos (*dynamic solvers*), paquetes de física que se conectan a los solucionadores a través de una interfaz de física estándar, programas para la inicialización, y el sistema de asimilación de datos variacionales WRF (WRF-Var). Hay dos solucionadores dinámicos en el WSF: el solucionador WRF de investigación avanzada (ARW, Advanced Research WRF) desarrollado principalmente en el NCAR, y el NMM (Modelo de mesoescala no-hidrostático, Nonhydrostatic Mesoscale Model) desarrollado en NCEP (Shainer *et al*., 2009).

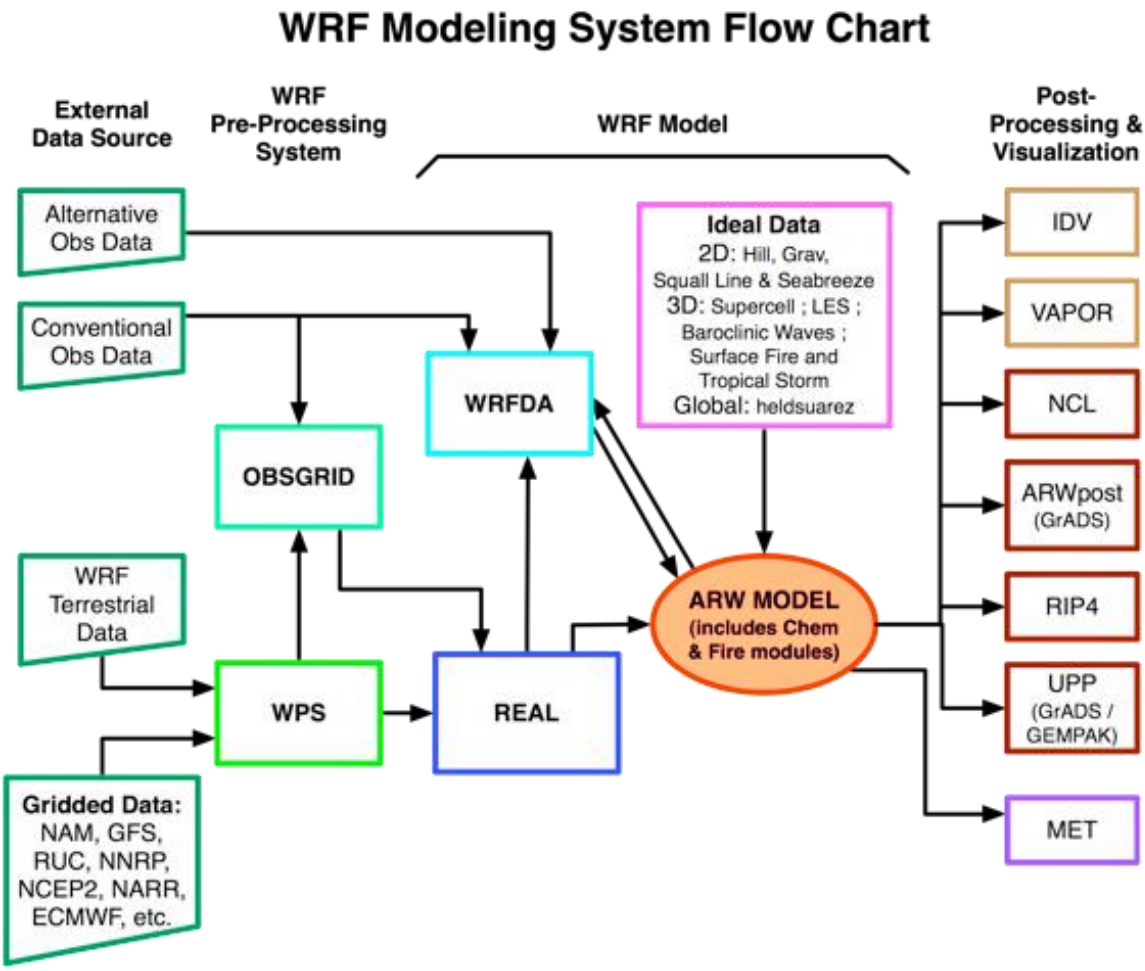

Figura 1. Diagrama de flujo del sistema de modelado WRF (NCAR, 2012).

WSF tiene un alto nivel de organización, lo que le dota de las siguientes características: es altamente modular, contiene un código fuente único para el mantenimiento, permite descomposición de dominios de dos niveles para generalidad de manejo paralelo y de memoria compartida, es portable en una serie de plataformas de computación disponibles, tiene soporte para múltiples solucionadores dinámicos y módulos de física, promueve la separación del código científico del código de paralelización y otros temas específicos de la arquitectura, establece una API de aplicaciones de E/S que permite que varios paquetes externos sean instalados con el WRF, permitiendo así que el WRF se apoye fácilmente en varios formatos de datos; además, soporta la ejecución eficiente en una amplia gama de plataformas de computación distribuida (y compartida, tipos escalares y vectoriales) que incluye soporte para aceleradores (por ejemplo, GPU) (Skamarock *et al*., 2008).

La arquitectura del WRF está organizada en 3 capas (ver Fig. 2). La capa Driver se encarga de la asignación de memoria, iniciación de anidamiento, pasos de tiempo, de la E/S, y de los bucles de tiempo a nivel superior. La capa de Mediación contiene los Solucionadores (Solver), que toman un objeto de dominio y le hace avanzar un paso a la vez, para ello cuentan con rutinas de anidamiento, interpolación, y retro-alimentación, las llamadas a paso de mensajes se encuentran aquí como parte de la rutina del solucionador. Finalmente, la capa de Modelo contiene la programación de la dinámica y física, además de las rutinas del modelo WRF que están escritas para realizar cálculos sobre una forma de tamaño arbitrario (Dudhia, 2014; Michalakes *et al*., 2004).

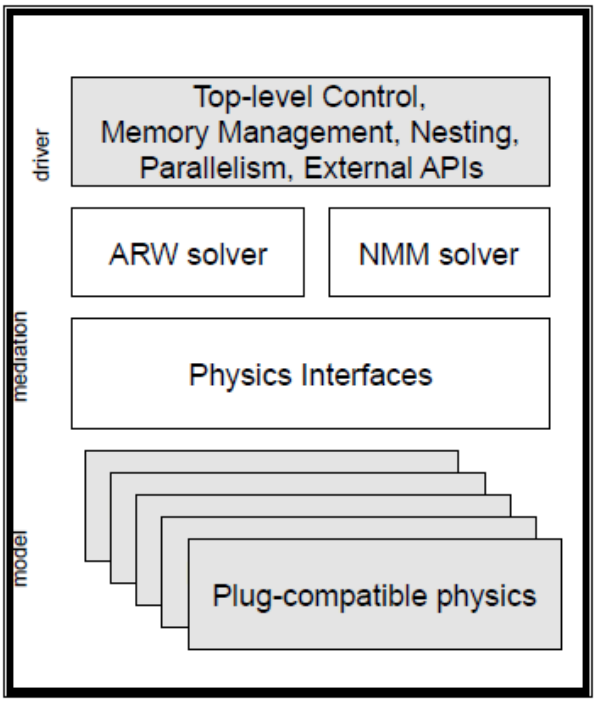

Figura 2. Plataforma de software WRF (Shainer *et al.*, 2009).

2. EVALUACIÓN DEL RENDIMIENTO DEL WRF EN EL CLÚSTER DE CEDIA

En la sección anterior se presentó información técnica relacionada con la estructura funcional del WRF, y se destaca al programa de integración numérica (*wrf.exe*) como el de mayor exigencia computacional. Por lo tanto, es el programa más importante a considerar en un estudio de evaluación del rendimiento y profiling. Los otros programas que forman parte del flujo de procesos del modelo WRF tienen requerimientos de computación relativamente pequeños en comparación con el programa *wrf*. Sin embargo, algunos de ellos también ofrecen la opción de ser ejecutados en paralelo, como es el caso del programa de inicialización *real.exe*.

Es así que, dos benchmarks de evaluación de rendimiento que giran en torno al programa *wrf.exe* son empleados. El primero consiste en ejecutar varias repeticiones de una simulación de un caso de uso seleccionado usando diferente número de procesos MPI sobre diferentes nodos, y medir el tiempo de duración de dichas simulaciones, de manera que se puede obtener un valor promedio del tiempo de simulación por cada número de procesos. Esto, a fin de evaluar el rendimiento del WRF en un entorno de memoria distribuida y principalmente evaluar la escalabilidad obtenida al usar varios nodos. El segundo benchmark, consiste en usar un programa de tipo tracer/profiler que permita profundizar la exploración del comportamiento interno del *wrf.exe* en un entorno de memoria distribuida, prestando especial énfasis en las comunicaciones, que podrían constituir un punto clave para el funcionamiento del WRF. El caso de uso que se empleará para las simulaciones relacionadas con los dos benchmarks es el Conus de 12 km.

a. Clúster de CEDIA

Los benchmarks que se describen en este artículo fueron ejecutados sobre el clúster de CEDIA, un equipo de altas prestaciones compuesto de un front-end, doce nodos de cómputo y una conexión GigaEthernet. Los nodos de cómputo poseen 2 procesadores Intel con seis u ocho núcleos y 96 GB de memoria RAM. En lo referente a software, el clúster está basado en el sistema operativo Linux y dispone de compiladores GNU e Intel para C, C++, Fortran y MPI.

b. Conus de 12 kilómetros

Este caso de uso contiene un dominio simple, de tamaño medio y cubre un rango de tiempo de 3 horas para la fecha 24 de octubre de 2001, con un paso de tiempo (time step) de 72 segundos. El periodo son las horas 25-27 (3 horas), partiendo de un archivo de restart desde el final de la hora 24. La resolución espacial es de 12 km sobre el territorio continental de Estados Unidos (ver Fig.3), de allí el nombre CONUS (Michalakes, 2008).

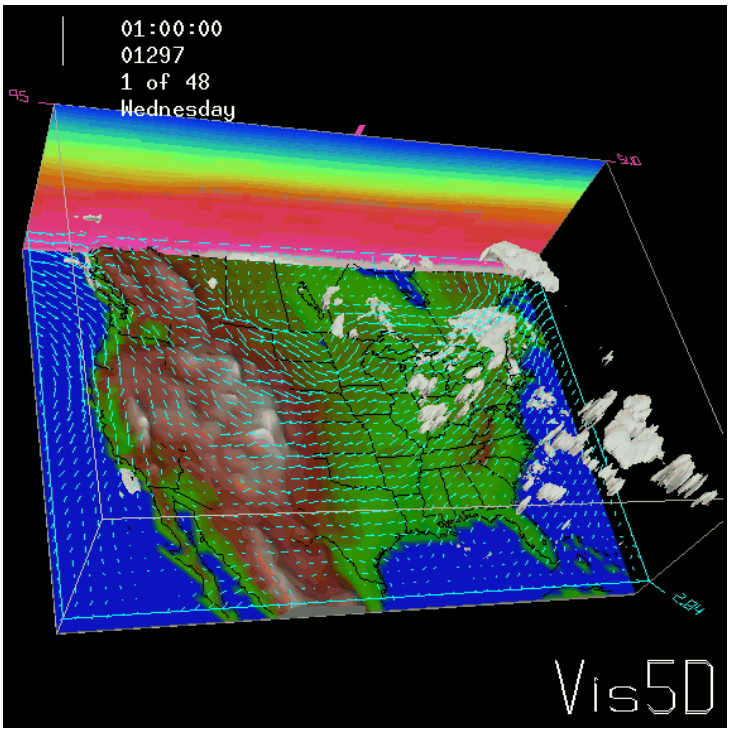

Figura 3. Captura de la animación del caso de uso Conus de 12 km (Michalakes, 2008).

c. Benchmark 1: Evaluación del rendimiento del WRF en un entorno de memoria distribuida

En este benchmark se plantea un mecanismo relativamente sencillo de evaluación del rendimiento y la escalabilidad del WRF en un entorno de memoria distribuida, mediante el uso del tiempo de ejecución de sets de simulaciones usando el comando time de Linux. El método consiste en ejecutar varias veces una misma simulación usando varios procesos MPI. De esta forma, se consigue un tiempo promedio de simulación para cada número de procesos MPI. Además también se emplea varias librerías MPI, ya que de acuerdo a Shainer et al., 2009 la implementación MPI empleada para ejecutar el modelo WRF tiene influencia directa en la eficiencia y productividad del WRF.

Para la ejecución de las pruebas de este benchmark se usó tres nodos de manera exclusiva, cada uno con dos procesadores six-core. El número máximo de procesos MPI ejecutados en cada nodo no supera el número máximo de cores físicos disponibles por el nodo. Esto debido a que en pruebas realizadas se observó que al ejecutar simulaciones del WRF en un nodo usando un número de procesos MPI que es mayor al número de cores físicos y menor que el número de cores lógicos, se producía un proceso de *oversubscription* (The Open MPI Project, 2014) que produjo un serio deterioro en el rendimiento de las simulaciones. Entonces, a fin de evitar una sobre-utilización de los recursos de cómputo y tomando en cuenta que cada uno de los nodos posee dos CPUs six-core (12 cores por nodo), el número máximo de procesos MPI que se puede utilizar es 36 (3 nodos x 12 cores por nodo).

Para la medición del tiempo de ejecución también llamada wall-clock time o wall time, se empleó el comando *time* de Linux, que debido a que no es un mecanismo de medición de gran exactitud, es considerado una fuente de incertidumbre en las mediciones. Además, como es usual en este tipo de

simulaciones, se debe considerar diferentes fuentes de incertidumbre que pueden afectar en mayor o menor medida los tiempos de simulación que se intenta medir. Por consiguiente, a fin de obtener valores estadísticamente representativos, cada simulación por número de procesos MPI se ejecutará 10 veces y se usará un promedio de los tiempos de ejecución medidos.

En la Fig. 4 se presenta los resultados de este primer benchmark que evalúa la escalabilidad del WRF sobre el clúster usando varias librerías MPI. Como se mencionó anteriormente, las tareas fueron asignadas secuencialmente entre los tres nodos empleados, usando asignaciones de tareas a nivel de núcleos, con hasta 12 tareas por nodo.

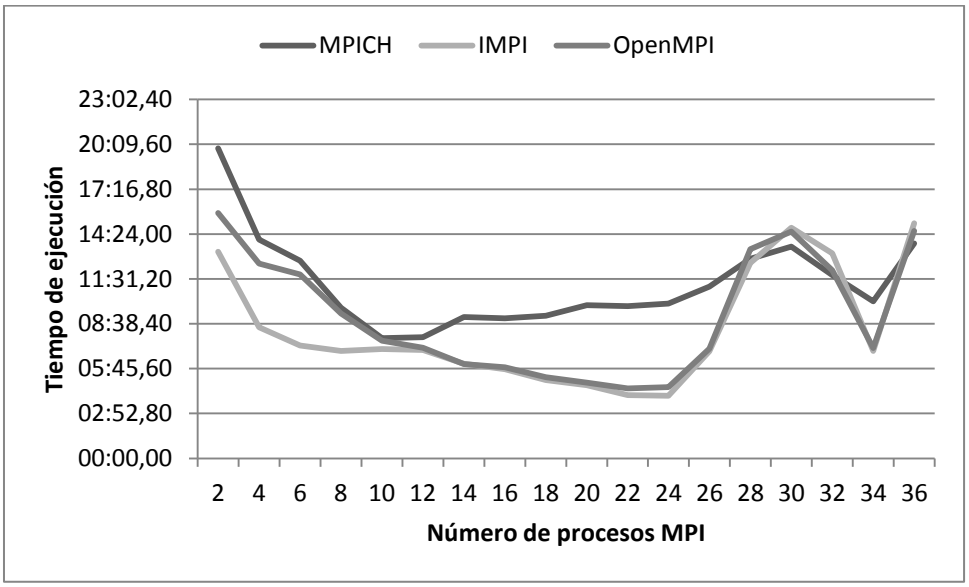

Figura 4. Tiempos de ejecución promedio para el caso de uso Conus12 km usando entre 2 y 36 procesos (3 nodos).

A partir de la Fig. 4 se puede apreciar cuatro características del benchmark de escalabilidad: (1) La librería que demuestra un mejor rendimiento es la librería Intel MPI (IMPI), (2) el rendimiento de las librerías Intel MPI y OpenMPI es bastante parecido, (3) la librería Mpich presenta el peor rendimiento global, y (4) la escalabilidad sufre un proceso de degradación a partir del uso de 24 cores. Esta escasez de escalabilidad ocurre a partir del uso de 3 nodos, y establece que la ejecución del caso de uso Conus12 km tiene una escalabilidad aceptable usando hasta dos nodos de cómputo. Usar tres o más nodos resulta en un incremento del tiempo de ejecución, en lugar de una disminución. Esta pérdida de rendimiento puede ser provocada por un cuello de botella que se manifiesta a partir del uso de 3 nodos. La comunicación es una posible causa para la pérdida de rendimiento observada. En el segundo benchmark se emplea herramientas de profiling a fin de localizar las posibles causas de pérdida de escalabilidad halladas en este benchmark.

d. Benchmark 2: Profiling de escalabilidad del caso de uso Conus 12 km en un entorno de memoria distribuida

La ejecución del primer benchmark arrojó una escasa escalabilidad del modelo WRF al usar el caso de uso Conus12 km. Por esto, el presente benchmark busca recabar mayor información de los procesos internos que se llevan a cabo durante la ejecución del modelo WRF en un entorno de memoria distribuida, con el objetivo de detectar y corregir los posibles cuellos de botella o puntos de pérdida de rendimiento. Para esto se usó ITAC (Intel ® Trace Analizer and Collector), una herramienta gráfica para la comprensión del comportamiento de aplicaciones MPI, encontrar rápidamente los cuellos de botella, y lograr un alto rendimiento para aplicaciones paralelas basadas en arquitectura Intel (Intel ®, n.d.).

A diferencia del benchmark 1, el presente no se centra en la medición del tiempo total de simulación; más bien, se enfoca en analizar los dos componentes/grupos principales del tiempo de simulación: (i) tiempo de aplicación y (ii) tiempo de rutinas MPI. El tiempo de aplicación es el tiempo durante el cual se ejecutan tareas de cálculo relacionadas con el modelo climático WRF; mientas que, el tiempo de rutinas MPI, es el tiempo durante el que se ejecutan las rutinas de la librería MPI que realizan diferentes tipos de funciones relacionadas con el intercambio de mensajes entre procesos (funciones de configuración, de comunicaciones individuales, comunicaciones colectivas, etc.). Para llevar a cabo este análisis, se ejecuta las simulaciones activando las opciones de recolección de datos de trace del ejecutable de la librería MPI de Intel (*mpirun -trace...*). Al finalizar cada simulación, se dispone de información de trace que es evaluada usando ITAC a fin de desglosar el tiempo de ejecución total en tiempo de aplicación y tiempo de rutinas MPI. ITAC proporciona esta división de tiempos por cada proceso MPI. Por lo que es necesario obtener un promedio de todos los procesos MPI involucrados por cada simulación.

Para este benchmark se empleó entre 2 y 72 procesos MPI, en hasta 6 nodos de cómputo, con un máximo de 12 procesos por nodo. Se usa un mayor número de nodos y procesos que en el benchmark anterior para poder analizar la influencia que tiene la comunicación al usar varios nodos.

Cada simulación es ejecutada usando un número diferente de procesos MPI en paralelo, y cada proceso MPI es dividido en tiempo de aplicación y tiempo de rutinas MPI. Por esta razón, se obtuvo el tiempo promedio de los tiempos de aplicación (Avg. App) y el tiempo promedio de rutinas MPI (Avg. MPI) para cada simulación. Aunque el caso de uso es el mismo del benchmark anterior, existen diferencias en los tiempos de ejecución total obtenidos debido al overhead impuesto por el tracer. Además de las medidas de tiempo mencionadas, se usa dos razones de proporcionalidad, una para la evaluación del decrecimiento del tiempo de aplicación (Speedup App), y otra para el crecimiento del tiempo MPI (Slowdown MPI). Dichos indicadores son calculados a partir de una simulación referencial, que para este caso es la simulación en dos procesos MPI. De manera que, "Speedup App" viene dado por el tiempo promedio de aplicación de la simulación usando dos procesos dividido para el tiempo promedio de la aplicación evaluada (AvgApp2/AvgApp). Mientras que, "Slowdown MPI" es el tiempo promedio MPI de la simulación evaluada divido para el tiempo promedio MPI de la simulación de referencia (AvgMPI/AvgMPI2).

Los resultados son representados en la Fig. 5, usando un gráfico de líneas que ilustra los indicadores de aceleración/ralentización, en la Fig. 6 a través de un gráfico de área en stack con los promedios de los dos grupos de tiempos (aplicación y MPI) y en la Fig. 7 con un gráfico de área en stack con los porcentajes de los dos grupos. En la Fig. 5 se identifica dos partes de diferente comportamiento (separados por una línea entrecortada): la primera entre las simulaciones con 2 hasta 24 procesos, y la segunda a partir de la simulación que usa 24 procesos. La primera parte muestra una tendencia de crecimiento lineal para el indicador de aceleración de las funciones de aplicación; a su vez, la pendiente de la tendencia de crecimiento del indicador de ralentización de las funciones MPI, es pequeña y no muestra un comportamiento muy variante. Esto se podría considerar como un comportamiento deseable, ya que indica que la mejora que es obtenida en la optimización del tiempo de aplicación es lo suficientemente grande como para que el overhead de comunicación no sea significativo.

Sin embargo, la segunda parte muestra un comportamiento muy diferente, un más oscilatorio para las dos tendencias y con un crecimiento grande y oscilatorio para la tendencia del indicador de ralentización de las funciones MPI, que como se puede observar en las Figs. 6 y 7 se convierte en el componente predominante a partir de la simulación que usa 24 procesos. El comportamiento de la segunda parte es totalmente indeseable, ya que indica que a partir del proceso 24 el overhead de comunicación -implícito en las funciones MPI- empieza a crecer exageradamente, al punto de ser un factor que entorpece gravemente la paralelización del modelo. De manera que, en lugar de resultar en una reducción, resulta en un incremento del tiempo de simulación.

El gráfico de la Fig. 6 muestra que el tiempo de aplicación tiene un decrecimiento exponencial como es usual al dividir las tareas de procesamiento en tareas paralelas. Sin embargo, el crecimiento caótico y exagerado del tiempo de ejecución de las rutinas MPI a partir de la simulación con 24 procesos, es el factor clave de la degradación del rendimiento, ya que pasa de ser un factor menor (en las simulaciones con procesos 2-24), a ser un factor dominante (en las simulaciones con 26 procesos).

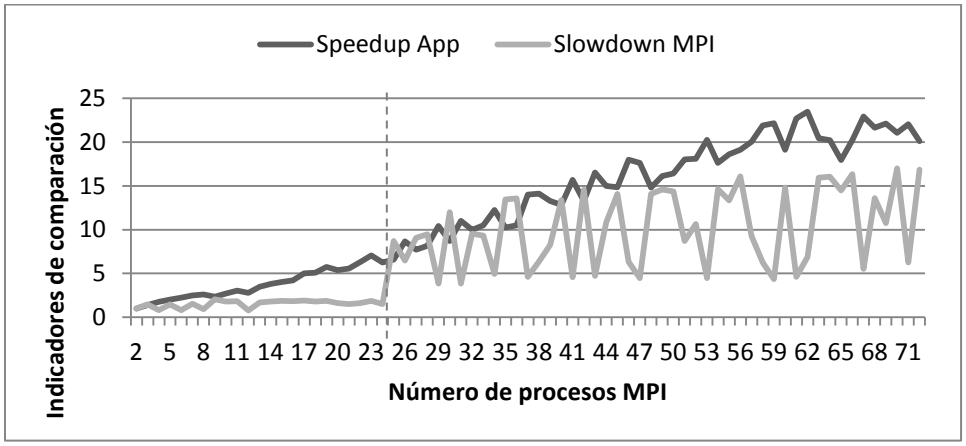

Figura 5. Indicadores de comparación de las funciones de aplicación y MPI para el caso de uso Conus12 km.

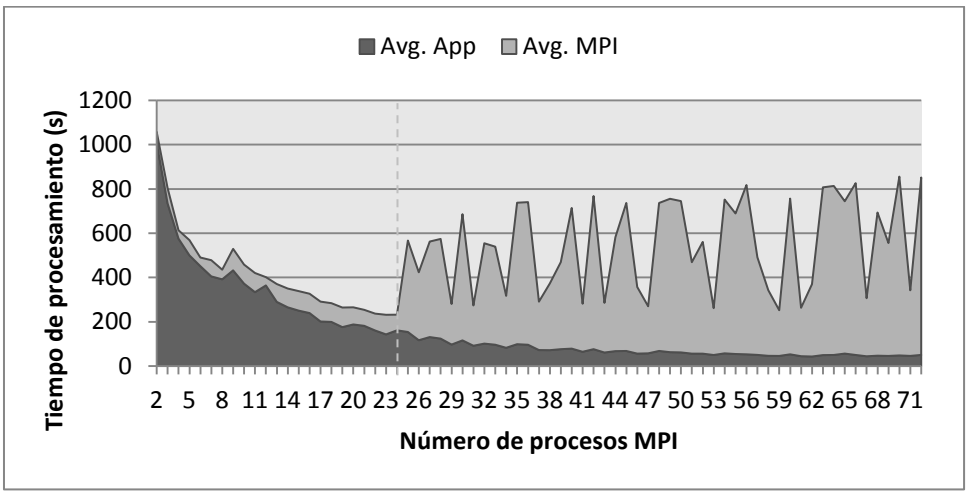

Figura 6. Tiempo de procesamiento promedio en stack de los grupos de funciones de aplicación y MPI para las simulaciones del caso de uso Conus12 km.

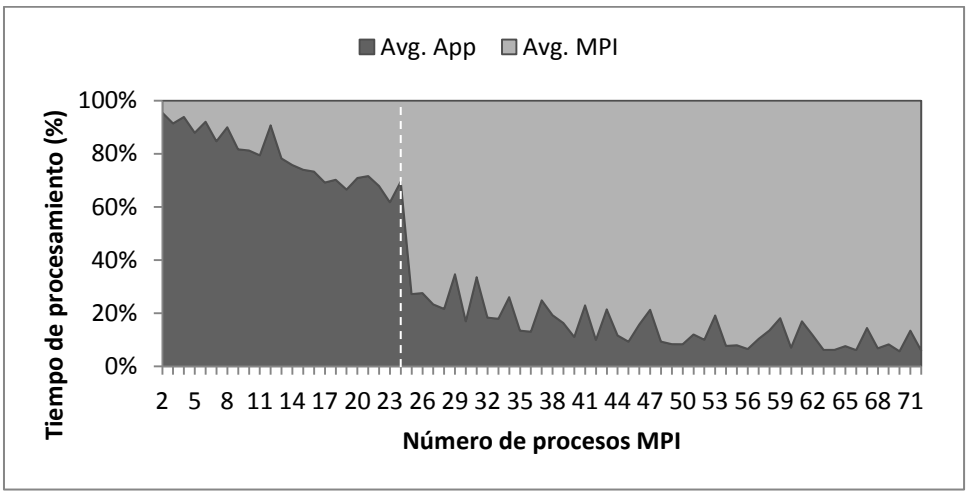

Figura 7. Tiempo de procesamiento en porcentaje para los grupos de funciones de Aplicación y MPI para las simulaciones correspondientes al caso de uso Conus12 km.

La matriz de comunicación (ver Fig.8) entre los procesos MPI muestra el patrón de comunicación del WRF (HPC Advisory Council, 2012) para una simulación que utiliza 24 procesos distribuidos en 2 nodos. Dicha figura representa las comunicaciones punto a punto entre procesos MPI, donde los procesos del eje vertical son los emisores y los del eje horizontal son los receptores. El volumen total de datos transmitidos entre 2 procesos individuales está entre 1 y 3 GB, y en total la comunicación global alcanza 56.93 GB (ver Fig.10). Esto confirma el gran requerimiento de comunicación existente. No obstante, las comunicaciones críticas son las que ocurren entre diferentes nodos (comunicación inter-nodo), las cuales, en la Fig. 13, están enmarcadas en dos recuadros que en suma alcanzan aproximadamente 8 GB, cantidad que podría ser considerada exigente para una red GigaEthernet.

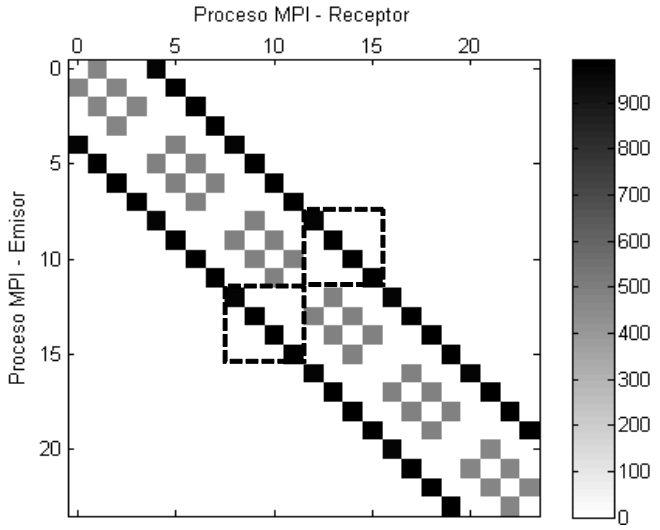

Figura 8. Matriz de comunicación entre procesos MPI para el caso de uso Conus12 km usando 24 procesos. Se resalta en dos recuadros las zonas de comunicación inter-nodo.

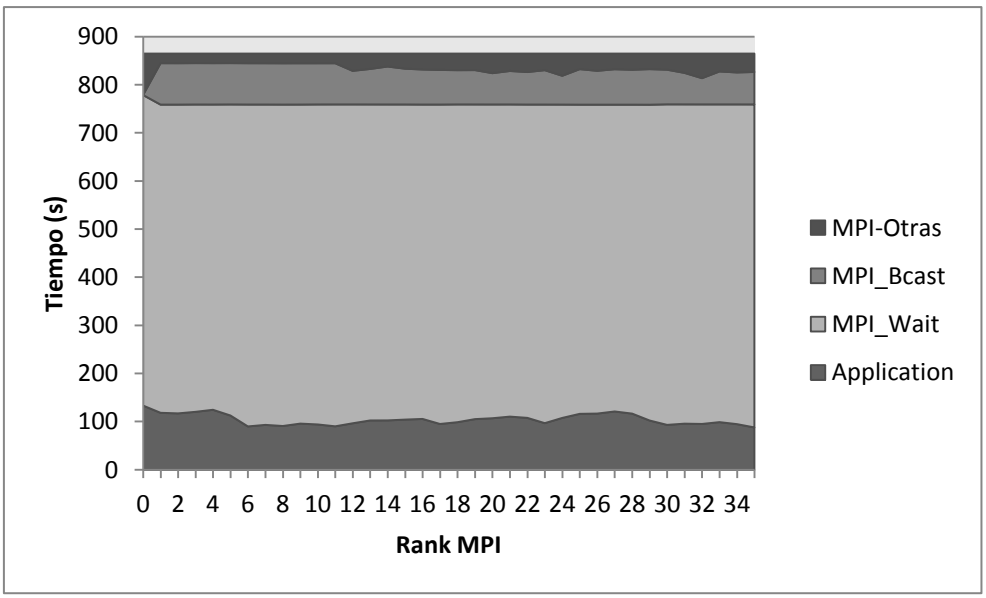

Figura 9. Tiempo de aplicación y funciones MPI para el Conus12 km usando 36 procesos MPI.

Para reforzar el análisis que atribuye la degradación de la comunicación a la comunicación al usar 3 o más nodos, en la Fig. 10 se identifica un desglose de los tiempos de simulación en tiempos de aplicación y llamadas a rutinas MPI, para una simulación de 36 procesos MPI en 3 nodos, que constituye la situación referente para la pérdida de rendimiento mencionada anteriormente en esta sección. Al usar 3 nodos, la función MPI_Wait se convierte en la dominante absoluta, dejando muy

por debajo al tiempo de aplicación, que debería ser el tiempo de procesamiento predominante. Dado que, la función MPI_Wait espera a que una tarea de envío-recepción no-bloqueante se complete (LLNL, 2003), esto implica que a partir del uso de 3 o más nodos de cómputo se presenta un retardo en las comunicaciones individuales que afecta de sobremanera al desempeño general del modelo, debido al costo de las comunicaciones punto a punto.

Para profundizar el análisis del cuello de botella de comunicación, se incluye un gráfico (ver Fig.10) que ilustra el volumen total de información transferida en las comunicaciones punto a punto entre procesos MPI. Se desglosa el volumen total en volumen de datos que se transfieren mediante comunicaciones intra-nodo y el volumen de datos inter-nodo.

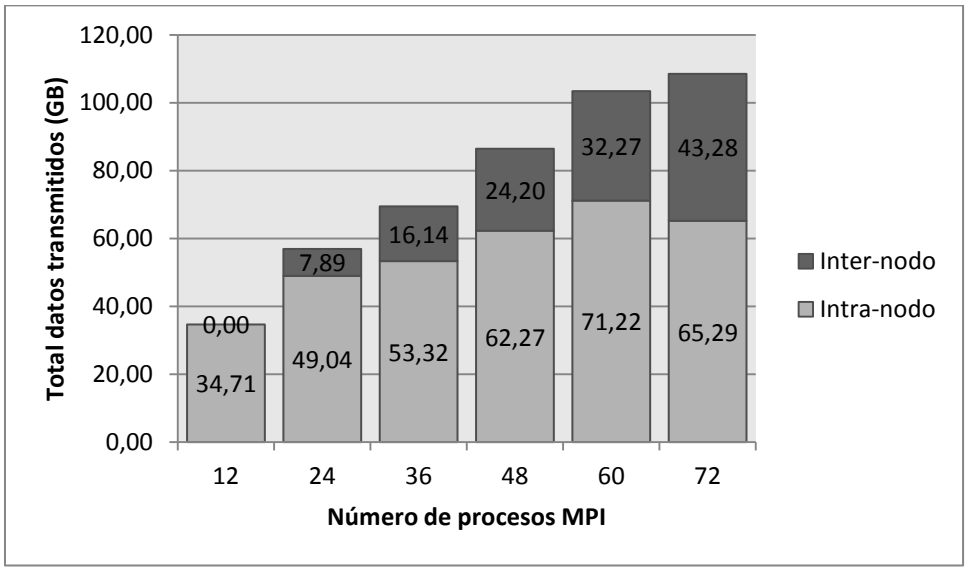

Figura 10. Cantidad total de datos trasmitidos en comunicaciones punto a punto entre procesos MPI para el caso de uso Conus12 km, divida para los dos tipos de comunicaciones posibles: intra-nodo e inter-nodo.

La Fig. 10 muestra que el total de datos enviado en comunicaciones individuales se incrementa proporcionalmente con el número de procesos MPI. Es decir, a medida que se usa más nodos y por ende más procesos MPI, la carga para las comunicaciones se incrementa proporcionalmente, haciendo que la escalabilidad del modelo requiera de un ancho de banda eficiente que permita manejar adecuadamente la gran carga de comunicación. Este incremento en las exigencias de comunicación aplica tanto a las comunicaciones intra-nodo, como a las comunicaciones inter-nodo.

En resumen, los resultados presentados en este benchmark contribuyen a afirmar que el cuello de botella que ocasiona la falta de escalabilidad del WRF para el caso de uso Conus12 km en el clúster de CEDIA, es la comunicación, que no da cabida suficiente a los grandes requerimientos presentados por el WRF. El rendimiento del WRF no escala al usar más de dos máquinas al usar una red GigaEthernet. Esto concuerda con los resultados obtenidos en (HPC Advisory Council, 2012).

3. CONCLUSIONES

En el presente artículo se ejecutó dos benchmarks del WRF usando el caso de uso Conus de 12 km, para evaluar el rendimiento y escalabilidad del modelo en un clúster de alto rendimiento. A partir de la información técnica recabada y de las primeras pruebas realizadas, se determinó que el ejecutable más importante del modelo de pronóstico climático WRF, es el ejecutable de integración numérica denominado wrf.exe. Por consiguiente, los dos benchmarks empleados se centraron en la ejecución de simulaciones de dicho programa. El primer benchmark consistió en medir el tiempo de procesamiento

de varias simulaciones usando entre 2 y 36 procesos MPI sobre 3 nodos, asignando hasta 12 procesos por nodo. A su vez, el segundo benchmark consistió en analizar mediante el profiler ITAC los tiempos de aplicación y tiempos de simulación de una serie de simulaciones que usaron entre 2 y 72 procesos MPI en 6 nodos de cómputo, asignando hasta 12 procesos por nodo. La razón por la que se usó un mayor número de nodos en el segundo benchmark, fue para profundizar el análisis del efecto de la comunicación en la escalabilidad del modelo.

Los benchmarks demostraron que el modelo WRF no es escalable al usar tres o más nodos de cómputo en el clúster de CEDIA, debido a un cuello de botella en la comunicación. Usando la información arrojada por el profiler ITAC, se pudo establecer que esto se produce por la gran cantidad de tiempo consumido por las funciones MPI, al emplear 3 o más nodos. En particular la función MPI más representativa fue la función MPI_Wait, que espera a que una tarea de envío o recepción no-bloqueante se complete. Por lo tanto, son las comunicaciones las que están tardando más de la cuenta, y por ende, ralentizando de sobremanera el tiempo total de procesamiento. Esto fue más evidente al analizar la carga de comunicación que presentó un comportamiento creciente lineal a medida que se usó más nodos de cómputo. Así, por ejemplo, al usar un nodo de cómputo (12 procesos) la carga total de comunicación fue de 34.71 GB; mientras que, al usar 6 nodos de cómputo (72 procesos), la carga de comunicación llegó a l08.56 GB. En conclusión, la red GigaEthernet con la que cuenta actualmente el clúster de CEDIA, no es suficiente para aplicaciones de gran exigencia de comunicación como el WRF. Este resultado concuerda con el del estudio del Consejo de Asesoría HPC (2012).

Se propone como trabajo futuro ejecutar los benchmarks en un clúster que cuente con una red de comunicación de mayores prestaciones. De manera que, se pueda contrastar los resultados obtenidos en este estudio.

AGRADECIMIENTOS

El presente manuscrito fue desarrollado en el contexto del proyecto “Integración de computación de altas prestaciones y manejo de grandes volúmenes de datos al análisis y predicción de clima en el austro ecuatoriano” financiado por el Departamento de Investigación de la Universidad de Cuenca (DIUC). Los autores desean expresar gratitud al Consorcio Ecuatoriano para el Desarrollo de Internet Avanzado (CEDIA), por haber proporcionado las facilidades de acceso al clúster que sirvió como recurso de cómputo base para las pruebas de rendimiento realizadas.

BIBLIOGRAFÍA

Dudhia, J., 2014. WRF modeling system. Overview. Disponible en *http://www2.mmm.ucar.edu/wrf/users/tutorial/201201/WRF_Overview_Dudhia.ppt.pdf*, 37 pp.

HPC Advisory Council, 2012. Weather Research and Forecasting (WRF). Performance Benchmark and Profiling. Disponible en *http://www.wrf-model.org/index.php*.

Intel ®, n.d. Intel® Trace Analyzer and Collector [WWW Document]. URL *https://software.intel.com/en-us/intel-trace-analyzer*.

LLNL, 2003. MPI_Wait [WWW Document]. URL *https://computing.llnl.gov/tutorials/mpi/man/MPI_Wait.txt*.

Michalakes, J., 2008. WRF V3 Parallel Benchmark Page [WWW Document]. URL *http://www2.mmm.ucar.edu/wrf/WG2/benchv3/*.

Michalakes, J., J. Dudhia, D. Gill, T. Henderson, J. Klemp, W. Skamarock, W. Wang, 2004. The weather research and forecast model: Software architecture and performance. Disponible en *http://opensky.library.ucar.edu/collections/OSGC-000-000-009-711*.

Michalakes, J., R. Loft, A. Bourgeois, 2001. Performance-portability and the Weather Research and Forecast Model. Disponible en *http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.8.3781*.

NCAR, 2012. ARW Version 3 modeling System User's Guide. Disponible en *http://www2.mmm.ucar.edu/wrf/users/docs/user_guide/ARWUsersGuide.pdf_sav*, 184 pp.

Shainer, G., T. Liu, J. Michalakes, J. Liberman, J. Layton, Jeff, O. Celebioglu, Scot A. Schultz, J. Mora, D. Cownie, 2009. Weather Research and Forecast (WRF) Model Performance and Profiling Analysis on Advanced Multi-core HPC Clusters. Disponible en *http://www.linuxclustersinstitute.org/conferences/archive/2009/PDF/Shainer_64557.pdf*, 14 pp.

Skamarock, W., J.B. Klemp, J. Dudhia, D.O. Gill, D.M. Barker, M.G. Duda, X.-Y. Huang, W. Wang, J.G. Powers, 2008. A description of the Advanced Research WRF Version 3. Disponible en *http://opensky.library.ucar.edu/collections/TECH-NOTE-000-000-000-855*, 125 pp.

The Open MPI Project, 2014. FAQ: Running MPI jobs [WWW Document]. *URL http://www.open-mpi.org/faq/?category=running#oversubscribing.*

WRF, n.d. The Weather Research & Forecasting Model Website. [WWW Document]. *URL http://www.wrf-model.org/index.php.*