Asociación Argentina
de Mecánica Computacional

Mecánica Computacional Vol XXVII, págs. 2927-2941 (artículo completo)
Alberto Cardona, Mario Storti, Carlos Zuppa. (Eds.)
San Luis, Argentina, 10-13 Noviembre 2008

LABORATORIO DE SIMULACIÓN NUMÉRICA DE FLUJOS A SUPERFICIE LIBRE

Sergio Nesmachnow[a] **y Gabriel Usera**[b]

[a]*Centro de Cálculo, Instituto de Computación, Facultad de Ingeniería, Universidad de la República, Uruguay, sergion@fing.edu.uy, http://www.fing.edu.uy/inco/grupos/cecal/hpc*
[b]*Instituto de Macánica de los Fluidos e Ingeniería Ambiental, Facultad de Ingeniería, Universidad de la República, Uruguay, gusera@fing.edu.uy, http://www.fing.edu.uy/imfia*

Palabras clave: Mecánica computacional, computación de alto desempeño.

Resumen. Este trabajo presenta las actividades llevadas a cabo en el marco del proyecto *Laboratorio de Simulación Numérica de Flujos a Superficie Libre*, desarrollado en la Facultad de Ingeniería, Universidad de la República, Uruguay. Los objetivos principales del proyecto consisten en i) instrumentar una plataforma computacional que permita mejorar decisivamente la capacidad de abordar en el ámbito nacional problemas de diseño de muelles, pilas y estribos de puentes, el análisis de impacto ambiental de obras hidráulicas en ríos y estuarios, y el diseño y maniobrabilidad de barcazas en áreas restringidas de navegación; y ii) adaptar un modelo numérico tridimensional de simulación de flujos, desarrollado en el Instituto de Mecánica de los Fluidos e Ingeniería Ambiental de la Facultad de Ingeniería, para la simulación de flujos a superficie libre, utilizando la capacidad computacional provista mediante la implantación de un cluster multiprocesador. En este artículo se presentan las actividades de diseño, instalación y configuración del cluster multiprocesador y se describen las aplicaciones basadas en el modelo `caffa3d.MB`, que implementa el método de volúmenes finitos aplicado a la simulación numérica de flujos viscosos o turbulentos, tridimensionales y con transporte de escalares, utilizando mallas curvilíneas estructuradas por bloques. La implementación del modelo incorpora el uso de la computación paralela en ordenadores multiprocesadores mediante directivas de OpenMP, aprovechando la descomposición del dominio en bloques de malla. En los resultados preliminares se han verificado factores de rendimiento de 1.6x en computadores de doble procesador, y en la actualidad se está trabajando en el diseño de una versión del modelo capaz de ejecutar sobre una plataforma de memoria distribuida para aprovechar la escalabilidad incremental de la plataforma computacional implementada.

1. INTRODUCCIÓN

La simulación numérica consiste en la recreación de un proceso natural en intervalos discretos de tiempo, mediante la utilización de modelos matemáticos que reflejan propiedades relevantes del fenómeno en cuestión. Las ecuaciones de Navier-Stokes son la base de los modelos matemáticos en fluidodinámica, pero por su complejidad se requiere un gran poder de cómputo para llevar a cabo simulaciones para escenarios realistas. Tradicionalmente, la disponibilidad de poder de cómputo imponía limitaciones a la dimensión de los problemas a abordar mediante simulación numérica, así como a los tiempos de simulación y a la precisión de los resultados obtenidos. Sólo los investigadores con acceso a un supercomputador disponían del poder de cálculo para simular sobre escenarios realistas y para largos períodos de tiempo. En los últimos años, con la masiva popularización de los computadores de bajo costo, los investigadores han logrado obtener a un costo relativamente bajo el poder de cómputo necesario para experimentar con técnicas de simulación numérica para problemas complejos en escenarios realistas. Esta oportunidad abre nuevas perspectivas en nuestros países, donde la financiación limitada usualmente condiciona el tipo de experimentos necesarios para modelar problemas de gran porte, y del mismo modo, restringe el acceso a recursos computacionales de alto costo.

Este artículo presenta las actividades desarrolladas en el proyecto *Laboratorio de Simulación Numérica de Flujos a Superficie Libre*, que se propuso en el año 2006 con el objetivo de instrumentar una plataforma computacional que permitiera mejorar decisivamente la capacidad de abordar en el ámbito nacional problemas de diseño de muelles, pilas y estribos de puentes, el análisis de impacto ambiental de obras hidráulicas en ríos y estuarios, y el diseño y maniobrabilidad de barcazas en áreas restringidas de navegación. La plataforma computacional corresponde a una arquitectura cluster formada por componentes de bajo costo. Para modelar los problemas involucrados se propuso adaptar un modelo numérico tridimensional para la simulación de flujos a superficie libre. El resto del artículo se organiza del modo que se describe a continuación. La sección 2 presenta la motivación del proyecto y sus objetivos principales. Las actividades de diseño, instalación y evaluación de la plataforma computacional se describen en la sección 3, que constituye el principal aporte del trabajo. La sección 4 presenta el modelo numérico desarrollado en el marco del proyecto y ejemplos de su aplicación. Por último, la sección 5 presenta las perspectivas de trabajo actual y futuro, y la sección 6 breves conclusiones sobre la experiencia de trabajo.

2. LABORATORIO DE SIMULACIÓN NUMÉRICA

En los últimos años, el vertiginoso desarrollo de la capacidad de cálculo de los computadores y la drástica reducción de su costo, ha creado la oportunidad para abordar un número cada vez mayor de temáticas prácticas mediante la simulación numérica. En el área de mécanica de los fluidos computacional, la simulación de flujos a superficie libre ha tenido un notorio desarrollo por su capacidad para modelar problemas relacionados con el sector económico del transporte. Entre las aplicaciones de mayor relevancia en el ámbito nacional, pueden mencionarse: a) el diseño de muelles, puertos y de pilas y estribos de puentes en su interacción con el cuerpo de agua, desde el punto de vista de la propia obra y del efecto sobre el medio ambiente (e.g. arrastre de sedimentos, erosión local, etc.) y b) el diseño de vehículos marinos y fluviales de transporte de cargas y el análisis de su comportamiento en zonas de navegación restringidas. En esta línea temática, el proyecto *Laboratorio de Simulación Numérica de Flujos a Superficie Libre* fue planteado en el año 2006, proponiendo como principal objetivo instrumentar una plataforma computacional para abordar mediante simulación numérica dos casos específicos:

- El estudio de los patrones de flujo en torno a pilas y estribos de puentes, un tema de suma importancia considerando el crecimiento y la rehabilitación de la infraestructura vial y portuaria del país, propulsado por importantes niveles de inversión (i.e. el convenio entre el Ministerio de Transporte y Obras Públicas y el Banco Mundial, con un monto global de U$S 100 millones, para la expansión y rehabilitación de los muelles y marinas de varios puertos sobre el Río de la Plata; y la proyectada terminal granelera en el puerto de La Paloma, valorada en U$S 25 millones). Un elemento común en estas grandes obras portuarias es la presencia de la interacción entre cuerpo fluido y obra (apoyos de pilas y estribos de puentes, muelles, etc.); el proyecto propone estudiar el caso del flujo de agua en torno a pilas y estribos de un puente y su relación con la erosión local provocada.

- Los efectos de sobre calado (*squat*) y *wake wash* en buques de carga que navegan en aguas poco profundas, fenómenos con importantes implicaciones económicas, de seguridad y medioambientales, y que afectan al tráfico de pasajeros y mercaderías en rutas que se desarrollan total o parcialmente en zonas de navegación restringidas. En el Uruguay, este tipo de transporte marítimo de cargas ha crecido significativamente en los últimos años, alcanzando el 37 % del PBI del sector y un 2.5 % del PBI nacional. Los buques realizan gran parte de su navegación en aguas someras; las olas supercríticas que generan son de pequeña amplitud en comparación con las de una tormenta, pero tienen un gran período, acumulando mucha energía y siendo muy poco dispersivas. Estas olas inciden sobre la orilla, aumentando rápidamente de amplitud y de poder de penetración cuando el barco ya ha pasado, produciendo daños en infraestructuras costeras y medioambiente y generando un fuerte movimiento horizontal en las partículas finas en el lecho marino y fluvial, con su incidencia ecológica. Existe asimismo evidencia documental de que el *squat* ha sido la razón básica de cierto tipo de accidentes marinos, con considerables costos económicos y elevados costos humanos y ecológicos, y que la pérdida de maniobrabilidad es causa de accidentes provocados por la colisión de trenes de gabarras contra infraestructuras viales.

En el campo computacional, la reciente generación de procesadores con capacidad de ejecución de múltiples hilos y arquitectura multinúcleo ha propuesto nuevos desafíos y posibilidades para explotar la computación paralela. Sin embargo, el costo de paquetes numéricos comerciales para simulación numérica en mécanica de los fluidos computacional continúa siendo excesivo para nuestro medio y no ha acompañado la baja de costos del hardware. Las licencias para uso comercial/industrial de paquetes como CFX-5 o Fluent superan los U$S 30.000 anuales (las licencias académicas son más económicas, pero es ilegal su utilización con propósito comercial/industrial). Por estos motivos, existe una oportunidad para desarrollar modelos numéricos que permitan a organismos y empresas nacionales acceder a las tecnologías de computación de alto desempeño, para mejorar su competitividad al tener acceso a un servicio tecnológico como el mencionado.

El proyecto que se describe propuso como principal objetivo instrumentar una plataforma computacional que permitiera mejorar decisivamente la capacidad de abordar en el ámbito nacional los problemas presentados, utilizando un modelo de simulación numérica de flujos viscosos o turbulentos, tridimensionales y con transporte de escalares. Se propuso validar experimentalmente los resultados de los modelos numéricos mediante ensayos en modelos físicos sobre el canal de pruebas y el canal de olas, complementando las actividades de simulación numérica de flujos a superficie libre y el análisis experimental.

3. EL CLUSTER MEDUSA

Esta sección presenta los detalles sobre el diseño, instalación y evaluación de desempeño del cluster Medusa, la plataforma computacional implementada en el marco del proyecto.

3.1. Evaluación de alternativas

A continuación se describen las alternativas analizadas para el diseño e instalación del cluster Medusa. Se presentan los aspectos evaluados para la elección de elementos de procesamiento, tecnología de interconexión y software de uso general, tomando en cuenta la restricción de financiación impuesta por el proyecto (un monto de U$S 10.000 para la totalidad de la infraestructura computacional).

3.1.1. Procesadores

El poder de procesamiento es el elemento primordial en un cluster dedicado al cálculo intensivo. El análisis de las opciones de procesadores se concentró en varios aspectos: el número de nodos de ejecución (que condiciona el espacio físico de instalación, el consumo energético y el costo efectivo de adquisición, operación y mantenimiento), la velocidad de reloj y registros manejados por el procesador (que condicionan el desempeño individual de cada nodo de cómputo, y afectan el rendimiento del cluster en su conjunto), el tamaño de la memoria física y virtual (que determina la capacidad de manejo de datos) y el diseño y la velocidad del bus del sistema (que permiten evaluar la velocidad de procesamiento efectivo de la arquitectura, tratando de evitar cuellos de botella en las transferencias de datos). Tomando en cuenta las condiciones de presupuesto restringido del proyecto, el análisis incluye el precio efectivo de cada una de las alternativas en el mercado, orientándose a determinar la opción con mejor relación precio/desempeño. A continuación se presentan las principales características de los procesadores evaluados. La comparación se limita a los procesadores con arquitecturas de 64 bits de mayor uso en clusters de alto desempeño, según el ranking de supercomputadores TOP 500 de julio de 2006: Itanium, Opteron, UltraSparc y Power (TOP500, 2006).

- Los procesadores Itanium de Intel utilizan paralelismo explícito a nivel de instrucción, dejando al compilador las decisiones sobre las instrucciones a ejecutar en paralelo. Itanium 1 vio la luz en 2001, pero debido a su pobre desempeño solo compitió por el segmento bajo del mercado. En 2002, Intel y HP presentaron Itanium 2, orientándose a servidores de procesamiento masivo. Aunque un supercomputador con Itanium 2 alcanzó el #2 en el TOP 500 en 2004, la arquitectura no logró gran difusión. Al momento de instalar el cluster Medusa, el costo de un procesador Itanium de 3GHz. en Uruguay estaba en el entorno de U$S 4000.

- Opteron, fabricado en 2003 por AMD, superó las prestaciones de sus predecesores y contribuyó al fracaso de Itanium. Opteron se destaca por la ejecución nativa de código de 32 bits sin degradar su desempeño, el direccionamiento de hasta 40 bits de memoria física, el bajo consumo energético y el mecanismo optimizado de transferencia de datos (tecnología *hypertransport*) que fue un gran avance respecto a modelos previos. Opteron posee 9 unidades de ejecución, 3 de ellas de punto flotante, y tiene el controlador de memoria integrado y diseñado para disminuir retardos de acceso. Entre las opciones evaluadas al momento de instalar el cluster Medusa, Opteron era la línea de más bajo consumo energético y de precio más accesible: un computador con procesador Opteron doble núcleo de 2.2 GHz. tenía un costo de U$S 1400.

- UltraSparc IV fue presentado por SUN en 2004, basado en una arquitectura de dos núcleos, un controlador de memoria integrado para reducir los tiempos de acceso a datos, y memoria caché de nivel 2 externa (el acceso es más lento que en arquitecturas con caché integrada). Los componentes se conectan por un bus Sun Fire Plane. El UltraSparc IV+ incorporó caché interna e incrementó la velocidad de los núcleos hasta 2GHz. Al momento de instalar el cluster Medusa, el costo de un servidor Sun FIRE V490 doble núcleo UltraSparc IV de 2.1 GHz. era de U$S 10000.

- La arquitectura Power de IBM se presentó en 2001 con dos núcleos de ejecución con un reloj de 1.9 GHz. En 2004 el procesador Power 5 incrementó el tamaño de caché e integró el controlador de memoria para mejorar el acceso a datos. Dos controladores (Fabric y GX Controller) manejan la comunicación entre el procesador y los otros componentes del sistema. La línea Power posee el mayor poder de cómputo de las arquitecturas evaluadas: 8 unidades de ejecución por núcleo, 2 de ellas para operar con punto flotante, y tiene el bus con el mayor ancho de banda, alcanzando 35 Gb/s. Como contrapartida, tiene el mayor consumo energético. Al momento de instalar el cluster Medusa, un equipo IBM 9111-520 doble núcleo Power 5 de 1.9 GHz. tenía un costo de U$S 9000.

Se estudiaron las prestaciones reales de los procesadores evaluados, analizando los resultados de la ejecución de benchmarks estándar sobre clusters de alto desempeño, disponibles públicamente. Se evaluó el desempeño de la CPU en operaciones con números enteros y de punto flotante y número variable de núcleos de procesamiento, la velocidad de acceso a memoria y manejo de datos, y la capacidad de ejecución de una aplicación realista de mécanica de los fluidos computacional (modelada por el benchmark Fluent). El análisis reveló que Power 5 es la arquitectura más eficiente al operar con números de punto flotante, seguida de Itanium 2 y Opteron, mientras que UltraSparc IV tiene el peor desempeño. Opteron y Power 5 tienen las mejores velocidades de acceso a memoria y de manejo intenso de I/O, y también presentan el mejor desempeño en la ejecución de Fluent. Itanium 2 y UltraSparc IV aparecen como opciones menos competitivas. Los detalles numéricos de la comparación de eficiencia se encuentran disponibles en el informe de Salsano (2007).

Si el precio no fuera un factor relevante, Power 5 sería la mejor elección para implementar un cluster de alto desempeño. UltraSparc se descartó por su pobre eficiencia en operaciones de punto flotante. Itanium 2 y Opteron permitían alcanzar valores aceptables de eficiencia a un costo razonable, requiriendo Opteron un menor consumo energético. Tomando en cuenta estos factores, se decidió adoptar la arquitectura Opteron para el cluster Medusa, por presentar el mejor compromiso entre costo y desempeño. La decisión está avalada por estudios previos y contemporáneos, y por el importante número de clusters Opteron en TOP 500.

3.1.2. Tecnologías de interconexión

La tecnología de interconexión es un factor crítico en el desempeño de un computador distribuido. Para el diseño del cluster Medusa se evaluaron las tecnologías de interconexión más populares para un cluster de alto desempeño: Gigabit Ethernet, Myrinet, SCI, Quadrics e InfiniBand. Los factores analizados fueron la latencia (el tiempo de enviar un mensaje de tamaño nulo), el ancho de banda (que determina la velocidad de transferencia de datos), la escalabilidad (que evalúa la capacidad de incrementar el número de nodos sin afectar el desempeño de las comunicaciones) y el costo de implantación (basado en estimaciones propias del mercado y datos del trabajo de Bode et al. (2004)).

Gigabit Ethernet se destaca por ser una solución de bajo costo que logra un ancho de banda de 1 Gb/s sobre cables UTP cat 6 y fibra óptica. Se recomienda su uso para clusters de hasta 24 nodos, ya que para clusters de mayor tamaño se requieren switches costosos. La escalabilidad de Gigabit Ethernet está limitada por el overhead impuesto por el protocolo TCP/IP, que impide agregar múltiples switches para incrementar el ancho de banda (el límite para un cluster con Gigabit Ethernet es de 256 nodos, aunque las comunicaciones se degradan al superar los 64 nodos (Chen et al., 2004)). Al momento de instalar el cluster Medusa el costo de una interconexión basada en Gigabit Ethernet era menor a U$S 50 por nodo, sin embargo el valor unitario decrece notoriamente para clusters de mediano tamaño, teniéndose un tope de costo de aproximadamente U$S 300 para 16 nodos.

El relevamiento realizado para evaluar las tecnologías estudiadas indicó que las alternativas de alto desempeño (Myrinet, SCI, Quadrics y InfiniBand) permiten alcanzar mayores velocidades de interconexión, pero requieren de hardware específico. Como consecuencia, su costo de instalación es muy elevado (entre U$S 2000 y U$S 4000 al momento de instalar el cluster Medusa, quedando por fuera del alcance económico del proyecto). Gigabit Ethernet es la tecnología con menor ancho de banda y latencia más alta, pero su costo es significativamente menor, y es factibile como solución para interconectar clusters de bajo costo.

Por los motivos expuestos, se decidió interconectar los nodos del cluster Medusa mediante Gigabit Ethernet. Esta opción constituye la mejor alternativa considerando el costo de instalación y las prestaciones y deja la puerta abierta para posibles mejoras (interconexión por fibra óptica, mecanismos para acelerar el manejo de TCP/IP y posible migración a 10 Gigabit Ethernet cuando sea posible financiar el costo de los switches de alta velocidad).

3.1.3. Software

En la comunidad científica no hay discusión sobre el sistema operativo más adecuado para clusters de alto desempeño: el 85 % de los computadores de TOP 500 utilizan Linux, y el 76 % emplean una distribución de uso libre (TOP500, 2006). La elección del sistema operativo se limitó a comparar las distribuciones de uso libre disponibles al momento de instalar el cluster Medusa: Debian, Fedora y Caos. Los factores evaluados incluyeron la experiencia en clusters de alto desempeño, el soporte brindado por su comunidad (que redunda en una distribución más estable), y la calidad de la documentación existente.

Debian es una distribución con más de 15000 paquetes, mantenida por una amplia comunidad y bien documentada. Su versión estándar no soporta procesadores SMP, siendo necesario recompilar el kernel. La versión 3.1 soporta arquitecturas de 64 bits, incluyendo AMD Opteron, aunque para lograr el máximo desempeño requiere una extensión del kernel (AMD64) que se encontraba en fase de evaluación desde 2005, planeándose su incorporación para fines del 2006.

Fedora es una distribución de propósito general que cuenta con el respaldo de Red Hat, y es mantenida por una gran comunidad que ha generado extensa documentación. Soporta las arquitecturas de 64 bits (entre ellas AMD Opteron) y tiene una extensión que permite simplificar la infraestructura de datos. Al momento de instalar el cluster Medusa, la última versión era Fedora Core 5, que incluía una interfaz de instalación guiada para simplificar la tarea.

La distribución Caos se basa en Debian, Red Hat y FreeBSD. Se orienta a la eficiencia computacional, y provee una extensión robusta del kernel para arquitecturas x86 de 64 bits (no tiene soporte para Power y UltraSparc). Caos no es una distribución de propósito general, la mantiene un equipo pequeño de especialistas y está poco documentada. Caos tiene un ciclo de vida de 3 a 5 años, y el último release disponible se puso en producción en mayo de 2005.

Se analizó la disponibilidad de herramientas semiautomáticas de instalación y administración de clusters de alto desempeño, que engloban aplicaciones para simplificar la instalación, monitoreo y administración de los recursos de cómputo y las comunicaciones. Los productos evaluados fueron Open Source Cluster Application Resources (OSCAR) y Rocks.

OSCAR es una compilación autoinstalable que provee el software para instalar y administrar un cluster: bibliotecas y compiladores para programación paralela y distribuida (e.g. PVM, MPI), herramientas de administración (e.g. C3), y herramientas de monitoreo (Ganglia). El componente de instalación permite crear una imagen de un nodo, y luego instalar y configurar automáticamente los restantes nodos del cluster. Varios componentes adicionales ofrecen un entorno integrado para la administración del cluster, utilizando una base de datos para almacenar el software y su configuración. OSCAR es de acceso libre, es soportado por la distribución Fedora y existen antecedentes bien documentados de la instalación de clusters de alto desempeño utilizando OSCAR en tiempos razonables y con gran simplicidad operativa.

Rocks es un paquete basado en la distribución CentOS, que integra una serie de herramientas de instalación (utilizando una base de datos para almacenar el estado de los nodos del cluster y el software instalado) y varios componentes adicionales: ekV para administración unificada del cluster, SNMP para monitorear la carga y bibliotecas para procesamiento paralelo (PVM, MPI). Rocks se basa en paquetes, posibilitando una instalación sencilla y rápida, y permite utilizar *rollovers* para instalar un cluster específico para un determinado tipo de aplicaciones.

Considerando sus prestaciones, la simplicidad de instalación y la calidad de la documentación generada por la comunidad de desarrolladores, se decidió utilizar Fedora Core 5 como distribución de Linux a instalar en el cluster Medusa. Para automatizar el proceso de instalación y simplificar la administración y monitoreo, se decidió utilizar el paquete OSCAR.

3.2. Instalación de Medusa

La arquitectura propuesta para el cluster Medusa trata de simplificar la administración y contemplar la escalabilidad incremental del cluster, tratando de obtener el mayor poder de cómputo posible para ejecutar las aplicaciones. El cluster se organiza en un nodo maestro y cinco nodos esclavos. El nodo maestro es el único punto de acceso a los recursos computacionales, cumple la función de lanzar las tareas a ejecutar en el cluster, opera como servidor de archivos del dominio, y contiene un repositorio de instalación y un servidor PXE, para actualizar e instalar software en los nodos esclavos y nuevos nodos del cluster. Las herramientas administrativas se encuentran instaladas en el nodo maestro: la interfaz de acceso y administración, el software monitor de recursos Ganglia y otros utilitarios. La organización del cluster permite que solo el nodo maestro dedique una parte de su poder de cómputo a la administración, mientras que los nodos esclavos se dedican exclusivamente a la resolución de las tareas de cómputo intensivo.

Los nodos originales del cluster son equipos Sun Fire X2100 x64 Server, con procesador AMD Opteron 175 doble núcleo, con un reloj de 2.2 Ghz y 2MB de memoria caché L2. Cada nodo dispone de 2 GB de memoria, un disco rígido de 80 GB., y dos puertos Ethernet de 10/100/1000 Mb/s. El consumo promedio es de 230 W. y el consumo máximo es de 300W. En estado operativo, cada nodo tiene una emisión de sonido de 76 dB., y funciona en el rango de temperatura ambiente de 5°C a 35°C. Las dimensiones de cada nodo corresponden a una altura de 4.3 cm., un ancho de 42.55 cm., un profundidad de 50 cm., y el peso es de 13 Kg. En el año 2008 se incorporó un nuevo recurso de cómputo: un equipo Intel Core 2 Quad Q6600, con 4 nodos de procesamiento a 2.40 GHz., 8 MB de caché L2, 4 GB de memoria y disco rígido de 80 GB, que fue adquirido por un costo de U$S 1000. En la actualidad, el cluster Medusa dispone de un total de 16 nodos de cómputo.

En una primera instancia, la interconexión entre nodos se instrumentó mediante Ethernet de 100 Mb/s. En el correr del año 2007 se adquirieron dos switches Gigabit Ethernet, para proveer interconexión a 1 Gb/s. El cluster está conectado a un conjunto de UPS que garantizan el filtrado de las variaciones de corriente eléctrica, y en caso de una caída de la energía poder mantener el cluster funcionando en forma autónoma durante un período de tiempo razonable.

Cada nodo tiene instalado Fedora Core 5 para procesadores SMP de 64 bits, las bibliotecas PVM 3.4.5, MPICH 1.2.7, MPICH2 1.0.4, ACML (la implementación de BLAS especifica para AMD Opteron) y los compiladores Intel FORTRAN 9.0, Intel C 9.0 y GFORTRAN. El nodo maestro tiene las herramientas administrativas del cluster: Torque para el manejo de recursos, Maui para algoritmos de despacho, Gold para estadísticas de uso, Ganglia para monitoreo de recursos, lm_sensors para el control de temperatura, y la interfaz web que actúa de front-end para administración remota y para el acceso de usuarios.

3.3. Evaluación de desempeño

Una vez instalado el cluster Medusa, se realizaron experimentos para evaluar su desempeño computacional, con el objetivo de validar los argumentos considerados con en la fase de análisis y las decisiones adoptadas en la fase de diseño. Los experimentos se orientaron a identificar debilidades y fortalezas de la solución de hardware implementada, y bajo la necesidad de contar con valores para cuantificar el rendimiento computacional del cluster para la resolución de problemas de gran porte.

3.3.1. Metodología y herramientas utilizadas

La evaluación del cluster Medusa contempló aspectos de los modelos teóricos de desempeño más difundidos (Bailey and Snavely, 2005): la capacidad de cómputo del procesador; la velocidad de acceso a memoria principal, secundaria y caché; y la calidad de la red de comunicaciones. También se evaluó la escalabilidad de la arquitectura, para determinar la capacidad de mejorar el desempeño al utilizar más procesadores. Se emplearon benchmarks bien conocidos que permiten evaluar diversas métricas de desempeño y comparar con otras configuraciones: la suite HPC Challenge (HPCC) (Luszczek et al., 2005), y los benchmarks I/O Bench y PEAK de la suite PMAC (Snavely et al., 2002).

HPCC reúne siete benchmarks que evalúan diversas métricas de eficiencia, y dispone de los resultados de la evaluación de 150 clusters, permitiendo comparar con otros sistemas. Los paquetes incluidos en la suite HPCC son *HPL*, una implementación distribuida de LINPACK que resuelve sistemas lineales densos en doble precisión; *DGEMM*, que multiplica matrices en doble precisión utilizando un algoritmo de partición en bloques para reutilizar datos y minimizar el acceso a memoria principal, en versiones secuencial y multiprocesador; *STREAM*, que evalúa la eficiencia de acceso a memoria principal mediante operaciones de sobre vectores y matrices de gran tamaño; *PMT*, que analiza el intercambio de mensajes en una multiplicación distribuida de matrices densas, y permite evaluar el impacto de comunicar grandes volúmenes de datos al resolver problemas realistas; *RandomAccess*, que evalúa la velocidad de actualización de direcciones de memoria aleatorias; *FFT*, un benchmark que se basa en la ejecución de transformadas de Fourier discretas en doble precisión, en sus versiones secuenciales y de ejecución distribuida y *Communication Bandwidth and Latency*, un conjunto de tests para evaluar el ancho de banda y la latencia para varios patrones estándar de comunicación sobre mensajes de tamaño variable.

I/O Bench es un benchmark sintético que evalúa la eficiencia de las operaciones de lectura, escritura y actualización de memoria secundaria, contemplando diversos patrones de acceso (secuencial, hacia atrás y aleatorio), presentando resultados de ancho de banda máximo, mínimo y promedio. *PEAK* es un test que evalúa la eficiencia del procesador mediante un ciclo de operaciones que incluye divisiones, productos y evaluación de un polinomio de quinto grado.

Los benchmarks se ejecutaron sobre el cluster Medusa en un escenario dedicado, para evitar la interferencia de otras aplicaciones. Se realizaron varias ejecuciones de cada benchmark para evitar factores imprevistos e intentar reducir la influencia del no determinismo al trabajar con procesos distribuidos asincrónicos. Los valores numéricos obtenidos para cada benchmark y un análisis detallado de los resultados, se presentan en Nesmachnow and Salsano (2007).

La evaluación de desempeño permitió identificar importantes factores que pueden afectar el rendimiento del cluster Medusa. Asimismo, se lograron validar las decisiones que condicionaron el diseño del cluster. La eficiencia del procesador Opteron justificó su elección como la mejor alternativa para el cluster, considerando su relación precio/desempeño. Opteron alcanza un desempeño en operaciones de punto flotante similar a Power e Itanium, mientras que posee buena eficiencia en el acceso a memoria secundaria, y muy buenos valores de ancho de banda para diversos patrones de lectura y escritura. El acceso a memoria principal a través del bus *hypertransport* de Opteron supera a otras soluciones consideradas. Además, se verificó la capacidad de compartir el bus por los dos núcleos de cada procesador para acceder a la memoria física del sistema. La red de comunicaciones implementada en primera instancia (Ethernet de 100 Mb/s) afectaba negativamente la eficiencia de aplicaciones distribuidas, comprometiendo el desempeño y la escalabilidad del cluster. Se identificó como línea prioritaria mejorar la infraestructura de la red de comunicaciones, por encima de adquirir nuevos recursos de cómputo. Este aspecto fue cubierto en el año 2008 con la adquisición de switches Gigabit Ethernet. Con su nueva configuración de red, la infraestructura de cómputo es más eficiente para la aplicación del modelo de paralelismo de memoria distribuida, permitiendo abordar problemas de mayores dimensiones, contemplar la mejora en la precisión de resultados mediante, y sacar mejor provecho de la escalabilidad incremental del cluster.

3.4. Interfaz de acceso y administración

Para simplificar el uso del cluster Medusa por parte de usuarios y administradores, se diseñó una interfaz de acceso y administración integrando tecnologías que proveen diferentes funcionalidades. La interfaz, de nombre *Fenton*, fue implementada usando el lenguaje de scripting PHP, y permite el acceso a los recursos del cluster mediante una URL pública, la administración remota del cluster y la ejecución de trabajos por parte de los usuarios. Un sistema subyacente se encarga de la planificación de trabajos y del manejo de los nodos de cómputo. La base del sistema es Torque, una aplicación de código abierto que gerencia los recursos de cómputo, provee soporte para la programación paralela, y se ocupa de iniciar los trabajos y de la interacción con los usuarios. Torque es potenciado por Maui, una biblioteca que se enfoca en la planificación de trabajos, y con Gold, un sistema de contaduría que registra y maneja el uso de recursos en clusters de alto desempeño. Para analizar el estado del cluster se integró la interfaz Fenton con el sistema de monitoreo distribuido Ganglia, que permite acceder a las estadísticas históricas y actuales de diversas métricas de carga de los nodos de cómputo. La administración de clientes, usuarios y trabajos es soportada por una base de datos en PostGreSQL.

La figura 1 presenta ejemplos de la pantalla de la interfaz de acceso y administración desarrollada para el cluster Medusa.

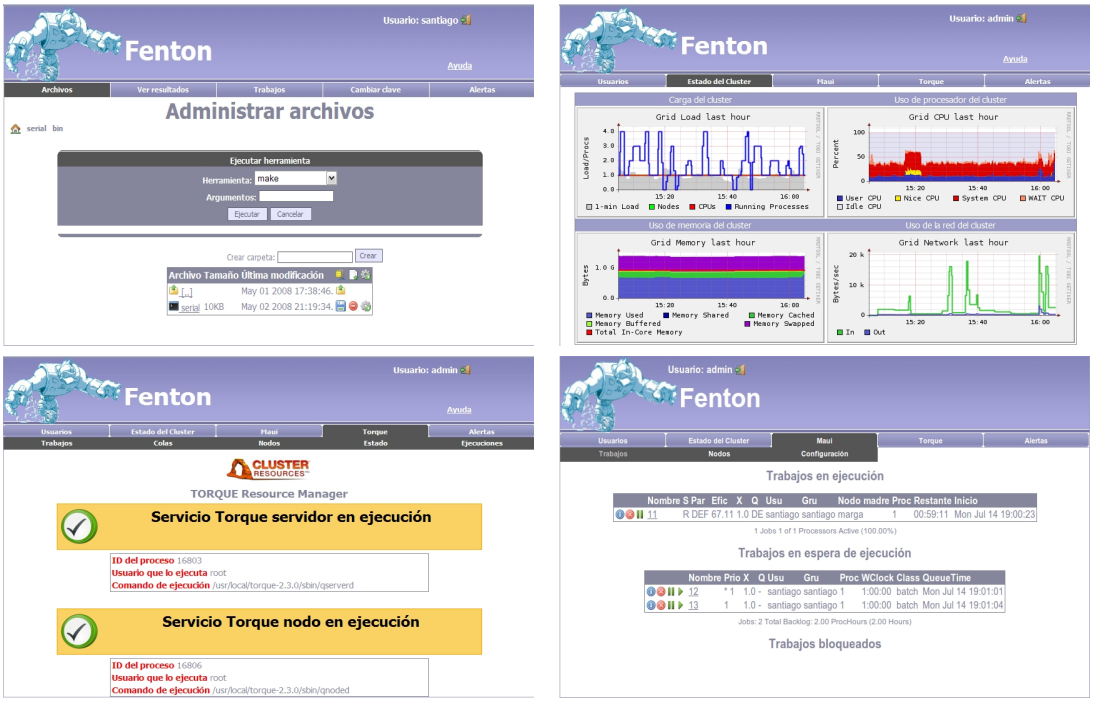

Figura 1: Interfaz de acceso y administración del cluster Medusa.

Detalles complementarios sobre el cluster, sus áreas de aplicación e investigadores involucrados, pueden consultarse en el sitio web de Medusa (2008).

4. APLICACIONES

4.1. El modelo caffa3d.MB

El modelo `caffa3d.MB` es una implementación del método de volúmenes finitos aplicado a la simulación numérica de flujos viscosos o turbulentos, tridimensionales y con transporte de escalares. `caffa3d.MB` utiliza un interpolador lineal mejorado que permite construir las aproximaciones discretas a las ecuaciones de movimiento del fluido, y representa la geometría del dominio de cálculo mediante mallas curvilíneas estructuradas por bloques. Las grillas estructuradas por bloques permiten describir dominios con geometrías complejas, para los cuales es difícil construir una malla completamente estructurada (de bloque único) con buenas propiedades. La interfaz entre bloques se maneja de forma implícita, al igual que el resto del dominio, evitándose de esta manera una degradación de la convergencia y de la eficiencia computacional del método. Las celdas de una interfaz entre dos bloques pueden asociarse “una a una” o en la estrategia de “muchas a una”, definiendo bloques que permiten refinar localmente la malla. Complementariamente, el modelo permite incorporar en el dominio de cálculo piezas rígidas en movimiento y estudiar su interacción con el fluido, utilizando interfaces deslizantes entre bloques.

El modelo matemático empleado en `caffa3d.MB` considera las ecuaciones de balance de masa (1) y balance de momento (2) para un flujo newtoniano incompresible bajo campo gravitatorio, incorporando la aproximación de Boussinesq para los efectos de flotación causados por pequeñas variaciones de densidad inducidas por la temperatura. Se considera también la ecuación de conservación para un escalar pasivo genérico ϕ (que incluye en particular el caso de la temperatura) (3). Las ecuaciones se presentan en su formulación integral (la forma canónica en el método de volúmenes finitos), favoreciendo una formulación conservativa. Las ecuaciones discretizadas del modelo numérico se otienen mediante la aplicación iterada de las ecuaciones (1), (2) y (3) a cada elemento de volumen que compone la malla. Para imponer las condiciones de borde se aplica un tratamiento diferenciado de los términos de flujo en la discretización de las ecuaciones del modelo numérico para los elementos en la frontera del dominio de cálculo, incorporando una nueva estructura de vecindad con puntos adicionales para la malla en los centros de caras de frontera, que simplifica la especificación de las condiciones de borde (Usera et al., 2008).

Complementando el modelo numérico se desarrolló un método algebraico multigrilla para trabajar con el enfoque de bloques estructurados y resolver de manera eficiente las ecuaciones de presión. El modelo numérico tridimensional `caffa3d.MB` se desarrolló tomando como base el modelo bidimensional `caffa` de Ferziger and Peric (1999), y se encuentra disponible públicamente en su sitio web www.fing.edu.uy/imfia/caffa3d.MB.

$$\int_S (\vec{v} \cdot \hat{n}_S)\ dS = 0 \tag{1}$$

$$\begin{aligned} \int_\Omega \rho \frac{\partial u}{\partial t}\, \partial\Omega + \int_S \rho u (\vec{v} \cdot \hat{n}_S)\, dS = \int_\Omega \rho\beta\,(T - T_{ref})\,\vec{g} \cdot \hat{e}_1)\, \partial\Omega + \\ \int_S (-p\hat{n}_S) \cdot \hat{e}_1)\, dS + \int_S (2\mu D \cdot \hat{n}_S)\, \hat{e}_1)\, dS \end{aligned} \tag{2}$$

$$\int_\Omega \rho \frac{\partial \phi}{\partial t}\, \partial\Omega + \int_S \rho\phi (\vec{v} \cdot \hat{n}_S)\, dS = \int_S \Gamma\,(\nabla\phi\, \hat{n}_S)\ dS \tag{3}$$

Para mejorar la eficiencia computacional del modelo `caffa3d.MB` y permitir la simulación de escenarios realistas, se incorporó el uso de la computación paralela en multiprocesadores de memoria compartida, aprovechando la descomposición del dominio en bloques y mediante la introducción de directivas de compilación OpenMP. Aunque el esquema de OpenMP carece de la flexibilidad del paradigma de pasaje de mensajes, provee un método conceptualmente simple y de eficiencia comprobada en sistemas SMP. La técnica de descomposición de dominio puede adaptarse de modo sencillo al esquema de OpenMP en el caso de grillas estructuradas que propone el modelo `caffa3d.MB`. La estructura de datos diseñada para soportar grillas estructuradas se utiliza como base para distribuir las tareas de cómputo a diferentes procesadores en un computador SMP. El esquema de paralelismo contempla ejecutar en un único procesador las tareas asociadas a las interfaces entre bloques, para evitar inconvenientes en el acceso simultáneo a datos en la memoria compartida.

Las principales ventajas del enfoque de paralelismo implementado son la simplicidad de programación y la capacidad de controlar la descomposición de dominio diseñando grillas que contemplen estrategias de división balanceadas para obtener alta eficiencia computacional. En este modelo de paralelismo no existen tiempos dedicados a transmitir información entre procesos, ya que todos ellos tienen acceso a los registros de la memoria compartida. Dado que la inclusión de directivas OpenMP puede realizare sin modificar la lógica del programa, queda asegurado que el algoritmo paralelo será capaz de alcanzar los mismos resultados que obtiene la versión secuencial, salvo por errores de redondeo del sistema de representación de punto flotante asociados a los diferentes ordenamientos cuando se suman elementos del dominio completo (i.e. al calcular los residuos). Como contrapartida, las desventajas del modelo se relacionan con la incapacidad de aprovechar la escalabilidad incremental de los recursos de cómputo, y con las limitaciones de adaptabilidad a estructuras donde no sea posible identificar a priori un esquema de particionamiento de bloques y de balance de carga que permita sacar provecho a la técnica de procesamiento paralelo utilizada.

En el marco del proyecto *Laboratorio de Simulación Numérica de Flujos a Superficie Libre* se ha experimentado con el modelo `caffa3d.MB` paralelo utilizando OpenMP sobre los nodos bi procesadores del cluster Medusa, alcanzándose niveles de *speedup* entre 1.6 y 1.8, correspondientes a una eficiencia computacional de 0.8 y 0.9 respectivamente. Estos resultados son muy promisorios, y muestran la capacidad de utilización de los recursos computacionales para la resolución de instancias complejas de los problemas que propone abordar el proyecto.

4.2. Módulo de superficie libre

El módulo para simulación de flujos a superficie libre se implementó en base al método `CICSAM` propuesto por Ubbink and Issa (1999). `CICSAM` se basa en el método de volúmenes finitos y sigue un enfoque completamente conservativo (al igual que el modelo de base `caffa3d.MB` con el cual sería acoplado), aspecto que resultó especialmente atractivo para su elección. El desplazamiento de la superficie libre es seguido mediante el cálculo de un campo escalar que es discontínuo en la interfaz y uniforme lejos de ella. Por este motivo, el método no requiere de complejas estrategias de reconstrucción de la interfaz, que queda definida por las superficies de nivel del campo escalar para el valor de la discontinuidad. `CICSAM` implementa estrategias especiales de discretización para asegurar la definición abrupta de la interfaz y conservar acotado el campo escalar, al tiempo que se conserva el carácter implícito del método, para permitir representar la eventual ruptura y reagrupamiento de la interfaz. La estrategia de discretización combina métodos de interpolación corriente arriba y corriente abajo, con limitadores que aseguran valores acotados para los flujos entre celdas. El acople del módulo de superficie libre con el modelo `caffa3d.MB` no presentó mayores dificultades, considerando la simplicidad de manejo de las estructuras de datos definidas en el modelo de base, especialmente para describir la geometría del dominio de cálculo.

4.3. Casos de validación

Para la validación del modelo se ha experimentado con dos casos de estudio relacionados con la rotura de una presa en un escenario sin obstáculos, y en un escenario donde el agua choca con un obstáculo cilíndrico. En el primer caso, una región de agua inicialmente en reposo y en contacto con el aire, comienza a derramarse al eliminarse la barrera que la contiene. El escenario tiene una relativa simplicidad geométrica, pero entraña dificultades en la simulación debido a la desaceleración abrupta de la masa de agua al golpear con la pared opuesta. La figura 2 muestra la evolución del sistema simulado en diversos instantes de tiempo.

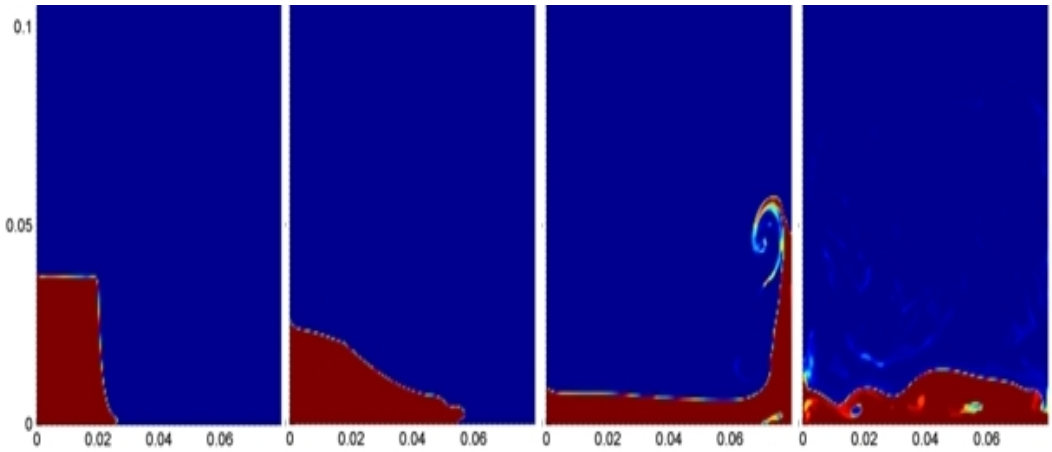

Figura 2: Evolución del agua (zona roja) al derramarse, desplazar el aire (zona azul) y golpear con la pared opuesta.

La figura 3 presenta la evolución de la estructura de la interfaz para el caso de ejemplo presentado en la Figura 2. Se observa que la interfaz conserva su compacidad a lo largo del movimiento, generándose relativamente pocas eyecciones de fluido. Esta es una propiedad importante que deben reflejar los métodos de seguimiento de interfaces.

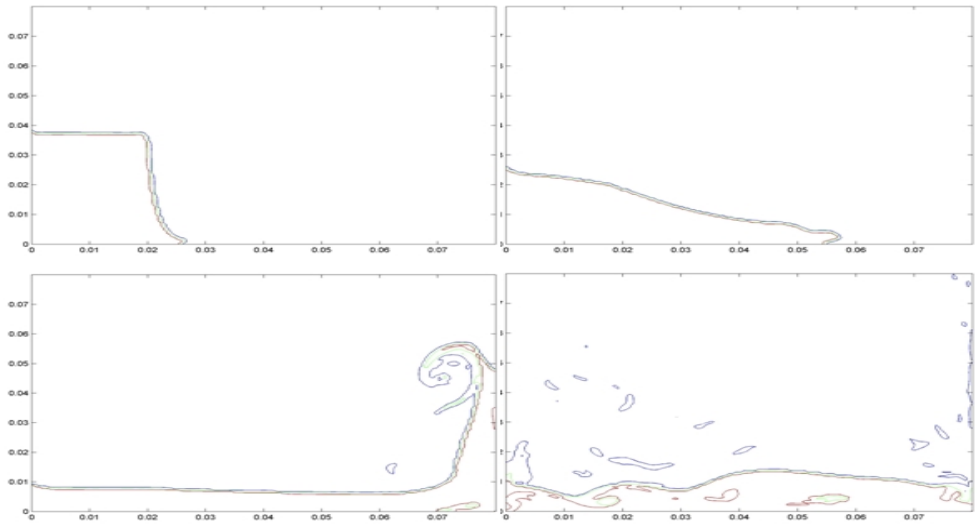

Figura 3: Evolución de la estructura de la interfaz para la secuencia de la figura 2.

El segundo caso de validación corresponde a la rotura de una presa que enfrenta un obstáculo cilíndrico. A diferencia del caso anterior, que es bidimensional, en este caso el flujo que se desarrolla en torno a la pila es tridimensional, requiriendo un mallado más elaborado. La figura 4 presenta un esquema de la malla computacional diseñada para modelar este caso, una malla tipo C que rodea el obstáculo cilíndrico produciendo una distribución de celdas ajustada a la geometría), y un ejemplo de evolución del flujo en el segundo caso de validación.

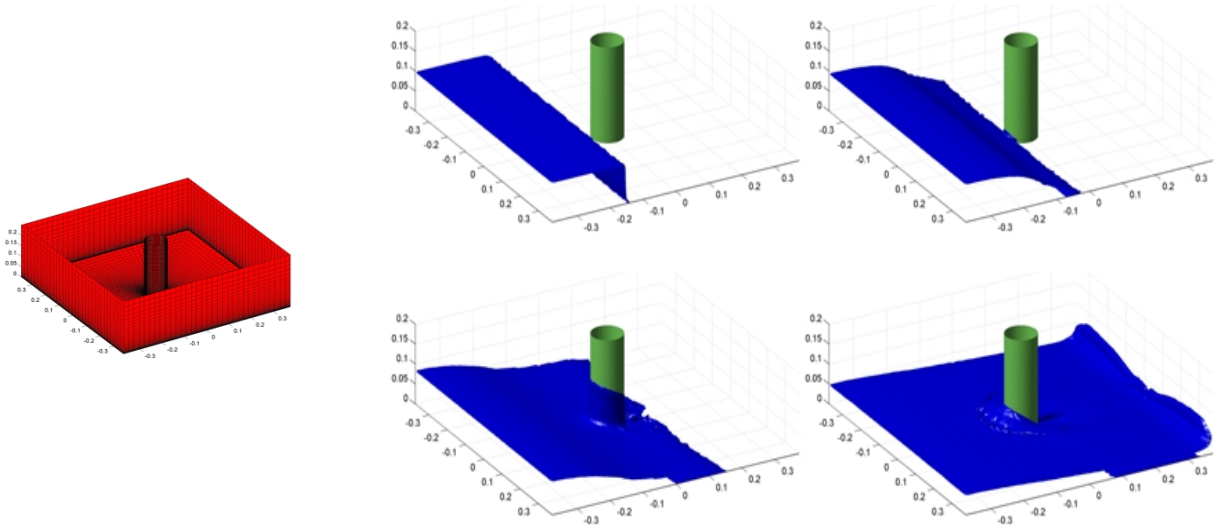

Figura 4: Malla computacional y evolución del flujo en el segundo caso de validación.

El caso de prueba presentado permitió verificar que el modelo implementado resulta adecuado para el tipo de aplicaciones previstas en los objetivos del proyecto.

5. PERSPECTIVAS DE TRABAJO ACTUAL Y FUTURO

El proyecto *Laboratorio de Simulación Numérica de Flujos a Superficie Libre* se encuentra en sus últimas etapas de ejecución. Los avances más recientes corresponden a la aplicación del módulo de superficie libre a simulaciones numéricas para el caso del flujo en torno a una pila. Complementariamente, se han realizado ensayos y recopilación bibliográfica experimental sobre erosión local en pilas y se ha avanzado con los ensayos físicos de modelos de barcazas en el canal de olas, bajo un régimen de corriente uniforme y distintos tirantes. El grupo del proyecto se encuentra trabajando en estos temas en la actualidad.

Las principales líneas de trabajo futuro se relacionan con la mejora de eficiencia computacional del modelo número. En este sentido, se propone implementar una versión paralela del modelo `caffa3d.MB` que aplique dos niveles de paralelismo, agregando al modelo de paralelismo SMP sobre cada nodo de cómputo una implementación que incorpore el paradigma de memoria distribuida para aprovechar la escalabilidad incremental de la plataforma computacional. Complementariamente, se planifica extender el análisis del efecto de sobre calado en buques de carga navegando en aguas poco profundas. Por último, se han manejado diversas opciones para el mantenimiento y mejora de la capacidad computacional del cluster Medusa. En este sentido, se ha avanzado en contactos para la utilización del cluster por parte de otros grupos de investigación y de consultoras nacionales, y para la aplicación de los modelos desarrollados en el marco del proyecto en el ámbito industrial, para estudiar un modelo de protección contra heladas. Se prevé incorporar nuevos recursos de cómputo al cluster con la financiación a obtener en estas iniciativas.

6. CONCLUSIONES

El proyecto *Laboratorio de Simulación Numérica de Flujos a Superficie Libre* ha cumplido con sus objetivos principales. La Facultad de Ingeniería dispone de un cluster con 16 nodos de cómputo, implementado con un costo menor a U$S 10000, para la ejecución de modelos y simulaciones numéricas. El éxito del proyecto motivó a la Facultad de Ingeniería a solicitar financiación para la implantación de un cluster con 256 procesadores, por la cual se está compitiendo con otras propuestas en la actualidad y respecto al cual se espera tener novedades sobre la decisión a fines del corriente año 2008. Con la infraestructura instalada es posible abordar diversos problemas de diseño de infraestructura hidráulica (muelles, pilas y estribos de puentes) y analizar su impacto ambiental en ríos y estuarios, y el diseño y maniobrabilidad de barcazas en áreas restringidas de navegación. Se diseñó un módulo de superficie libre para el modelo numérico tridimensional de simulación de flujos `caffa3d.MB`, que se ha utilizado para estudiar los problemas presentados utilizando la capacidad computacional del cluster multiprocesador.

Este artículo ha presentado una visión global del proyecto, enfocándose principalmente en las actividades de diseño, instalación y configuración del cluster multiprocesador y describiendo las aplicaciones basadas en el modelo numérico empleado.

REFERENCIAS

Bailey D. and Snavely A. Performance modeling: Understanding the past and predicting the future. In *Proceedings of Euro-Par*, pages 185–195. 2005.

Bode B., Hill J., and Benjegerdes T. Cluster interconnect overview. In *Proceedings of the annual conference on USENIX Annual Technical Conference*, pages 41–41. 2004.

Chen H., Wyckoff P., and Moor K. Performance evaluation of a gigabit ethernet switch and myrinet using real application cores. In *Proceedings of Hot Interconnects*. 2004.

Ferziger J. and Peric M. *Computational Methods for Fluid Dynamics*. Springer, Berlin, 1999.

Luszczek P., Dongarra J., Koester D., Rabenseifner R., Lucas B., Kepner J., McCalpin J., Bailey D., and Takahashi D. Introduction to the HPC challenge benchmark suite. Technical Report Paper 57493, ICLUT-05-01, Lawrence Berkeley National Laboratory, 2005.

Medusa. Homepage del cluster Medusa. http://www.fing.edu.uy/imfia/medusa/web/, 2008.

Nesmachnow S. and Salsano E. Evaluación del desempeño computacional del cluster Medusa. InCo, Facultad de Ingeniería, Universidad de la República, Uruguay, 2007. Disponible en http://www.fing.edu.uy/inco/pedeciba/bibliote/reptec/TR0712.pdf.

Salsano E. Proyecto de grado: Clusters alto desempeño. InCo, Facultad de Ingeniería, Universidad de la República, Uruguay, 2007. Disponible en http://www.technetworld.info/.

Snavely A., Carrington L., Wolter N., Labarta J., Badia R., and Purkayastha A. A framework for performance modeling and prediction. In *Proceedings of the 2002 ACM-IEEE conference on Supercomputing*, pages 1–17. IEEE Computer Society Press, 2002.

TOP500. Top 500 supercomputing sites. http://www.top500.org, 2006. Consultada en junio de 2006.

Ubbink O. and Issa R. A method for capturing sharp fluid interfaces on arbitrary meshes. *Journal of Computational Physics*, 153(1):26–50, 1999. ISSN 0021-9991.

Usera G., Vernet A., and Ferré J. A parallel block-structured finite volume method for flows in complex geometry with sliding interfaces. *Flow, Turbulence and Combustion*, 80(3):346–350, 2008. ISSN 1573-1987.