Perfil Preliminar de las Comunicaciones Intercluster

Fernando G. Tinetti* Walter Aróztegui

III-LIDI, Facultad de Informática, UNLP
50 y 115, 1900, La Plata
Argentina

CeTAD, Facultad de Ingeniería, UNLP
48 y 116, 1900, La Plata
Argentina

Reporte Técnico PLA-001-2006[1]
Marzo 2006

Resumen

Inicialmente, este reporte muestra una forma sencilla en que se puede obtener un perfil de rendimiento de las comunicaciones entre dos clusters interconectados. Además, se muestran los resultados obtenidos en la experimentación con dos clusters correspondientes a subredes diferentes de una red Internet B. Si bien mucha de la descripción se orienta a detalles muy relacionados con redes de computadoras, no se pierde de vista que el objetivo final es la utilización de varios clusters para llevar a cabo cómputo paralelo intercluster. Se documenta, principalmente, el experimento básico, la justificación del mismo, y algunos detalles relacionados con la utilización del ancho de banda de la red utilizado, que es un recurso compartido para la utilización de los servicios de Internet por cientos de otras computadoras.

Finalmente, se comentan a modo de resumen las características más relevantes desde la perspectiva de cómputo paralelo de este tipo de experimentos. Por un lado, es importante contar con una herramienta al menos metodológica para una evaluación preliminar del rendimiento de las comunicaciones intercluster, y por el otro se debe cuantificar (en la medida de lo posible) la validez de estos resultados preliminares. Esta cuantificación debería ser útil para evaluar la continuidad del trabajo y/o del rendimiento de cómputo paralelo intercluster.

1. Introducción

Tal como se ha mencionado en reportes previos [6] [7], para efectivizar cómputo paralelo intercluster se debe contar con

- Una conexión viable, lo cual incluye una mínima sustentabilidad técnica de las comunicaciones TCP/IP entre los clusters casi como en una red local. Esto implica analizar todo lo referente a la seguridad, al menos en términos de firewalls de protección entre las redes locales.
- Una caracterización mínima y *confiable* (al menos con cotas conocidas/determinadas) del rendimiento de las comunicaciones entre los clusters.

Y con esto es posible comenzar a desarrollar/estudiar algoritmos y aplicaciones específicas, para aprovechar la capacidad de cómputo disponible entre los clusters. Aunque *a priori* pueda suponerse que serán necesarios nuevos algoritmos paralelos (y, de hecho, se pueden proponer y analizar), al llegar a este estudio en profundidad se necesita una caracterización cuantificada del rendimiento disponible para las transferencias de datos entre los clusters a utilizar.

En este reporte técnico, la idea es proponer y llevar a cabo un conjunto mínimo de experimentos y con el mínimo de requerimientos tanto a nivel de hardware, software y ancho de banda a utilizar. A diferencia de la mayoría de los experimentos y estudios en el área de cómputo paralelo en clusters, no se puede asegurar la disponibilidad absoluta de la red de interconexión, dado que entre los clusters se utiliza la interconexión *estándar* y ya instalada para uso de Internet. Por lo tanto, no tendría sentido establecer como requerimiento que esta interconexión se utilice de manera excluyente, porque podría significar la exclusión (justamente) de posiblemente cientos de usuarios/computadoras que llevan a cabo sus tareas usuales incluyendo tráfico en Internet. De esta manera, aunque cada uno de los clusters sea de uso *exclusivo* para cómputo paralelo (algo bastante usual, de hecho), es prácticamente imposible

* Investigador Asistente CICPBA
[1] PLA: sigla de Parallel Linear Algebra

mantener el requisito de exclusividad en las conexiones entre los clusters. Más aún, es técnicamente muy complicado aún establecer una determinada calidad de servicio entre los clusters, dado que las conexiones muy posiblemente están administradas por personal ajeno a cómputo paralelo y también ajeno a los problemas de rendimiento de los *patrones* de las comunicaciones para cómputo paralelo.

La idea de minimizar los requerimientos se torna importante cuando todas las computadoras de cada cluster no están disponibles todo el tiempo para cómputo paralelo. Se podría dar el caso, por ejemplo, de destinar todas las computadoras solamente para software de producción, es decir para la resolución de un problema puntual en particular que es importante para la/s institución/es involucrada/s. El resto del tiempo, siguiendo con el ejemplo, las computadoras podrían ser utilizadas con otros propósitos en un ambiente clásico de red local de un laboratorio/oficina.

Por otro lado, dado que se tienen ámbitos de administración de las redes involucradas muy diferentes (con posiblemente muchos administradores de red involucrados implícitamente), también es importante tener una idea inicial de las características de conectividad entre los clusters. Se puede dar el caso de no tener disponible la conexión durante algunos períodos de tiempo y sería importante conocer cuáles, las razones, y si es posible evitar tales períodos de desconexión.

2. Entorno y Objetivos de la Experimentación

Tal como se adelantara en parte en la sección anterior, el entorno de utilización de cómputo paralelo intercluster puede ser muy variable. Se pueden dar casos donde los clusters no están totalmente disponibles para cómputo paralelo, la conexión no se tiene permanentemente y el rendimiento de las comunicaciones intercluster vería en el tiempo por no estar dedicada exclusivamente a cómputo paralelo, o a comunicar los clusters involucrados. Por estas razones, se recurre a un entorno de experimentación que, aunque *específico* (es *uno* de muchos posibles), presenta varias de las características mencionadas previamente. Los dos laboratorios involucrados son:

- III-LIDI
- CeTAD

Cada uno de estos laboratorios tiene acceso a una red local que es la que se utilizará para cómputo paralelo. En cada una de las redes locales, se utilizará para los experimentos una sola computadora, dado que solamente se tienen que identificar las características de la interconexión entre las redes. Ambas redes locales son subredes de una misma red Internet B. Esto significa que, en realidad, no se llega a utilizar la *salida* a Internet de la institución a la que pertenece la red B (la UNLP, en este caso), sino que se comparte *todo* o *parte* del tráfico Internet dentro de la red. En el caso de la red local que se utiliza en el III-LIDI no se tiene acceso exclusivo y, de hecho, es una sala de computadoras que se utiliza para dar clases a alumnos varias veces en la semana. Esto involucra no solamente carga en las computadoras y en la red de interconexión sino también la posibilidad de que al terminar la clase el docente directamente apague todas las computadoras, incluyendo la que se necesita para llevar a cabo los experimentos.

La red de interconexión existente entre los laboratorios es, en realidad, la red de interconexión inicialmente instalada para el acceso a Internet de las facultades involucradas. Las características más importantes de esta interconexión son:

- Cada una de las redes locales pertenecen a una subred diferente.
- Tiene más de cinco routers intermedios involucrados en la transferencia de datos.
- No se tiene acceso a la mayoría de los routers intermedios, ni siquiera se conoce qué interfase de red tiene cada uno de ellos.
- No se conocen las políticas de seguridad de cada uno de estos routers intermedios. En particular, no se sabe qué *ports* están filtrados por *firewalls*, o aún si tienen *firewalls* en funcionamiento (con reglas *activas*), por ejemplo.

En este contexto, se tiende a que la información que se obtiene de los experimentos sea para estimar los parámetros desconocidos de la interconexión de las redes locales utilizadas.

Aunque el principal motivo para la experimentación alrededor de las comunicaciones entre clusters es el rendimiento, se deben tener en cuenta las características dinámicas de la interconexión, tal como se ha mencionado antes. En el entorno de un único cluster, también es importante caracterizar el rendimiento y esto usualmente se lleva a cabo obteniendo los valores de latencia y ancho de banda que caracterizan a

las comunicaciones entre dos computadoras. En el entorno intercluster quizás se tengan valores mínimos y/o máximos, pero además es importante conocer con cierto detalle todas las demás características de la interconexión entre los clusters. Entre estas características se pueden mencionar:

- Fallas en la interconexión.
- Si existen períodos de mayor o menor disponibilidad de ancho de banda.
- La dependencia (si existe) de la latencia respecto del tráfico existente en la red de interconexión de los clusters.
- La existencia de filtros de seguridad entre los clusters, permanentes o planificados en intervalos.

Y en función de estos objetivos o, más específicamente, para cuantificar estas características, se definen los experimentos a realizar. La idea básica de los experimentos es la misma que la reportada ampliamente en [1] [2] [3] en el contexto de disponibilidad de CPU en redes locales, y propuesta, en cierta forma en [8] para comunicaciones en una red local no completamente disponible.

3. Descripción de los Experimentos

Tal como se comenta anteriormente, la idea es utilizar el mínimo de recursos tanto para cómputo como para las comunicaciones. Desde la perspectiva de cómputo, evidentemente se necesitan computadoras para llevar a cabo los experimentos. Desde la perspectiva de las comunicaciones, se utilizan uno o varios enlaces que no necesariamente se pueden disponer en forma exclusiva para cómputo paralelo (o para los experimentos, en este caso) dado que el motivo principal de su existencia es el de proveer conexión a Internet.

El experimento más sencillo pero muy significativo en cuanto a la importancia de la información sumistrada sigue siendo el ping-pong de mensajes. Para llevar a cabo este experimento se necesitan solamente dos procesos y, en el contexto de cómputo intercluster, dos computadoras. Es por esta razón que se utiliza solamente una computadora de cada red local utilizada. Esta estrategia/definición es general, es decir que aunque se utilicen más de dos clusters, la cantidad de máquinas a usar de cada cluster será una sola. De hecho, para los experimentos habrá a lo sumo solamente dos computadoras funcionando en total, dado que no tiene sentido llevar a cabo más de un experimento ping-pong simultáneamente.

Una vez definido que el experimento básico será el ping-pong de mensajes y elegidas las computadoras de cada cluster, hay varias alternativas para implementar el ping-pong [5]. Dado que inicialmente no se tiene ninguna clase de información, el mismo comando ping de Linux será suficiente para recoger los datos para obtener la información preliminar de las comunicaciones entre los clusters. Es importante resaltar que aunque se hagan experimentos extensivos e intensivos, con muchas recolecc\'on de resultados con su consiguiente estabilidad estadística, la información seguirá siendo preliminar ¿Por qué? sencillamente por la diferencia existente entre el transporte de los datos con el protocolo ICMP (Internet Control Management Protocol) [4] (usado en forma directa por el comando ping) y los protocolos y/o técnicas usados para la comunicación confiable entre procesos de una aplicación paralela. Esto implica que, por un lado, se simplifica el estudio y la experimentación preliminar de la interconexión, pero aún se desconocen todos los detalles más específicos de rendimiento. En realidad, con ICMP se estaría en la mejor situación de rendimiento y en una de las peores de confiabilidad, dado que los datos obtenidos surgirán de los experimentos satisfactorios, lo cual implica que no hay problemas de confiabilidad.

En este punto se tienen definidos los detalles más significativos de los experimentos, pero aún restan los parámetros de ejecución de los mismos. Estos parámetros están relacionados inicialmente con lo que se necesita para cuantificar/aproximar los dos índices básicos de las comunicaciones punto a punto: latencia y ancho de banda. Estos índices normalmente se utilizan en el modelo de tiempo de las comunicaciones dado por

$$t(n) = \alpha + \beta n \tag{1}$$

donde α es el tiempo de latencia (o *startup*) y β es la inversa del ancho de banda, es decir el tiempo por item de datos a transferir y n es la cantidad de items o datos a transferir. Siguiendo la idea de simplificación de los experimentos, la latencia, o *startup time* se puede estimar con mensajes de tamaño mínimo o cero si fuera posible. En realidad, cualquier valor menor de 10 bytes puede ser útil, ya que lo que se intenta estimar específicamente es:

- La sobrecarga de los protocolos o *pila de protocolos* impuesta por la implementación del sistema operativo utilizado.

- El tiempo de transporte físico mínimo, que implica también la interfase con el subsistema de I/O de cada computadora.

y se debe recordar que en el contexto de cómputo paralelo en clusters es muy poco probable tener mensajes de menos de 1 KB entre procesos. Se debe notar aquí también que se está asumiendo que la sobrecarga de las bibliotecas o rutinas de comunicación de procesos de un programa paralelo es nula (lo cual es muy poco probable). Por otro lado, para la estimación del ancho de banda se pueden utilizar mensajes relativamente grandes, para los cuales se puede asumir que la mayor parte del tiempo de comunicaciones se utiliza para la transferencia física de los datos de un proceso a otro. En todos los casos, se deben tener suficientes datos para que la información tenga validez desde el punto de vista estadístico. Este último punto es particularmente complicado de sostener y/o justificar en el contexto de cómputo intercluster donde, justamente, se intentan capturar cambios relativamente importantes en el rendimiento de las comunicaciones. Sin embargo, la idea es aquí que si hay cambios importantes se puedan cuantificar, caracterizar su probabilidad o relacionar con algún factor externo e independiente del cómputo paralelo que, por lo tanto, se desconoce.

Si bien es bastante sencillo definir tamaños de mensajes *pequeños* (el límite inferior es, claramente, cero) no es el caso para la definición de los mensajes *grandes*. Se debe tener en cuenta que las aplicaciones paralelas son muy variadas y de muy variados patrones de cómputo también. En este caso, se deben tener en cuenta dos puntos importantes que necesariamente restringen el tamaño de los datos a transferir en los experimentos:

1. El comando ping normalmente establece un máximo en la cantidad de datos que se envían/reciben. Este tamaño está estrechamente relacionado con la simplicidad del comando y el protocolo utilizado (ICMP).

2. Se utilizará una red de interconexión no exclusiva para cómputo paralelo y, por lo tanto, es deseable no *inundar* esta red con tráfico diferente del que la originó y mantiene su razón de existir: Internet. En este sentido, se tiene un problema relativamente importante por cuanto se debe usar la red para llevar a cabo cómputo paralelo pero se debe también dejar ancho de banda para las aplicaciones que normalmente corren usando estas interconexiones. Si bien es importante notar que el tráfico ICMP se descarta en caso de sobrecarga *extrema* de los routers, también es importante recordar que toda sobrecarga termina afectando de una manera u otra a las aplicaciones que usan la red.

Por lo tanto, los tamaños elegidos inicialmente son 8 bytes para los mensajes *pequeños* (con los cuales se tiene una idea de latencia o *startup time*) y 20000 bytes (aproximadamente 20 KB) para los mensajes *grandes* (con los cuales se tiene una idea del ancho de banda).

Aunque estén definidos los tamaños de los mensajes, es necesario definir también la frecuencia con las que se los utilizará, para tener una mejor idea del ancho de banda que efectivamente podrán llegar a utilizar los experimentos. El mismo comando ping, por su propio funcionamiento, define que se envía/recibe un mensaje por segundo a menos que se tenga privilegios de usuario root y se indique explícitamente otro intervalo de tiempo. Esto, en principio, ya impone un límite máximo de tráfico. Sin embargo, se necesitan datos de los mensajes de las dos longitudes, es decir de 8 bytes y de 20 KB. No tiene sentido usar dos comandos ping concurrentes porque puede llevar a errores de interferencia de las comunicaciones entre ellos dado que normalmente existe una sola placa de interfase de comunicaciones en cada una de las computadoras. Es así que la idea será usar un comando ping para mensajes de 8 bytes y luego, en secuencia, un comando ping para mensajes de 20 KB. De hecho, la secuencia utilizada es de la forma:

ping -c 3 -i 1 -s 8 IPdestino; ping -c 1 -s 20000 IPdestino

donde IPdestino es el número IP de la computadora destino (o su nombre, aunque usando el IP se evita tráfico con el servidor de nombres) y los demás parámetros tienen el siguiente significado:

- -c 3/1: se envían/reciben 3/1 paquetes. La idea inicial es imponer una cantidad máxima de mensajes para que el comando ping termine y no quede ejecutándose indefinidamente (que es el funcionamiento normal, a menos que se indique esta cantidad máxima de mensajes). En el primer ping no es tan importante la cantidad en sí misma, dado que son mensajes cortos (8 bytes de datos) sino que el comando ping termine y provea las estadísticas que corresponden. En el segundo comando ping sí es importante limitar la cantidad de paquetes, dado que pueden imponer una carga importante en la red de comunicaciones.

- -i 1: se envían/reciben paquetes cada 1 segundo. Aunque es el funcionamiento estándar, es interesante establecerlo explícitamente por medio de este parámetro para evitar confusiones posteriores.

- -s 8/20000: los paquetes transportan 8/20000 bytes de datos de usuario. Con este parámetro se cambia el funcionamiento estándar que establece un único tamaño de paquetes. Es decir que el primer ping de la secuencia se utiliza para recoger datos de mensajes pequeños (latencia) y el segundo para los mensajes grandes (ancho de banda).

Se debe notar que en el segundo comando ping no se utiliza el parámetro -i, dado que hay un único paquete que se envía/recibe. En realidad, se termina utilizando un parámetro opcional más, que es el de timeout porque se ha comprobado que cuando hay problemas de conectividad puede darse que el comando ping no termina. Es algo no documentado (no debería pasar, en principio) pero en cualquier caso se puede evitar, por ejemplo, con

-w 4

que indica que el comando ping termina después de transcurridos 4 segundos independientemente de la cantidad de paquetes y de lo que haya sucedido con ellos. Por otro lado, esta opción del comando genera mayor regularidad en los resultados obtenidos dado que, cuando la conexión falla algunas versiones del comando ping tienen un tiempo de respuesta muy lento. Esto genera que cuando la conexión falla se tienen datos en intervalos de tiempo muy irregulares y, generalmente, mucho mayores que los generados cuando se tiene conexión disponible. Esto tiene ingerencia directa en la presentación de resultados, dado que se tiene que ajustar la escala de tiempo de las muestras.

En este punto se tienen no solamente las definiciones más importantes de los experimentos sino también los detalles más específicos de los parámetros a usar. De hecho, en este punto se puede ya definir que se usen los comandos ping definidos antes de manera iterativa y se recojan los resultados en un archivo para su análisis posterior. De hecho, el script utilizado para la recolección de los datos es

```
while true; do \
( \
  date >> pp1; \
  ping -c 3 -i 1 -s 8 -w 4 IPdestino > ver; \
  cat ver | grep [A/] >> pp1; \
  date >> pp20k; \
  ping -c 1 -s 20000 -w 2 IPdestino > ver; \
  cat ver | grep [A/] >> pp20k; \
  sleep 5 \
); done
```

es decir que se recogen en dos archivos diferentes (pp1 y pp20k) los datos correspondientes a los dos tamaños de mensajes definidos y, además, se establece un tiempo relativamente corto (5 segundos) durante el cual no se introduce ningún dato en la red. Con este script, se puede calcular muy fácilmente que el ancho de banda máximo que se utiliza de la red de comunicaciones es de 20000 + 3 bytes por cada 9 segundos. Asumiendo que la red de interconexión es de 10 Mb/s (lo cual es bastante probable, por el uso de placas Ethernet de 10 Mb/s en los routers), esto implica que poco más de 17780 bits por segundo representan solamente un poco más del 0.17 % del ancho de banda disponible. Por lo tanto la sobrecarga de los experimentos podría considerarse *despreciable*. Tanto el archivo destino de los resultado como el "disparo" del script anterior quedan por definirse, pero lo más sencillo es dejar un solo archivo de salida y para el arranque de la ejecución del script se usó una línea de comandos del estilo

at -f script now + 1 minute

que no hace más que poner en la cola de "at" el comando script a ajecutarse un minuto después de ser ejecutada la propia línea de comandos.

Es evidente que el volumen de los datos recolectados es proporcional al tiempo durante el cual se llevan a cabo los experimentos. Más allá del tamaño de los archivos involucrados, suele ser un problema el manejo de los datos en las planillas de cálculo, dado que normalmente están *limitadas* a varias decenas de miles de filas. Solamente para tener una idea de la magnitud, para cada tamaño de paquete ICMP se tienen 9600 *muestras* por día. Para tener una idea de toda una semana (de forma tal que se incluyan horarios de *oficina*, *horas pico*, *horas de clases*, *noches*, etc.) sería muy importante tener los datos de toda una semana, y esto implica más de 67000 datos a procesar. Aunque los archivos de texto con resultados son fácilmente manejados con herramientas de compresión *estándares* (gzip/gunzip, por ejemplo), no es tan sencillo manejar planillas de cálculo con 65000 filas. Por supuesto siempre se puede recurrir al *submuestreo* de los datos. Este *pre-procesamiento* de los datos de salida de los comandos ping se puede hacer con una combinación de tareas de líneas de comandos, dentro de editores de texto y también en las mismas planillas de cálculo. Generalmente, se tienen más datos de los que se necesitan y algunas veces

de los que se pueden manejar de manera estándar en las planillas de cálculo usuales. Algunas formas sencillas para el submuesteo se pueden hacer, por ejemplo, con comandos del tipo

```
grep -n " " salida | \
grep -v 0:[A-Z] | \
grep -v 2:[A-Z] | \
grep -v 4:[A-Z] | \
grep -v 6:[A-Z] | \
grep -v 8:[A-Z] | \
cut -d: -f2-10 > salida_div_2
```

que no hace nada más que tomar una de cada dos *muestras*, considerando que una y sólo una muestra está contenida en una línea de texto.

4. Resultados

Se llevaron a cabo los experimentos detallados anteriormente, donde se dedicó una computadora del CeTAD para ejecutar el script mostrado en la sección anterior y no se definió ninguna política sobre la computadora del aula de computadoras para verificar también los períodos de encendido/apagado de la misma. Los experimentos se llevaron a cabo en período de clase y, por lo tanto, la sala de computadoras no es de uso exclusivo para cómputo paralelo (en este caso, para llevar a cabo la experimentación).

Para la presentación de los mensajes pequeños se eligió mostrar sobre el eje **x** el tiempo de la muestra (día de la semana, fecha y hora en formato hh:mm:ss) y en el eje **y** el tiempo de ida y vuelta (sl rtt: round trip time) del paquete ICMP generado por el comando ping. Los períodos en los que no aparecen datos son de desconexión, es decir que (por alguna razón) no había conectividad entre los clusters. También se elige eliminar los tiempos excesivamente altos de los gráficos por tres razones:

- son menos del 1 por mil.
- están siempre ligados a una pérdida de conexión. Es decir que inmediatamente después de estos tiempos la conexión entre los clusters ya no es posible.
- cambian la "escala" de los gráficos y los demás datos casi no se pueden visualizar.

Para la presentación de los resultados de los mensajes grandes, se ha elegido mostrar en el eje **x** el tiempo de la muestra (día de la semana, fecha y hora) y en el eje **y** el ancho de banda que corresponde. La presentación de los resultados en términos de ancho de banda es interesante por dos razones:

- El ancho de banda *disponible* (para las aplicaciones paralelas) es lo que se quiere *estimar* con estos mensajes.
- Determina un máximo absoluto, que está dado por el hardware de interconexión. Más específicamente, el máximo ancho de banda será el máximo ancho de banda del hardware con el mínimo rendimiento de todos los segmentos de interconexión (enlaces entre dos routers intermedios) entre los clusters.

En las subsecciones que siguen se muestran los resultados y se comentan las características más importantes de la interconexión que se pueden observar a partir de estos resultados. Inicialmente se muestran los datos de una semana de clase y luego dos días en particular de esa semana. Con el objetivo de mostrar el rendimiento de la interconexión con los dos clusters disponibles y la red de interconexión mayormente libre se incluyen resultados de una semana de vacaciones.

4.1. Experimentos con *Recursos Compartidos Utilizados*

Los primeros resultados corresponden a una semana de actividad *normal* tanto en los laboratorios involucrados (III-LIDI y CeTAD) como en la sala de computadoras que se utiliza desde el III-LIDI, como en la red de interconexión de los clusters. La sala de computadoras que se utiliza desde el III-LIDI se *comparte* con horarios de clases (incluyendo la computadora involucrada en los experimentos), la computadora del CeTAD está completamente dedicada a los experimentos y las redes involucradas (tanto *dentro* de cada cluster como entre los clusters) es compartida con todo el tráfico estándar de la red Internet B de la cual los clusters son subredes.

4.1.1. Una Semana *Completa* (Lunes a Sábado)

La Fig. 1 muestra los resultados obtenidos de tiempos de ida y vuelta (*rtt: round ttrip time*) en milise-

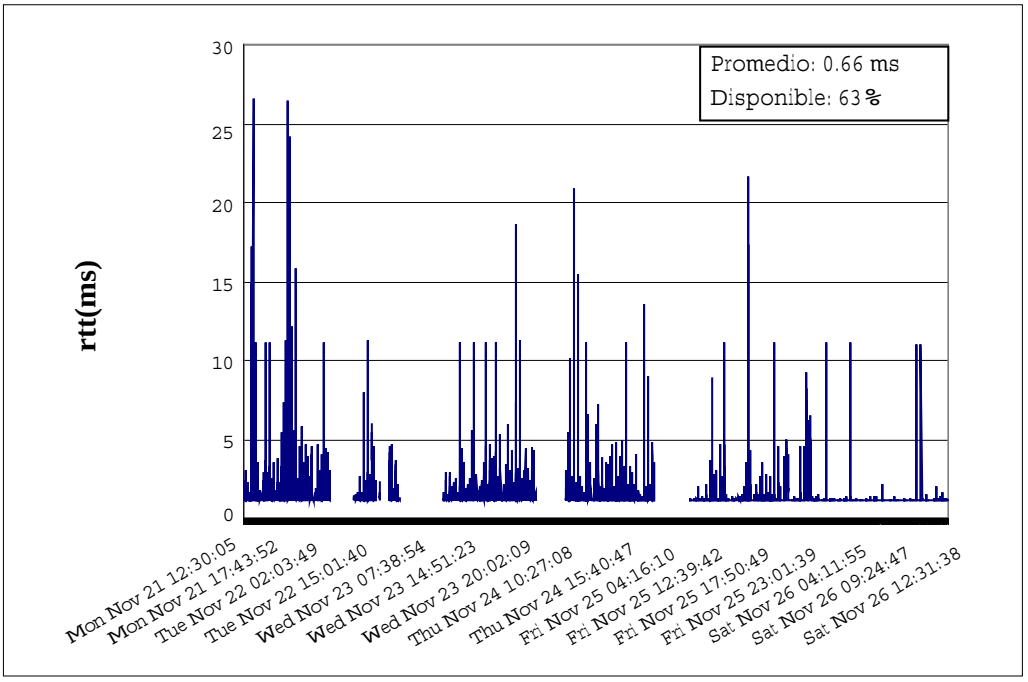

Figura 1: Tiempos de Mensajes de 8 Bytes.

gundos durante una semana de clases (Lunes 21 Nov. 2005 - Sábado 26 de Nov. 2005). En la misma figura se muestran algunos datos resumidos: en promedio, cuando la conexión no falló, el tiempo en una sola dirección (one way message time) es de 0.66 ms (Promedio), y la conexión estuvo disponible el 63 % del tiempo durante el cual se tomaron las muestras (Disponible). Los espacios en los cuales no se muestran resultados son los correspondientes a los períodos en los cuales los paquetes ICMP no tuvieron respuesta. Se debe notar que para estos experimentos no se usó la opción -w y, por lo tanto, los *vacíos* en la figura no necesariamente son proporcionales a los períodos de pérdida de conexión. Estos períodos de pérdida de conexión, por otro lado, se corresponden directamente con el apagado de la computadora *no disponible* del lado del III-LIDI, no con la *caída* de enlaces/routers. Respecto del tiempo de *latencia* que se quiere aproximar/cuantificar, se debe recordar que una interconexión TCP en una red local Ethernet de 100 Mb/s tiene una latencia de aproximadamente 0.052 ms. Aunque *a priori* se podría suponer que la conección no es *confiable*, en realidad los mayores problemas de conectividad surgen de la falta de control sobre las máquinas involucradas. De hecho los mayores períodos en los cuales los comandos ping no se completan *satisfactoriamente* (con un *ICMP echo reply*) se dan por las noches y/o a partir de la finalización de una clase. Siguiendo con este punto, a partir de experimentos que se llevaron a cabo durante todo casi el mes de Noviembre de 2005 solamente hubo una desconexión que no se debió a este tipo de problemas de falta de control sobre las máquinas involucradas: durante un fin de semana no hubo energía eléctrica y/o falló intermitentemente durante la mayor parte del fin de semana. Todos los demás inconvenientes para las respuestas a los comandos ping se debieron al apagado de la computadora que se accede del lado del III-LIDI.

La Fig. 2 muestra los resultados obtenidos de ancho de banda (para paquetes ICMP de 20 KB) en bytes/segundo durante la misma semana de la figura anterior: una semana de clases (Lunes 21 Nov. 2005 - Sábado 26 de Nov. 2005). Está claro que estos experimentos con mensajes *grandes* tuvieron los mismos problemas de conectividad que los experimentos con mensajes *pequeños*. Por lo tanto, el porcentaje de disponibilidad de la conexión es el mismo: 63 % del tiempo de los experimentos. Aunque la impresión visual de la Fig. 2 pueda indicar otra cosa, el promedio de ancho de banda es *muy bueno* teniendo en cuenta que la red física tiene la capacidad de 10 Mb/s. Sin embargo, en términos relativos con lo que sucede en cada uno de los clusters, lo mínimo que se suele tener disponible es 100 Mb/s, lo cual en teoría podría tranferir datos a 12.5 MB/s y, más realista, 10 MB/s. En este sentido, el ancho de banda es de menos del 10 % de lo que se tiene en cada uno de los clusters. Por otro lado, se debe recordar que la

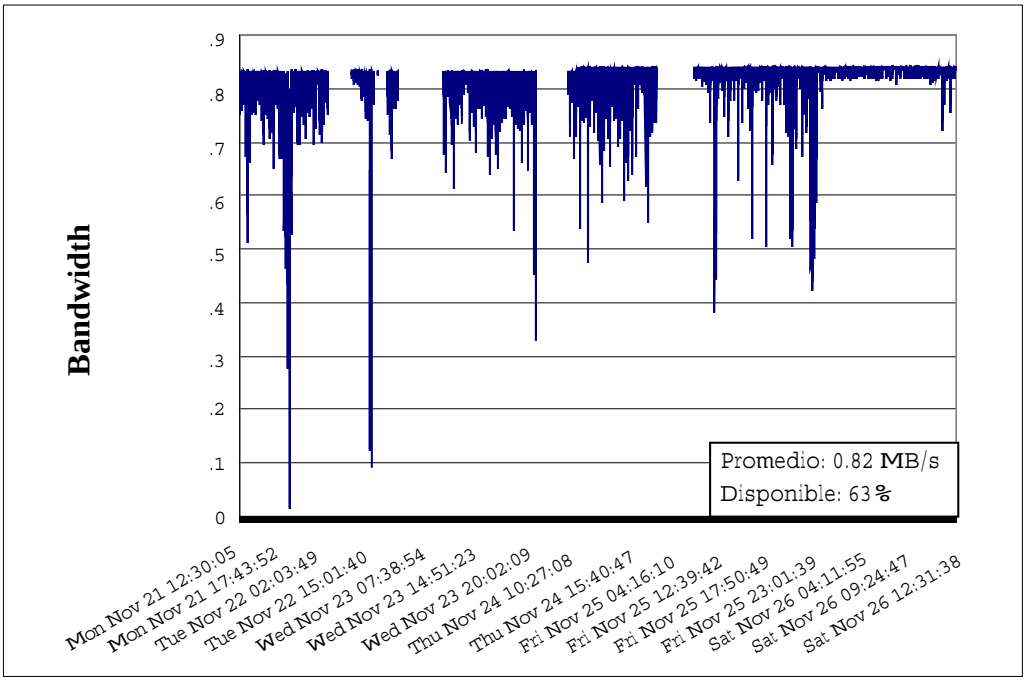

Figura 2: Ancho de Banda (Paquetes de 20KB).

aproximación del ancho de banda con paquetes ICMP de 20 KB deja de lado muchos detalles propios de las conexiones entre procesos de una aplicación paralela que normalmente involucran sobrecarga que no se tiene en cuenta o no se puede cuantificar de esta manera.

Quizás a modo de resumen/conclusión de los resultados que se han mostrado hasta aquí, la latencia tiene más de un orden de magnitud de penalización sobre la que se puede obtener en una red local con TCP. Volviendo al contexto de cómputo paralelo, tener una latencia de 0.66 ms puede generar muchas restricciones a nivel de granularidad mínima posible de mantener sin penalización excesiva de rendimiento. Por el lado de la estimación del ancho de banda, teniendo en cuenta el rendimiento de la red de interconexión, el porcentaje que se obtiene es relativamente bueno (aproximadamente el 80 %). Sin embargo, dado que la red de conexión intercluster es 10 veces más lenta que las de más bajo costo utilizadas en los clusters, este rendimiento es comparativamente muy bajo. Volviendo al contexto de cómputo paralelo, esto también implica serias restricciones en términos de rendimiento. Aunque la idea de escalar las aplicaciones siempre ha sido muy apropiada para el rendimiento de cómputo paralelo (en el área de cómputo numérico intensivo, al menos), esta idea no parecería ser *aplicable* de manera inmediata en el contexto de cómputo intercluster. Es claro que los cambios a nivel de latencia como de ancho de banda con las conexiones intercluster indican que las aplicaciones paralelas tendrán que, de alguna manera, *adaptarse* al rendimiento disponible de la red de interconexión. Lo que parece ser otro *problema a resolver* es el de la disponibilidad de las computadoras, más que el de la disponibilidad de la interconexión. Claramente, las interconexiones dedicadas al tráfico Internet son compartidas pero, también, tienden a ser *estables* en cuanto a disponibilidad. No es lo que sucede en términos de las computadoras, donde compartidas, tal como lo muestran los experimentos, suele ser sinónimo de *no disponibles*.

Con el objetivo de visualizar mejor los resultados, se han elegido dos de los días mostrados antes para graficar los datos obtenidos. En particular, se mostrarán tanto el día con menor disponibilidad como el de mayor disponibilidad de la interconexión.

4.1.2. Día de Menor *Disponibilidad* de Interconexión

Los tiempos de *latencia* (mensajes de 8 bytes) del día con menor disponibilidad de interconexión se muestran en la Fig. 3, donde la conexión funcionó *satisfactoriamente* alrededor del 46 % del tiempo total (24 hs.). Aunque reiterativo, se hace necesario recordar que los tiempos de inactividad se corresponden con el apagado de la máquina involucrada en los experimentos *del lado del III-LIDI*. Básicamente, el último docente de la clase del día anterior apagó todas las computadoras de la sala y los docentes correspondientes a este día en particular encendieron las máquinas para sus respectivas clases y luego volvieron a dejar

todas las computadoras apagadas. Es muy interesante notar que el valor promedio de la latencia es de

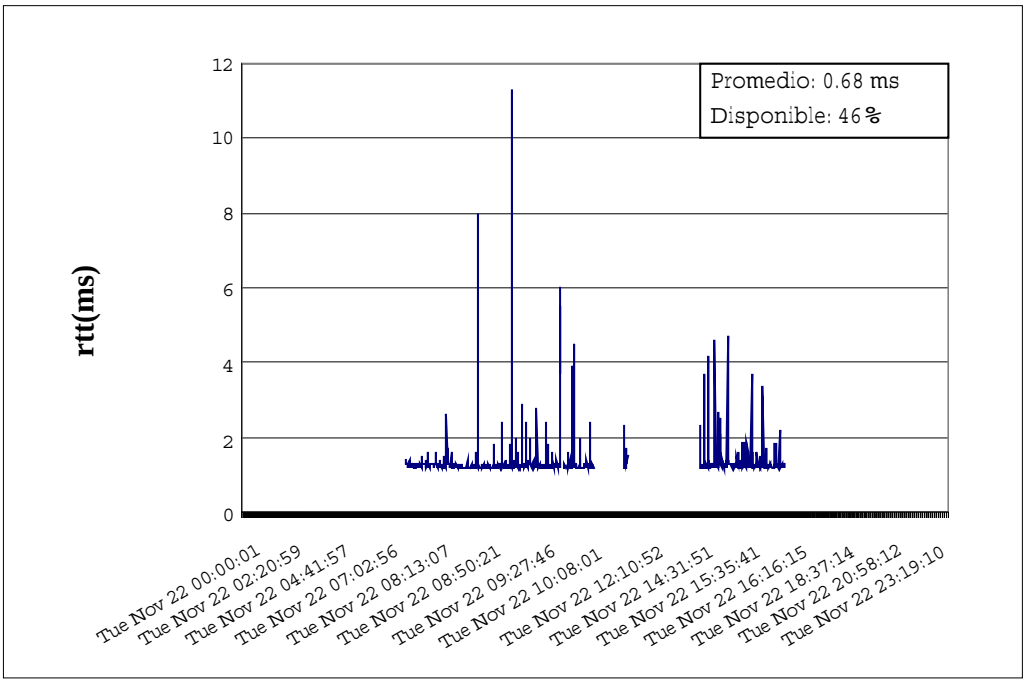

Figura 3: Startup en un Día con Baja Disponibilidad de la Interconexión.

muy poco más del 3 % con respecto al de toda la semana, tal como se muestra en la Fig. 1. Esta relación se da a pesar de que, en realidad, los datos corresponden casi en su totalidad al tiempo durante el cual la sala de computadoras se usó para clases. Por lo tanto, la computadora *del lado del III-LIDI* solamente estuvo disponible durante los períodos de tiempo de las clases y es importante recalcar que durante tales períodos se dan dos condiciones muy importantes e interesantes a tener en cuenta:

- Son períodos de uso estándar de Internet en términos de la red de interconexión. Se podría decir que son los horarios *clásicos de oficina*, es decir cuando la mayoría de los usuarios de Internet están efectivamente utilizando Internet desde sus computadoras en las facultades involucradas.

- Son períodos en los cuales se aumenta el tráfico relacionado con Internet generado desde/hacia la misma sala de computadoras donde se dan clases, dado que los alumnos normalmente utilizan algún tipo de servicio relacionado con Internet desde la misma sala.

También es importante que aunque hay picos de tiempo mucho mayores al promedio, su importancia relativa y, por lo tanto, su frecuencia es bastante baja. Y también es posible observar a partir de la comparación de la Fig. 1 con la Fig. 3 que los picos de latencia con la sala mayormente ocupada son menores a otros picos de latencia que se dieron durante otros tiempos de la semana, por lo tanto los picos más importantes no son los que tienen relación directa con el tráfico relacionado con Internet desde/hacia la sala de computadoras.

Los datos correspondientes del *ancho de banda* (estimado a partir de paquetes ICMP de 20 KB) se muestran en la Fig. 4, donde también se muestra el ancho de banda promedio en los períodos en los cuales la interconexión funcionó *correctamente*. Comparando la información de la Fig. 4 con la de la Fig. 2 es significativo que el promedio de *ancho de banda* se mantiene sin cambios, a pesar de que tanto la computadora involucrada en los experimentos como la red de interconexión están siendo utilizadas por otras aplicaciones. Evidentemente estas aplicaciones no imponen una carga significativa en la red de interconexión de forma tal que los paquetes ICMP de 20 KB se vean retrasados. De esta manera se podría confirmar (quizás *indirectamente*) que el ancho de banda no está afectado tanto por el hecho de compartir la red de interconexión sino por el hardware/software de esta interconexión. También es muy significativo que hay un solo *pico* (*valle*, en realidad), que identifica una caída importante en el rendimiento de la red de interconexión aunque tanto el horario como el uso de la sala están relacionados con uso relativamente *máximo* de los servicios de comunicaciones relacionados con Internet.

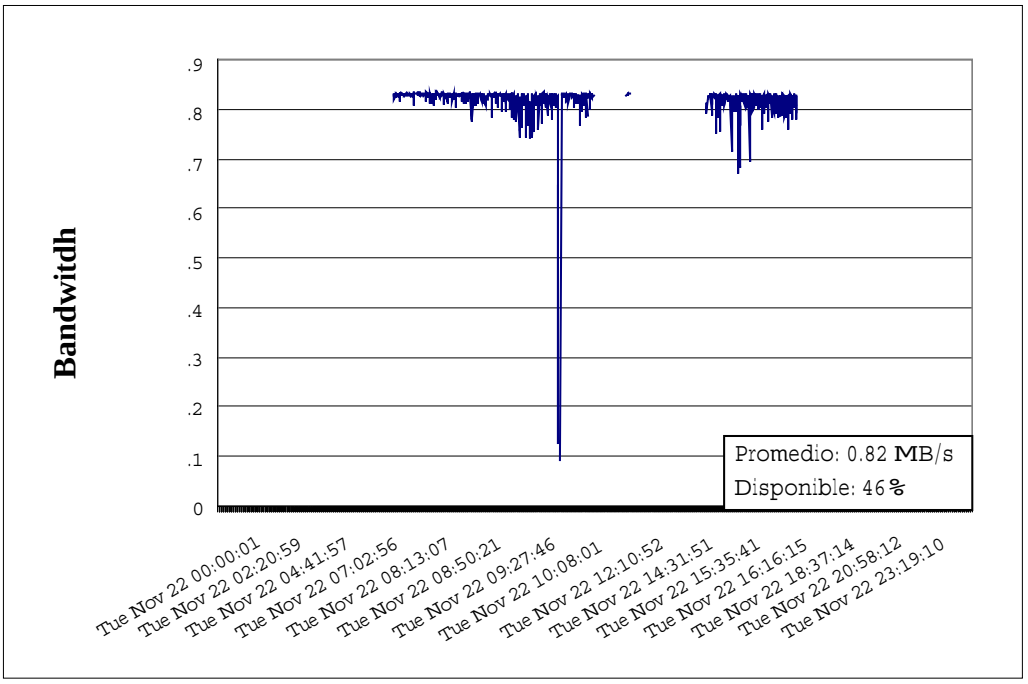

Figura 4: Ancho de Banda en un Día con Baja Disponibilidad de la Interconexión.

Parece bastante claro en este punto que toda la red (en sentido genérico) relacionada con la interconexión de los clusters utilizados está directamente orientada al uso de los servicios de Internet hacia/desde fuera de la red Internet B de la cual forman parte ¿Por qué? si los servicios de Internet internos fueran utilizados en forma intensiva, claramente las subredes, funcionando mayormente a 100 Mb/s *rápidamente* consumirían todo el ancho de banda disponible de las interconexiones de 10 Mb/s estándares entre los routers intermedios involucrados. Esto seguiría siendo cierto aún cuando todas las interconexiones entre los routers fueran de 100 Mb/s, dado que hay muchas subredes involucradas entre los dos clusters en los cuales se llevaron a cabo los experimentos. Dicho de otra manera, las conexiones cliente/servidor usuales relacionadas con Internet (ej.: http, ftp) en su mayoría tienen uno de los procesos (cliente o servidor) fuera de la red B Internet de la cual forman parte los clusters. Dado que es bastante usual que las salidas al exterior de cada red Internet sean bastante más lentas que 10 Mb/s, está claro que estas conexiones tienden a dejar libre el ancho de banda disponible entre las subredes internas. En cierta forma, a diferencia de los clásicos *backbones* de las universidades americanas o europeas, las interconexiones internas entre las subredes de la red B aquí funcionan como multiplexores/cuellos de botella hacia la salida a Internet con respecto a la red Internet B de la cual los clusters forman parte. De hecho, aún en los períodos de mayor uso de Internet por parte de los usuarios, la red de interconexión de 10 Mb/s queda a disposición y puede ser aprovechada para uso de transferencia de datos entre las computadoras de la red *interna*. Quizás ésta sea una de las conclusiones más importantes de estos experimentos, aunque no es claro si esta situación se da en todas las Universidades/instituciones de investigación.

4.1.3. Día de Mayor *Disponibilidad* de Interconexión

Los tiempos de *latencia* (mensajes de 8 bytes) del día con mayor disponibilidad de interconexión se muestran en la Fig. 5, donde la conexión funcionó *satisfactoriamente* alrededor del 65 % del tiempo total (24 hs.). Esta es una de las figuras donde más claramente se nota a nivel visual el problema que generan los períodos en los cuales los comandos ping no se completaron satisfactoriamente. Es evidente que hay más datos de los períodos en los cuales hay conexión que los períodos en los cuales no la hay. De hecho, el espacio sin datos en la Fig. 5 claramente no representa de manera visual que es el 35 % del total (debería ser de bastante más de 1/4 del total). Como se aclaró antes, esto se debe a que los comandos ping sin la opción -w toman mucho tiempo cuando no se completan satisfactoriamente, a pesar de tener claramente especificado:

- La cantidad de paquetes.

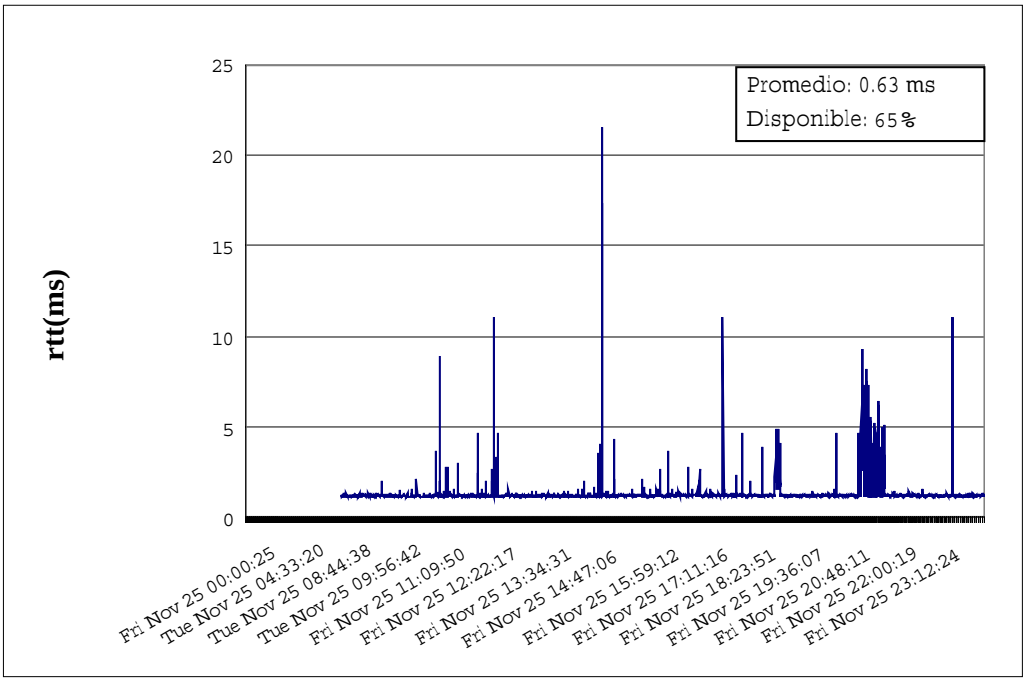

Figura 5: Startup en un Día con Alta Disponibilidad de la Interconexión.

- El tiempo de espera entre paquetes de 1 segundo.

A partir de estos parámetros se podría suponer que, como el comando ping es para verificar la existencia de conexiones, se debería terminar el comando en aproximadamente k segundos, siendo k la cantidad de paquetes.

Como era de esperar a partir del análisis de la Fig. 1 y la Fig. 3, el promedio de tiempo de latencia de un día de alta disponibilidad es menor que el promedio de toda la semana. Esto se debe a que en un día de alta disponibilidad de conexiones muchas de las muestras corresponden a la noche y otros períodos de tiempo en los cuales la red de interconexión no se utiliza. También era de esperar que la diferencia no es muy grande: menos del 5 % menor que el promedio de toda la semana.

La evolución durante un día de los tiempos de latencia podría considerarse *normal* excepto por lo que sucede aproximadamente entre las 20:00 y las 21:00. En este perído se puede suponer que no hay grandes requerimientos de interconexión en la red, pero sin embargo se tienen muchos tiempos de latencia relativamente altos (entre 3 y 9 ms) respecto del promedio (0.63 ms). Una de las posibles razones es que (mayormente *del lado del CeTAD*) se dejan conexiones automáticas para requerimientos de archivos vía ftp (o servicios similares) que suelen comenzar inmediatamente después de lo que se considera *horario de oficina*. Esto naturalmente genera una carga importante sobre la red de interconexión que no implica pérdida de rendimiento para usuarios interactivos sino para aplicaciones *batch*, lo cual no es un problema importante sino todo lo contrario desde la perspectiva de los administradores de las redes involucradas, dado que *descargan* la red de los horarios en que se llevan a cabo mayormente las conexiones con *usuarios interactivos*. Sin embargo, está claro que para las aplicaciones paralelas esto sí es un problema importante dado que aún los mensajes cortos ICMP se ven retrasados en una proporción muy importante, lo cual implica que la latencia de comunicaciones también será afectada en términos de pérdida de rendimiento. Sin embargo, las condiciones vuelven a ser más favorables a partir de las 21:00 (aproximadamente) y, de hecho, la red estará mayormente libre incluyendo el sábado (tal como se puede ver en la Fig. 1 y en la Fig. 2).

Los datos correspondientes del *ancho de banda* (estimado a partir de paquetes ICMP de 20 KB) se muestran en la Fig. 6, donde también se muestra el ancho de banda promedio en los períodos en los cuales la interconexión funcionó *correctamente*. También en esta figura se puede notar visualmente que la proporción de datos del tiempo de desconexión no es necesariamente acorde al 65 % del tiempo en el cual los comandos ping se completaron satisfactoriamente. El ancho de banda promedio estimado a partir de paquetes ICMP de 20 KB (0.83 MB/s) no cambia mucho respecto de los del resto de la semana (0.82 MB/s) ni respecto del día con menor disponibilidad de interconexión (0.82 MB/s). Quizás sería esperable

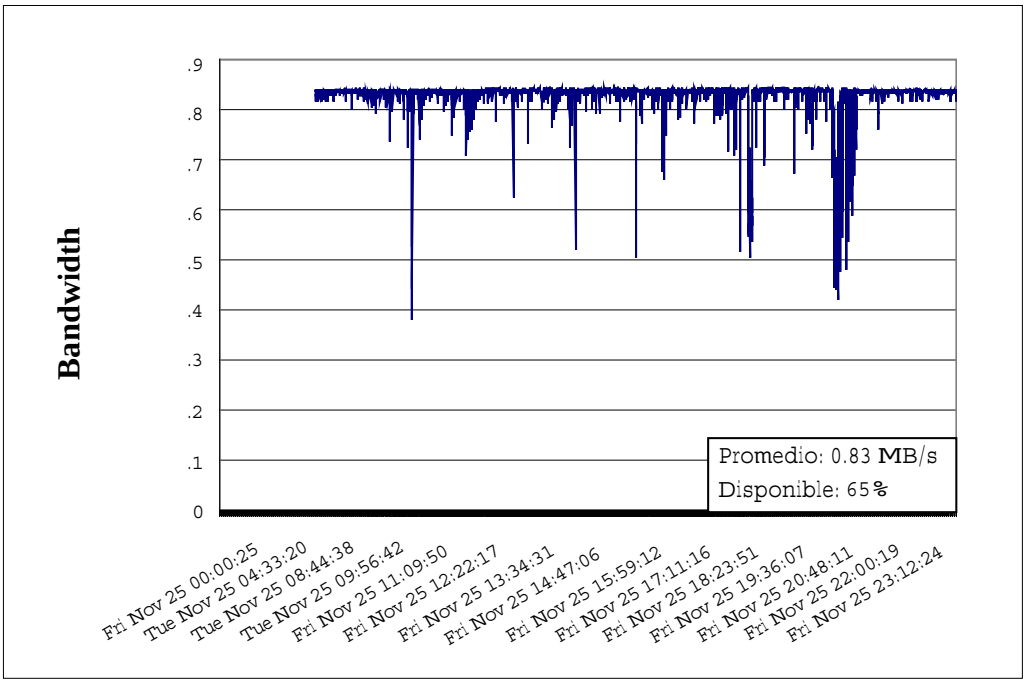

Figura 6: Ancho de Banda en un Día con Alta Disponibilidad de la Interconexión.

un aumento, dado que se toman datos de un período de tiempo mayor por las noches involucradas, pero se equilibra el tráfico que se menciona antes y que se puede visualizar en esta Fig. 4 también. Nuevamente entre las 20:00 y las 21:00 hay un incremento notable en el tráfico presente en la red de interconeción de los dos clusters, y esto se puede notar en la caída de rendimiento a nivel de ancho de banda. Es importante recalcar que este tráfico *extra* respecto del normal afecta casi por igual al tiempo de latencia (mensajes relativamente pequeños) como al ancho de banda (mensajes relativamente mayores). En el caso del ancho de banda, en este período de las 20:00 a las 21:00, el ancho de banda está por debajo de 0.8 MB/s cuando el promedio es de 0.83 MB/s (lo cual indica, además, que se tienen períodos de mejor ancho da banda aún). Tal como se puede notar en la figura de tiempos de latencia (Fig. 5), la situación mejora notablemente aproximadamente a partir de las 21:00. Es importante notar que los perídos de mucho tráfico en la red son claramente identificables, son relativamente pequeños (respecto del total) y, además son relativamente pocos. En los resultados que se han mostrado hasta aquí, en toda la semana se tiene solamente uno: el viernes por la noche, inmediatamente después de terminado el horario de oficina. La identificación automática de estos períodos parece sencilla y factible.

4.2. Experimentos con *Recursos Básicamente Libres*

Con el objetivo de analizar el rendimiento de la red de interconexión existente entre los clusters en el mejor de los casos, se llevaron a cabo los mismos experimentos, pero durante el mes de enero de 2006. Se podría decir que específicamente este mes es el de menos actividad en general y de menos actividad sobre la red de interconexión en particular. Aunque es posible que las mismas condiciones se den durante las noches o en los fines de semana en cualquier época del año, también es cierto que algunas conexiones como las asociadas a los servicios de e-mail son más frecuentes y/o de mayor volumen en los períodos en los cuales hay actividades normales en las facultades involucradas. Además, también es posible, durante Enero, verificar el funcionamiento de la red de interconexión asumiendo que hay poco personal de mantenimiento de la misma. Se supone que en estas condiciones, si hay un problema de conectividad es muy probable que lleve más tiempo su identificación/reparación (aunque la probabilidad de que haya un problema debería ser menor dado que hay menor tráfico en la red).

La Fig. 7 muestra el *round trip time* de los paquetes de 8 bytes en una semana similar a la mostrada anteriormente en la Fig. 1: desde un lunes a las 12:30 hasta el sábado siguiente a las 12:30. Una de las primeras diferencias que se pueden notar comparando la Fig. 7 con la Fig. 1 es que no aparecen períodos en los cuales la conexión haya fallado. Más específicamente, cada paquete ICMP *echo-request* tuvo su

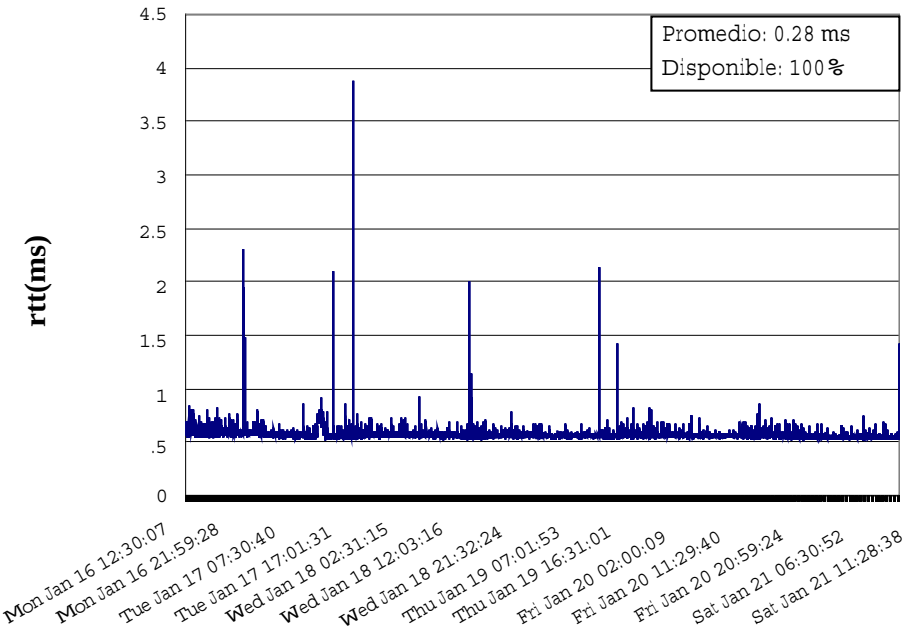

Figura 7: Tiempo de Latencia en una Semana con Red de Interconexión *Libre*.

correspondiente *echo-reply* y, por lo tanto, la conexión estuvo disponible durante todo el tiempo reportado en la Fig. 7 a diferencia de lo que se muestra en la fig. 1. La segunda gran diferencia entre este período de inactividad en la red de interconexión con respecto al de una semana de clases se puede verificar en los *picos* de tiempo de *latencia*: mientras que en una semana de actividad normal pueden haber varios picos de tiempo de más de 10 ms (con algunos, aunque muy pocos, de más de 50 ms), en una semana dentro del período de vacaciones no se superan los 4 ms de tiempo de ida y vuelta para un paquete ICMP de 8 bytes y hay un sólo pico de más de 2.5 ms. También relacionado con este punto se puede identificar una diferencia más, que también es muy importante en la comparación de la Fig. 7 con la Fig. 1: el tiempo promedio de latencia de mensajes (estimado a partir del tiempo de un paquete ICMP de 8 bytes en una sola dirección) es de aproximadamente 0.28 ms con la red de interconexión casi sin uso, mientras que es de 0.66 ms cuando la red de interconexión tiene carga *normal*. Esto representa una mejora de aproximadamente 58 %, lo cual de todas maneras no implica un buen tiempo de latencia de comunicaciones para procesamiento paralelo. Resumiendo lo que se puede apreciar a partir de la Fig. 7 y comparándolo con lo que sucede en una semana de actividades *normales* (Fig. 1):

- No hay problemas de falta de conexión entre los clusters. En principio, no se tienen *interferencias* en la conexión por las otras actividades en los clusters (dado que no las hay) y tampoco aparecen problemas de conexión por ser un período con menor actividad en general.
- No hay picos de tiempos de latencia muy elevados, como sucede en los períodos de actividades usuales. Los picos son muy pocos y de poca magnitud (valor absoluto).
- Los tiempos promedio de latencia mejoran ostensiblemente a más de la mitad de los que se obtienen en tiempos de actividades normales sobre la red de interconexión.

Por lo tanto, como era de esperar, el rendimiento de la interconexión de ambos clusters es cercana al óptimo. Se tiene a partir de aquí una primera posibilidad de cuantificación de las diferencias entre ambos períodos de actividad y sus correspondiente rendimiento.

Los datos correspondientes del *ancho de banda* (estimado a partir de paquetes ICMP de 20 KB) se muestran en la Fig. 8, donde también se muestra el ancho de banda promedio. Aunque no se pueden observar en la Fig. 8, hubo algunos paquetes ICMP *echo request* sin su correspondiente *echo reply*. Sin embargo, solamente fueron 12 comandos ping los que no se completaron satisfactoriamente con respecto a más de 60000 que sí se completaron satisfactoriamente. Es muy interesante que esto sucedió en algunas madrugadas e, incluso, en la madrugada del sábado. Sin embargo, la red no tuvo problemas por cuanto

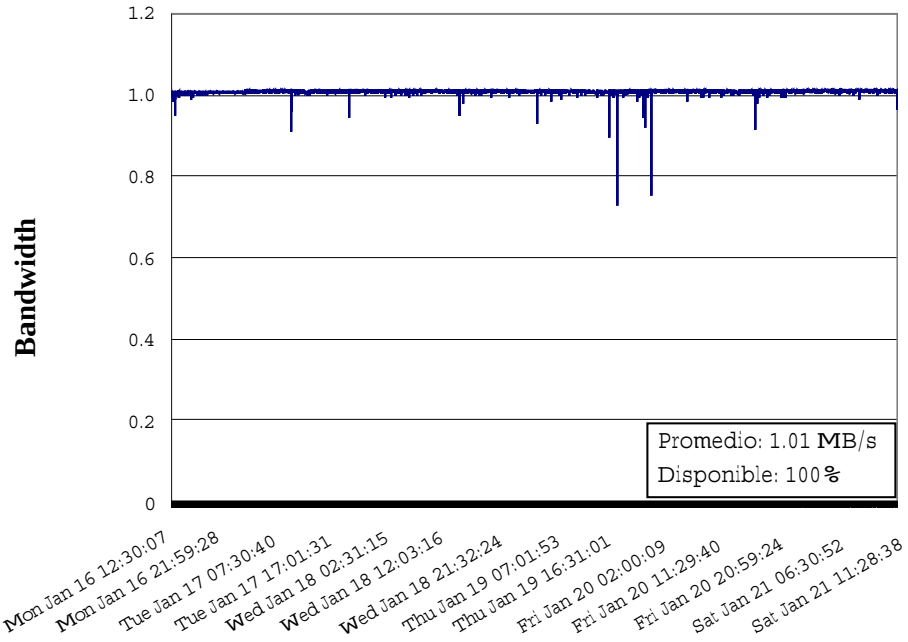

Figura 8: Ancho de Banda en una Semana con Red de Interconexión *Libre*.

fueron paquetes aislados y sin mucha relación entre ellos (en cuanto a los *instantes de tiempo* en los cuales sucedieron). Como era de esperar, el ancho de banda mejora bastante con respecto al de una semana en un período de clases (Fig. 2) y, de hecho, se obtiene aproximadamente lo mismo que se obtendría en una red de interconexión local con Ethernet de 10 Mb/s.

5. *Scripts* Utilizados en los Experimentos

Esta sección se utiliza solamente para tener un único espacio de referencia para los *scripts* utilizados en los experimentos. Básicamente se incluye (*copia*, de hecho) el script para recolectar los resultados y el script para reducir la cantidad de muestras a la mitad. En el primer script (copiado de una de las secciones anteriores) se asume que `IPdestino` es la dirección IP de la computadora que responderá los ICMP *echo request* y, además, los resultados se dejan en los archivos pp1 (de los paquetes de 8 bytes) y pp20k (de los paquetes de 20 KB).

```
while true; do \
( \
  date >> pp1; \
  ping -c 3 -i 1 -s 8 -w 4 IPdestino > ver; \
  cat ver | grep [A/] >> pp1; \
  date >> pp20k; \
  ping -c 1 -s 20000 -w 2 IPdestino > ver; \
  cat ver | grep [A/] >> pp20k; \
  sleep 5 \
); done
```

Y asumiendo que el archivo de comandos se llama, justamente, script, la ejecución de este script se puede lanzar con una línea de comando del tipo:

at -f script now + 1 minute

El segundo script contiene una secuencia de comandos posible para reducir a la mitad la cantidad de muestras recolectadas en los archivos pp1 y pp20k. Se asume, en este caso, que los archivos están comprimidos en formato .gz (gzip) y los archivos correspondientes con la mitad de las muestras serán ver1_half (con los datos de los paquetes de 8 bytes) y ver20k_half (con los datos de los paquetes de 20 KB). Este script en particular puede ser mucho más elaborado utilizando awk/gawk, pero se deja de esta manera

solamente como un ejemplo con la combinación de los comandos grep y cut. De hecho, los comandos wc no son más que para conocer la cantidad de filas que deberán ser manejadas en las planillas de cálculo que se utilicen.

```
gunzip pp1.gz
grep -n " " pp1 > ver
cat ver | grep -v 0:[A-Z] | grep -v 2:[A-Z] | grep -v
4:[A-Z] | grep -v 6:[A-Z] | grep -v 8:[A-Z] >  ver_half
cut -d: -f2-10 ver_half > ver1_half

gunzip pp20k.gz
grep -n " " pp20k > ver
cat ver | grep -v 0:[A-Z] | grep -v 2:[A-Z] | grep -v
4:[A-Z] | grep -v 6:[A-Z] | grep -v 8:[A-Z] >  ver_half
cut -d: -f2-10 ver_half > ver20k_half

wc -l ver1_halfnn
wc -l ver20k_halfnn
```

Referencias

[1] Litzkow M., M. Livny, M. Mutka, "Condor - A Hunter of Idle Workstations", Proceedings of the 8th International Conference of Distributed Computing Systems, pages 104-111, June, 1988.

[2] Mutka M., M. Livny, "Profiling Workstations' Available Capacity for Remote Execution", Performance '87,12th IFIP WG 7.3, pp. 529-544, December 1987.

[3] Mutka M., M. Livny, "The Available Capacity of a Privately Owned Workstation Environment", Performance Evaluation, vol. 12, no. 4 pp. 269-284, July, 1991.

[4] Postel J., RFC 792 - Internet Control Message Protocol, DARPA Internet Program, Protocol Specification, September 1981.

[5] Tinetti F. G. "Cómputo Paralelo en Redes Locales de Computadoras", Tesis Doctoral, Universidad Autónoma de Barcelona, Facultad de Ciencias, Marzo 2004.

[6] Tinetti F. G., Aróztegui W., "Bibliotecas de Pasaje de Mensajes y Cómputo Intercluster", Reporte Técnico PLA-003-2005, III-LIDI, Facultad de Informática, UNLP, CeTAD, Facultad de Ingeniería, UNLP, Argentina, Septiembre 2005. Disponible en https://lidi.info.unlp.edu.ar/~fernando/publis/portsrep.pdf

[7] Tinetti F. G., Aróztegui W., "Instalación y Configuración de ssh para Cómputo Intercluster", Reporte Técnico PLA-002-2005, III-LIDI, Facultad de Informática, UNLP, CeTAD, Facultad de Ingeniería, UNLP, Argentina, Junio 2005. Disponible en https://lidi.info.unlp.edu.ar/~fernando/publis/intercl1.pdf

[8] Tinetti F. G., Quijano A. A., "Capacidad de Comunicaciones Disponible para Cómputo Paralelo en Redes Locales Instaladas", Proceedings VIII Congreso Argentino de Ciencias de la Computación (CACIC), Fac. de Ciencias Exactas y Naturales, Universidad de Buenos Aires, Buenos Aires, Argentina, p. 125, Octubre 2002.