

Sistema de predicción y evaluación de disponibilidad operativa de recursos en Desktop Grids

Sergio Ariel Salinas¹, Carlos García Garino^{1,2} y Alejandro Zunino³

¹ Instituto para las Tecnologías de la Información y las Comunicaciones (ITIC), UNCuyo, Mendoza, Argentina {ssalinas, cgarcia}@itu.uncu.edu.ar

² Facultad de Ingeniería, UNCuyo, Mendoza, Argentina

³ ISISTAN, Facultad de Ciencias Exactas, UNICEN, Tandil, Argentina
azunino@exa.unicen.edu.ar

Abstract. La computación voluntaria permite la creación de Desktop Grids mediante la integración de distintos recursos dispersos en diferentes organizaciones. El objetivo de estos entornos es utilizar el tiempo ocioso de los recursos para ejecutar procesos de usuarios de la Grid. Los recursos en este tipo de Grid, se caracterizan por presentar un patrón de comportamiento que permite aplicar técnicas para predecir el estado de los mismos. De esta manera para cada instante de tiempo es posible predecir el estado del recurso. El sistema de planificación puede utilizar esta información para seleccionar aquellos recursos que mejor se ajusten a los requerimientos de tiempo de los procesos que debe asignar. Para ello es necesario proveer una heurística que permita comparar la predicción de disponibilidad de cada recurso. Esta información permite optimizar la asignación de procesos que realiza el sistema de planificación de la Grid Computacional. En este trabajo, se presenta un sistema para predecir y evaluar la disponibilidad operativa de recursos en Desktop Grids. Los resultados muestran que el sistema de planificación obtiene mejoras en el tiempo de ejecución de procesos al utilizar la información provista por el sistema propuesto.

1 Introducción

La computación voluntaria reúne esfuerzos de distintas organizaciones para compartir capacidad de cómputo. El entorno de computación distribuida resultante se denomina Desktop Grid. Los recursos incluyen clusters, laptops y en gran parte máquinas de escritorios distribuidas en las organizaciones participantes. El sistema de planificación (SP) de la Grid computacional utiliza el tiempo ocioso de estos recursos para ejecutar procesos de usuarios. Para ello, el SP selecciona los recursos que se ajustan a las características de hardware y software que requieren las tareas a ejecutar. Un factor importante es el tiempo que los recursos estarán disponible para ejecutar los procesos asignados. La disponibilidad depende del uso que el usuario realice del recurso.

Una característica de los recursos en Desktop Grids es que presentan un patrón de comportamiento. Por ejemplo, se espera que las laptops presenten una mayor frecuencia de encendido y apagado del equipo, también una mayor intermitencia en la participación de la red. Las computadoras de escritorios es muy probable que permanezcan encendidas una fracción del día y que se encuentren apagadas en horas de la noche. Los

equipos de computadoras de los laboratorios de las universidades probablemente se utilicen en conjunto en los mismos intervalos de tiempo. Los clusters se espera presenten una mayor dedicación respecto a los recursos mencionados anteriormente.

Debido a que es posible identificar un patrón de comportamiento en los recursos es factible estimar los estados que adoptarán en el tiempo. Distintas técnicas estadísticas se han utilizado para predicción de carga de CPU y fallas de recursos. La predicción del estado de los recursos permitiría al planificador optimizar la selección de los mismos. De esta manera, el proceso de selección incluiría una comparación entre el tiempo de ejecución requerido y el tiempo de disponibilidad estimado para cada recurso. Las técnicas de predicción estiman el posible estado del recurso para un instante de tiempo. De esta manera, es posible crear una ventana de tiempo con la predicción de los distintos estados. El planificador puede utilizar esta información para seleccionar el recurso cuya ventana de tiempo mejor se ajuste a los requerimientos de tiempo de ejecución del proceso. Para ello requiere de alguna medida o heurística para realizar esta comparación.

El presente trabajo, presenta un sistema que realiza una predicción y evaluación del comportamiento esperado de los recursos que permite optimizar el tiempo de ejecución de procesos de usuarios en Desktop Grids. Los experimentos demuestran que el planificador obtiene mejoras en la selección de recursos utilizando la información provista por el sistema.

El trabajo se organiza de la siguiente manera, en la sección 2 se introducen los trabajos relacionados con técnicas de predicción y su aplicación. En la sección 3 se presenta el modelo del sistema propuesto. La sección 4 se presentan los experimentos y resultados. Finalmente en la sección 5 se presentan las conclusiones y trabajos futuros.

2 Trabajos relacionados

Existen diversos esfuerzos que intentan modelar el comportamiento de un recurso con el objetivo de predecir el comportamiento del mismo. La predicción se aplica principalmente en entornos distribuidos para optimizar el balance de carga de CPU y predicción de fallas. En general, los métodos predictivos realizan un análisis del comportamiento pasado del recurso y aplican distintas técnicas para predecir su desempeño. Las técnicas de predicción utilizadas con mayor frecuencia son del tipo estadísticas, por ejemplo análisis de series de tiempo.

El sistema Network Weather Service [1, 2] realiza un análisis de series de tiempo para realizar predicciones sobre el rendimiento de recursos en un entorno distribuido. Para ello el sistema selecciona dinámicamente la técnica predictiva que presenta mayor precisión.

El sistema Real-time Scheduling Advisor (RTSA) [3] realiza un análisis estadístico de series de tiempo para crear una lista de recursos de acuerdo a consultas basadas en restricciones. Estas restricciones, definidas por el usuario, incluyen tipo de CPU, tiempo necesario para llevar a cabo la ejecución del trabajo, plazo de ejecución y nivel de confianza del recurso.

Otra área de aplicación de los modelos predictivos es el análisis y predicción de fallas en recursos. El objetivo es determinar la probabilidad de que un recurso finalice con éxito la ejecución de un trabajo en un intervalo de tiempo. El modelo de predicción

de fallas con filtro presentado en [4] analiza una serie de eventos que indican si un recurso está encendido o apagado. El análisis se realiza sobre datos históricos donde se han filtrado eventos periódicos tales como tareas de mantenimiento del recurso.

Las técnicas de predicción también se utilizan en el contexto de Grid para optimizar el funcionamiento de los schedulers.

La técnica denominada Exponential Smoothing se utiliza para el análisis de series de tiempo y predicción. En [5] se aplica este método para implementar un balance de carga de trabajos dinámico en la Grid computacional PlanetLab [6]. Los resultados demuestran que la predicción lograda es efectiva y permitió mejorar los tiempos de ejecución de trabajos distribuidos.

En [7] se describe un framework P2P para una Grid que realiza una predicción de la disponibilidad de recursos utilizando información relacionada con el grupo de computadoras al que pertenece un recurso. La predicción de la disponibilidad de cada recurso se calcula semanalmente considerando los datos registrados la semana previa. Esta medida se obtiene del cociente entre la cantidad de veces que observó el recurso disponible y el total de observaciones realizadas.

La presente propuesta, a diferencia de los trabajos mencionados, provee al sistema planificador además de la predicción de estado del recurso, un conjunto de indicadores que caracterizan la disponibilidad de los recursos. Esta disponibilidad, permite al planificador seleccionar los recursos con mayor probabilidad de finalizar un trabajo de acuerdo a los requerimientos de tiempo de cada tarea. El sistema propuesto, permite la utilización de distintos métodos predictivos. En la presente propuesta se utilizan algoritmos de clasificación del área de inteligencia artificial ya que tienen buena predicción.

3 Modelo del sistema de predicción y evaluación de disponibilidad

El modelo propuesto tiene como objetivos proveer información descriptiva del recurso, predecir la disponibilidad del mismo en el tiempo y proveer un conjunto de indicadores que permitan evaluar y comparar la predicción obtenida. La arquitectura del modelo propuesto está compuesta por un conjunto de módulos y se muestra en la figura 1. Los módulos permiten observar y registrar el comportamiento del recurso en el tiempo y partir de esta información predecir el posible estado del mismo.

La predicción asigna a cada instante de tiempo un estado del recurso esperado. La disponibilidad del recurso se representa mediante una secuencia de estados en un intervalo de tiempo determinado. El sistema evalúa y provee un conjunto de indicadores relacionados con la disponibilidad. Estas medidas permiten al sistema planificador comparar y seleccionar recursos de acuerdo a los requerimientos de tiempo de las tareas a ejecutar.

El sensor de actividad del recurso es responsable de medir el nivel de actividad de la CPU y la cantidad de memoria RAM disponible para un instante de tiempo determinado. Esta información se almacena generando un historial del comportamiento del recurso. Una vez que se alcanza cierto umbral se eliminan los registros de mayor antigüedad.

El módulo predictor elabora una predicción de los posibles estados en los que se encontrará el recurso en las próximas t unidades de tiempo. Si bien, es posible utilizar

distintos métodos predictivos, en este trabajo se emplean algoritmos de clasificación del área de inteligencia artificial. Los mismos han demostrado ser eficientes y precisos en la clasificación de datos desconocidos en función de observaciones realizadas en el pasado. Es posible estimar el grado de precisión que ofrece cada algoritmo de clasificación, y para este caso, no requieren una capacidad de cómputo importante. De esta manera, es posible aplicar un conjunto de algoritmos de clasificación y utilizar el resultado del clasificador de mayor precisión. A diferencia de los métodos estadísticos, en general estos algoritmos no asumen una distribución de probabilidad de la información.

El módulo de preprocesamiento realiza una transformación de la información registrada por el sensor de actividad del recurso. Este preprocesamiento es necesario para generar el conjunto de entrenamiento requerido por los algoritmos de clasificación utilizados.

El resultado de la predicción es expresado como una ventana de tiempo con los posibles estados del recurso. El módulo evaluador utiliza esta información para calcular un conjunto de indicadores relacionados con la disponibilidad del recurso.

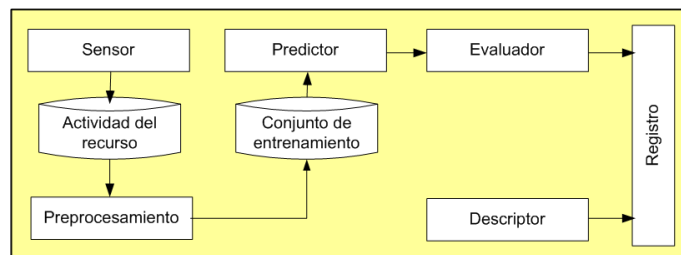


Fig. 1: Modelo Conceptual

El módulo descriptor tiene como objetivo obtener automáticamente datos del recurso tales como, características de hardware, software, datos de configuración de conectividad a la red y servicios disponibles. Esta información, es de utilidad para determinar la capacidad de cómputo del recurso y la forma de accederlo.

Finalmente, el módulo denominado registro es responsable de reunir y administrar información generada por los módulos i.e. predicción y evaluación de la disponibilidad del recurso y características del mismo. El módulo de registro provee la información necesaria para que el sistema de planificación pueda distribuir trabajos en los recursos disponibles. Esta información se encuentra disponible en repositorios donde el módulo de registro envía la información del recurso. En las siguientes subsecciones se presenta cada módulo en detalle.

3.1 Sensor de actividad del recurso

El sensor de actividad del recurso es responsable de monitorear y registrar la actividad del recurso cada un intervalo de tiempo determinado. La unidad de tiempo utilizada para realizar las muestras es el segundo. La información que registra el sensor se almacena

en el recurso y es la siguiente: fecha y hora de la muestra, porcentaje de carga de la CPU, y porcentaje de memoria libre. La fecha y hora se sincroniza con un servidor al momento de iniciar el recurso para evitar inconsistencias. La variación de la carga de CPU y memoria disponible permite luego determinar cambios de estado del recurso. En caso de que el recurso utilice múltiples sistemas operativos entonces la información se almacena en un espacio de almacenamiento común a los mismos.

El funcionamiento del sensor no es invasivo por lo tanto no afecta el rendimiento del recurso. La información registrada se almacena considerando una ventana de tiempo de seis meses desde el momento actual, eliminando periódicamente los registros no comprendidos en ese periodo de tiempo. La información que registra el sensor luego es utilizada por el módulo predictor que se describe en la sección 3.2.

3.2 Módulo Predictor

El objetivo del módulo predictor es estimar y predecir los posibles estados de un recurso para un intervalo de tiempo. Este cálculo se realiza aplicando técnicas de clasificación del área de Inteligencia Artificial. El módulo puede utilizar distintos algoritmos y considerar los resultados de aquel que presente mayor precisión en la predicción.

Por ejemplo, el clasificador C4.5, una extensión del algoritmo ID3 [8], se basa en la generación de árboles de decisión. Este clasificador, en el contexto de la presente propuesta, mostró una precisión del 90 % en el proceso de predicción. El conjunto de entrenamiento utilizado por el algoritmo se obtiene luego de realizar un preprocesamiento de la información obtenida por el sensor.

El resultado del proceso de clasificación se representa mediante dos elementos: el vector de disponibilidad y el encabezado del vector de disponibilidad. El encabezado del vector contiene datos relacionados con el vector de disponibilidad tales como la zona horaria considerada en el cálculo, fecha en que se generó el vector, y el intervalo de tiempo para el cual se realizó la predicción. El encabezado también incluye el clasificador utilizado en la predicción, la precisión del mismo y la frecuencia considerada para realizar la estimación.

El vector de disponibilidad es una secuencia de valores donde cada posición representa el estado esperado del recurso en instantes de tiempo equidistantes. El primer valor de vector representa el estado en el instante de tiempo inicial t_0 , el segundo valor el estado en el instante $t_1 + k$ donde la variable k es la frecuencia considerada para realizar la estimación. La unidad de medida de la variable k es el segundo y por defecto su valor es de 600 segundos.

La extensión del vector depende del intervalo de tiempo para el cuál se realiza la predicción y del valor de la variable k . Por defecto, el vector representa los posibles estados del recurso en las próximas 12 horas desde el instante inicial en que se estima el vector.

El vector de disponibilidad y su respectivo encabezado son utilizados por el módulo evaluador que se describe en la subsección 3.4.

3.3 Módulo de preprocesamiento

Los algoritmos de clasificación requieren un conjunto de registros previamente etiquetados denominado conjunto de entrenamiento para poder clasificar nuevos registros. El módulo de preprocesamiento es responsable de la generación del conjunto de entrenamiento que utiliza el clasificador. Cada un cierto intervalo de tiempo se realiza un preprocesamiento de la información registrada por el sensor de actividad del recurso. El proceso asigna una etiqueta a los valores registrados por el sensor. De acuerdo a los niveles de carga de CPU y memoria RAM se asignan las etiquetas *disponible* y *ocupado*. Los umbrales de carga de CPU y memoria disponible indican la presencia de procesos o uso del recurso por parte del usuario. Estos valores determinan la etiqueta que se debe asignar a cada registro.

El resultado del proceso de preprocesamiento es una tupla de valores que incluye: mes, día, hora, minuto, estado de la CPU, estado de la memoria RAM y estado del recurso. El conjunto de entrenamiento se almacena en un repositorio que luego es utilizado por el clasificador. El repositorio se mantiene actualizado con los últimos registros etiquetados con una ventana de tiempo de seis meses. De acuerdo a esta ventana de tiempo los registros de mayor antigüedad se eliminan.

Existen intervalos de tiempo en los cuales el sensor no ha registrado información debido a que el recurso se ha encontrado apagado. Para cada instante de tiempo se crea una tupla donde se indica que el estado del recurso es *apagado*. Estos registros también forman parte del conjunto de entrenamiento.

3.4 Módulo Evaluador

El módulo evaluador es responsable del cálculo de un conjunto de indicadores que permite caracterizar la disponibilidad de un recurso representada por el vector de disponibilidad. Esta información puede no ser suficiente, por ejemplo el vector puede indicar que es probable que el recurso se encuentre disponible durante 120 minutos en las próximas 12 horas. La disponibilidad mencionada puede darse con distintos grados de fragmentación es decir que 120 minutos pueden ser la suma de intervalos de 20 o 30 minutos. Es posible que estos intervalos de disponibilidad se presenten cada 1 hora o más. Por este motivo se crean un conjunto de indicadores que tienen como objetivo proveer información adicional respecto a las características del vector de disponibilidad. Los indicadores que se obtienen son los siguientes: disponibilidad, fragmentación, continuidad, extensión de la disponibilidad, y estabilidad.

La disponibilidad de un recurso representada por la variable *disponibilidad* es la suma de los instantes de tiempo para los cuales el recurso se estima estará disponible según el vector de disponibilidad.

El grado de fragmentación del vector de disponibilidad identificado por la variable *fragmentacion* mide en cuantos intervalos de tiempo se divide la disponibilidad temporal del recurso. Esta medida es una forma de estimar la posible intermitencia en la disponibilidad de un recurso.

El indicador *continuidad* se mide como la relación entre la disponibilidad del recurso y la fragmentación de la disponibilidad según la siguiente expresión:

$$continuidad = \frac{disponibilidad}{fragmentacion}$$

El objetivo de este indicador es proveer una medida que represente el grado de continuidad en la estimación de la disponibilidad temporal.

La variable *extdisp* mide la cantidad de intervalos de tiempo entre el primer y último instante de tiempo en el cual el recurso se estima se encuentre disponible. Esta medida permite calcular el indicador que caracteriza la estabilidad del recurso definido por la siguiente ecuación:

$$estabilidad = \frac{continuidad}{extdisp}$$

Este indicador representa que tan estable se espera se comporte un recurso a través del tiempo.

En la tabla 1 se muestran los indicadores pertenecientes a un conjunto de vectores de disponibilidad para siete recursos diferentes.

Vector de Disponibilidad	V1	V2	V3	V4	V5	V6	V7
Disponibilidad	5	5	5	5	5	5	5
Fragmentación	1	2	2	3	3	4	5
Continuidad	5	2,5	2,5	1,7	1,7	1,3	1,0
Extensión disponibilidad	5	10	6	7	10	10	10
Estabilidad	1,00	0,25	0,42	0,24	0,17	0,13	0,10

Table 1: Ejemplo de indicadores pertenecientes a diferentes de vectores de disponibilidad

De acuerdo al conjunto de vectores de disponibilidad que se muestra en el ejemplo, se espera que el recurso que contiene el vector de disponibilidad V1 sea apropiado para el procesamiento de trabajos críticos. Para el recurso con vector V2 probablemente puedan asignarse tareas breves y de bajo impacto respecto al proceso global. De esta manera, se provee un conjunto de estimadores que permita caracterizar los recursos disponibles en el tiempo. Esta información se envía al módulo de registro que se describe en la subsección 3.6.

3.5 Módulo descriptor del recurso

Este módulo es responsable de adquirir y administrar información descriptiva del recurso. Esta información se considera estática ya que en general no varía con el tiempo. El módulo obtiene en forma automática y sin intervención del usuario datos tales como características técnicas de hardware, sistemas operativos instalados, aplicaciones y servicios disponibles, y configuración de conectividad a la red. Esta información se actualiza cuando varía alguno de sus atributos y se provee al módulo denominado registro que se describe en la siguiente subsección.

3.6 Módulo de Registro

El módulo de registro es responsable de reunir y administrar la información relacionada con el recurso. El objetivo de este módulo es publicar la información reunida por medio

de un sistema de descubrimiento de recursos. De esta manera se espera proveer información adicional que describa a un recurso no sólo por medio de sus características técnicas sino desde su comportamiento esperado.

El módulo de registro exporta la información reunida en formatos estandar compatibles para ser utilizados por otros sistemas incluso por usuarios administradores de red. En este último caso es posible evaluar el rendimiento de los equipos y permite también medir la capacidad de cómputo de distintas áreas de una organización.

4 Experimentos y resultados

El proceso de simulación tiene como objetivo observar y evaluar la eficiencia en el proceso de selección de recursos. Para ello se consideraron dos estrategias de selección de recursos: selección aleatoria y selección basada en información provista por el sistema de predicción.

Con el fin de simplificar la evaluación de ambos criterios de selección se considera que cualquier recurso tiene la capacidad de cómputo suficiente para ejecutar cualquier trabajo. El tiempo de ejecución del trabajo es independiente de la capacidad del recurso y se estima de acuerdo a las unidades de tiempo en las cuales el recurso se encuentra disponible.

En el proceso de simulación se utilizó información proveniente de diferentes computadoras pertenecientes al Instituto Tecnológico Universitario de la Universidad Nacional de Cuyo, Mendoza, Argentina. En cada recurso se instaló un sensor de actividad que permitió registrar cada un intervalo de tiempo de un minuto la actividad del mismo. El sensor registró la fecha, hora, porcentaje de carga de la CPU y porcentaje de memoria utilizada. El proceso de monitoreo se realizó durante 20 días en 15 recursos pertenecientes al laboratorio de computadoras del instituto. La información de monitoreo de cada recurso se integró en una base de datos para ser utilizada en el proceso de simulación.

La información registrada por cada sensor reveló que la actividad de los recursos se extiende desde aproximadamente las 08 hs hasta las 16 hs. Considerando este intervalo de tiempo los experimentos se dividieron en todas etapas. La primera etapa comprende un intervalo de tiempo que se extiende de 08:00 a 12:00 hs y la segunda etapa de 12:01 a 16:00 hs. Esta división de horarios se realizó para facilitar el análisis de los resultados.

Por otra parte, en cada etapa se utilizaron cuatro tipos de configuraciones. En la primera el número de trabajos generados fue equivalente al 30% de los recursos considerados en el proceso de simulación. En la segunda configuración este número representó el 60 % de los recursos, en la tercera y cuarta configuración el 100 % y el 120 % respectivamente. En este último caso, la cantidad de trabajos supera en un 20 % la cantidad de recursos. De esta manera se generaron distintos escenarios para evaluar el sistema.

La simulación incluyó 8.607 trabajos en diferentes iteraciones de acuerdo a la configuración mencionadas. En cada iteración se seleccionó el momento de inicio del trabajo de acuerdo a la etapa que se estaba evaluando. El tiempo de duración se elige en forma aleatoria de un intervalo de 10 a 60 minutos. Con esta información se solicita al sistema de planificación que seleccione dos recursos. Un recurso se determina en forma aleatoria y el otro recurso de acuerdo a la información provista por el sistema.

En ambos casos si el recurso se encuentra encendido acepta el trabajo y su ejecución se lleva a cabo en los momentos en que el mismo se encuentra disponible. La disponibilidad depende de si el usuario se encuentra utilizando el mismo o no. Si el recurso se apaga durante el procesamiento de un trabajo el mismo se reanuda cuando el recurso se encuentre operativo nuevamente.

En ambas estrategias cuando un recurso recibe un trabajo para su ejecución no es considerado en el proceso de selección hasta no finalizar su ejecución. El resultado del proceso de simulación permite evaluar el tiempo total de ejecución de cada trabajo. Esta medida considera desde el momento en que el trabajo se asigna al recurso hasta que se finaliza su ejecución. En la tabla 2 se muestran los resultados obtenidos.

Etapa	Carga	Trabajos	SRR	RDN	Indistinto	RDS	RDN	Indistinto
Etapa 1	30 %	500	243	94	163	49 %	19 %	32 %
	60 %	1000	420	260	320	42 %	26 %	32 %
	100 %	1500	599	407	494	40 %	27 %	33 %
	120 %	1942	661	563	718	34 %	29 %	37 %
Etapa 2	30 %	363	206	55	102	57 %	15 %	28 %
	60 %	734	348	164	222	48 %	22 %	30 %
	100 %	1112	483	276	353	43 %	25 %	32 %
	120 %	1456	549	392	515	38 %	27 %	35 %
Total		8607	3509	2211	2887	41 %	25 %	34 %

Table 2: Resultados obtenidos del proceso de simulación.

La primera columna indica la etapa a la que corresponde los resultados. La segunda columna indica el nivel de carga de trabajos generados respecto al total de recursos. Las columnas RDN y SRR se refieren a los resultados obtenidos para el sistema de selección aleatoria y el sistema de predicción y evaluación respectivamente.

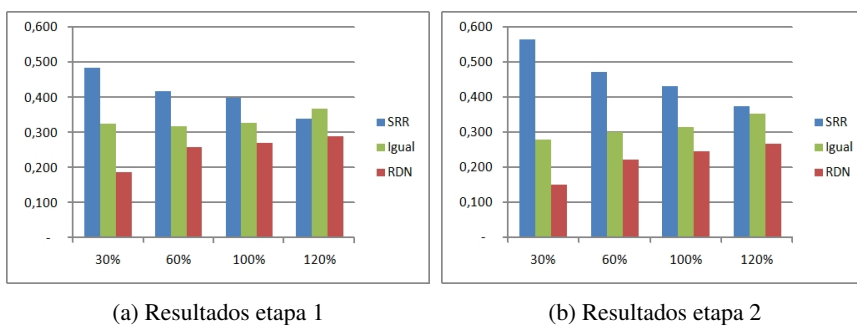


Fig. 2: Resumen de resultados obtenidos en las etapas de simulación 1 y 2.

En ambas columnas se muestra la cantidad de ocasiones en las cuales la elección del recurso optimizó el tiempo total de ejecución. La columna denominada indistinto representa el número de ocasiones en las cuales la selección de recursos de ambas estrategias presentaron el mismo tiempo de ejecución. Finalmente, las últimas tres columnas indican los porcentajes que representan cada medición realizada.

Los resultados muestran que en 8.607 trabajos simulados, el 95 % de la veces el sistema permitió seleccionar un recurso que demandó un tiempo de ejecución menor o igual respecto a al proceso de selección aleatoria. El 57 % de las veces se obtuvieron mejoras con tiempos de ejecución hasta 8 veces menores respecto a la selección aleatoria. La figura 2 muestra una representación gráfica de los resultados en tiempos de ejecución.

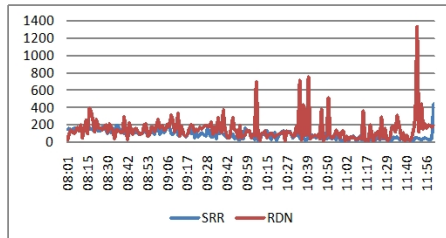
A continuación se muestran gráficos que representan las curvas de los tiempos promedios de ejecución que demandó cada trabajo según el recurso al cual se asignó. Para poder representar gráficamente los resultados se realizó un promedio de los tiempos de ejecución para los instantes de tiempos considerados.

Las figuras 3 y 4 muestran los resultados para la primera y segunda etapa respectivamente. En cada gráfico, el eje de las ordenadas representa el tiempo que demandó la ejecución del trabajo y el eje de las abscisas el instante de tiempo en que comienza a ejecutarse el trabajo. La curva identificada como SRR representa el tiempo promedio de ejecución registrado por recursos sugeridos por el sistema. La curva denominada RDN muestra el tiempo promedio de ejecución registrado cuando los recursos se seleccionan aleatoriamente sin información adicional. En todos los casos se puede observar que las curvas son irregulares esto se debe a que el tiempo de duración de cada trabajo se determina en forma aleatoria.

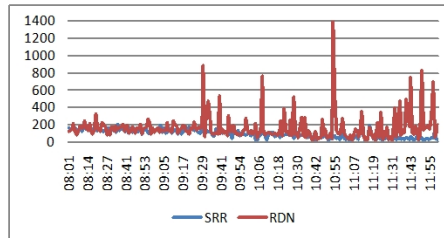
En la figura 3.a y 3.b es posible observar una mayor irregularidad en la curva que representa la selección aleatoria. Esta irregularidad se debe a que existe un menor número de trabajos que recursos disponibles y a la naturaleza aleatoria de la selección de recursos. Por otra parte, la curva que representa los tiempos de ejecución obtenidos utilizando el SRR presenta un menor grado de irregularidad. Esto demuestra que los recursos sugeridos por el SRR lograron un menor tiempo de ejecución respecto al criterio de selección aleatoria.

En la figura 3.c y 3.d se observa una mayor irregularidad en la curva que representa los resultados de utilizar el sistema de predicción. El motivo se debe a que el número de trabajos a distribuir es igual o mayor incluso que el número de recursos disponibles. De esta manera la cantidad de recursos a seleccionar disminuye ya que seguramente se encuentran procesando algún trabajo asignado previamente.

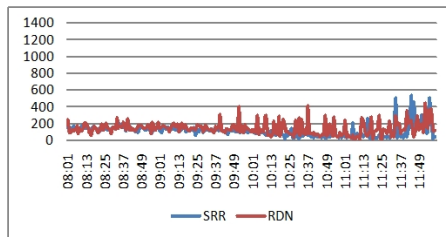
La figura 4 muestra los resultados obtenidos en la segunda etapa durante el intervalo de tiempo que se extiende de 12:01 a 16:00 hs. En todos los casos se observa un mayor nivel de irregularidad en los tiempos de ejecución. Esto se debe porque en esa franja horario el usuario realizó un uso intensivo de los recursos. De todas formas es posible observar que en general la curva que representa los tiempos de ejecución obtenidos para el sistema se mantiene por debajo del resultado para el criterio de selección aleatoria.



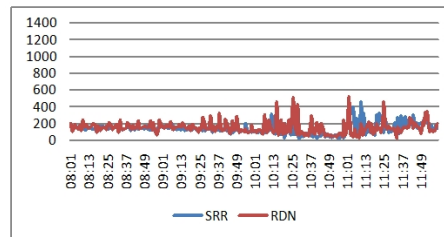
(a) Recursos requeridos 30 % del total exist.



(b) Recursos requeridos 60 % del total exist.

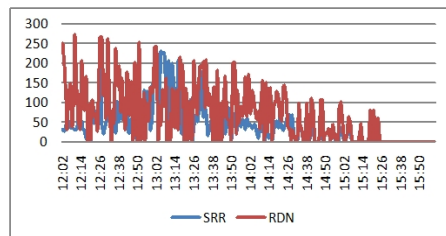


(c) Recursos requeridos 60 % del total exist.

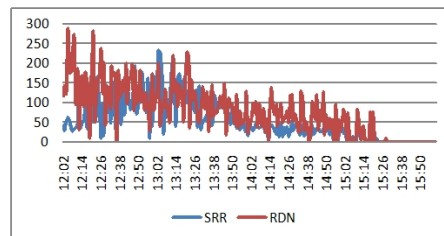


(d) Recursos requeridos 60 % del total exist.

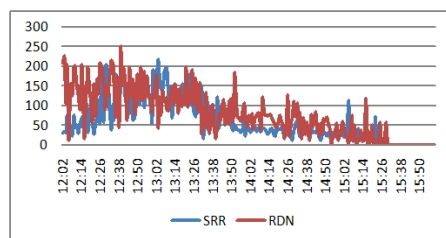
Fig. 3: Tiempo promedio de ejecución registrados en la etapa 1.



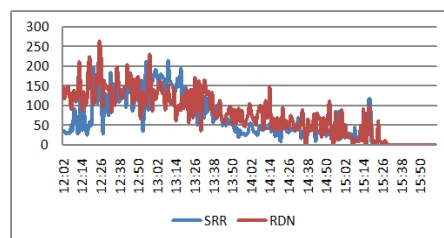
(a) Recursos requeridos 30 % del total exist



(b) Recursos requeridos 60 % del total exist



(c) Recursos requeridos 100 % del total exist



(d) Recursos requeridos 120 % del total exist

Fig. 4: Tiempo promedio de ejecución registrados en la etapa 2.

5 Conclusiones y trabajos futuros

En el presente trabajo se presentó un modelo para asistir al sistema de planificación en la selección de recursos. El sistema realiza una predicción del estado de cada recurso utilizando algoritmos de clasificación del área de inteligencia artificial. Para permitir la selección de recursos se provee un conjunto de indicadores que permiten evaluar la disponibilidad de los recursos. El sistema reúne esta información junto con las características de hardware y software del recurso. De esta manera el sistema de planificación puede organizar la distribución de trabajos.

El rendimiento del planificador utilizando selección aleatoria de recursos y selección utilizando información del comportamiento del recurso se analizó en distintos escenarios. En algunos casos el número de trabajos era inferior a la cantidad de recursos disponibles y en otros casos este número era superior. Los resultados muestran que en 8.607 trabajos simulados el 95 % de la veces el sistema permitió seleccionar un recurso que demandó tiempo de ejecución menor o igual respecto a la selección aleatoria. El 57 % de las veces se obtuvieron mejoras con tiempos de ejecución hasta 8 veces menores respecto a la selección aleatoria.

Se espera en trabajos futuros aumentar el número de recursos participantes y analizar el comportamiento de los mismos. Por otra parte, es posible realizar un estudio sobre el impacto en el rendimiento del sistema si los trabajos pueden migrar de un recurso a otro. También se espera integrar el sistema de predicción y evaluación a un sistema de descubrimiento de recursos.

References

1. Wolski, R., Spring, N.T., Hayes, J.: The network weather service: a distributed resource performance forecasting service for metacomputing. *Future Generation Computer Systems* **15**(5-6) (1999) 757 – 768
2. Wolski, R.: Experiences with predicting resource performance on-line in computational grid settings. *SIGMETRICS Perform. Eval. Rev.* **30** (March 2003) 41–49
3. Dinda, P.: A prediction-based real-time scheduling advisor. In: *Parallel and Distributed Processing Symposium., Proceedings International, IPDPS 2002, Abstracts and CD-ROM.* (2002) 10 –17
4. Kang, W., Grimshaw, A.: Failure prediction in computational grids. In: *Simulation Symposium, 2007. ANSS '07. 40th Annual.* (march 2007) 275 –282
5. Dobber, M., Koole, G., van der Mei, R.: Dynamic load balancing experiments in a grid. In: *Proceedings of the Fifth IEEE International Symposium on Cluster Computing and the Grid (CCGrid'05) - Volume 2 - Volume 02.* CCGRID '05, Washington, DC, USA, IEEE Computer Society (2005) 1063–1070
6. Chun, B., Culler, D., Roscoe, T., Bavier, A., Peterson, L., Wawrzoniak, M., Bowman, M.: Planetlab: an overlay testbed for broad-coverage services. *SIGCOMM Comput. Commun. Rev.* **33** (July 2003) 3–12
7. Ramachandran, K., Lutfiyya, H., Perry, M.: Decentralized approach to resource availability prediction using group availability in a p2p desktop grid. *Future Generation Computer Systems* (2010)
8. Salzberg, S.L.: C4.5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993. *Machine Learning* **16** (1994) 235–240 10.1007/BF00993309.