

Investigación: Desafíos en Sistemas Computacionales Heterogéneos

Malcolm Davis, Miembro Estudiantil, IEEE

mdavis.cr@ieee.org

Resumen—Los clusters con nodos heterogéneos se han vuelto populares en la comunidad científica y en otras áreas ya que se puede aumentar el rendimiento y disminuir el consumo energético en la resolución de un problema. Este documento contiene un resumen de lo investigado acerca de los sistemas heterogéneos, tanto a nivel macro como a niveles micro así como los pros y contras de utilizar los mismos y los desafíos que se enfrentan.

Palabras clave: Sistemas Heterogéneos, Eficiencia Energética, Computación de Alto Rendimiento, Paralelismo, CMP, GPUs, CPUs, FPGA.

I. INTRODUCCIÓN

Durante años, los sistemas homogéneos lideraron el mercado de la computación; siendo un sistema homogéneo un sistema con nodos, o núcleos con las mismas características y que teóricamente tienen la misma respuesta en términos de tiempo y consumo de energía ante un problema. Pero, últimamente, los sistemas heterogéneos han tomado fuerza; estos sistemas al contrario de los homogéneos son sistemas que se usan para aprovechar las diferencias entre los núcleos o nodos para aumentar el rendimiento y disminuir el consumo energético [9].

Existen diferentes tipos de heterogeneidad, entre los mencionados por Zahnran [9] se encuentra el tipo que se da

en un mismo procesador debido a los DVFS de los núcleos modificando su rendimiento según la temperatura, el segundo está relacionado a los procesadores que empaquetan diferentes tipos de núcleos en sí teniendo diferentes rendimientos, y el último tipo es el que se da al utilizar diferentes tipos de unidades de procesamiento como combinaciones de CPUs, GPUS, FPGAS y otros.

Existen otros tipos de sistemas Heterogéneos como el que se estudia en [4] que su heterogeneidad se presenta por la diferencia de tiempo de acceso a los datos; o los mencionados en [8] que tienen diferentes cantidades de almacenamiento principal y/o secundario. Un ejemplo de un sistema heterogéneo se puede observar en la figura 1.

En general los sistemas heterogéneos se pueden volver muy complejos [8], y se presentan desafíos a la hora de querer aprovechar estos recursos de la mejor manera. Esta investigación se enfocará en entender las ventajas y desventajas de diferentes sistemas heterogéneos, los desafíos que se presentan tanto en a nivel de hardware como de software y algunas soluciones que proponen diferentes investigadores y operadores de infraestructuras de servicios en la nube.

II. ESTADO DEL ARTE

II-A. Hardware

Como se mencionó anteriormente, las diferencias en las características específicas en cada nodo o en cada unidad

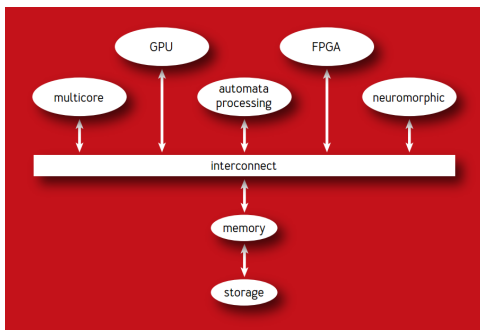


Figura 1. Ejemplo Sistema Heterogéneo[9]

de procesamiento de ese nodo causan heterogeneidad en un sistema; para las grandes compañías que tienen clusters de gran escala, la heterogeneidad se da en sus sistemas debido a, entre otros factores, las diferentes capas del sistema que se compran a lo largo del tiempo, si bien es cierto que estas compañías invierten en tecnologías específicas para suplir la necesidad de un mercado [8] i.e Google con sus TPUs, la mayoría de la heterogeneidad en este tipo de sistema se da porque con los años cambian las arquitecturas de segmentos del sistema al existir mejor tecnologías en términos de costo-beneficio.

Otro elemento importante a la hora de diseñar sistemas computacionales heterogéneos es la **memoria**, tanto principal como secundaria, pero uno de los desafíos importantes que se han encontrado es el del diseño de la jerarquía de memoria para un sistema heterogéneo; la cual afecta directamente la respuesta del sistema, y un sistema con diferentes módulos de memoria ya sea de diferente capacidad o frecuencia. Se han realizado varios esfuerzos para intentar resolver estos desafíos [4], [1]. Un ejemplo claro de las mejoras que se pueden hacer, es lo expuesto en el paper de Flores et al [4] donde al modificar las características físicas de las inter-conexiones y modificar la forma en que se accedía a los datos y a las direcciones, lograron mejorar el rendimiento hasta un 10 % y disminuyeron el consumo energético un 38 % a cambio de aumentar la heterogeneidad del sistema. Otro ejemplo de lo que se puede hacer para mejorar la respuesta de un sistema, es el propuesto por [1] que es un esquema de distribución de datos que

distribuye la información en los servidores de almacenamiento secundario considerando carga de la aplicación y el comportamiento de los servidores, dividiendo los archivos en regiones dependiendo de la carga de entrada y salida obteniendo una mejora de casi un factor de 27 para lecturas y un factor cercano a 13 para escritura.

Otro factor importante en un sistema computacional heterogéneo es el **tipo de unidad de procesamiento** utilizado [6], dependiendo del tipo de problema a resolver, y tomando en cuenta que normalmente los sistemas se crean para la resolución general de problemas y el tipo de problema puede cambiar muy rápido, es importante tener diferentes unidades computacionales para poder asignar a cada problema y aprovechar mejor los recursos, la asignación de estos recursos se hablará más adelante en este documento así como los problemas asociados a la programación de la solución del problema. Por ahora, cabe destacar algunas unidades de procesamiento que se usan en el mercado y sus ventajas. Los procesadores comunes, ya sean con un núcleo único o multi-núcleo son los que se han utilizado por años para el procesamiento de datos, son buenos para realizar varias operaciones sobre los datos, pueden tener una extensión de su ISA o arquitectura del set de instrucciones para poder manejar operaciones vectoriales y así comportarse similar a los procesadores vectoriales, que aplican operaciones sobre registros vectoriales en vez de registros escalares como los anteriores. Los procesadores gráficos aplican pocas instrucciones a una cantidad grande de datos y las FPGAs se utilizan para replicar mejor el comportamiento de los circuitos sin tener que implementarlos ni simularlos por software, esto quiere decir que son más económicos que implementar el circuito y más rápidos que simularlo.

Hay muchos factores que pueden influenciar en el **consumo energético** de un sistema, entre los cuales están los sistemas de DVS(Dynamic Voltage Scaling) de los procesadores, que modifican la tensión de cada núcleo modificando su consumo energético, pero esto también modifica la frecuencia del procesador lo cual modifica su desempeño. Otro factor que

está relacionado con el consumo energético per se del sistema es la disipación de calor de sus elementos, se han realizado varios estudios con respecto a esto [2], [4], [5], [7]. Una de las soluciones propuestas por Kumar et al [5], es un sistema de asignación de recursos "consciente" de la heterogeneidad que asigna los recursos tomando en cuenta varios factores para optimizar tanto el rendimiento como el consumo energético, con esta solución lograron hasta una mejora del consumo energético de 50 %.

II-B. Software

En el apartado anterior se habló de como puede el hardware contribuir a la heterogeneidad de un sistema, pero el software también puede afectar el rendimiento de un nodo lo cuál agrega heterogeneidad al mismo [7]. Si los nodos tienen diferentes sistemas operativos, o versiones de los mismos, diferentes bibliotecas, y en general diferente configuración, el comportamiento de los mismos es diferente, y esto unido a los aportes de hardware es lo que le da la particularidad a cada nodo y son factores a tomar en cuenta a la hora de asignar recursos si se quiere aprovechar las fortalezas del sistema. Y entre más heterogeneidad más complicada se vuelve la asignación, sin tomar en cuenta que el dinamismo de los problemas a resolver también influye.

El problema yace en que la **Asignación de Recursos** de recursos actual no toma en cuenta todos estos aspectos, se enfocan en reservar los recursos a partir de una parte de estos parámetros, ya que es muy complejo hacer el análisis de todos y a la vez asignar los recursos a los trabajos a una velocidad aceptable. Depende del diseñador del sistema priorizar los parámetros que son más importantes para los objetivos por el cuál el sistema se crea y así elegir cuál algoritmo de asignación de recursos es el más apropiado para el problema. Para resolver esto, varias investigaciones se han realizado [5], [7], [6]. Al implementar el sistema de asignación de recursos similar al del apartado anterior, pero propuesto por Zhang et al [7] se logra obtener una mejora considerable del rendimiento del sistema, esto por utilizar los recursos de una mejor manera y reasignar

los mismos en caso de ser necesario.

Por último, como se puede entender por lo anterior, la heterogeneidad de un sistema lo hace complejo, y los paradigmas de programación paralela actuales no toman en cuenta estos factores, por eso algunos proponen mejorar algunos **lenguajes** para facilitar la tarea al usuario así como madurar los **compiladores** para que también aporten al proceso, el problema está en que al cambiar estas herramientas y hacerlas específicas para un sistema en específico se pierde generalidad.

III. CONCLUSIONES

Los sistemas heterogéneos son herramientas importantes en el procesamiento de datos tanto para la investigación científica como para el uso cotidiano, y en el futuro cercano serán la norma al diseñar cualquier sistema; pero para poder realmente aprovechar los recursos de estos sistemas y hacerlo de la mejor manera, se debe mejorar la forma en que se asignan recursos para la ejecución de procesos ya que es uno de los principales desafíos del campo.

Dependiendo del enfoque del sistema y los objetivos del proyecto, se puede priorizar la forma de asignación de recursos para mejorar el consumo energético.

Además, existe un área bastante grande de investigación en cuanto a los algoritmos de asignación de recursos, modificaciones posibles a los lenguajes y mejoras a los compiladores.

IV. ANEXOS

REFERENCIAS

- 1 S. He, Y. Wang, X. Sun and C. Xu, "HARL: Optimizing Parallel File Systems with Heterogeneity-Aware Region-Level Data Layout". IEEE Transactions on Computers, vol. 66, no. 6, pp. 1048-1060, 2017.
- 2 S. Pagani, A. Pathania, M. Shafique, J. Chen and J. Henkel, "Energy Efficiency for Clustered Heterogeneous Multicores". IEEE Transactions on Parallel and Distributed Systems, vol. 28, no. 5, pp. 1315-1330, 2017.
- 3 S. Pagani, H. Khdr, J. Chen, M. Shafique, M. Li and J. Henkel, "Thermal Safe Power (TSP): Efficient Power Budgeting for Heterogeneous Manycore Systems in Dark Silicon". IEEE Transactions on Computers, vol. 66, no. 1, pp. 147-162, 2017.
- 4 A. Flores, M. Acacio and J. Aragón, "Address Compression and Heterogeneous Interconnects for Energy-Efficient High-Performance in Tiled CMPs". 2008 37th International Conference on Parallel Processing, 2008.

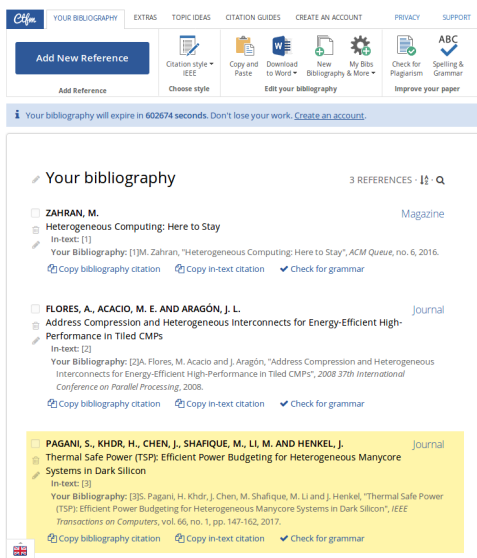


Figura 2. Gestor Bibliográfico Cite this For Me

- 5 G. Xie, G. Zeng, X. Xiao, R. Li and K. Li, ".*Energy-Efficient Scheduling Algorithms for Real-Time Parallel Applications on Heterogeneous Distributed Embedded Systems*". IEEE Transactions on Parallel and Distributed Systems, vol. 28, no. 12, pp. 3426-3442, 2017.2008.
- 6 F. Bower, D. Sorin and L. Cox, "*The Impact of Dynamically Heterogeneous Multicore Processors on Thread Scheduling*". IEEE Micro, vol. 28, no. 3, pp. 17-25, 2008.
- 7 Q. Zhang, M. Zhani, R. Boutaba and J. Hellerstein, "*Dynamic Heterogeneity-Aware Resource Provisioning in the Cloud*". IEEE Transactions on Cloud Computing, vol. 2, no. 1, pp. 14-28, 2014.
- 8 C. Reiss, A. Tumanov, G. Ganger, R. Katz and M. Kozuch, "*Heterogeneity and dynamicity of clouds at scale*". Proceedings of the Third ACM Symposium on Cloud Computing - SoCC '12, 2012.
- 9 M. Zahran, "Heterogeneous Computing: Here to Stay", *ACM Queue*, no. 6, 2016.