

Suporte à Execução Eficiente de Aplicações em Plataformas com Paralelismo Multi-Nível

Vinícius Garcia Pinto

Lucas Mello Schnorr – Nicolas Maillard

Grupo de Processamento Paralelo e Distribuído (GPPD)
Instituto de Informática - UFRGS

24 de Abril de 2014

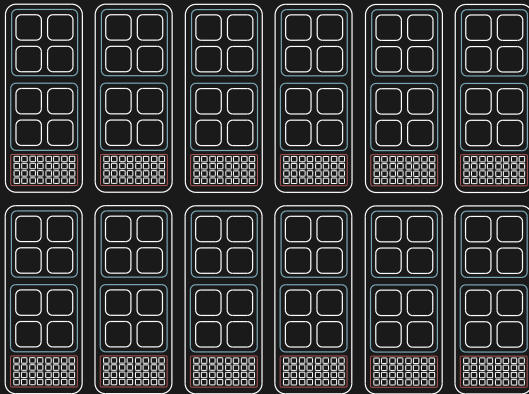


Fonte: PopSci 2014

Tianhe-2

- #1 no TOP500 (Jun/2013, Nov/2013, Jun/2014 e Nov/2014);
- 16000 nós
 - 2x Intel E5-2692v2 (12 cores físicos);
 - 64GB RAM;
 - 3x Intel Xeon Phi 31S1P (171 cores físicos);
 - 24GB (3x8) RAM;
- 1.34 PB e ~ 3000000 cores;
- 33.8×10^{15} FLOPS;

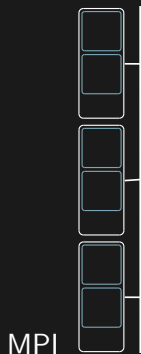
Considerações Iniciais



- Como explorar de forma eficiente este hardware?
 - processos, threads, locks, transferências de memória/mensagens;
- Exascale?
 - 10^{18} FLOPS;

Suporte à Execução Eficiente em Plataformas com Paralelismo Multi-Nível

- Solução usual:
 - APIs específicas para cada camada de paralelismo do hardware da arquitetura;



- Solução usual:
 - APIs específicas para cada camada de paralelismo do hardware da arquitetura;
- Pode funcionar, porém:
 - aplicações menos legíveis e pouco portáteis;
 - esforço de programação considerável;
 - difícil de portar em casos de necessidade de alterações parciais no *hardware* e/ou no *software*.

Suporte à Execução Eficiente em Plataformas com Paralelismo Multi-Nível

- Mecanismo de Abstração:
 - API + Ambiente de Execução;
 - Algumas ferramentas focam em sistemas com memória distribuída;
 - Outras em explorar o paralelismo de nós multiprocessados auxiliados por aceleradores;



- WORMS [Pinto et al. 2012] e XKaapi [Gautier et al. 2013]
 - suporte a sistemas com CPUs multicore e GPUs;
 - balanceamento de carga por roubo de tarefas;
 - XKaapi oferece também:
 - abstração entre os espaços de memória;
 - dependências entre tarefas por fluxo de dados;
 - sobreposição de transferências de memória e computação.

- StarPU [Augonnet et al. 2010]
 - suporte a sistemas com CPUs multicore, GPUs e Cell BE;
 - escalonamento de tarefas com base em modelos desempenho de execuções prévias;
 - tempo de execução e transferências de memória;

- Charm++ [Vasudevan et al. 2013]
 - suporte a sistemas com memória compartilhada e memória distribuída;
 - balanceamento dinâmico de carga;
 - G-Charm
 - suporte a GPU;
 - sobreposição entre transferências de dados e computação para aumentar a taxa de ocupação dos recursos;

- Objetivo:
 - "explorar e validar técnicas que permitam a execução eficiente de aplicações paralelas em plataformas paralelas atuais";
- Próximas Etapas:
 - identificar, projetar e avaliar soluções para:
 - escalonamento e balanceamento de carga;
 - abstração sobre diferentes níveis de memória;
 - atribuição de trabalho em arquiteturas híbridas;

Obrigado!



Vinícius Garcia Pinto
vinicius.pinto@inf.ufrgs.br



-  Asanovic, Krste et al. (2009). “A view of the parallel computing landscape”. Em: *Communications of the ACM* 52.10, pp. 56–67. ISSN: 0001-0782. DOI: <http://doi.acm.org/10.1145/1562764.1562783>.
-  Augonnet, Cédric, Samuel Thibault e Raymond Namyst (2010). *StarPU: a Runtime System for Scheduling Tasks over Accelerator-Based Multicore Machines*. Research Report RR-7240, p. 33. URL: <https://hal.inria.fr/inria-00467677>.
-  Gautier, Thierry et al. (2013). “XKaapi: A Runtime System for Data-Flow Task Programming on Heterogeneous Architectures”. Em: *Proc. 27th Int. Symp. Parallel and Distributed Process. (IPDPS)*. pp. 1299-1308: IEEE Comput. Soc. ISBN: 978-0-7695-4971-2. DOI: 10.1109/IPDPS.2013.66. URL: <http://dx.doi.org/10.1109/IPDPS.2013.66>.
-  Pinto, Vinicius Garcia e Nicolas Maillard (2012). “Work Stealing on Hybrid Architectures”. Em: *13th Symp. on Computer Systems (WSCAD-SSC)*. pp. 17-24, Los Alamitos: IEEE Computer Soc. DOI: 10.1109/WSCAD-SSC.2012.28.

-  PopSci (2014). *Popular Science - New Chinese Supercomputer Goes Beta (Tianhe 2 Ahead Of Schedule And On Budget)*. Disponível em <http://www.popsci.com/blog-network/eastern-arsenal/new-chinese-supercomputer-goes-beta-tianhe-2-ahead-schedule-and-budget>. Acesso em Abril de 2015.
-  TOP500 (2015). *TOP500 Supercomputer Sites*. Disponível em <http://www.top500.org/>. Acesso em Abril de 2015.
-  Vasudevan, R., Sathish S. Vadhiyar e Laxmikant V. Kalé (2013). “G-Charm: An Adaptive Runtime System for Message-driven Parallel Applications on Hybrid Systems”. Em: *Proc. 27th Int. Conf. on Supercomputing*. ICS '13. Eugene, Oregon, USA: ACM, pp. 349–358. ISBN: 978-1-4503-2130-3. DOI: 10.1145/2464996.2465444. URL: <http://doi.acm.org/10.1145/2464996.2465444>.