

Análise de desempenho e consumo de uma plataforma ARM big.LITTLE com uso da biblioteca StarPU

Bruno M. Muenchen¹, João V. F. Lima¹

¹ Universidade Federal de Santa Maria (UFSM) - Brasil

Resumo. Este trabalho apresenta uma análise do desempenho e do consumo energético de uma plataforma ARM big.LITTLE usando como benchmark um algoritmo SPMV desenvolvido com uso da biblioteca StarPU, visando estudar tempo de execução e consumo utilizando diferentes números de threads e diferentes cores do dispositivo. Os resultados apontam que tanto tempo de execução quanto consumo são menores quando usamos os cores de alto desempenho.

1. Introdução

Um dos principais objetivos na atualidade da computação de alto desempenho é economia de energia. Com um limite estabelecido em 20MW para construção de um sistema *Exascale* não faltam estudos nessa área. Estes estudos não concentram-se apenas na criação de novas tecnologias de hardware, mas também de novas técnicas de desenvolvimento de software. A chegada de sistemas heterogêneos ao mercado trouxe consigo novas tecnologias de desenvolvimento, como CUDA e OpenCL, por exemplo. Essas tecnologias acabam aumentando o custo de desenvolvimento e manutenção, visto sua complexidade, além de reduzirem a portabilidade destes sistemas. Com base nisso, este trabalho visa estudar arquiteturas ARM que utilizam a tecnologia big.LITTLE [Jeff 2012], seu desempenho e consumo energético em diferentes situações, utilizando um algoritmo *Sparse Matrix-Vector Multiplication* (SPMV) criado com a biblioteca StarPU [Augonnet et al. 2009], que facilita a programação paralela baseada em tarefas e permite uma fácil adaptação do código já existente para diversas plataformas.

2. Materiais e métodos

Os testes foram executados em uma Odroid XU3, que usa a tecnologia big.LITTLE, combinando o desempenho dos processadores Cortex A-15 com a eficiência energética dos Cortex A-7. O dispositivo em questão utiliza o processador Samsung Exynos5, que conta com 4 núcleos rodando a 2GHz (A-15) e outros 4 núcleos rodando a 1.3GHz (A-7). Possui um total de 2GB LPDDR3 RAM e uma GPU Mali-T628 MP6, com suporte a OpenCL 1.1. A configuração de consumo de energia no momento dos testes foi definida para obter maior desempenho.

O algoritmo utilizado para os testes foi a implementação do SPMV, disponível junto com a biblioteca StarPU, a qual usamos a versão 1.2.0. StarPU é uma biblioteca para escalonamento de tarefas em arquiteturas *multi-core* heterogêneas [Augonnet et al. 2009]. O algoritmo original foi modificado para contabilizar o tempo desde o início até o final da execução. Os dados dos testes foram coletados após 30 execuções do SPMV com diferentes configurações, sendo elas: 1, 2, 4, 6 e 8 CPUs, 2 GPUs e 6 CPUs + 2 GPUs. A política de escalonamento utilizada pelo StarPU foi a *deque model data aware* (DMDA). Esta política escalona as tarefas onde o tempo de execução delas será mínimo usando modelos de desempenho de execução obtidos previamente através da calibragem do StarPU.

3. Resultados

Nas Figuras 1 e 2 pode-se observar o tempo e consumo obtidos com as diferentes execuções do algoritmo. Percebe-se um aumento quando utilizadas mais de 4 *threads*, o que é justificado pelo fato de os *cores* A-7, mais lentos, começarem a ser utilizados, aumentando o tempo e consumo.

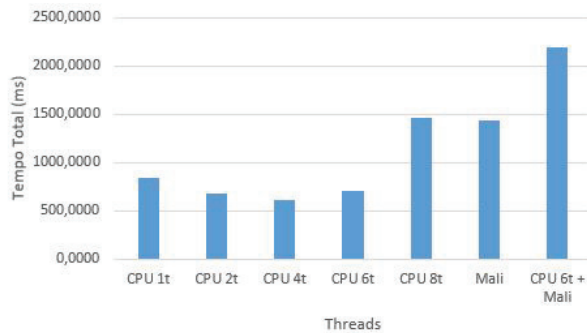


Figura 1. Tempo de execução (s)

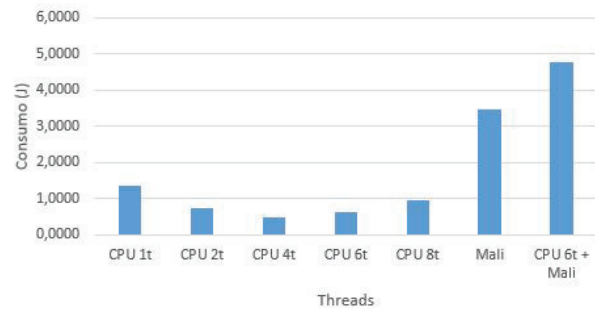


Figura 2. Consumo (J)

A execução com utilização dos 8 *cores* e da GPU Mali teve um tempo muito elevado e similar. Utilizar os 8 *cores* implica em usar 4 *cores* mais lentos, além de fazer concorrência com as *threads* de escalonamento de tarefas do StarPU. A menor frequência da GPU justifica o aumento do tempo e consumo, e as *threads* usadas pelo StarPU para escalonar e controlar as tarefas da GPU, que rodam na CPU, também colaboram com isso. A execução utilizando CPUs e GPUs foi ainda mais lenta que as anteriores, apresentando um tempo quase igual a soma dos tempos das execuções com 6 *cores* e Mali.

4. Conclusão

Este artigo apresentou uma comparação de desempenho de um dispositivo ARM com a tecnologia big.LITTLE usando como *benchmark* um algoritmo SPMV. Foi possível observar um aumento de consumo e de tempo de execução quando os *cores* mais lentos começaram a ser utilizados, bem como um aumento ainda mais expressivo com a utilização da GPU Mali. Além disso, ficou evidente a contribuição do StarPU no desenvolvimento de aplicações que suportem arquiteturas heterogêneas, sendo que o mesmo código foi executado tanto na CPU quanto na GPU sem a necessidade de compilação para uma arquitetura específica. Como trabalhos futuros, pretende-se estudar mais características da GPU Mali, utilizando outros *benchmarks*, com intuito de complementar as informações aqui apresentadas.

Referências

- Augonnet, C., Thibault, S., e Namyst, R. (2009). StarPU: a runtime system for scheduling tasks over accelerator-based multicore machines. *Euro-Par 2009 Parallel Processing*, 5704:863–874.
- Jeff, B. (2012). Big. little system architecture from arm: saving power through heterogeneous multiprocessing and task context migration. In *Proceedings of the 49th Annual Design Automation Conference*, pages 1143–1146. ACM.