Evento	Salão UFRGS 2022: SIC - XXXIV SALÃO DE INICIAÇÃO CIENTÍFICA DA UFRGS
Ano	2022
Local	Campus Centro - UFRGS
Título	Aumentando a eficiência de algoritmos de grafos em HPC
Autor	MARCELO KOJI MOORI
Orientador	ANTONIO CARLOS SCHNEIDER BECK FILHO

Improving the Efficiency of Graph Algorithm Executions on HPC

The growing need for extracting information from large graphs has been pushing the development of parallel graph algorithms. However, the highly irregular structure of the real-world graphs limits the performance and energy improvements of graph applications. In this paper, we show that, in most cases, using all the available cores of the multiprocessor is not always the best option in terms of the aforementioned non-functional requirements. Based on that, we propose GraphKat, a framework that enables the simultaneous processing of several algorithms/graphs instead of executing them serially (i.e., one after another), increasing efficiency in terms of performance and energy. GraphKat works in two steps: (i) it characterizes the graph applications with a specific number of threads based on their efficiency levels; and (ii) it defines the execution order of all graph applications in the target system. We evaluate the GraphKat framework considering the algorithms available in the GAP Benchmark Suite (GAPBS). TheGAPBS provides a high-performance implementation of a representative set of graph processing algorithms, being a representative baseline of the state-of-art. We also evaluate 10 input graphs extracted from SNAP13 (see Table 1 – Amazon, California, Google, Pennsylvania, Youtube, Berkley, Texas, Wikitalk, Orkut, and cit-Patent). They are a representative set of real-world input graphs covering the two broad classes of topologies (meshes and social networks) with different characteristics and sizes (varying from ∼300 thousand to ∼4 million nodes). Experimental results on three multicore processors (Intel and AMD) show that GraphKat improves the overall system's efficiency related to performance (up to 434.26×) and energy-saving (up to 245.21×), and reduces the graph applications' execution time (up to 17.70×) and energy consumption (up to 6.64×) compared to the default execution of parallel applications on HPC systems.