Software Support for Regular & Irregular Application in Parallel Computing

Da Li, Michela Becchi

University of Missouri - Columbia

Abstract

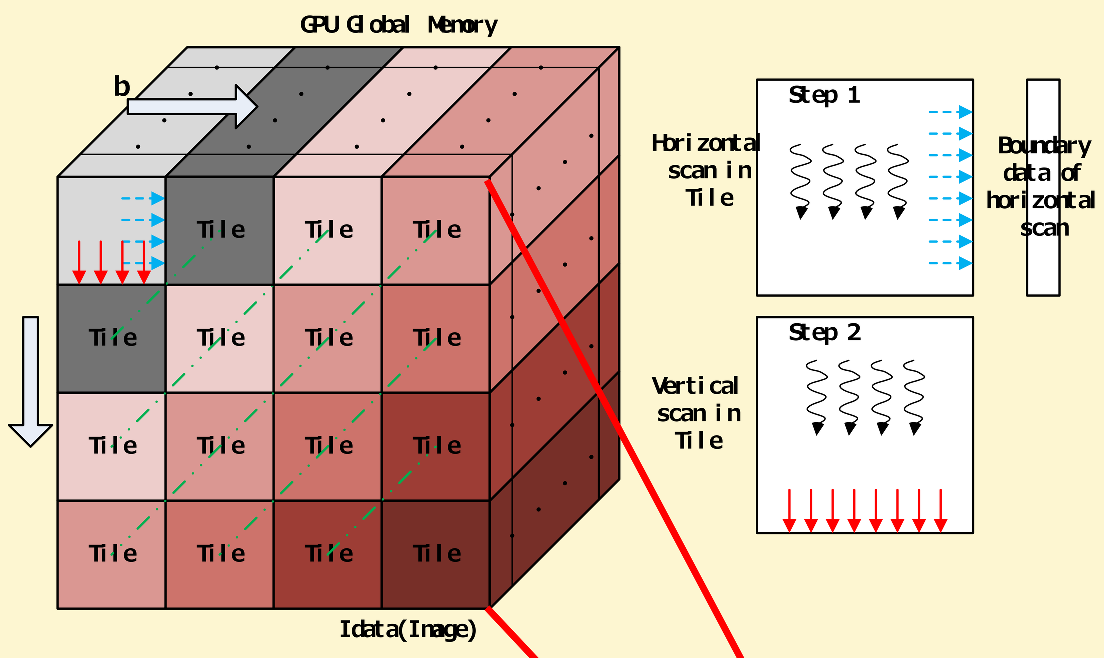
Today's applications can be divided into two basic categories: regular and irregular applications. The parallel computing community has acquired a deep understanding of the characteristics of regular applications and has proposed numerous abstractions, technologies and solutions to facilitate their parallelization. However, the research on irregular applications is still at an initial phase and the availability of programming models and tools targeting these applications is still limited.

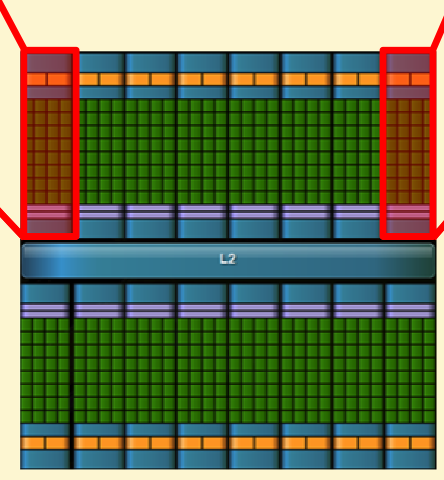
My research focuses on programming models and system software support for regular and irregular algorithms on parallel architecture. More specifically, it can be summarized in the following aspects:

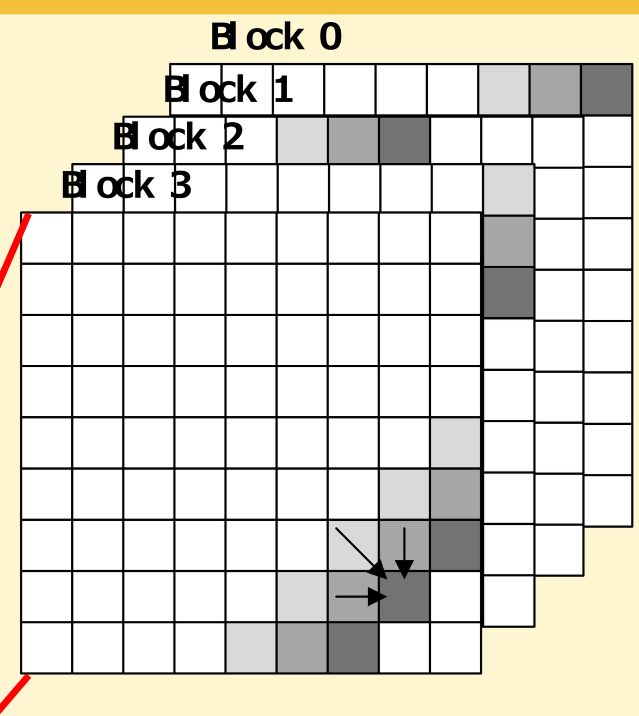
a) Parallelization of regular and irregular applications on GPU

b) Programming model and runtime design for regular and irregular applications

Application-specific Acceleration







Integral Histogram

Needleman-Wunsch

Regular & Irregular applications

- Control structures
- Data structures
- Communication patterns

Software Innovation

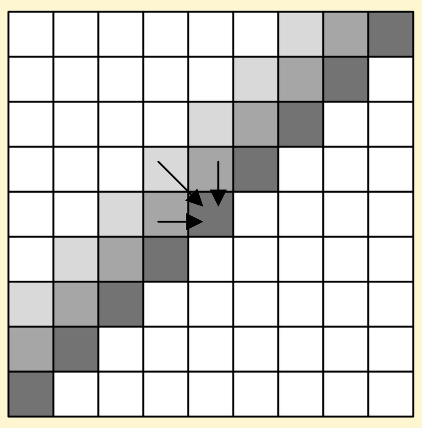
- Programming model
- Compiler
- Runtime

Equation-based Programming Model & Runtime for Matrix-based Regular Applications

Applications

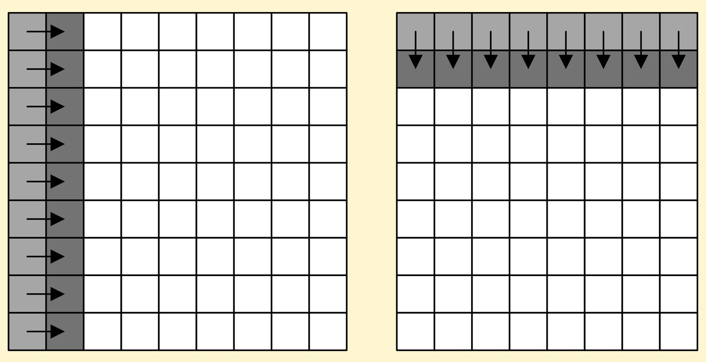
Needleman-Wunsch

$$M(i,j) = \max \begin{cases} M(i-1, j-1) + S(x_i, y_j) \\ M(i-1, j) + G \\ M(i, j-1) + G \end{cases}$$

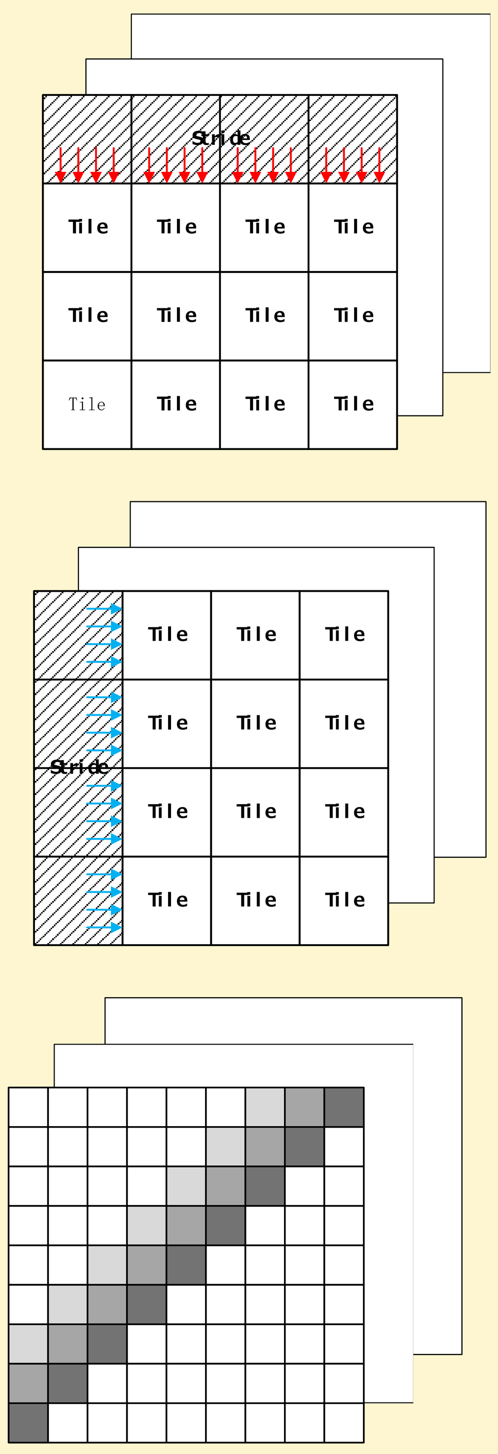


Integral Histogram

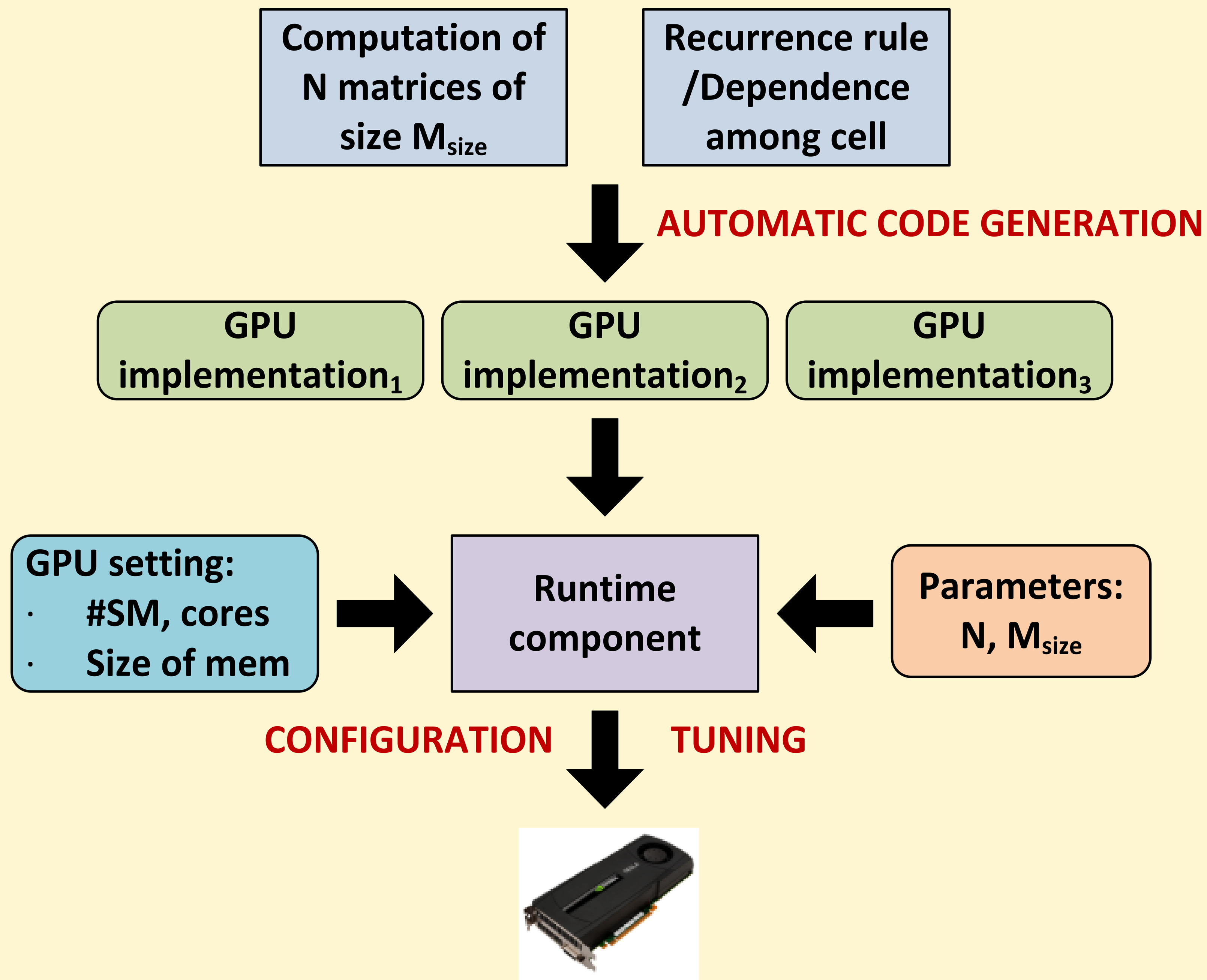
$$H(x,y,b) = \sum_{i=0}^{x} \sum_{j=0}^{y} Q((i,j),b)$$



Parallelization Pattern



Programming Model and Runtime



Static parallelism: A single mapping decision for each execution

Adaptive Runtime for Graph Algorithms

Motivation

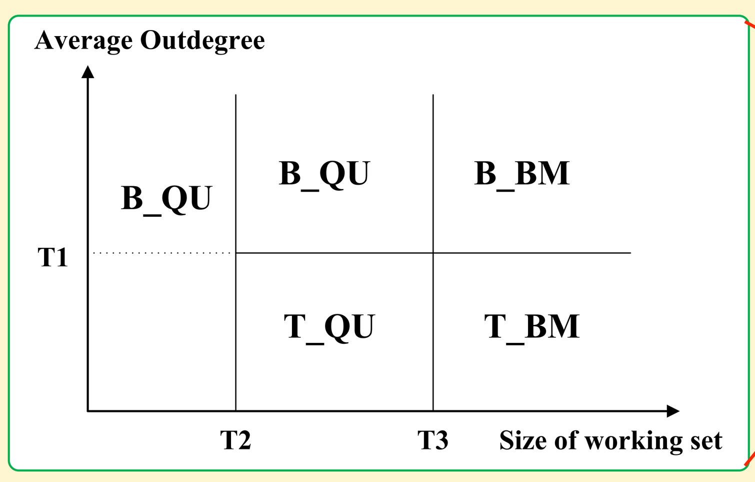
Application

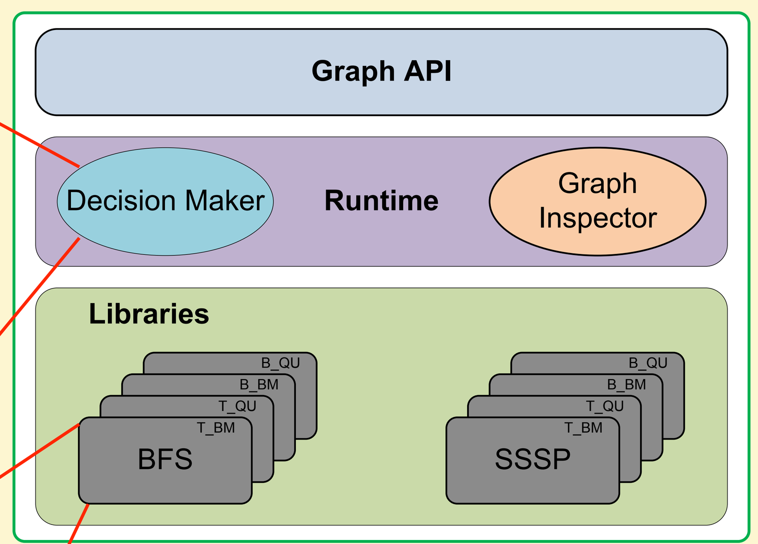
- BFS (breadth-first search)
- SSSP (single source shortest path)

Dataset (real-world graphs)

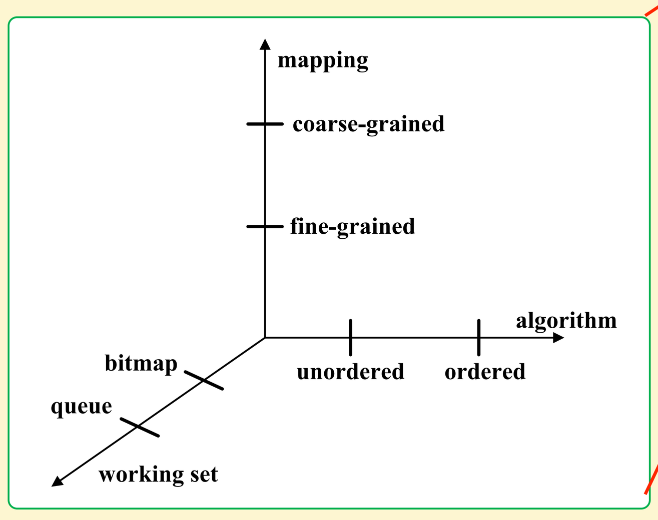
Network	#Nodes	#Edges	Node Outdegree		
			min	max	avg
CO-road	435,666	~1M	1	8	2.4
CiteSear	434,102	~16M	1	1,188	73.9
P2p	36,692	~0.18M	0	1,383	10.0
Amazon	396,803	~1.7M	0	10	8.4
Google	739,454	~2.5M	0	456	6.9
SNS	4,308,452	~34.5M	0	20,293	16.0

Decision Space





Implementation Space



- Mapping decisions based on static and dynamic information
- Adaptive mapping at runtime
- Overhead's reduction by decreasing sampling rate
- ***Dynamic* parallelism: Multiple mapping decisions for each execution**

- On GPU, unordered algorithm is usually better than ordered one
- T: fine-grained (thread-based) mapping
- B: coarse-grained (block-based) mapping
- QU: queue
- BM: bitmap

Contact us:
dlx7f@mail.missouri.edu