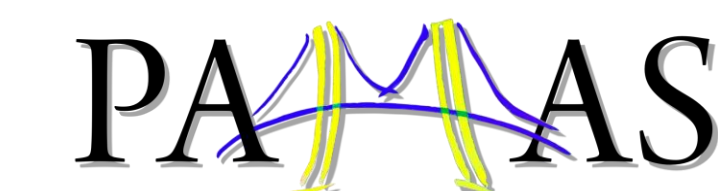


Scalable HMM based Inference Engine in Large Vocabulary Continuous Speech Recognition

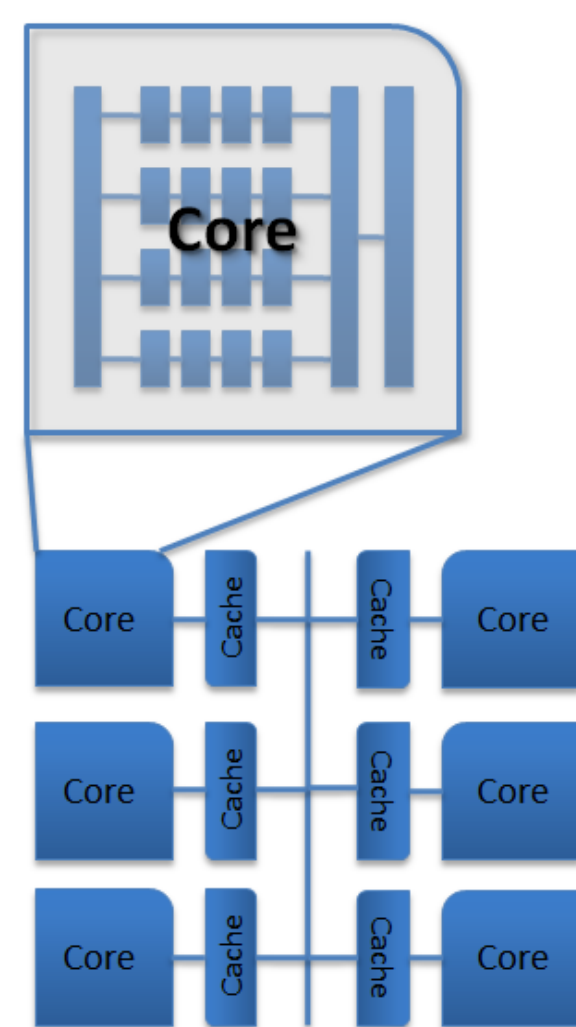
Jike Chong, Kisun You, Youngmin Yi, Ekaterina Gonina, Christopher Hughes, Wonyong Sung, Kurt Keutzer



Accelerating Speech Recognition on Multicore and Manycore Platforms



Target Parallel Platforms

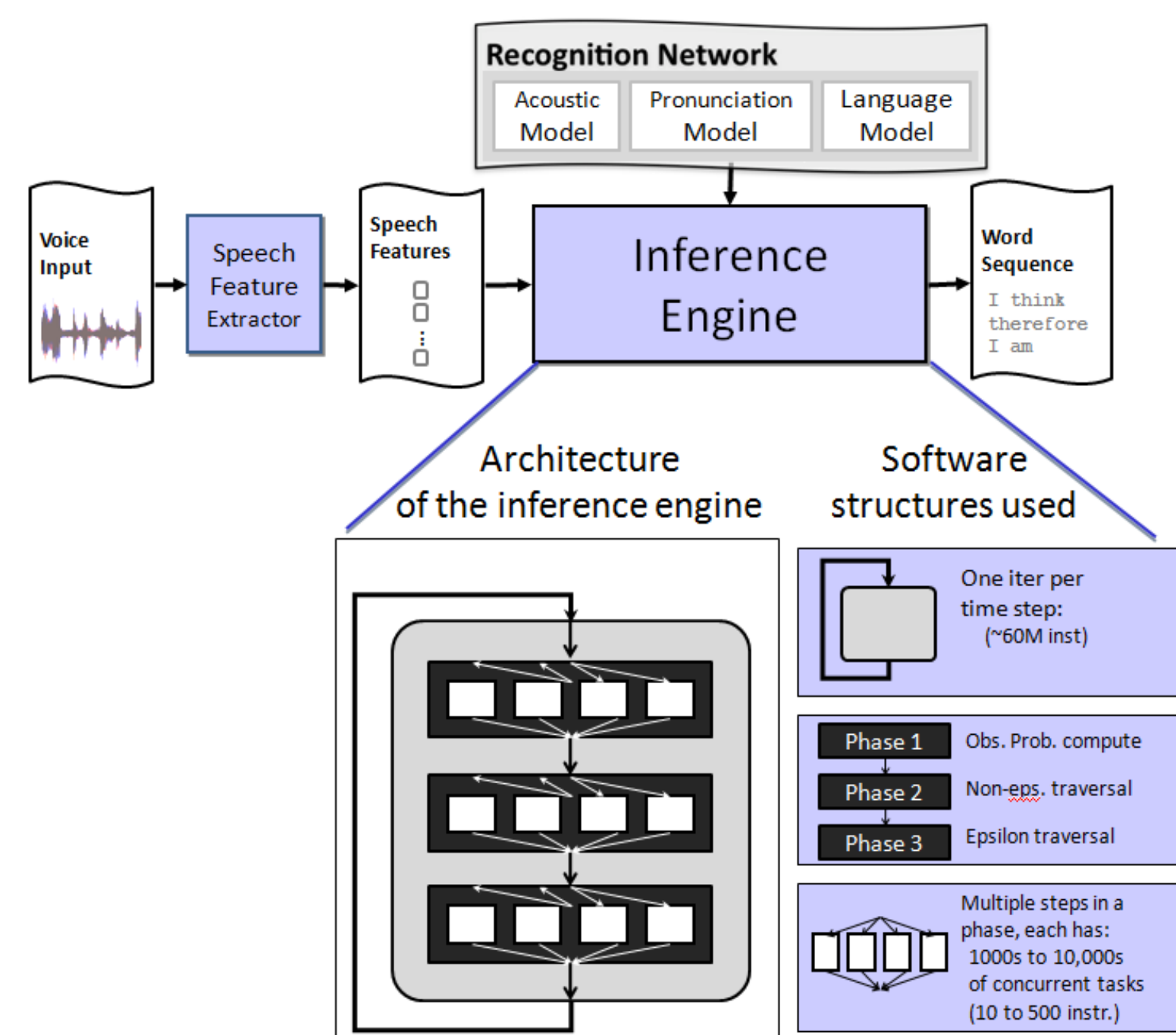


- Platforms targeted – Multicore and Manycore with SIMD units
- Architecture trend:
 - Increasing vector unit width
 - Increasing numbers of cores per die
- Application implications:
 - Must optimize synchronization cost
 - Must increase SIMD efficiency

Ongoing work investigates algorithm styles that can be applied for multicore and manycore programming

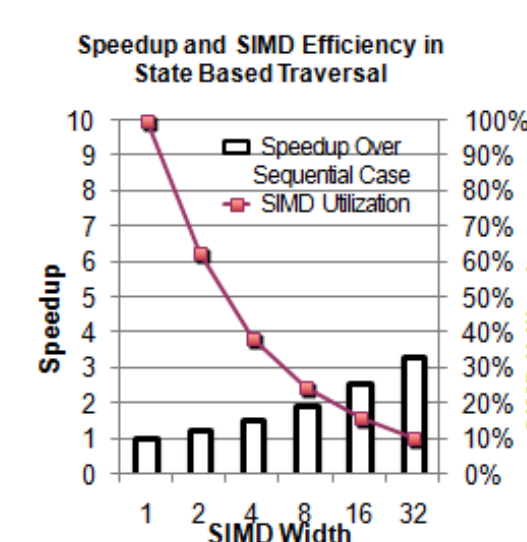
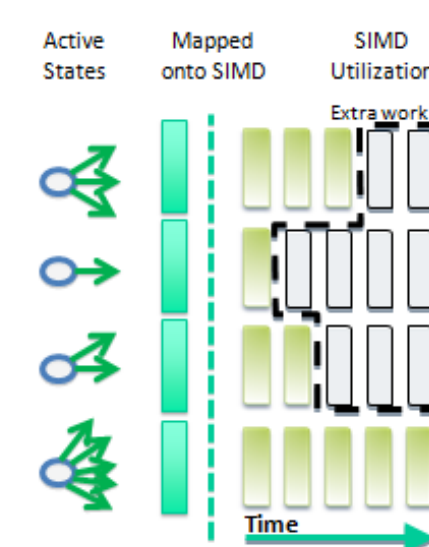
Speech Recognition Inference Engine

- Parallel graph traversal through irregular network
 - Guided by a sequence of input audio vectors
 - Computing on continuously changing data working set
- Implementation challenges
 - Define a scalable software architecture to expose fine-grained application concurrency
 - Efficiently synchronize between an increasing number of concurrent tasks
 - Effectively utilize the SIMD-level parallelism

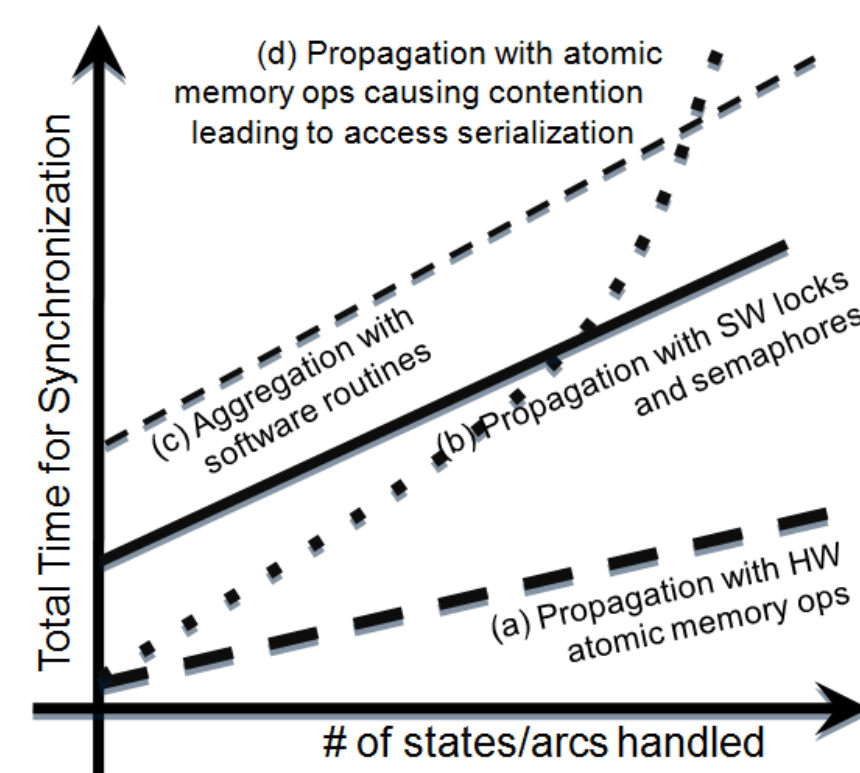


Algorithm Design Space Exploration

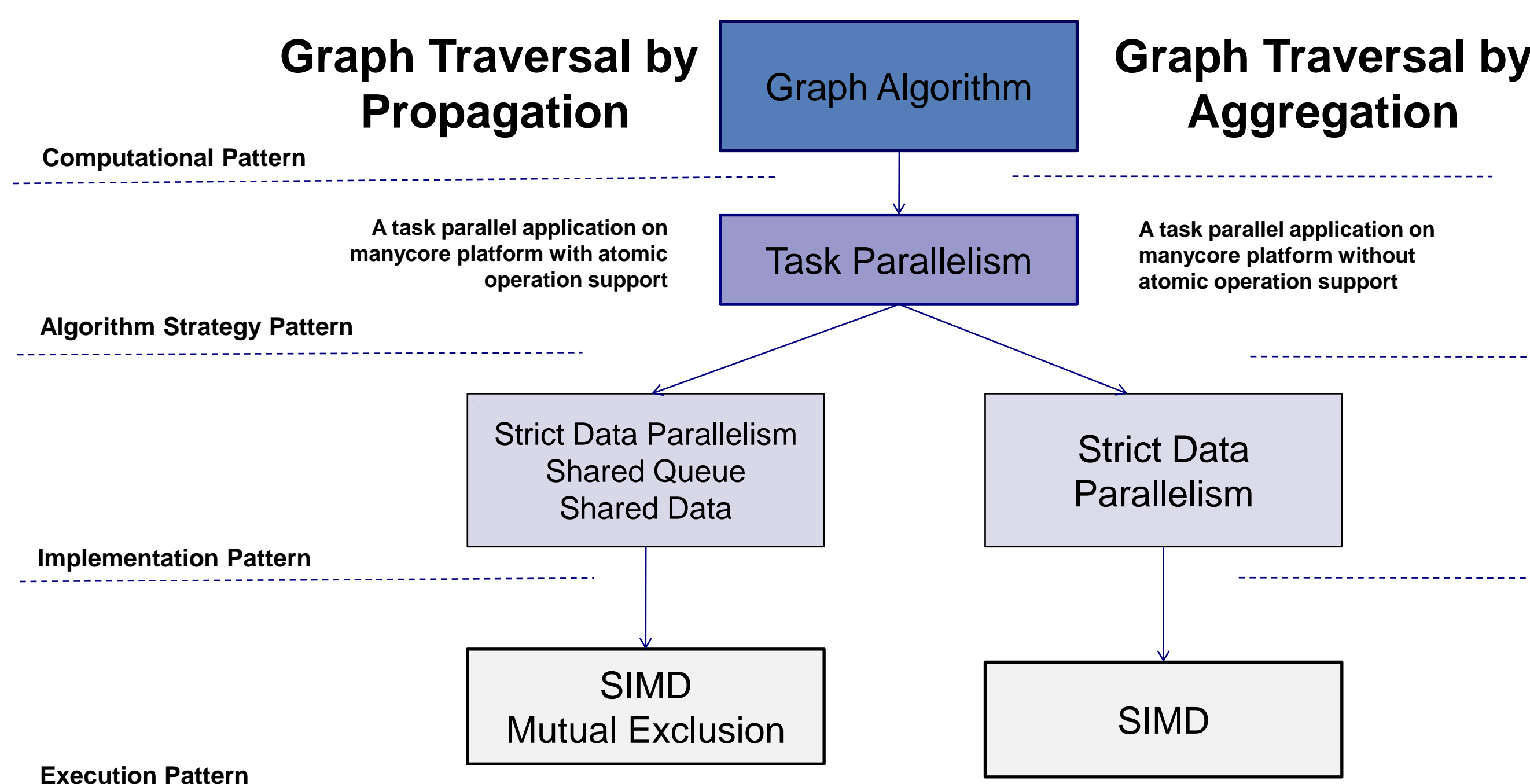
Arc-based Propagation Approach	Arc-based Aggregation Approach	Arc-based One arc at a time	Transition Evaluation Granularity
State-based Propagation Approach	State-based Aggregation Approach	State-based All out-going / in-coming arcs at a state	Addressing SIMD Utilization Efficiency
Propagate Traversal organized at source state	Aggregate Traversal organized at destination state	Graph Traversal Techniques Addressing Core level Synchronization Efficiency	



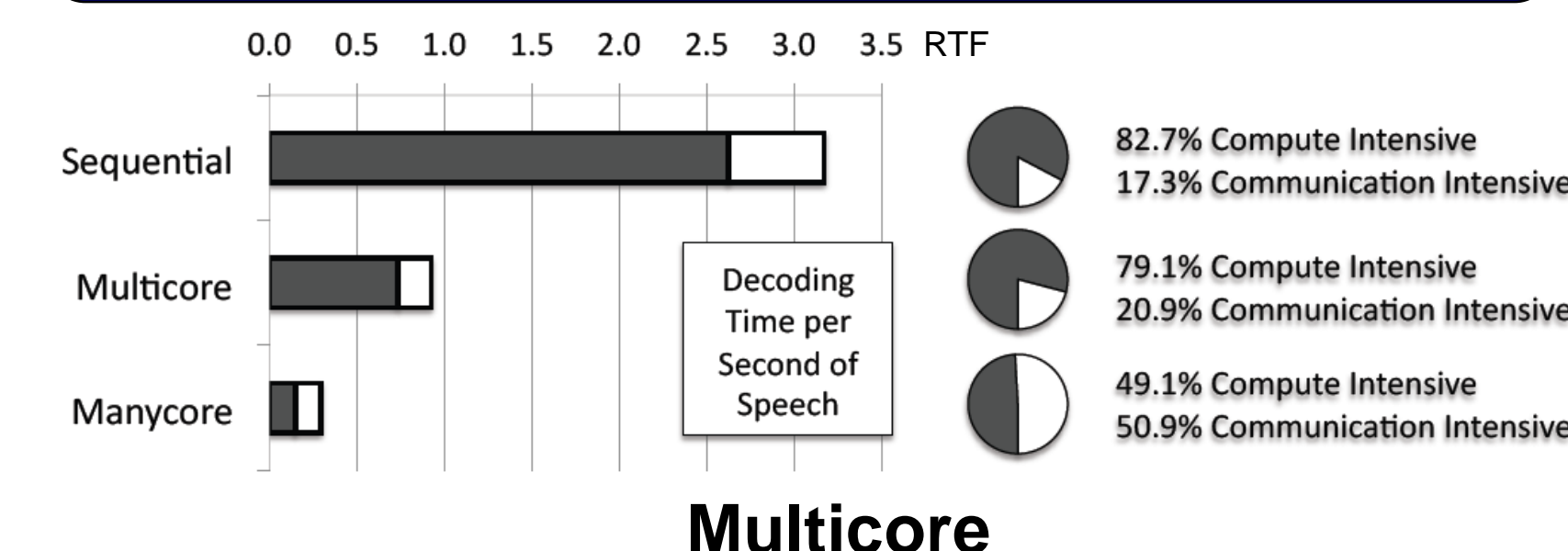
- Efficient graph traversal techniques will:
 - Reduce the task management overhead
 - Enable gaining additional speedup in scaling to more cores
- Efficient transition evaluation granularity is also important:
 - SIMD efficiency is indicative how well the algorithm would respond to HW increases in SIMD width



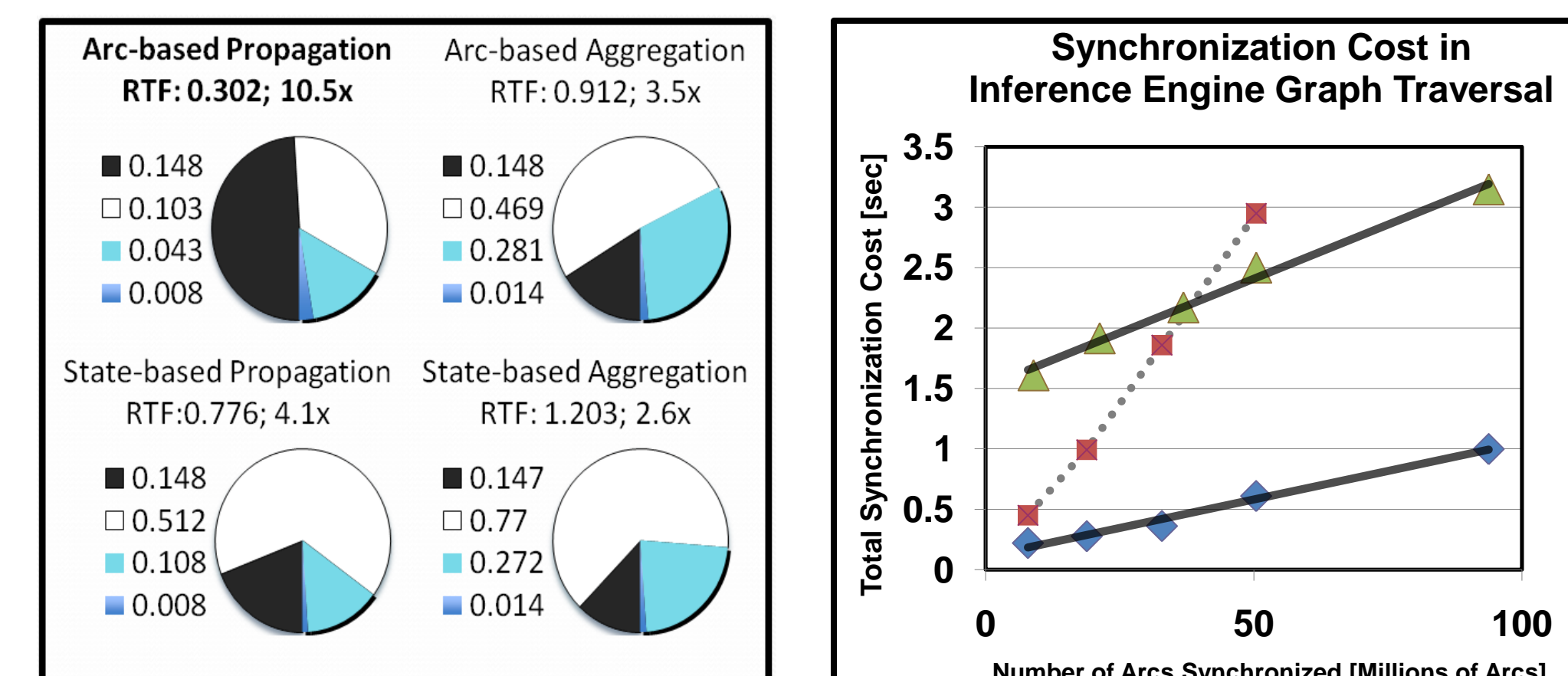
Patterns in the Graph Traversal Phase



Inference Engine Evaluation



- Speed up varies between phases:
 - 4-20x for compute intensive phases
 - 3-4x for communication intensive phases
 - Communication intensive phases becoming proportionally more important



Conclusions

- We have defined and implemented a parallel software architecture:
 - Less than 2.5% sequential overhead
 - Significant potential for further speedup in future platforms
- We have explored the algorithmic design space on two HW platforms:
 - The fastest algorithm style differed between platforms
 - Propagate techniques were able to use efficient HW atomics on both platforms
 - SIMD optimization will become more important going beyond four lanes