

Symphony: Probabilistic Graphical Models for Scheduling Heterogeneous Processors

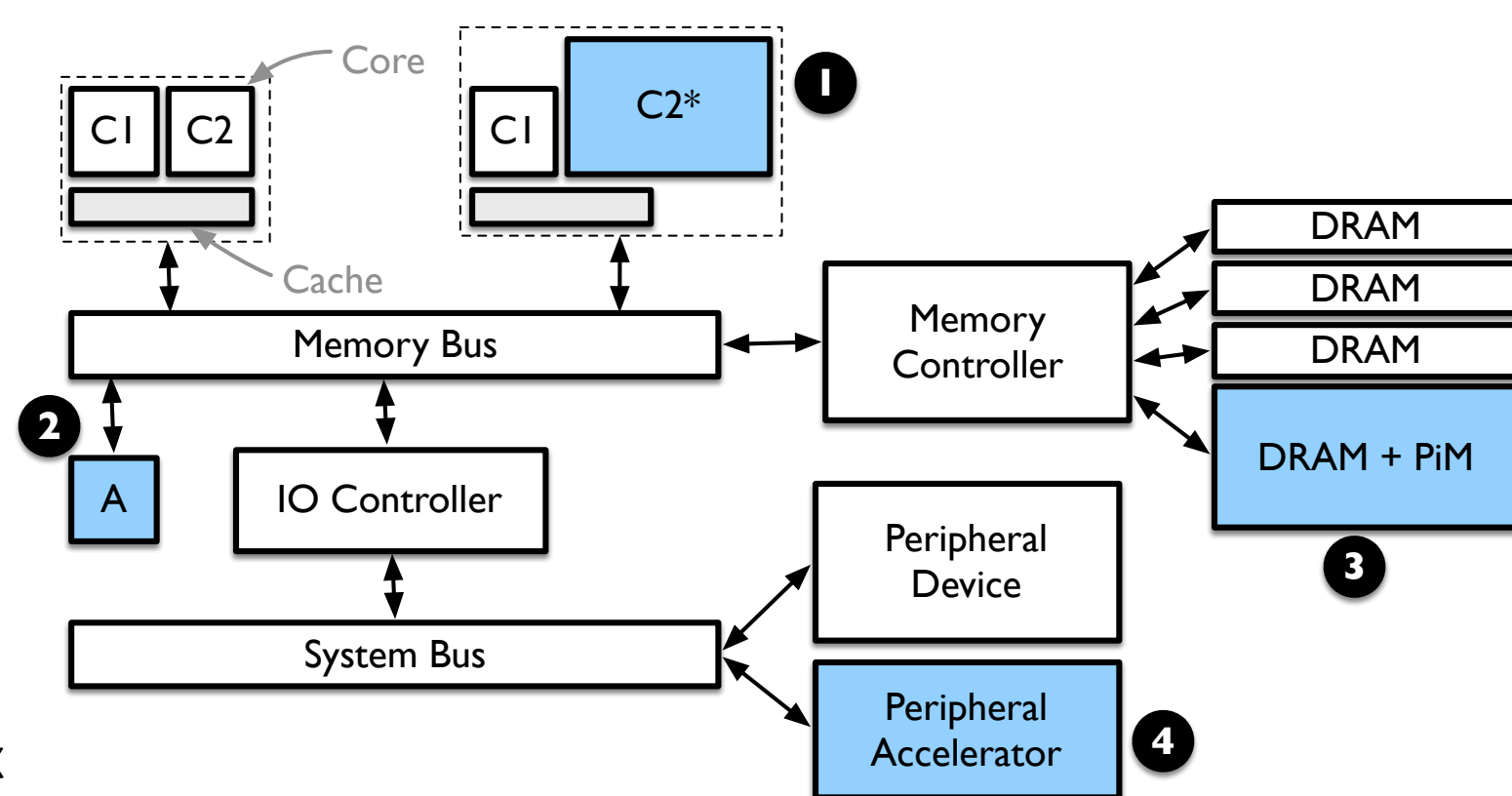
Subho S. Banerjee, Steven S. Lumetta, Zbigniew T. Kalbarczyk, Ravishankar K. Iyer
Departments of Computer Science and Electrical and Computer Engineering

Summary

- Presents the design of the CompGen system at UIUC
- Targeted at data analytics in the healthcare domain
 - Analytics tools and methods
 - Computer system architecture and design
- System consists of
 1. Hardware Component: The Computational Genomics Accelerator (TCGA)
 2. Software Component: Symphony Runtime
- Use intelligent (Bayesian) scheduling strategy to
 1. Abstract away low-level architectural details
 2. Model architectural resource utilization
- Variant Calling and Genotyping as a driving example for the system

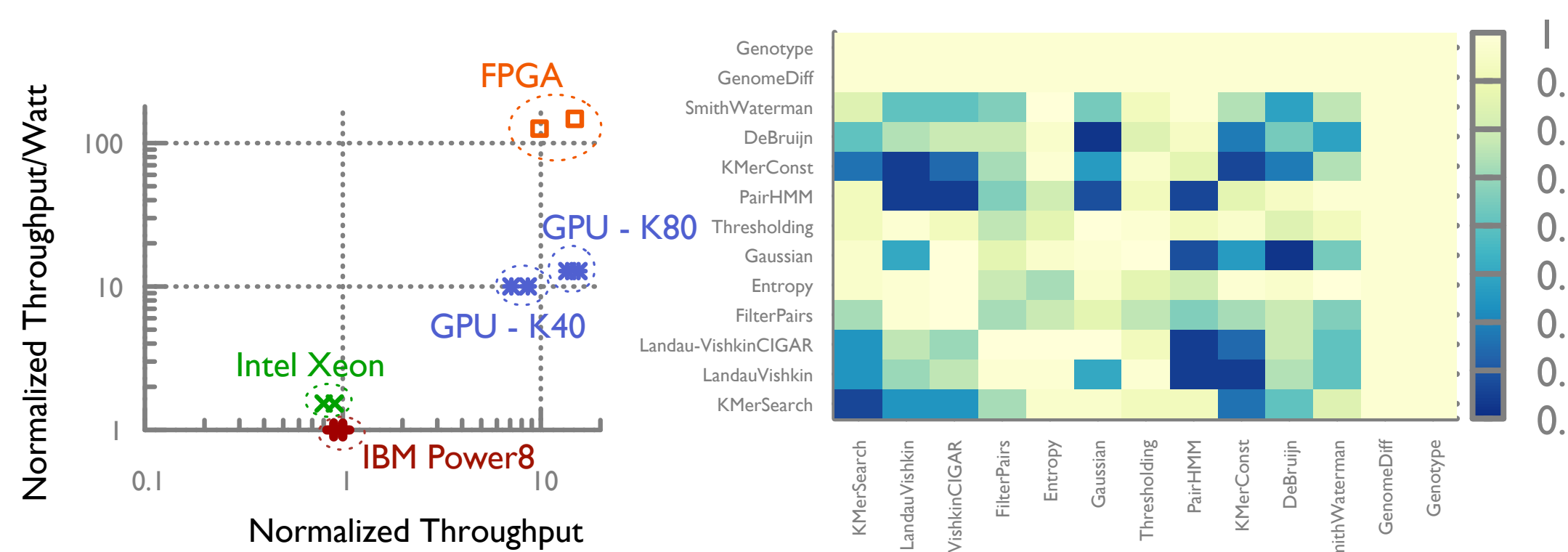
Motivation: Accelerator Centric Computing

- Accelerators are becoming an integral part of modern computing systems
- Programming and deploying applications on such systems is becoming increasingly complex



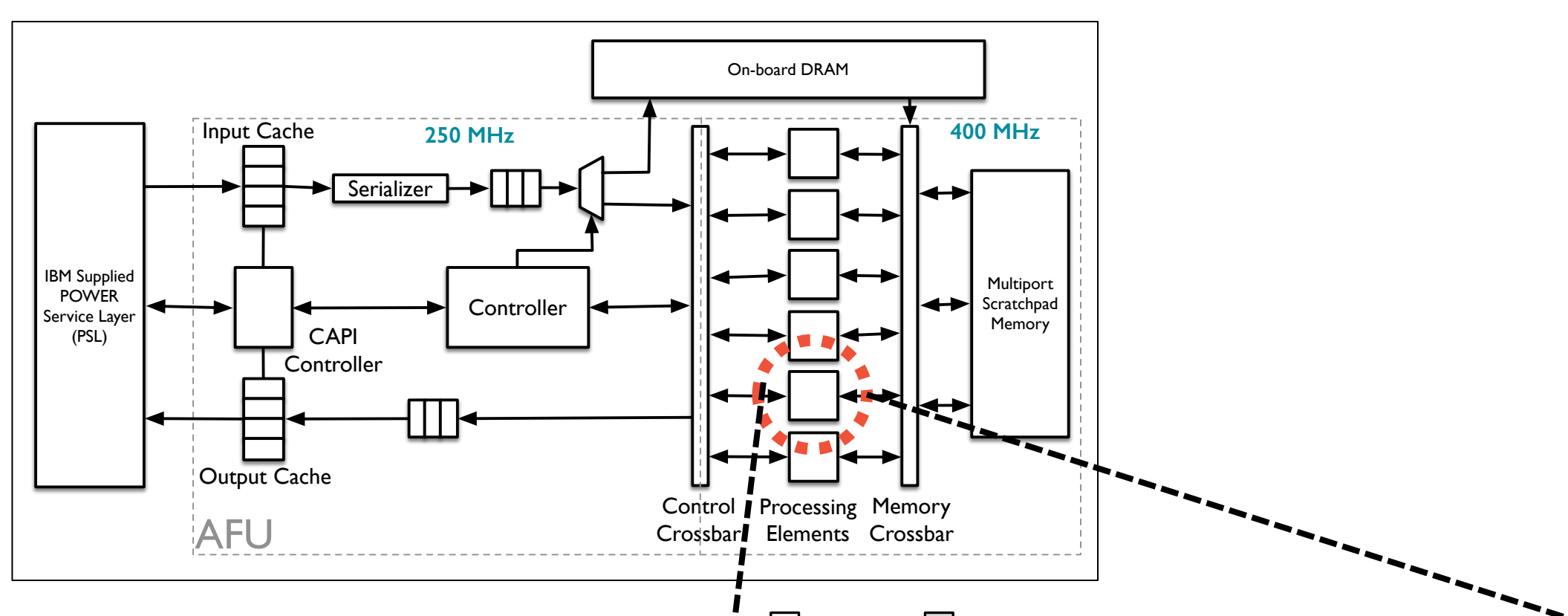
What are the key parameters to consider when designing systems/deploying applications to systems?

1. Processor Affinity: Some computations are inherently more efficient on certain processor architectures
2. Data Locality: Moving data through the system costs performance and energy
3. Shared Architectural Resource Contention between co-located tasks



The Hardware Component: TCGA

TCGA is a dynamically reconfigurable co-processor that executes computational kernels that are ubiquitous in computational genomics applications

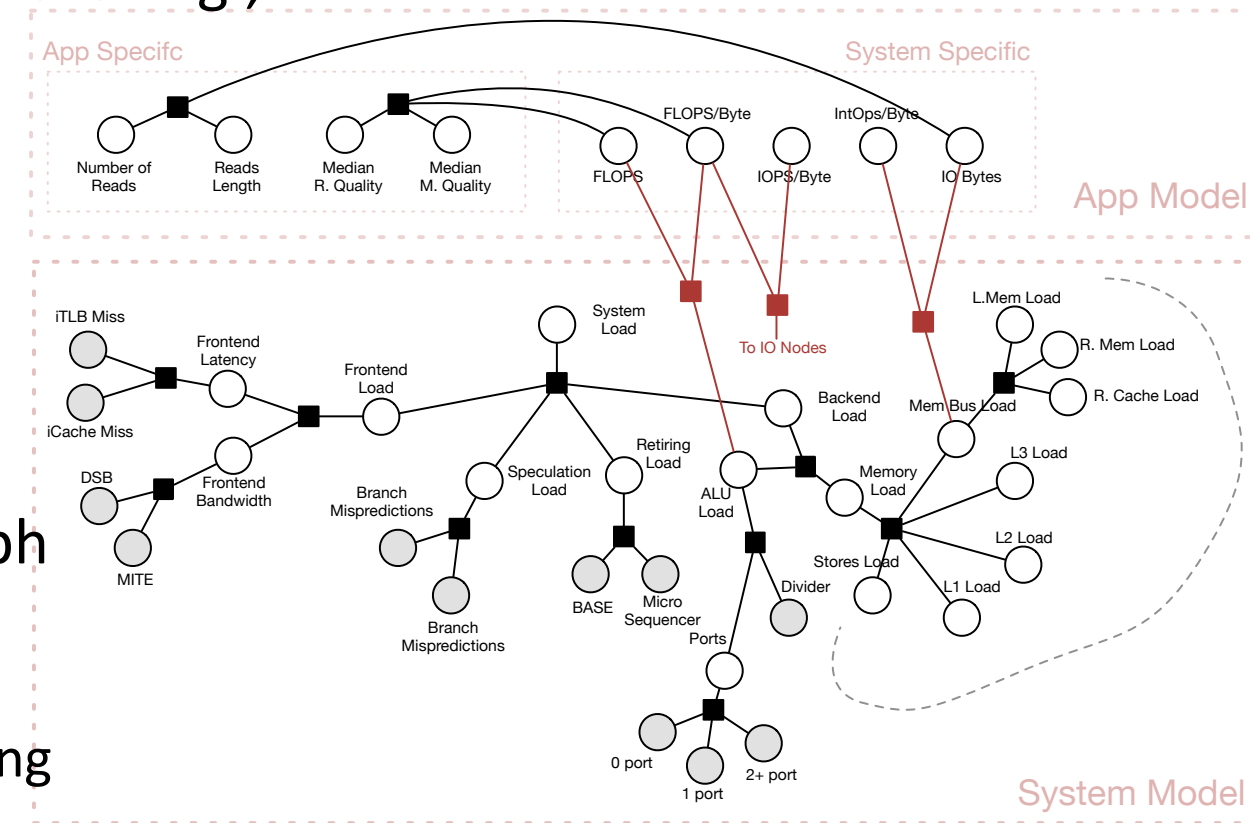
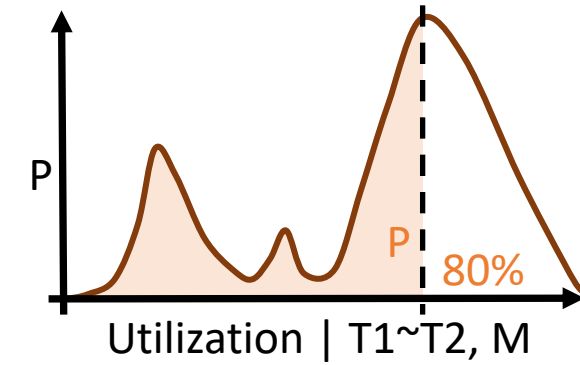


- Example: Levenshtein Distance Computation, Hidden Markov Models
- Prototyped on FPGA boards
- Interfaced with the host processor using IBM CAPI

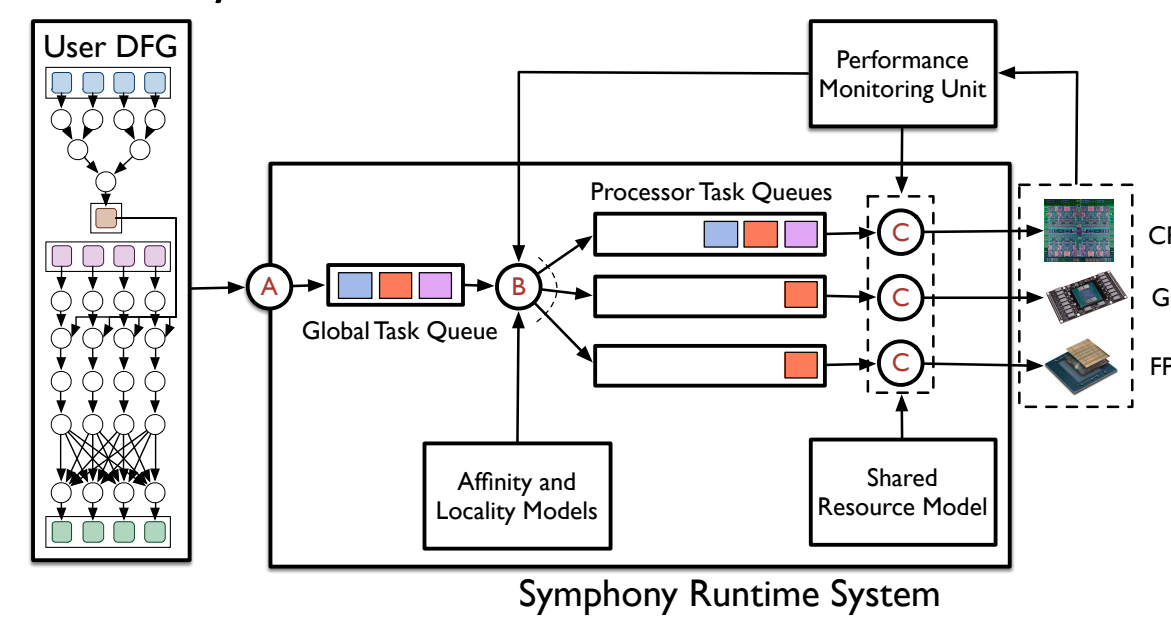
The Software Component: Symphony

- Probabilistic interpretation of resource utilization and performance of tasks running on heterogeneous processors and accelerators
- Reduces to testing probabilistic assertions over a model of applications requirements and system architecture. E.g.,

$$P(U \geq 80\% | T1 \sim T2, M) \geq 90\%$$



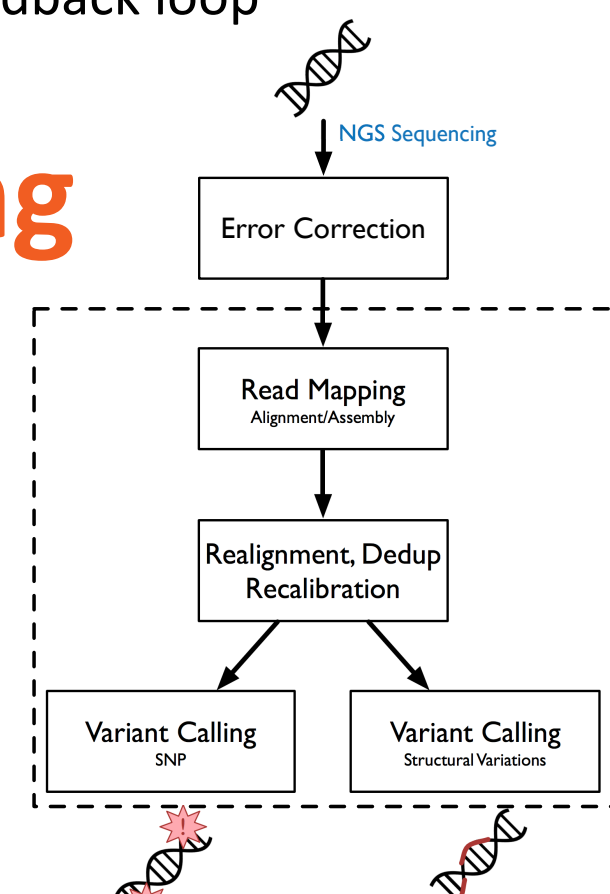
- System is modeled as a Factor Graph
 - Factors derived from processor performance manuals
 - Hidden states' values computed using Bayesian inference



- System state estimator can then be coupled with known scheduling algorithms
 - Measurement driven estimation of current system state
 - Optimality Guarantees
 - Closed feedback loop

Variant Calling and Genotyping

- Identifies and characterizes mutations in NGS data
 - Map NGS data to reference genome
 - Correct for noisy data
 - Differentiate strings in the presence of noise and ploidy
- First phase of the personalized medicine flow
 - Recurrently used
- Data intensive part of NGS analytics



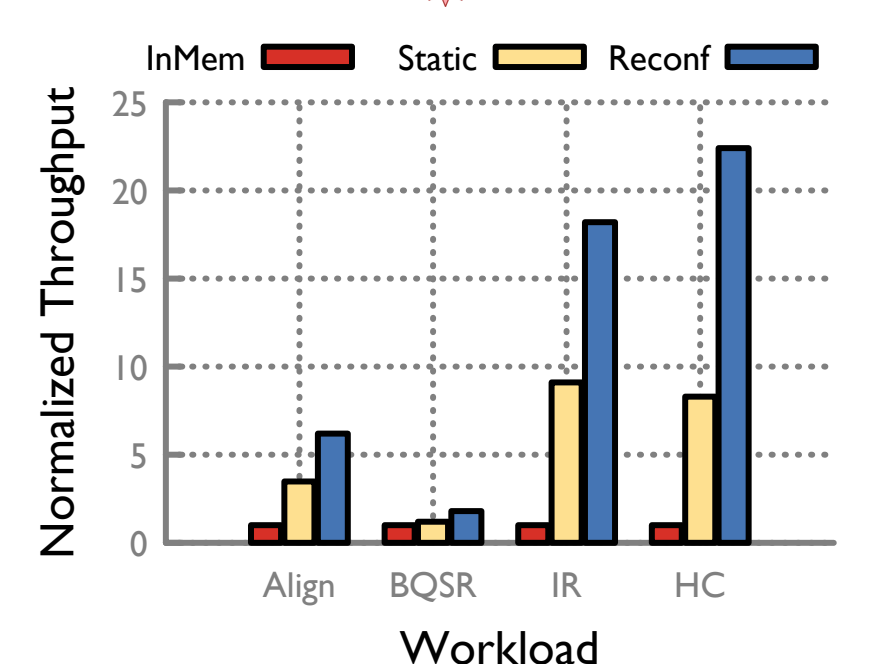
Results

Human genome runs, reduction in time from **73 hours to under 45 minutes** (baseline: single 2-socket CPU)

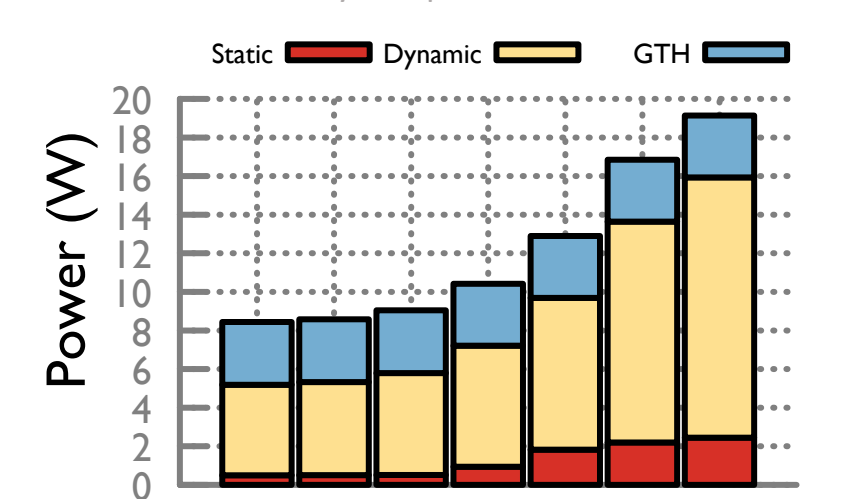
- 88x improved runtime performance
- 210x in terms of performance-per-watt

TCGA presents significant improvement in performance

- ~20x increase in throughput
- ~15x decrease in energy consumption



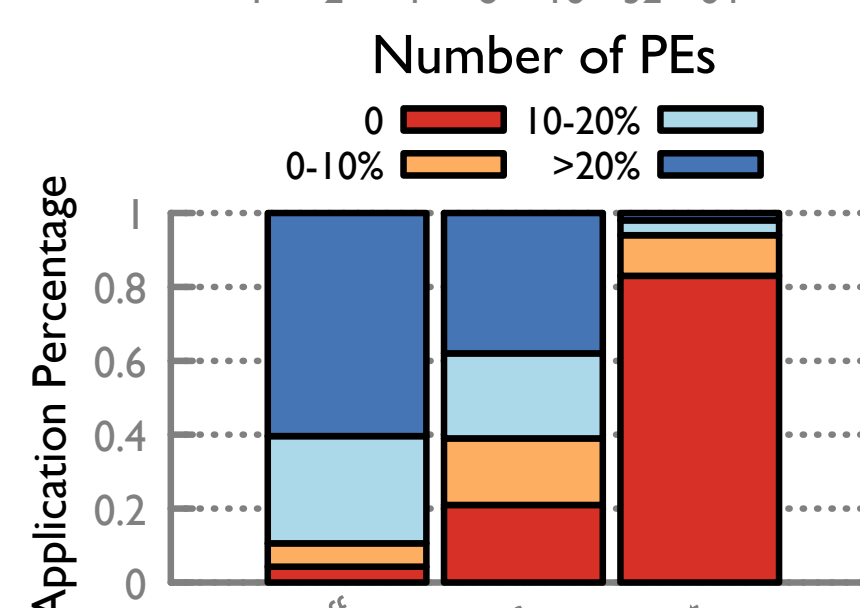
(Baseline "InMem" uses in-memory computation with NVIDIA K80 GPUs)



Symphony allows users to maintain level of abstraction and minimizes interference between co-located jobs

Conclusions

- Demonstrate the use of hardware-software co-design for computer systems for accelerated data-analytics
 - Efficiency in terms of performance
 - Efficiency in terms of energy efficiency
- Applied to several problems data analytics in computational genomics
- Potential application in a broader set of problems



Acknowledgements

This research was supported by several grants: in part by the National Science Foundation under Grant No. CNS 13-37732; in part by the Blue Waters sustained-petascale computing project supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) and the state of Illinois; and in part by IBM, Xilinx, Intel for providing equipment support.

