

MÉLANGE: Spring 2018

Sanjay Rajopadhye
Colorado State University

Plan for today

- Introductions
 - Name, year, status
- PARRIC: Polyhedral Acceleration of RNA-RNA Interaction Computations
- Future directions

NSF SPX: PARRIC (submitted)

- RNA-RNA Interaction (RRI) is an important scientific challenge
 - Potential cure for cancer
- Computational models of RRI (piRNA, IRIS) are important but very expensive in
 - Time: $\theta(N^3M^3)$ for lengths M and N
 - Space $\theta(N^2M^2)$
 - We will show **100-million-fold** speedup

Colorado State University ³

Need for Speed

- piRNA is slow and a memory hog
 - For two sequences of length 100 each, piRNA takes 3.5 hours on a 64-core machine with 512GB RAM
 - Cannot handle sequences longer than 200 ($N*M > 40k$)
 - Machine goes unresponsive **must be rebooted**
- For whole genome analysis:
 - 30,000 genes (~2k length)
 - 500 “interesting” small RNAs (length ~100)
 - i.e., 15 million calls to piRNA
 - Each call to piRNA would take 3.5*8000 hrs
 - On an **8 Terabyte machine**
 - 15 million calls would take **50 million years**
 - We will do it in **six months on 100 department machines**

Colorado State University ⁴

How to get there

- Easy parallelization (use 10^6 “large enough” machines on the cloud) is too expensive
- Make piRNA run **efficiently** on small RAM machines (e.g., 16GB)
- Speed it up on 100 machines in the department
 - Need 10^6 -fold speedup on each machine
 - 1000-fold by using locality/parallelism/vectorization
 - 1000-fold by filtering on only 0.1% **interesting pairs** of sequences
 - Still needs 1000-fold speedup of **filtering program**

Colorado State University 5

WE can get there

Showed 100-fold speedup of “miniapp” called (OSP)²

- 100-times simpler than piRNA
- On small (fits in RAM) problem sizes
- We have the expertise to do this on multi-cores & also GPUs

Colorado State University 6

Abstractions

Six abstractions

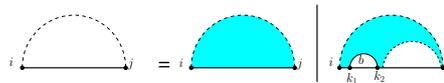
- Algorithms (Alg)
- Applications (App)
- Programming Languages and Systems (PLS)
- Architecture and Systems (Arch)
- Extensible Distributed Systems (EDS)
- Performance Portability (Perf)

Explain which abstractions you break, what you propose, and how you will validate

Colorado State University ⁷

Existing Abstractions

For application specialist (e.g., piRNA author)



Eddy Diagrams

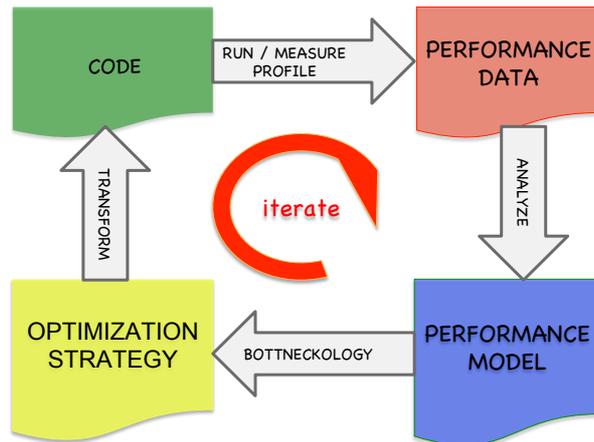
CODE

$$Q_{i,j} = \sum_{i \leq k_1 < k_2 \leq j} Q_{k_1, k_2}^b Q_{k_2, j}$$

Recurrence Equations

Colorado State University ⁸

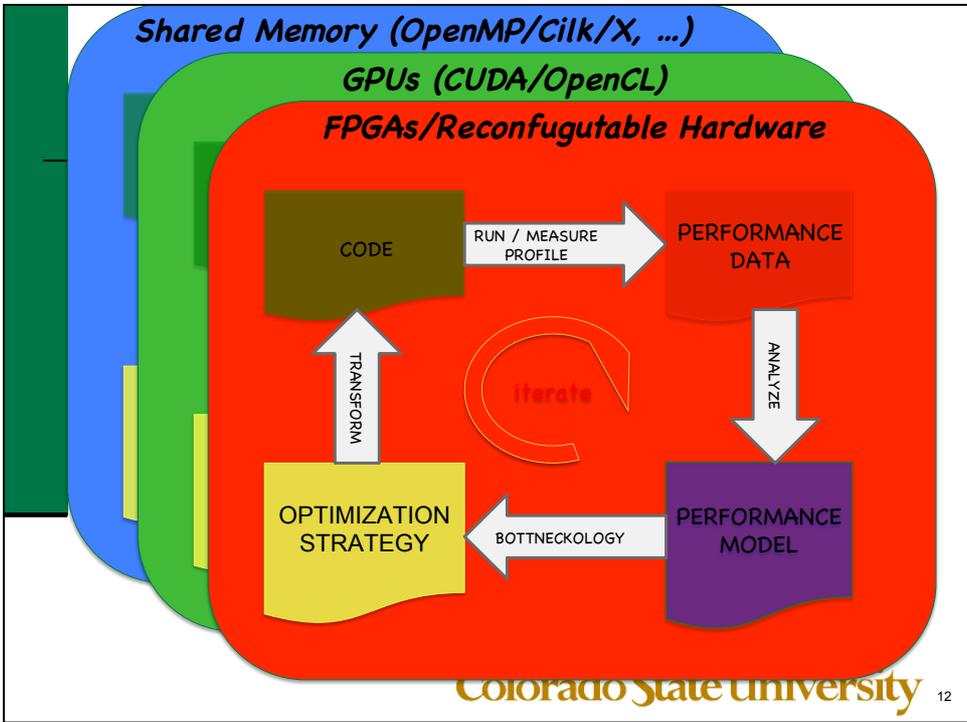
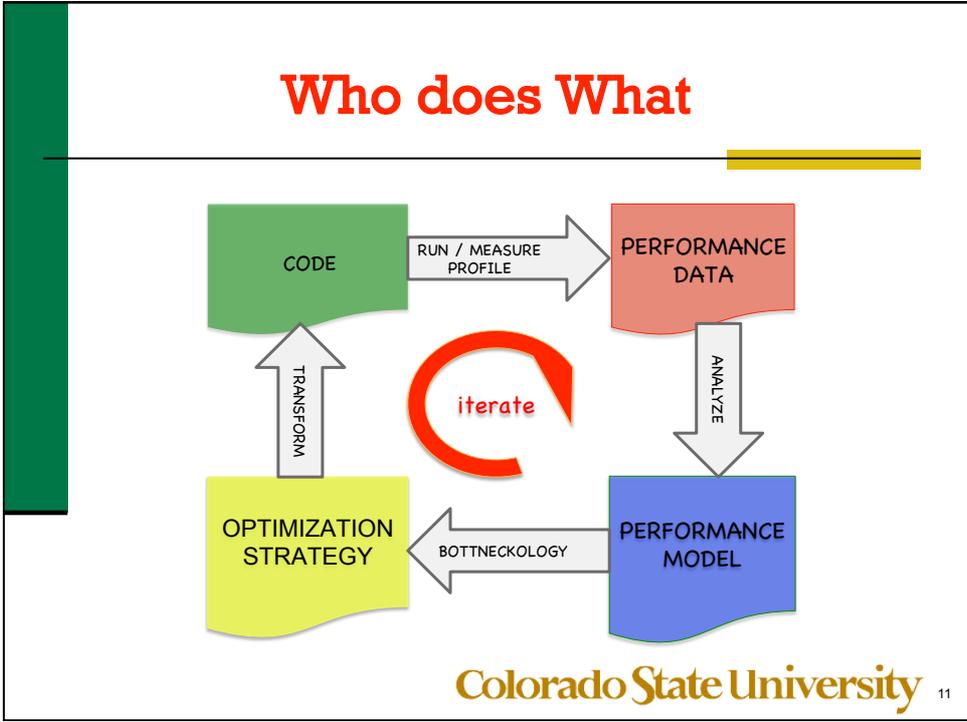
HPC specialist + Tool builder



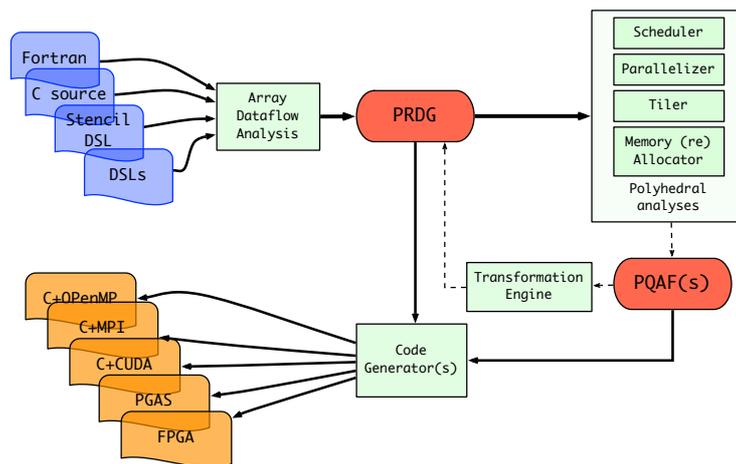
Colorado State University 9

CODE

Colorado State University 10



Polyhedral Model (Zorro)



Colorado State University 13

Polyhedral Challenges

- **piRNA** is beyond current tools (few tens of lines to **kLoC**)
- **Multilevel tiling**: virtual memory, DRAM, caches, (and 2-levels of parallelism: cores & vector units)
- **Legality of tiling**: all six dimensions must be tiled: **is that legal?**
 - GKT: middle serialization (known since 1979, still not fully automatic)
- **Raise** the level of abstraction:
 - Sloppy Equations: Hamid should write ~100 eqns, not kLoC in C++
- **Simultaneously schedule & tile reductions**
 - On **(OSP)²** P_{Lu}T_O **slows down** the program (mostly)

Colorado State University 14

Possible Projects

- Three orthogonal choices
 - Algorithm/Program:
 - Matrix multiplication: *MM*
 - Matrix multiplication on max-plus semiring *MPMM*
 - Optimal String Parenthesization: *OSP*
 - Square of that: $(OSP)^2$
 - Others
 - Platform: *CPU* or *GPU* or *FPGA* or ???
 - Problem size: *standard* (fits in DRAM) or *large* (needs virtual memory)
 - Bottleneckology: take it as far as you can

Colorado State University 15