
Explicit Sequential Programming for Implicit Parallel Performance on Many Cores

Devesh Tiwari and Yan Solihin

ARPERS Research Group

Architecture Research for PErformance Reliability and Security

Electrical and Computer Engineering

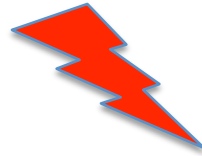
NC State University

NC STATE UNIVERSITY

Problem

Sequential Programs

Gap between



Many Cores



In 2020, Who is my that friend in red ?

Opportunity

Bleed yourself, Save Others

We are Good Friends !

Really?

This is what you have given so for! Threads, Synchronization Primitives



This is what I end up managing(?) races, complex thread interactions



Good Friend Qualifications

Scalability till eternity

Determinism as legal compliance

ZERO Synchronization and communication burden

Intuition*

Childhood days, game of fitting boxes with my brother and others , inherently almost sequential.

Two Teams

First all working simultaneously, speculating and trying, all busy all the time.

Second One actively working while others practicing in parallel until the real turn comes.

* looking back helps !



From Intuition to Solution

Split the program into phases

Each phase-thread 1.Input Set 2.Output Set

Each phase-thread convert to local phase variables

Spread each phase to one particular core

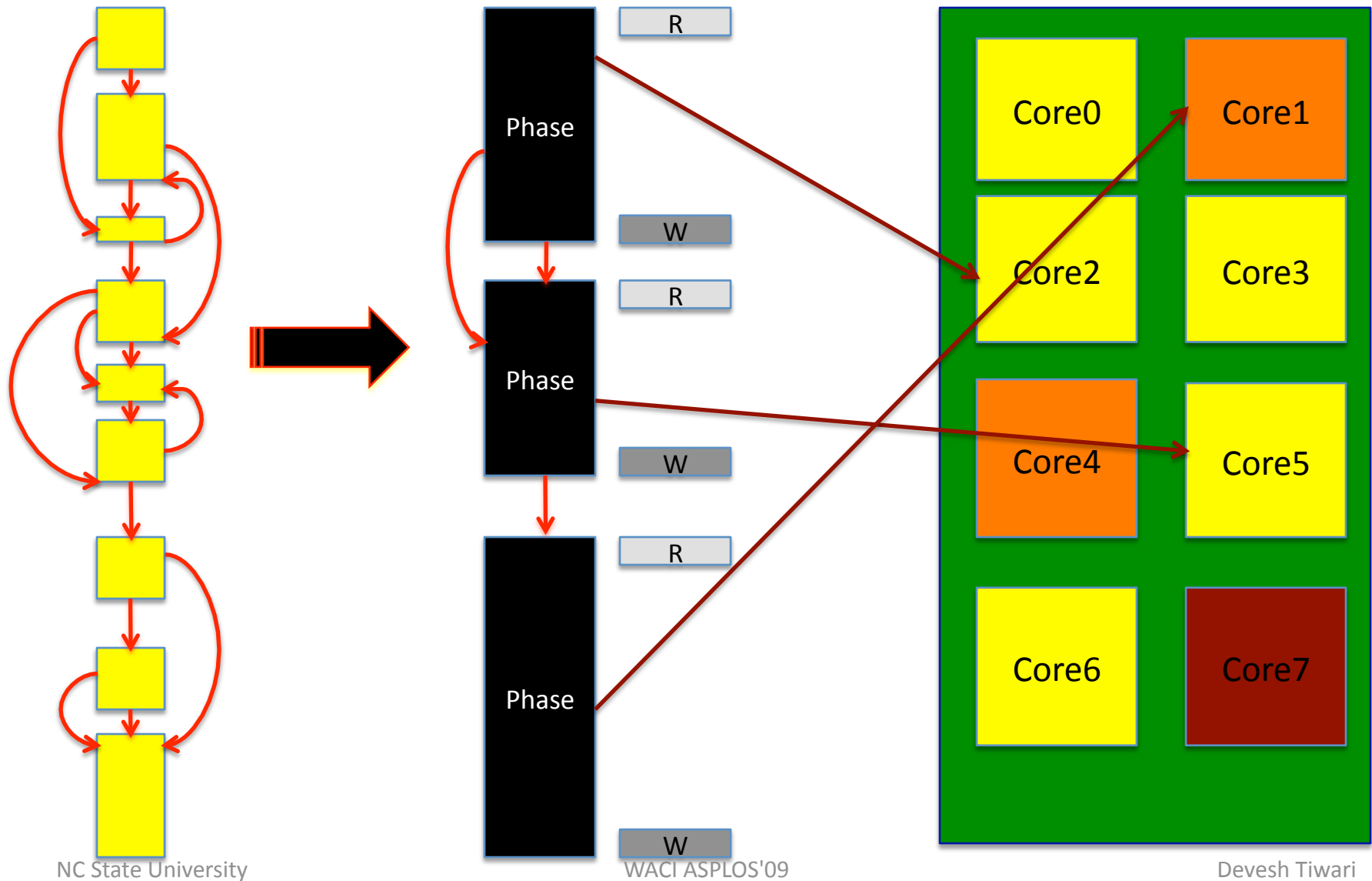
Start dummy executions of each phase-thread

Take care of correctness

Master thread signals when your turn comes

Do final-run on your turn

View from 36000 feet



Opportunities

Redundant Execution: Training a phase

- Cache effects, Branch prediction

- Online Optimizations

- Memorize expensive things

- Asymmetric cores, find out the suitable core

- Trade off cache space and other resources for a while

- Dummy executions to make friends and offer help

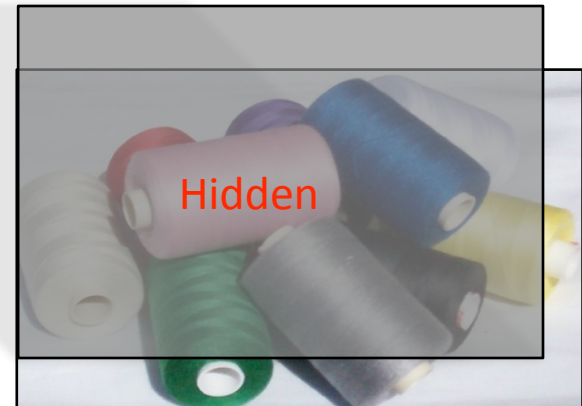
Learning without barrier

- learn phase specific details while I/O in another phase

- Preventing biased learning - input criticality prediction

finally, aha !

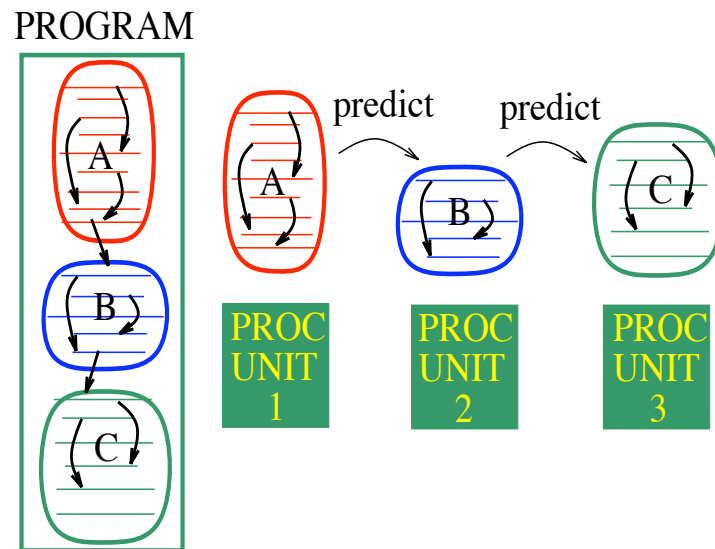
Break the sequential Program in many black boxes with input-output set variables and execute redundantly on different cores while working on box-local variables and marinating the correct order of final execution.



Need to Look Back ?

Looking Back to Look Forward

Multiscalar, Dynamic Multi Threading, Run ahead Execution
Time to revamp all these more aggressively in
many core era.



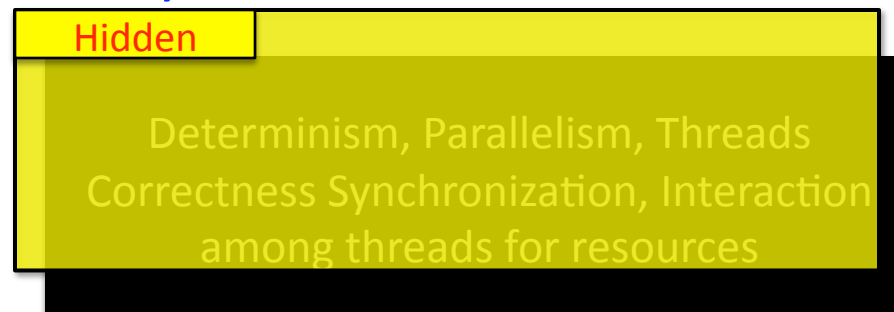
Different Perspective

Programming Model Perspective

Expose Black Box (phase based) Programming to
Programmers, Intuitive to Programmers

Sequential thinking and programming, implicit parallel
performance.

- Deterministic
- Highly Modular and Scalable
- ZERO burden of synchronization and communication



Power

Microwave anyway came free with my computer ! Every morning I turn on my computer to make breakfast and check e-mails.



Dummy executions in low power mode.
Restrict number of dummy executions.
Save power by memorizing stuff.

CAT Warmer (WACI 06), Water Heater (WACI 08)

Acknowledgement

CESR Group Members

S⁴: Siddhartha, Salil, Sandeep, Samih

Figures from

Guri Sohi, Speculative Multithreaded Processor, Talk at Intel, August 2001.

<http://i29.tinypic.com/>

<http://timetobleed.com/>

<http://wikimedia.org/>

<http://www.vivid.ro/>

<http://www.taillightking.com/>

<http://www.greatlittlebox.com/>

<http://www.joe-ks.com/>

Questions

Devesh Tiwari

devesh.dtiwari@gmail.com

<http://www4.ncsu.edu/~dtiwari2>