Basic Block Distribution Analysis to Find Periodic Behavior and Simulation Points in Applications

Timothy Sherwood *Erez Perelman*

Brad Calder

University of California, San Diego

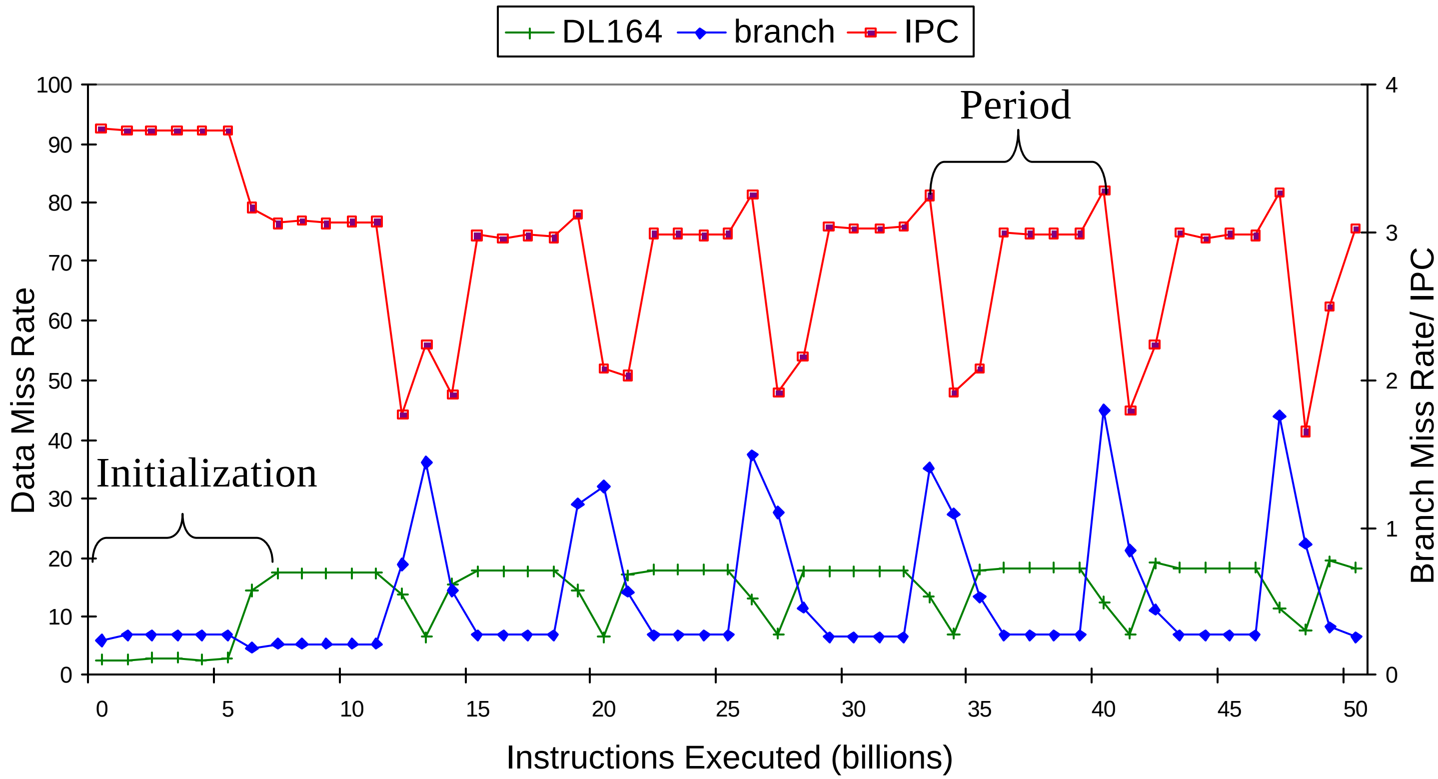
Motivation

- Architecture researchers conduct detailed pipeline simulations
- Length of detailed pipeline simulation
  - Simple Scalar: 400 million instruction per hour
  - Spec programs: 300 billion instructions
  - Complete run: 1 month
- Limited simulation time and processing power
- Often only a subset of whole program is simulated
- Subset should represent the overall behavior of the program

Phases of Execution

- Initialization phase
  - Initialize data structures and set up for the rest of execution
  - Does not represent overall behavior of program
  - Current methods: fast forward or check points
- Steady state
  - Programs tend to be written in a nested loop fashion
  - Correlated with looping behavior of program

Cyclic Behavior of Wave



University of California San Diego

Goals of Research

- Automatically generate:
  - Length of initialization phase
  - Period length
    - Cyclic portion of execution
  - Ideal starting simulation point
    - For a given number of instructions
- Confidence of simulation points
  - Estimation of accuracy

University of California
San Diego

Outline

- Basic Block Distribution Analysis
- Initialization Phase
- Period
- Where to Simulate
- Conclusion

Approach

- A way to represent snapshots of program
- A metric that compares snapshots to whole program
- Uniquely identify phases of execution
- Signal processing for period computation

Program Fingerprint

- Metric independent method to represent program
- Basic Blocks uniquely identify the code executed
  - Directly affects program behavior
- Unique representation of program execution interval
    - BB vector

Basic Block Vector

BB	Assembly Code of bzip	
1	srl	a2, 0x8, t4
	and	a2, 0xff, t12
	addl	zero, t12, s6
	subl	t7, 0x1, t7
	cmpeq	s6, 0x25, v0
	cmpeq	s6, 0, t0
	bis	v0, t0, v0
	bne	v0, 0x120018c48
2	subl	t7, 0x1, t7
	cmple	t7, 0x3, t2
	beq	t2, 0x120018b04
3	ble	t7, 0x120018bb4
4	and	t4, 0xff, t5
	srl	t4, 0x8, t4
	addl	zero, t5, s6
	cmpeq	s6, 0x25, s0
	cmpeq	s6, 0, a0
	bis	s0, a0, s0
	bne	s0, 0x120018c48
5	subl	t7, 0x1, t7
	gt t7, 0x120018b90	
...	...	

BB Vector		
BB#	# times executed	Normalized
1	100	0.250626
2	89	0.223057
3	83	0.208020
4	71	0.177944
5	56	0.140350
...	...	...

Basic Block Vector Comparison

- Target Vector: BB vector of complete run
- Interval Vector: BB vector of a continuous interval of execution in program
- Vector Difference: how close BB vector is to the target vector

BB Interval Vector	
BB#	Normalized
1	0.250626
2	0.223057
3	0.208020
4	0.177944
5	0.140350
...	...

–

BB Target Vector	
BB#	Normalized
1	0.341624
2	0.159242
3	0.205486
4	0.242058
5	0.051590
...	...

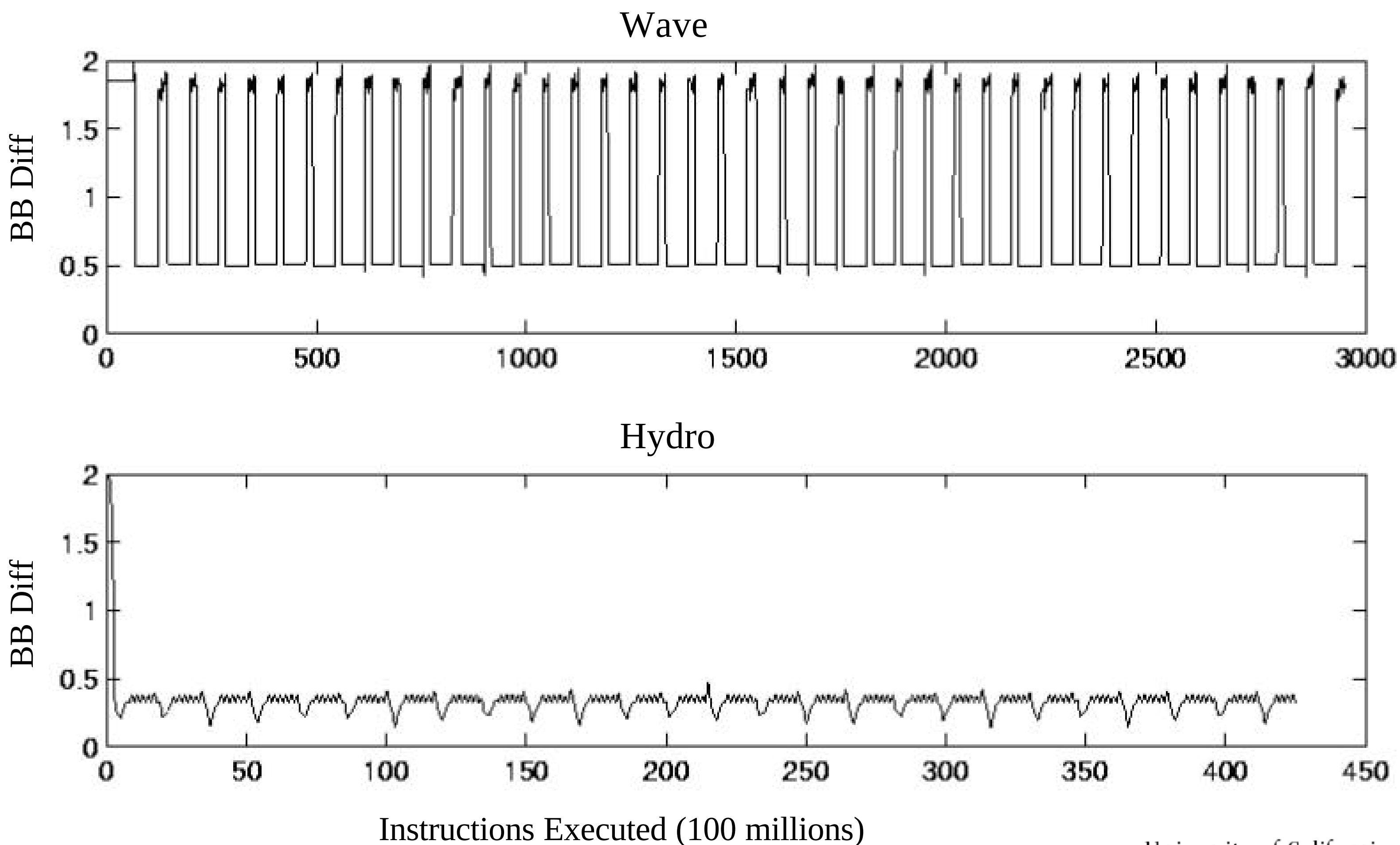
=

Diff BB Vector	
BB#	Abs Diff
1	0.090998
2	0.063815
3	0.002534
4	0.064114
5	0.088760
...	...

S

0.310221

Basic Block Difference Graph



University of California San Diego

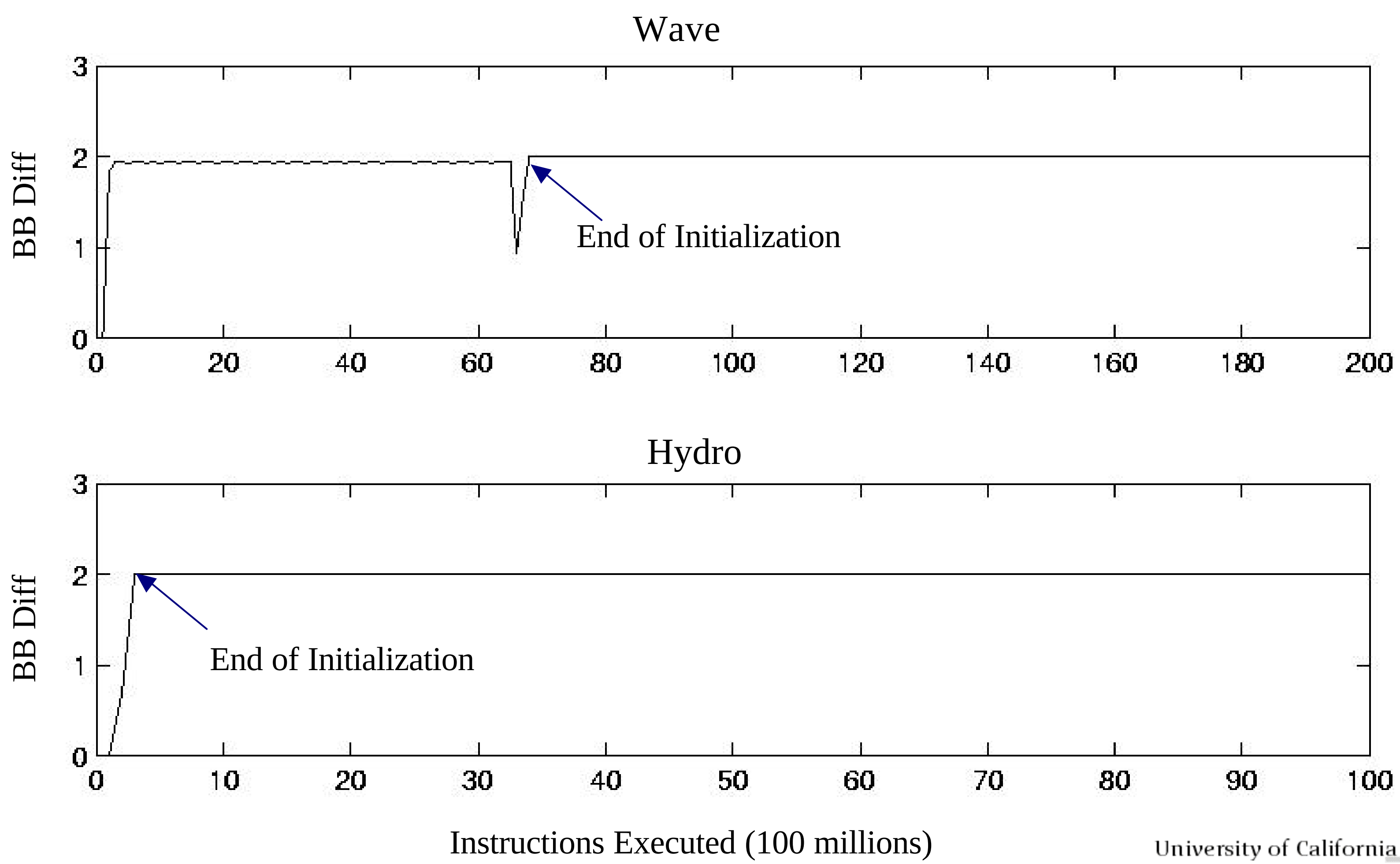
Outline

- Basic Block Distribution Analysis
- Initialization Phase
- Period
- Where to Simulate
- Conclusion

Initialization Phase

- Create a Basic Block Difference Graph of initialization
  - Target vector is first 100 million instructions
- End of Initialization
  - The max vector diff point in graph
    - In most cases is 2

Initialization Phase



University of California San Diego

Outline

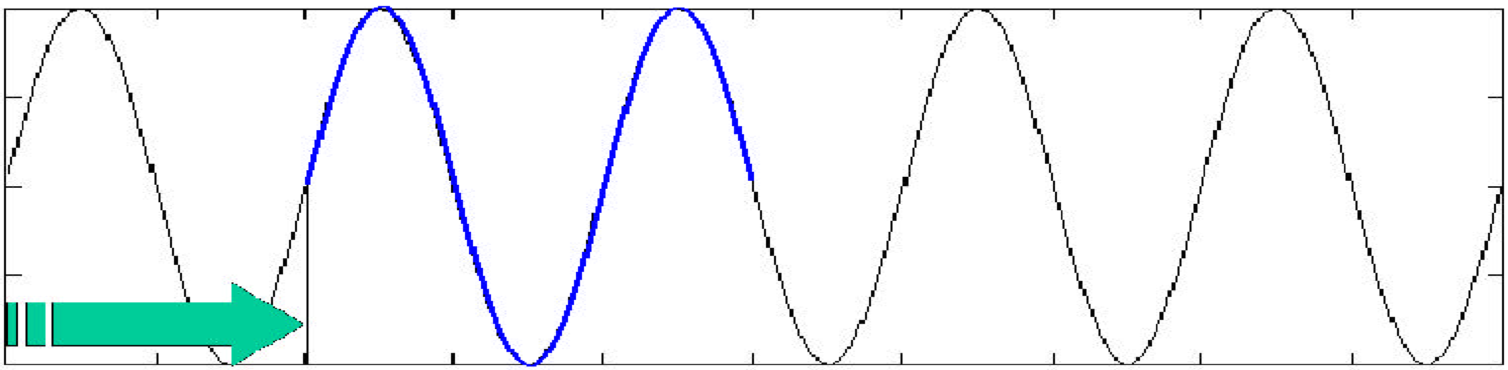
- Basic Block Distribution Analysis
- Initialization Phase
- Period
- Where to Simulate
- Conclusion

Signal Processing Theory

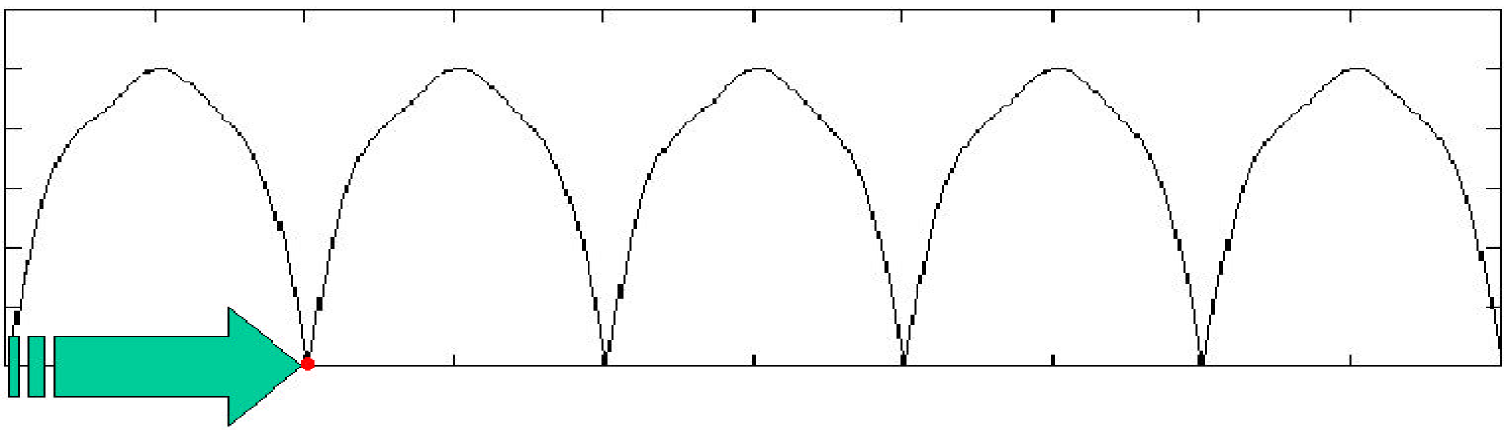
- Treat BB Diff Graph as a signal
- Signal shift and comparison
  - Signal shift will go in-and-out of phase
  - Comparison to evaluate phase
- Period deduced from phase cycle

Signal Difference Example

Signal Phasing



Period Difference Graph

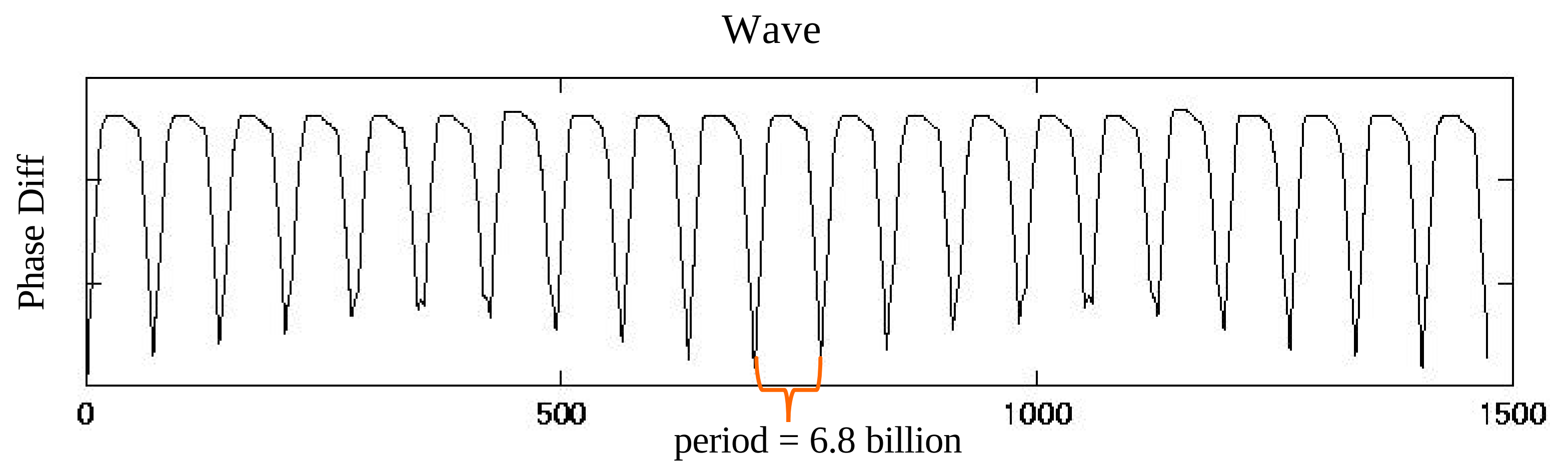


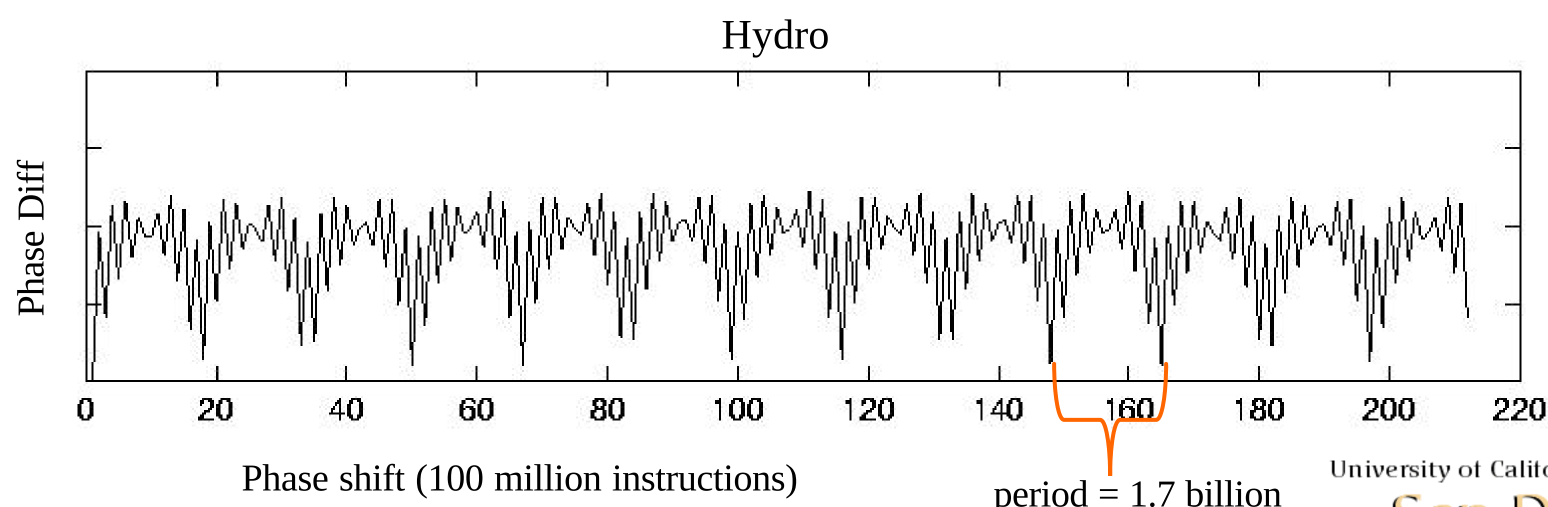
nia
San Diego

Period

- Start signal at end of initialization
  - Pick portion to shift to be quarter length of signal
- Shifting: generate Period Difference Graph
  - Minimums correlate to period-synchronized shifts
  - Amplifies the cycle over the BB Diff Graph
- Calculate period
  - Find all minimums
  - Calculate average distance between adjacent minimums

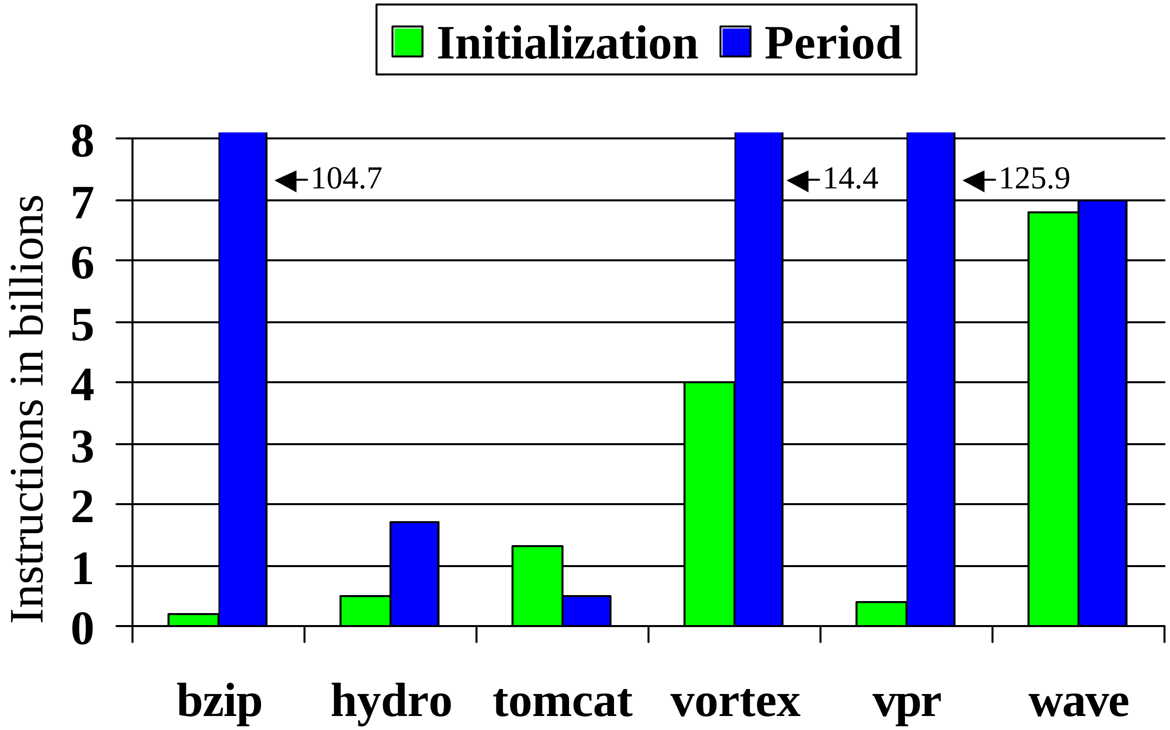
Period Difference Graphs





University of California San Diego

Initialization and Period



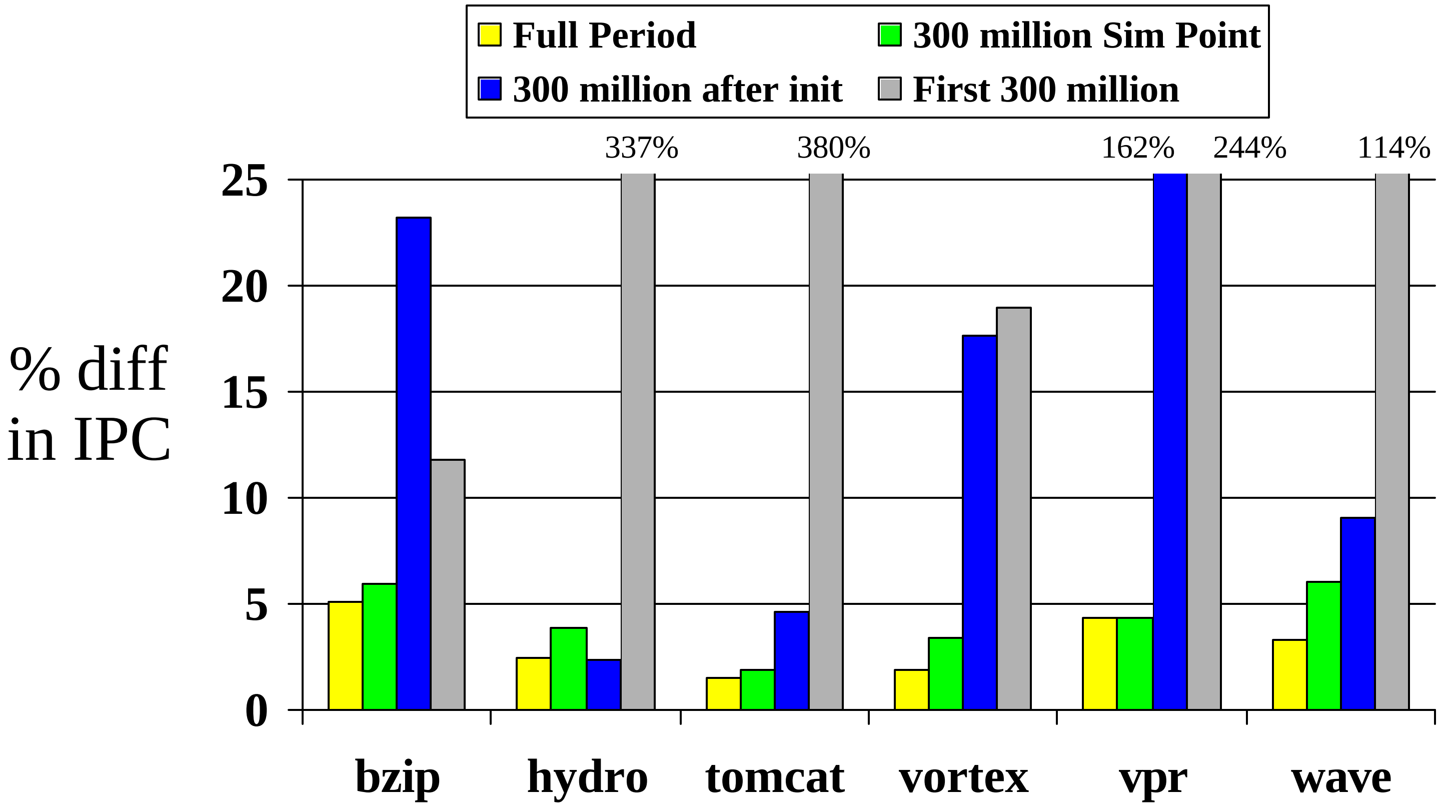
Outline

- Basic Block Distribution Analysis
- Initialization Phase
- Period
- Where to Simulate
- Conclusion

Where to simulate

- Not always possible to simulate full period
- Basic Block Distribution Analysis generates best simulation point for desired simulation duration
  - User inputs desired simulation duration
  - BB Distribution Analysis generates a BB Difference Graph with BB vector length equal to sim duration
  - Take min point in BB Difference Graph
    - Start simulation at that point

Accuracy of Simulation Points



University of California San Diego

Simulation Point Tool

- Input:
  - Program BB execution history
    - BB vector for every execution interval
  - Desired simulation duration
- Output:
  - End of initialization phase
  - Length of 1 period
  - Best simulation point

Key Points

- Focused on continuous simulation
- Basic Block approach is metric independent and correlates to program behavior
- Program behavior varies during execution
- Beneficial to find the best simulation point
- Not necessary to simulate full cycle for a good sample of overall program behavior

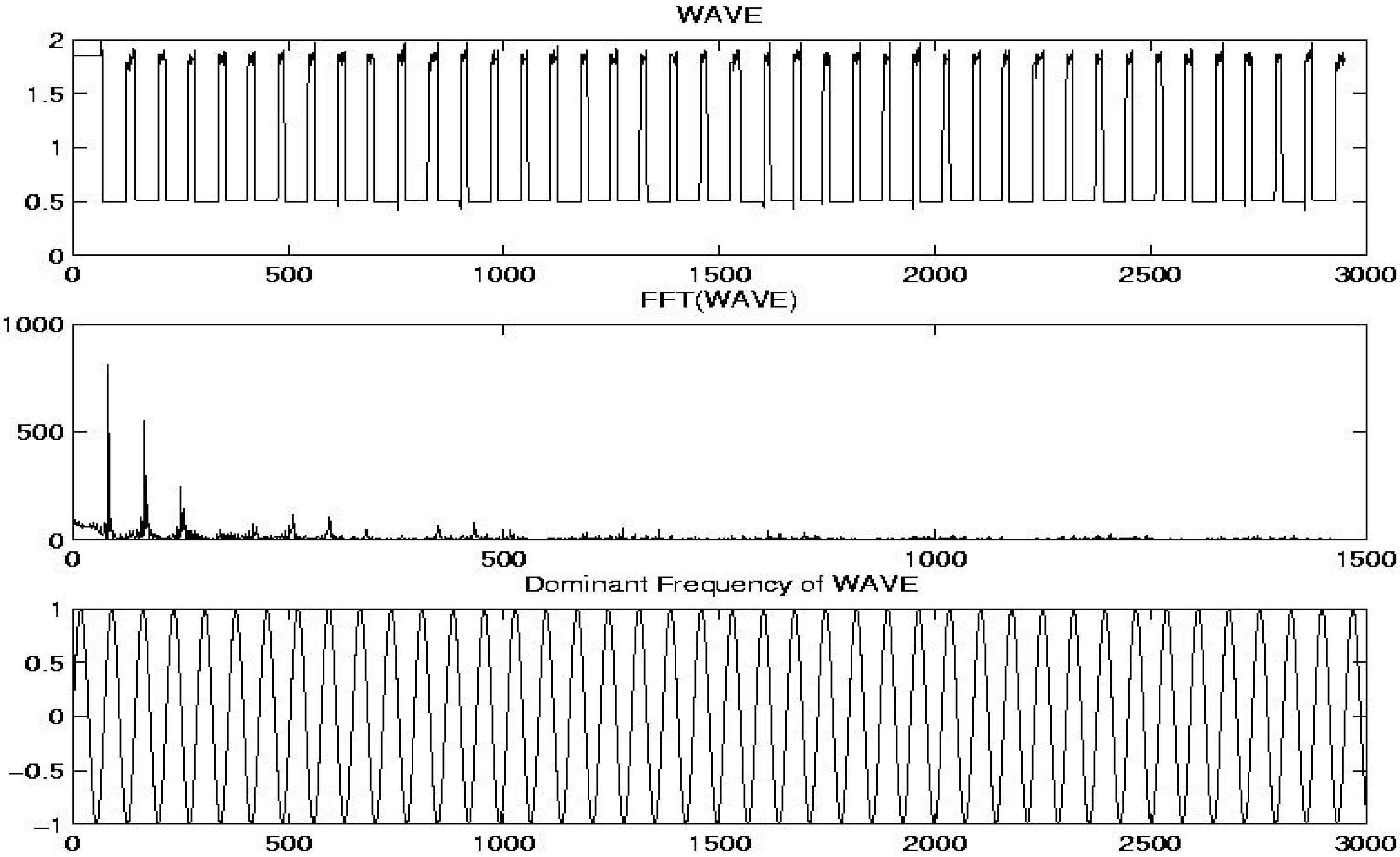
Conclusions

- BBDA is an effective method to find the initialization phase, period, and where to simulate programs
- BBDA is a time-conserving tool for researchers
- BBDA 300 million instructions simulation point produce average IPC error rates < 6%

Current Work

- Period with Fourier Analysis
  - Fast Fourier Transform
  - Breaks down signal into dominant frequencies
  - Period derived from dominant frequency
- Benefits
  - Multiple periods throughout execution

Fourier Analysis



University of California San Diego