A Study of Mobile Device Utilizations

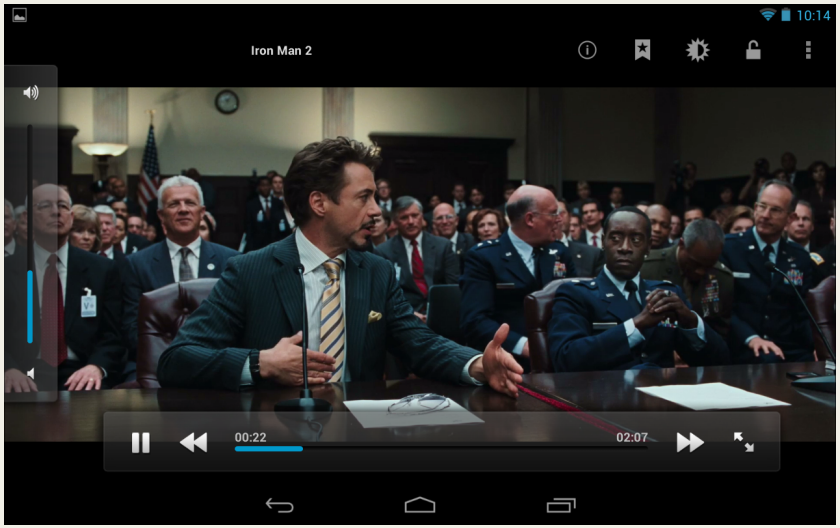
Cao Gao*, Anthony Gutierrez*, Madhav Rajan†,
Ronald G. Dreslinski*, Trevor Mudge*, and Carole-Jean Wu†

University of Michigan – Ann Arbor*

Arizona State University †

Introduction

- Mobile devices are taking over the function of desktops
  - Web browser, games, high-definition videos, etc.
- Driving force in building more powerful hardware



[1] Source: http://www.dataentryconsultants.com/website-services/mobile-website/

Introduction

- "More powerful hardware = More cores!"

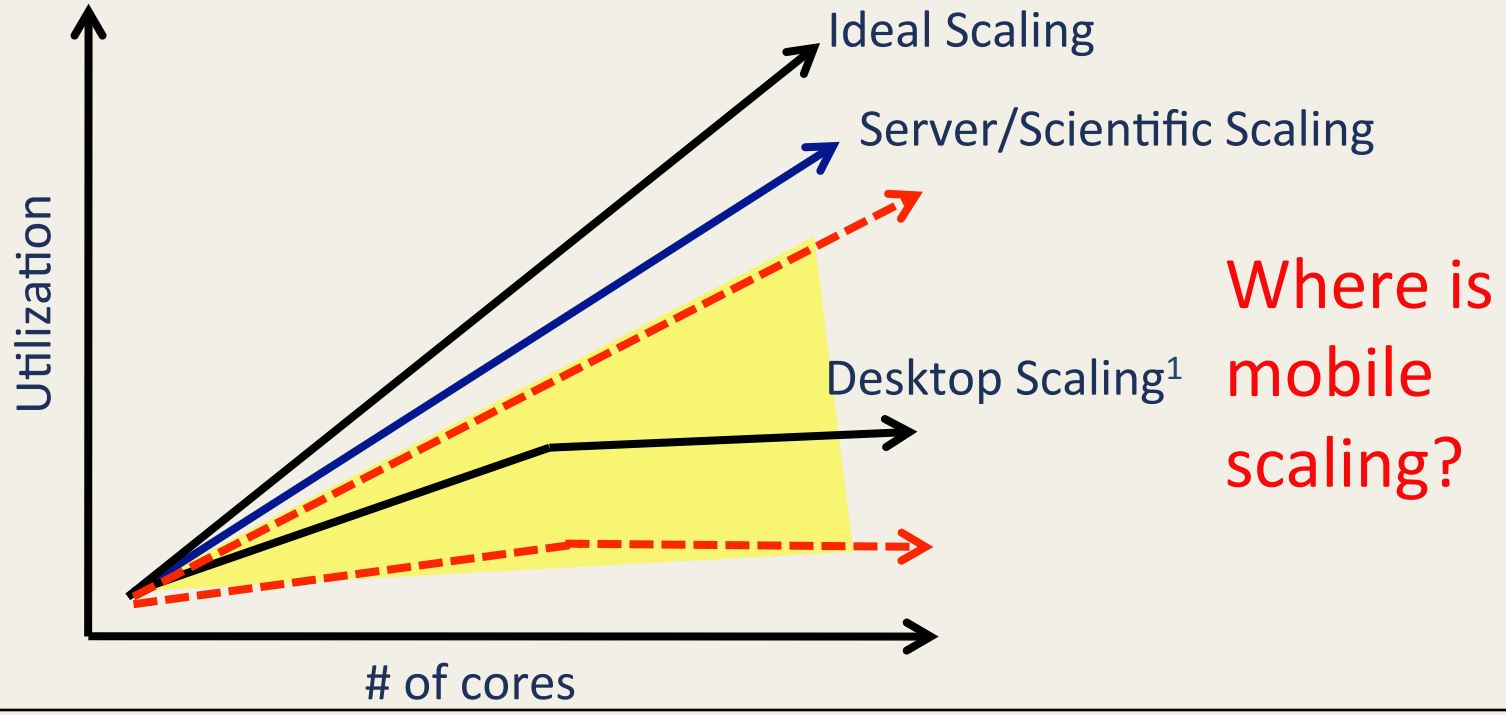
Dual-core `11 | Quad-core `12 | Octa-core `13

- However dual core designs still exist…
  - iPhone 6 (Sept. 2014) has a dual-core Apple-A8
  - Dual core Android smartphones are still on the high-end market

How are quad- and octa- cores utilized?

Utilizing multi-core is hard

- Server and scientific workloads already parallel
- But utilization of desktop/laptop workloads is low
- How about mobile applications?
  - Better idea for provisioning future hardware
  - Give insights to mobile application programmer



Reference: [1]Blake et al. ISCA'10

Contribution of this work

- Construct a representative mobile benchmark suite
- Evaluate the utilization of multi-core mobile devices
  - How utilized are they?
  - Why do we observe such utilization?
- Analyze some design issues and suggest suitable architecture for mobile devices

Outline

- Methodology
  - Metric
  - System setup
  - Benchmarks
- Results and Analysis
- Design Issues
- Conclusions

Metric

- **Measure *"Thread Level Parallelism"* (TLP)**
- CPU usage may underestimate concurrency due to idle time
- TLP = usage / percentage of non-idle time
  - *Number of cores being used; for a quad-core system, max = 4, min = 1*

- c_i = fraction of time **i** cpus are doing work
  - c_0 = Idle (no cpus are doing work)
  - c_1 = 1 cpu doing work, c_2 = 2 cpus doing work, etc

$$TLP = \frac{\sum_{i=1}^{n} c_i i}{1 - c_0}$$

TLP is a metric of concurrent execution on the non-idle portions of the application

Reference: Flautner et al. ASPLOS'00, Blake et al. ISCA`10

Systems setup

Board		Odroid XU+E *On-board sensor measuring CPU power*
SoC		Exynos 5410
CPU[1]	Big cluster	**Cortex-A15 Quad Core** @ 1.6GHz Out-of-order triple-issue 32KB Private L1 I/D cache, Shared 2MB L2 cache
	Little cluster	**Cortex-A7 Quad Core** @ 1.2GHz In-order dual-issue 32KB Private L1 I/D cache, Shared 512KB L2 cache
GPU		PowerVR SGX544MP3 (tri-core) @ 480MHz
Memory		2G DDR3 RAM
OS		Android 4.2.2 (Jelly Bean)

[1] can only use cores in one cluster at a time

Systems setup

Board		Odroid XU+E *On-board sensor measuring CPU power*
SoC		Exynos 5410
CPU[1]	Big cluster	**Cortex-A15 Quad Core** @ 1.6GHz Out-of-order triple-issue 32KB Private L1 I/D cache, Shared 2MB L2 cache
GPU		PowerVR SGX544MP3 (tri-core) @ 480MHz
Memory		2G DDR3 RAM
OS		Android 4.2.2 (Jelly Bean)

All experiments ran on a real, state of the art mobile platform

[1] can only use cores in one cluster at a time

Benchmarks

- 22 test applications in 10 categories:

• Web browser	• Music player
• Games	• Navigation
• Video player	• Image viewer
• Social networking	• Office productivity
• Communication	• File browser

- Results fully reproducible, low variance
- Automated real test actions using *adb* command line and RERAN[1]

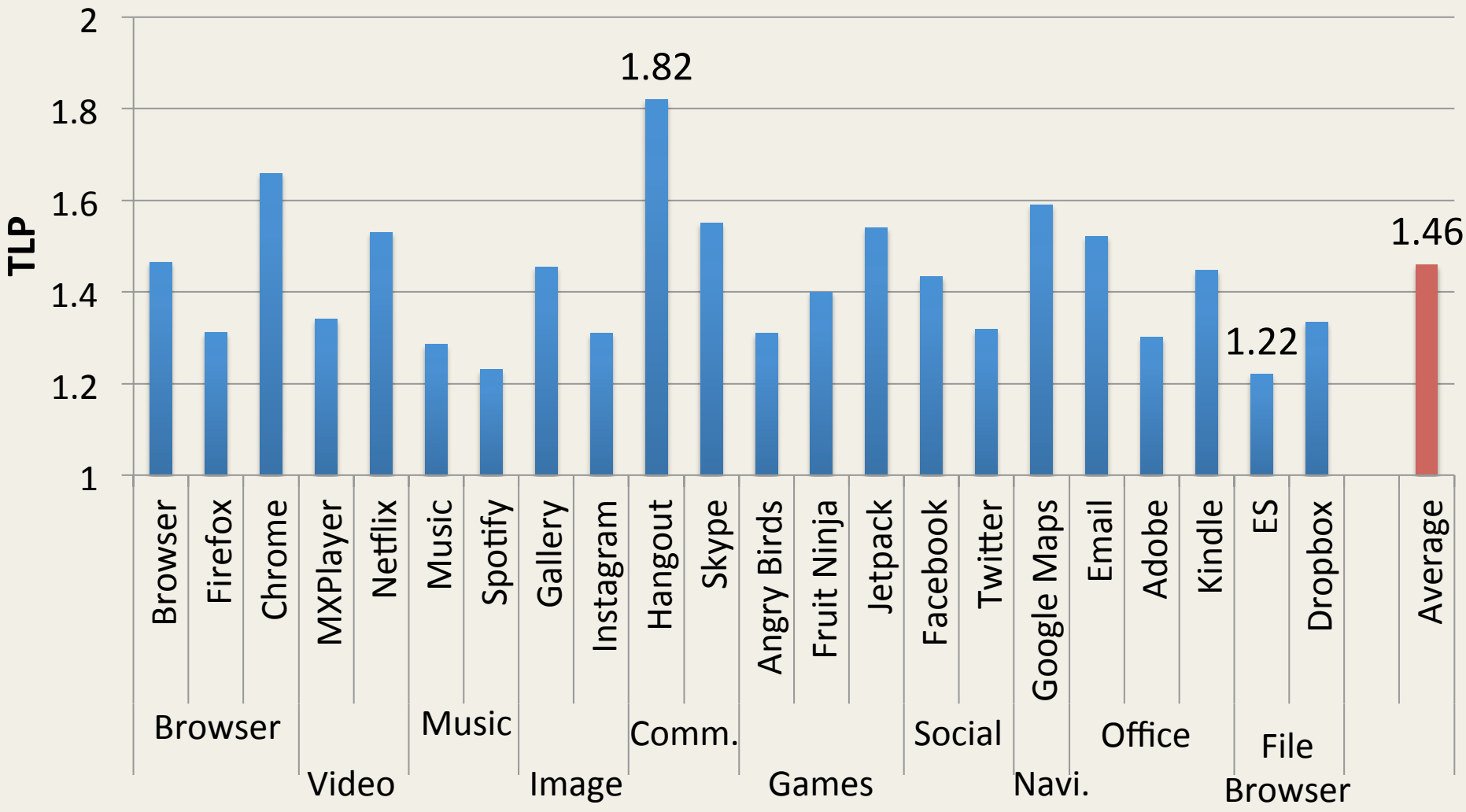
Reference: [1]Bohmer et al. MobileHCI'11

Outline

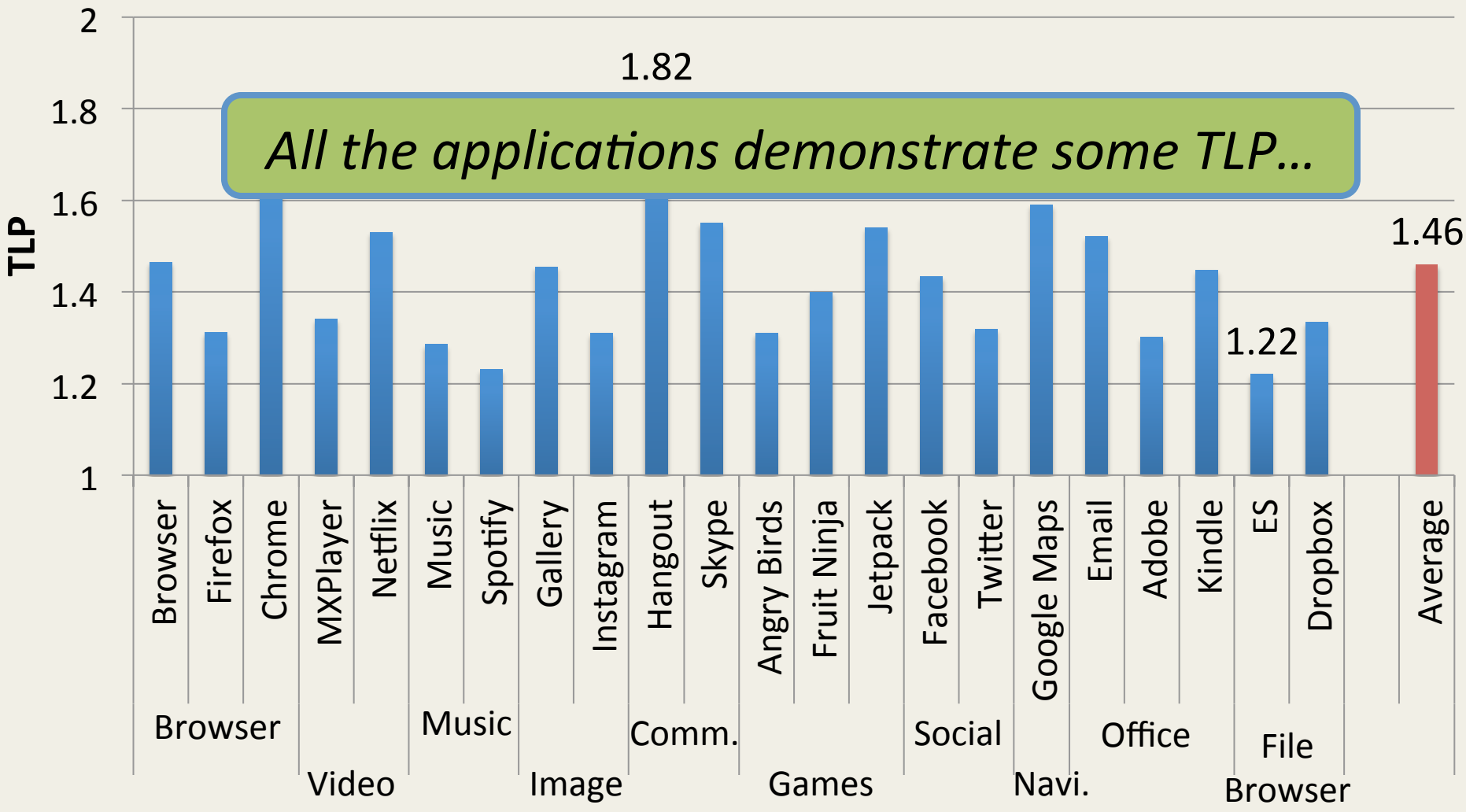
- Methodology
- Results and Analysis
  - How utilized are they?
    - Overall TLP
    - Core Scaling
  - Why do we observe such utilization?
    - Multi-tasking
    - GPU utilization
- Design Issues
- Conclusions

11

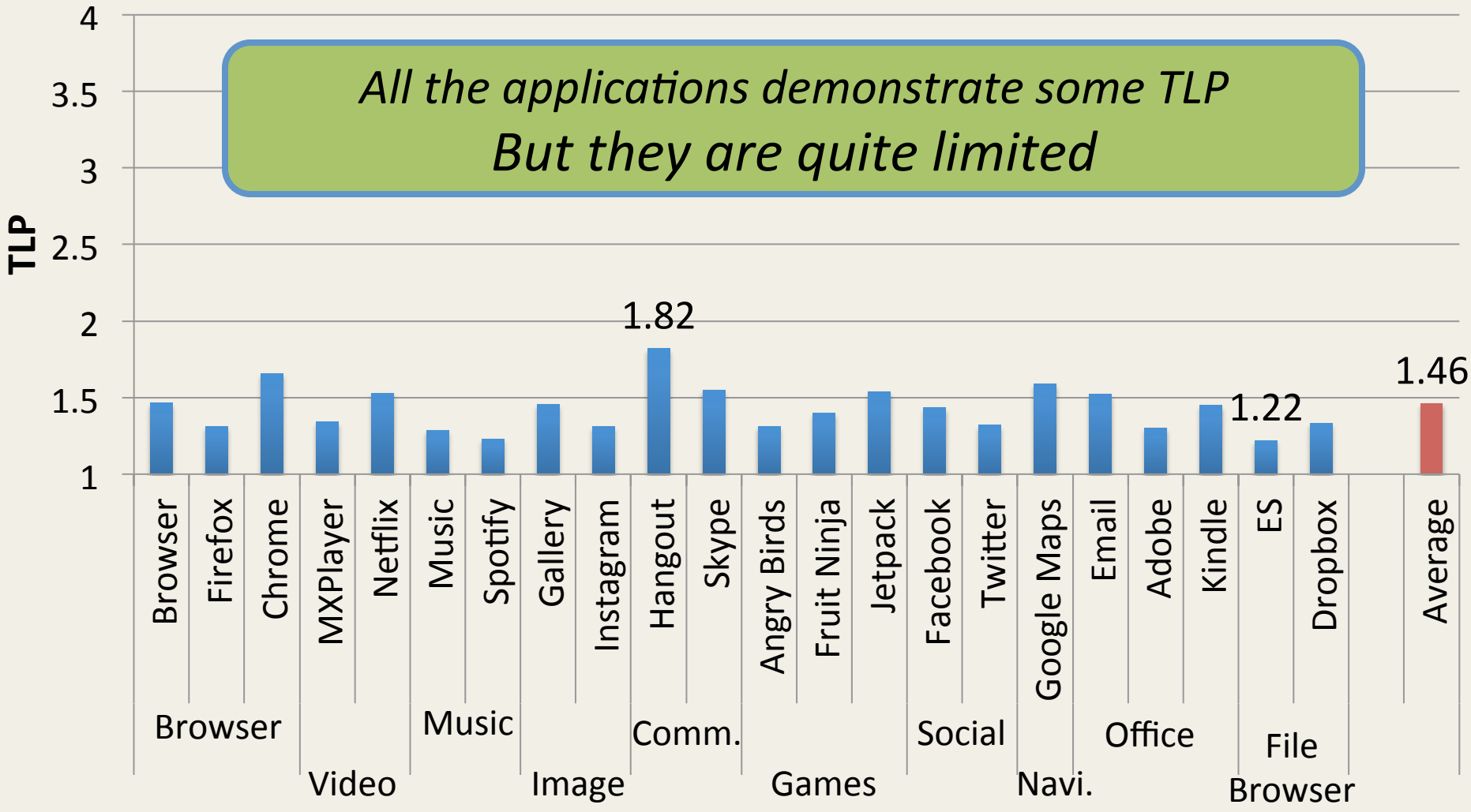
How much utilization in average?



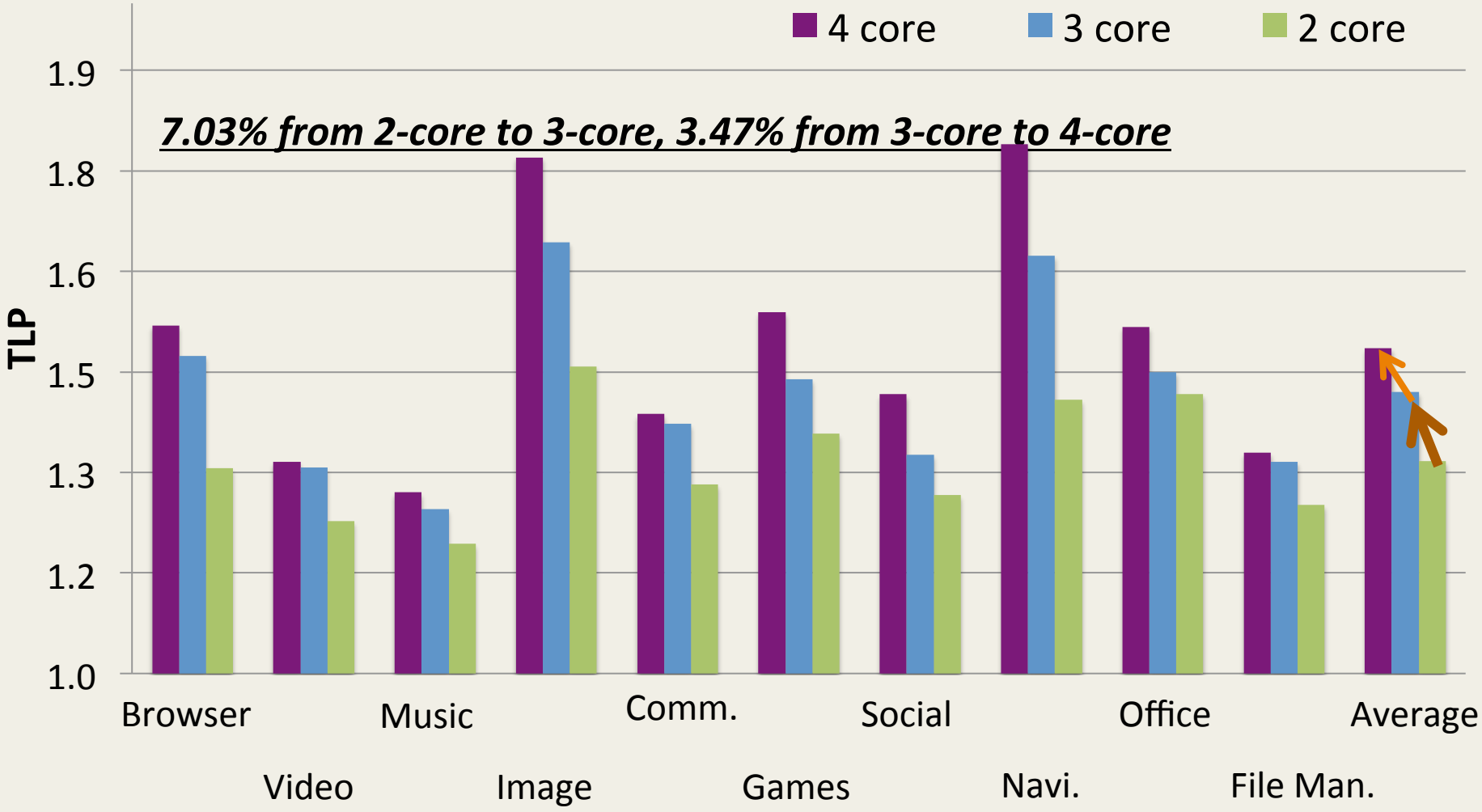
How much utilization in average?



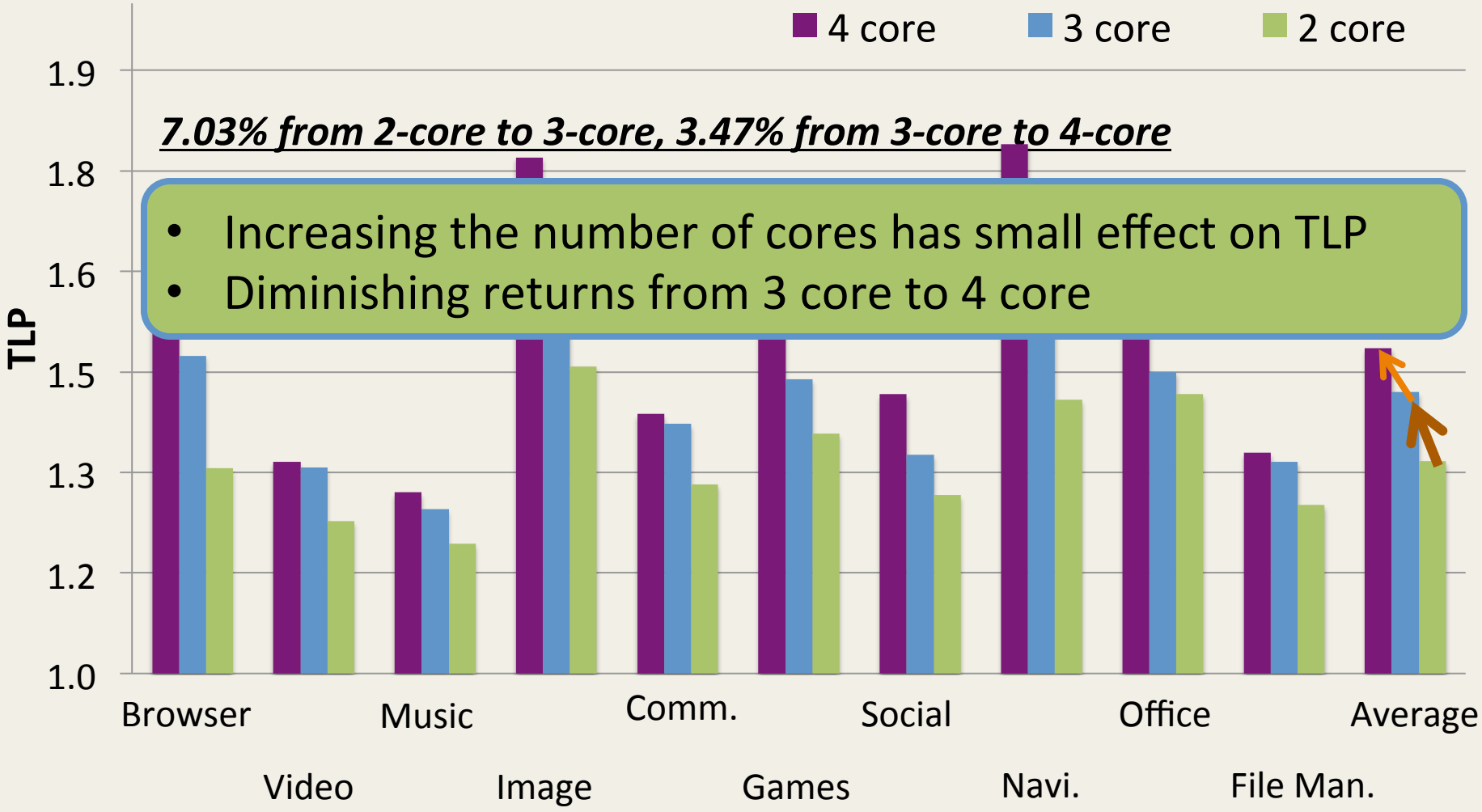
How much utilization in average?



How much more TLP we get by adding cores?



How much more TLP we get by adding cores?



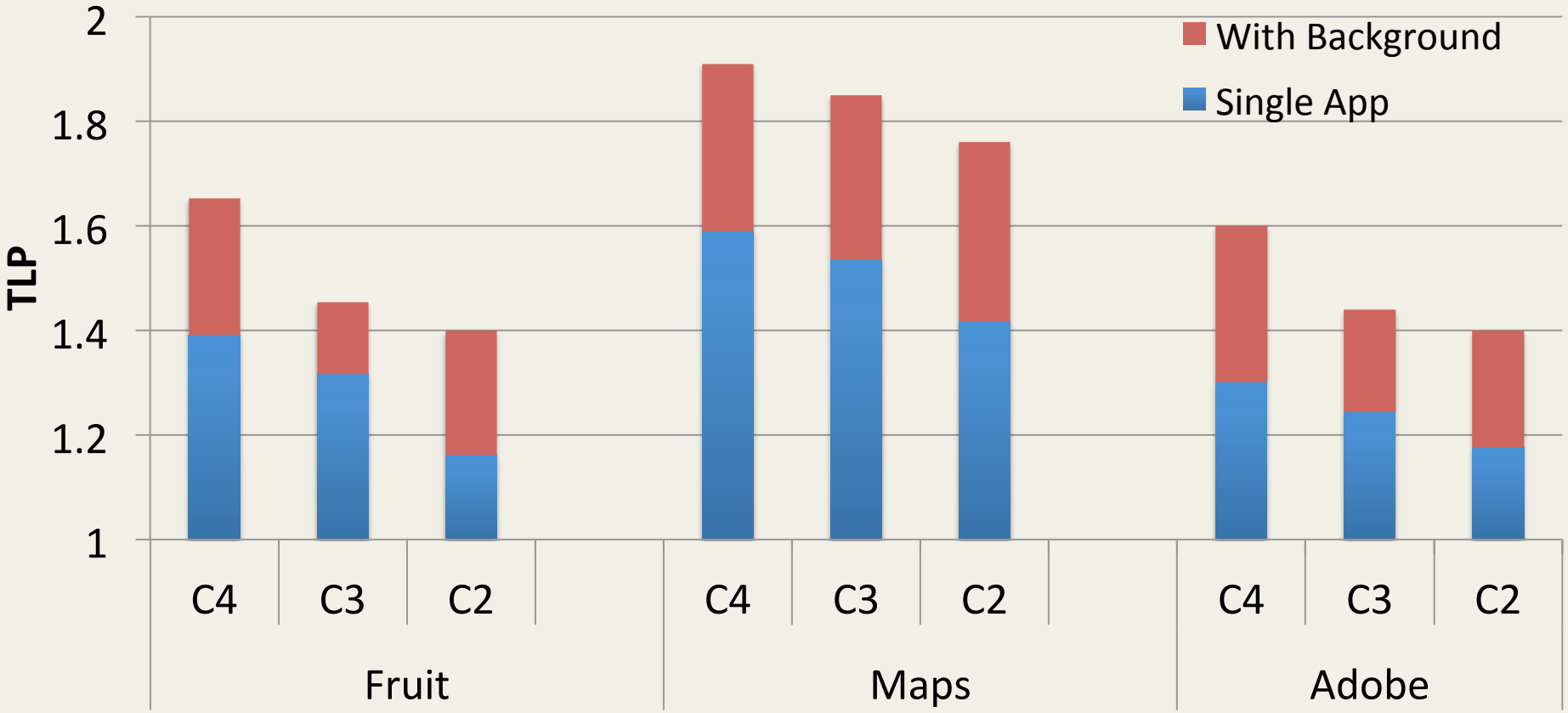
Outline

- Methodology
- Results and Analysis
  - How utilized are they?
    - Overall TLP
    - Core Scaling
  - Why do we observe such utilization?
    - Multi-tasking
    - GPU utilization
- Suggestions
- Conclusions

17

How about multi-tasking?

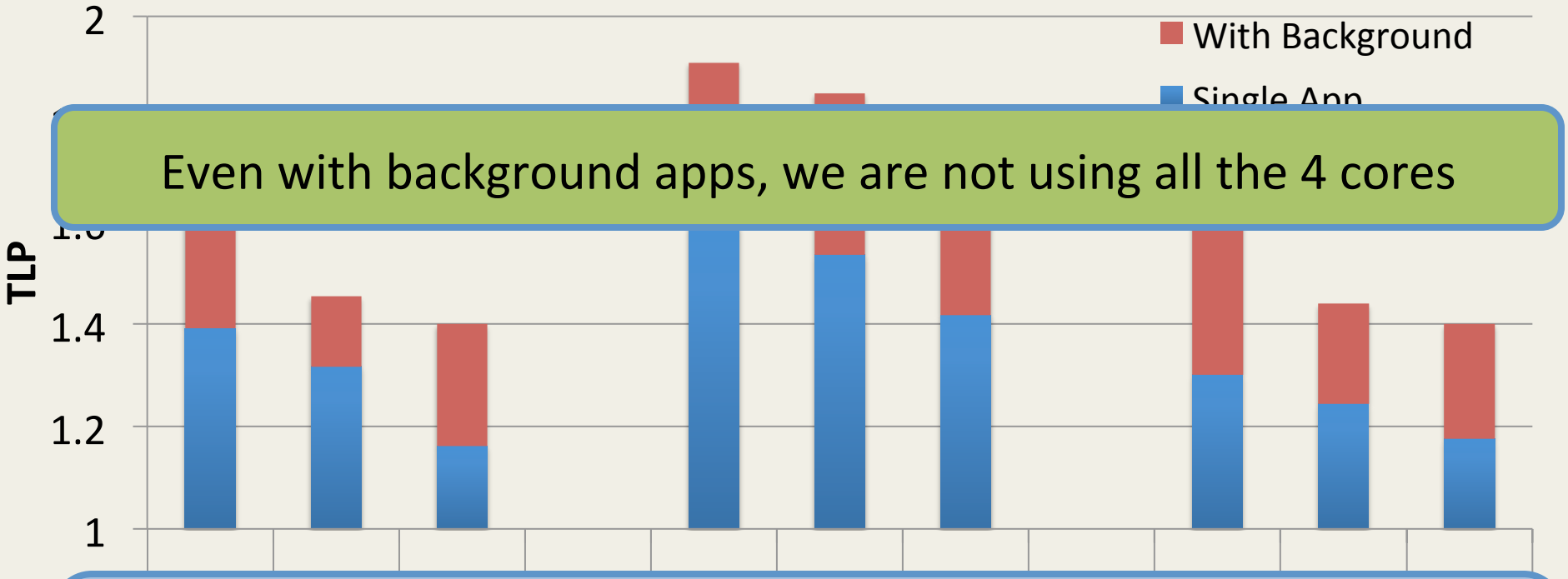
Background: Hangout, Spotify, Email



Reference: Falaki Mobisys`10

How about multi-tasking?

Background: Hangout, Spotify, Email

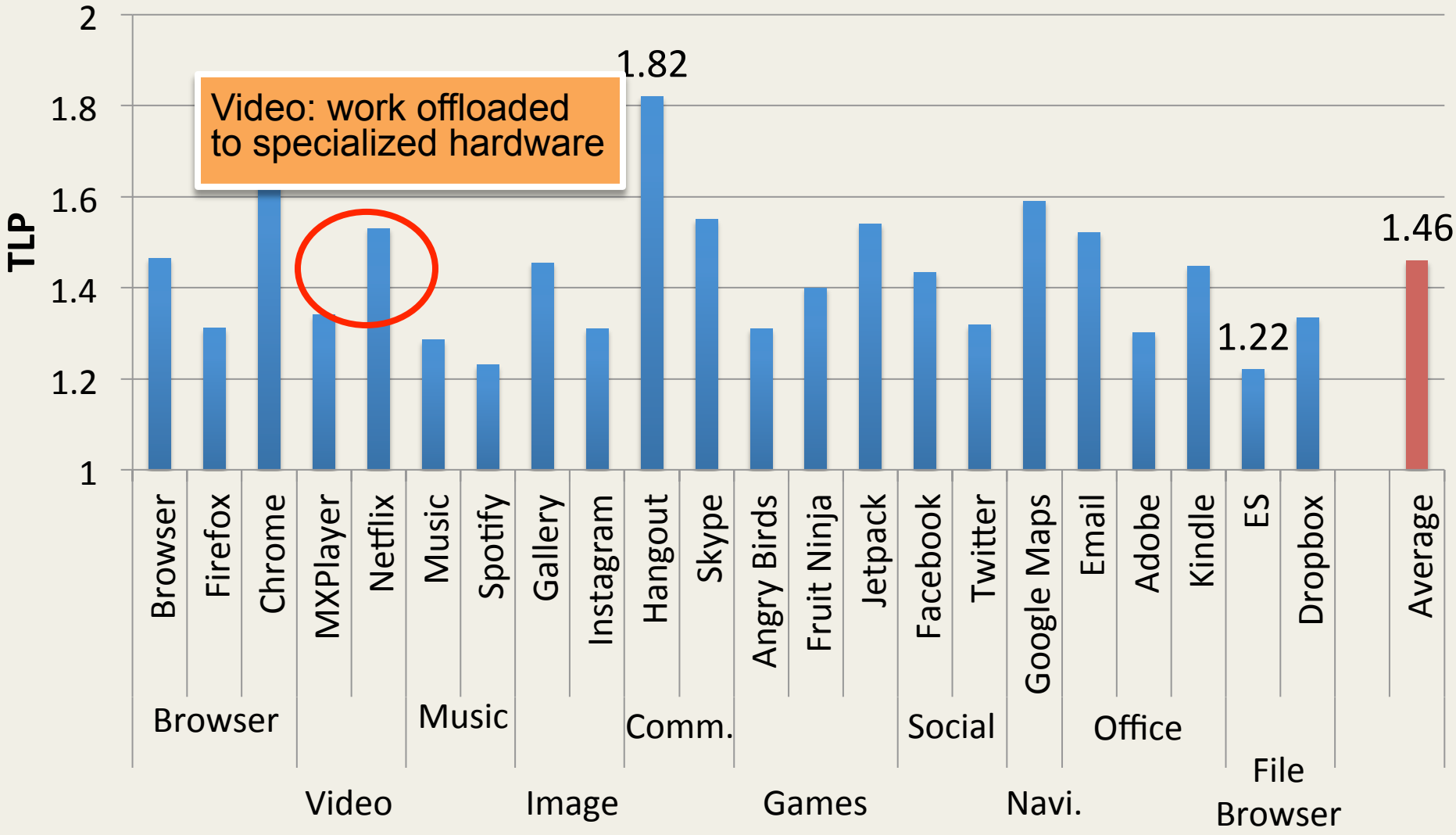


Even with background apps, we are not using all the 4 cores

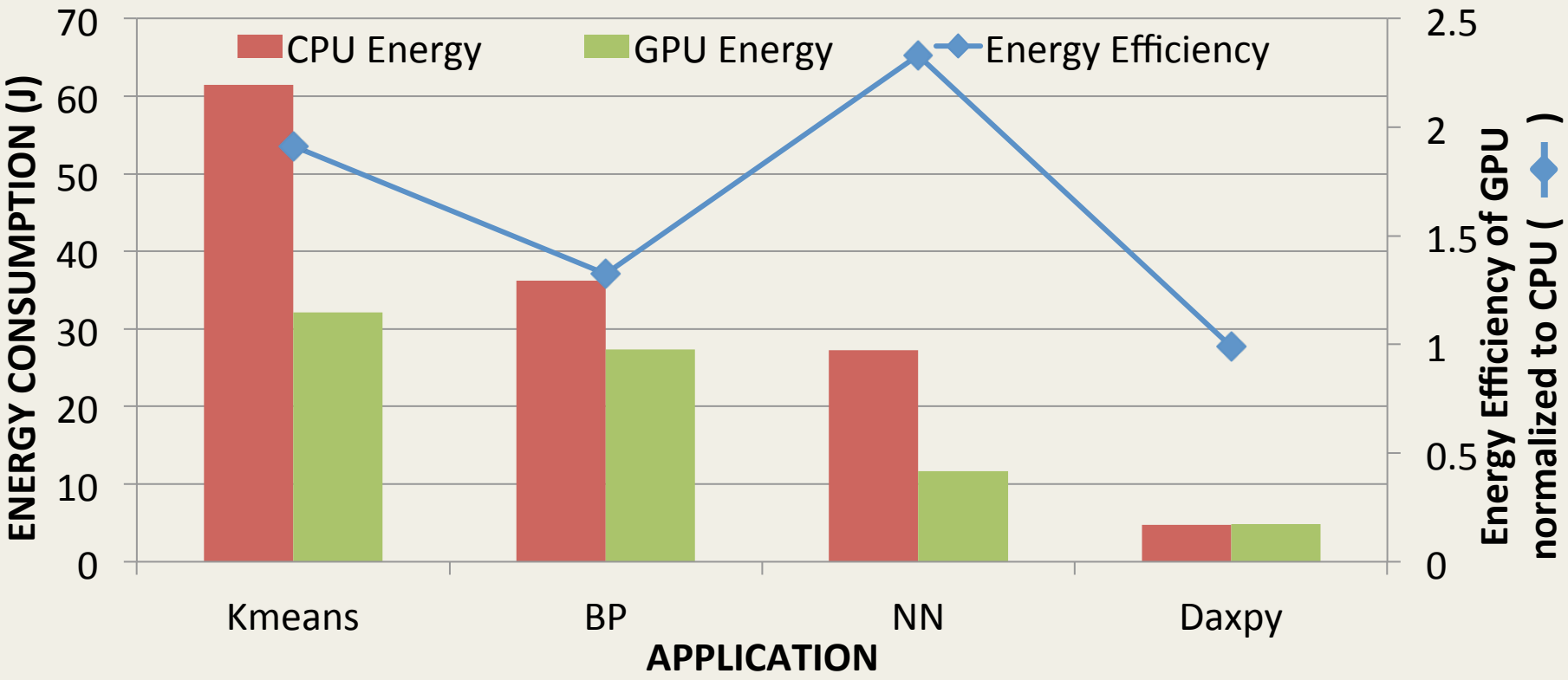
- Less computation involved on mobile devices than desktop
  - Small screen size, short usage duration, simpler task
  - *It may change in the future*

Reference: Falaki Mobisys`10

Reason 2: some work will be offloaded from CPU



GPU offloading



- Opportunities to offload computations to GPU for future mobile applications

21

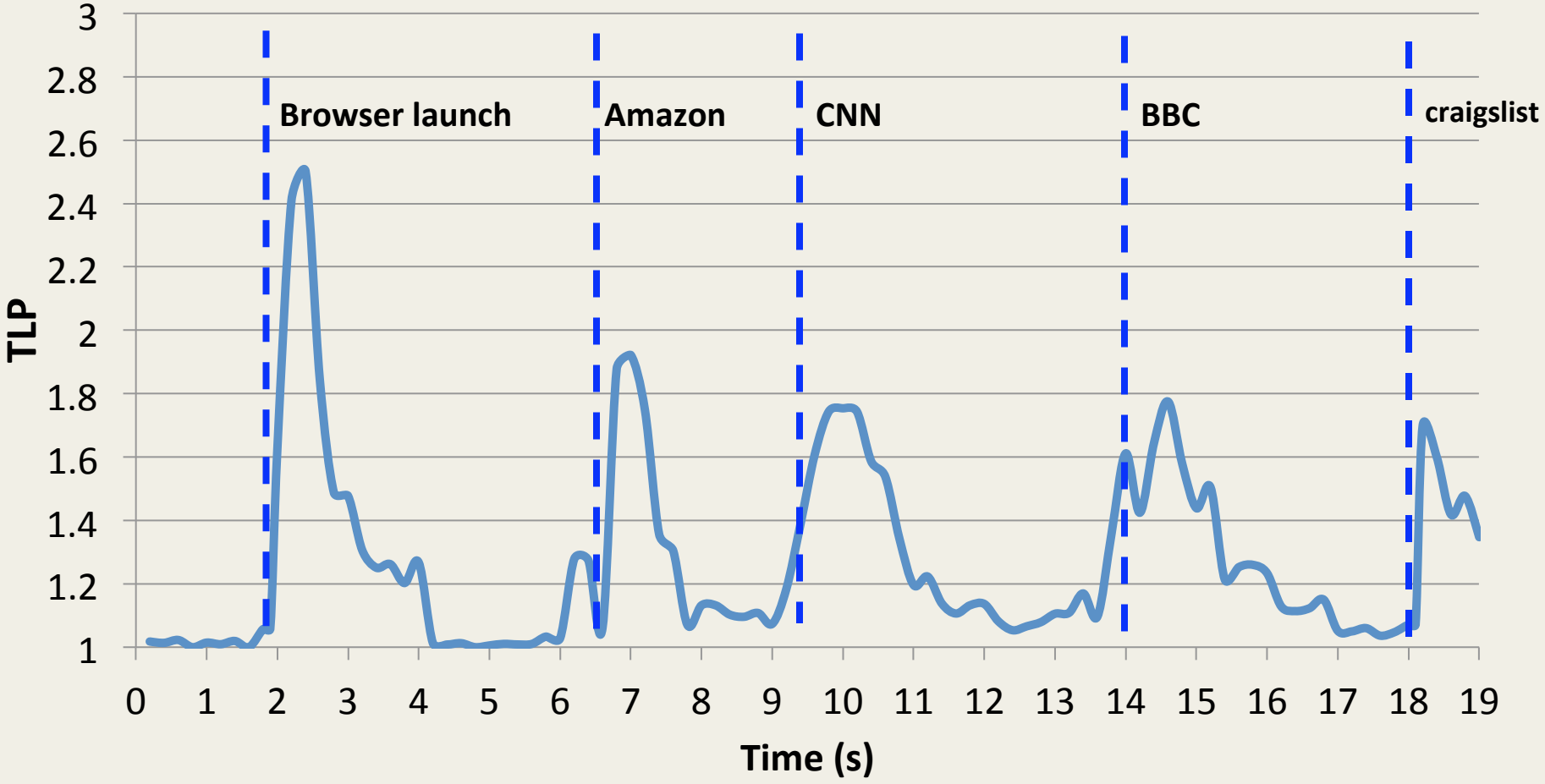
Outline

- Methodology
- Results and Analysis
- Design Issues
  - Handling time-variant TLP
  - Achieving energy efficiency
- Conclusions

22

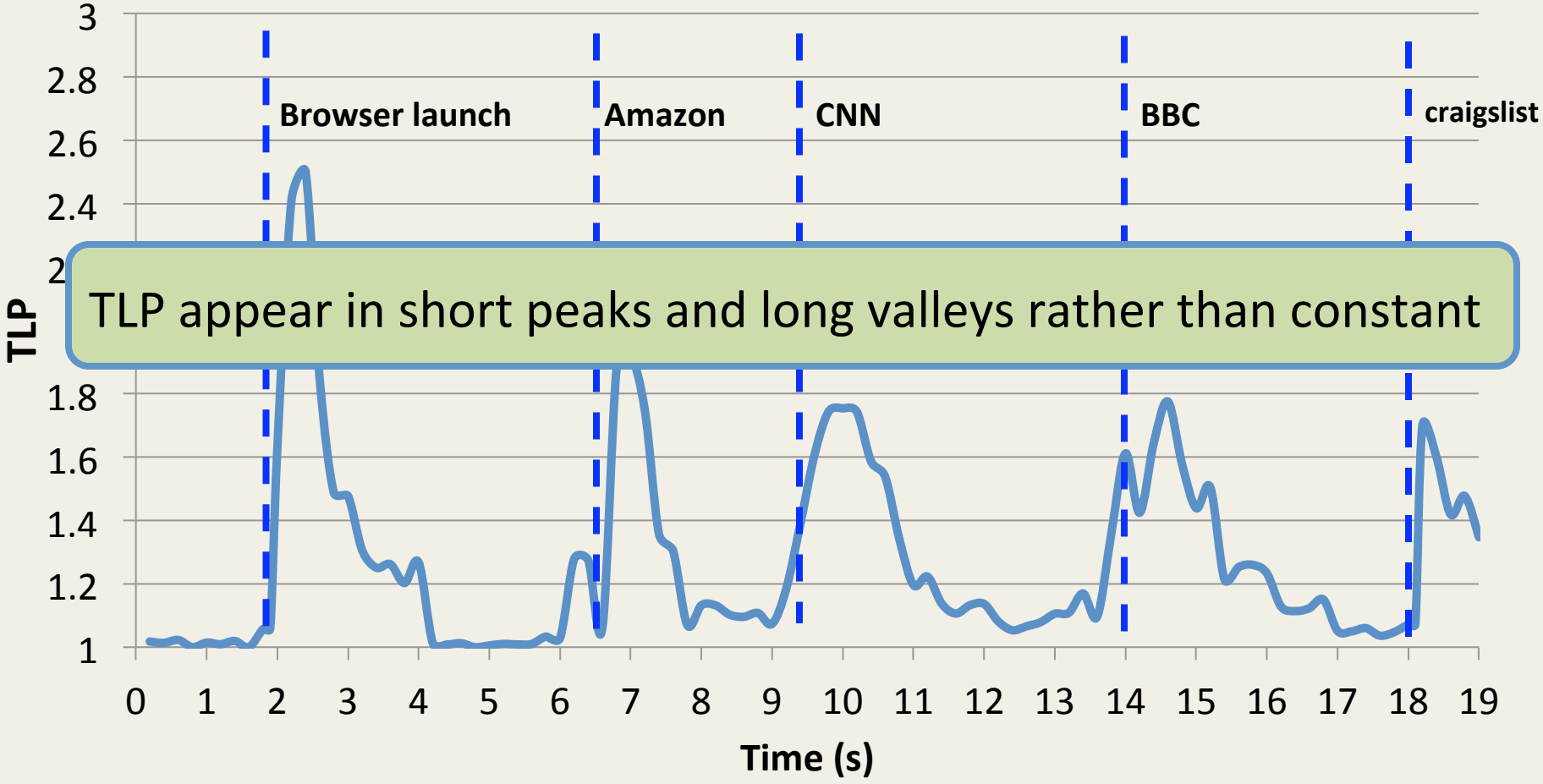
TLP over time

- Measure the first 20 seconds of stock browser and BBench launching

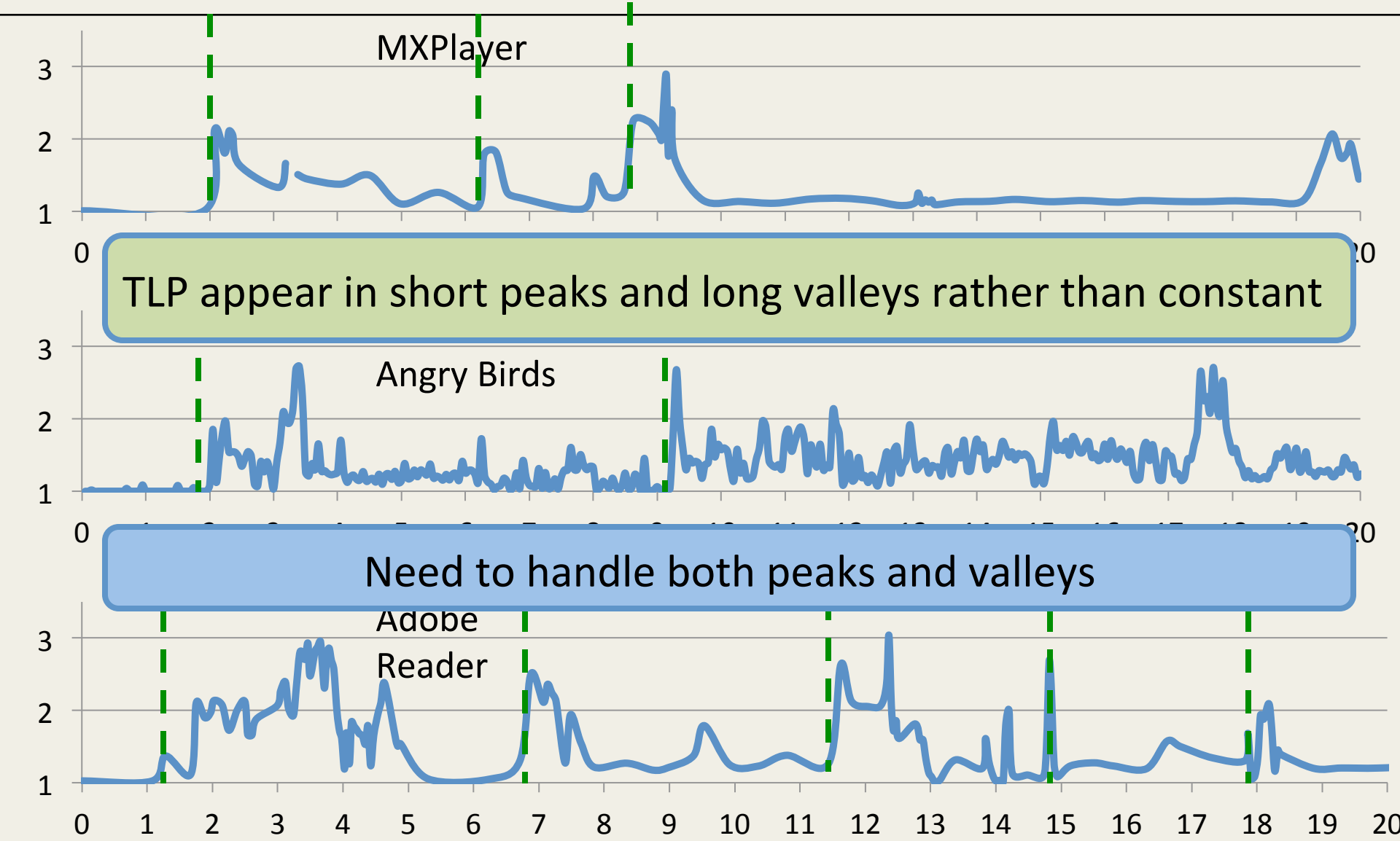


TLP over time

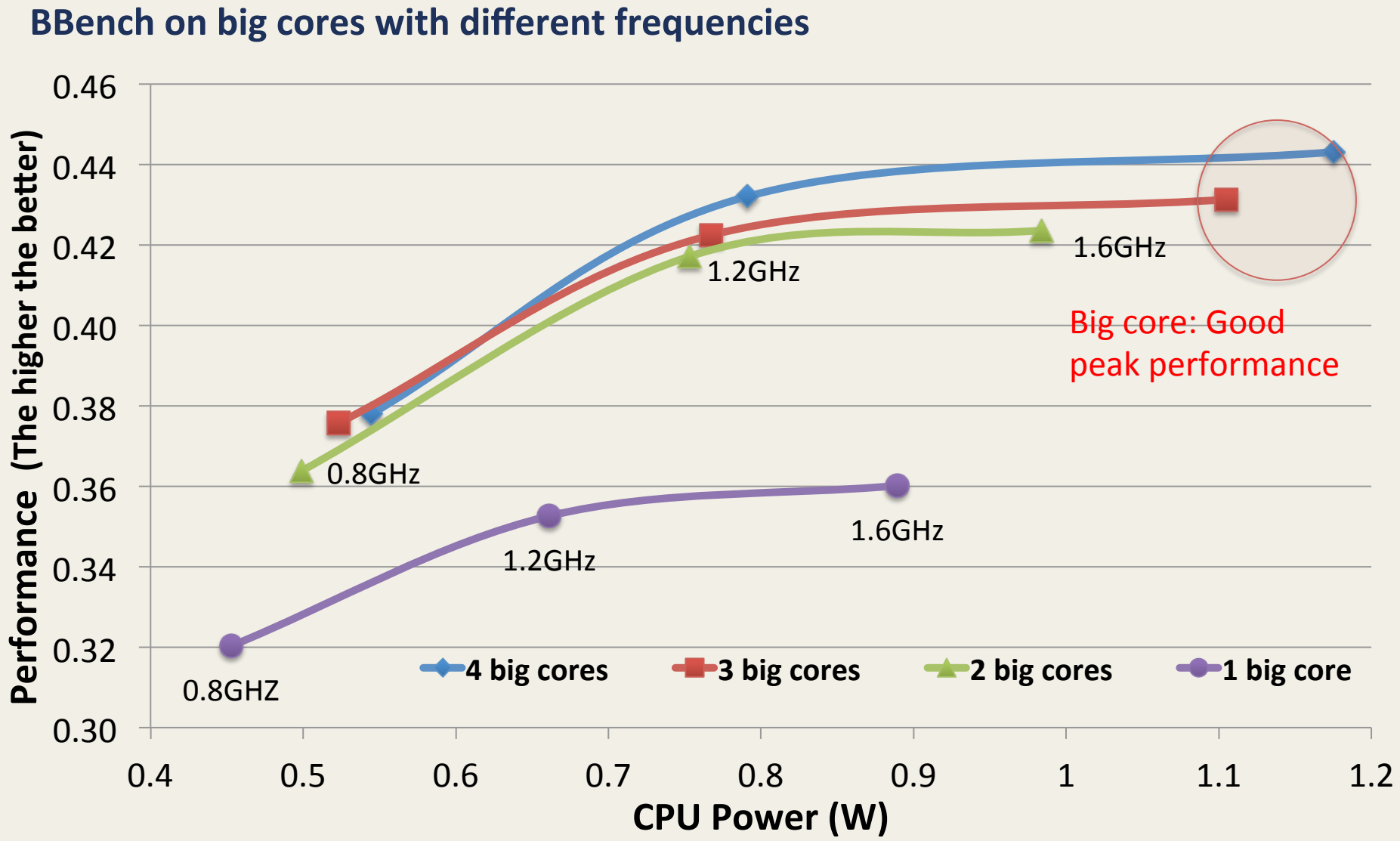
- Measure the first 20 seconds of stock browser and BBench launching



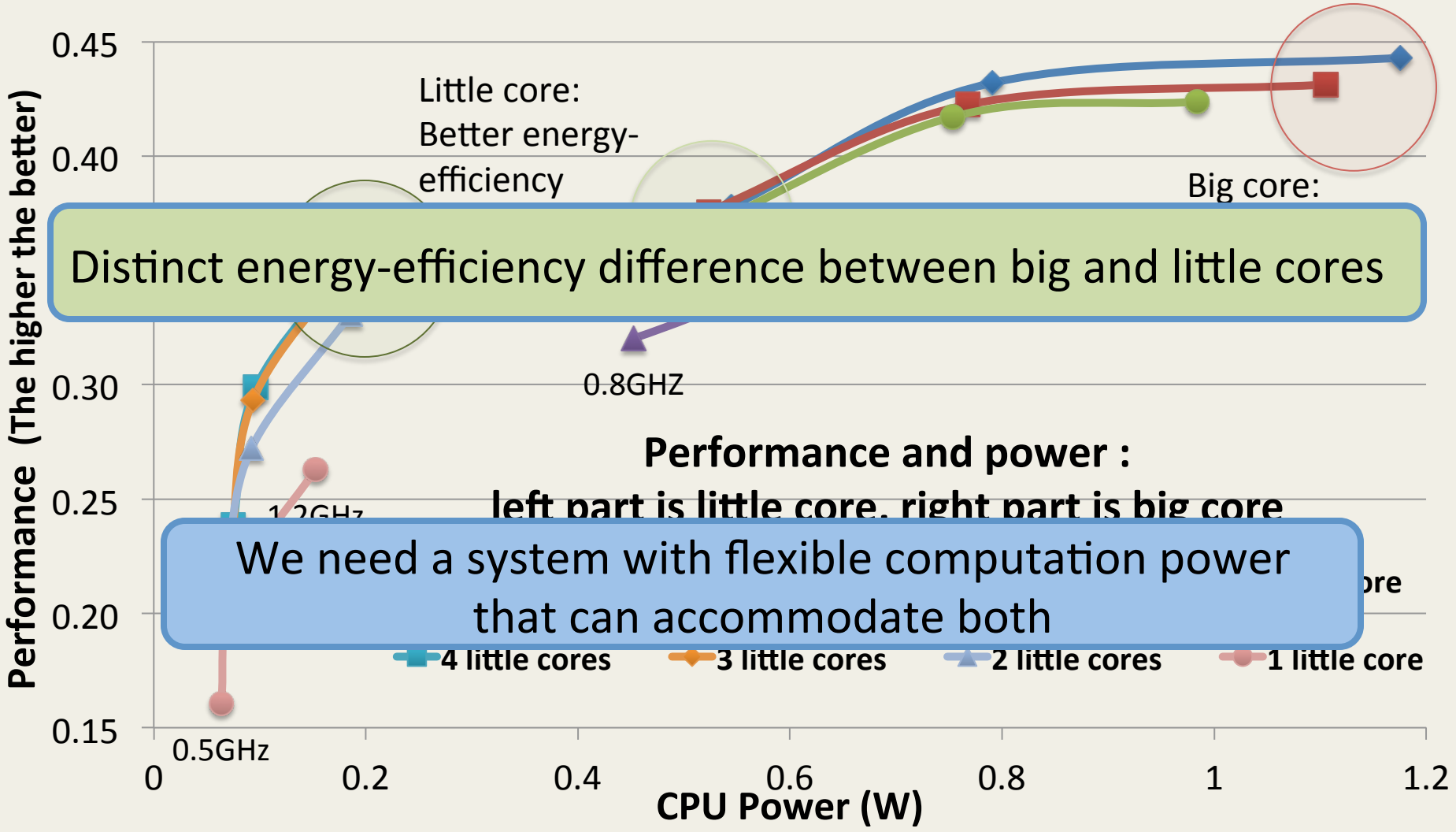
TLP over time



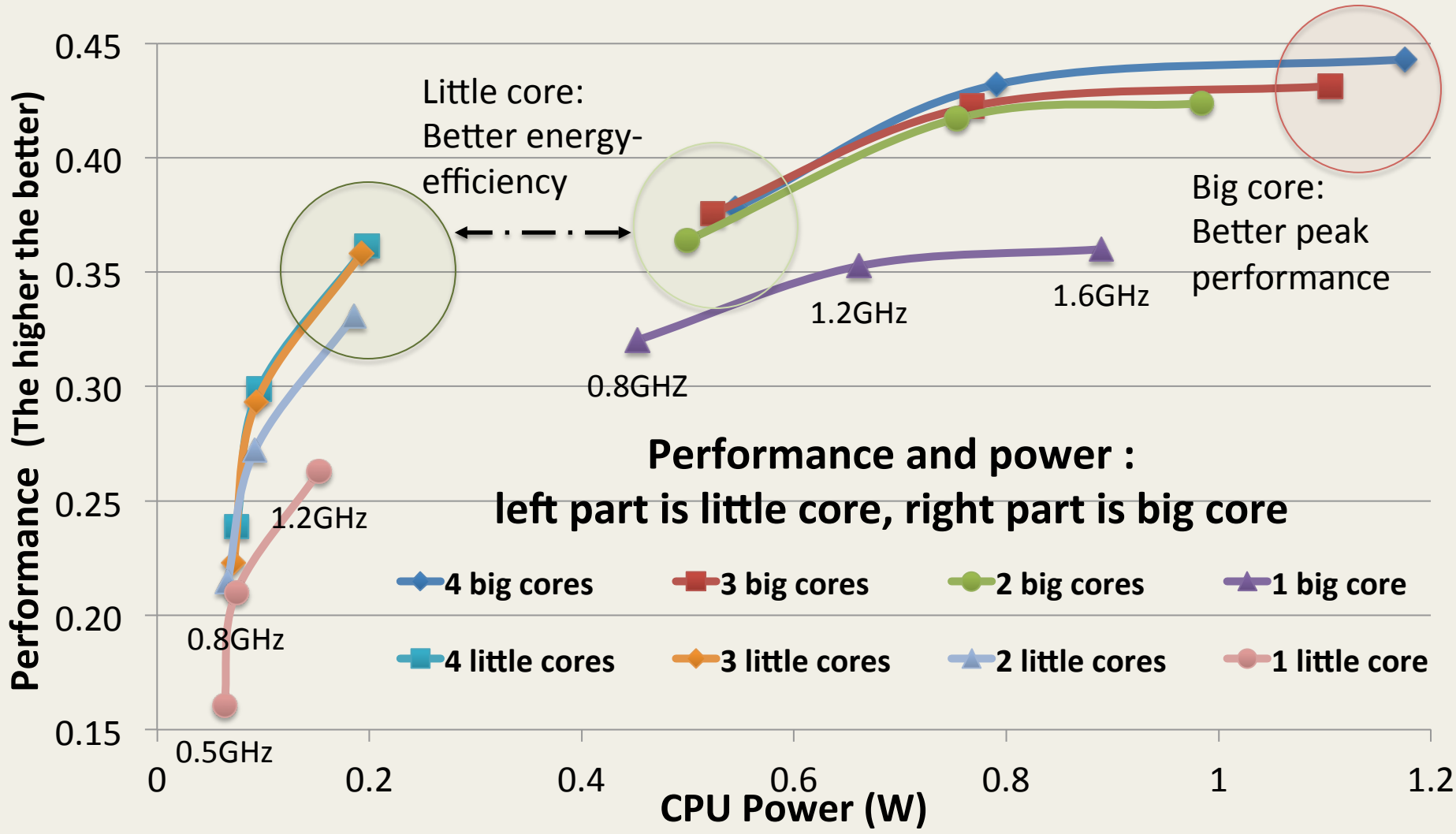
Big core can handle peaks



But little core is better at valleys



But little core is better at valleys



Outline

- Methodology
- Results and Analysis
- Design Issues
  - Handling time-variant TLP
    - need performance and energy-efficiency at different phases
  - Achieving energy efficiency
    - one core cannot satisfy both performance and energy-efficiency
  - Flexible system: high performance + good energy-efficiency
- Conclusions

29

Conclusions

- Current mobile applications cannot effectively utilize multi-core processors
  - Average TLP is low: 1.46
  - Diminishing return when adding cores
  - GPUs and ASICs further take away the parallelism that can be exploited

- Mobile applications can benefit from a flexible system for both high performance and energy-efficiency
  - TLP exhibit short peaks and long valleys rather than constant
  - Need a flexible system: high performance + good energy-efficiency

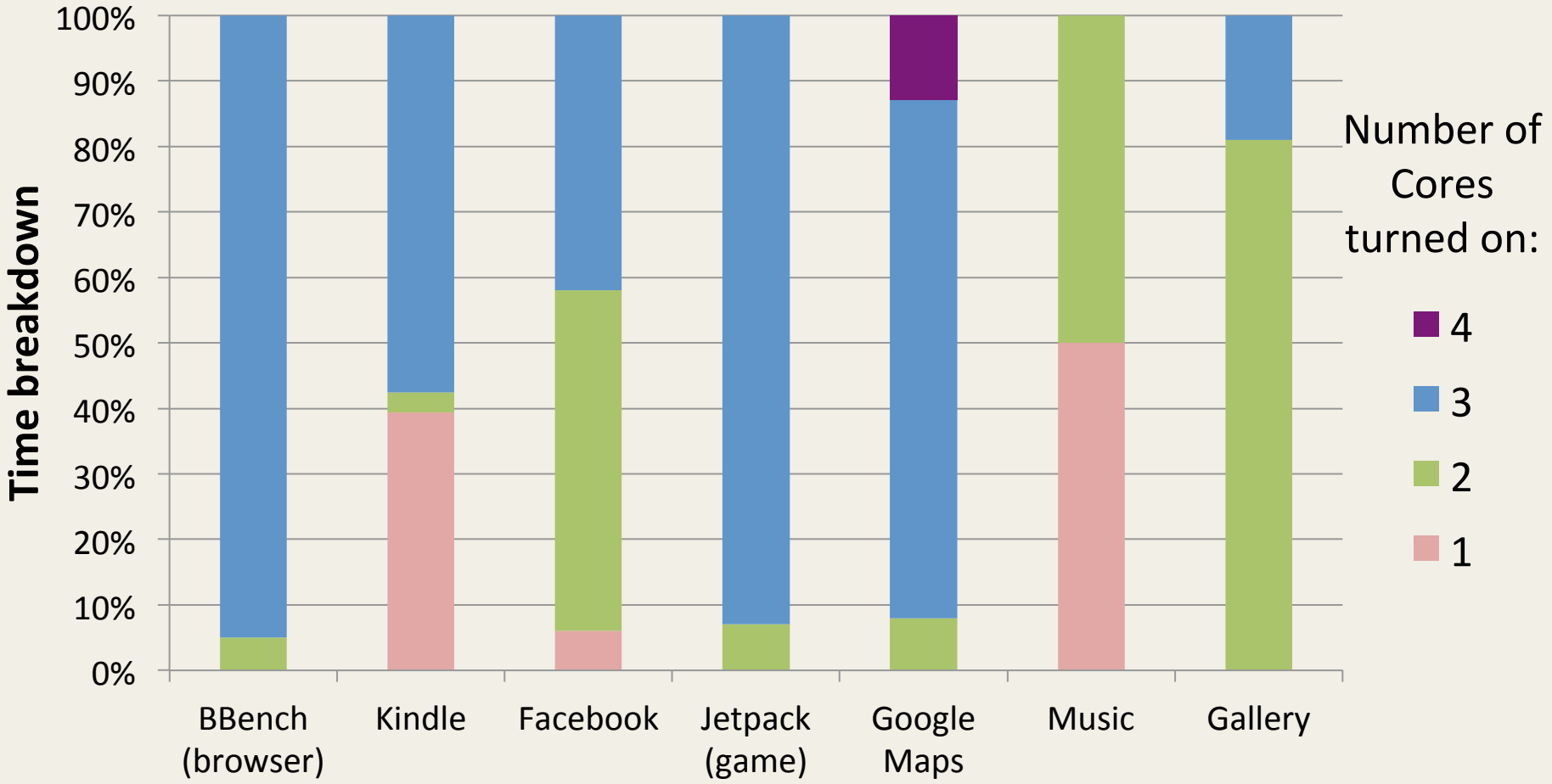
Thank you

Backup Slides

31

Motivation

Are quad cores being turned on at all?

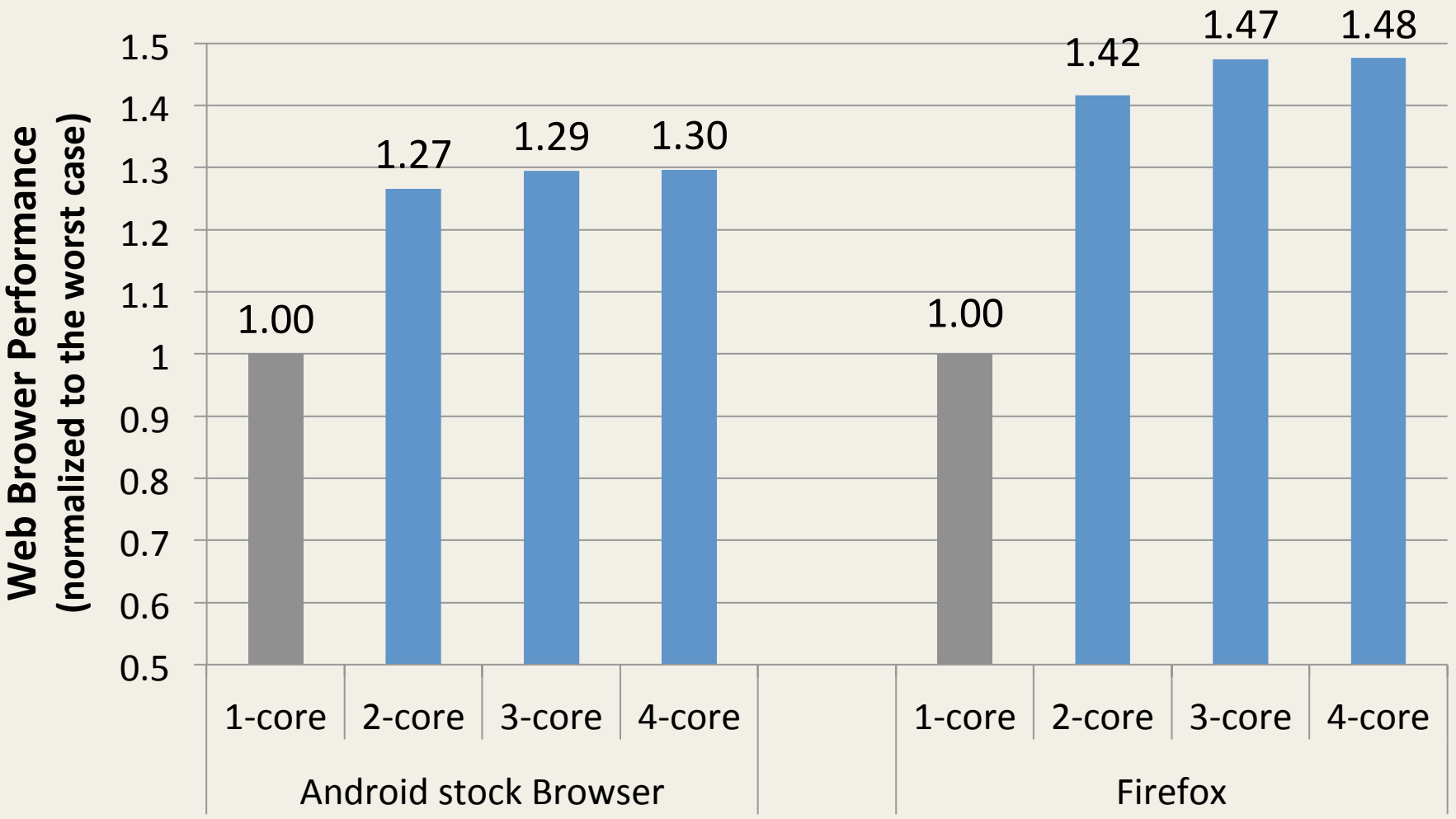


- *For most of the applications, the fourth core is always shut down, and most of the time the third core is not activated as well.*

32

Motivation

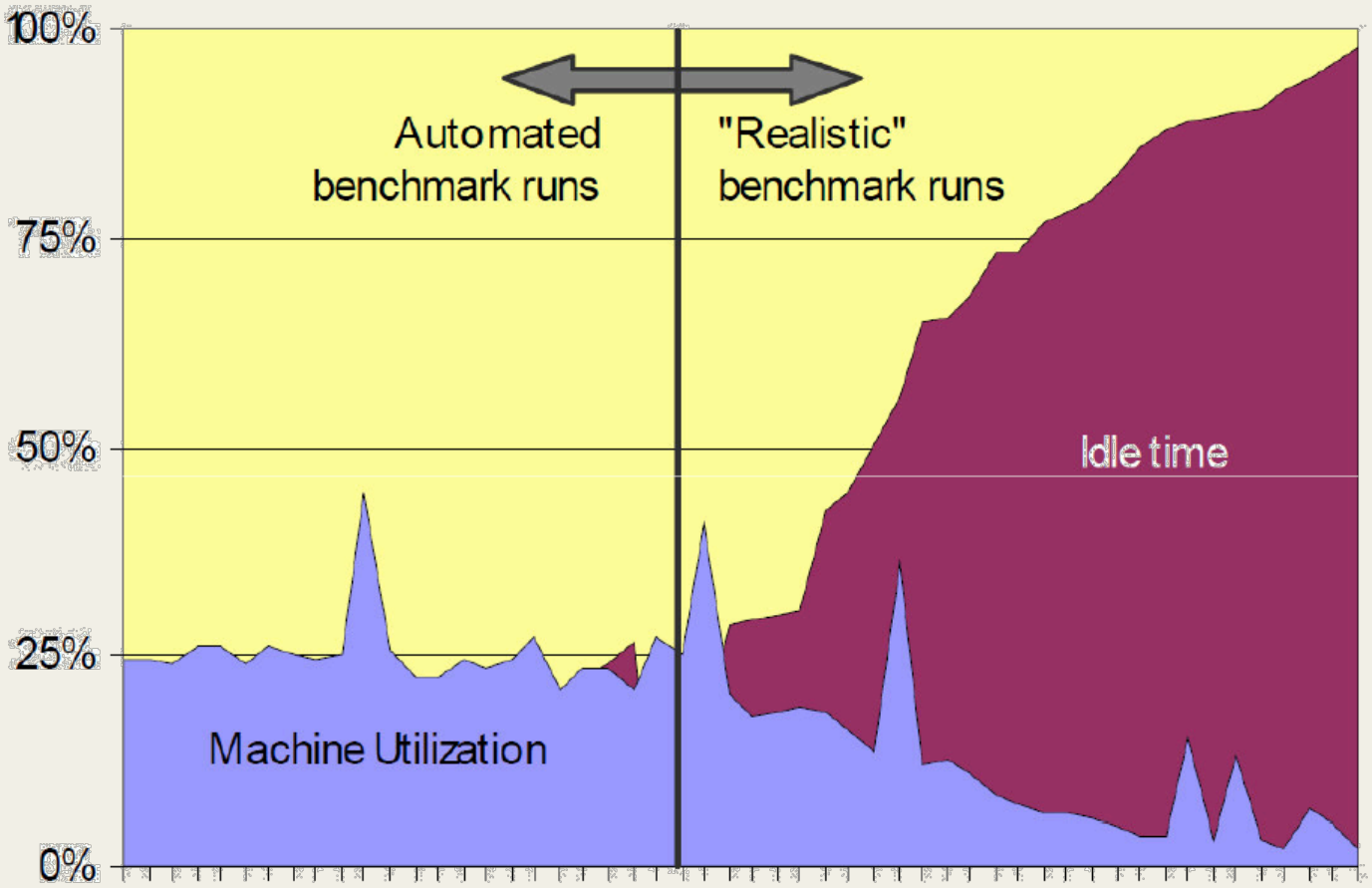
Are quad cores actually helpful?



- *Adding the 3rd and 4th core gives little performance improvement here*

Metric

- CPU utilization may underestimate concurrency due to idle time[1]



Reference: [1]Flautner et al. ASPLOS'00, Blake et al. ISCA`10

BBench

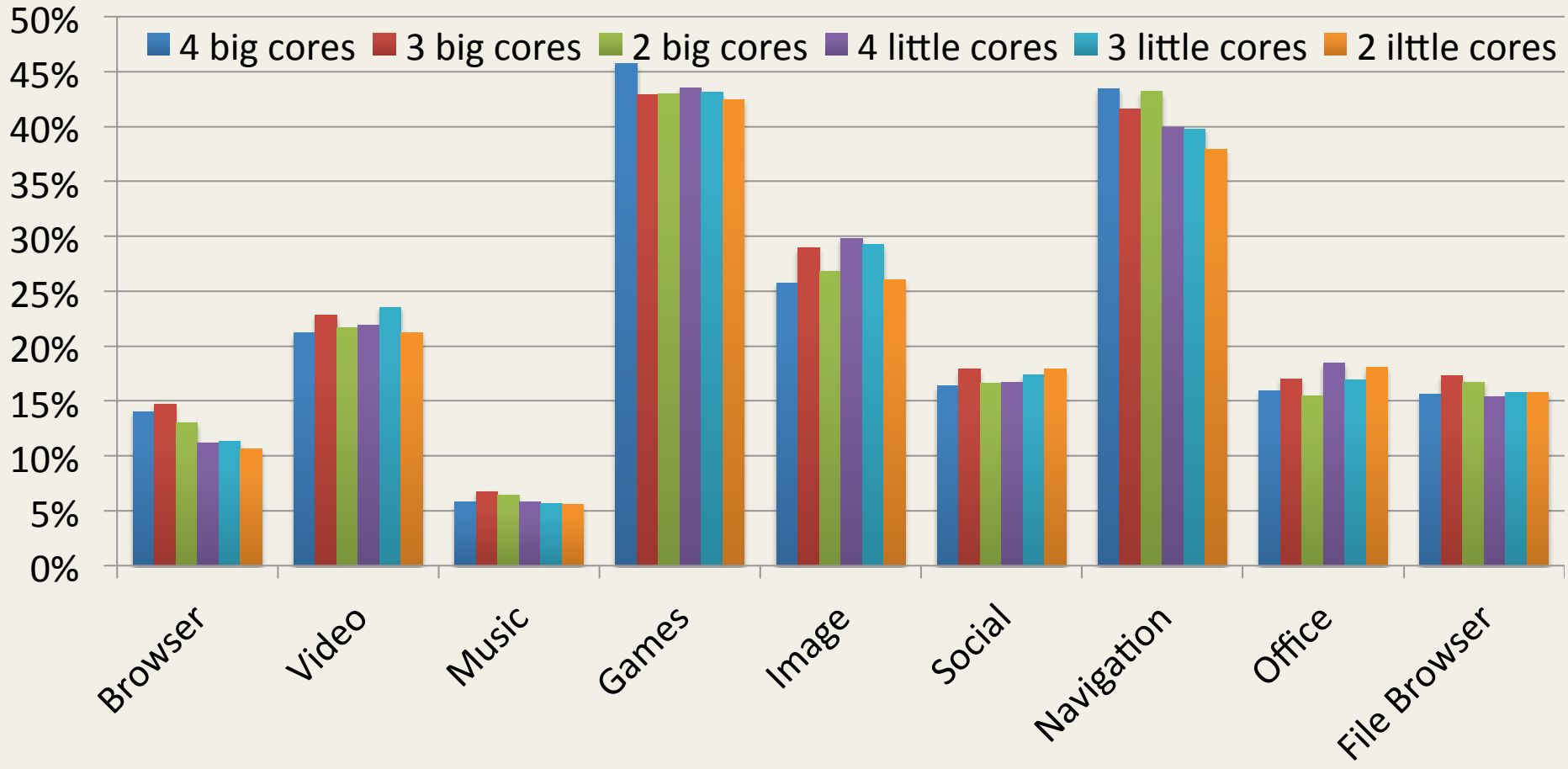
- BBench is an automatic web page rendering benchmark
  - 11 of the most popular web pages [Google]
  - Sites chosen to cover wide spectrum
  - JavaScript used to automate process

Results

Site Name	Cold Start Time	Avg Warm Page Rendering Time (ms)	Std Dev of Warm Runs	%Coeff Var of Warm Runs
amazon	2238	2458.00	34.00	1.38
bbc	3440	3225.50	99.50	3.08
cnn	3339	2885.50	223.50	7.75
craigslist	838	637.00	62.00	9.73
ebay	1344	1528.50	220.50	14.43
espn	2365	2469.00	124.00	5.02
google	751	657.00	28.00	4.26
msn	2975	2745.00	146.00	5.32
slashdot	3818	4311.50	314.50	7.29
twitter	2611	2410.50	278.50	11.55
youtube	2874	2460.00	191.00	7.76

Geometric Mean of Average Warm Runs	2039.22

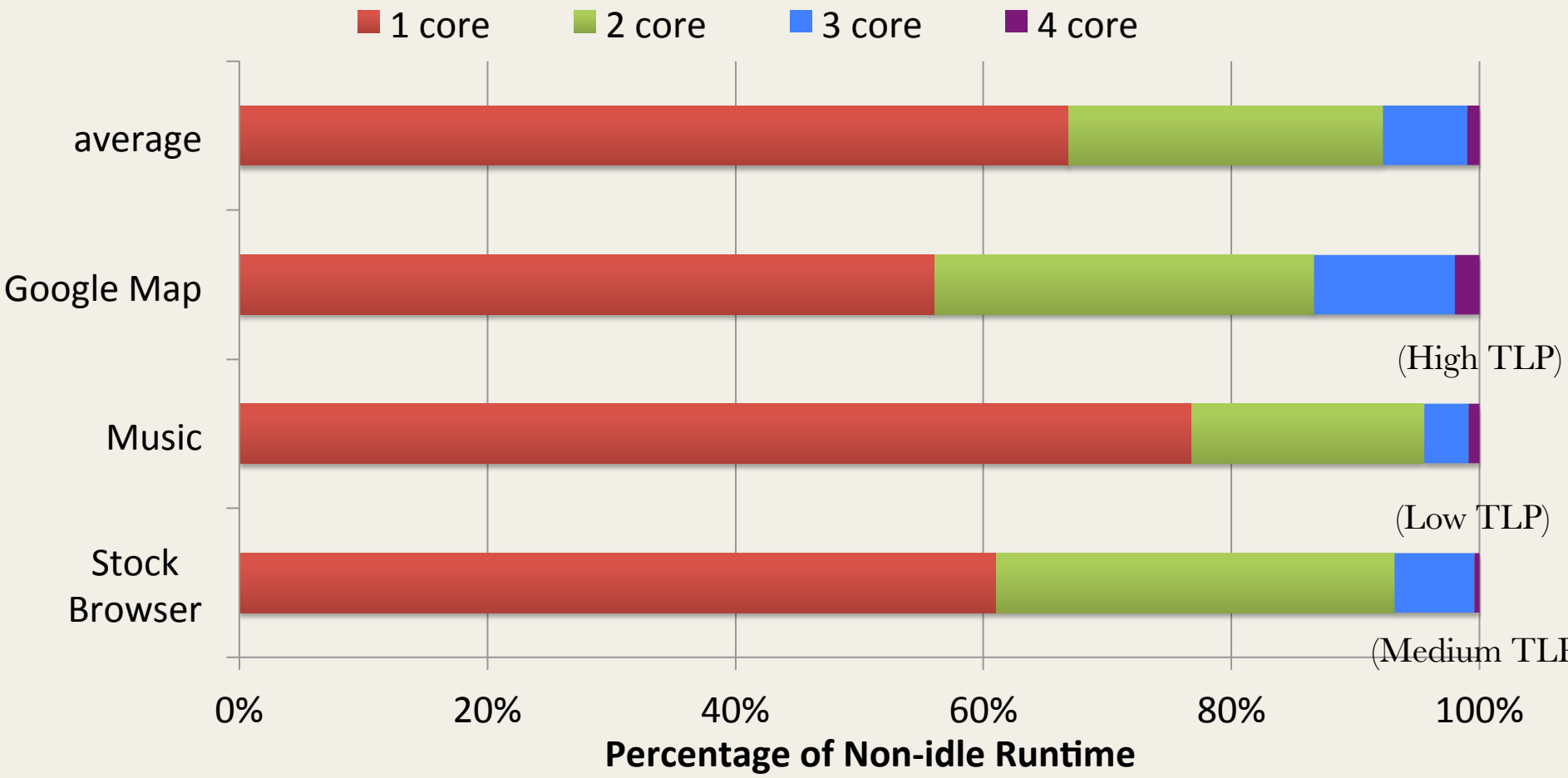
GPU utilization



- Some applications consumes a substantial amount of GPU
- Further reduces the amount of parallelism that CPU can exploit

36

How frequently do we use the fourth core?



The system is mostly using less than 2 cores; "peak TLP" rarely happens