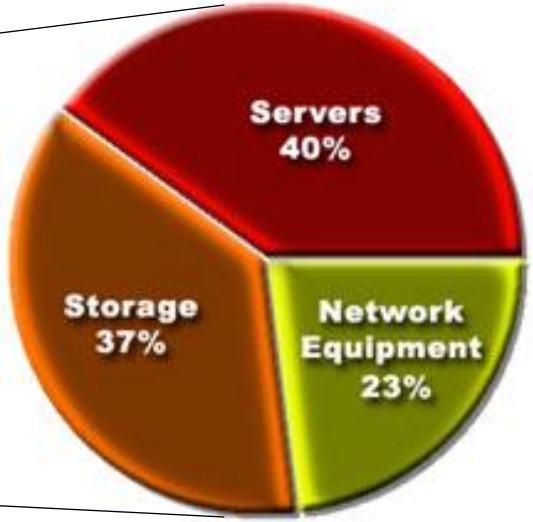
QUANTITATIVE ESTIMATION OF THE PERFORMANCE DELAY WITH PROPAGATION EFFECTS IN DISK POWER SAVINGS

FENG YAN,[1] XENIA MOUNTROUIDOU,[1] ALMA RISKA,[2] EVGENIA SMIRNI[1]

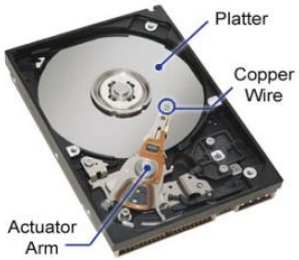
[1] Computer Science Department
College of William and Mary

[2] EMC Corporation

WHY DISKS?

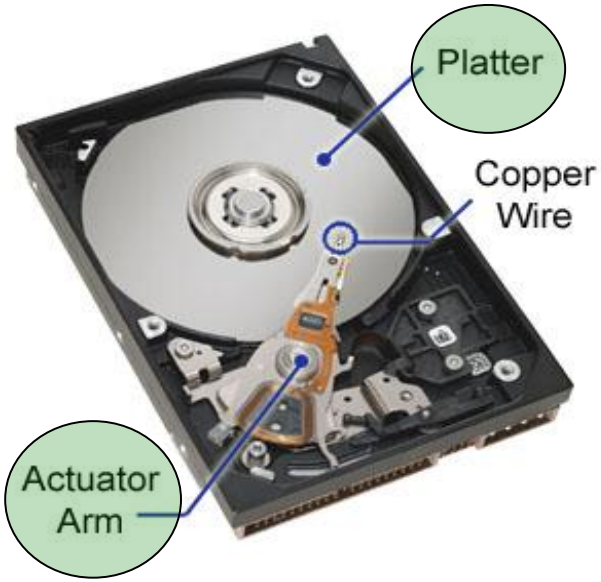


- Storage: main power consumption component
- Disk level: portability and scalability, blackbox
- Disks are underutilized: power savings potential



DISK POWER SAVING MODES

Mode	Power Saving	Penalty
Arm unloaded Full rotation speed	48% of operational	0.5 sec
Arm unloaded Reduced rotation speed	60% of operational	1 sec
Arm unloaded No rotation Electronics on	70% of operational	8 sec
Disk fully spin down	95% of operational	20 sec



- Hitachi Global Storage Technologies, *"Power and acoustics management"*
- Seagate Technology, *"Constellation ES: High capacity storage designed for seamless enterprise integration"*

SCHEDULING TARGET

How to do the power savings?

What is a good scheduling strategy?

User Performance Guarantees

Max Power Saving Amount

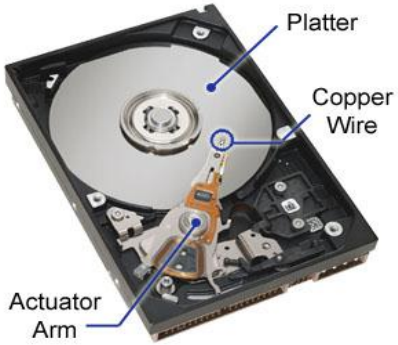
SCHEDULING TARGET

User Performance Guarantees

Max Power Saving Amount

Penalty Time: from power saving to active

How to schedule transparently?



Performance Degradation = Extra Delay / Original RT

≤ Pre-defined User Performance Target

SCHEDULING TARGET

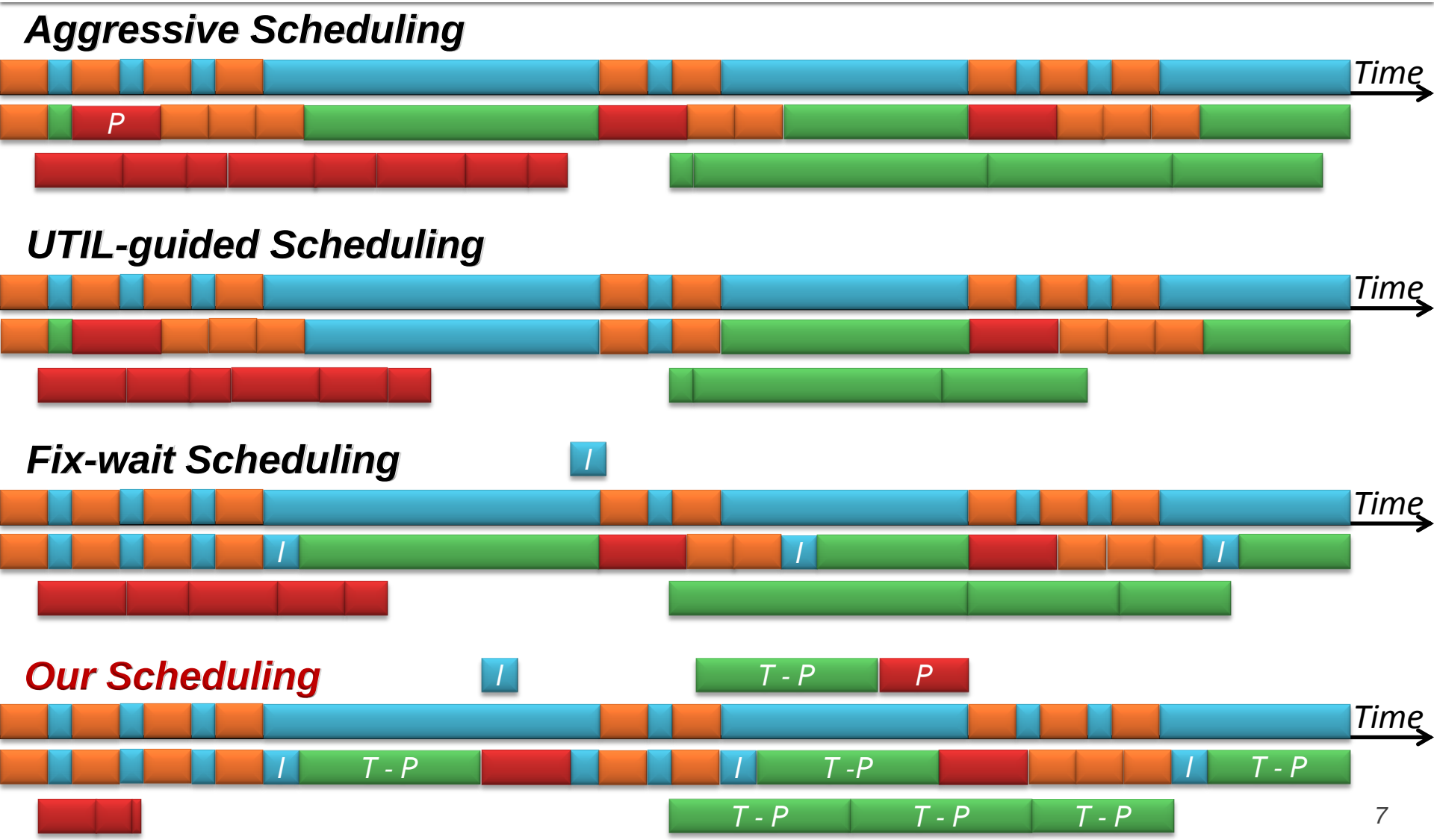
User Performance Guarantees

Max Power Saving Amount

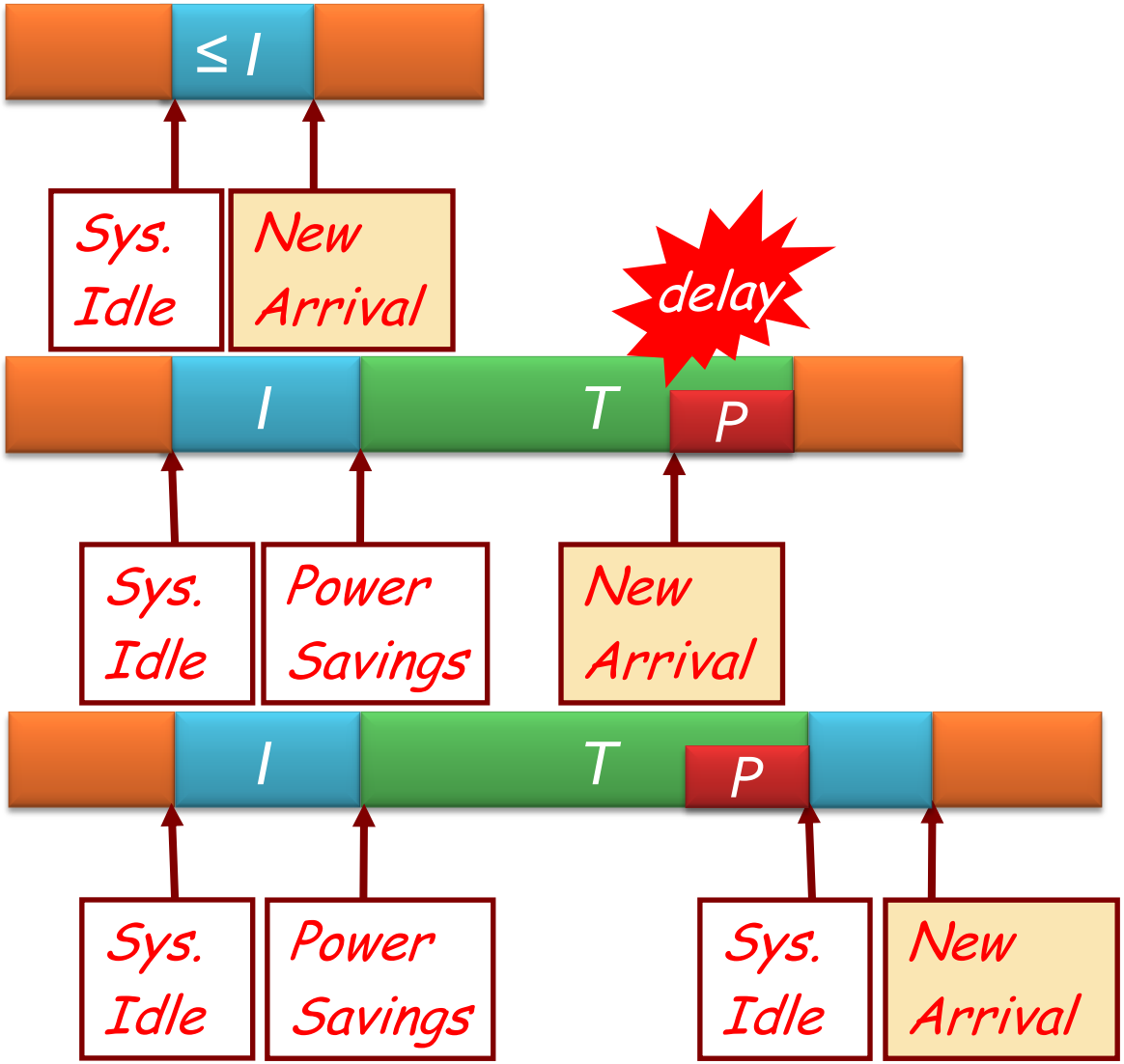
Power Saving Amount =

Time in Power Saving Mode / Total Idle Time

STATE OF THE ART SCHEDULING



OUR SCHEDULING



Idle time ≤ I

- *Power Savings* ×
- *Performance Degrade* ×

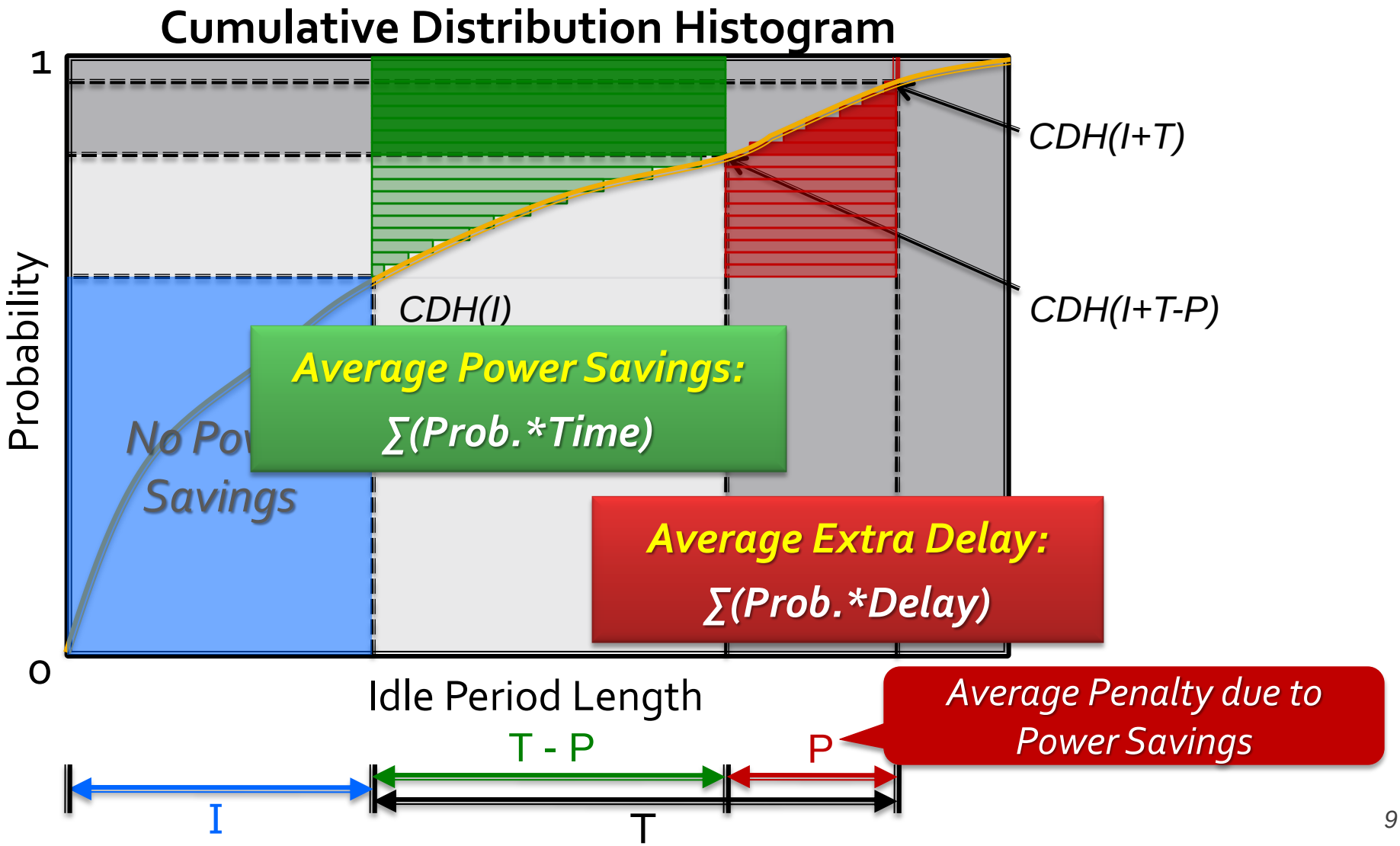
I < Idle time < I + T

- *Power Savings* √
- *Performance Degrade* √

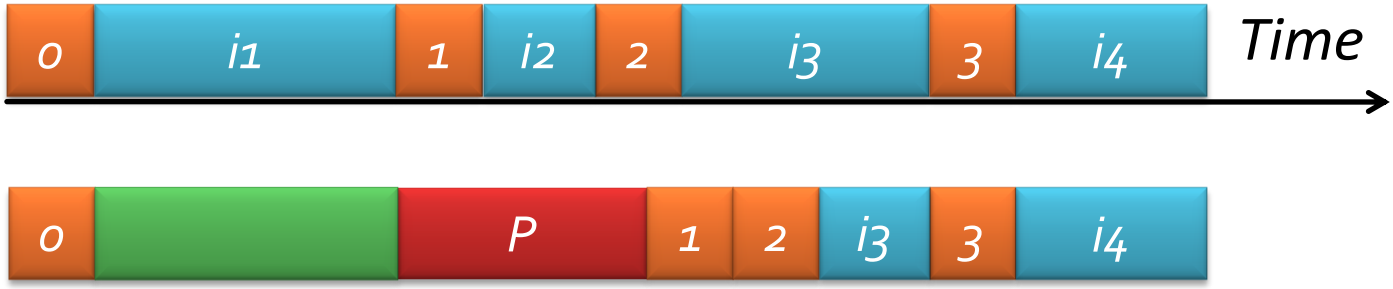
Idle time ≥ I + T

- *Power Savings* √
- *Performance Degrade* ×

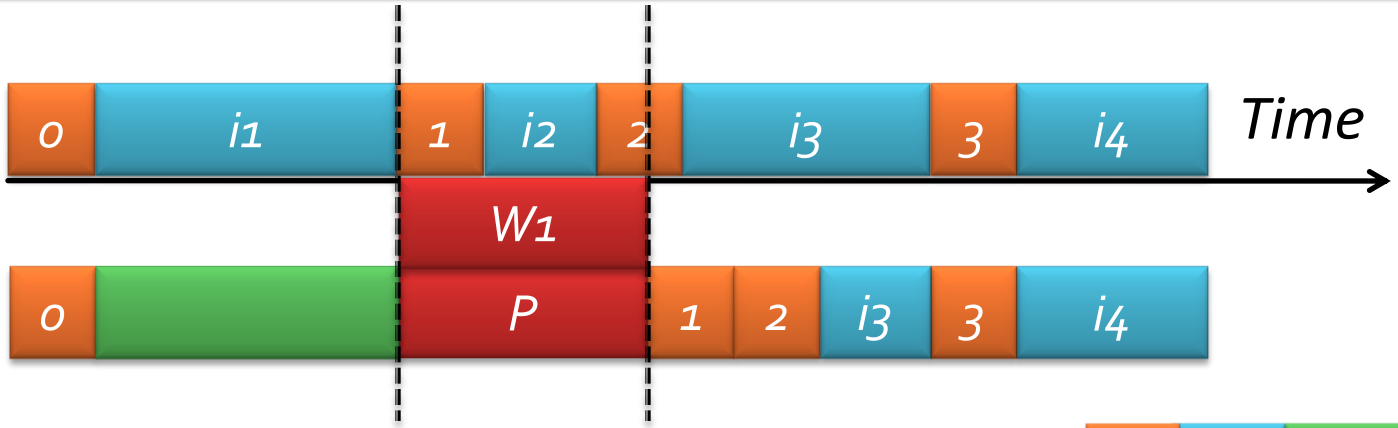
OUR SCHEDULING



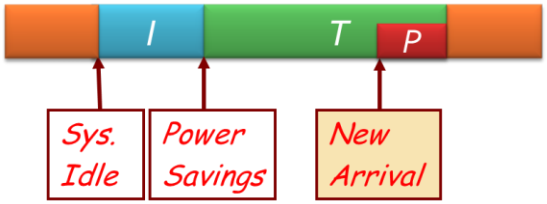
DELAY PROPAGATION



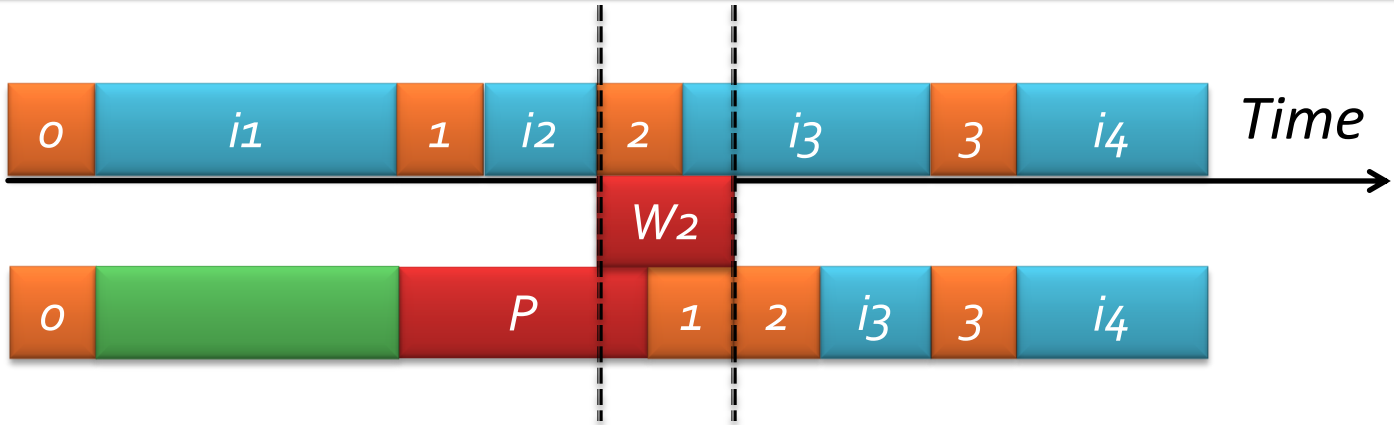
DELAY PROPAGATION



1st delay	If $I < i_1 < I + T$ $1 \leq W_1 \leq P$

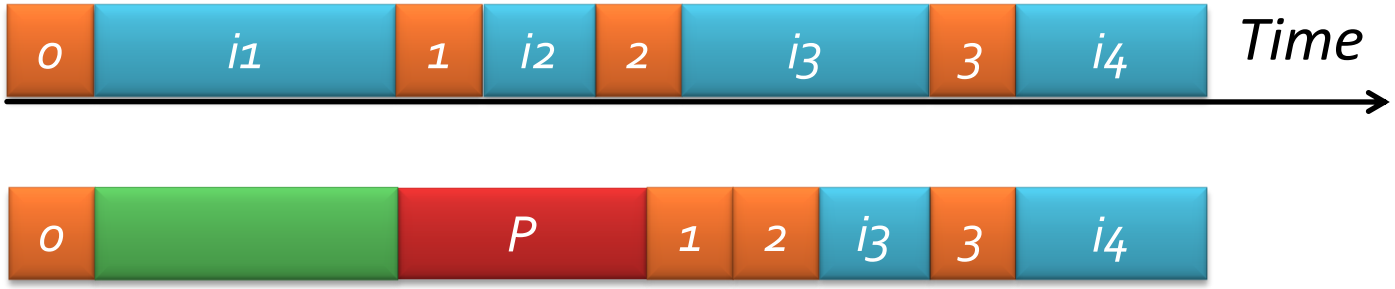


DELAY PROPAGATION



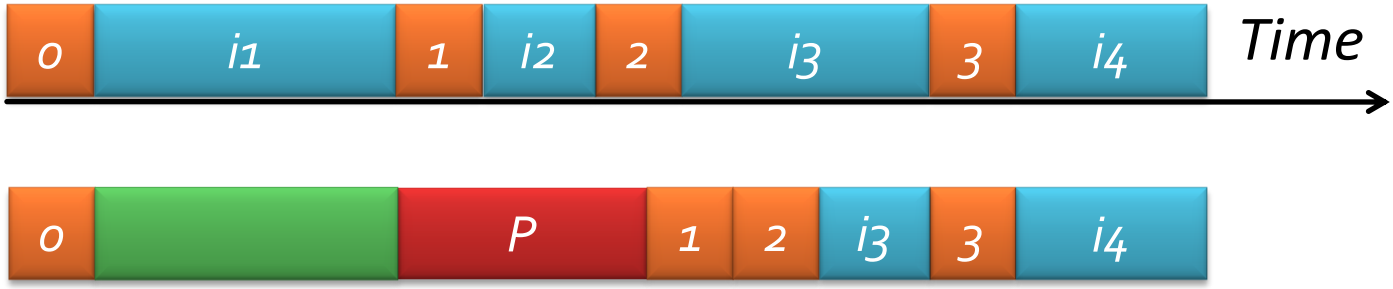
1st delay	If $I < i_1 < I + T$ $1 \leq W_1 \leq P$
2nd delay	If $W_1 > i_2$, $W_2 = W_1 - i_2 \leq P - 1$

DELAY PROPAGATION



1st delay	If $l < i_1 < l + T$ $1 \le W_1 \le P$	Delay P Prob(P)
2nd delay	If $W_1 > i_2$, $W_2 = W_1 - i_2 \le P - 1$	Delay P – 1 Prob(P-1)
⋮		
k^{th} delay	If $(i_2 + i_3 + \ldots + i_k) < W_1 < (i_2 + i_3 + \ldots + i_k + i_{(k+1)})$, $W_k = W_1 - i_2 - i_3 - \ldots - i_k \le P - k + 1 \quad (1 \le k \le P)$	Delay P - k + 1 Prob(P-k+1)

DELAY PROPAGATION



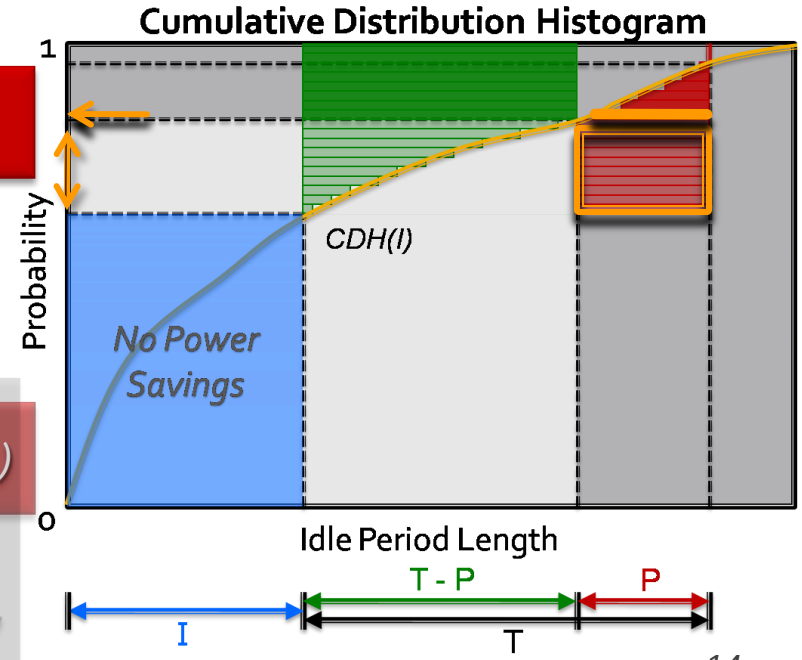
$$Prob(P) = CDH(I+T-P) - CDH(I)$$

Only at first delay

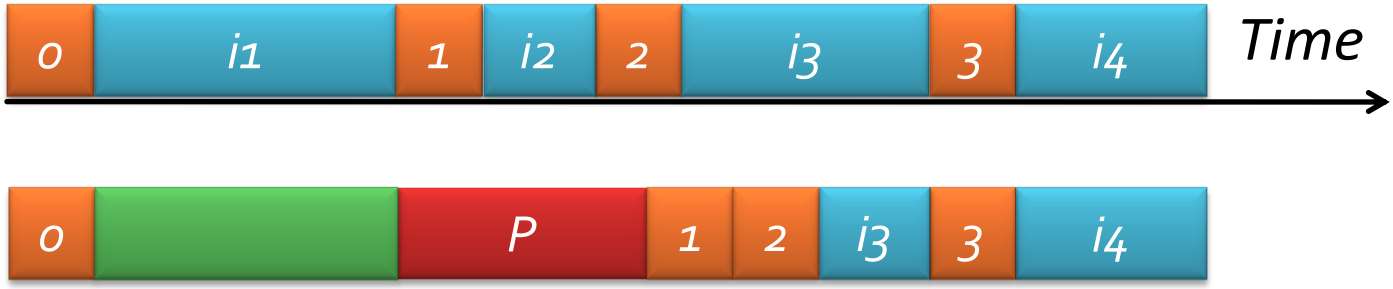
$$Prob(P-1) = CDH(I+T-P+1) - CDH(I+T-P) + Prob(P)*CDH(1)$$

Case1: at first delay

Case2: at second delay



DELAY PROPAGATION



Average Extra Delay:

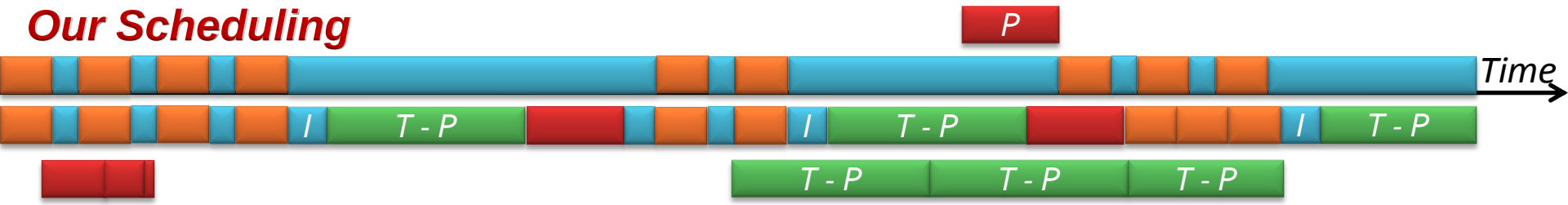
∑(Prob.*Delay)

Delay P
Prob(P)

Delay P – 1
Prob(P-1)

Delay P - k + 1
Prob(P-k+1)

SCHEDULING TARGET



User Performance Guarantees

Max Power Saving Amount

EVALUATION

Enterprise Traces

General Trace Description

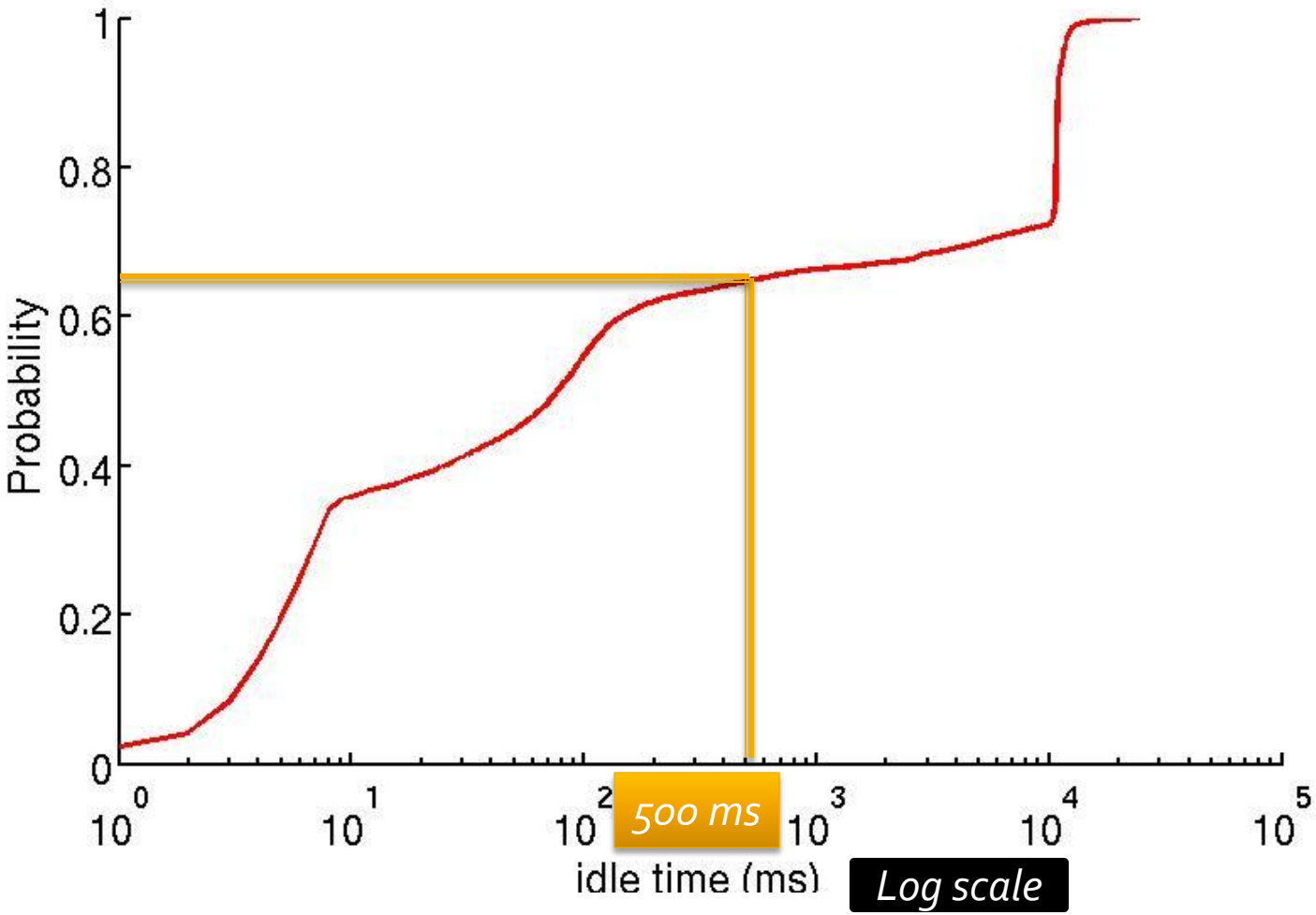
Seagate

Trace	UTIL (%)	Idle Length		Mean Arrival Rate	Mean Service Rate
		Mean (ms)	CV		
CODE1	5.6	192.6	8.4	0.0089	0.1596
CODE2	0.7	1681.6	2.3	0.0013	0.1859
FILE1	1.7	767.5	2.3	0.0033	0.1938
FILE2	0.7	2000.2	2.3	0.0011	0.1596
	low UTIL	*challenge necessity of CDH*			

EVALUATION

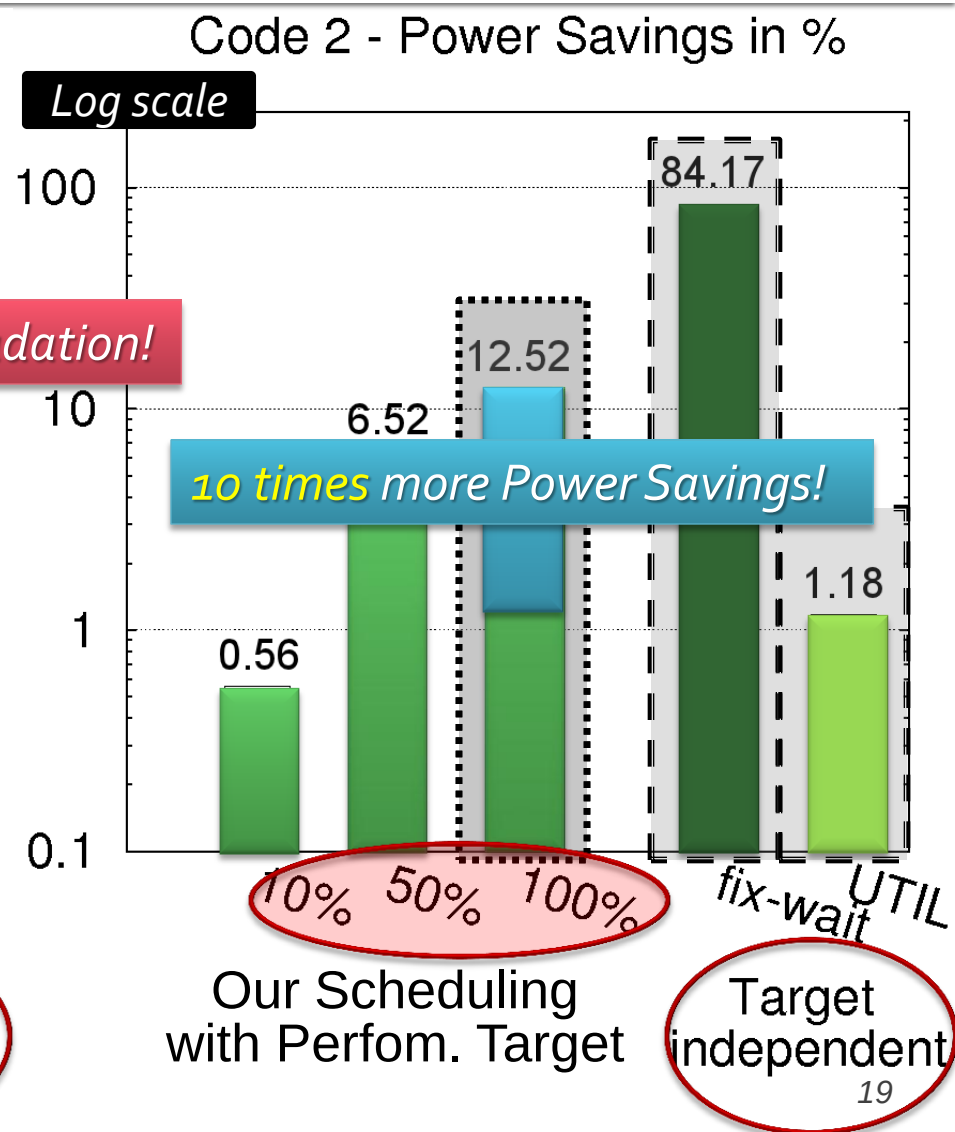
Workload Analysis

CDH of idle period length



EVALUATION

Results



CONCLUSIONS

Performance Delay Estimation with

Delay Propagation Effects

Verified with enterprise trace driven simulations

User Performance Guarantees

Max Power Saving Amount

FUTURE WORK

- **Explore clustering idleness case**
- e.g. autocorrelation in consecutive idle periods

- **Cross correlation with busy periods**
- Better estimation and scheduling

- **Implementation**
- Linux kernel + IO driver
- Benchmark

THANK YOU!

Questions?