

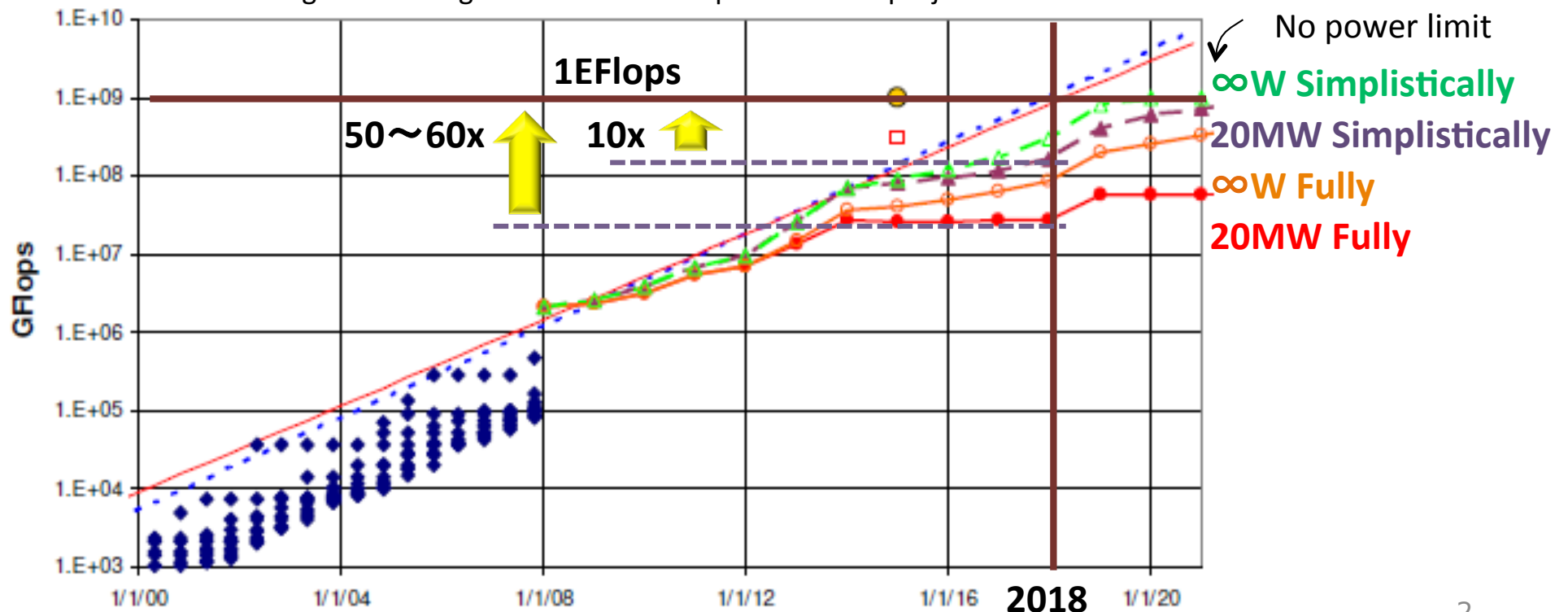
CPU-Memory Power Shifting on High Performance Computing

Koji Inoue
Kyushu University

Hitting the Power Wall in HPC!

- DARPA Target: 1 Exa-FLOPS@20~30MW
- Need 50x performance/watt improvement
 - E.g., 10PFlops@13MW (e.g. K computer)

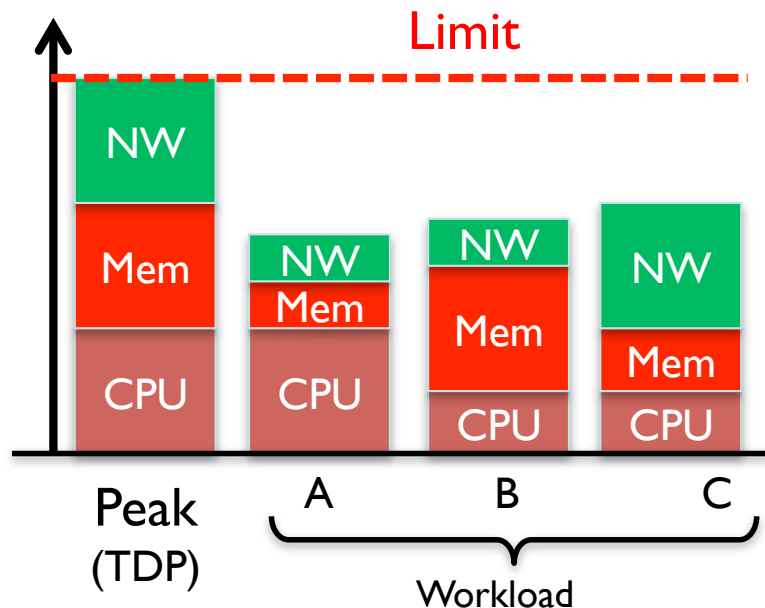
Figure 7.14: Light node strawman performance projections



Adaptive Power Management on Overprovisioned Systems

Conventional

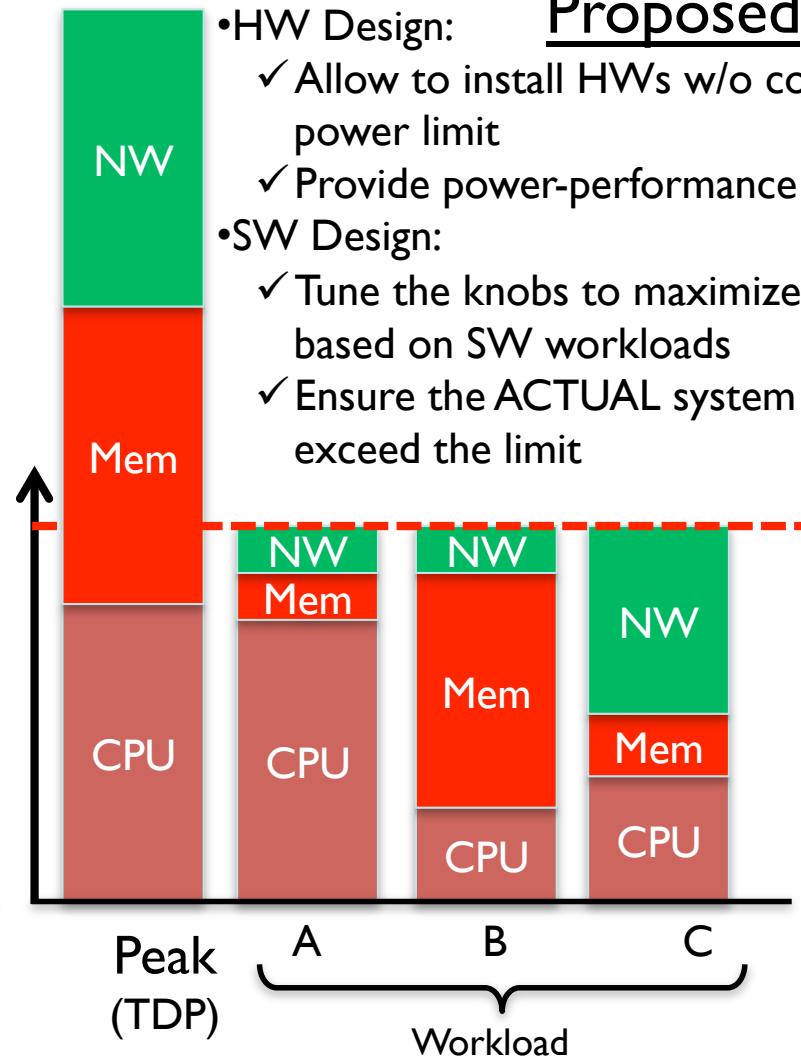
- HW Design
 - ✓ Ensure the PEAK system power does NOT exceed the limit
- SW Design
 - ✓ Try to maximize the activity of HW components



TDP: Thermal Design Power

Proposed

- HW Design:
 - ✓ Allow to install HWs w/o considering the power limit
 - ✓ Provide power-performance knobs (e.g. DVFS)
- SW Design:
 - ✓ Tune the knobs to maximize the performance based on SW workloads
 - ✓ Ensure the ACTUAL system power does NOT exceed the limit



Power Budgeting for Green High-Performance Computing

- Power is the 1st order design constraint
- Problem: How can we use the limited power budget effectively?
- Answer: Power Shifting
 - Optimize power budget allocation based on ...
 - Characteristics of application programs
 - System Organization
 - Situation of power conditions (energy harvesting, battery, etc.)

Power Measurement/Control Platform

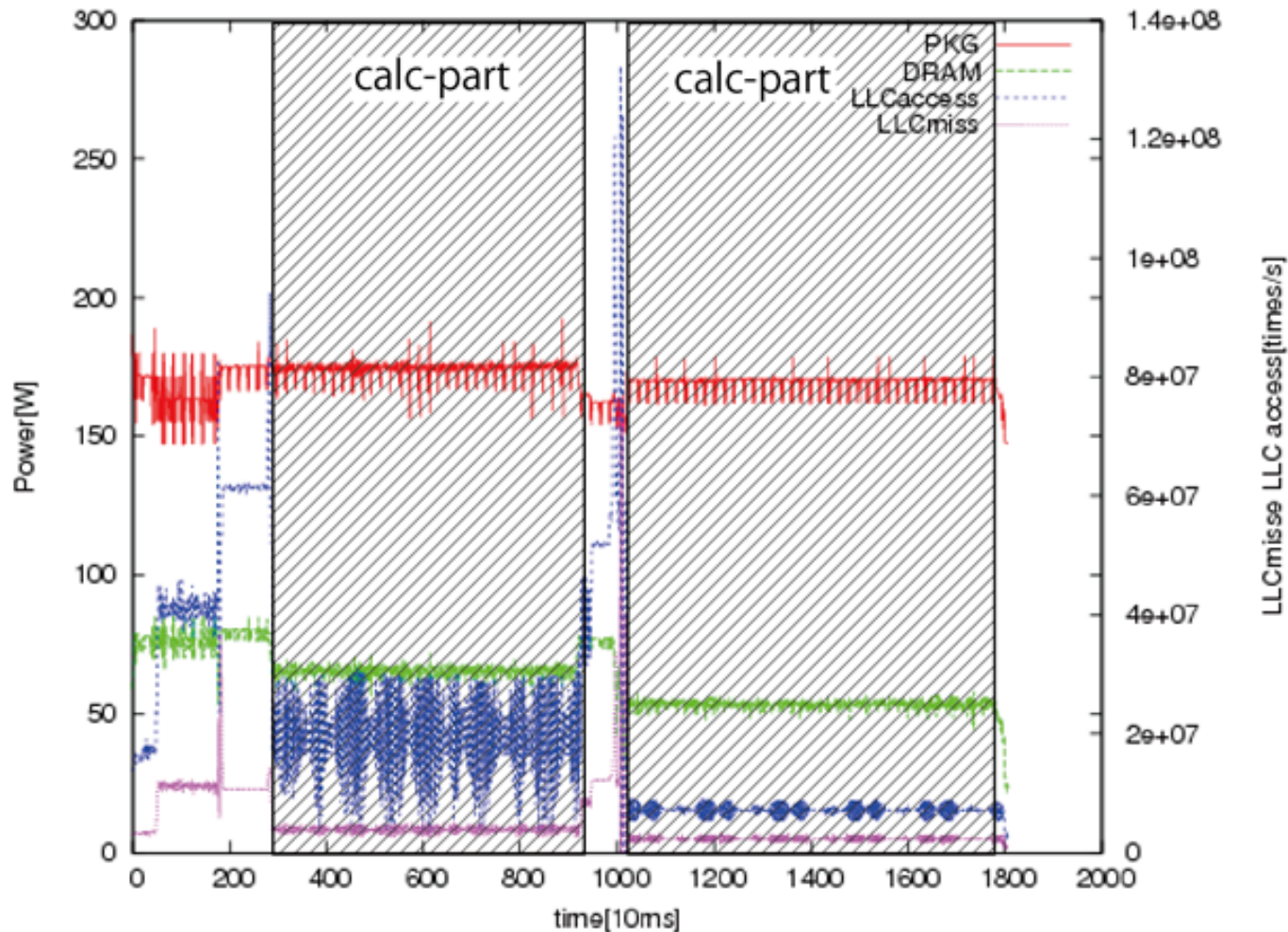
- Intel RAPL (Running Average Power Limit)
 - Measurement
 - CPU package
 - DRAM
 - Control (Capping)
 - CPU package
 - DRAM (no support in CX400)
- Power management APIs
 - Insert APIs into application source codes
 - Fine-grain measurement and control



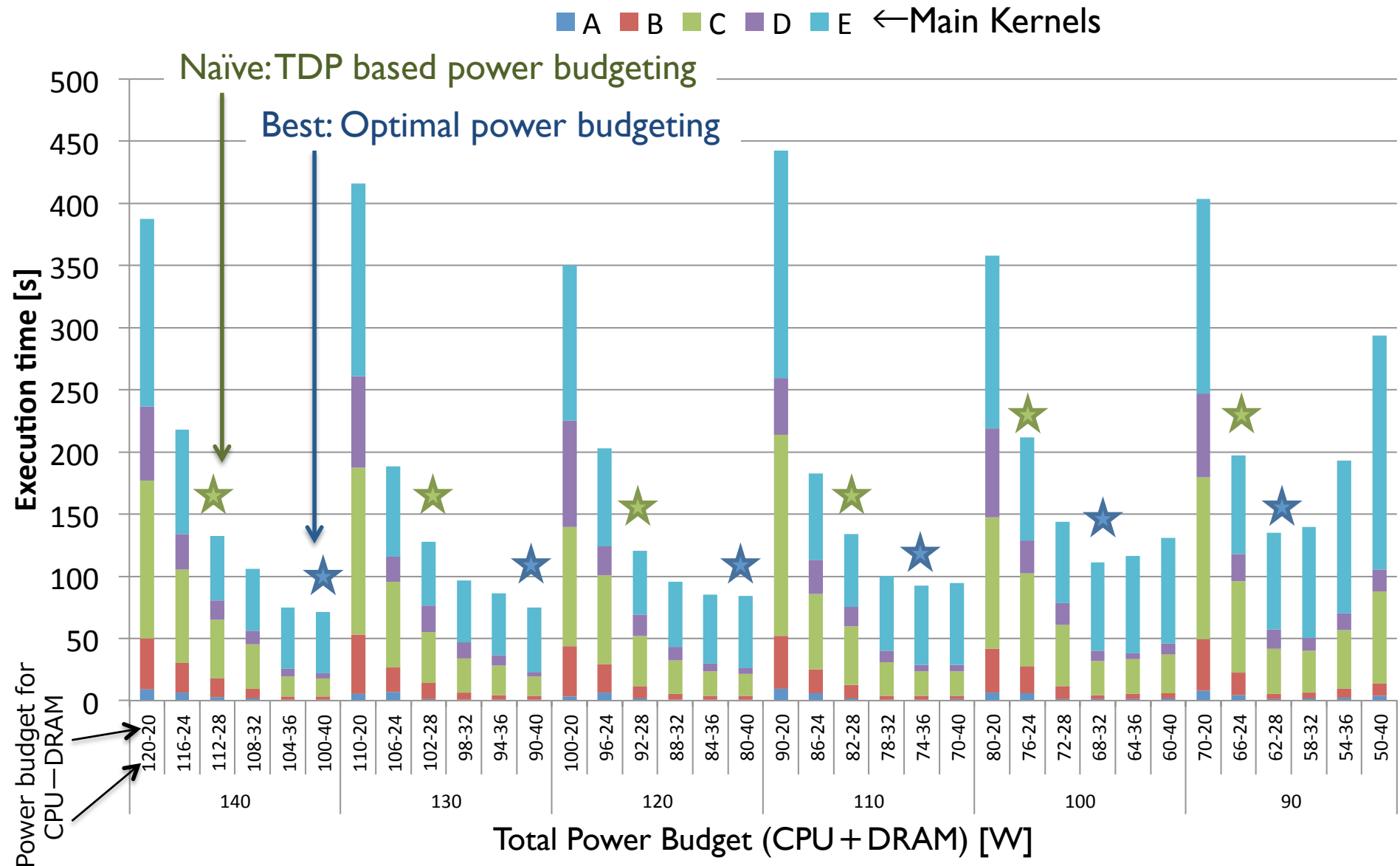
PRIMERGY CX400		
CPU	Architecture	8 cores Xeon E5 2680
	Frequency	2.7GHz (172.8GF)
	Cache	L2: 256KB/core L3: 20MB/CPU
DRAM	DDR3-800	8GB x 8 /CPU
B/F	51.2/172.8	0.30
Node	Number of CPUs	2
	Memory size	128GB
System	Number of nodes	1476 (23616 cores)
	Rmax	510.1 TFlops
	Node comm.	InfiniBand FDR

Power Characteristics

~ A Case for MHD Simulation ~



Potential of Static Power Budgeting



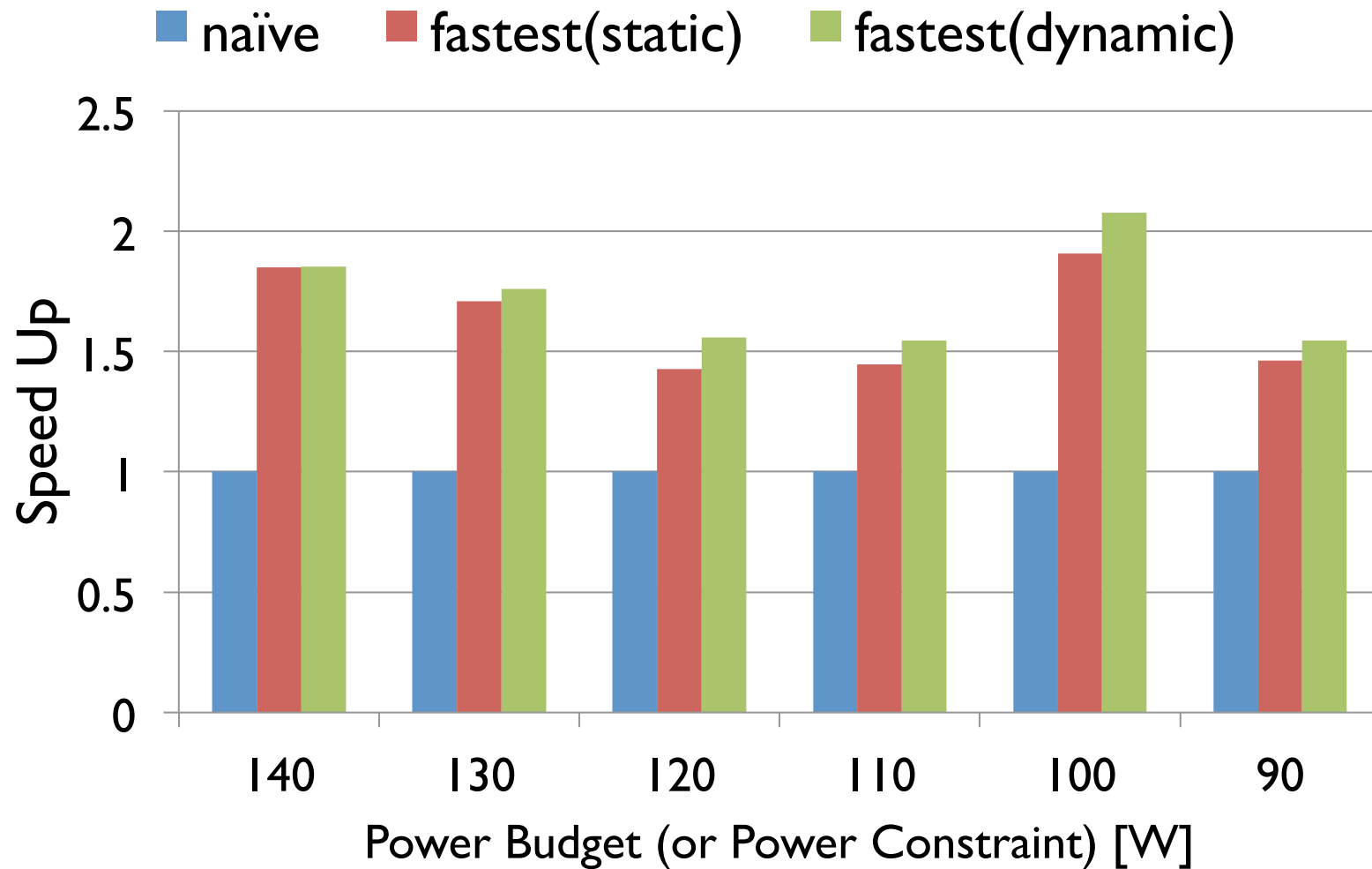
Analysis

 Highest performance

		A	B	C	D	E			A	B	C	D	E
140	120-20	9.04436	40.78316	127.411	59.36324	151.098	110	90-20	9.55334	42.42236	161.6456	45.89492	182.9402
	116-24	6.37768	23.91736	75.42682	28.38964	83.65048		86-24	6.1914	18.98878	60.79354	27.37266	69.43712
	112-28	2.86352	14.76684	47.2427	15.87258	51.5267		82-28	1.74454	10.76196	47.38888	15.74866	58.19622
	108-32	1.53968	8.1761	35.55314	10.96	49.97478		78-32	0.97586	2.98458	26.81844	9.20046	60.2556
	104-36	0.82382	2.66794	15.68744	6.34678	49.24056		74-36	0.98752	2.83744	19.68414	5.0287	64.08158
	100-40	0.83454	2.7584	14.0136	4.62882	49.22654		70-40	1.02242	2.75304	19.81506	5.04108	66.17626
130	110-20	5.62726	47.62156	134.2444	73.36722	155.1247	100	80-20	6.20774	35.24548	106.0428	71.4866	138.7436
	106-24	6.92304	19.572	68.97942	20.4666	72.72526		76-24	5.99138	21.68094	74.81612	26.12136	83.27296
	102-28	1.4581	12.82234	40.79068	21.28478	51.45904		72-28	1.25226	10.13248	49.56568	17.4538	65.46454
	98-32	1.0092	5.53238	27.5923	12.7634	49.50754		68-32	1.15174	3.3722	27.47576	8.14828	71.01516
	94-36	0.85158	3.46384	23.91928	8.51032	49.55542		64-36	1.4323	3.81416	28.06454	4.52192	78.44644
	90-40	0.86138	2.78592	15.77782	3.70614	51.70394		60-40	1.63522	4.39828	31.2576	8.44862	85.29442
120	100-20	3.3495	40.38996	96.09186	85.20938	124.9532	90	70-20	8.25766	41.06074	130.4021	67.02224	156.8064
	96-24	6.46996	22.5775	71.62598	23.6602	78.54256		66-24	4.15348	18.50054	73.70576	21.39618	79.58886
	92-28	1.53528	10.01838	40.30686	17.10306	51.372		62-28	1.37868	4.27844	36.21752	15.17108	78.01822
	88-32	0.85936	4.34696	26.94304	11.1015	52.6243		58-32	1.75086	4.76856	33.86146	10.23006	89.12674
	84-36	0.94046	2.9368	19.52648	6.35926	55.61054		54-36	2.5374	6.83384	47.383	13.47268	122.7257
	80-40	0.8968	3.0781	17.52374	4.53542	58.24958		50-40	3.96664	10.0048	73.85626	17.86022	188.0572

Execution time of each kernels [sec]

Potential of Dynamic Power Budgeting



Conclusions

- Summary
 - Power is the most serious issue for post-petascale computing
 - The Keys: “Overprovision” and “Power budgeting (capping)”
- Ongoing Work
 - Find the best power budgeting (semi-)automatically
 - Maximize effective performance under power constraints