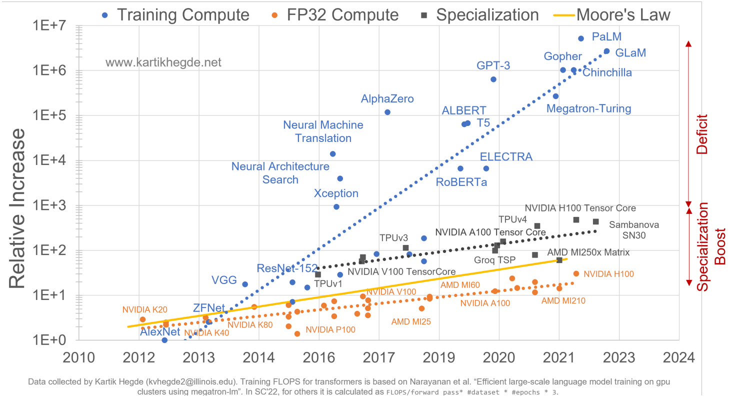
Energy aware compute scheduling in heterogeneous networks

Parimal Parag

Electrical Communication Engineering
Indian Institute of Science

The growing performance deficit

Deep learning compute demand[1]



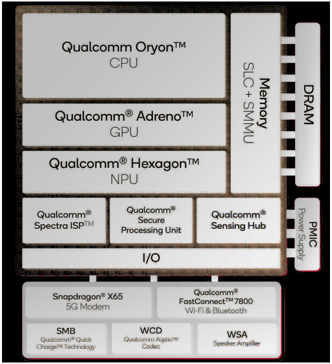
- End of Moore's law and Dennard scaling
- Can no longer keep adding more transistors and increasing core frequencies
- We're accustomed to increasing compute

[1] Image credit: https://kartikhegde.substack.com/p/accelerating-deep-learning-in-the

Heterogeneous Computing[2]

Sources of heterogeneity

- Different generations of servers and accelerators
- Pooling of all available compute resources (CPUs, GPUs, NPUs)
- Compute resources may be run at different speeds for energy conservation
- Compute cores optimized for different operating regions, to deal with dynamic workloads
- External factors—network bottlenecks, data affinity, etc.



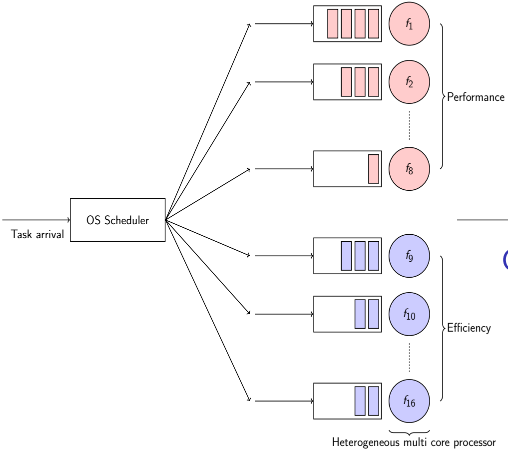
Achieving sustainability goals

- Idle cores save power
- Dynamic voltage and frequency scaling

[2] Image: https://www.anandtech.com/show/21445/qualcomm-snapdragon-x-architecture-deep-dive

Energy aware compute scheduling[3]

Heterogeneous multi core system



Objective

- Minimize core energy consumption
- Meet service guarantees such as mean or tail compute latency

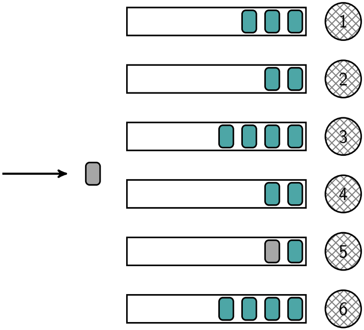
Control parameters at cores

- Load balancing
- Operating frequency
- Idle or active state

[3] A. Priya, R. Choudhury, S. Patni, H. Sharma, M. Mohanty, K. Narayanam, U. Devi, P. Moogi, P. Patil, and P. Parag. Energy-minimizing workload splitting and frequency selection for guaranteed performance over heterogeneous cores. *ACM International Conference on Future and Sustainable Energy Systems (e-Energy)*. pp. 308–322, June 2024.

Load balancing policies

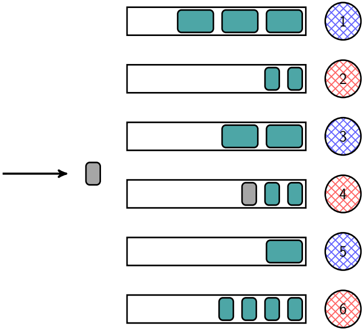
Homogeneous servers



- Join shortest queue (1) [a]

[a] W. Winston, "Optimality of the shortest line discipline," J. App. Prob., 14(1), 181–189, Mar 1977.

Heterogeneous servers

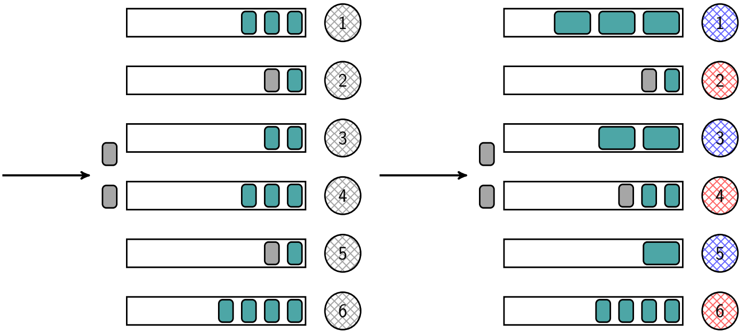


- Join smallest workload (1) [a]

[a] R. R. Weber, "On the optimal assignment of customers to parallel servers," J. App. Prob., 15(2), 406–413, Jun 1978.

Load balancing policies—parallel processing of subtasks

Join the shortest queue/workload (k)



- Equivalent to (n, k) fork-join system [4]

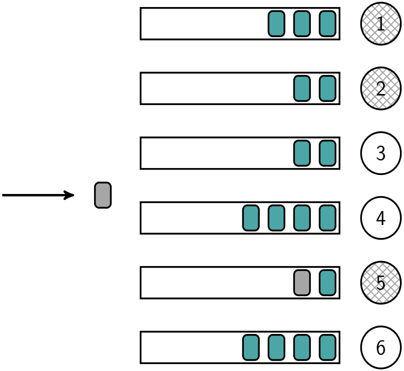
Pro Minimizes the mean task completion time

Con Feedback overhead linearly scaling in the number of servers

[4] A. Badita, P. Parag, and J.-F. Chamberland. Latency analysis for distributed coded storage systems. IEEE Transactions on Information Theory. 65(8):4683–4698, Aug 2019.

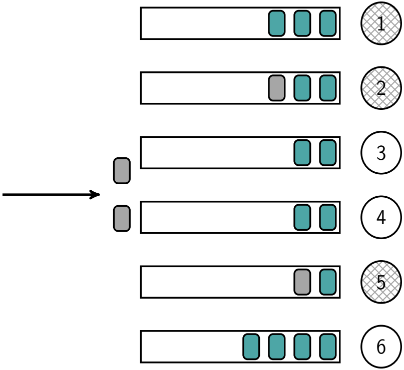
Load balancing policies—low overhead alternative [5]

Power-of-d (1)



- Equivalent to $(d, 1)$ fork-join queue
- When $d = n$, it is JSW

Power-of-d (k)

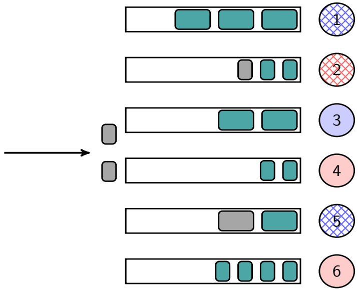


- Equivalent to (d, k) fork-join queue
- When $d = n$, it is (n, k) fork-join

[5] M. Mitzenmacher, "The power of two choices in randomized load balancing," IEEE Trans. Parallel Distrib. Syst., vol. 12, no. 10, pp. 1094–1104, Oct. 2001.

Heterogeneous servers

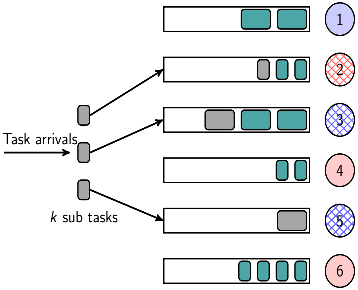
Low overhead alternative[a]



- (d, k) fork-join system
- Sample d servers, join k smallest

[a] R. Jinan, A. Badita, T. P. Bodas, and P. Parag. Load balancing policies without feedback using timed replicas. *Performance Evaluation*. 162, 102381, Nov 2023.

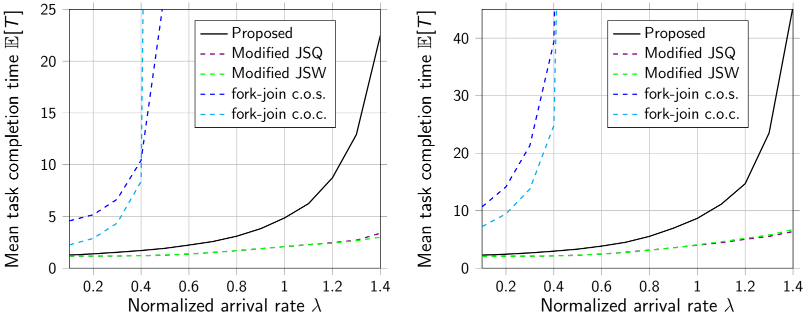
Zero overhead alternative[a]



- (k, k) fork-join system
- Probabilistic selection of slow servers

[a] M. Mohanty, G. Gautam, V. Aggarwal, and P. Parag. Analysis of fork-join scheduling on heterogeneous parallel servers. *IEEE/ACM Transactions on Networking*. 32(6):4798–4809, Dec 2024.

Performance comparison

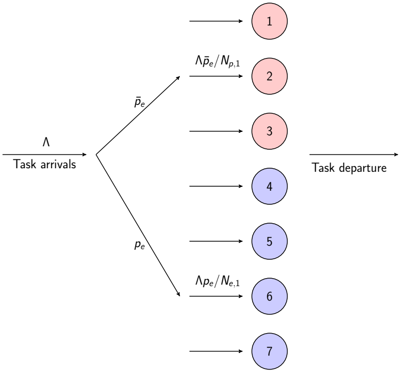


(a) $n = 10^2,\ k = 10$

(b) $n = 10^3,\ k = 10^2$

Figure: System with $f_s = 0.5$, $(\mu_s, \mu_f) = (0.5, 2.5)$

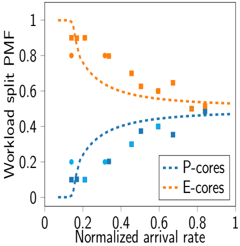
Energy aware compute scheduling



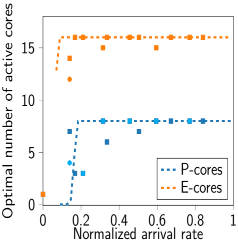
Optimal selection probability and frequency allocation

- Split workload between performance and efficiency cores according to single selection probability p_e^*
- Common frequency among all active cores of identical type $f^* \triangleq \frac{1}{\alpha}\left(\Lambda\gamma + \frac{1}{w}\right)$
- Number of active cores $N_{e,1}^*, N_{p,1}^*$

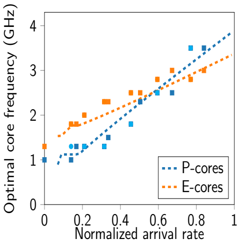
Optimal splitting between classes



Optimal workload split



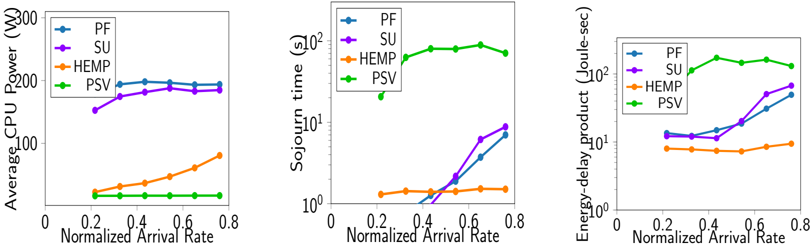
Optimal number of active cores



Optimal frequency

- Probabilistic split between two core types
- Constant frequency for all active cores of one type

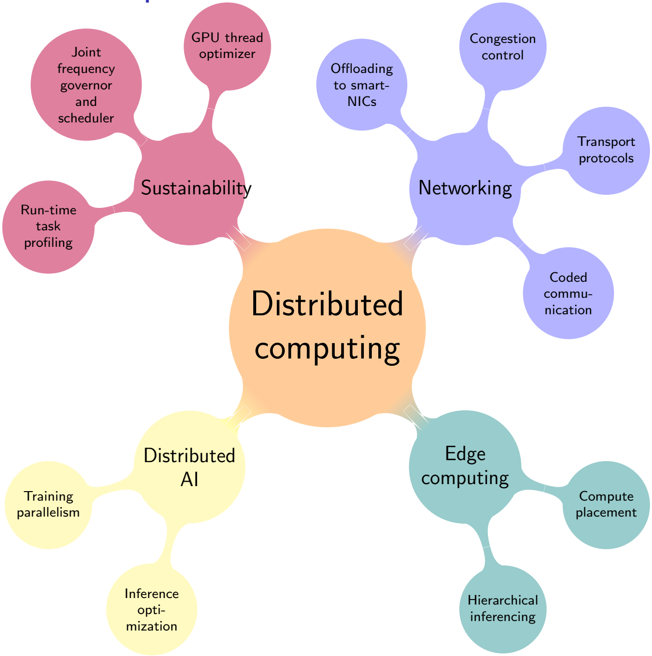
Comparison with Linux frequency governors



Total power   Sojourn time   Energy-delay product

TensorFlow Lite workload
PF: Performance   **SU**: Schedutil   **PSV**: Powersave

Research Landscape



Acknowledgements

CENTRE FOR NETWORKED INTELLIGENCE
Indian Institute of Science

IBM

intel

Qualcomm

NRF National Research Foundation

References

- M. Mohanty, G. Gautam, V. Aggarwal, and P. Parag. Analysis of fork-join scheduling on heterogeneous parallel servers. *IEEE/ACM Transactions on Networking*. 32(6):4798–4809, Dec 2024.
- A. Priya, R. Choudhury, S. Patni, H. Sharma, M. Mohanty, K. Narayanam, U. Devi, P. Moogi, P. Patil, and P. Parag. Energy-minimizing workload splitting and frequency selection for guaranteed performance over heterogeneous cores. *ACM International Conference on Future and Sustainable Energy Systems (e-Energy)*. pp. 308–322, Jun 2024.
- R. Jinan, A. Badita, T. P. Bodas, and P. Parag. Load balancing policies without feedback using timed replicas. *Performance Evaluation*. 162, 102381, Nov 2023.
- R. Jinan, G. Gautam, P. Parag, and V. Aggarwal. Asymptotic analysis of probabilistic scheduling for erasure-coded heterogeneous systems. *ACM SIGMETRICS Performance Evaluation Review*. 50(4):8–10, Mar 2023.
- A. Badita, P. Parag, and J.-F. Chamberland. Latency analysis for distributed coded storage systems. IEEE Transactions on Information Theory. 65(8):4683–4698, Aug 2019.