TaskFlow: An Energy- and Makespan-Aware Task Placement Policy for Workflow Scheduling through Delay Management.

Laurens Versluis & Alexandru Iosup

Vrije Universiteit Amsterdam

2022-04-09

Introduction

- We combine three topics in “TaskFlow: An Energy- and Makespan-Aware Task Placement Policy for Workflow Scheduling through Delay Management.”
  - We will cover these topics in order.
- We hope these topics inspire the community to further explore this direction.
- We hope the community finds more usages for the underlying idea.

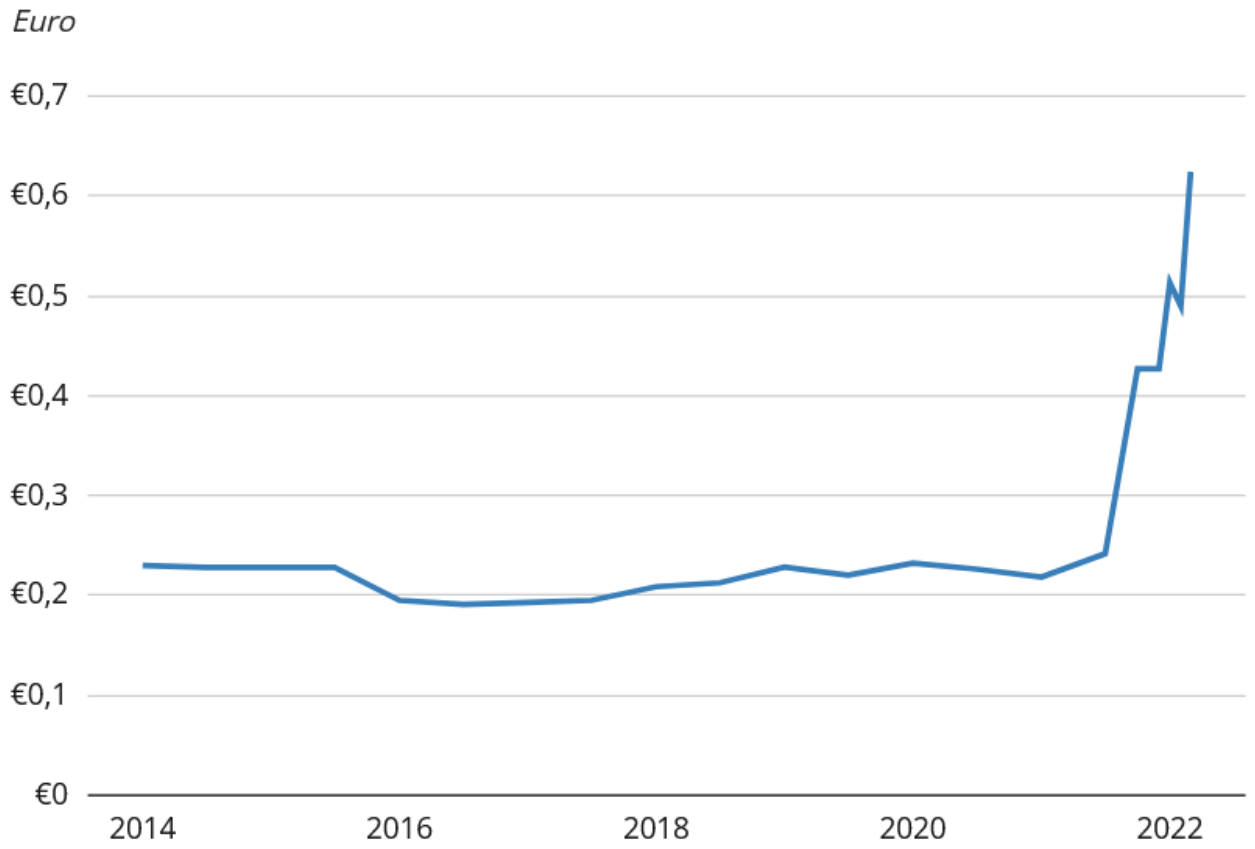
Image source: http://weknowyourdreams.com/single/gift/gift-11

Today's Datacenters

- Many aspects of Datacenters are growing:
  - Their sizes;
  - Their usage;
  - Their revenue;
  - And thus, their impact on societies and economies.
- Several metrics are important for both customer and provider. Two common [1]:
  - Costs
  - Performance (throughput, makespan, runtime, etc.)

Energy In Datacenters

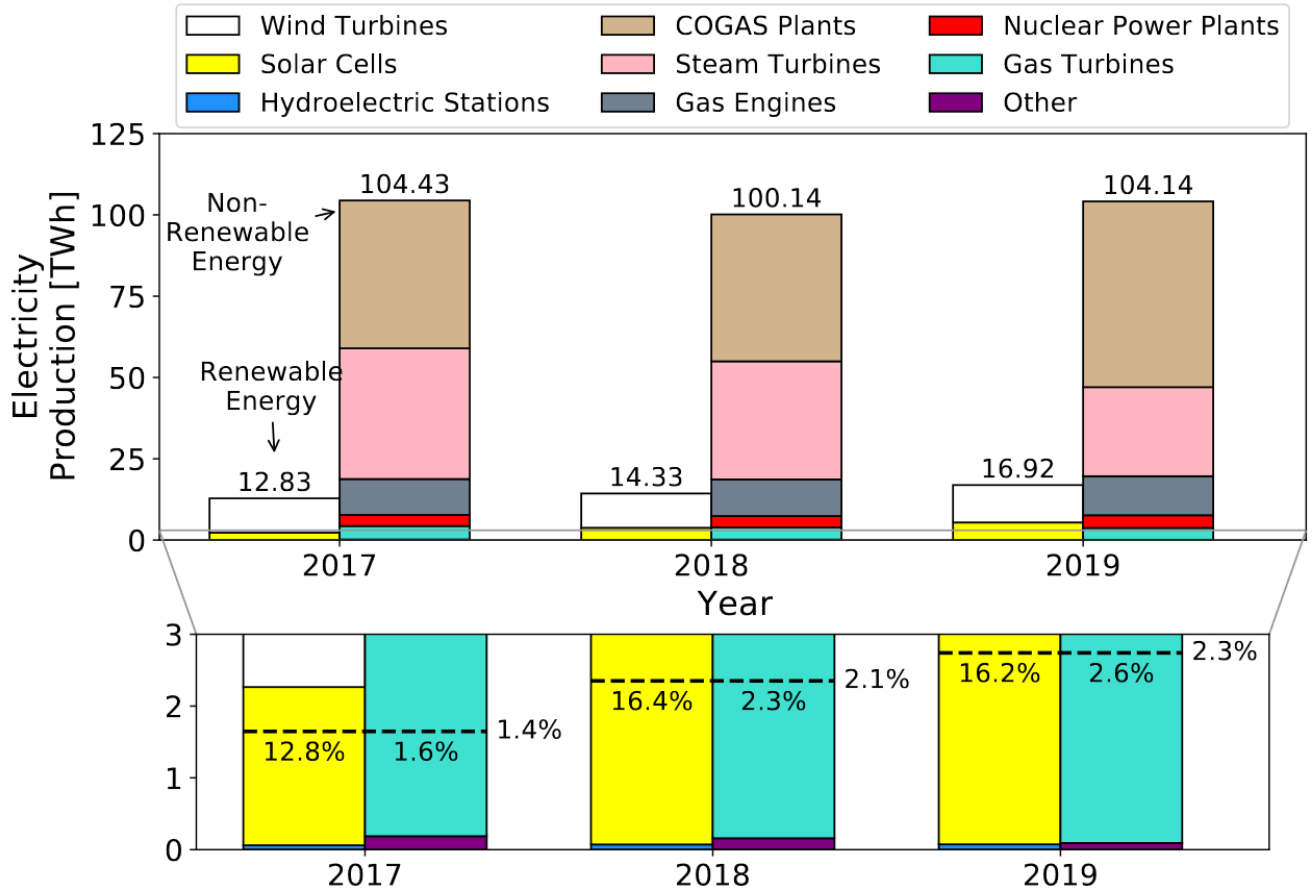
1. Direct impact on costs.



Bron: Overstappen.nl

Energy In Datacenters

1. Direct impact on costs.
2. Direct impact on our climate.
    1. The Netherlands: datacenters consumed 1.4% of all power in 2017, 2.3% in 2019



Energy In Datacenters

1. Direct impact on costs.
2. Direct impact on our climate.
    1. The Netherlands: datacenters consumed 1.4% of all power in 2017, 2.3% in 2019
3. Scrutiny from governments increasing.
    - In the Netherlands: consumption of subsidized green energy.

Facebook owner Meta suspends Zeewolde, Netherlands data center due to political pushback

Project is dead - for now

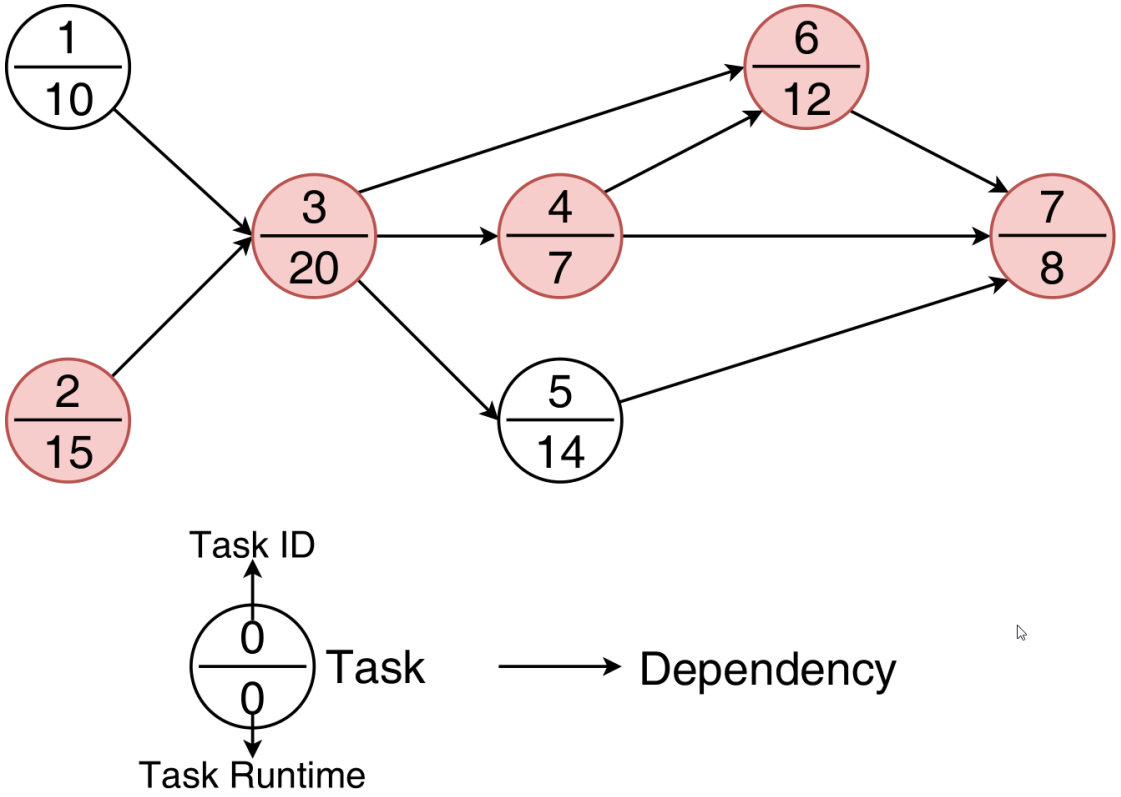
March 30, 2022 By: Sebastian Moss Comment

Energy In Datacenters

1. Direct impact on costs.
2. Direct impact on our climate.
    1. The Netherlands: datacenters consumed 1.4% of all power in 2017, 2.3% in 2019
3. Scrutiny from governments increasing.
    - In the Netherlands: consumption of subsidized green energy.
4. Public image.

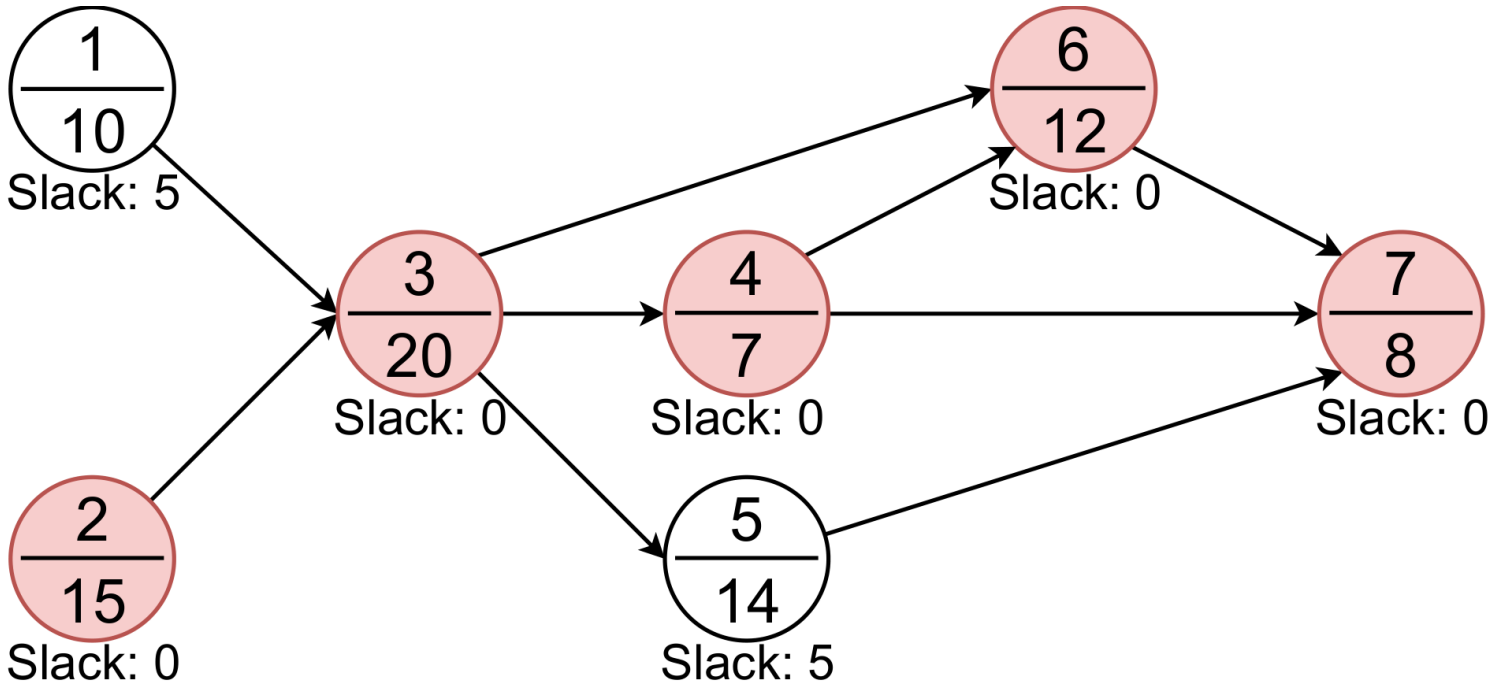
Workflows Scheduling

- One of the most common job types [2].
- Model of Coffman and Graham [3].
  - Directed Acyclic Graph (DAG).
- Critical path, marked red.
- Tasks are the most granular elements.
  - ➢ We can choose the location (the machine) of execution (via a task placement policy).



Delay Management

- Some tasks have leeway, i.e., they can delay without impacting the critical path of the job.
- Previous example, assuming all tasks arrive at time = 0:



- We call the room for delay “slack”.

Computing Slack

- We can compute the slack of all tasks in a workflow in $O(|V| + |E|)$ time and $O(|V| + |E|)$ space.
  - V is the set of vertices (tasks in the workflow).
  - E is the set of edges (constraints between tasks).
- Good alternative for the algorithm of Li et al. [4]
  - Their algorithm has a runtime of $O(|V|^2)$.
- Core of the algorithm is based on topological sorting.
  - Runtime $O(|V| + |E|)$ single-threaded.
  - Runtime can be reduced via parallel (and thus distributed) computing [5].
- See details in our article.

How Much Slack is there in Realistic Workloads?

- Traces from the Workflow Trace Archive [6], split by domain.
- All domains contain slack.

Task Slack [ms] per Domain.

	1^{st}	25^{th}	50^{th}	mean	75^{th}	99^{th}
Engineering	0	0	23,113	92,903	76,621	1,140,122
Industrial	0	9,000	31,000	136,728	101,000	1,987,000
Scientific	0	0	80	169,695	1,952	3,200,659

Can we exploit Slack to Reduce Energy Consumption?

- Two ideas to trade slack for energy:

1. Run tasks on slower, yet more power efficient hardware.

Model	Base Clock [GHz]	Logical Cores	TDP [W]	W/GHz/ core
AMD Ryzen Threadripper 3990X	2.9	128	280	0,75
AMD Ryzen 9 5900X	3.7	24	105	1,18
Intel i9-10900K	3.7	20	125	1,69
AMD Ryzen 7 5800X	3.8	16	105	1,73
Intel i7-10700K	3.8	16	125	2,06
AMD Ryzen 5 5600X	3.7	12	65	1,46
Intel i5-10600K	4.1	12	95	1,93

2. Delay tasks on a machine using Dynamic Voltage & Frequency Scaling (DVFS).
   - We average the numbers reported by Dhiman et al. [7] for their single and multi-threaded workloads.

ID	1	2	3	4
Delay	0%	22.86%	53.44%	147.28%
Energy Savings	0%	8.6%	12.60%	12.40%

First Check using Rough Numbers

- Assumptions:
    1. Workflows are inspected independently.
    2. Task runtimes are based on the fastest machine w.r.t. clock speed.
    3. Task runtimes scale linearly in clock speed.
    4. All machines support DVFS per core.

We acknowledge that in a system workflows would run in parallel, and thus influence each other's execution and that the other settings do not hold in general.

This exercise is purely to check if the idea is viable, and to get a feeling for the theoretical gains that can be had.

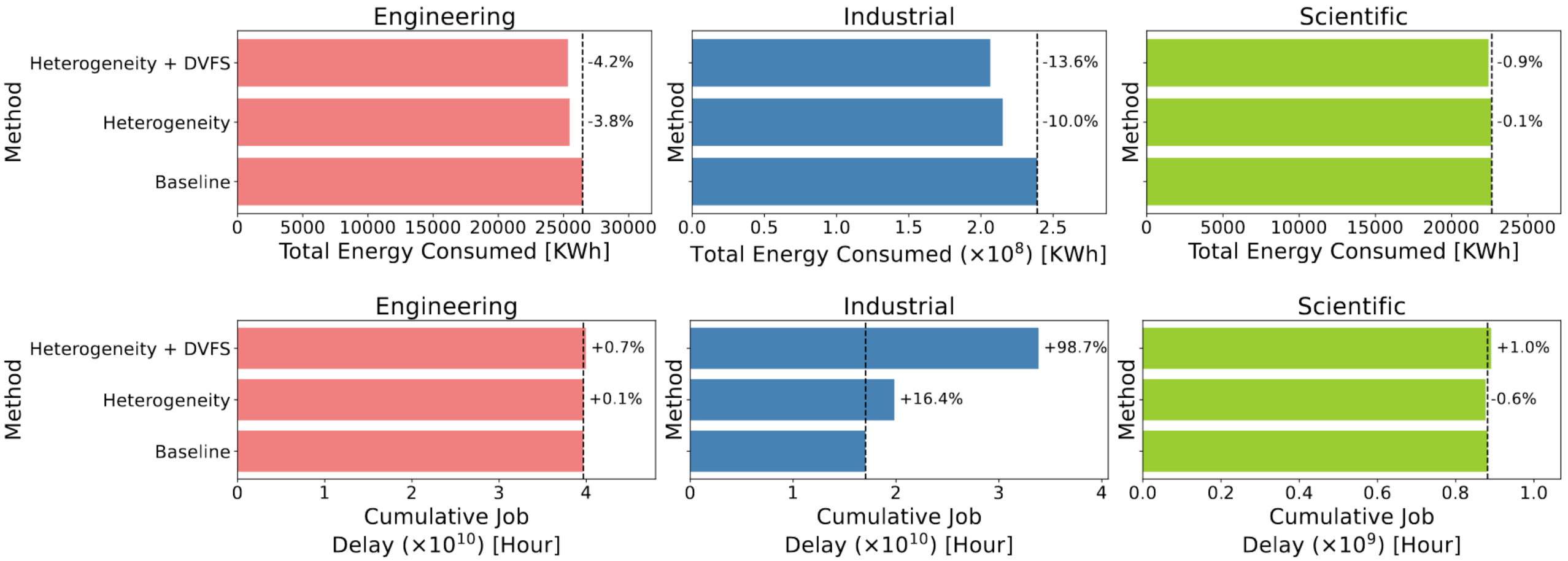
Rough Number Results

- We investigated three approaches:
  1. The impact of heterogeneity (fastest vs. most power-efficient system).
  2. The impact of DVFS.
  3. Exploit heterogeneity and apply DVFS on leftover slack, if any.
- Energy reduction results of the third configuration:
  - Scientific: -18.45%
  - Engineering: -32.11%
  - Industrial: -47.00%

Simulation

- Now that we know our idea could work, we implement the idea in a task placement policy called TaskFlow.
- TaskFlow:
  - Selects tasks in a First Come, First Serve (FCFS) manner.
  - Performs backfilling, i.e., loop through the queue in case the task at the head of the queue doesn't fit.
  - Delays tasks via heterogeneity and additionally with DVFS (optional) within the slack bound.
    - Avoids impacting the makespan: improve adoptability of the policy.
- Simulation allows us to investigate the cascading effects of delaying tasks, making the results more realistic, and process large workloads in little time using few resources.
  - Simulation using a modified OpenDC (https://opendc.org/) version.
  - Workflow traces from the Workflow Trace Archive used.
  - Similar assumptions as our rough number section, see article for details.
- We compare TaskFlow against a throughput-oriented task placement policy.
  - Puts tasks FCFS on the fastest resource available, does not apply DVFS.
  - Acts as a baseline.

Simulation Results Look Promising



Conclusion: using slack to reduce power consumption looks worth following up, yet need to deal with task starvation as observed with the industrial workflows.

Future Work

- Performance of TaskFlow on a real system.
- Comparison with State-of-the-Art (perhaps it can be combined with an approach?)
- We hope to see the community finds novel ways to exploit slack.
- Ideas for future work:
  - Mitigate Stragglers by running speculative copies of tasks
  - Reduce deadline violations by scheduling a task in a “slack-gap”.
  - Improve TaskFlow by using an anti-starvation approach.
  - A brokering approach to trade “your” job/task’s slack for some (financial?) benefits.
- Feel free to reach out to the AtLarge research group - https://atlarge-research.com/

TaskFlow: An Energy- and Makespan-Aware Task Placement Policy for Workflow Scheduling through Delay Management.

Laurens Versluis & Alexandru Iosup

Vrije Universiteit Amsterdam

2022-04-09

References

[1] Versluis, L., & Iosup, A. (2021). A survey of domains in workflow scheduling in computing infrastructures: Community and keyword analysis, emerging trends, and taxonomies. *Future Generation Computer Systems*, *123*, 156-177.

[2] Versluis, L., Van Eyk, E., & Iosup, A. (2018, April). An analysis of workflow formalisms for workflows with complex non-functional requirements. In *Companion of the 2018 ACM/SPEC International Conference on Performance Engineering* (pp. 107-112).

[3] Coffman, E. G., & Graham, R. L. (1972). Optimal scheduling for two-processor systems. *Acta informatica*, *1*(3), 200-213.

[4] Li, Z., Ge, J., Hu, H., Song, W., Hu, H., & Luo, B. (2015). Cost and energy aware scheduling algorithm for scientific workflows with deadline constraint in clouds. *IEEE Transactions on Services Computing*, *11*(4), 713-726.

[5] Sanders, P., Mehlhorn, K., Dietzfelbinger, M., & Dementiev, R. (2019). *Sequential and Parallel Algorithms and Data Structures*. Springer.

[6] Versluis, L., Mathá, R., Talluri, S., Hegeman, T., Prodan, R., Deelman, E., & Iosup, A. (2020). The workflow trace archive: Open-access data from public and private computing infrastructures. *IEEE Transactions on Parallel and Distributed Systems*, *31*(9), 2170-2184.

[7] Dhiman, G., Pusukuri, K. K., & Rosing, T. (2008). Analysis of dynamic voltage scaling for system level energy management. USENIX HotPower, 8.

Rough Number Results

- Compare fastest with most power efficient machine from the previous table.
- Apply DVFS level so that task is delayed within slack bounds.

Average Energy Reduction per Domain using Heterogeneity

Domain	Engineering	Industrial	Scientific
Energy reduction	28.31%	41.61%	16.68%
Overall average		41.47%	

Average Energy Reduction per Domain using DVFS

Domain	Engineering	Industrial	Scientific
Energy reduction	7.62%	11.35%	4.24%
Overall average		11.31%	

Average Energy Reduction per Domain using Heterogeneity and DVFS

Domain	Engineering	Industrial	Scientific
Energy reduction	32.11%	47.00%	18.45%
Overall average		46.85%	