Modeling, simulation and analysis of job scheduling policy changes on supercomputers

João Pedro M. N. dos Santos
Supervisor: Antônio Tadeu A. Gomes

Laboratório Nacional de Computação Científica

Project “Apoio a Usuários e Projetos de Supercomputação do Estado do Rio de Janeiro no Uso Eficiente do Ambiente Computacional Multiusuário Santos Dumont”

FAPERJ
Fundação Carlos Chagas Filho de Amparo à Pesquisa do Estado do Rio de Janeiro

Introduction

Motivation

Topics on HPC usage

- HPC systems plays a pivotal role in advancing research and development across diverse scientific and engineering domains.
- Supercomputers frequently represent a significant financial investment.
- The efficient utilization of these supercomputers becomes critical.
- Sterling, Anderson, and Brodowicz 2018.

Motivation

Job schedulers

- Are resource management subsystems (RMS) available in HPC systems.
- These subsystems implement *scheduling policies* that allow managing and optimizing the allocation of computational resources to user tasks (*jobs*).

Motivation

Workload characteristic

- High heterogeneity of jobs.
- This heterogeneity has made job schedulers increasingly complex to set up.
- It is not uncommon for job scheduling policies to be revisited regularly (Gomes 2018).

Motivation

Workload analysis

- Several pieces of recent work have been developed to analyze the workload of supercomputers (Feng et al. 2018; Gomes 2018; Hart 2022; Jakobsche, Lachiche, and Ciorba 2023; Rodrigo, Östberg, et al. 2018; Wang, Shen, and Li 2021).
- One aspect largely uncovered in the surveyed work is the effect of job scheduling policy changes in both the job input variables and the system output variables.

Motivation

Workload simulation

- Many pieces of work have developed RMS simulators (Dutot et al. 2017; Rodrigo, Elmroth, et al. 2018; Jokanovic, D'Amico, and Corbalan 2018; Galleguillos et al. 2020; Klusáček, Soysal, and Suter 2020; Simakov et al. 2022).
- Fidelity in representing *user behavior* in RMS simulators should be as important as representing *system behavior* though.
- Variations in policy during an **optimization process** may lead to valid jobs in the workload traces being incompatible with the newly generated policy.

Motivation

Objectives

- Propose a methodology for exploring the configuration space of job scheduling in supercomputers.
- This methodology combines two complementary approaches:
  1. analyzing the effects of scheduling policy changes on user and system behavior, and
  2. accommodating jobs incompatible with scheduling policy variations during the simulation process (*job-shaping*).
- By integrating empirical analysis and simulation, this work provides some of the underlying constructs needed for future simulation-based optimization frameworks.

Methodology

Proposed methodologies

Exploring the configuration space of job scheduling polices

- The *exploratory analysis*—we aim at assessing the impact of job scheduling policy changes in relation to user behavior and system response.
- The *workload-based, discrete event simulations*—we propose our own RMS simulator aiming at both *accuracy* and *efficiency*.

Proposed methodology

Data collection - workload analysis

- Throughout the whole period, submissions from 130 different projects were observed.
- 57 present in both BC and AC periods.
- The projects present in both periods represent 85% of the CPU time consumed in the queues considered, totaling 63,101 submissions.
- 34,187 jobs (54.18%) were submitted in the BC period and 28,914 (45.82%) in the AC period.

Proposed methodology

Data collection - workload simulation

- Only job submissions to queues with thin nodes (`cpu`, `cpu_long`, `cpu_scal`, and `cpu_small`, this last one created in the AC period) were considered .
- A total of 68,845 job records were initially considered in the simulation.
- In the BC period, there are 30,586 (44%) job records, and in the AC period, 38,259 jobs records (56%). Each period is composed by an interval of 334 days.

Proposed methodology

Exploratory analysis

- When submitting a job, the user specifies the *expected job geometry* **(GEO)**: the product of the number of requested CPU cores and the wall-clock time estimation.
- The user behavior may be also characterized by the interarrival time **(IAT)** between job submissions.
- From the system's point of view, it is possible to observe the waiting time in queue **(QWT)** of submitted jobs.

Proposed methodology

Exploratory analysis

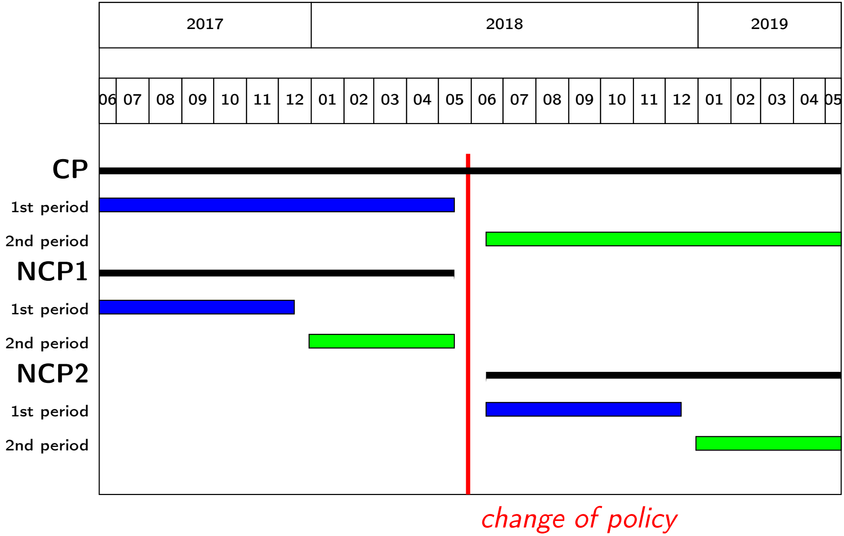
- Has there been a change in user behavior after a change in the policy?
- If there were such a change in user behavior, was it implied by the change in the policy?
- Has the system response in terms of QWT changed?

Proposed methodology

Exploratory analysis

- We analyze the empirical cumulative distribution function (ECDF) and the descriptive statistics of the measures of IAT, GEO and QWT during the adjacent time periods before and after the change in policy.

Scenarios for analysis of ECDFs of IAT, GEO and QWT



Proposed methodology

Exploratory analysis

- We employed the two-sample Kolmogorov–Smirnov (KS) statistical test for each of the measures to compare the ECDF of the 1st period ($ECDF_{1st}(\cdot)$) to the ECDF of the 2nd period ($ECDF_{2nd}(\cdot)$).

Proposed methodology

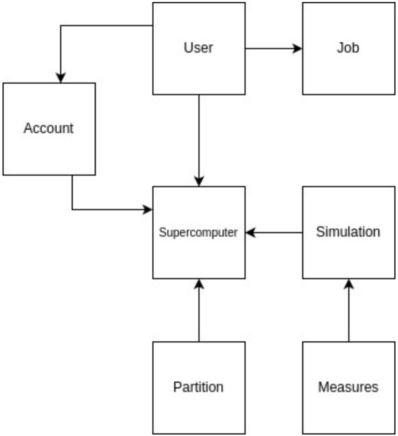
Workload-based, discrete event simulations

- We used SimPy[1] as our discrete event simulation engine.
- Our RMS simulator was designed based on entities that we have implemented over SimPy to compose a typical HPC system:[2] **Supercomputer**, **Partition**, **Account**, **User** and **Job**.

[1] `https://simpy.readthedocs.io/`

[2] The code is available at `https://gitlab.com/itdf/hpc_sim`

Proposed methodology



(a) Domain model of our RMS simulator

RMS queue configuration

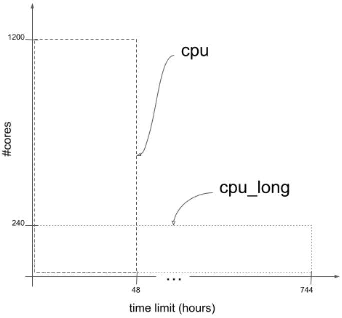
Scenarios 1

- cpu, which allows jobs allocating from 1 to $1,200$ CPU cores with a maximum time limit of 48 hours; and
- `cpu_long`, which allows jobs allocating from 1 to 240 CPU cores with a maximum time limit of 744 hours (31 days).

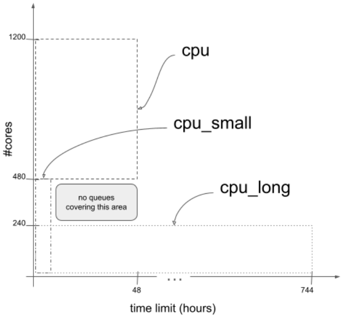
Scenarios 2

- modification of the minimum amount of allocatable CPU cores for the queue cpu to 480; and
- creation of a third queue: `cpu_small`, which allows jobs allocating from 1 to 480 CPU cores with a maximum time limit of 2 hours.

RMS queue configuration



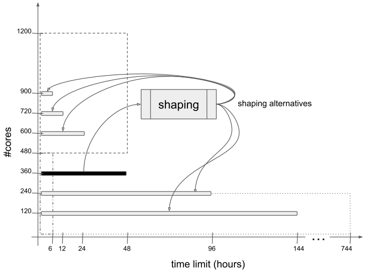
(b) Queues before change



(c) Queues after change

Example of a geometry area uncovered by a change in policy.

Job shaping strategy



(a) Examples of job shaping

Job shaping is a strategy for bringing jobs from real workload traces into compliance with policies being tested as part of an optimization of an RMS configuration.

Job shaping strategy

Remarks

- The exploratory analysis enabled the discovery of expected behaviors for the simulation when compared with real data.
- We observed that the job-shaping strategy allows capturing effects present in the workload while varying queue configurations.

Workload analysis

Two-sample KS tests for IAT and GEO

IAT

Scenario	Hypotheses two-sided	less	greater
CP	H_1	H_1	H_1
NCP1	H_1	H_1	H_1
NCP2	H_1	H_0 (p-value = 1.0)	H_1

GEO

Scenario	Hypotheses two-sided	less	greater
CP	H_1	H_1	H_1
NCP1	H_1	H_1	H_1
NCP2	H_1	H_0 (p-value = 0.98)	H_1

Change in user behavior **not generally** due to change in policy

- Multifactorial behavior
- Influence of external variables difficult to filter out
  - conference deadlines, vacations, etc

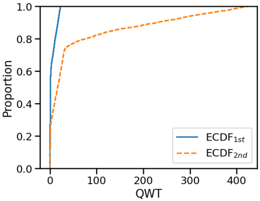
Two-sample KS tests for QWT

Scenario	Hypotheses two-sided	less	greater
CP	H_1	H_1	H_1
NCP1	H_1	H_0 (p-value $= 0.99$)	H_1
NCP2	H_1	H_1	H_1

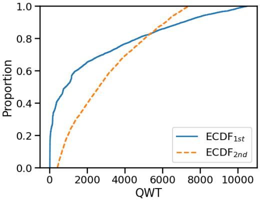
Unexpected rejection of H_0 for less hypothesis in CP scenario

- Not in agreement with our previous results presented in Gomes 2018

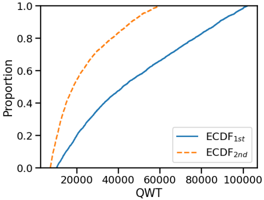
A closer look at the ECDFs of QWT in the CP scenario



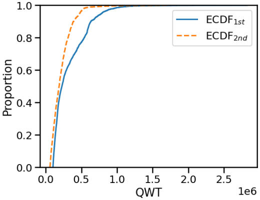
(a) Q1



(b) Q2



(c) Q3



(d) Q4

Long* × *short

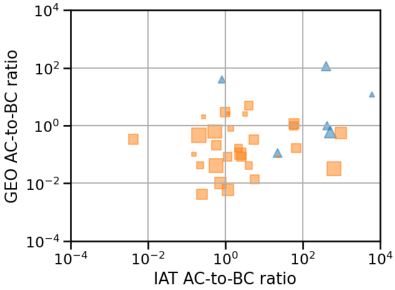
5,360-sec
"threshold"

AC-to-BC ratios within projects

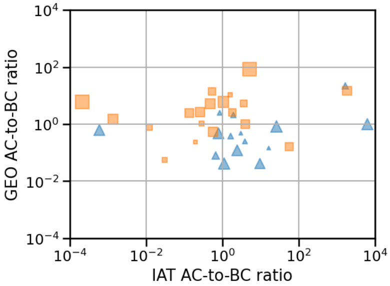
AC-to-BC ratio of a measure

- Median of the measure in the AC period divided by the median of the measure in the BC period
- A project with a decrease (resp. increase) in a specific measure—as expressed by its median—after the change in policy will have an AC-to-BC ratio less (resp. greater) than 1

AC-to-BC ratios within projects



(e) jobs with short QWT



(f) jobs with long QWT

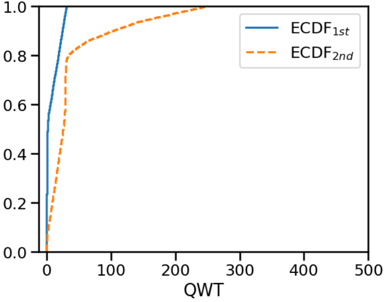
Log-log plot of AC-to-BC ratios in the medians of IAT and GEO. Each data point—of size proportional on $\log_{10}$ scale to the AC-to-BC ratio of the QWT—is plotted as a **triangle** (resp. **square**) if the ratio is less (resp. greater) than 1.

Workload simulation

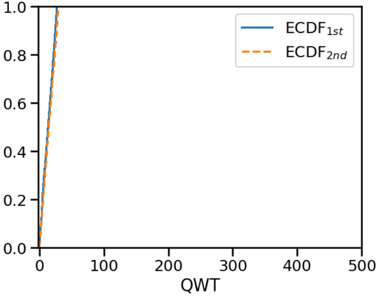
Comparing real data analysis with simulated results

- The statistics about inputs show that it had no changes, as expected.
- Both workload periods were simulated in ≈ 10 minutes in a single-core python kernel.
- We then compare the ECDFs of the QWT obtained through real data and through the simulation.

ECDFs of the QWT - Q1

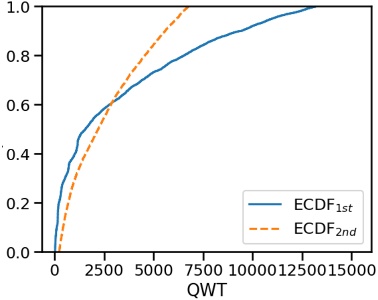


(a) Q1 - real

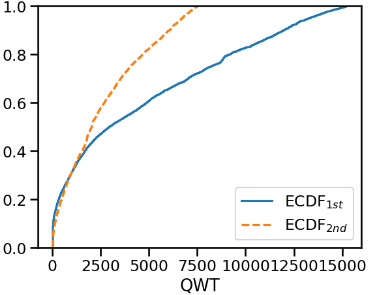


(b) Q1 - simulated

ECDFs of the QWT - Q2

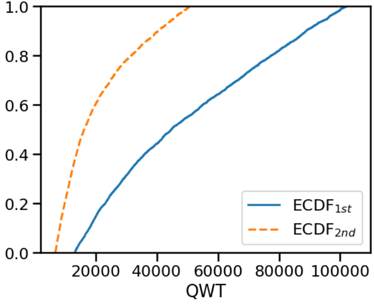


(c) Q2 - real

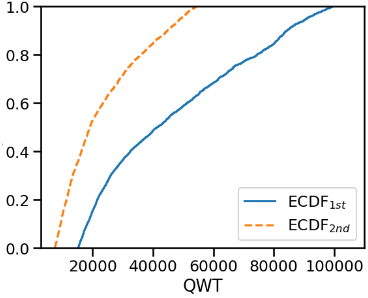


(d) Q2 - simulated

ECDFs of the QWT - Q3

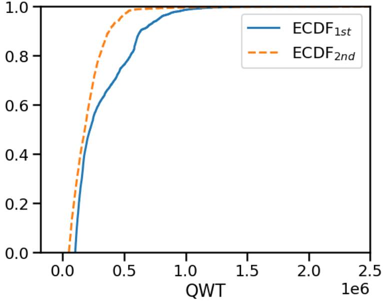


(e) Q3 - real

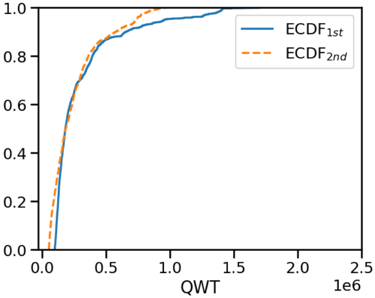


(f) Q3 - simulated

ECDFs of the QWT - Q4



(g) Q4 - real



(h) Q4 - simulated

BC workload into AC configuration

- We identified that the system behavior, measured by QWT, was not captured from the BC period workload simulation with the AC period configuration.
- This was because some of the jobs did not fit the new policy.

Workload processing with job shaping

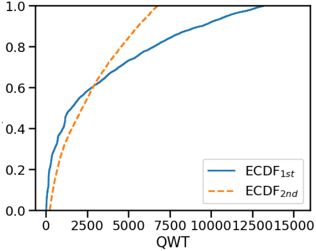
- We assessed two scenarios of shaping these unprocessed jobs considering new resource management policies:
  1. Doubling the requested amount of cores and halving the time limit for the `cpu` queue;
  2. Halving the requested amount of cores and doubling the time limit for the `cpu_long` queue.

Comparison between the ECDF of real and histogram sampling simulated data

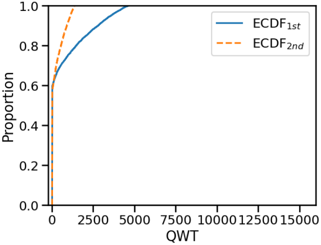
Findings

- We observed a high variability in the system's response.
- Effects existing in the source workload trace were captured by the simulation.

ECDFs of the QWT - Q2

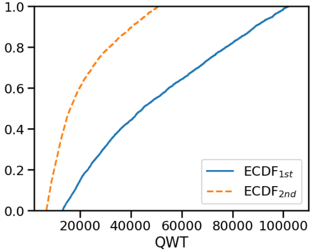


(i) Q2 – real

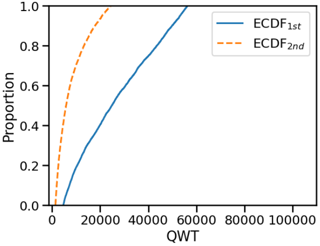


(j) Q2 – histogram sampling simulated

ECDFs of the QWT - Q3



(k) Q3 – real



(l) Q3 – histogram sampling simulated

Conclusion

Main findings

- **Nuanced effects** of scheduling adjustments
- Benefits of **"affinity"-based interpretation** of behavioral changes
  - Inline with Rodrigo, Östberg, et al.
- Job shaping allows capturing changes in system behavior simply and effectively.
- Parameterizing the job shaping strategy allows for an in-depth evaluation of optimization opportunities, offering insights into how configurations impact overall system performance.

Takeaways

- Need for **adaptive, data-driven scheduling algorithms** to meet the evolving demands of HPC-intensive R&D
  - C.f. Carastan-Santos and Camargo; Gaussier et al.; Tsafrir, Etsion, and Feitelson

Limitations and future work

- Take **system utilization** into account
- Explore other factors than project membership (e.g. shared nodes)
- Evolve the job shaping technique, allowing the evaluation of more complex scenarios, e.g. involving workflows.

Accepted papers

Impact of job scheduling policy changes on user behavior and system response: The case of the Santos Dumont supercomputer in Brazil.

A job shaping strategy to accommodate workload traces under varying resource management policies.

Thank you!

João Pedro M. N. dos Santos, Antônio Tadeu A. Gomes
macleure@posgrad.lncc.br, atagomes@lncc.br

Laboratório Nacional de Computação Científica

FAPERJ
Fundação Carlos Chagas Filho de Amparo à Pesquisa do Estado do Rio de Janeiro

References

- Carastan-Santos, Danilo and Raphael Y. de Camargo. "Obtaining Dynamic Scheduling Policies with Simulation and Machine Learning". In: *SC17: Int'l Conference for High Performance Computing, Networking, Storage and Analysis*. 2017.
- Dutot, Pierre-François et al. "Batsim: A Realistic Language-Independent Resources and Jobs Management Systems Simulator". In: *Job Scheduling Strategies for Parallel Processing*. Ed. by Narayan Desai and Walfredo Cirne. Cham: Springer International Publishing, 2017, pp. 178–197. ISBN: 978-3-319-61756-5.
- Feng, Jinghua et al. "Workload Characterization and Evolutionary Analyses of Tianhe-1A Supercomputer". In: *Computational Science – ICCS 2018*. Ed. by Yong Shi et al. Cham: Springer Int'l Publishing, 2018, pp. 578–585. ISBN: 978-3-319-93698-7.

- Galleguillos, Cristian et al. "AccaSim: a customizable workload management simulator for job dispatching research in HPC systems". In: *Cluster Computing* 23.1 (Mar. 2020), pp. 107–122. ISSN: 1573-7543. DOI: 10.1007/s10586-019-02905-5. URL: https://doi.org/10.1007/s10586-019-02905-5.
- Gaussier, Eric et al. "Improving backfilling by using machine learning to predict running times". In: *SC15: Int'l Conference for High Performance Computing, Networking, Storage and Analysis*. 2015.
- Gomes, Antônio Tadeu Azevedo. *Assessing the behavior of HPC users and systems: The case of the Santos Dumont supercomputer*. Lecture of the XIX Brazilian Symposium on High-Performance Computing Systems (WSCAD), São Paulo, Brazil. Oct. 2018. URL: http://wscad.sbc.org.br/2018/programa.html.

- Hart, David. "Measuring XSEDE: Usage Metrics for the XSEDE Federation of Resources: A comparison of evolving resource usage patterns across TeraGrid and XSEDE". In: *Practice and Experience in Advanced Research Computing*. Boston, MA, USA: Association for Computing Machinery, 2022. ISBN: 9781450391610. DOI: 10.1145/3491418.3530765. URL: https://doi.org/10.1145/3491418.3530765.
- Jakobsche, Thomas, Nicolas Lachiche, and Florina M. Ciorba. "Investigating HPC Job Resource Requests and Job Efficiency Reporting". In: *22nd Int'l Symposium on Parallel and Distributed Computing (ISPDC)*. 2023.
- Jokanovic, Ana, Marco D'Amico, and Julita Corbalan. "Evaluating SLURM Simulator with Real-Machine SLURM and Vice Versa". In: *2018 IEEE/ACM Performance Modeling, Benchmarking and Simulation of High Performance Computer Systems (PMBS)*. IEEE, 2018, pp. 72–82.

Klusáček, Dalibor, Mehmet Soysal, and Frédéric Suter. "Alea – Complex Job Scheduling Simulator". In: *Parallel Processing and Applied Mathematics*. Ed. by Roman Wyrzykowski et al. Cham: Springer International Publishing, 2020, pp. 217–229.

Rodrigo, Gonzalo P., Erik Elmroth, et al. "ScSF: A Scheduling Simulation Framework". In: *Job Scheduling Strategies for Parallel Processing*. Ed. by Dalibor Klusáček, Walfredo Cirne, and Narayan Desai. Cham: Springer International Publishing, 2018, pp. 152–173. ISBN: 978-3-319-77398-8.

Rodrigo, Gonzalo P., P.-O. Östberg, et al. "Towards understanding HPC users and systems: A NERSC case study". In: *Journal of Parallel and Distributed Computing* 111 (2018), pp. 206–221. ISSN: 0743-7315. DOI: https://doi.org/10.1016/j.jpdc.2017.09.002. URL: https://www.sciencedirect.com/science/article/pii/S0743731517302563.

Simakov, Nikolay A. et al. "Developing Accurate Slurm Simulator". In: *Practice and Experience in Advanced Research Computing 2022: Revolutionary: Computing, Connections, You*. Association for Computing Machinery, 2022.

Sterling, Thomas, Matthew Anderson, and Maciej Brodowicz. "Chapter 5 - The Essential Resource Management". In: *High Performance Computing*. Ed. by Thomas Sterling, Matthew Anderson, and Maciej Brodowicz. Boston: Morgan Kaufmann, 2018, pp. 141–190. ISBN: 978-0-12-420158-3. DOI: https://doi.org/10.1016/B978-0-12-420158-3.00005-8. URL: https://www.sciencedirect.com/science/article/pii/B9780124201583000058.

Tsafrir, Dan, Yoav Etsion, and Dror G. Feitelson. "Backfilling Using System-Generated Predictions Rather than User Runtime Estimates". In: *IEEE Transactions on Parallel and Distributed Systems* 18.6 (2007), pp. 789–803. DOI: 10.1109/TPDS.2007.70606.

Wang, Qiqi, Yu Shen, and Jing Li. "User-level Workload Analysis for Supercomputers". In: *4th Int'l Conference on Software Engineering and Information Management*. Yokohama, Japan: Association for Computing Machinery, 2021, pp. 68–73. ISBN: 9781450388955. DOI: 10.1145/3451471.3451483. URL: https://doi.org/10.1145/3451471.3451483.