Quantum Optimization Benchmarking Library

The Intractable Decathlon

Dr. Stefan Woerner
Principal Research Scientist
Manager, Applied Quantum Science
IBM Quantum, IBM Research Europe - Zurich

IBM

Quantum Optimization Working Group

nature reviews physics

https://doi.org/10.1038/s42254-024-00770-9

Review article

Check for updates

Challenges and opportunities in quantum optimization

Amira Abbas[1,2], Andris Ambainis[3], Brandon Augustino[4], Andreas Bärtschi[5], Harry Buhrman[1,2,6], Carleton Coffrin[7], Giorgio Cortiana[8], Vedran Dunjko[9], Daniel J. Egger[10], Bruce G. Elmegreen[11], Nicola Franco[12], Filippo Fratini[13], Bryce Fuller[14], Julien Gacon[10,15], Constantin Gonciulea[16], Sander Gribling[17], Swati Gupta[4], Stuart Hadfield[18,19], Raoul Heese[20], Gerhard Kircher[13], Thomas Kleinert[21], Thorsten Koch[22,23], Georgios Korpas[24,25,26], Steve Lenk[27,28], Jakub Marecek[25], Vanio Markov[16], Guglielmo Mazzola[29], Stefano Mensa[30], Naeimeh Mohseni[8], Giacomo Nannicini[31], Corey O'Meara[8], Elena Peña Tapia[10], Sebastian Pokutta[32,33], Manuel Proissl[10], Patrick Rebentrost[34], Emre Sahin[30], Benjamin C. B. Symons[30], Sabine Tornow[35], Víctor Valls[36], Stefan Woerner[10], Mira L. Wolf-Bauwens[10], Jon Yard[37], Sheir Yarkoni[38], Dirk Zechiel[21], Sergiy Zhuk[36] & Christa Zoufal[10]

[arXiv: https://arxiv.org/abs/2312.02279]

Quantum Optimization Working Group

Quantum Optimization Benchmark Library
The Intractable Decathlon

Thorsten Koch [*1,2], David E. Bernal Neira[3], Ying Chen[4], Giorgio Cortiana[5], Daniel J. Egger[6], Raoul Heese[7], Narendra N. Hegade[8], Alejandro Gomez Cadavid[8,9], Rhea Huang[10], Toshinari Itoko[11], Thomas Kleinert[12], Pedro Maciel Xavier[3,13], Naeimeh Mohseni[5], Jhon A. Montanez-Barrera[14], Koji Nakano[15], Giacomo Nannicini[10], Corey O'Meara[5], Justin Pauckert[16], Manuel Proissl[6], Anurag Ramesh[3], Maximilian Schicker[1], Noriaki Shimada[11], Mitsuharu Takeori[11], Víctor Valls[17], David Van Bulck[18], Stefan Woerner [†6], and Christa Zoufal [‡6]

[1] *Zuse Institute Berlin*
[2] *Technische Universität Berlin*
[3] *Davidson School of Chemical Engineering, Purdue University*
[4] *National University of Singapore*
[5] *E.ON Digital Technology GmbH*
[6] *IBM Quantum, IBM Research Europe – Zurich*
[7] *NTT Data*
[8] *Kipu Quantum GmbH*
[9] *University of the Basque Country UPV/EHU*
[10] *University of Southern California*
[11] *IBM Quantum, IBM Research Tokyo*
[12] *Quantagonia*
[13] *Federal University of Rio de Janeiro*
[14] *Forschungszentrum Jülich*
[15] *Hiroshima University*
[16] *T-Systems International GmbH*
[17] *IBM Quantum, IBM Research Europe – Dublin*
[18] *Ghent University*

April 4, 2025

[arXiv: https://arxiv.org/abs/2504.03832]

Goal: Demonstrating Quantum Advantage in Optimization

What is quantum advantage?

"Solving a problem faster, better, or cheaper when using quantum computing than when using classical computing alone."

Goal: Demonstrating Quantum Advantage in Optimization

What is quantum advantage?

“Solving a problem faster, better, or cheaper when using quantum computing than when using classical computing alone.”

- What problems to look into?
- How to solve them?
- What algorithms to compare to?
- How to measure speed, quality, and cost?

Personas in [Quantum Optimization] Benchmarking*

Goals in Benchmarking

Applications Benchmarking: The goal is to find the best possible algorithms–classical or quantum–to solve a given problem instance. Thus, benchmarks must be model-independent to allow all possible approaches to solve a problem. This is the only level that ultimately allows demonstrating quantum advantage.

Algorithm Benchmarking: The goal is to identify suitable strategies for setting hyper-parameters, identifying bottlenecks, improving algorithms, and tracking progress over time. Since algorithm benchmarking does not entail comparing against all possible algorithms, it will not allow demonstrating quantum advantage. Nevertheless, it can be used to estimate an algorithm's scaling, which may facilitate the identification of potential asymptotic scaling advantages and help track progress toward quantum advantage.

System Benchmarking: The goal is to identify the best way to run a fixed algorithm for a fixed problem on a given platform. This includes tuning algorithmic hyperparameters or parameters of the execution environment, e.g., for error suppression and mitigation, and to confirm that the algorithm is working as expected. It can also be used for application-centric hardware benchmarking.

*) Adapted from Finžgar et al., IEEE QCE2022

Personas in [Quantum Optimization] Benchmarking*

Goals in Benchmarking

Applications Benchmarking: The goal is to find the best possible algorithms–classical or quantum–to solve a given problem instance. Thus, benchmarks must be model-independent to allow all possible approaches to solve a problem. This is the only level that ultimately allows demonstrating quantum advantage.

Algorithm Benchmarking: The goal is to identify suitable strategies for setting hyper-parameters, identifying bottlenecks, improving algorithms, and tracking progress over time. Since algorithm benchmarking does not entail comparing against all possible algorithms, it will not allow demonstrating quantum advantage. Nevertheless, it can be used to estimate an algorithm's scaling, which may facilitate the identification of potential asymptotic scaling advantages and help track progress toward quantum advantage.

System Benchmarking: The goal is to identify the best way to run a fixed algorithm for a fixed problem on a given platform. This includes tuning algorithmic hyperparameters or parameters of the execution environment, e.g., for error suppression and mitigation, and to confirm that the algorithm is working as expected. It can also be used for application-centric hardware benchmarking.

- Focus on model-independent benchmarks
- Compare all possible ways to solve problems (classical, quantum, ...)
- Benchmarking of complete workflow

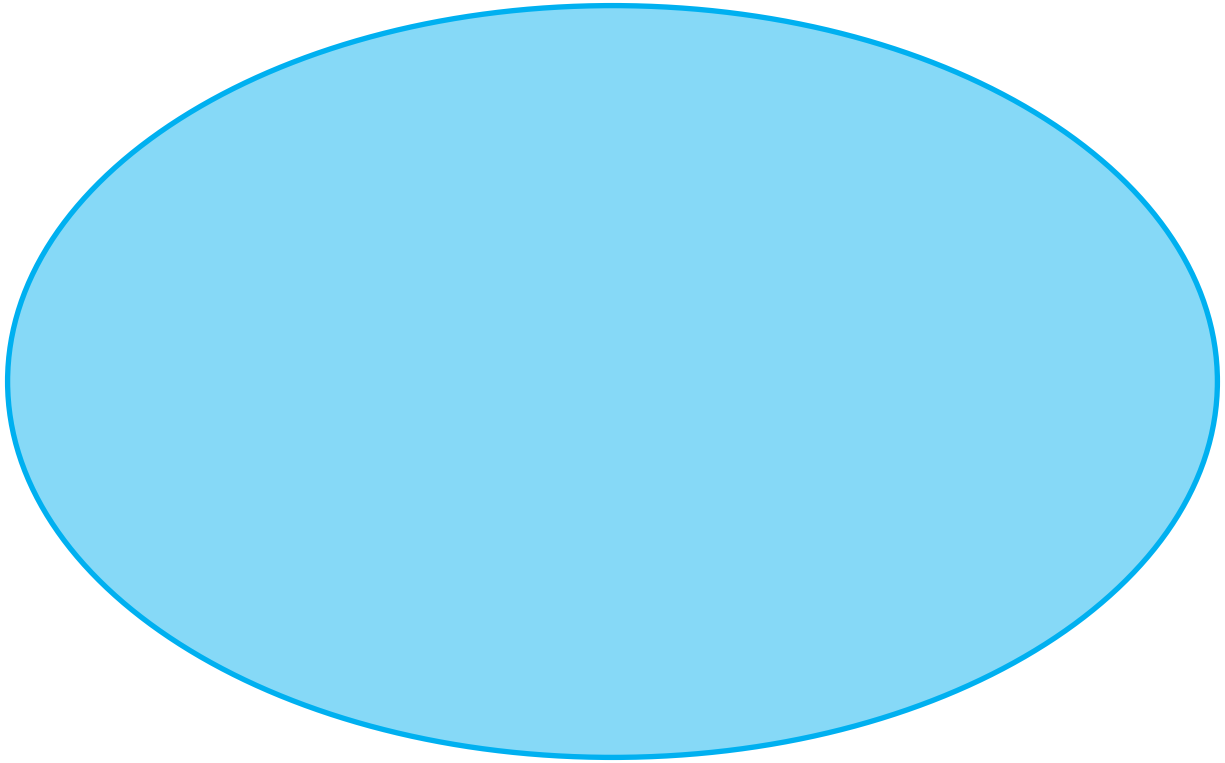
*) Adapted from Finžgar et al., IEEE QCE2022

What problems to look into?

- NP-hardness alone does not imply that a concrete instance of a problem is actually difficult to solve classically.
- Most classical & quantum optimization algorithms are heuristics, i.e., there are no a priori performance guarantees (there might be a posteriori guarantees).
- Many instances of NP-hard problems can be solved efficiently to provable global optimality using heuristics.
- Benchmarks should be run on the same problems to enable comparability of results.
- We need to empirically demonstrate their performance on concrete (difficult) problem instances.

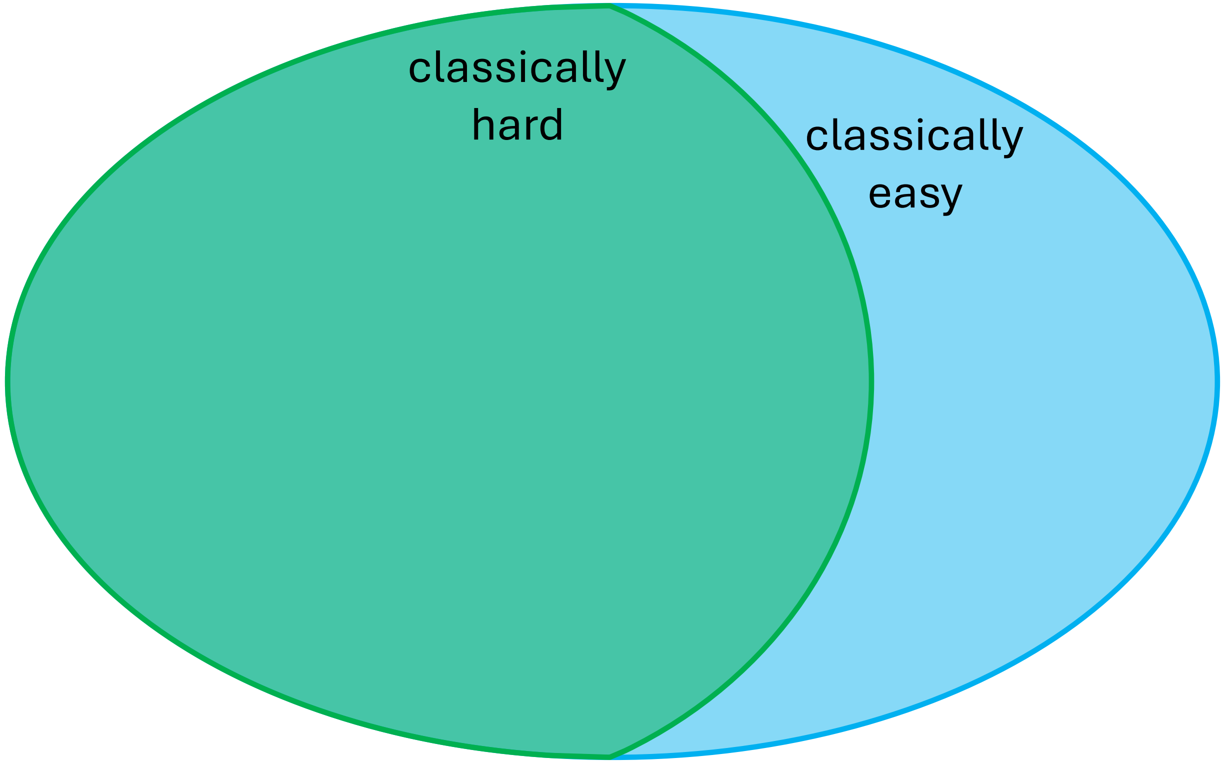
Strategies to
find quantum advantage
in optimization

Optimization Problems



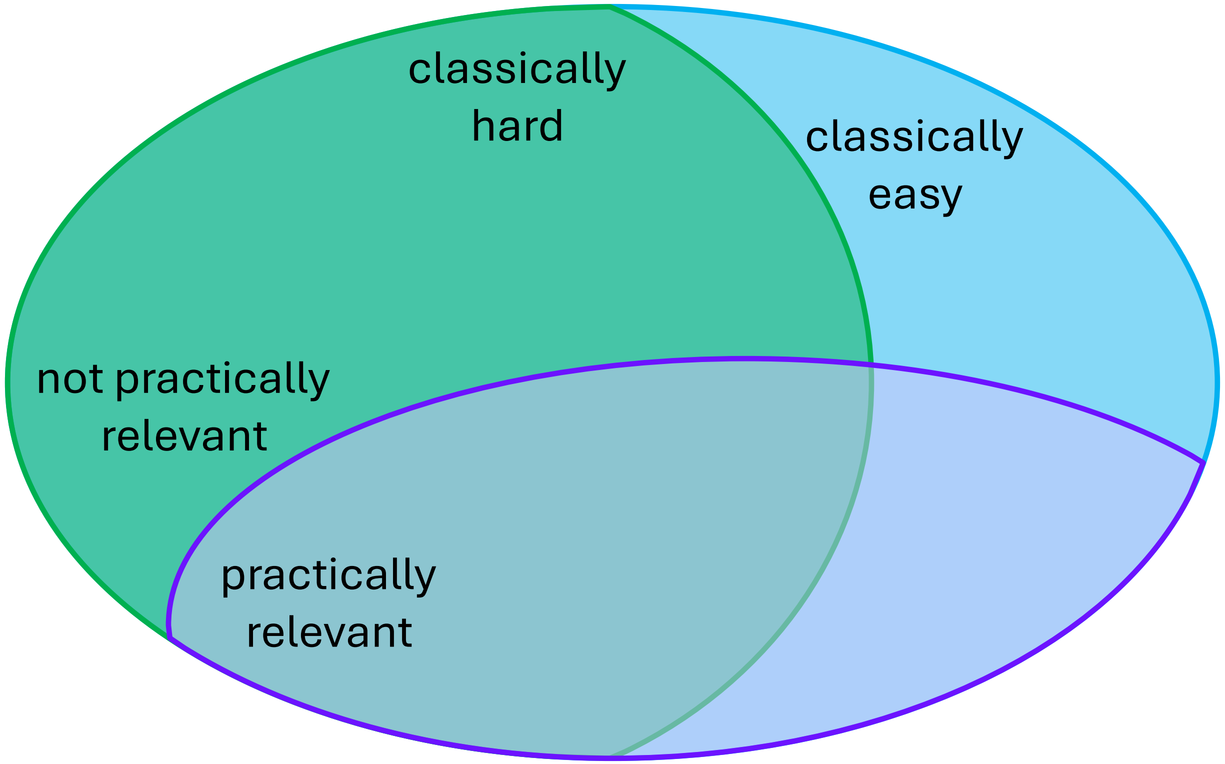
Strategies to
find quantum advantage
in optimization

Optimization Problems

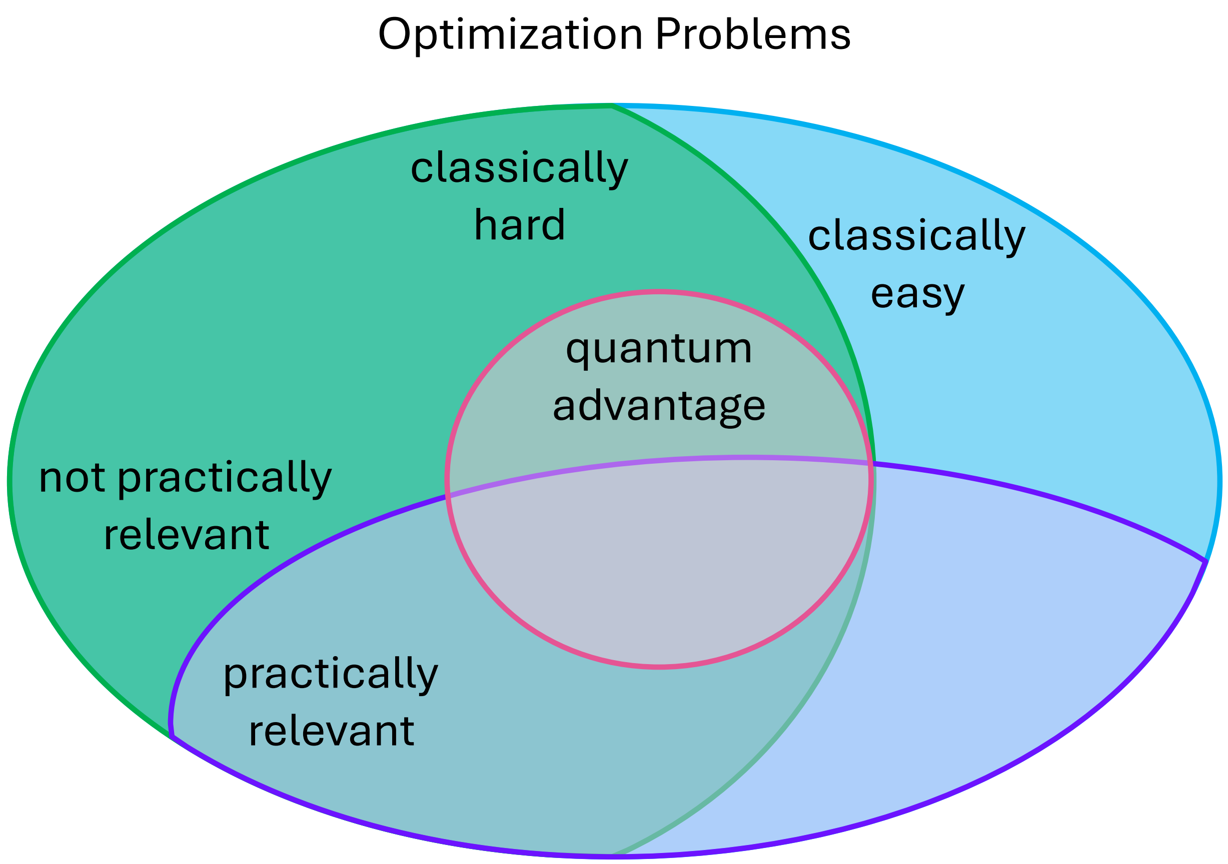


Strategies to
find quantum advantage
in optimization

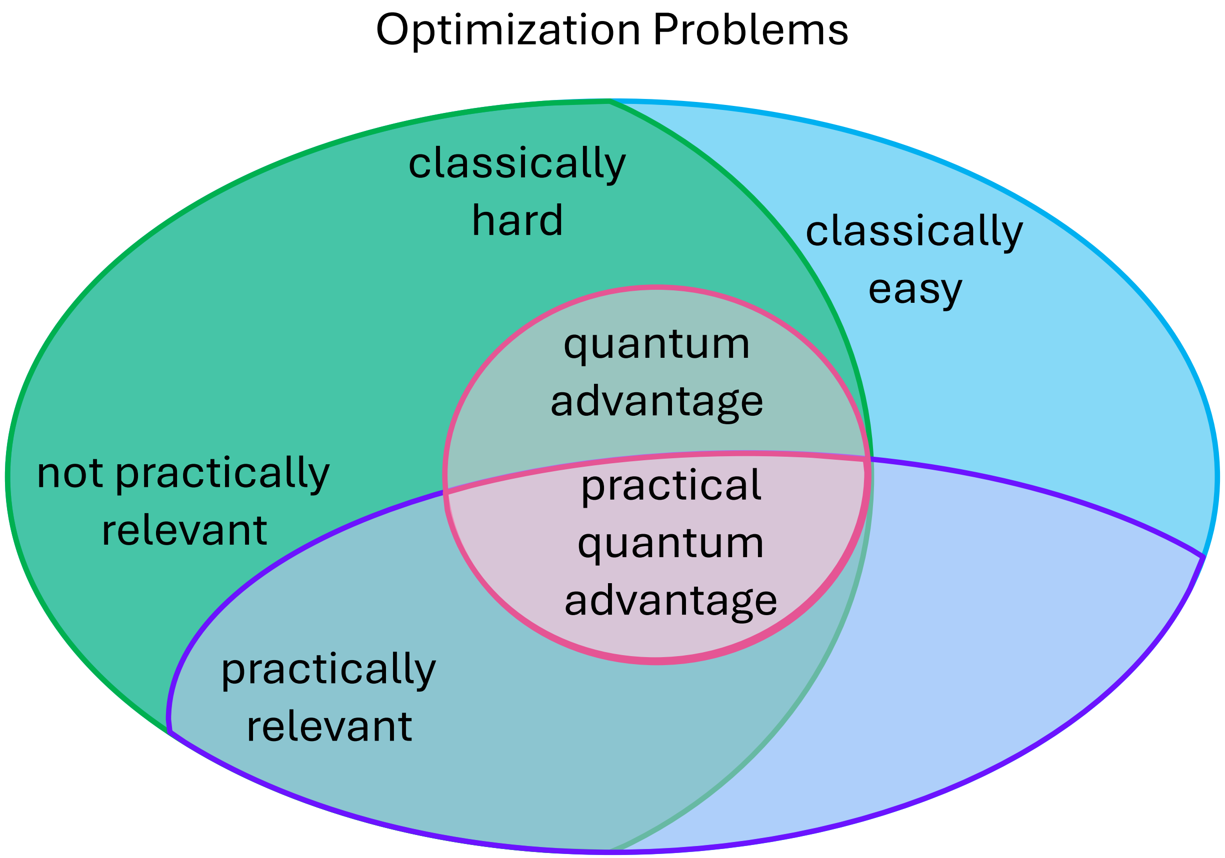
Optimization Problems



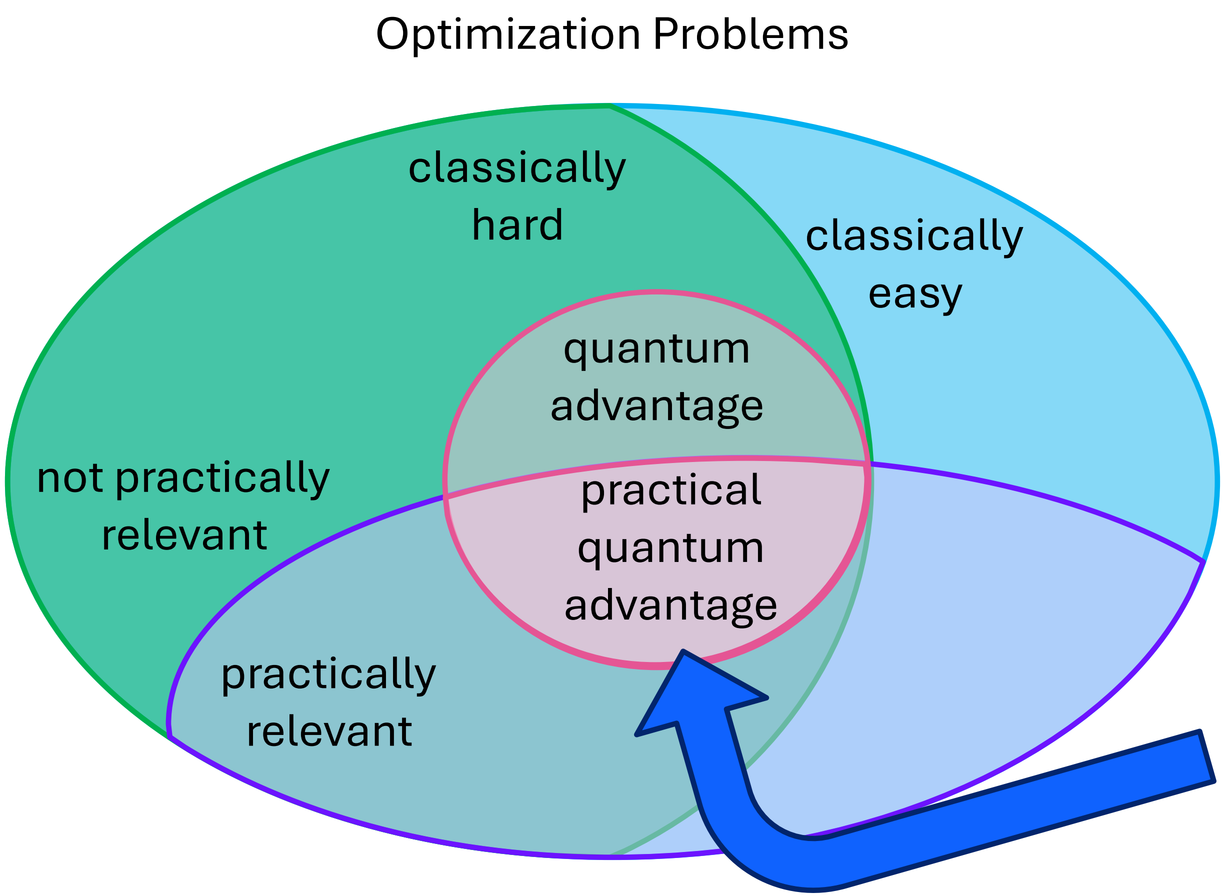
Strategies to
find quantum advantage
in optimization



Strategies to
find quantum advantage
in optimization



Strategies to
find quantum advantage
in optimization

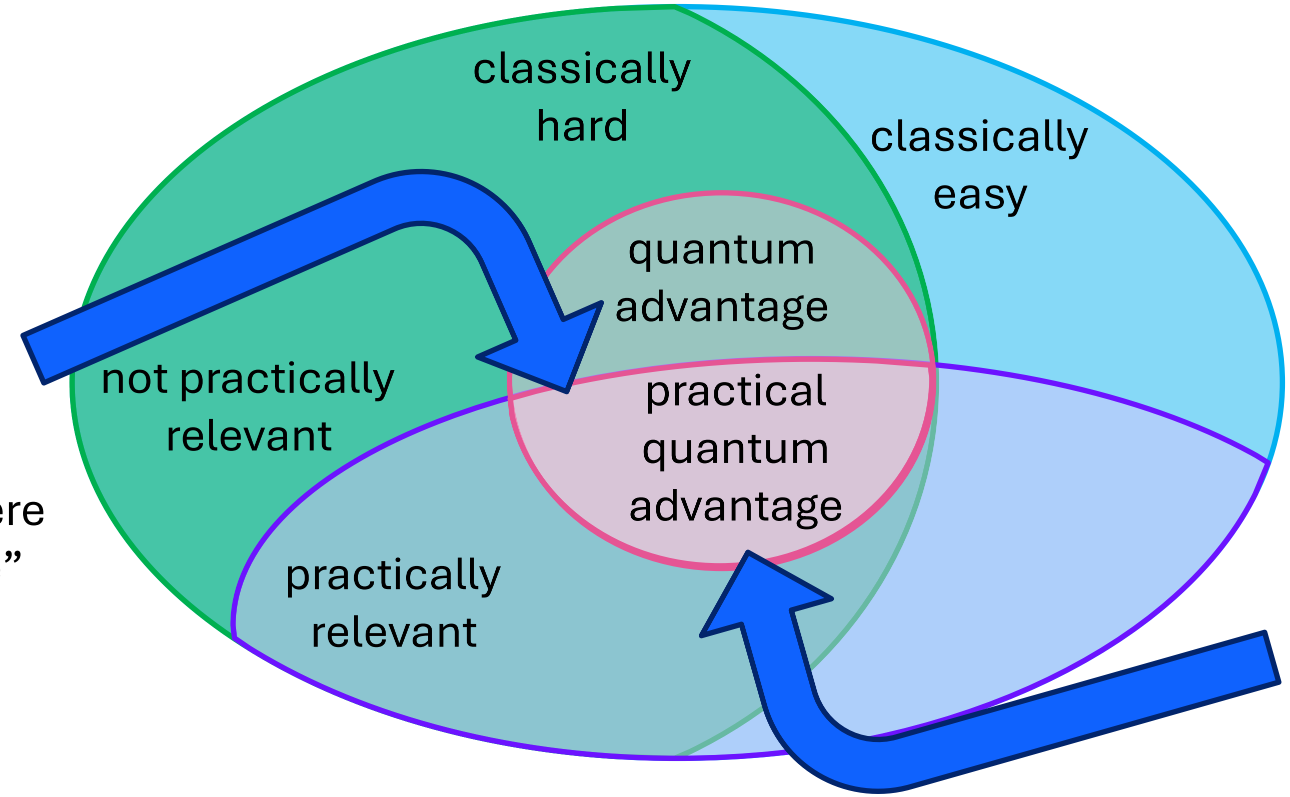


“Use Case-Driven”
Start with practically relevant problem:

- Make sure it is “classically difficult”
- Investigate whether there is “quantum advantage”

Strategies to
find quantum advantage
in optimization

Optimization Problems



“Math-Driven”
Start with classically difficult problem:

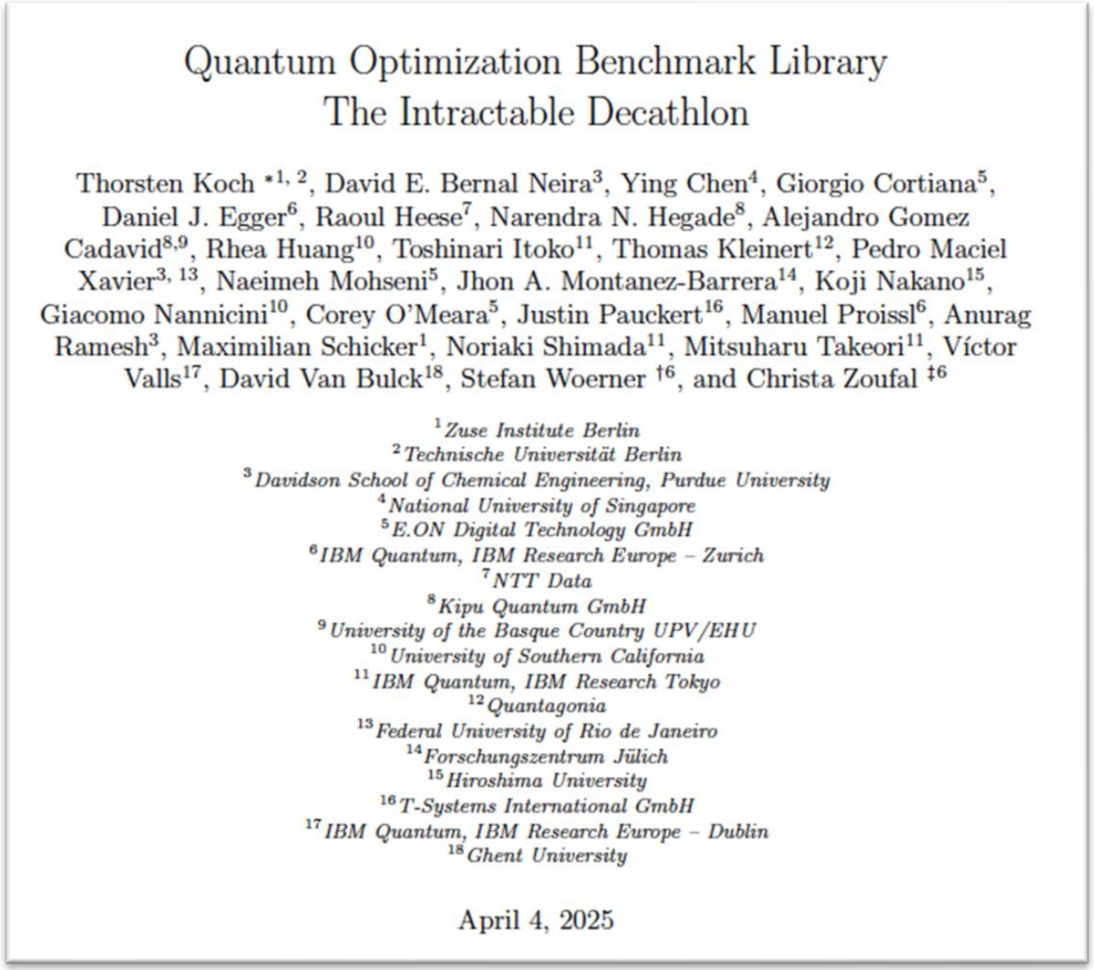
- Investigate whether there is “quantum advantage”
- Make sure it is ”practically relevant”

“Use Case-Driven”
Start with practically relevant problem:

- Make sure it is “classically difficult”
- Investigate whether there is “quantum advantage”

Quantum Optimization Benchmarking Library (QOBLIB)

https://git.zib.de/qopt/qoblib-quantum-optimization-benchmarking-library



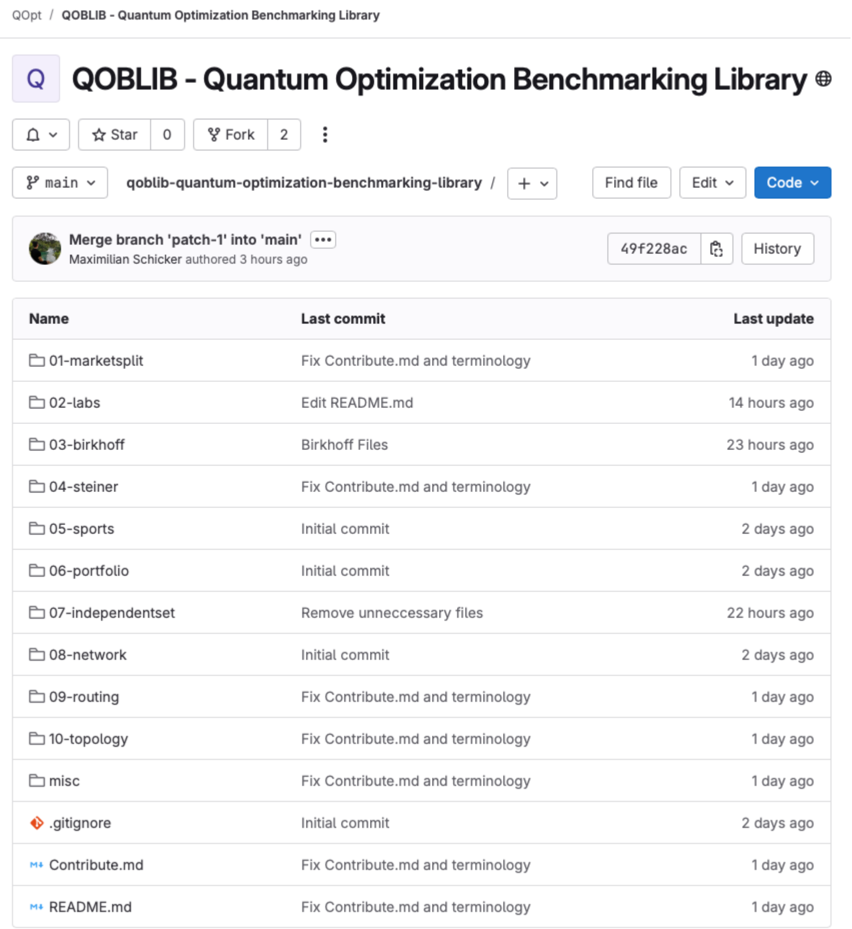
Quantum Optimization Benchmark Library
The Intractable Decathlon

Thorsten Koch *[1, 2], David E. Bernal Neira[3], Ying Chen[4], Giorgio Cortiana[5], Daniel J. Egger[6], Raoul Heese[7], Narendra N. Hegade[8], Alejandro Gomez Cadavid[8,9], Rhea Huang[10], Toshinari Itoko[11], Thomas Kleinert[12], Pedro Maciel Xavier[3, 13], Naeimeh Mohseni[5], Jhon A. Montanez-Barrera[14], Koji Nakano[15], Giacomo Nannicini[10], Corey O'Meara[5], Justin Pauckert[16], Manuel Proissl[6], Anurag Ramesh[3], Maximilian Schicker[1], Noriaki Shimada[11], Mitsuharu Takeori[11], Víctor Valls[17], David Van Bulck[18], Stefan Woerner †[6], and Christa Zoufal ‡[6]

[1] *Zuse Institute Berlin*
[2] *Technische Universität Berlin*
[3] *Davidson School of Chemical Engineering, Purdue University*
[4] *National University of Singapore*
[5] *E.ON Digital Technology GmbH*
[6] *IBM Quantum, IBM Research Europe – Zurich*
[7] *NTT Data*
[8] *Kipu Quantum GmbH*
[9] *University of the Basque Country UPV/EHU*
[10] *University of Southern California*
[11] *IBM Quantum, IBM Research Tokyo*
[12] *Quantagonia*
[13] *Federal University of Rio de Janeiro*
[14] *Forschungszentrum Jülich*
[15] *Hiroshima University*
[16] *T-Systems International GmbH*
[17] *IBM Quantum, IBM Research Europe – Dublin*
[18] *Ghent University*

April 4, 2025

[https://arxiv.org/abs/2504.03832]



→ Hosted by Zuse Institute Berlin (ZIB), which also hosts the well-known "Mixed Integer Programming Library" (MIPLIB)

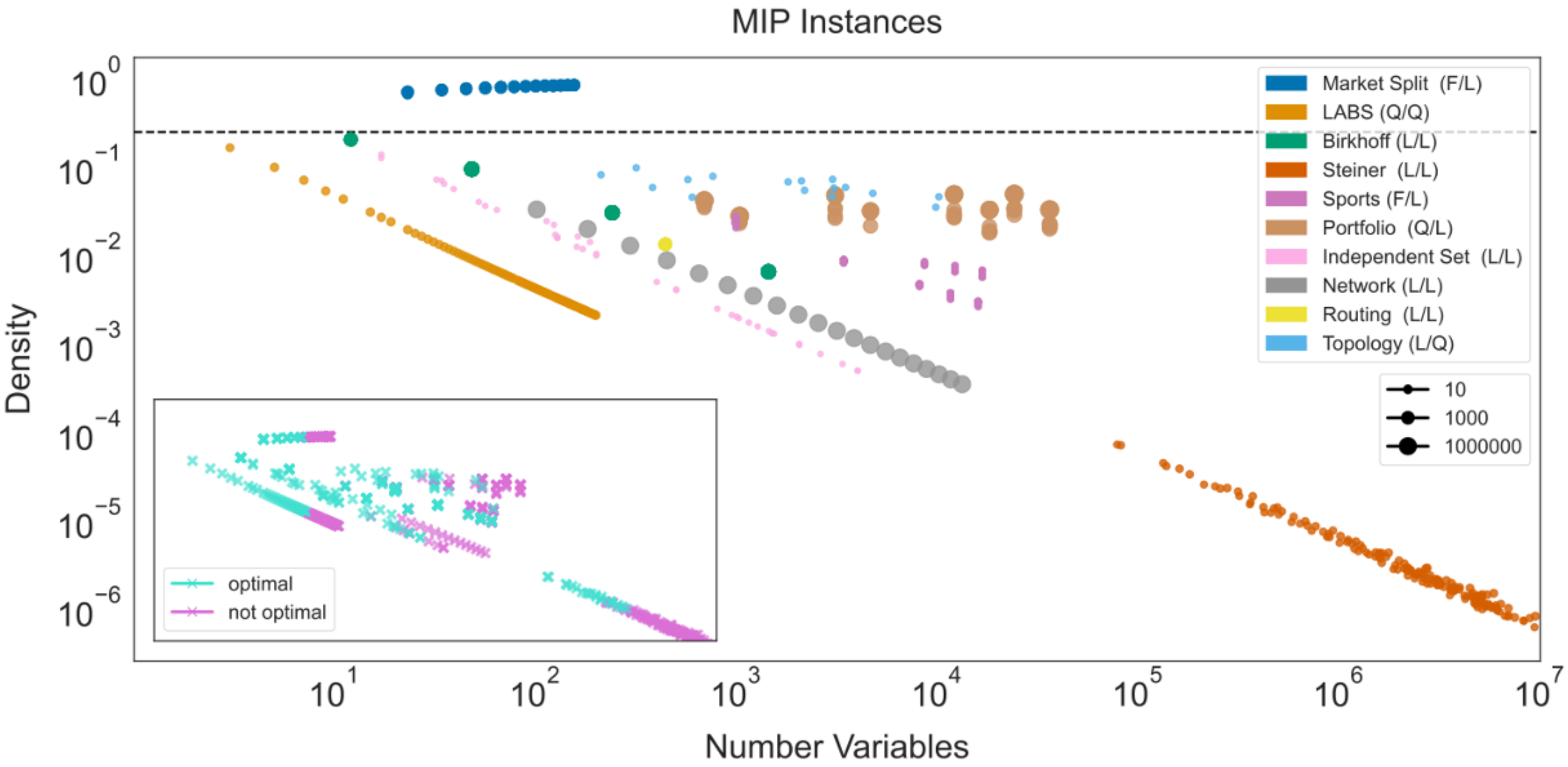
”The Intractable Decathlon”

	MIP							QUBO		
Problem	**Type**	**Dense**	**#Vars**	**Coeffs**	**Constr**	**Feas**	**Bin**	**Dense**	**#Vars**	**Coeffs**
Market Split	F / L	d	78	48	1	no	yes	d	70	$\sim 5 \cdot 10^6$
LABS	Q / Q	s	81	4	1	yes	yes	s	820	$\sim 4 \cdot 10^4$
Birkhoff	L / L	s	240	10^4	3	yes	no	d	3,480	$\sim 3 \cdot 10^{10}$
Steiner	L / L	s	423,360	3	9	no	yes	-	-	-
Sports	F / L	s	8,608	2	> 10	no	no	s	11, 791	$\sim 4.5 \cdot 10^3$
Portfolio	Q / L	s*	690	$\sim 3 \cdot 10^4$	2	yes	yes	d	690	$\sim 2 \cdot 10^9$
Independent Set	L / L	s	500	1	1	yes	yes	d	500	2
Network	L / L	s	1,211	10^6	5	yes	no	s	46,330	$\sim 2.5 \cdot 10^{19}$
Routing	L / L	s	-	-	< 10	yes	no	s	-	-
Topology	L / Q	s	2,176	2	$4 - 7$	yes	no	s	-	-

- 10 problem classes with different characteristics.
- Multiple instances per class from easy to hard to track progress over time, including hard instances that are relatively small!
- Classical baseline results for all instances, quantum baseline results for a first subset.
- Instances are expected to be “primal hard” (finding good solutions is hard) not only “dual hard” (proving optimality is hard).
- Model-independent, we provide the data, and illustrative MIP and QUBO formulations.

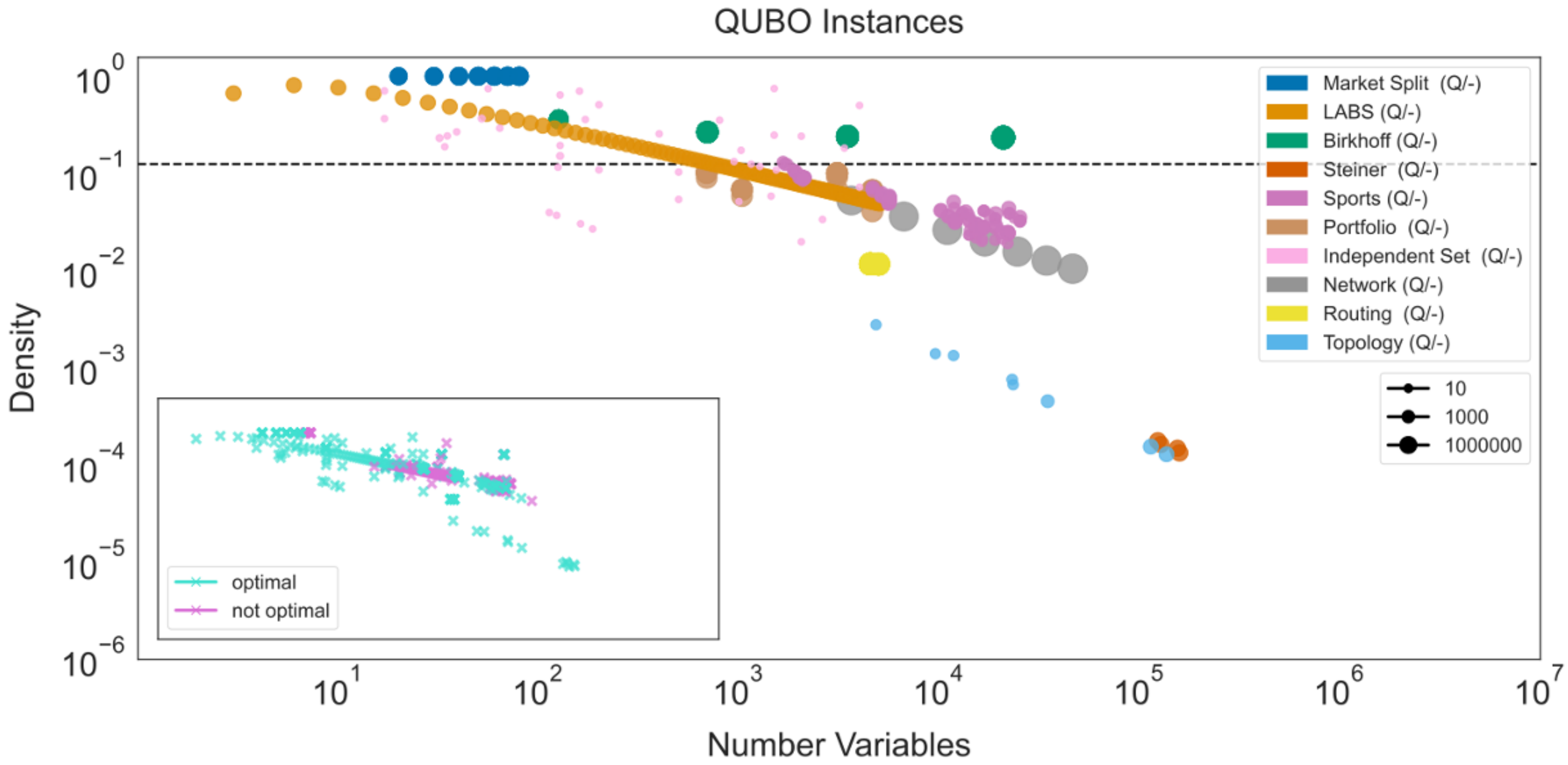
Quantum Optimization Benchmarking Library (QOBLIB)

https://git.zib.de/qopt/qoblib-quantum-optimization-benchmarking-library



Quantum Optimization Benchmarking Library (QOBLIB)

https://git.zib.de/qopt/qoblib-quantum-optimization-benchmarking-library



Mapping from MIP to QUBO changes the picture! Can lead to

- increasing number of variables
- increasing density
- larger range of coefficients

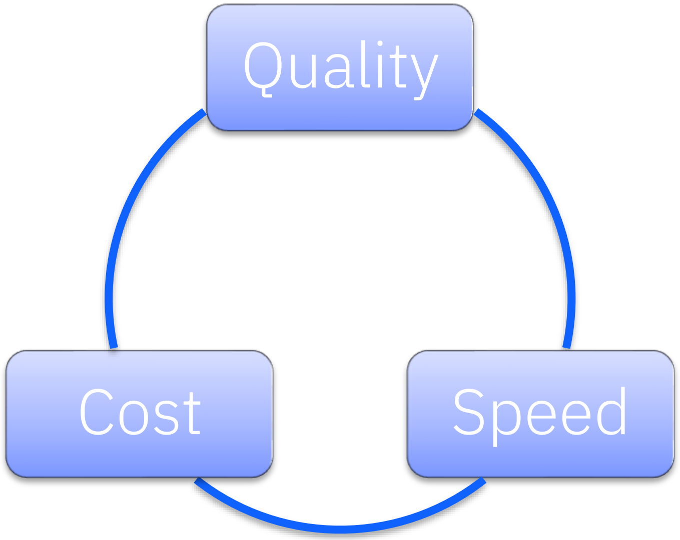
Performance Metrics

Metrics need to be well-defined and transparent to enable fair and systematic benchmarks!

Quality: Best found objective function value (incl. time series)

Speed: End-to-end wall clock time (excluding queueing times)
→ if an algorithm can be parallelized, that's good!

Cost: Hard to measure, not objective!
→ Transparent report of the involved computational resources (QPUs, CPUs, GPUs, other special-purpose accelerators...)



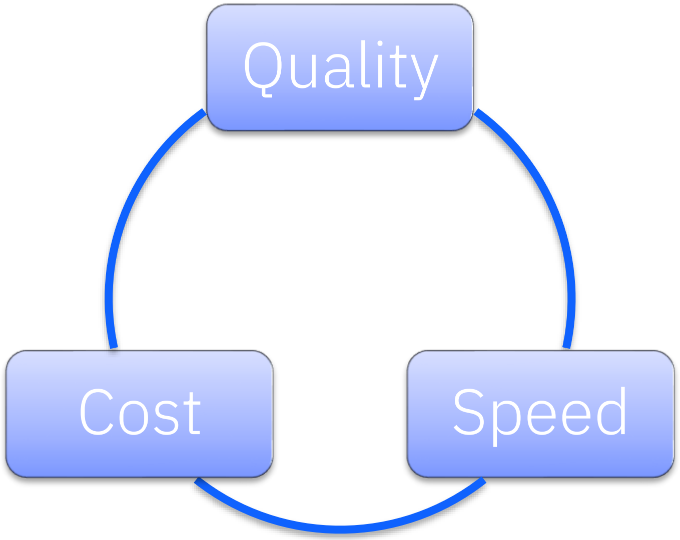
Performance Metrics

Metrics need to be well-defined and transparent to enable fair and systematic benchmarks!

Quality: Best found objective function value (incl. time series)

Speed: End-to-end wall clock time (excluding queueing times)
→ if an algorithm can be parallelized, that's good!

Cost: Hard to measure, not objective!
→ Transparent report of the involved computational resources (QPUs, CPUs, GPUs, other special-purpose accelerators…)



Trade-offs between metrics are possible!

Due to possible trade-offs between metrics, primary focus is on quality, i.e., to improve solution quality for unsolved instances

→ Every improvement in quality is a potential quantum advantage!

Submission Template

Problem Identifier	Identifier of the considered problem instance.
Submitter	Name(s) of the submitter(s) and affiliation(s).
Date	Date of submission.
Reference	Reference to a paper/repository with more details.
Best Objective Value	The best objective value found by the algorithm across all repetitions.
Optimality Bound	Lower bound (minimization) or upper bound (maximization) for the optimal objective value, if supported; otherwise, set to N/A.
Modeling Approach	Describe how the considered problem instance is modeled.
#Decision Variables	Total number of decision variables.
#Binary Variables	Number of binary decision variables.
#Integer Variables	Number of integer decision variables.
#Continuous Variables	Number of continuous decision variables.
Decision Variables Range	Range of the decision variables, i.e., min/max values.
#Non-Zero Coefficients	Number of non-zero coefficients in objective function and constraints.
Coefficients Type	Type of coefficients such as integer, binary, continuous.
Coefficients Range	Range of non-zero coefficients, i.e., min/max values.
Workflow	Description of the optimization workflow: pre-processing, pre-solvers, optimization algorithms, and post-processing, etc.
Algorithm Type	Indicate whether the algorithm is deterministic or stochastic.
#Runs	The number of times the experiment has been repeated.
#Feasible Runs	The number of times a run found a feasible solution.
#Successful Runs	Number of runs that found a feasible solution with objective value $\leq (1+\epsilon) * f_{\min}$ (minimization) or $\geq (1-\epsilon) * f_{\max}$ (maximization), where $f_{\min/\max}$ is the best solution found by the algorithm.
Success Threshold	The threshold ϵ to define a successful run.
Total Runtime	Total runtime to run the complete workflow.
CPU Runtime	CPU runtime to run the workflow.
GPU Runtime	GPU runtime to run the workflow.
QPU Runtime	QPU runtime to run the workflow.
Other HW Runtime	Runtime on other hardware to run the workflow.

Summary

- We introduced a framework to demonstrate quantum advantage in optimization and track progress towards this goal. Classical baseline results and first illustrative quantum results are provided as well.

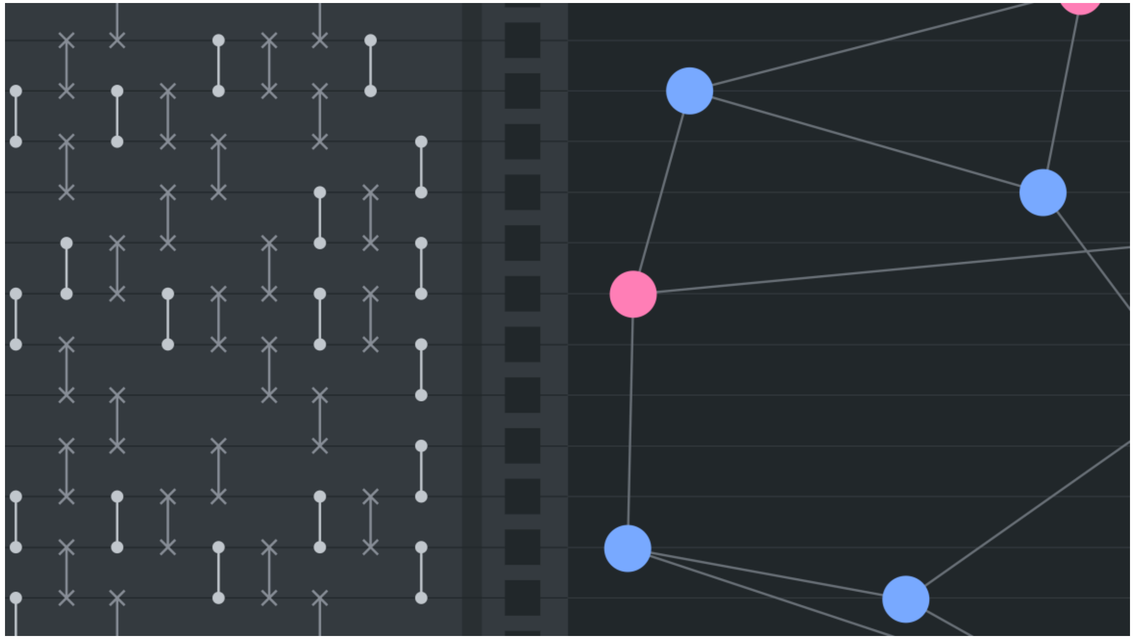
- Benchmarking of full workflow → Use the best of quantum and classical optimization.

- Will enable us to identify where quantum optimization (heuristic) algorithms may help (and where not).

- Model-independent → We need to look into new ways to model problems and solve them.

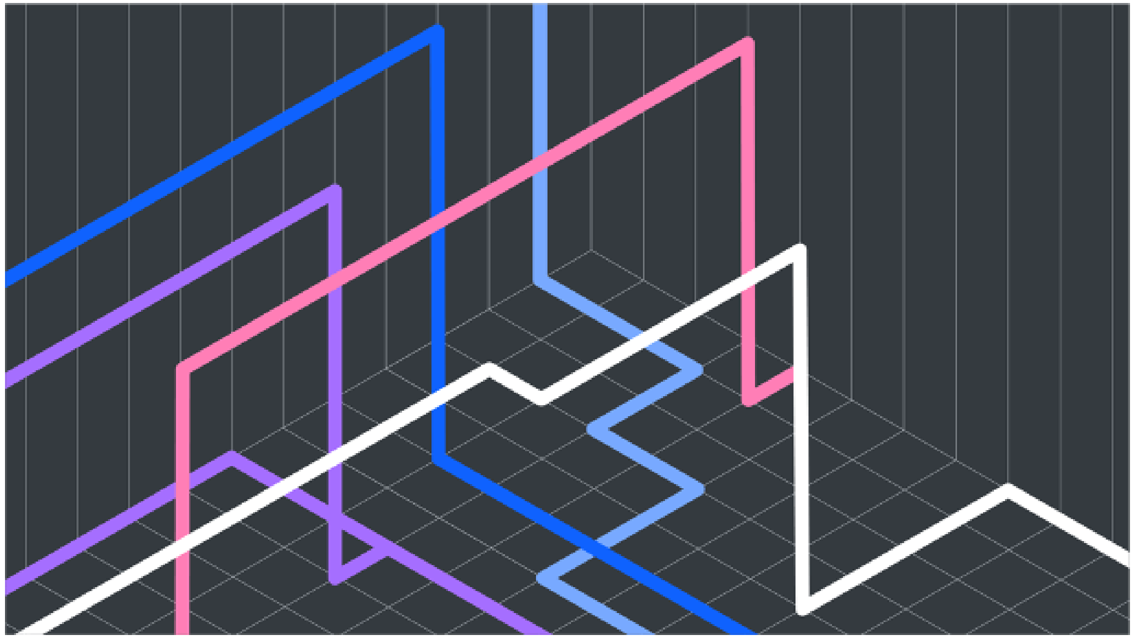
- Too many different possibilities to test, thus, this needs to be a community effort!

→ This is just the beginning, everybody is invited to contribute solutions to QOBLIB!
(classical & quantum submissions)



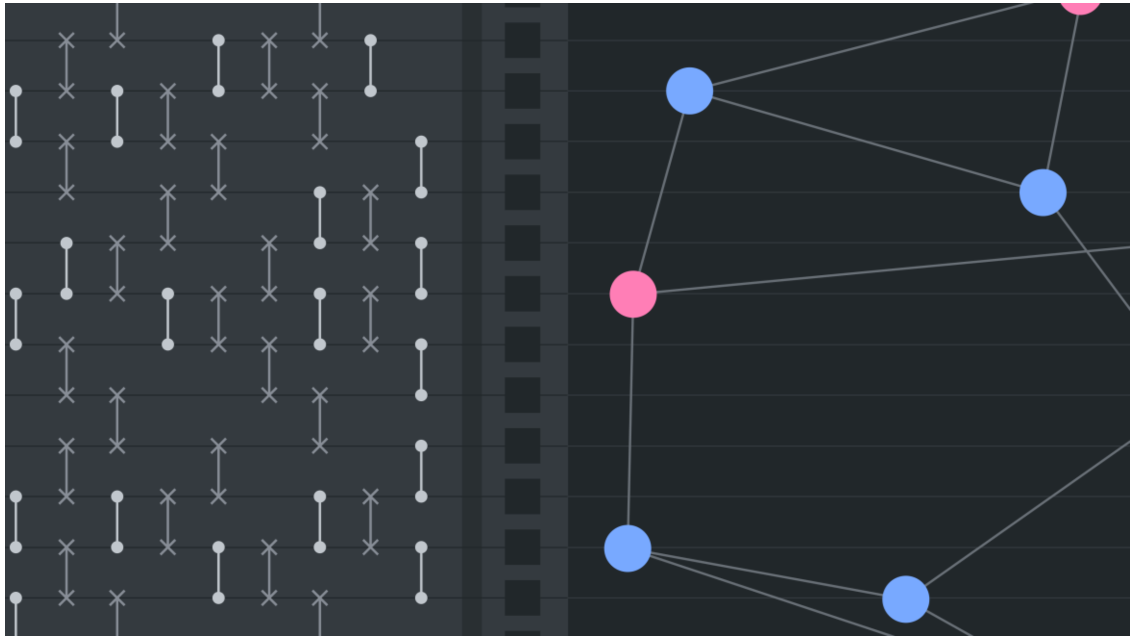
Exploring the potential for quantum advantage in mathematical optimization

https://www.ibm.com/quantum/blog/optimization-white-paper



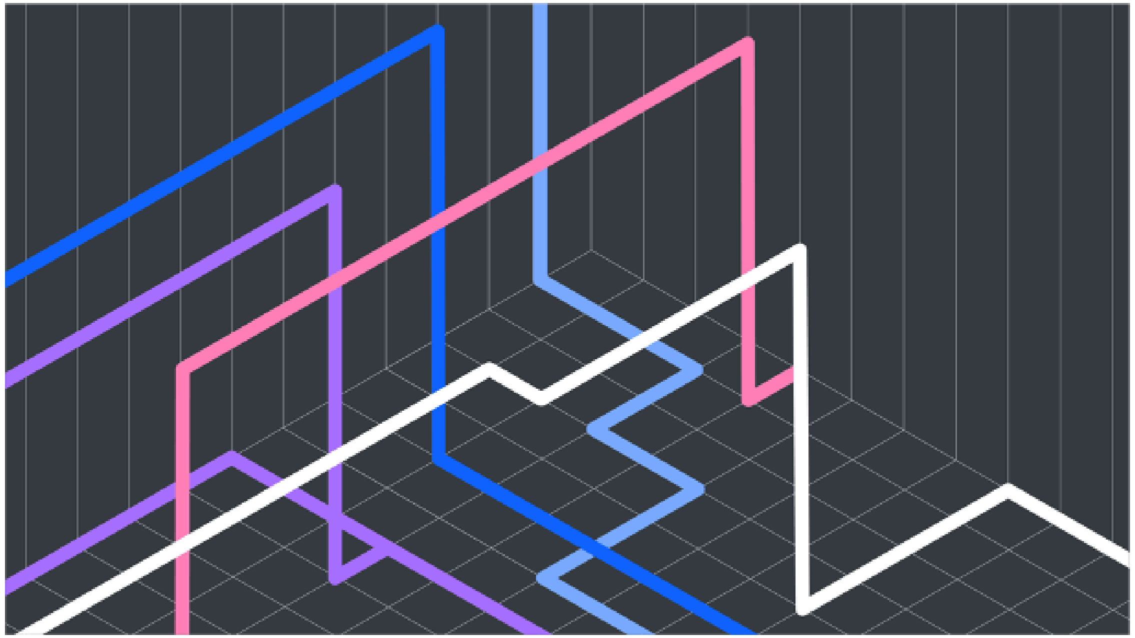
New Quantum Optimization Benchmarking Library invites researchers to put algorithms to the test

https://www.ibm.com/quantum/blog/quantum-optimization-benchmarking



Exploring the potential for quantum advantage in mathematical optimization

https://www.ibm.com/quantum/blog/optimization-white-paper



New Quantum Optimization Benchmarking Library invites researchers to put algorithms to the test

https://www.ibm.com/quantum/blog/quantum-optimization-benchmarking

Thank you for your attention!

IBM **Quantum**

wor@zurich.ibm.com