Public report on Quantinuum's Quantum Circuit Tensor Network simulation challenge

Pablo Andres-Martinez

July 26, 2025

1 Introduction

The guidelines for the challenge, along with the associated circuit suite are publicly available at [5]. Please refer to the `Instructions.pdf` file for details on the simulation task, the definition of the figures of merit used for evaluating the results and a description of the different circuit families included in the benchmarking suite.

1.1 Context

Quantum circuits simulators are a valuable tool for the rapid development of the quantum industry: access to quantum devices is on high demand, leading to long queue times, but much prototyping of ideas can be carried out on classical computers, reserving the use of quantum computers for the tasks and scales where simulation is unfeasible. The field of quantum simulation has a plethora of approaches each tailored to a different regime: for small number of qubits, statevector simulators [2] are the standard choice and, for purely Clifford circuits, stabiliser methods trivialise simulation [3]. This challenge focuses on *tensor network* methods which, as a rule of thumb, are the preferred choice when the circuit uses a large number of qubits (say, more than 40), contains more non-Clifford gates than qubits and produces low entanglement. The third condition is the most subtle one, since it is not obvious to infer how much entanglement a circuit creates simply from observing its size.

There are many different algorithms within the field of tensor network simulation and, once again, each has its particular strengths. Most of the participants of this challenge used some variant of Matrix Product State (MPS) simulation, which is a widely used general-purpose method. It is important to note, however, that MPS is rarely the choice for refuting quantum advantage experiments: much more nuanced and cutting-edge approaches are used in those cases. This challenge was not designed to explore the boundary where quantum advantage lies: the circuits that were simulated were challenging, but they were not at the scale for quantum advantage. And yet, these circuits are beyond the capabilities of statevector [2] and stabiliser [3] simulation. For the latter two, the state of the art is well known thanks to survey works [2], but there is a gap in the literature on benchmarking of tensor network simulators.

This challenge is an initiative to encourage the proliferation of benchmarking of tensor network methods. It should not be considered a thorough survey, since the number of participants is only a small fraction of all tensor network simulation software available, and the design of the challenge has multiple inherent limitations, as discussed in section 3.

1.2 Description

The approximate nature of tensor network simulators makes comparison of the performance between participants a subtle matter. The participant's simulators differ not only in implementation details, but also on the truncation strategies used to obtain approximations of the target state.

In this challenge, participants were asked to achieve certain target fidelities and report the wall-clock time to complete the simulation, including preprocessing and sampling of shots. The figure of merit for the accuracy of the simulation is the *mirror fidelity*: the overlap between the $|0\rangle$ state and the approximate state obtained after simulating $C^\dagger C|0\rangle$ for a quantum circuit C, where participants had to guarantee that no cancellation of gates between C and $C^\dagger$ occurred in preprocessing.

The circuits in the benchmarking suite were provided by research teams at Quantinuum and are publicly available at [5]. The suite contains five families of circuits originating from five different projects, and each of the families has its own characteristics. The number of qubits and gates for each of these circuits is included in the `metadata.csv` file at [5]. For more information about the nature of the circuits, refer to the `Instructions.pdf` file at [5].

2 Participants

Here I list the four participants of the public tier. For more information about the participants and their submissions, please visit [5].

- *pytket-mps*. The package pytket-cutensornet [1] is an open sourced library developed by Quantinuum that is used to simulate pytket circuits using Nvidia's CuTensorNet library. The MPS algorithm is a custom implementation developed using cuTensorNet's lower level primitives "contract" and "decompose", rather than Nvidia's own MPS implementation.

- *QuantumRings*. QuantumRings SDK [6] implements the Schmidt basis using a tensorized representation. The SDK works in a CPU only mode, GPU enabled mode, and a hybrid mode.

- *qmatchatea*. Open source library for quantum circuit simulation, part of the Quantum Tea initiative [7]. Developed by the University of Padova. All results correspond to the MPS algorithm.

- *MIMIQ*. Quantum circuit emulator based on the Matrix Product State (MPS) formalism, featuring advanced circuit preconditioning and Matrix Product Operator (MPO) compression to efficiently simulate large and moderately entangled quantum systems. Find more about them at [4].

2.1 Hardware

Below I provide a summarised table of the compute resources that each participant used. For a more detailed description, please refer to the "README" files on each of the participants submissions at [5].

Participant	**CPU**	**GPU**	**RAM**
pytket-mps	AMD EPYC 7763	Nvidia A100	80 GB (vRAM)
Quantum Rings	—	Nvidia H100	80 GB (vRAM)
qmatchatea (default)	Intel Xeon Platinum 8358	Nvidia A100	60 GB (vRAM)
qmatchatea (CPU)	AMD EPYC 9654	—	160 GB
MIMIQ	Intel Xeon w5-2545	—	128 GB

All participants but *MIMIQ* used GPU-based simulators. Furthermore, *qmatchatea* used a CPU backend for the simulations that would run out of memory in GPU.

3 Limitations of the challenge's design

The design choices of the challenge impose some limitations to the interpretation of the results. It is important to keep these limitations in mind when drawing conclusions from this report.

Target fidelity. Three regimes of accuracy of simulation were proposed, with the highest (and main goal) being a mirror fidelity ≥ 0.9, and the others being ≥ 0.6 and ≥ 0.2. Even though mirror fidelity of 0.9 is already a stringent requirement, it does not necessarily imply that the accuracy of the simulation is high enough so that the results have practical application. Interestingly, the results indicate that for some families, such as **mvsp** and **chemistry_uccsd**, higher fidelity thresholds were within reach (see appendix A). Targeting higher fidelity would potentially affect the results presented in section 5, as discussed in section 6.3.

Fine tuning. A natural consequence of the challenge's design is that participants would fine tune their simulator parameters to obtaining mirror fidelities that were just above the threshold: smaller fidelities leave more margin for truncation and, hence, can give an edge when comparing runtime. Some participants obtained fidelities closer to the threshold than others, which implies the results presented in 5 are not in a completely even ground. For more details on the mirror fidelities reported by each participant, see appendix A.

Validation. This challenge asks for participants to run the simulations on their own systems and report the results. It relies on trust that everyone makes every effort to report faithful results. There was some level of cross-validation enabled by the design of the challenge: participants had not seen these circuits in advance and the results of all participants were revealed at the same time. Participants were asked to calculate some expectation values on the final state of the circuits so that cross-validation could be performed, as discussed in section 4. This, as well as studying the results from each participant and how they scale with circuit hardness, would help identify outliers in the presence of result tampering, either intentional or accidental. After reviewing the results, and as of the date of writing, I have no reason to believe there has been any foul play. All outliers found in the data so far have reasonable explanations, and the more relevant ones are discussed in this text. In the coming weeks, some limited level of reproduction of results will be carried out to further verify the validity of the data, if any inconsistencies are found, a new version of this document, made publicly available at [5], will discuss the findings.

4 Cross-validation of expectation values

We requested participants to submit the expectation values of certain selected observables for each of the circuits in the benchmarking suite. We should observe strong agreement between participants when comparing high fidelity simulations. To visualise the results from cross-validation, I provide the script `expval_diff_heatmap.py` in [5], which produces one heat map per circuit: two of these are displayed in Fig. 1. Each cell of the heat map corresponds to a pairing of participants, and the metric for comparison is the largest of the absolute differences across the expectation values for the corresponding circuit. We generally find good agreement between participants' submissions whenever they report a mirror fidelity ≥ 0.9. However, in the case of **mvsp** circuits, we find that every participant has at least one instance where their expectation values noticeably differ from others'.

For most of these **mvsp** circuits, *MIMIQ* reports mirror fidelities of 0.90 while other participants provide fidelities from 0.91 up to essentially 1.0 (see appendix A). Due to *MIMIQ*'s position in the rankings presented in the following section, I requested simulations of higher fidelity from *MIMIQ* in order to validate the correctness of their algorithm.[1] The expectation values provided by *MIMIQ* for these new simulations coincide with the results from *QuantumRings* (the participant that provided highest fidelity submissions for **mvsp**) down to

[1]Namely, for the three circuits where they had largest difference from other participants: *mvsp_gaussian2d_fourier_d5_n25*, *mvsp_gaussian2d_fourier_d5_n40* and *mvsp_gaussian2d_fourier_d10_n10*.

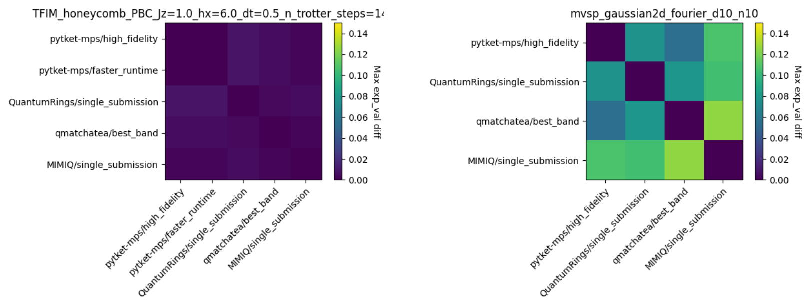

Figure 1: Each cell of the heat map corresponds to a pairing of participant's submissions. The value displayed in each cell is the largest of the absolute differences across the expectation values for the corresponding circuit. Outliers were investigated further, as discussed in the text.

the fourth decimal place.[2]. Since the goal of the challenge was to surpass the ≥ 0.9 threshold of mirror fidelity, the original submission from *MIMIQ* is valid and it is used in the evaluation of the results in the following section.

5 Results

I now present the rankings for each circuit family, according to the mirror fidelity bands specified in section 2.4 of the `Instructions.pdf` file. The figures in this section are generated from the participant's submissions by running the script `ranking.py` from [5]. Only figures for the target mirror fidelity thresholds ≥ 0.9 and ≥ 0.6 are presented. The fidelity band ≥ 0.2 was not plotted because there are very few data points that are not already included in the band ≥ 0.6, and these data points do not bring any surprises in the relative performance of the simulators. The main insight to be gained from the additional data points of the fidelity band ≥ 0.2 is that there are 7 circuits from the **condensed_matter** family that could be simulated with low fidelity, but were not simulated in the higher accuracy regimes.

In the figures, the circuits of each family are ordered according to the time it took the lead participant to run them. This aids the eye in identifying the best results. However, for other participants, the circuits that are easier to simulate may be different, causing the zig-zagging observed in the figures.[3]

I summarise my interpretation of the results for each circuit family separately. You may find a description of each of these circuit families in the `Instructions.pdf` file at [5]. You may also find information on the number of qubits and gates for each of these circuits in the `metadata.csv` file at the same location.

[2]Note that *MIMIQ* did not have access to the results of other participants at the time

[3]Bar plots would avoid this caveat but, given the large number of data points, I find this visualisation provides a better summary of the results at a glance.

condensed_matter. See Fig. 2. Roughly half of the circuits could not be simulated with high fidelity by any participant. For mirror fidelity ≥ 0.9, *MIMIQ* leads the ranking, except for two of the harder circuits, where *qmatchatea* leads: in one case by a small runtime difference, in the other case due to being the only participant that managed to simulate the circuit. *MIMIQ* is dramatically faster in the easier circuits of this family. For mirror fidelity ≥ 0.6 we see a similar behaviour, this time *pytket-mps* leading in one data point.

chemistry_uccsd. See Fig. 3. These circuits were comprised of many Pauli exponentials on multiple qubits. Participants whose simulators supported Pauli exponentials natively had an edge over the rest in this circuit suite. Equivalent circuits where the Pauli exponentials had been decomposed into 1-qubit and 2-qubit gates were also provided, however, these had an optimisation pass applied on them which, in hindsight, might have been in detriment to simulators that attempted to run these decomposed circuits, as the structure of the circuit is obscured by the optimisation. All participants submitted mirror fidelities above ≥ 0.9 exclusively, so both plots are identical. *pytket-mps* leads in all but four circuits: in two of them, *qmatchatea* runs faster, and in the other two, *pytket-mps* failed to complete the simulations. *QuantumRings* is faster than *qmatchatea* on two of the harder circuits.

mvsp. See Fig. 4. *MIMIQ* leads across the board, with a dramatic runtime difference. *qmatchatea* comes second for most of the circuits, while *QuantumRings* is the second for the harder ones. The plots for mirror fidelities ≥ 0.9 and ≥ 0.6 are qualitatively the same, with improved performance from *qmatchatea*. As discussed in section 4, for every pairing of participants, there is some circuit in this family where their expectation values differ in a non-negligible way. This seems to indicate that the level of fidelity achieved for these circuits is insufficient for accurate estimation of these quantities. Nevertheless, submissions from all participants consistently agree on which expectation values are larger/smaller, implying relative information of the magnitude of different expectation values can be successfully derived from these simulations. *QuantumRings* reported essentially 1.0 (perfect) fidelities for most of these circuits (see appendix A).

qmci. See Fig. 5. I did not expect it would be possible to simulate these circuits with high fidelity. *MIMIQ* is the only participant that managed to do so. The two plots look the same because *MIMIQ* provided only one submission. An important caveat is that *MIMIQ* reports the same memory required to represent the final state of each of these circuits. This may be due to the fact that all of the circuits in this family are designed to calculate the same result, but with different levels of accuracy (assuming an ideal execution of the circuit). Further investigation would be required to understand whether *MIMIQ*'s truncation strategy causes the approximated final state to be the same for all of these circuits, or if they simply have the same structure (hence the same memory requirement) but the coefficients are different.

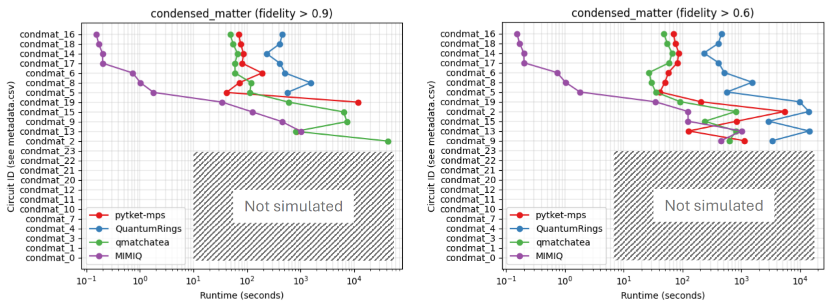

Figure 2: Results for the **condensed_matter** circuit family. Each label in the vertical axis corresponds to a circuit, as listed in the `metadata.csv` file at [5]. On the left, only submissions that achieved mirror fidelity of ≥ 0.9 are included. On the right, submissions with fidelity ≥ 0.6 are included.

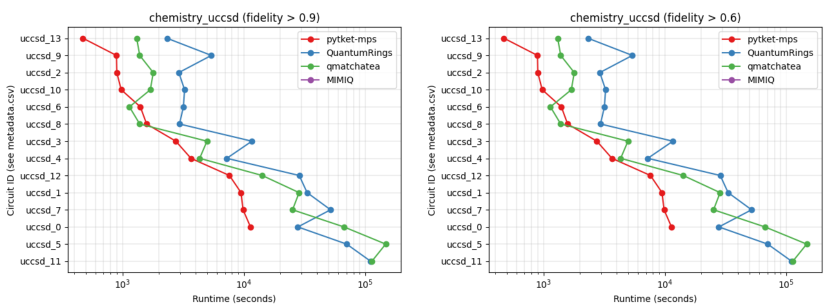

Figure 3: Results for the **chemistry_uccsd** circuit family. Each label in the vertical axis corresponds to a circuit, as listed in the `metadata.csv` file at [5]. On the left, only submissions that achieved mirror fidelity of ≥ 0.9 are included. On the right, submissions with fidelity ≥ 0.6 are included.

qec_non_ft. See Fig. 6. These circuits are Clifford and, consequently, can be trivially simulated by stabiliser methods [3]. These family of circuits was included in this challenge in hope of seeing some stabiliser-tensor network hybrid showcased by some of the participants. This was not the case, and the results show the poor scaling of tensor networks on Clifford circuits. *MIMIQ* leads in this family in all circuits that participants managed to simulate. Both plots are identical.

6 Conclusions

When deriving conclusions from this report, it is important to consider the caveats discussed in section 3. In particular, we must remember that the results may change for higher target fidelity, and that the evaluation is not in a

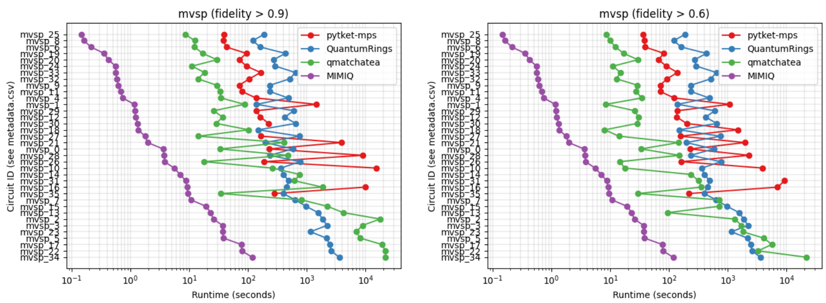

Figure 4: Results for the **mvsp** circuit family. Each label in the vertical axis corresponds to a circuit, as listed in the `metadata.csv` file at [5]. On the left, only submissions that achieved mirror fidelity of ≥ 0.9 are included. On the right, submissions with fidelity ≥ 0.6 are included.

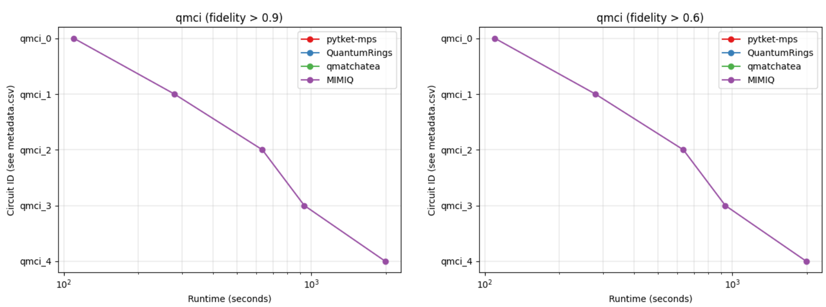

Figure 5: Results for the **qmci** circuit family. Each label in the vertical axis corresponds to a circuit, as listed in the `metadata.csv` file at [5]. On the left, only submissions that achieved mirror fidelity of ≥ 0.9 are included. On the right, submissions with fidelity ≥ 0.6 are included.

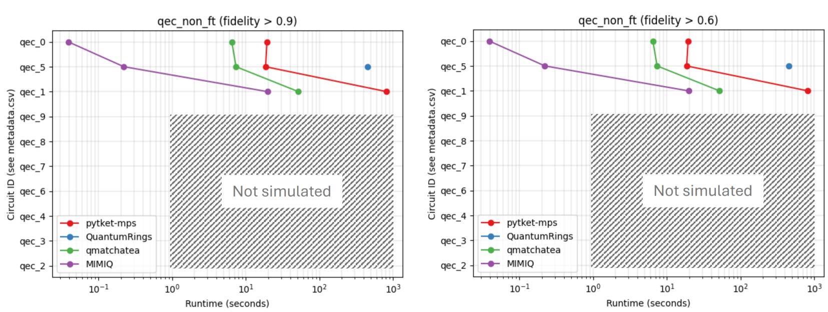

Figure 6: Results for the **qec_non_ft** circuit family. Each label in the vertical axis corresponds to a circuit, as listed in the `metadata.csv` file at [5]. On the left, only submissions that achieved mirror fidelity of ≥ 0.9 are included. On the right, submissions with fidelity ≥ 0.6 are included.

completely even ground (see appendix A).

For this circuit suite and fidelity regime, *MIMIQ* is the leader overall. For the harder circuits, however, the rest of the participants would occasionally have an edge over *MIMIQ*. I believe this reflects the fact that *MIMIQ* is the only participant running exclusively on CPUs and that, at certain point, the compute power of GPUs is necessary to achieve the highest performance. More thoughts on this appear in section 6.1.

qmatchatea is a consistent alternative that performs well on all families (except **qmci**). *pytket-mps* is comparable to *qmatchatea* for the most part, but *pytket-mps* scales worse with circuit hardness and is generally behind, except in the case of the **chemistry_uccsd** family. *QuantumCircuits* is competitive on the hardest circuits.

6.1 GPU vs CPU crossover

CPU backends being faster than GPU backends for easier circuits, with a crossover point after which the GPUs have an advantage, is something that has been reported in the literature of statevector simulation before [2], and I have observed it in tensor network simulation as well. My intuition, derived from profiling our own simulator *pytket-mps*, is that this phenomenon occurs when the memory required to represent the state is low. In such a situation, GPUs are not using 100% of their throughput and the overheads of CPU-GPU communication dominate the runtime.

Consequently, I expected CPU backends (namely, *MIMIQ*) to perform well on circuit families with low entanglement, such as **mvsp** and **chemistry_uccsd**. I did not expect, however, that it was possible to push simulation on CPUs to be competitive at simulating circuits as hard as the ones included in the **condensed_matter** family. Nevertheless, it is reasonable to consider that increasing the required level of fidelity may lead to simulations where GPU performs better than CPU. However, it is important to notice that the regime where GPUs are more performant than CPUs may be narrow, as we reach the point where the simulation requires more RAM than GPUs have.

6.2 Memory footprint

Participants were asked to provide the memory required to store the tensor network representing the state at the end of the circuit. This information is available at [5]. Since tensor network representations scale in memory with respect to the virtual bond dimension, this can serve as a rough indication of the level of entanglement on the final approximated state. Importantly, although in the case of MPS we can argue that small bond dimension implies low entanglement, the other direction of the implication does not necessarily hold: you may have large bond dimension due to Schmidt coefficients that are essentially zero and, therefore, do not contribute significantly to the entanglement (you would usually truncate these). Furthermore, qubit ordering in the tensor network representation is also essential, since long-range entanglement contributes to the virtual bond dimensions more dramatically than nearest-neighbour entanglement.

It is worth pointing out that the states at the end of all circuits in the **mvsp** family and the **chemistry_uccsd** family are represented by small MPS: often below 1 MB, and rarely above 100 MB. This equates to bond dimensions of only one or two digits, which is extremely low for MPS. This is especially the case in the submissions from *MIMIQ* and *pytket-mps*, but generally applies to all participants. There are two possible interpretations of this: either the ideal final states of these circuits have very low entanglement, or the truncations were so aggressive that relevant entanglement has been lost. We can confidently rule out the second option, since participants have also provided an estimate of the fidelity, independent from mirror fidelity, that is calculated in terms of the truncated Schmidt coefficients truncated. Generally, such an estimate to the fidelity provides a lower bound to the actual state fidelity. Considering that in both **mvsp** and **chemistry_uccsd** families these estimates are comparable to the mirror fidelity, which is above 0.9, this provides further evidence that the truncations were not overly aggressive.

The natural conclusion is that the ideal final states of the circuits in both **chemistry_uccsd** and **mvsp** families have low entanglement. This does not prevent intermediate states in the circuit from having larger entanglement. Nevertheless, my general impression is that both **chemistry_uccsd** and **mvsp** families were relatively easy to simulate due to low entanglement; although in the case of **chemistry_uccsd** this was only exploitable by participants that could take advantage of the structure of the circuit, applying the Pauli exponentials natively or appropriately batching the decomposed circuits.

6.3 Simulation accuracy

An immediate follow up question after evaluating these results is: "can we simulate these circuits with higher fidelity?". For the most part, the answer is "yes", but with rapidly increasing overheads. I would like to point out that a mirror fidelity of ≥ 0.9 is quite high precision, considering many of these circuits have tens of thousands of two-qubit gates. Still, this fidelity is not high enough for some practical applications, as evidenced by the large variance in expectation values observed in the **mvsp** family. Further iterations of this challenge would benefit from requesting higher fidelities from participants.

6.4 Further contributions

I welcome participants (either the ones listed here or new ones) to create PRs on the public repository [5] to include new submissions in order to maintain the results of the challenge up to date with progress in the field. Please contact `pablo.andresmartinez@quantinuum.com` if you wish to contribute.

In the case of new contributions, a separate branch will be created for each version, indicating the date of the most recent submission from each of the participants.

6.5 To all participants: thank you!

At Quantinuum, we are aware that participating in this challenge meant a considerable investment in compute resources and employee time. We sincerely appreciate the effort you have put into your submissions, which have all been of excellent quality.

We believe the field of tensor network simulation would benefit from more benchmarking initiatives, and we are grateful for your enthusiasm to contribute towards this.

References

[1] *Documentation for pytket-cutensornet.* URL: `https://docs.quantinuum.com/tket/extensions/pytket-cutensornet/`.

[2] Amit Jamadagni Gangapuram, Andreas Läuchli, and Cornelius Hempel. "Benchmarking quantum computer simulation software packages: State vector simulators". In: *SciPost Physics Core* 7.4 (Nov. 2024). ISSN: 2666-9366. DOI: `10.21468/scipostphyscore.7.4.075`. URL: `http://dx.doi.org/10.21468/SciPostPhysCore.7.4.075`.

[3] Craig Gidney. "Stim: a fast stabilizer circuit simulator". In: *Quantum* 5 (July 2021), p. 497. ISSN: 2521-327X. DOI: `10.22331/q-2021-07-06-497`. URL: `http://dx.doi.org/10.22331/q-2021-07-06-497`.

[4] *MIMIQ main page.* URL: `https://qperfect.io/index.php/mimiq/`.

[5] *Public repository with the participant's submissions.* URL: `https://github.com/CQCL/public_tn_sim_challenge_results`.

[6] *Quantum Rings page.* URL: `https://www.quantumrings.com`.

[7] *Quantum Tea page.* URL: `https://www.quantumtea.it/`.

A Margins above the target fidelity threshold

In this appendix, I present figures displaying the mirror fidelities reported by each participant. The purpose of this appendix is to contextualise the results displayed in section 5 in terms of the fidelity achieved. This is motivated by the fact that different participants surpassed the target mirror fidelity 0.9 with different margins and, hence, the results plotted in section 5 are not in completely even ground. The figures are generated using the script `boxplot_margin_fidelity.py` available at [5].

Important note. The instructions for the challenge (described in the file `Instructions.pdf` at [5]) explicitly indicated that evaluation of the results would be in terms of runtime of the simulations that surpassed the target mirror fidelity thresholds. Consequently, participants were not competing to try to achieve the highest fidelity they could, but rather a trade-off of runtime versus fidelity. This is essential to keep in mind when reviewing Figure 7, as indicated in its caption.

For all families but **mvsp**, we observe that participants achieved similar distributions of fidelities. In the case of the **mvsp** family, *MIMIQ* did extensive fine tuning to obtain a tight match with the target mirror fidelity threshold, whereas *QuantumRings* maximised fidelity.

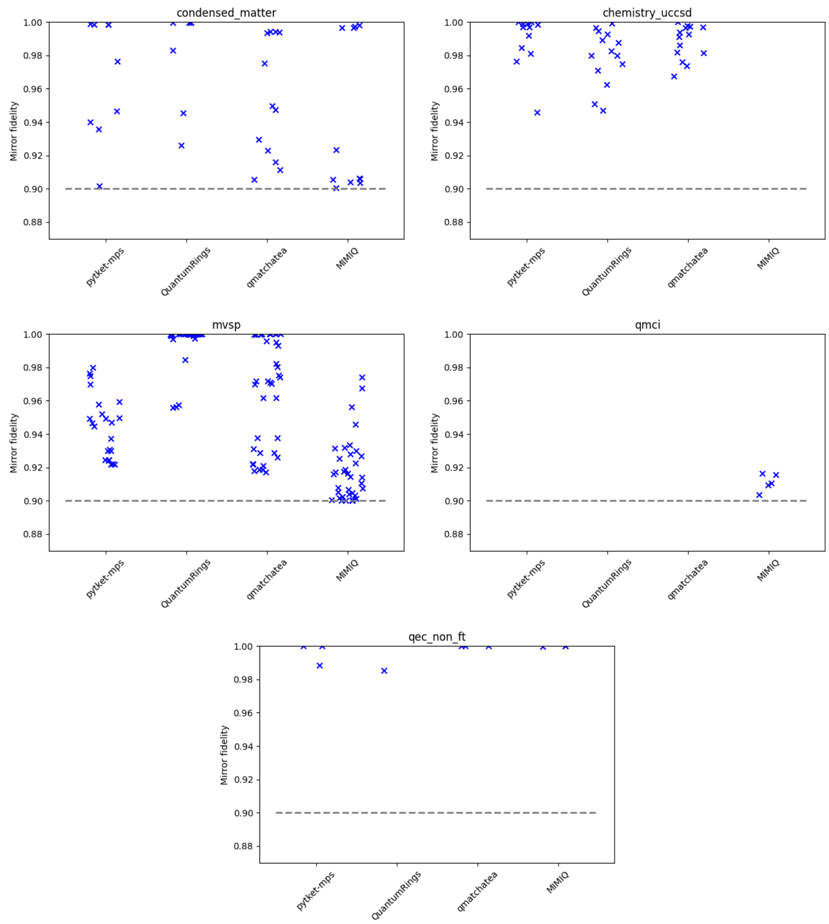

Figure 7: Scatter plot displaying, for each circuit family, the distribution of mirror fidelities obtained by each participant. Each data point corresponds to a circuit whose simulation was achieved with mirror fidelity above the 0.9 threshold (drawn with a dashed line). **These do not represent the highest fidelities that these simulators can achieve**, but rather the fidelities of the fastest simulations that still surpassed the 0.9 threshold.