Meta-Designing Quantum Experiments with Language Models

Sören Arlt
Max Planck Institute for the Science of Light
soeren.arlt@mpl.mpg.de

Haonan Duan
University of Toronto, Vector Institute
haonand@cs.toronto.edu

Felix Li
UC Berkeley
fzli@berkeley.edu

Sang Michael Xie
Stanford University
xie@cs.stanford.edu

Yuhuai Wu
xAI
yuhuai@x.ai

Mario Krenn
Max Planck Institute for the Science of Light
mario.krenn@mpl.mpg.de

Abstract

Artificial Intelligence (AI) has the potential to significantly advance scientific discovery by finding solutions beyond human capabilities. However, these superhuman solutions are often unintuitive and require considerable effort to uncover underlying principles, if possible at all. Here, we show how a code-generating language model trained on synthetic data can not only find solutions to specific problems but can create meta-solutions, which solve an entire class of problems in one shot and simultaneously offer insight into the underlying design principles. Specifically, for the design of new quantum physics experiments, our sequence-to-sequence transformer model generates interpretable Python code that describes experimental blueprints for a whole class of quantum systems. We discover general and previously unknown design rules for infinitely large classes of quantum states. The ability to automatically generate generalized patterns in readable computer code is a crucial step toward machines that help discover new scientific understanding – one of the central aims of physics.

1 Introduction

Quantum physics is a notoriously unintuitive field of study. Despite this, it has developed to a point where some of its most difficult-to-conceptualize effects - such as entanglement - could become the basis of a new generation of technological development. Due to difficulties in designing experimental setups by hand, computational design techniques are used to deliver solutions, which surpass designs by human experts Krenn et al. [2020]. E.g. for a given target quantum state, a machine can design the experimental setup which creates the state, but interpretation and generalization of the results is left to the researcher and is often an exceptionally hard challenge, if possible at all.

Here we introduce the process of *meta-design*. Instead of designing one solution for a single target (i.e. one experimental setup for the creation of one quantum state), we train and sample a sequence-to-sequence transformer to design a meta-solutions in the form of programming code. A meta-solution solves an infinitely large class of targets (a class of quantum state) by generating different experimental setups for different integer values of `N`. Our approach is successful in designing

Machine Learning and the Physical Sciences Workshop, NeurIPS 2024.

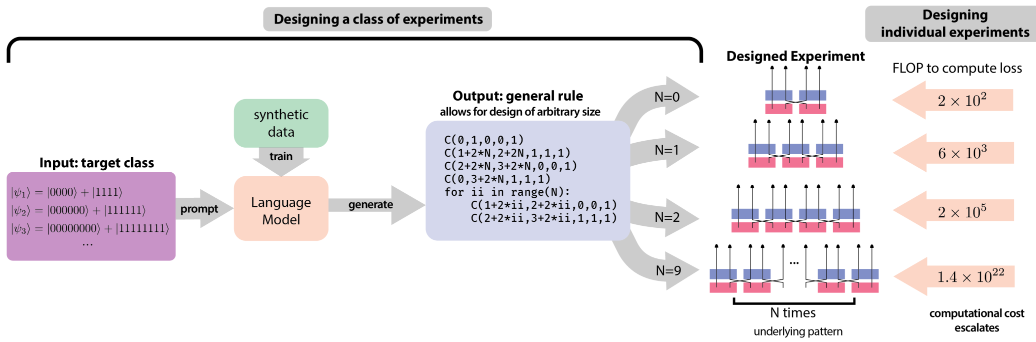

Figure 1: **Meta-designing a class of experiments via code generation avoids exploding computational costs for the design of larger experiments.** Left side: Our process takes the first three states from a class of target quantum states and - when successful - produces a Python code which generates the correct experimental setup for arbitrary sizes. The setup for n particles consists of n paths leading to n detectors. The function call `C(p1,p2,m1,m2,w)` denotes placing a photon pair source at a crossing of paths `p1` and `p2`, creating photons with modes `m1` and `m2` (shown as color) and a weight `w` (introducing possible phases). Right side: Designing an experimental setup which produces a target quantum state is very fast for small particle numbers. But the computational cost explodes as the target state grows.

meta-solutions for many interesting classes of quantum states, several of which were not known previously. The readability of the code representation helps to uncover the underlying patterns in the class of solutions. Therefore, our technique is a step towards AI methods that can help to gain new understanding in physics De Regt [2017], Krenn et al. [2022], Barman et al. [2024].

2 Related Work

AI for discovery in quantum physics. AI techniques have been previously applied to the search for experimental setups in quantum physics Krenn et al. [2016], Knott [2016], Nichols et al. [2019], Wallnöfer et al. [2020], Prabhu et al. [2020], Krenn et al. [2021], Ruiz-Gonzalez et al. [2023], Goel et al. [2024], Landgraf et al. [2024], nanophotonic structures Molesky et al. [2018], Sapra et al. [2020], Ma et al. [2021], Gedeon et al. [2023], and quantum circuits Ostaszewski et al. [2021], Nägele and Marquardt [2023], Kottmann [2023], Zen et al. [2024], MacLellan et al. [2024]. All of these works have in common that the algorithm produces only a single solution and not a meta-solution that represents large classes of solutions.

Transformers for math and physics Transformer architectures have demonstrated remarkable success in solving a wide range of mathematics and physics reasoning tasks. Lample and Charton [2019] and Kamienny et al. [2022] show that a transformer-based sequence-to-sequence model can tackle symbolic math problems such as symbolic integration, differential equations and symbolic regression. AlphaGeometry [Trinh et al., 2024] has achieved remarkable performance in solving geometry problems at an olympiad level. Alfarano et al. [2023] finds that by training transformers on synthetic data, they can accurately predict the Lyapunov functions of polynomial and non-polynomial dynamical systems. In the field of theoretical high-energy physics, Cai et al. [2024] applies transformers to compute scattering amplitudes.

3 Background Quantum Physics

We choose the design of quantum optics experiments as proof of concept and point to the great potential in applying the approach in other fields. Quantum optics is concerned with photons, the fundamental particles of light. A photon can have different polarization modes, e.g. horizontal (mode 0) or vertical (mode 1). A basic property of quantum particles is that they can be in a superposition of multiple modes, i.e. they can be considered to be two things at the same time. The state $|\psi\rangle$ of one photon in equal superposition can be expressed in Dirac notation as $|\psi\rangle = |0\rangle + |1\rangle$. We omit the normalization factor for all quantum states shown in this work for readability. It can be assumed that all states are normalized. Another important concept in quantum physics is *entanglement*, where multiple photons are in a state where they cannot be described independently, such as the superposition of three particles being in the superposition of either all particles being in mode 0 or all particles being in mode 1, $|\psi\rangle = |000\rangle + |111\rangle$. This state is called the GHZ state Greenberger et al. [1990], Pan et al. [2000]. In quantum optics, highly entangled states can be created by combining probabilistic photon pair sources. For sufficiently large particle numbers, the task of designing these setups becomes too difficult even for current computational methods Ruiz-Gonzalez et al. [2023] as they become too computationally expensive (see right side of Fig. 1).

4 Methods

We introduce meta-design, the idea of generating a meta-solution that can solve a whole class of solutions (in our case, for design problems of quantum states). Our meta-solutions are Python codes that can generate blueprints of experimental setups. We train a sequence-to-sequence transformer on synthetic data to translate from a class of quantum states to Python code and sample the model to discover programs for a collection of target classes.

Meta-design for Quantum Experiments A famous class of quantum states are the GHZ states, which are shown in the left box of Fig. 1. They are superposition of particles being either in mode 0 or mode 1 with an increasing number of photons (4, 6, 8, ...). We now aim to find a program `construct_setup(N)` which generates the correct experimental setup for creating GHZ states for a given particle number N. This is possible because the GHZ states follow a specific pattern. The solution to the problem is shown in Fig. 1. After constructing the setup, we can compute the expected quantum state that emerges at the detectors.

Data (generate random `B`, compute `A`) On an abstract level, we can describe the subject of our work as translating from sequence `A` (a list of three quantum states) to sequence `B` (python program). Direction `B→A` (computing the resulting quantum states from experimental setups) follows clear instructions and can be considered *easy*. Direction `A→B` is highly non-trivial. Using a simple set of rules, we can generate a random python code (sequence `A`), which contains instructions for how to set up an experiment. Each code contains the variable index N. This means that the code will result in a different experimental setup for each value of N. Simulating the experiment for $N = 0, 1, 2$, we produce three states , making up sequence `B`. The maximum length for both sequences during data generation is 640 tokens. Both sequences are tokenized by a hand-picked vocabulary dictionary. We spend about 50,000 CPU hours on generating 56 million samples.

Training (learn `A→B`) We train the model with a standard encoder-decoder transformer architecture Vaswani et al. [2017], with Pre-Layer Normalization Xiong et al. [2020] and learned positional encoding Gehring et al. [2017]. We choose the dimensions $n_{emb} = 512$, $n_{\text{layer}} = 18$, $n_{\text{heads}} = 8$. We use a learned positional encoding, as we are not attempting to apply our model to unseen lengths. The model has approximately 133 million parameters and is trained for 750k steps with a batch size of 256 (approximately 2.5 epochs on a dataset of 56 million samples). The learning rate of the Adam optimizer Kingma and Ba [2014] was 10^{-4} for the first epoch and was then lowered to 10^{-5}. The training was performed on four A100-40GB GPUs.

5 Evaluating on unknown targets

Our goal is now to apply the trained model to quantum state classes of particular interest (because of particular mathematical or physical properties) for which the code (sequence `B`) is unknown and predict the correct meta-solution. We have compiled a collection of twenty target classes based on a collection of quantum states found in Ruiz-Gonzalez et al. [2023] – all of these states have exceptional properties that have been studied previously. The first three states of each target class are explicitly shown in the appendix. They are expressed as strings in the same way in which they are given to the model as input. For each target, we sample the model with top-p sampling ($p = 0.5$ and temperature 0.2, Chen et al. [2021], Li et al. [2023]) for four hours on one RTX 6000 GPU, which produces 800-2500 samples (depending on the target class). We evaluate the resulting codes by executing them to produce experimental setups for $N = 0, 1, 2, 3, 4$ (training data was generated only for $N = 0, 1, 2$). We compute the states which are produced by these setups and compute their

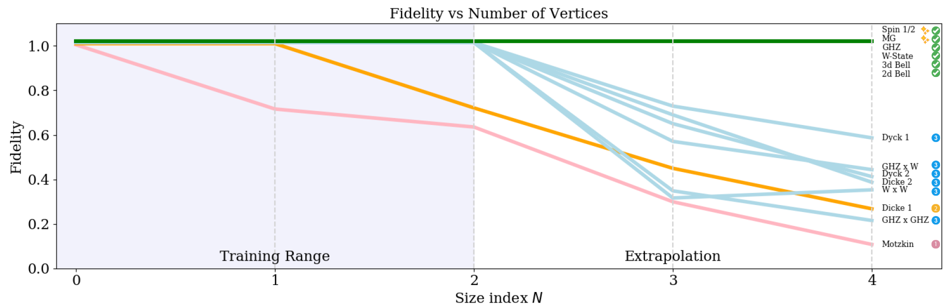

Figure 2: **Our approach discovers two previously unknown and four previously known generalizations.** We show the resulting fidelities of the best produced code for 14 of the 20 target classes. The green line represents the six target classes which our approach produces codes which correctly extrapolate beyond the first three elements. The blue lines show classes for which the best generated codes have fidelity one for the first three elements of the class, but do not extrapolate beyond. The orange and red line are representatives of the 8 cases, for which the model was not able to predict correct solutions up to $N = 3$. The full table of target classes with their maximum correct N is shown in the appendix.

fidelity with respect to the corresponding target state. The fidelity ranges from 0 (orthogonal to target) to 1 (perfect match). In Fig. 2 we show the fidelities of the best sample for fourteen of the twenty target classes.

Successful meta-design of codes (6 out of 20 cases) For these classes, the output extrapolates beyond what the model was trained to do, i.e. match the states for $N = 0, 1, 2$. Most importantly, two out of six classes which our method successfully solves, were previously unknown and thus constitute a genuine discovery. The Spin-$\frac{1}{2}$ states Ruiz-Gonzalez et al. [2023], Bernien et al. [2017] and the ground states of the Majumdar-Gosh Model Ruiz-Gonzalez et al. [2023], Chhajlany et al. [2007].

Codes with unexpected generalizations (6 out of 20 cases) For these cases the model produces codes, which generate the correct states for the first three elements of the class, but produces experiments that produce states other than the expected ones. These cases are interesting to examine because the model successfully performs the task it was trained for, as the first three states match the input sequence. There is potential in examining these cases in more detail to see if the pattern they follow is interesting from a physics side, as they might represent new unexplored classes of quantum states.

Codes which fail to match the first three states (8 out of 20 cases) These cases could be either too complex for the model to give the correct prediction, or generalisations cannot exist at all for physical reasons, given the amount of quantum resources we provide.

6 Proof of concept for other fields

To show that meta-design can be applied beyond the field of designing quantum optics experiments, we also applied it to a task in quantum circuit design. We created 5 million samples of randomly generated python codes for constructing quantum circuits For each code we computed the resulting quantum states for $N = 1, 2, 3$ with the `qiskit` framework and concatenated them to create the source sequence. We trained a small model ($n_{\text{embed}} = 512, n_{\text{head}} = 8, n_{\text{layer}} = 6$) for two hours on 4 A100-40GB GPUs. The model could correctly solve the task of creating GHZ states of increasing size. No sampling was necessary as the correct code is generated through greedy decoding. This indicates that the model would also be capable of solving more complicated tasks with sampling.

7 Discussion

We demonstrate how a language model can produce a meta-solution for a physical design task. The meta-solution is described in the form of computer code, which itself produces solutions to large generalizations of the design question. In our examples, we discover previously unknown generalizations of experimental setups for interesting quantum states. Our method is not constrained to quantum physics but can be directly implemented in other domains, such as the discovery of new microscopes Rodríguez et al. [2023], new gravitational wave detectors Krenn et al. [2023], new experimental hardware for high-energy physics Baydin et al. [2021], or the design of new functional molecules Pollice et al. [2021]. The representation of solutions as python code significantly enhances the understandability and generalizability of AI-driven discoveries.

Acknowledgements

The authors thank Ben Newman for useful discussions.

References

Alberto Alfarano, François Charton, Amaury Hayat, and CERMICS-Ecole des Ponts Paristech. Discovering lyapunov functions with transformers. In *The 3rd Workshop on Mathematical Reasoning and AI at NeurIPS'23*, 2023.

Kristian Gonzalez Barman, Sascha Caron, Tom Claassen, and Henk De Regt. Towards a benchmark for scientific understanding in humans and machines. *Minds and Machines*, 34(1):1–16, 2024.

Atilim Güneş Baydin, Kyle Cranmer, Pablo de Castro Manzano, Christophe Delaere, Denis Derkach, Julien Donini, Tommaso Dorigo, Andrea Giammanco, Jan Kieseler, Lukas Layer, Gilles Louppe, Fedor Ratnikov, Giles Strong, Mia Tosi, Andrey Ustyuzhanin, Pietro Vischia, and Hevjin Yarar. Toward machine learning optimization of experimental design. *Nuclear Physics News*, 31(1):25–28, 2021.

Hannes Bernien, Sylvain Schwartz, Alexander Keesling, Harry Levine, Ahmed Omran, Hannes Pichler, Soonwon Choi, Alexander S Zibrov, Manuel Endres, Markus Greiner, et al. Probing many-body dynamics on a 51-atom quantum simulator. *Nature*, 551(7682):579–584, 2017.

Tianji Cai, Garrett W Merz, François Charton, Niklas Nolte, Matthias Wilhelm, Kyle Cranmer, and Lance J Dixon. Transforming the bootstrap: Using transformers to compute scattering amplitudes in planar n= 4 super yang-mills theory. *arXiv:2405.06107*, 2024.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *arXiv:2107.03374*, 2021.

Ravindra W Chhajlany, Piotr Tomczak, Antoni Wójcik, and Johannes Richter. Entanglement in the majumdar-ghosh model. *Physical Review A*, 75(3):032340, 2007.

Henk W De Regt. *Understanding scientific understanding*. Oxford University Press, 2017.

Johannes Gedeon, Emadeldeen Hassan, and Antonio Calà Lesina. Free-form inverse design of arbitrary dispersive materials in nanophotonics. *arXiv:2305.00234*, 2023.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. Convolutional sequence to sequence learning. In *International conference on machine learning*, pages 1243–1252. PMLR, 2017.

Suraj Goel, Saroch Leedumrongwatthanakun, Natalia Herrera Valencia, Will McCutcheon, Armin Tavakoli, Claudio Conti, Pepijn WH Pinkse, and Mehul Malik. Inverse design of high-dimensional quantum optical circuits in a complex medium. *Nature Physics*, pages 1–8, 2024.

Daniel M. Greenberger, Michael A. Horne, Abner Shimony, and Anton Zeilinger. Bell's theorem without inequalities. *American Journal of Physics*, 58(12):1131–1143, December 1990.

Pierre-alexandre Kamienny, Stéphane d'Ascoli, Guillaume Lample, and Francois Charton. End-to-end symbolic regression with transformers. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 10269–10281. Curran Associates, Inc., 2022.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

PA Knott. A search algorithm for quantum state engineering and metrology. *New Journal of Physics*, 18(7): 073033, 2016.

Jakob S. Kottmann. Molecular quantum circuit design: A graph-based approach. *Quantum*, 7:1073, August 2023.

Mario Krenn, Mehul Malik, Robert Fickler, Radek Lapkiewicz, and Anton Zeilinger. Automated search for new quantum experiments. *Physical review letters*, 116(9):090405, 2016.

Mario Krenn, Manuel Erhard, and Anton Zeilinger. Computer-inspired quantum experiments. *Nature Reviews Physics*, 2(11):649–661, September 2020.

Mario Krenn, Jakob S Kottmann, Nora Tischler, and Alán Aspuru-Guzik. Conceptual understanding through efficient automated design of quantum optical experiments. *Physical Review X*, 11(3):031044, 2021.

Mario Krenn, Robert Pollice, Si Yue Guo, Matteo Aldeghi, Alba Cervera-Lierta, Pascal Friederich, Gabriel dos Passos Gomes, Florian Häse, Adrian Jinich, AkshatKumar Nigam, Zhenpeng Yao, and Alan Aspuru-Guzik. On scientific understanding with artificial intelligence. *Nature Reviews Physics*, 4(12):761–769, 2022.

Mario Krenn, Yehonathan Drori, and Rana X Adhikari. Digital discovery of interferometric gravitational wave detectors. *arXiv:2312.04258*, 2023.

Guillaume Lample and François Charton. Deep learning for symbolic mathematics. *arXiv:1912.01412*, 2019.

Jonas Landgraf, Vittorio Peano, and Florian Marquardt. Automated discovery of coupled mode setups. *arXiv:2404.14887*, 2024.

Raymond Li, Loubna Ben Allal, Yangtian Zi, Niklas Muennighoff, Denis Kocetkov, Chenghao Mou, Marc Marone, Christopher Akiki, Jia Li, Jenny Chim, Qian Liu, Evgenii Zheltonozhskii, Terry Yue Zhuo, Thomas Wang, Olivier Dehaene, Mishig Davaadorj, Joel Lamy-Poirier, João Monteiro, Oleh Shliazhko, Nicolas Gontier, Nicholas Meade, Armel Zebaze, Ming-Ho Yee, Logesh Kumar Umapathi, Jian Zhu, Benjamin Lipkin, Muhtasham Oblokulov, Zhiruo Wang, Rudra Murthy, Jason Stillerman, Siva Sankalp Patel, Dmitry Abulkhanov, Marco Zocca, Manan Dey, Zhihan Zhang, Nour Fahmy, Urvashi Bhattacharyya, Wenhao Yu, Swayam Singh, Sasha Luccioni, Paulo Villegas, Maxim Kunakov, Fedor Zhdanov, Manuel Romero, Tony Lee, Nadav Timor, Jennifer Ding, Claire Schlesinger, Hailey Schoelkopf, Jan Ebert, Tri Dao, Mayank Mishra, Alex Gu, Jennifer Robinson, Carolyn Jane Anderson, Brendan Dolan-Gavitt, Danish Contractor, Siva Reddy, Daniel Fried, Dzmitry Bahdanau, Yacine Jernite, Carlos Muñoz Ferrandis, Sean Hughes, Thomas Wolf, Arjun Guha, Leandro von Werra, and Harm de Vries. Starcoder: may the source be with you! *arXiv:2305.06161*, 2023.

Wei Ma, Zhaocheng Liu, Zhaxylyk A Kudyshev, Alexandra Boltasseva, Wenshan Cai, and Yongmin Liu. Deep learning for the design of photonic structures. *Nature Photonics*, 15(2):77–90, 2021.

Benjamin MacLellan, Piotr Roztocki, Stefanie Czischek, and Roger G Melko. End-to-end variational quantum sensing. *arXiv:2403.02394*, 2024.

Sean Molesky, Zin Lin, Alexander Y Piggott, Weiliang Jin, Jelena Vucković, and Alejandro W Rodriguez. Inverse design in nanophotonics. *Nature Photonics*, 12(11):659–670, 2018.

Maximilian Nägele and Florian Marquardt. Optimizing zx-diagrams with deep reinforcement learning. *arXiv:2311.18588*, 2023.

Rosanna Nichols, Lana Mineh, Jesús Rubio, Jonathan CF Matthews, and Paul A Knott. Designing quantum experiments with a genetic algorithm. *Quantum Science and Technology*, 4(4):045012, 2019.

Mateusz Ostaszewski, Lea M Trenkwalder, Wojciech Masarczyk, Eleanor Scerri, and Vedran Dunjko. Reinforcement learning for optimization of variational quantum circuit architectures. *Advances in Neural Information Processing Systems*, 34:18182–18194, 2021.

Jian-Wei Pan, Dik Bouwmeester, Matthew Daniell, Harald Weinfurter, and Anton Zeilinger. Experimental test of quantum nonlocality in three-photon greenberger–horne–zeilinger entanglement. *Nature*, 403(6769):515–519, February 2000.

Robert Pollice, Gabriel dos Passos Gomes, Matteo Aldeghi, Riley J Hickman, Mario Krenn, Cyrille Lavigne, Michael Lindner-D'Addario, AkshatKumar Nigam, Cher Tian Ser, Zhenpeng Yao, and Alán Aspuru-Guzik. Data-driven strategies for accelerated materials design. *Accounts of Chemical Research*, 54(4):849–860, 2021.

Mihika Prabhu, Charles Roques-Carmes, Yichen Shen, Nicholas Harris, Li Jing, Jacques Carolan, Ryan Hamerly, Tom Baehr-Jones, Michael Hochberg, Vladimir Čeperić, et al. Accelerating recurrent ising machines in photonic integrated circuits. *Optica*, 7(5):551–558, 2020.

Carla Rodríguez, Sören Arlt, Leonhard Möckl, and Mario Krenn. Xlumina: An auto-differentiating discovery framework for super-resolution microscopy. *arXiv:2310.08408*, 2023.

Carlos Ruiz-Gonzalez, Sören Arlt, Jan Petermann, Sharareh Sayyad, Tareq Jaouni, Ebrahim Karimi, Nora Tischler, Xuemei Gu, and Mario Krenn. Digital discovery of 100 diverse quantum experiments with pytheus. *Quantum*, 7:1204, 2023.

Neil V Sapra, Ki Youl Yang, Dries Vercruysse, Kenneth J Leedle, Dylan S Black, R Joel England, Logan Su, Rahul Trivedi, Yu Miao, Olav Solgaard, Robert L Byer, and Jelena Vučković. On-chip integrated laser-driven particle accelerator. *Science*, 367(6473):79–83, 2020.

Trieu H Trinh, Yuhuai Wu, Quoc V Le, He He, and Thang Luong. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Julius Wallnöfer, Alexey A Melnikov, Wolfgang Dür, and Hans J Briegel. Machine learning for long-distance quantum communication. *PRX Quantum*, 1(1):010301, 2020.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. On layer normalization in the transformer architecture. In *International Conference on Machine Learning*, pages 10524–10533. PMLR, 2020.

Remmy Zen, Jan Olle, Luis Colmenarez, Matteo Puviani, Markus Müller, and Florian Marquardt. Quantum circuit discovery for fault-tolerant logical state preparation with reinforcement learning. *arXiv:2402.17761*, 2024.

A Target classes

In the following table we show the hand-picked targets. These are classes of quantum states which are of interest in different areas of quantum physics. For each class, the first three states (four, six and eight particles) are given as strings in the same way that they are used to prompt the model. The column "correct states" shows, up to which index N the best model output matches the target (the first N states are correct). An infinity sign ∞ means, that the meta-solution perfectly matches the target.

State Name	size	Quantum State string	correct states	previously known
Spin 1/2	4	+1[xxxx] +1[xxyx] +1[xyxx] +1[yxxx] +1[yxyx]	∞	**unknown**
	6	+1[xxxxxx] +1[xxxyxx] +1[xxyxxx] +1[xyxxxx] +1[xyxyxx] +1[yxxxxx] +1[yxxyxx] +1[yxyxxx]		
	8	+1[xxxxxxxx] +1[xxxxyxxx] +1[xxxyxxxx] +1[xxyxxxxx] +1[xxyxyxxx] +1[xyxxxxxx] +1[xyxxyxxx] +1[xyxyxxxx] +1[yxxxxxxx] +1[yxxxyxxx] +1[yxxyxxxx] +1[yxyxxxxx] +1[yxyxyxxx]		
Majumdar-Ghosh	4	-1[xxyy] +2[xyxy] -1[xyyx] -1[yxxy] +2[yxyx] -1[yyxx]	∞	**unknown**
	6	-1[xxyxyy] +1[xxyyxy] +1[xyxxyy] -1[xyxyyx] -1[xyyxxy] +1[xyyxyx] -1[yxxyxy] +1[yxxyyx] +1[yxyxxy] -1[yxyyxx] -1[yyxxyx] +1[yyxyxx]		
	8	-1[xxyxyxyy] +1[xxyxyyxy] +1[xxyyxxyy] -1[xxyyxyxy] +1[xyxxyxyy] -1[xyxxyyxy] -1[xyxyxxyy] +2[xyxyxyxy] -1[xyxyxyyx] -1[xyxyyxxy] +1[xyxyyxyx] -1[xyyxxyxy] +1[xyyxxyyx] +1[xyyxyxxy] -1[xyyxyxyx] -1[yxxyxyxy] +1[yxxyxyyx] +1[yxxyyxxy] -1[yxxyyxyx] +1[yxyxxyxy] -1[yxyxxyyx] -1[yxyxyxxy] +2[yxyxyxyx] -1[yxyxyyxx] -1[yxyyxxyx] +1[yxyyxyxx] -1[yyxxyxyx] +1[yyxxyyxx] +1[yyxyxxyx] -1[yyxyxyxx]		
Bell pairs 2d	4	+1[xxxx] +1[xxyy] +1[yyxx] +1[yyyy]	∞	known
	6	+1[xxxxxx] +1[xxxxyy] +1[xxyyxx] +1[xxyyyy] +1[yyxxxx] +1[yyxxyy] +1[yyyyxx] +1[yyyyyy]		
	8	+1[xxxxxxxx] +1[xxxxxxyy] +1[xxxxyyxx] +1[xxxxyyyy] +1[xxyyxxxx] +1[xxyyxxyy] +1[xxyyyyxx] +1[xxyyyyyy] +1[yyxxxxxx] +1[yyxxxxyy] +1[yyxxyyxx] +1[yyxxyyyy] +1[yyyyxxxx] +1[yyyyxxyy] +1[yyyyyyxx] +1[yyyyyyyy]		
Bell pairs 3d	4	+1[xxxx] +1[yyxx] +1[zzxx]	∞	known
	6	+1[xxxxxx] +1[xxyyxx] +1[xxzzxx] +1[yyxxxx] +1[yyyyxx] +1[yyzzxx] +1[zzxxxx] +1[zzyyxx] +1[zzzzxx]		
	8	+1[xxxxxxxx] +1[xxxxyyxx] +1[xxxxzzxx] +1[xxyyxxxx] +1[xxyyyyxx] +1[xxyyzzxx] +1[xxzzxxxx] +1[xxzzyyxx] +1[xxzzzzxx] +1[yyxxxxxx] +1[yyxxyyxx] +1[yyxxzzxx] +1[yyyyxxxx] +1[yyyyyyxx] +1[yyyyzzxx] +1[yyzzxxxx] +1[yyzzyyxx] +1[yyzzzzxx] +1[zzxxxxxx] +1[zzxxyyxx] +1[zzxxzzxx] +1[zzyyxxxx] +1[zzyyyyxx] +1[zzyyzzxx] +1[zzzzxxxx] +1[zzzzyyxx] +1[zzzzzzxx]		
GHZ	4	+1[xxxx] +1[yyyy]	∞	known
	6	+1[xxxxxx] +1[yyyyyy]		
	8	+1[xxxxxxxx] +1[yyyyyyyy]		
W	4	+1[xxxy] +1[xxyx] +1[xyxx] +1[yxxx]	∞	known
	6	+1[xxxxxy] +1[xxxxyx] +1[xxxyxx] +1[xxyxxx] +1[xyxxxx] +1[yxxxxx]		
	8	+1[xxxxxxxy] +1[xxxxxxyx] +1[xxxxxyxx] +1[xxxxyxxx] +1[xxxyxxxx] +1[xxyxxxxx] +1[xyxxxxxx] +1[yxxxxxxx]		

State Name	size	Quantum State string	correct states	previously known
GHZ x W	4	+1[xxxy] +1[xxyx] +1[yyxy] +1[yyyx]	3	unknown
	6	+1[xxxxxy] +1[xxxxyx] +1[xxxyxx] +1[yyyxxy] +1[yyyxyx] +1[yyyyxx]		
	8	+1[xxxxxxxy] +1[xxxxxxyx] +1[xxxxxyxx] +1[xxxxyxxx] +1[yyyyxxxy] +1[yyyyxxyx] +1[yyyyxyxx] +1[yyyyyxxx]		
W x W	4	+1[xyxy] +1[xyyx] +1[yxxy] +1[yxyx]	3	unknown
	6	+1[xxyxxy] +1[xxyxyx] +1[xxyyxx] +1[xyxxxy] +1[xyxxyx] +1[xyxyxx] +1[yxxxxy] +1[yxxxyx] +1[yxxyxx]		
	8	+1[xxxyxxxy] +1[xxxyxxyx] +1[xxxyxyxx] +1[xxxyyxxx] +1[xxyxxxxy] +1[xxyxxxyx] +1[xxyxxyxx] +1[xxyxyxxx] +1[xyxxxxxy] +1[xyxxxxyx] +1[xyxxxyxx] +1[xyxxyxxx] +1[yxxxxxxy] +1[yxxxxxyx] +1[yxxxxyxx] +1[yxxxyxxx]		
Dicke 2	4	+1[xzzx] +1[zxzx] +1[zzxx]	3	unknown
	6	+1[xxzzxx] +1[xzxzxx] +1[xzzxxx] +1[zxxzxx] +1[zxzxxx] +1[zzxxxx]		
	8	+1[xxxzzxxx] +1[xxzxzxxx] +1[xxzzxxxx] +1[xzxxzxxx] +1[xzxzxxxx] +1[xzzxxxxx] +1[zxxxzxxx] +1[zxxzxxxx] +1[zxzxxxxx] +1[zzxxxxxx]		
GHZ x GHZ	4	+1[xxxx] +1[xxyy] +1[yyxx] +1[yyyy]	3	unknown
	6	+1[xxxxxx] +1[xxxyyy] +1[yyyxxx] +1[yyyyyy]		
	8	+1[xxxxxxxx] +1[xxxxyyyy] +1[yyyyxxxx] +1[yyyyyyyy]		
Dyck 2	4	+1[yyzz] +1[yzyz]	3	unknown
	6	+1[yyyzzz] +1[yyzyzz] +1[yyzzyz] +1[yzyyzz] +1[yzyzyz]		
	8	+1[yyyyzzzz] +1[yyyzyzzz] +1[yyyzzyzz] +1[yyyzzzyz] +1[yyzyyzzz] +1[yyzyzyzz] +1[yyzyzzyz] +1[yyzzyyzz] +1[yyzzyzyz] +1[yzyyyzzz] +1[yzyyzyzz] +1[yzyyzzyz] +1[yzyzyyzz] +1[yzyzyzyz]		
Dyck 1	4	+1[yzxx]	3	unknown
	6	+1[yyzzxx] +1[yzyzxx]		
	8	+1[yyyzzzxx] +1[yyzyzzxx] +1[yyzzyzxx] +1[yzyyzzxx] +1[yzyzyzxx]		
Dicke 1	4	+1[xzxx] +1[zxxx]	2	unknown
	6	+1[xxzzxx] +1[xzxzxx] +1[xzzxxx] +1[zxxzxx] +1[zxzxxx] +1[zzxxxx]		
	8	+1[xxxzzzxx] +1[xxzxzzxx] +1[xxzzxzxx] +1[xxzzzxxx] +1[xzxxzzxx] +1[xzxzxzxx] +1[xzxzzxxx] +1[xzzxxzxx] +1[xzzxzxxx] +1[xzzzxxxx] +1[zxxxzzxx] +1[zxxzxzxx] +1[zxxzzxxx] +1[zxzxxzxx] +1[zxzxzxxx] +1[zxzzxxxx] +1[zzxxxzxx] +1[zzxxzxxx] +1[zzxzxxxx] +1[zzzxxxxx]		
Dicke 5	4	+1[zzzx]	2	unknown
	6	+1[xzzzxx] +1[zxzzxx] +1[zzxzxx] +1[zzzxxx]		
	8	+1[xxzzzxxx] +1[xzxzzxxx] +1[xzzxzxxx] +1[xzzzxxxx] +1[zxxzzxxx] +1[zxzxzxxx] +1[zxzzxxxx] +1[zzxxzxxx] +1[zzxzxxxx] +1[zzzxxxxx]		
AKLT	4	-1[xzxx] +1[yyxx] -1[zxxx]	2	unknown
	6	-1[xyzxxx] +1[xzyxxx] +1[yxzxxx] -1[yzxxxx] -1[zxyxxx] +1[zyxxxx]		
	8	-1[xyyzxxxx] +1[xyzyxxxx] +2[xzxzxxxx] -1[xzyyxxxx] +1[yxyzxxxx] -1[yxzyxxxx] -1[yyxzxxxx] +1[yyyyxxxx] -1[yyzxxxxx] -1[yzxyxxxx] +1[yzyxxxxx] -1[zxyyxxxx] +2[zxzxxxxx] +1[zyxyxxxx] -1[zyyxxxxx]		
Motzkin small	4	+1[xyxx] +1[zzxx]	2	unknown
	6	+1[xyzxxx] +1[xzyxxx] +1[zxyxxx] +1[zzzxxx]		
	8	+1[xxyyxxxx] +1[xyxyxxxx] +1[xyzzxxxx] +1[xzyzxxxx] +1[xzzyxxxx] +1[zxyzxxxx] +1[zxzyxxxx] +1[zzxyxxxx] +1[zzzzxxxx]		

State Name	size	Quantum State string	correct states	previously known
Dicke 3	4	+1[xyzx] +1[xzyx] +1[yxzx] +1[yzxx] +1[zxyx] +1[zyxx]	1	unknown
	6	+1[xxyzxx] +1[xxzyxx] +1[xyxzxx] +1[xyzxxx] +1[xzxyxx] +1[xzyxxx] +1[yxxzxx] +1[yxzxxx] +1[yzxxxx] +1[zxxyxx] +1[zxyxxx] +1[zyxxxx]		
	8	+1[xxxyzxxx] +1[xxxzyxxx] +1[xxyxzxxx] +1[xxyzxxxx] +1[xxzxyxxx] +1[xxzyxxxx] +1[xyxxzxxx] +1[xyxzxxxx] +1[xyzxxxxx] +1[xzxxyxxx] +1[xzxyxxxx] +1[xzyxxxxx] +1[yxxxzxxx] +1[yxxzxxxx] +1[yxzxxxxx] +1[yzxxxxxx] +1[zxxxyxxx] +1[zxxyxxxx] +1[zxyxxxxx] +1[zyxxxxxx]		
Dicke 4	4	+1[xxyy] +1[xyxy] +1[xyyx] +1[yxxy] +1[yxyx] +1[yyxx]	1	unknown
	6	+1[xxxxyy] +1[xxxyxy] +1[xxxyyx] +1[xxyxxy] +1[xxyxyx] +1[xxyyxx] +1[xyxxxy] +1[xyxxyx] +1[xyxyxx] +1[xyyxxx] +1[yxxxxy] +1[yxxxyx] +1[yxxyxx] +1[yxyxxx] +1[yyxxxx]		
	8	+1[xxxxxxyy] +1[xxxxxyxy] +1[xxxxxyyx] +1[xxxxyxxy] +1[xxxxyxyx] +1[xxxxyyxx] +1[xxxyxxxy] +1[xxxyxxyx] +1[xxxyxyxx] +1[xxxyyxxx] +1[xxyxxxxy] +1[xxyxxxyx] +1[xxyxxyxx] +1[xxyxyxxx] +1[xxyyxxxx] +1[xyxxxxxy] +1[xyxxxxyx] +1[xyxxxyxx] +1[xyxxyxxx] +1[xyxyxxxx] +1[xyyxxxxx] +1[yxxxxxxy] +1[yxxxxxyx] +1[yxxxxyxx] +1[yxxxyxxx] +1[yxxyxxxx] +1[yxyxxxxx] +1[yyxxxxxx]		
GHZ 3d x GHZ 3d	4	+1[xxxx] +1[xxyy] +1[xxzz] +1[yyxx] +1[yyyy] +1[yyzz] +1[zzxx] +1[zzyy] +1[zzzz]	1	unknown
	6	+1[xxxxxx] +1[xxxyyy] +1[xxxzzz] +1[yyyxxx] +1[yyyyyy] +1[yyyzzz] +1[zzzxxx] +1[zzzyyy] +1[zzzzzz]		
	8	+1[xxxxxxxx] +1[xxxxyyyy] +1[xxxxzzzz] +1[yyyyxxxx] +1[yyyyyyyy] +1[yyyyzzzz] +1[zzzzxxxx] +1[zzzzyyyy] +1[zzzzzzzz]		
Motzkin	4	+1[xyzx] +1[xzyx] +1[zxyx] +1[zzzx]	1	unknown
	6	+1[xxyyxx] +1[xyxyxx] +1[xyzzxx] +1[xzyzxx] +1[xzzyxx] +1[zxyzxx] +1[zxzyxx] +1[zzxyxx] +1[zzzzxx]		
	8	+1[xxyyzxxx] +1[xxyzyxxx] +1[xxzyyxxx] +1[xyxyzxxx] +1[xyxzyxxx] +1[xyzxyxxx] +1[xyzzzxxx] +1[xzxyyxxx] +1[xzyxyxxx] +1[xzyzzxxx] +1[xzzyzxxx] +1[xzzzyxxx] +1[zxxyyxxx] +1[zxyxyxxx] +1[zxyzzxxx] +1[zxzyzxxx] +1[zxzzyxxx] +1[zzxyzxxx] +1[zzxzyxxx] +1[zzzxyxxx] +1[zzzzzxxx]		