

Interpretable Representations of Artificially Discovered Experiments in Quantum Optics via Virtual Reality

Master's Thesis in Physics

by

Philipp Schmidt
March 14, 2024

Max-Planck Institute for the Science of Light
Friedrich-Alexander-Universität Erlangen-Nürnberg



Supervisor: Dr. Mario Krenn

Declaration of Academic Integrity:

I, Philipp Schmidt, declare, that this thesis submitted to achieve the academic degree "Master of Science" is my own original work, written by myself. I did not use any other sources or aids than those referenced in this thesis.

Place, Date

Signature

Acknowledgements:

Firstly, I want to thank Dr. Mario Krenn for providing me the opportunity to work in his group and for his supervision and guidance throughout this project. Secondly, I want to express thanks towards Sören Arlt and Carlos Ruiz Gonzalez for co-supervision and constant availability for any questions I randomly came up with during my work even when they were busy with their own projects.

I also want to thank Dr. Carla Rodríguez for always pushing me to believe in my work as well as Dr. Xuemei Gu for sharing her great expertise when I needed it.

I thank Carlos and Sören for proofreading this thesis.

Deep thanks go towards my fiancée, for never giving up on me.

Abstract

Artificial Intelligence (AI) has become increasingly powerful in assisting researchers with their tasks. This development poses the challenge, of how human scientists can obtain true scientific understanding when solutions to research problems are no longer produced by them, but by AI systems. Any such answer to a research question has the potential to contain new insights and concepts implicitly discovered and used by AI, but thus far unknown to humans. Thereby AI solutions create a demand for explanations that the AIs themselves often do not satisfy. Human scientists then are restricted to attempting to deduce the underlying principles by studying the AI results alone. Depending on the complexity of the results this can be very difficult. One possibility of alleviating this inherent challenge is to increase the researcher's capability to meaningfully engage with the AI output. This thesis explores the application of Virtual Reality (VR) in analyzing AI-designed quantum optics experiments in graph form. It presents an experimental VR application developed for this task and demonstrates the ability to obtain new understanding through its use. A VR-augmented artificial discovery workflow is presented, incorporating the AI in a human-in-the-loop cycle by restricting its search space using human intuition. Results obtained include new understanding of specific high-dimensional state generation experiments and the human-assisted AI discovery of a measurement setup for highly entangled states beyond qubits. In addition, an AI-discovered resource-efficient variant of a core principle in quantum communication is conceptualized and then expanded to higher dimensions. The obtained understanding is subsequently used to extend an already known AI-inspired experimental concept obtained via the graph representation in previous work.

Contents

1	Introduction	1
1.1	Scientific Insight from Artificial Discovery	2
1.2	Complex Entanglement in Quantum Optics	3
1.3	Immersive Technologies for Natural Science	3
1.3.1	What are Immersive Technologies	4
1.3.2	VR Experiences for Analysing Data in Natural Science	5
2	Physical Background	8
2.1	Entanglement	8
2.2	An introduction to entanglement swapping	10
3	From Path Identity to PyTheus	13
3.1	Path Identity for Generating Quantum Mechanical Superpositions	13
3.2	Encoding Experiments in Colored Graphs	16
3.3	PyTheus	21
4	AriadneVR	26
4.1	Setup and Workflow:	26
4.2	Graph-encoded Experiment Analysis in Virtual Reality	27
4.2.1	Visualisation Features;	27
4.2.2	Exploration Features:	30
5	Virtual Reality-augmented Discovery	32
5.1	Interpretable Structures	32
5.1.1	Graphs with candidate structures for interpretability	32
5.1.2	Visualizing Loops of Zero-Contribution	35
5.1.3	Discovery of a 2-dimensional Halo in a 3-particle 5-dimensional GHZ-state graph	36
5.1.4	Conclusions on Finding Interpretable Structures	38
5.2	Steering PyTheus by Geometry Based Constriction of the Search Space	38
5.2.1	2-dim. GHZ-state analyzers	39
5.2.2	3-dim. GHZ-state analyzers	41
5.2.3	Conclusions on Steered PyTheus Discoveries	42
5.3	Resource Efficient Multipair Entanglement Swapping	42
5.3.1	Three Pair Entanglement Swapping	43
5.3.2	Generalization to Arbitrary Odd Pairs	44
5.3.3	Application to d-dimensional Entanglement Swapping	54
5.3.4	Generalising beyond ES graphs and the Connection to the Halo concept	57
5.3.5	On the Contribution of Ariadne VR	59
6	Summary	60
7	Conclusion and Outlook	61

References	63
A Discovered Graphs	70
A.1 3-particle-7-dimensional GHZ-state	70
A.2 2dim.-GHZ-analyzer search:	70
A.2.1 Search Geometry	70
A.2.2 12-edge analyzer	70
A.2.3 20-edge analyzer	71
A.3 3dim.-GHZ-analyzer search:	71
A.3.1 Search Geometry:	71
A.3.2 4-particle 3-dim. GHZ-state analyzer	72
A.4 Semi-3d-entanglement-swapping	72
A.5 4-pair-3dimensional entanglement swapping:	72
B AriadneVR Documentation	73
B.1 UI-components	74
B.1.1 button	74
B.1.2 controller-menu	75
B.1.3 disc-menu	76
B.1.4 dataslate	76
B.1.5 controlpanel	77
B.1.6 graph-library	78
B.1.7 graph-editor	78
B.2 Graph-components	80
B.2.1 edge	80
B.2.2 vertex	80
B.2.3 perfect-matching	81
B.2.4 graph	82
C Code snippets	83
C.1 computing perfect matchings:	83
C.2 generating 2d-entanglement swapping graphs:	83

1 Introduction

Through more powerful machines, new techniques, and more available data Artificial Intelligence (AI) has found increasingly fruitful applications in assisting researchers in their work over recent years. Modern AI goes beyond standard computational research tasks, like simulation and statistical data analysis [1, 2]. Today, there is AI assisting mathematicians with inspiration [3], designing material prototypes and molecules [4, 5, 6, 7], discovering measurement techniques [8] and proposing experiments [9, 10, 11, 12]. Such applications can generate some form of new scientific output, thereby producing solutions to research problems that have not been discovered by human scientists. This poses a new challenge to researchers employing such methods. An AI-discovered solution to a specific scientific question, for example, a room-temperature superconductor would be of tremendous technological and scientific value. However, it is not guaranteed, that human researchers would easily be able to obtain new scientific insights from this solution. A deep understanding of the discovery is achieved when researchers can conceptualize the fundamental physics underlying the super-conductivity of this hypothetical material. Thus, scientific discovery does not automatically result in scientific understanding [1].

A perfect scientific AI would of course provide comprehensible explanations alongside their solutions, however, current AI generally lacks this capability [1]. In many cases, the way how the AI produced the output is not interpretable, due to the high complexity of the learned mathematical transformation between human-readable input and human-readable output making the process of the machine very obscure [2]. This issue has spawned a research field of its own, explainable AI [13], aiming to understand complex AI systems. However, even if the methodology of the AI algorithm is transparent, they do not necessarily provide the reasoning required to explain their output. Therefore, to obtain new understanding researchers are often required to analyze the AI results, in the hope of being able to find new generalizable scientific principles [1]. In a sense, they need to reverse engineer the fundamental insights hidden in the output. If the solutions are very complex this can be a very challenging task [1]. One approach to alleviate this problem is to improve the communication between AI and the researchers. This refers to finding analysis techniques that meaningfully improve a researcher's ability to engage with complex AI outputs. Examples are choosing better data representations or using improved ways of engaging with the output data [1].

In this work, improving the analysis methodology is explored by choosing a more appropriate platform, Virtual Reality (VR), for the output of a specific AI designing quantum optical experiments. Such experiments can be represented by colored graphs, and this thesis will examine ways to utilize this abstract encoding to leverage the benefits of VR for new artificially generated scientific understanding in quantum optics. To this end, a custom analysis tool, *AriadneVR*, was developed to realize a VR-augmented analysis workflow for graph-encoded experiments. *AriadneVR* is an experimental application and has been developed alongside its use on current AI-discovered experiments of interest. The results obtained with its assistance, constitute new insights into the graph representations of selected exper-

iments (see. Sec. 5.1), implementation, and successful testing of a VR-assisted human-in-the-loop cycle for new AI discoveries (see. Sec. 5.2), as well as detailed analysis of the AI-discovery of a resource-efficient entanglement swapping scheme (see. Sec. 5.3). The concepts extracted from the latter could be used to generalize to high dimensional entanglement swapping and further abstracted to a significant extension of, a known experimental concept obtained from the graph representation in other work [14]. AriadneVR and selected results from its application, obtained as part of this thesis, have been published in a preprint related to this work Ref. [15].

The following two chapters will briefly introduce the topic of AI-based discovery in natural science and the use of Virtual Reality for natural science research to provide context on the specific applications presented in this thesis.

1.1 Scientific Insight from Artificial Discovery

This section provides a brief overview of ways, in which modern AI contributes to scientific discovery in natural science with a focus on physics. It constitutes a summary of the relevant points from three recent review articles on the topic, Ref. [1], discussing AI contribution from a general conceptual angle, Ref. [2], a broad collection of state-of-the-art AI methods, and Ref. [16] a detailed summary of AI applications in quantum physics.

Krenn et al. outline three categories of AI contribution to scientific understanding, *"Computational Microscopes"*, a *"Resource of Inspiration"* and finally an *"Agent of Understanding"* [1]. These categories are intentionally broad to not tie themselves to specific AI applications and implementations. *"Agents of Understanding"* refer to hypothetical future AIs, creating scientific insight completely autonomously while efficiently communicating their findings to humans. Such systems do not fully exist yet and are not relevant to the context of this thesis, hence, they are excluded from the subsequent summary.

"Computational Microscopes" are AI applications that allow researchers to study processes otherwise inaccessible, i.e. algorithms generating data where it is hard or even impossible to obtain experimentally [1]. Examples are molecular- [17], or quantum dynamics simulations [18, 16]. Modern machine-learning techniques can make such simulations more efficient [2]. Such microscopes only contribute to scientific understanding if the data they provide leads human researchers to new conceptual insights [1]. Thus, their usefulness can be increased by improving the interpretability of the generated data. Krenn et al. introduce stereoscopic 3D representations via VR as one option, and data sonification as another [1]. Examples for sonification are found in Astronomy [19].

"Resources of Inspiration" describe AI systems that prompt scientists to develop new general understanding. In general, it can be divided into two categories: Algorithms that inspire through their output, and algorithms that inspire through their methods. The former refers to applications generating interesting novel scientific results, but also systems spotting irregularities or relations in data or literature. The latter describes cases, where analyzing a learned model or AI strategy leads to new understanding [1]. Examples of potential resources for

inspiration are algorithms proposing new scientific hypotheses or obtaining improved representations of data [2, 1]. In quantum physics examples of applications are designs of new experiments [9, 10, 11, 12, 20], discovering quantum error correction codes [21], or quantum circuit design [22] [16]. The generated results and learned strategies of such AI have the potential of implicitly using unknown physical principles. Similar to computational microscopes, these underlying concepts need to be extracted with human effort, potentially assisted by an appropriate representation of the output [1]. PyTheus [12], the AI responsible for the AI discoveries of quantum optics experiments discussed in this thesis is one of these algorithms.

1.2 Complex Entanglement in Quantum Optics

The AI-discovered experiments discussed in this work largely relate to the generation or measurement of complex photonic entangled states. Photonic entanglement is a key resource both in studying the fundamentals of quantum mechanics and modern technical applications in quantum computing, quantum cryptography and communication [23, 24]. In these contexts, photons are often treated as two-dimensional qubits, e.g. realized in their polarization degree of freedom. Such systems have been used to perform loop-hole free tests of Bell inequalities [25, 26] or in entanglement swapping [27, 28] a key component of quantum repeaters, required for building large quantum communication networks [29, 24]. Interesting applications result from systems with more than two qubits allowing new refutations of local realism other than Bell inequalities [30, 31] or the non-local generation of entanglement across multiple different locations [32]. Many of these applications are enhanced when using high-dimensional photon states e.g. by considering their orbital angular momentum degree of freedom [23]. High dimensional systems (qudits) lead to stronger violations of Bell inequalities and are more robust against noise, making them more easily verifiable in experiments [33, 23]. Using qudits also increases the robustness of quantum communication networks as well as the information capacity of each carrier [29, 24]. Recent experimental progress saw the generation of increasingly complex high dimensional entangled pair and multiphoton states [23, 34, 35, 36, 37, 38]. A key challenge in utilizing the resources provided by such experiments is realizing measurement setups distinguishing between different highly entangled states. Such analyzers are required to realize for example high-dimensional quantum teleportation schemes [24]. Even for 2-qubit systems, it is impossible to build complete analyzers, distinguishing between all Bell states, with linear optics alone [39]. In 2-qudit, partial analyzers require additional, ancillary photons or nonlinear optical processes to be realized [40, 24]. 3-dimensional teleportation of Bell-states has been experimentally achieved recently [41, 42] employing such partial analyzers. Those recent advancements indicate the large potential for new and interesting fundamental and technical applications of complex photonic entanglement

1.3 Immersive Technologies for Natural Science

Immersive technology is one of the methods proposed by Krenn et al. to be employed to increase researchers ability to extract insights from AI [1] and is a core component of this thesis work. Immersive displays have diverse use cases for science, ranging from putting animals in virtual environments to study their behavior

[43], to manipulating microscopic systems almost in real-time via a virtual reality interface [44]. An especially vibrant field is molecular analysis and design assisted by immersive technology [45, 46]. This thesis pursues the application of such techniques for artificial scientific discovery in quantum optics, analyzing abstract AI-discovered structures relating to fundamental aspects of quantum physics. In light of this application on AI-produced data this short overview focuses on data analysis and treatment applications in the natural sciences and adjacent topics.

1.3.1 What are Immersive Technologies

The idea of fully immersing oneself in a computer-simulated environment was conceived in 1965 by Sutherland in his essay *"The Ultimate Display"*. In it, he describes his vision of using computer displays to provide "a looking glass into a mathematical wonderland" [47]. He outlines incorporating multiple senses like hearing and touch into his display, both concepts that have since been realized [48, 49, 50]. With such a display, he conjectures "we can learn to know them [mathematical phenomena] as well as we know our own natural world." In 1968 he built the first head-mounted stereoscopic 3D display out of two miniaturized cathode ray tubes [51]. Since those pioneering experiments, immersive displays have undergone a variety of technical iterations. Today, they are sorted into two categories. This follows the nomenclature laid out in a recent primer on extended reality by Yuan et al. in Ref. [52].

Virtual Reality (VR): Virtual Reality is the complete substitution of the user's perception of the real world by a simulated environment. Most commonly this is achieved with head-mounted displays (HMD), but also includes CAVE environments [53], room-scale displays with screens or projection surfaces on all sides.

Augmented Reality (AR): Augmented Reality is the superposition of the real environment with virtual elements. This term is used both for applications using stereoscopic displays but also for much simpler systems overlaying a single image video feed from e.g. a smartphone, with simulated objects.

Generally speaking, most applications and devices lie on a spectrum between complete substitution and almost no modification of the real environment, referred to under the umbrella of Extended Reality (XR) in modern literature [52, 46]. Both augmented reality and the umbrella term of extended reality can also be referred to as Mixed Reality [52, 46]. This leads to a minor conflict of terminology. Hence, this work forgoes the use of the term Mixed Reality.

Through their relatively cheap availability on the consumer market, most modern XR applications use HMDs in conjunction with position and motion sensors. These devices track the user's head motion and gaze direction, potentially even including eye tracking. From the very beginning of the concept of XR in the 1960ties, interaction was a crucial part of the idea [47]. Modern headsets realize this by adding controllers tracked both in their 3 translational and 3 rotational degrees of freedom (DoF) or by motion tracking the user's hand without the need for a control surface. Beyond these general input methods, specialized haptic devices have been realized. An early example is the use of a robotic arm to simulate

the resistance of molecular force fields in 1990 [54]. Modern robotic arms are for example used in surgeon training where they simulate tissue resistance [50]. Beyond such specific devices there exist haptic gloves aiming to provide a general sensation of touch in virtual environments [50]. Thus, modern hardware can achieve stereoscopic 3D rendering while simultaneously providing intuitive interaction methods. This combination constitutes the defining toolkit offered by current XR technology.

The Meta Quest 2 The HMD used for VR during the work presented thesis is a Meta (formerly Oculus) Quest 2¹. It features two 6 DoF controllers but also supports hand gesture tracking with recent updates, however, the latter feature is not used in this work. It does not have eye-tracking technology but does not require external sensors for positional and directional tracking of the user's head or the controllers.

1.3.2 VR Experiences for Analysing Data in Natural Science

The toolkit outlined above enables a variety of benefits relating to data visualization and treatment. First and foremost, stereoscopic rendering enables the representation of data in true 3D as opposed to projective visualization on 2D screens. Representations can be freely adjusted, unrestricted by physical parameters like display size or room scale. Secondly, through motion tracking, the display is interactive in ways not possible on 2D displays. Users can move, scale, and rotate data to their needs and with intuitive gestures. In addition, potential web connection allows collaborative analysis even if the participants are far apart [55, 56, 46].

Examples of the use of VR in the natural sciences are found in many different fields. The unifying element of these applications is the presence of 3D-data or 3D-data representations. Prominent examples for the former are 3D-image stacks, for example, produced by confocal microscopy [57, 58] or medical scans [59, 60], but also single molecule localization microscopy point clouds [61, 62] or astronomical data [63, 64, 65]. Such tools are often not only for inspection but are geared towards performing vital data treatment tasks like image segmentation [59, 66].

Beyond imaging data, VR is used to visualize more abstract concepts, like molecular graphs or protein structures [67, 68, 69, 70, 71]. 3D-molecular structure is critical for the reactive behavior of molecules and studying it is a crucial aspect of molecular design tasks like drug design [45]. VR is used here to analyze structure, but also design or change molecules on the fly and even study their dynamic behavior when supplemented by simulations [45]. As generative AI is employed to suggest new drugs in this field [7], VR can be used to analyze the AI results [72]. Graphs are also a common way of representing relations in large datasets used for example in medicine and social science. With increasing size, such networks can become difficult to comprehend, or even fully visualize [73]. VR is not restricted by physical space and thus well suited to display such networks. As an example, VRNetzer [73] removes the visualization issue by allowing the user to choose different graph layouts and arbitrarily scale them to their

¹<https://www.meta.com/de/quest/products/quest-2/>

needs. It also includes a variety of graph exploration algorithms to support analysis.

The advantages of 3D visualization and interaction for data analysis tasks have been studied over many years [46]. Examples are the study of Laha et al. reporting significant benefits for analyzing volume data in immersive environments [74], or Ref. [75] reporting increased structural understanding of graphs in VR. Another example is the work of Millais, Jones, and Kelly claiming higher accuracy of insights gained from 3D visualization compared to 2D [76]. However, these examples suffer from low participant samples, and other work like that from Bach et al. suggests that the actual benefit of immersive analysis is task and medium-dependent [77]. Some modern applications mentioned before also include quantitative user studies reporting improved task completion times and better task performance [68, 78]. Research is conducted on the benefit of VR in the context of education and training in natural science [79, 80], and applications exist ranging from XR-assisted lab courses [81] to visualization aids for complex structures [82]. Such tools are meant to increase the comprehension of abstract concepts in students. A detailed review and evaluation on the state of the state-of-the-art research on the effects of immersive analysis are outside the scope of this thesis, however, the literature suggests that XR has real, but task- and implementation-dependent benefits for data analysis.

Applications of Particular Interest Here a few specific types of XR applications are presented that are of particular interest in the context of artificial scientific understanding as outlined in section 1.1.

One example are interactive molecular simulations. Tools like Narupa [71] allow the user to virtually touch and influence molecules. Users can initiate and observe chemical reactions by supplying the kinetic energy to initiate a reaction. As mentioned before one of the applications is in drug design, studying the reactive behavior of drug candidates [45]. Interactive molecular simulations are not a new idea [54, 83], but modern VR technology has elevated them to a new standard. They represent a direct implementation of Sutherland's "looking glass" [47], enabling humans to interact with nanoscopic representations of nature in otherwise unachievable ways. An application by Ferretti et al. not related to computational but physical microscopes takes this even further, connecting the user to real microscopic systems via VR in real-time [44]. Here, users can interact with microparticles and microorganisms within a virtual representation of a real system under a microscope. Their interactions are then transferred to the real system via optical tweezers. Virtual Reality is a crucial part of this application, allowing natural interaction with microscopic objects. Visualization can also be combined with other senses, as in the case of Ref. [49] where the surface of a protein visualized in VR is also expressed through sound or Ref. [48] where sonification is used to augment molecular simulations.

Through interactivity, XR hardware has the potential to improve the performance and efficiency of data treatment tasks. This can be used for example in the context of human-in-the-loop (HITL) analysis. HITL-analysis refers to tasks where an AI process is augmented by a human performing a certain task of the

analysis workflow [56]. An example is image segmentation, or point cloud labeling for machine learning tasks. VR hardware can provide a more efficient way of performing such tasks compared to regular computer input devices like a mouse and keyboard. An example is [84], an application where eye-tracking combined with VR visualization is utilized to track and mark cell paths in microscopy images. Other examples include data labeling, like tracing neurons in microscopy data [66]. Interactive molecular simulations have also been used to generate data for machine learning tasks, e.g by sampling the energy curves of chemical reactions by repeating them many times within the simulation in VR [85, 86].

XR is also used to study AI systems in the context of explainable AI. As in the previously mentioned case of network analysis, XR allows improved visualization of the complex internal relations resulting from machine learning tasks [87]. Thereby it can help elucidate the internal mechanism of such models. Lyu, Li, and Wang use interactive VR visualization in combination with the sonification of loss values for a more teaching-oriented approach in Ref. [88]. Their tool uses allow the user to observe and interact with small artificial neural networks during training to obtain an intuitive understanding of them.

2 Physical Background

At the core of this work, is the study of AI-proposed quantum optics experiments preparing or measuring different forms of photonic entanglement. This section serves as a basic introduction to these topics and provides the necessary terminology for later sections. It also provides a brief motivation in the form of important applications with an emphasis on entanglement swapping, due to its relevance to the AI discoveries discussed in later parts of this thesis.

2.1 Entanglement

Entanglement is one of the core emergent properties of quantum mechanics and the foundation of many of its paradoxical and unintuitive aspects [89]. It arises when two or more degrees of freedom in a quantum system are in a non-separable state, i.e. the state can not be written as a product state in any basis [90, pp. 267–269]:

$$|\psi\rangle \neq |\varphi_a\rangle |\varphi_b\rangle. \quad (2.1)$$

This unassuming property of some quantum systems has been a point of great discussion during the development of quantum mechanics [89]. Interpretational problems arise when one spatially separates the two degrees of freedom by ascribing each to a single particle in space. As their joint state cannot be written as a product state, the particles behave as a joint system regardless of physical distance. Thus, the global state changes instantly for both particles upon measuring one of them, seemingly breaking one of the core tenets of special relativity, that events can not influence each other when they are outside each other’s light cones. This constraint is called *locality* [89]. This issue led to the proposal that quantum mechanics is not a complete description of reality and needs to be augmented [91]. This culminated in the development of hidden-variable-theories underpinning quantum mechanics with additional unknown parameters responsible for its unintuitive predictions. Such hidden variables are also used to explain the inherent randomness of quantum mechanical measurements through probability distributions of these unknown properties. This is summarized by the *realism* constraint on theories, claiming that observations in a realistic theory should be predetermined by particle properties independent of measurements. The argument is, that would all hidden variable values be known, every measurement outcome would be fully deterministic [89]. In 1964 Bell offered a testable way of ruling out hidden variable descriptions of quantum mechanics by providing a quantifiable difference between the degrees of correlation encountered in hidden variable theories and quantum mechanics [92]. This has been turned into an experimentally testable inequality by Clauser et al. [93]. Since then experiments have been performed confirming that nature at the quantum level is neither local nor realistic [25, 26].

The profound properties of entanglement can be nicely explained using the informational approach of Brukner and Zeilinger outlined in a series of publications at the turn of the millennium [89, 94, 95]. They argue, that quantum randomness arises from the finiteness of information that can be carried in a quantum state. In their picture, a two-dimensional system can carry one bit of information. Hence, if e.g. an electron spin is aligned in the z-direction, the informational capacity is exhausted by the truth value of the statement: *the spin*

in the z -direction is up [94]. Other, independent statements, like *the spin in the x -direction is up* must thereby be fully undetermined. Two particles with two-dimensional Hilbert spaces hence carry two bits of information. Entanglement arises when the two-particle system cannot be accurately described by statements about the individual particles [89]. The degree of entanglement can then be defined by the exact amount of information in statements in joint properties. Maximal entanglement then means that all the available information capacity is used to define joint properties [95]. This offers an intuitive reason why quantum mechanical correlations can be larger than classical ones as demonstrated by the Bell inequalities and corresponding experiments. Classical correlations only arise from the properties of the individual single particles. For example, two particles in the state $|\uparrow\uparrow\rangle$ are classically correlated to be aligned, but this correlation is a consequence of the well-defined direction of each particle. For the state $|\uparrow\uparrow\rangle + |\downarrow\downarrow\rangle$, alignment is correlated independent of the direction of every single particle, as this is completely undetermined. Hence, quantum systems can contain more information in joint properties than in single-particle properties. They can have stronger correlations than what can be deduced from single-particle information alone [89].

An important example of maximally entangled states are the well-known Bell states for a 2-dimensional 2-particle system:

$$|\Phi^\pm\rangle = \frac{1}{\sqrt{2}} (|00\rangle \pm |11\rangle), \quad (2.2a)$$

$$|\Psi^\pm\rangle = \frac{1}{\sqrt{2}} (|01\rangle \pm |10\rangle). \quad (2.2b)$$

Here, 0 and 1 are used as general mode labels and can be replaced for example by horizontal (H) or vertical (V) polarization for photons or spin labels up (\uparrow) and down (\downarrow) for electrons. These states are maximally entangled, as the state of each particle is completely undetermined, all their information is in correlations [94].

A generalization of Bell-States to more than two particles are the Greenberger-Horne-Zeilinger (GHZ)-states. Originally they were conceived for three particles [30], but since have been expanded to n particles and dimensions larger than two [96, 97, 23].

$$|GHZ\rangle_n^d = \frac{1}{\sqrt{d}} \sum_{i=0}^{i=d-1} |i\rangle^{\otimes n}. \quad (2.3)$$

They share the property of maximal entanglement of Bell states. GHZ-states were principally constructed for an alternative to Bell's theorem for ruling out local realistic hidden-variable theories as they do not require inequalities to refute this worldview. [30, 96].

There are many other classes of entangled states exhibiting different degrees of entanglement. One way to classify the entanglement of complex systems is the Schmidt Rank Vector (SRV) [23]. It is the collection of the ranks of the reduced density matrices of each possible bi-partition of a quantum system. Due to the non-separability of entangled states, the state of a partition is mixed, having a rank strictly larger than one, as opposed to product states where the rank is exactly

one. If the rank is maximal the two subsystems of the partition are maximally entangled. This leaves a nuanced measure for the degree of entanglement of high-dimensional multi-particle systems. As an example, the Schmidt-Rank vector for a 2-dimensional 3-particle GHZ-state is $\{2,2,2\}$ as the reduced density matrix for any partition a is the fully mixed state:

$$\hat{\rho}_a = \frac{1}{2} (|00\rangle\langle 00| + |11\rangle\langle 11|)$$

2.2 An introduction to entanglement swapping

A large portion of this thesis's results deals with entanglement-swapping experiments. Here an introduction and a short overview of applications are presented for context following a very recent review on the current state of the field by Hu et al. [24].

Foundations of Entanglement Swapping Entanglement swapping was originally conceived as a source of entangled pairs to test Bell's inequalities [27]. The basic scheme is depicted in Figure 1(a). This was realized by Pan et al., demonstrating that it is possible to entangle photons that never interacted [28]. The basic scheme works as follows: First, one prepares two maximally entangled two-particle Bell states, e.g. $|\Phi^+\rangle$ and $|\Psi^-\rangle$, each from an independent source. Labeling the photons with letters $a - d$ one can rewrite the joined four particle state as:

$$\begin{aligned} |\Phi^+\rangle_{ab} |\Psi^-\rangle_{cd} = \frac{1}{2} & \left(|\Phi^+\rangle_{ac} |\Psi^-\rangle_{bd} \right. \\ & + |\Phi^-\rangle_{ac} |\Psi^+\rangle_{bd} \\ & - |\Psi^+\rangle_{ac} |\Phi^-\rangle_{bd} \\ & \left. - |\Psi^-\rangle_{ac} |\Phi^+\rangle_{bd} \right). \end{aligned} \quad (2.4)$$

Hence, performing a Bell State measurement on the photons b and d also projects the two remaining photons into a Bell State. This principle is independent of the initial choice of Bell states. The only resulting difference is a different pairing of Bell states on the right-hand side of equation (2.4). Note that the original experiments used two copies of $|\Psi^-\rangle$ as initial states. The choice here is made differently as the setups employed in later sections use $|\Phi^+\rangle|\Psi^-\rangle$ as the initial state. The original experiment uses a simple beam splitter setup with subsequent coincidence detection for the Bell state measurement. This type of measurement exploits the fact that due to the Hong-Ou-Mandel-effect[98], only the asymmetric singlet state $|\Psi^-\rangle$ leads to a photon in each output port of the beam splitter [28]. Hence, coincidence detection at the output ports of the beam splitter projects the two photons b and d in the singlet state.

Entanglement swapping is closely related to quantum teleportation. Essentially, the former can be seen as a special case of the latter. Quantum teleportation uses an entangled pair to transfer an unknown arbitrary quantum state by destroying it in one location and then recreating it somewhere else via quantum and classical communication channels. Entanglement swapping re-emerges from this technique when the particle carrying the unknown state is part of a Bell pair. Figure 1(b)

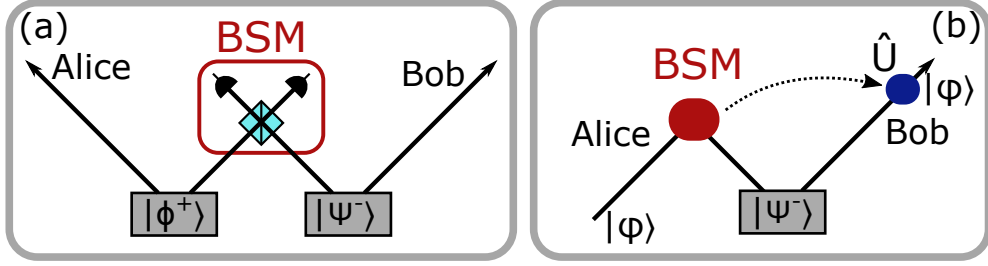


Figure 1: Schematic setups for entanglement swapping and quantum teleportation. **(a)** Entanglement swapping setup. Two particles, each part of a prepared Bell state are projected into a shared Bell state. Thereby the old partners sent to Alice and Bob are projected into a Bell state of their own. The Bell state measurement (BSM) is shown as in the original experiment [28], realized with a beam splitter and coincidence detection. **(b)** Basic quantum teleportation scheme. Alice receives an unknown state $|\varphi\rangle$ and one particle from a Bell state. After performing a Bell state measurement she transfers the result to Bob via a classical channel. This tells Bob which unitary operation \hat{U} to apply to his particle to recover the unknown state $|\varphi\rangle$ [99].

shows the standard teleportation setup [99, 24].

An early generalization of entanglement swapping is multiparticle entanglement swapping [32]. This scheme uses n -particle 2-dimensional GHZ states instead of Bell states. By performing a Bell- or GHZ state measurement on a subset of particles of two such states the remaining particles are projected into a GHZ-state of their own. This can also be used to create larger GHZ-states from smaller ones. E.g. two 4-particle GHZ-states combined with a Bell state measurement yield a 6-particle GHZ state after the swap. With such schemes, entanglement can be shared between multiple parties in a non-local way [32].

Applications of Entanglement Swapping The non-local establishment of entanglement possible with entanglement swapping can be used to create locality-loop-hole-free Bell test experiments [100]. For such tests, quantum mechanical entanglement must be verified at spacelike separation, i.e. at spacetime that cannot interact with each other. It has been shown that tests using high dimensional states produce a stronger violation of classical correlations, leading to more robust experiments [23, 24]. High-dimensional swapping and teleportation schemes require high-dimensional analyzers [24]. In many modern applications, the employed analyzers are incomplete, meaning they do not resolve the complete set of basis states, often only distinguishing one state versus all the rest [42, 101]. Complete analyzers cannot be realized without non-linear optics even in the 2-dimensional case [39]. Entanglement swapping also finds technical applications in quantum communication and quantum computing [24]. In the former, it constitutes a crucial element of quantum repeaters. These devices allow the transport of entanglement beyond the limits of loss rates and coherence length by performing consecutive swaps. Thereby, the distance any single state has to travel is much shorter than the distance between the final entangled pairs [24]. In quantum computing entanglement swapping allows the establishment of non-

local multipartite entanglement, a crucial step in the operation and upscaling of quantum computing networks [24].

3 From Path Identity to PyTheus

This section will provide the background behind PyTheus [12], the AI discovering the experiments at the core of this work as well as introduce the graph representation PyTheus is built upon. First, the fundamental experimental principle, Entanglement by Path Identity [102], that led to the development of the graph representation is introduced. Then, the concept of encoding linear quantum optics experiments as colored graphs is explained, before introducing the PyTheus library.

3.1 Path Identity for Generating Quantum Mechanical Superpositions

Path identity is an experimental concept in quantum optics, that allows for coherent superposition of photon states by removing "which path information" from photonic experiments. It relies on the fundamental property of quantum mechanics, that different possibilities for a process enter coherent superposition if there is no way of distinguishing between them. Critically, it is of no concern if two possibilities are actually distinguished or not, the possibility alone destroys coherence [103]. This short introduction to the path identity technique draws from the summary of research provided by Hochrainer et al. in their comprehensive review Ref. [103] as well as the more specific application for generating multipartite high-dimensional entanglement by path identity presented in Ref. [102].

By today, the fundamentals of the technique have been known for more than thirty years. However, after an initial decade of foundational works after its discovery, path identity has only been revived in 2014 and has since been expanded greatly in scope and applications [103]. The first implementation exploiting path identity is the Zou-Wang-Mandel experiment from 1991 [104], demonstrating the possibility for interference between photons based on superimposing their creation processes. A basic schematic of the experiment is shown in Figure 2(a). A simplified theoretical treatment of the experiment is sketched out here, see Ref. [103] for a more extensive version. The experiment is selected for cases where only one photon pair is created by the two identical weakly pumped non-linear crystals by spontaneous parametric down conversion. In the following discussion, the pair photons are labeled A and B . Photon A is either in paths a or c and

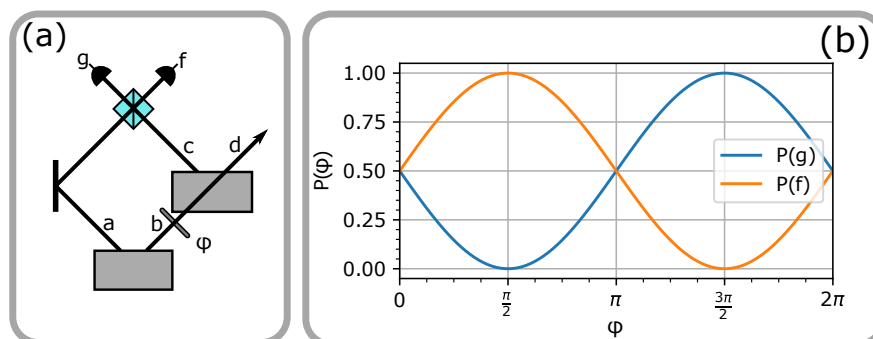


Figure 2: *The Zou-Wang-Mandel experiment. (a) Sketched setup of 2 pair sources with one of each emission path perfectly aligned. (b) Interference pattern of detection probabilities P observed at the beam splitter outputs dependent on the phase ϕ .*

photon B in b or d , depending on which crystal emits the pair. Hence, the input state of the beam splitter can be expressed in the path degree of freedom as:

$$|\psi_{AB}\rangle = \frac{1}{\sqrt{2}} \left(e^{i\varphi} \hat{a}^\dagger \hat{b}^\dagger + \hat{c}^\dagger \hat{d}^\dagger \right) |\text{vac}\rangle, \quad (3.1)$$

where \hat{x}^\dagger is the bosonic creation operator for a photon in path x . By perfectly aligning paths b and d the path modes are indistinguishable after the second crystal:

$$b \rightarrow d \Rightarrow |\psi\rangle = \frac{1}{\sqrt{2}} \left(e^{i\varphi} \hat{a}^\dagger \hat{d}^\dagger + \hat{c}^\dagger \hat{d}^\dagger \right) |\text{vac}\rangle = \frac{1}{\sqrt{2}} \hat{d}^\dagger \left(e^{i\varphi} \hat{a}^\dagger + \hat{c}^\dagger \right) |\text{vac}\rangle, \quad (3.2)$$

The state of the second photon $|\psi_A\rangle$ is now separable from photon B and a coherent superposition of path modes a and c :

$$|\psi_A\rangle = \frac{1}{\sqrt{2}} \left(e^{i\varphi} \hat{a}^\dagger + \hat{c}^\dagger \right) |\text{vac}\rangle. \quad (3.3)$$

By interfering the two path modes in a symmetric beam-splitter the input modes transform as:

$$\hat{a}^\dagger \rightarrow \frac{1}{\sqrt{2}} \left(i\hat{g}^\dagger + \hat{f}^\dagger \right) \quad (3.4a)$$

$$\hat{c}^\dagger \rightarrow \frac{1}{\sqrt{2}} \left(\hat{g}^\dagger + i\hat{f}^\dagger \right), \quad (3.4b)$$

leading to the output state:

$$\begin{aligned} \Rightarrow |\psi\rangle &= \frac{1}{2} \hat{d}^\dagger \left[e^{i\varphi} \left(i\hat{g}^\dagger + \hat{f}^\dagger \right) + \left(\hat{g}^\dagger + i\hat{f}^\dagger \right) \right] |\text{vac}\rangle \\ &= \frac{1}{2} \hat{d}^\dagger \left[\hat{g}^\dagger (1 + ie^{i\varphi}) + \hat{f}^\dagger (e^{i\varphi} + i) \right] |\text{vac}\rangle \\ &= \frac{1}{2} \hat{d}^\dagger \left[\hat{g}^\dagger \left(1 + e^{i(\varphi + \frac{\pi}{2})} \right) + i\hat{f}^\dagger \left(1 + e^{i(\varphi - \frac{\pi}{2})} \right) \right] |\text{vac}\rangle. \end{aligned} \quad (3.5)$$

Evidently, the coefficients of the \hat{g}^\dagger and \hat{f}^\dagger creators are phase shifted by π with respect to their φ -dependence, leading to the interference pattern at the output ports of the beam splitter depicted in Figure 2(b). This is remarkable, as the photon detected at the output ports of the beam splitter has not traveled through the phase plate. The reason for this effect is that there is no way to know in which crystal the photon originated. In fact, this information never existed, as Hochrainer et al. phrase it: *"The information of the photon's origin is not erased but was never created in the first place"* [103]. With misaligned paths b and d the interference vanishes as the state of the second photon A is no longer a coherent superposition like in equation (3.3). Instead, the degrees of freedom of photon B have to be traced out leading to a mixed state:

$$\begin{aligned} \hat{\rho}^A &= \text{Tr}_B (|\psi\rangle \langle\psi|) \\ &= \frac{1}{2} \sum_{p \in \{b,d\}} \langle 1_p | \left(|1_a, 1_b\rangle \langle 1_a, 1_b| + e^{i\varphi} |1_a, 1_b\rangle \langle 1_c, 1_d| + \right. \\ &\quad \left. e^{-i\varphi} |1_c, 1_d\rangle \langle 1_a, 1_b| + |1_c, 1_d\rangle \langle 1_c, 1_d| \right) |1_p\rangle \\ &= \frac{1}{2} |1_a\rangle \langle 1_a| + \frac{1}{2} |1_c\rangle \langle 1_c|. \end{aligned} \quad (3.6)$$

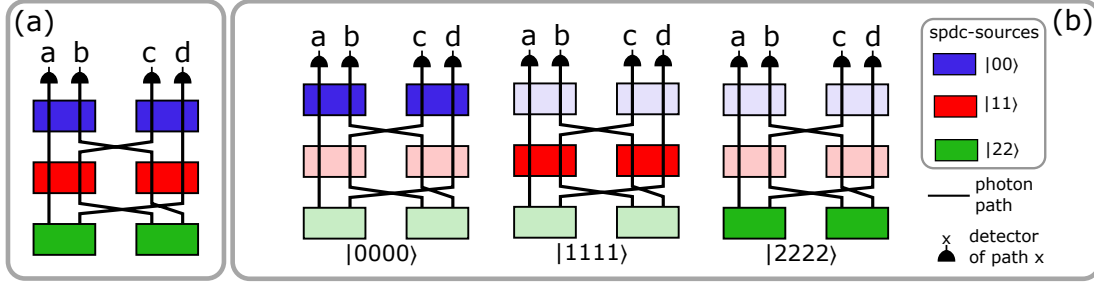


Figure 3: Proposed setup for 3-dimensional GHZ-state generation from Ref. [102]. **(a)** Setup consisting of 6 path-aligned pair sources emitting 2-photon orbital angular momentum states. The 0-mode is encoded as blue, and the 1-mode as red, and the 2-mode as green. **(b)** Three emission patterns leading to coincidence detection with their respective state. Through path identity, the four possibilities enter coherent superposition to form the GHZ state in eq. (3.7)

Thus, the phase dependence is lost and no interference is observed.

The Zou-Wang-Mandel experiment demonstrates that, by superimposing the path degree of freedom, quantum mechanical events can be made indistinguishable. This principle was quickly used to create sources for entangled photon pairs [105, 106], by superimposing both emission paths of two mode-shifted crystals, and more recently in measurement techniques with undetected photons [103]. In general, undetected-photon techniques rely on replacing the phase plate in the Zou-Wang-Mandel setup with a more complex sample. Its optical properties can then be deduced by changes in the observed interference pattern of photon A. Examples include imaging [107], spectroscopy [108] and microscopy [109].

Despite the relatively early adoption of path identity for constructing sources for entangled two-qubit states from regular pair sources [105, 106], it took many years until this principle was expanded to more general and complex state creation schemes. This advance was achieved through analysis of AI-generated proposals for such experiments [9, 102, 103]. The crucial step made by Krenn et al. in Ref. [102] is the realization, that combining paths from more than two coherently pumped pair sources can generate a variety of highly entangled multiparticle states in high dimension. Figure 3 shows one of their proposed setups, an experiment creating a 4-photon, 3-dimensional GHZ-state (see equation (3.7)). The experiment only consists of pair sources producing 2-photon orbital angular momentum product states $|00\rangle$, $|11\rangle$, and $|22\rangle$. The entangled state is then created through the clever alignment of emission paths. The desired GHZ state is obtained by selecting the experimental outcome on exactly one photon in each path called 4-fold coincidence detection [102]. The layout of photon paths restricts the possibilities for this outcome to three specific pairs of crystals, depicted in Figure 3(b). Through path identity and coherent pumping, these three patterns enter coherent superposition yielding the GHZ-state:

$$|GHZ\rangle_4^3 = \frac{1}{\sqrt{3}} (|0000\rangle + |1111\rangle + |2222\rangle) \quad (3.7)$$

This is only one of the multitude of setups proposed in reference [102], alongside high-dimensional Bell-states and setups for creating 2-dimensional GHZ-states of arbitrary even photon count to name some examples. Some of these experiments have since been realized, e.g. by Kysela et al. generating high dimensional bell states [36] and Bao et al., generating a three-dimensional four-particle GHZ-state [37]. These discoveries have subsequently led to a new representation of quantum optics experiments as graphs. This representation reduces the experimental complexity to path modes as vertices and correlations between paths introduced through pair sources as edges [103]. Such graphs are the basis for modern algorithms searching through the large combinatorial space of combinations of pair sources and path alignment layouts. They are introduced in the subsequent chapter.

3.2 Encoding Experiments in Colored Graphs

The graph representation of quantum optics allows the encoding of quantum optical experimental setups into colored undirected graphs. It is introduced and expanded upon in Refs. [110, 111, 112, 11, 12]. Essentially, it is a way to describe quantum states with graphs combined with instructions on how to create the states using linear photonics. However, it is not limited to state generation experiments but extends to state measurement devices, photonic gates, and communication protocols by simple reinterpretation of aspects of the state generation encoding [11, 12]. Initially, it was used to apply theorems from mathematical graph theory to derive statements about linear quantum optics [110], more recently it has proven itself to also be well-suited to explore the space of quantum optical experiments with computer algorithms to find new interesting setups [11, 12]. Important findings obtained with the graph representation have been experimentally verified recently [113, 114, 37], demonstrating the feasibility of the graph encoded experiments. The geometric nature of graphs is also what enables the application of Virtual Reality to such AI-discovered experiments in the ways proposed by this thesis.

Encoding State Generation State generation experiments aim at producing a specific photonic quantum state of light. This is a crucial prerequisite to many more advanced processes, like for example communication protocols that then manipulate the generated states to achieve other tasks [24]. However, generating complex photonic states of light and studying their properties is also a well-established method of probing the fundamentals of quantum mechanics itself [23]. One of the routine ways of generating photons in experiments is by weakly pumping non-linear crystals for spontaneous-parametric-down-conversion (SPDC), a 2^{nd} order non-linear process, where a high-frequency pump photon is converted into two lower frequency photons obeying both momentum and energy conservation [23]:

$$\omega_p = \omega_1 + \omega_2, \quad \vec{k}_p = \vec{k}_1 + \vec{k}_2, \quad (3.8)$$

where p stands for the pump photon. The simplified effective Hamiltonian of the process is given as [115]:

$$\hat{H} = i\hbar\eta \left(\hat{a}_1^\dagger \hat{a}_2^\dagger - \hat{a}_1 \hat{a}_2 \right) \quad (3.9)$$

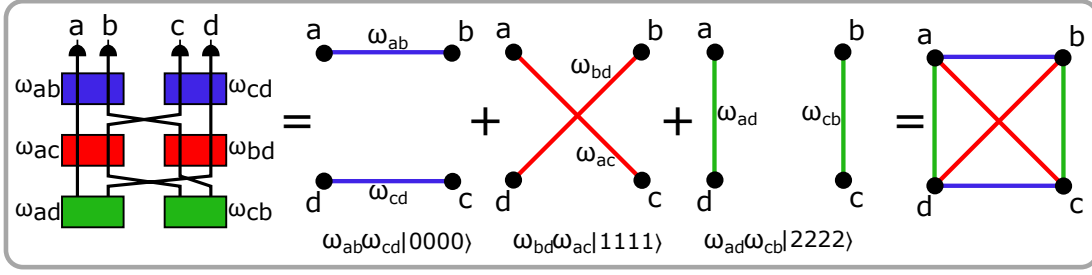


Figure 4: Transformation of the 3-dimension GHZ-state setup from Ref. [102] into a colored weighted graph. As in Fig. 3 color encodes photon mode. The three emission patterns from Fig. 3 correspond to three subgraphs combining into the full graph of the state. These subgraphs are the graph's perfect matchings, as every vertex is connected to exactly one other vertex. The GHZ-state results when all weights are chosen equal. Graph from Ref. [110]

The first term corresponds to the SPDC process, creating two photons from a pump photon, the second term is the reverse, sum frequency generation, absorbing two photons to create a pump photon. The coupling constant η is pump power dependent and describes the efficiency of the process. The pump field is treated classically here, its influence absorbed in the coupling constant [115]. As introduced in the previous section, networks of these sources in path alignment can create interesting photonic states through Entanglement by Path Identity [102]. The graph representation turns such networks of sources into networks of vertices. It achieves this abstraction by representing each path by a vertex and each source emitting into two paths as a colored edge connecting the corresponding vertices [110]. The color encodes the photon mode. Each edge is associated with a complex weight ω encoding amplitude and phase of the respective emission process [111]. Figure 4 depicts this transformation for the GHZ-state creation setup from Figure 3. The general output state of a graph G is then obtained as the output state of the corresponding network of sources by Taylor expanding the product of time evolution operators of all crystals j ; $\hat{U}_j = e^{-\frac{i}{\hbar} H_j t}$ [11]:

$$|\Psi\rangle = \Phi(G) |\text{vac}\rangle \approx \sum_n \frac{1}{n!} \left(\sum_{e \in G} \omega_e \hat{a}_{v_0, m_0}^\dagger \hat{a}_{v_1, m_1}^\dagger - h.c. \right)^n |\text{vac}\rangle, \quad (3.10)$$

where $h.c.$ is the hermitian conjugate and an edge e (source) is specified by the vertex (path) labels v_j , and mode labels m_j for $j \in 0, 1$. The complex amplitude of the process is encoded as the edge weight $\omega_e = g e^{i\Phi_e}$ with $g = \eta t$. Beyond single-pair emission, this weight function contains the creation of multiple pairs from each source and the re-absorption of photons through sum frequency generation. SPDC sources are probabilistic and are pumped with low power from single pair emission, hence $g \ll 1$. This has the effect that the more photons are required for a process, the lower the probability of the process occurring. However, small g also suppresses unwanted high-order terms from the expansion.

As shown in Figures 3 and 4, the emission events contributing to the output state of a network of sources are post-selected from this sum of all possible

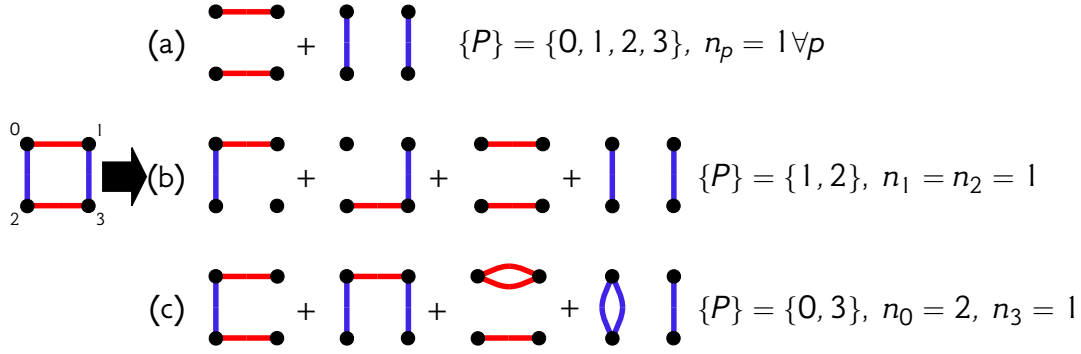


Figure 5: Different ways to interpret a graph-state based on selection criteria for an example graph. All edge weights are set to 1 for simplicity. **(a)** 4-fold coincidence detection criterion, leading to the contributing terms being the perfect-matchings of the graph. See equation (3.11a) for the output state. **(b)** 2-fold coincidence detection on paths 1 and 2. See equation (3.11b) for the output state. **(c)** Selecting for 2 photons in path 0 and 1 photon in path 3. Sources emitting twice are indicated with double edges. See equation (3.11c) for the output state.

events. N-fold coincidence, used in this example, i.e. one photon in every path, is the standard way of interpreting graphs, however, with equation (3.10) more general schemes are possible. For example, photon-number resolving detectors allow the selection of events on specific numbers of photons in certain paths. Only selecting on a subset of detectors corresponds to heralding states in the unconditioned paths [11]. Thus the general conditions for a selection scheme are specific photon numbers in specified paths. Conditioning a subset of paths $\{\mathcal{P}\}$ on a specific photon number $n_{p \in \{\mathcal{P}\}}$ selects the corresponding terms from equation (3.10). In the graph encoding, this equates to the coherent superposition of all subgraphs, where n_p connections can be reached by any edge counting method, simultaneously for all vertices p . This means edges can be counted arbitrarily often to meet the photon number criterion to account for the simultaneous emission and absorption of pairs from the same source. Depending on the criterion, the state represented by the graph is different. Figure 5 shows this mechanism for a few criteria on a simple four-vertex graph where all edge weights are set to 1. The graph displayed is related to GHZ-state creation [110]. The output states corresponding to the different criteria are given by selecting the appropriate terms from the series in equation (3.10). Contributions from orders higher than the minimal contributing order are neglected for simplicity. Those contributions are at least two orders higher for each example and therefore are suppressed for realistic $|\omega| \ll 1$:

cond. (a): $\{\mathcal{P}\} = \{0, 1, 2, 3\}, n_p = 1 \forall p \Rightarrow$

$$|\psi_a\rangle = \left(\hat{a}_{0,1}^\dagger \hat{a}_{1,1}^\dagger \hat{a}_{2,1}^\dagger \hat{a}_{3,1}^\dagger + \hat{a}_{1,0}^\dagger \hat{a}_{1,0}^\dagger \hat{a}_{2,0}^\dagger \hat{a}_{3,0}^\dagger \right) |\text{vac}\rangle; \quad (3.11a)$$

cond. (b): $\{P\} = \{1, 2\}$, $n_1 = n_2 = 1 \Rightarrow$

$$|\psi_b\rangle = \left(\hat{a}_{0,1}^\dagger \hat{a}_{0,0}^\dagger \hat{a}_{1,1}^\dagger \hat{a}_{2,0}^\dagger + \hat{a}_{1,0}^\dagger \hat{a}_{2,1}^\dagger \hat{a}_{3,0}^\dagger \hat{a}_{3,1}^\dagger + \hat{a}_{0,1}^\dagger \hat{a}_{1,1}^\dagger \hat{a}_{2,1}^\dagger \hat{a}_{3,1}^\dagger + \hat{a}_{1,0}^\dagger \hat{a}_{1,0}^\dagger \hat{a}_{2,0}^\dagger \hat{a}_{3,0}^\dagger \right) |\text{vac}\rangle, \quad (3.11b)$$

cond. (c): $\{P\} = \{0, 3\}$, $n_0 = 2, n_3 = 1 \Rightarrow$

$$|\psi_c\rangle = \left[\hat{a}_{0,1}^\dagger \hat{a}_{0,0}^\dagger \hat{a}_{1,1}^\dagger \hat{a}_{2,0}^\dagger \hat{a}_{2,1}^\dagger \hat{a}_{3,1}^\dagger + \hat{a}_{0,1}^\dagger \hat{a}_{0,0}^\dagger \hat{a}_{1,0}^\dagger \hat{a}_{1,1}^\dagger \hat{a}_{2,0}^\dagger \hat{a}_{3,0}^\dagger + \left(\hat{a}_{0,1}^\dagger \right)^2 \left(\hat{a}_{1,1}^\dagger \right)^2 \hat{a}_{2,1}^\dagger \hat{a}_{3,1}^\dagger + \left(\hat{a}_{0,0}^\dagger \right)^2 \left(\hat{a}_{2,0}^\dagger \right)^2 \hat{a}_{1,0}^\dagger \hat{a}_{3,0}^\dagger \right] |\text{vac}\rangle. \quad (3.11c)$$

Hence, to correctly interpret a graph, one requires both the graph AND an additional selection criterion specifying which terms to select from the infinite combinations of all possible emission events. N-fold coincidence detection selects a special class of subgraphs, called *perfect matchings*, subgraphs where every vertex is reached by exactly one edge [110]. Selecting it yields the non-normalized 4-particle 2-dimensional GHZ-state: $|0000\rangle + |1111\rangle$ from equation (3.11a) for the graph from Figure 5(a). See Figure 4 for more perfect matching examples.

Summarizing this to this point, the graph representation encodes quantum states by directly linking them to emission patterns of pair sources in coherent superposition. This is also the most straightforward way of building the setup from a graph via Entanglement by Path Identity. Every edge represents a pair source, every vertex a photon path, and sources whose edges share connection to the same vertex need to be perfectly aligned on the corresponding path. For coincidence detection, a detector is placed at the end of each path. However, the graph encoding can also represent linear optical devices like polarising and non-polarising beam splitters, phase and wave plates, or mode filters [111]. Such devices change the phase or mode of photons they interact with or the paths they can end up in. The graph represents the output state of the setup after all modifications are performed on the emitted photons.

Hence, any phase changes, mode flips or path modifications are included by construction. A graph itself is agnostic to the specific combination of optical elements used in creation but guarantees the existence of a setup generating the state. Figure 6 shows two setups for the GHZ-state creation graph already depicted in Figure 5, one based on Path Identity

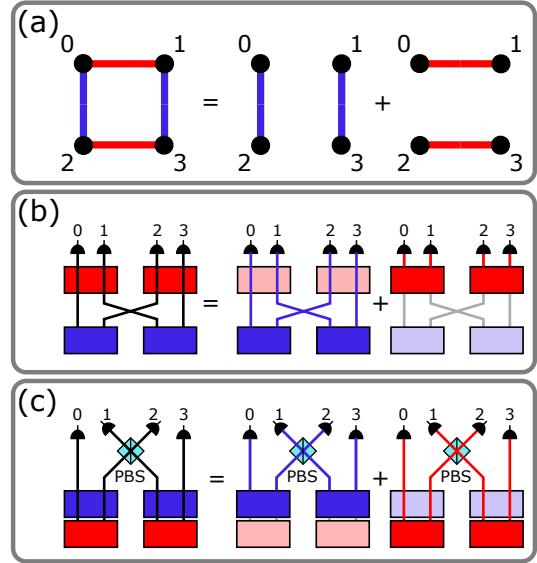


Figure 6: Different ways to build a setup from a graph. **(a)** 4-Particle 2-dimensional GHZ-state graph. **(b)** Setup based on Path Identity. **(c)** Setup based on bulk optics using polarizing beam splitters (PBS). Setups from Ref. [12] Fig.1

(Fig. 6(b)) and one based on bulk optics (Fig. 6(c)). This fact is crucial as it allows the graph to represent many different setups achieving the same output state, making it an efficient representation of search space of quantum optical setups for optimization [11]. Note that the setup does not change when choosing a different post-selection scheme, the graph represents the most general state the setup produces through equation (3.10). [11, 12, 110, 111, 112]

Graphs and Beam Splitters For this thesis, one important feature of the graph representation, is, how the representation of the state changes under the insertion of a beam-splitter in two paths. To reiterate, the graph describing the output state after the beam splitter also describes alternative ways of producing the output. However, this graph transformation is used in later chapters to relate different graphs to each other. In quantum optics, beam splitter relations are described in the form of creation operators, hence, a beam splitter with input ports a, b and output ports c, d is given by [90, p. 199]:

$$\begin{pmatrix} \hat{c}^\dagger \\ \hat{d}^\dagger \end{pmatrix} = \hat{B} \begin{pmatrix} \hat{a}^\dagger \\ \hat{b}^\dagger \end{pmatrix} \quad (3.12)$$

Where \hat{B} is the unitary beam-splitter matrix. Hence, the creator in the path a is transformed as:

$$\hat{a}^\dagger \rightarrow B_{00}^{-1} \hat{c}^\dagger + B_{01}^{-1} \hat{d}^\dagger \quad (3.13)$$

and in path b as:

$$\hat{b}^\dagger \rightarrow B_{10}^{-1} \hat{c}^\dagger + B_{11}^{-1} \hat{d}^\dagger. \quad (3.14)$$

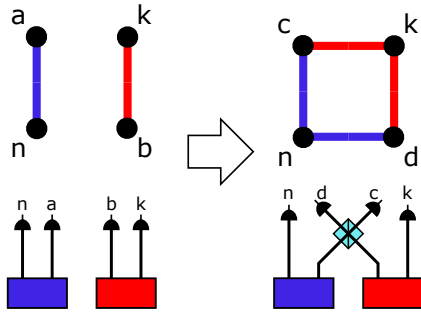


Figure 7: Graph transformation under insertion of a beam splitter

This means, that all photons in path a now can be in path c or d, with different weights dependent on the \hat{B} -coefficients. The same holds for path b. Hence, in the graph picture, all edges previously connected to vertices a or b are now connected to BOTH output paths c and d, with appropriately adjusted weights. The beam splitter adds edges to the graph as shown in Figure 7. The weights of the new edges are transformed by multiplying the old incoming edge weights with the beam splitter coefficients:

$$\omega_e \hat{a}_{a,m_0}^\dagger a_{n,m_1}^\dagger \rightarrow \omega_e B_{00}^* \hat{a}_{c,m_0}^\dagger a_{n,m_1}^\dagger + \omega_e B_{10}^* \hat{a}_{d,m_0}^\dagger a_{n,m_1}^\dagger \quad (3.15a)$$

$$\omega_e \hat{a}_{b,m_0}^\dagger a_{k,m_1}^\dagger \rightarrow \omega_e B_{01}^* \hat{a}_{c,m_0}^\dagger a_{k,m_1}^\dagger + \omega_e B_{11}^* \hat{a}_{d,m_0}^\dagger a_{k,m_1}^\dagger \quad (3.15b)$$

In later chapters equations (3.15) are used for the specific case of a beam splitter with $B_{00} = B_{11} = B_{10} = \frac{1}{\sqrt{2}}$ and $B_{01} = -\frac{1}{\sqrt{2}}$. The here-described relation between beam splitters and paths is introduced in Ref. [111].

Encoding other Types of Experiments Not all interesting quantum optics experiments revolve around state generation. For example, photonic quantum gates apply unitary transformations to different input states. Hence, the experiment has an input state and input-dependent output states. Similarly, state analyzer setups have an unknown input state and need to confirm or deny the existence of the measurement state. With appropriate modifications, the graph representation can describe both of these cases. A concept for encoding incoming states is required. This is achieved by declaring a set of vertices as incoming nodes, i.e. the beginnings of paths where the input state will be supplied to the measurement setup or gate. Edges connecting to these vertices are reinterpreted as a photon being supplied into the path encoded by the second vertex of the edge. Via the same mechanism, input nodes can also be used to describe single photon sources. Edges between input vertices are forbidden. The design task for a specific measurement or gate then uses the close relation between state generation and state propagation, known under the *Klyshko Advanced Wave Picture*[116] or the *Choi-Jamiołkowski*[117] isomorphism. In the graph representation, this isomorphism is realized by the fact, that a stage generation graph using single-photon sources generating the state $|\psi\rangle$ also describes a measurement setup distinguishing the the state $|\psi\rangle$ via its projector $|\psi\rangle\langle\psi|$. [11, 12].

Graph Endcoding: It is convenient to choose different vertex geometries to distinguish between vertex types thus indicating how a graph is to be interpreted at first glance. This encoding in table 1 is in accordance with the conventions of Ref. [12] and then extended to 3D visualizations. Single-photon sources and output vertices are omitted as they are not part of graphs shown in later sections. Many experiments require more photons to realize the desired state than are contained in said state. The additional photons are classified as ancillae and their paths have their own vertex shape in the graph representation.




Vertex	2D encoding	3D encoding
detector vertex	circle	sphere
ancilla vertex	square	cube
input vertex	triangle	tetrahedron
Edge	encoding	
	photon pair emission process into the vertex paths	
	photon traveling from an input vertex to a detector	
	edge with negative weight	

Table 1: Graph encoding of experiments in 2D and 3D. Negative weights in 3D are encoded using polyhedrons as indicators replacing the diamonds used in 2D drawings.

3.3 PyTheus

PyTheus [12] is an algorithm/library for designing quantum optics experiments. It is an expansion and improvement of an earlier algorithm, Theseus [11], built on top of the graph representation of quantum optics described in the previous section. PyTheus can discover state-generation experiments, measurement devices, photonic quantum gates, and communication protocols. It achieves this with a

combination of gradient descent and topological optimization in the form of edge pruning. PyTheus is the AI responsible for the artificial discoveries discussed in this work. The following description is a summary from the PyTheus article Ref. [12].

PyTheus Workflow The first step in discovering a new experiment with PyTheus is to define the target and available resources. The target includes the type of experiment, e.g. state generation as well as the type of optimization. PyTheus can optimize for a specific state or for finding a state that has certain properties, like maximizing a specific entanglement measure. The available resources specify the number of detectors, their type, the dimension of the photons in the paths leading to the detectors, how many ancillary photons are involved, available single photon sources, environment vertices, and the number of input and output paths. For state generation, an important part of the loss function is the fidelity, measuring the overlap between the current graph state $|\psi(G)\rangle$ against the target state $|\psi_T\rangle$, $F = |\langle\psi_T|\psi(G)\rangle|^2$. This is often combined with the count rate of the experiment, the expected rate of positive outcomes. Single-photon sources and input nodes implicitly impose topological constraints on the graph, by forbidding connections between them. Additional restrictions can be added explicitly by choice, e.g. if one wants to define two locally separated sets of detectors, such as in the case of entanglement swapping. To this end, edges connecting detectors to be separated are forbidden. Thus it is ensured, that the resulting experiment can produce the state in those two vertices without the photons having interacted or sharing a source. This instruction set defines the initial graph, the largest possible graph the instructions allow, representing all possible experiments achievable with this set. Together with the post-selection conditions, this defines the search space for the particular query.

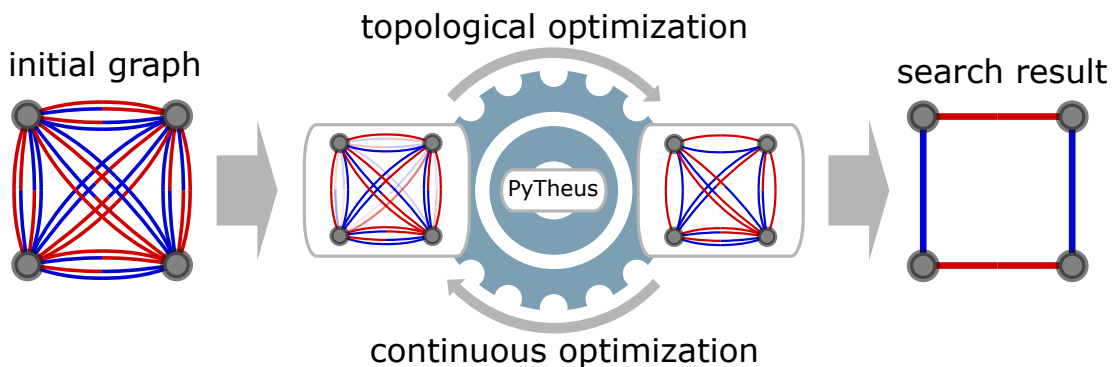


Figure 8: *PyTheus optimization loop. After the largest possible initial graph is constructed from the instruction set PyTheus alternates between optimizing graph weights in continuous optimisation and removing edges in topological optimisation. When no edge can be removed anymore the search terminates. Figure adapted from Ref. [15]*

The PyTheus optimization loop starting from the initial graph is conceptually visualized in Figure 8. The first step of the search is to optimize the conditioned weight function via gradient descent on all edge weights. In the second step, an unnecessary edge is removed from the graph. These two steps are iterated

over until no edge can be removed anymore without significantly increasing the loss. Due to random weight initialization, the resulting graph is not necessarily the sparsest possible graph. Hence, a single PyTheus query performs multiple searches to attempt to find a graph as sparse as possible representing the desired experiment. [12].

Discovery with PyTheus The principal application of PyTheus is the discovery of new experiments. However, the scientific discovery process does not stop at this step. Under the right circumstances, the experiments generated by PyTheus can lead researchers to develop generalizations or even new experimental concepts. As introduced in section 1.1 this is the main way PyTheus contributes to true scientific understanding. The graph representation assists this greatly as it abstracts experimental technicalities into simple graph elements, which through their geometric nature can be more intuitively understood [112, 11, 12]. In the past, using the graph representation has led to several generalizations like generalizing W -states to any even photon count [112] or generalizing Dicke states to higher particle numbers or dimension [12]. One more involved concept resulting from a PyTheus discovery is the *Hyperedge Assembly by Linear Optics* (Halo) concept [14]. This concept plays a role in the interpretations of the results of this thesis, which is why it is explained in further detail here.

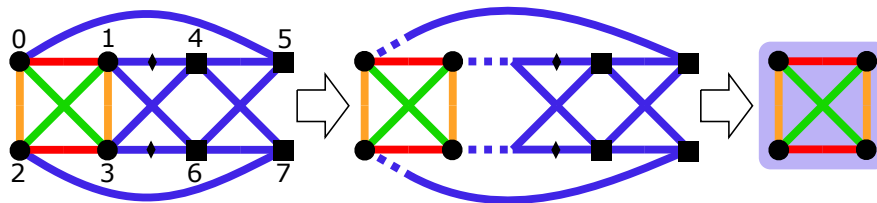


Figure 9: Graph generating $|\text{GHZ}\rangle_4^4$, edge weight signs are not explicitly drawn. Vertices 0-3 from the creation graph for the state $|\text{GHZ}\rangle_4^3$, vertices 4-7 are the ancillae required for the fourth dimension. They can be abstracted to generate a single Hyperedge connecting the four vertices 0-3. Graph adapted from Ref. [14].

In short: "a Halo [...] is a subsystem of a linear optics setup, which effectively acts as a probabilistic multi-photon source." [14]. It was discovered by searching for a four-particle-four dimensional GHZ state using four ancillae, i.e the state:

$$|\text{GHZ}\rangle_4^4 |0000\rangle = \frac{1}{\sqrt{4}} (|0000\rangle + |1111\rangle + |2222\rangle + |3333\rangle) |0000\rangle \quad (3.16)$$

The resulting graph, displayed in Figure 9, consists of two parts. One structure is the well-known [110] graph for creating a 4-particle 3-dimensional GHZ-state $|\text{GHZ}\rangle_4^3$. The second structure is constructed out of the four ancillae vertices. It effectively mimics a source for a correlated four-photon state $|0000\rangle$ emitting in the four paths encoded by the vertices 0-3. Hence, the ancillae structure can be abstracted to represent a *Hyperedge*, and edge connecting more than two vertices. As shown in reference [14], multiple such hyperedges can be combined to add further dimensions to the GHZ-state in equation 3.16. Thus, the Halo sub-graph represents an independent structure that can be reapplied to different setups

turning it into a generalizable building block to achieve correlated multi-photon emission in quantum optics. In the same work, Halos are identified and used for generalization in other graphs, like high-dimensional entanglement swapping, and high-dimensional CNOT-gates.

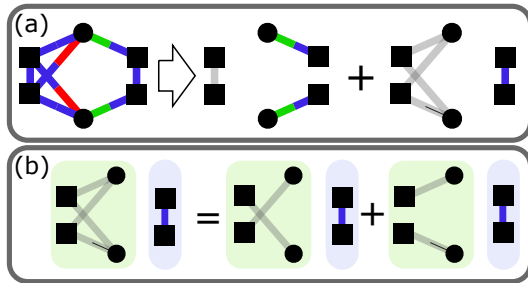


Figure 10: *Visualisation of perfect matchings present in Halo graphs. (a) Viewing the mode 2 (green) Halo, its state is either by external (first term) or internal edge covers (second term). (b) In cases where distinct subgraphs are present (here marked green and blue) the perfect matchings of the entire graph can be abbreviated.*

type of matching is called the internal matching of a Halo This corresponds to the second term in Figure 10(a). The single edge between the Halo ancillae completes the matchings of the base graph as shown in Figure 10(b). The second type describes matchings where the mimicked source is emitting, i.e. all vertices of the base graph the Halo is attached to receive their photon from the Halo. This corresponds to the first term in Figure 10(a) where the $|22\rangle$ state is supplied to the central vertex pair. Perfect matchings inconsistent with one of these categories must interfere destructively or be excluded by the selection criteria. In essence, a subgraph only is a Halo when in all non-interfering perfect matchings of the graph, the Halo either does not contribute to the base graph or provides photons to all paths it is connected to.

Due to their binary function, Halo graphs can also be used to illustrate a way how to abbreviate perfect matchings in cases where they consist of matchings of individual subgraphs. There, the entire set of matchings can be constructed from tensor products of the sums of matchings of the subgraphs. This is illustrated with a minimal example in Figure 10(b), where the two perfect matchings of the greyed subgraph are abbreviated. This method is used in section 5.3 for graphs consisting with excessive numbers of perfect matchings.

Examples like Halo demonstrate the power of the graph representation in deriving concepts from AI-discovered experiments. It represents the discovery of a generalizable, multipurpose concept for quantum optics experiments through Inspiration from AI results. However, the discovery also demonstrates the importance

of spotting and rationalizing patterns in graph structure to achieve this conceptualization. Hence, the step to a more intuitive and interactive visualization, like VR is a naturally occurring pursuit. The tool developed for this purpose is presented in the next section.

4 AriadneVR

This section introduces the VR tool, AriadneVR, developed as part of this thesis work. AriadneVR² is a web-based Virtual Reality visualization and analysis tool for the graph representation of quantum optics experiments described in section 3.2. It is built using A-Frame³, an open-source, freely available, HTML and Javascript framework made for developing virtual reality web applications. A-Frame is based on the Three.js⁴ open-source 3D rendering library. Within A-Frame a variety of open-source components exist, both officially- and community-maintained, allowing for low threshold access to VR development. As a web-based application, the tool requires no installation on the user device, but local- or web-hosting of a website, e.g. via GitHub pages⁵. AriadneVR was developed on a Meta (formerly Oculus) Quest 2, but can in principle be easily adapted to any VR device with sufficient controller functionality by adapting the input button mappings on the code level. A-Frame itself has implemented support for inputs from a variety of VR devices.

Performing graph analysis in VR is not a new approach and tools exist geared toward displaying large networks representing complex databases such as VRnetzer [73]. Such tools are purposes built for graph data analysis on a large scale, thus requiring tethers to dedicated PCs to run. As PyTheus graphs are comparatively small, this is not required for AriadneVR. The WebVR approach makes AriadneVR very lightweight in setup, deployment, and development thresholds. Hence, it is relatively easy for a non-expert in VR development, to start developing their tool within the webVR framework. These aspects were the main reason for choosing webVR and A-frame to develop a custom tool for PyTheus-generated graphs.

AriadneVR has been published as part of the preprint related to this work Ref. [15] and is publicly available on the Artificial-Scientist-Lab GitHub⁶. A more technical introduction to the software structure is given in Appendix B.

4.1 Setup and Workflow:

As mentioned AriadneVR needs to be hosted and served to the VR device via a network connection. It then runs fully locally as a Javascript application in the browser of the device. Information about graphs to be analyzed is stored in JSON format on the host platform. All such available graphs can then be visualized and analyzed from within the Virtual environment. To add graphs, they need to be added to the host platform. As PyTheus output files do not contain any positional information for vertices, a preprocessing step of finding an initial layout is required when adding PyTheus-generated graphs. A layout is defined using the spring-model-based Kamada-Kawai (KK) sorting algorithm [118] pre-implemented in the igraph [119] library. This algorithm has been chosen as it produces acceptable initial sortings for many PyTheus graphs. Pre-processing is done before the graphs are added to the host serving the site.

²The tool is named after Ariadne, the wife of the mythical hero Theseus, whom PyTheus is named after. She assists him in finding out the Labyrinth of the Minotaur with her famous thread.

³<https://aframe.io/>

⁴<https://threejs.org/>

⁵<https://pages.github.com/>

⁶<https://github.com/artificial-scientist-lab/AriadneVR>

AriadneVR does not have a built-in implementation of a graph sorting algorithm. During the analysis, the user can produce output files, e.g. to save certain graph configurations. These are downloaded as JSON files to the user's device for storage. To reinspect, they need to be again added to the host platform. In the case of hosting via GitLab pages, this process condenses to pushing the new files to the hosting remote. The resulting workflow loop is visualized in Figure 11.

AriadneVR is intended to be used stationary sitting at a desk for convenience, but standing or moving use is possible in a suitable environment.

4.2 Graph-encoded Experiment Analysis in Virtual Reality

AriadneVR is an experimental prototype application for the analysis of graph-encoded quantum optics experiments. Hence, its feature set grew dynamically from the basic visualization of graphs in 3D towards more interactivity based on the experiences obtained from using it. Here, the final set of features as well as the reasoning for their existence is provided to end up with a working VR-assisted analysis workflow as depicted in Figure 12. It summarizes as such; First, the user attempts to discover some form of structure in the graph, The goal is to achieve an interpretable arrangement, allowing to link the graph to conceptual ideas, either about the graph representation or the underlying physics. To test their understanding and hypotheses the user can perform on-the-fly modifications to the graph, to find generalizations of their concepts. If successful, this leads to the manual design of new experiments in graph form. If not, a user can use AriadneVR to define search geometries for PyTheus searches, to obtain more examples for study in the quest for generalizable understanding. The feature set enabling this workflow is outlined in the following sections.

4.2.1 Visualisation Features;

Encoded experiments have relatively small graphs, consisting mostly of single or low double-digit vertex numbers (see. Ref. [12] for 100 examples). As previously mentioned, the initial graph layout is chosen via the KK graph sorting algorithm [118]. The choice of algorithm is not critical, as the layout is not static. The small size of PyTheus graphs makes it feasible and effective to manually sort a graph into any desired structure. To this end, the user can grab and move vertices with the VR controllers to change the arrangement to their needs. This sorting process has shown itself to provide an intuitive engagement method for

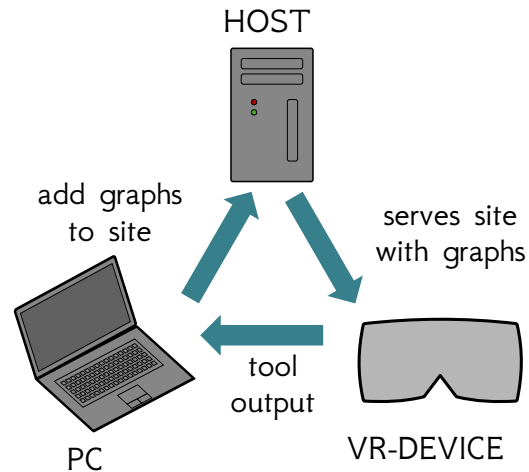


Figure 11: Workflow visualization of AriadneVR. Graph files preprocessed on a standard PC are added to the host, which serves the website to the VR device. Any output files generated by the tool are then extracted from the VR device to the PC, and can then subsequently added to the site via the host.

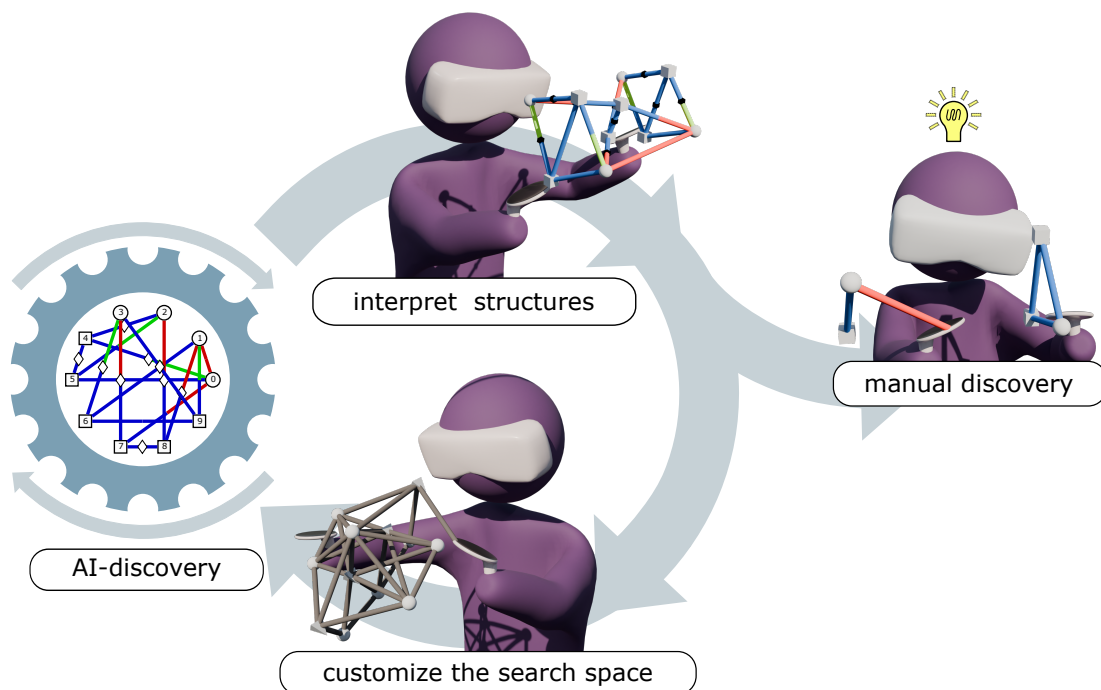


Figure 12: *VR-workflow, an AI-discovery is analyzed to find interpretable structures. This can directly lead to generalization through manual discovery. However, if this is not possible, the obtained structure can be utilized to define template geometries for further AI searches. Figure taken from the preprint related to this work Ref. [15], with permission of the authors.*

learning structural information about an unknown graph⁷. Hence, it is chosen as the main way of discovering structure over algorithmic approaches. This is entirely dependent on the relatively small size of the studied graphs. Such a manual approach will not be time-effective for large structures of hundreds of vertices. The entire graph structure can also be moved and rotated to be examined from all sides by gestures alone. The chosen scale of the graphs is roughly equivalent to objects easily maneuvered in one or two hands. Hence, generally speaking, the user will always see and interact with the entire graph. Again, this is possible and effective due to the small size of the networks.

The current state represented by the graph is tracked on a spawnable data slate in the form of all its perfect matchings (see Fig. 13(b)). If applicable, this slate shows the target state of the original optimization. Other post-selection methods are not currently implemented. The slate also allows spawning of the graph's perfect matchings as additional subgraphs via virtual buttons. These subgraphs can be moved and rotated, however their vertices are not interactable. They instead trace relative positional changes of the vertices in the main graph.

Thus, the visualization features of AriadneVR intend to allow the user to engage with graph structure to find optimal geometries and derive interpretations with no restriction on curiosity and creativity imposed by a fixed algorithmically

⁷This assessment is subjective based on my experience and has not been investigated quantitatively.

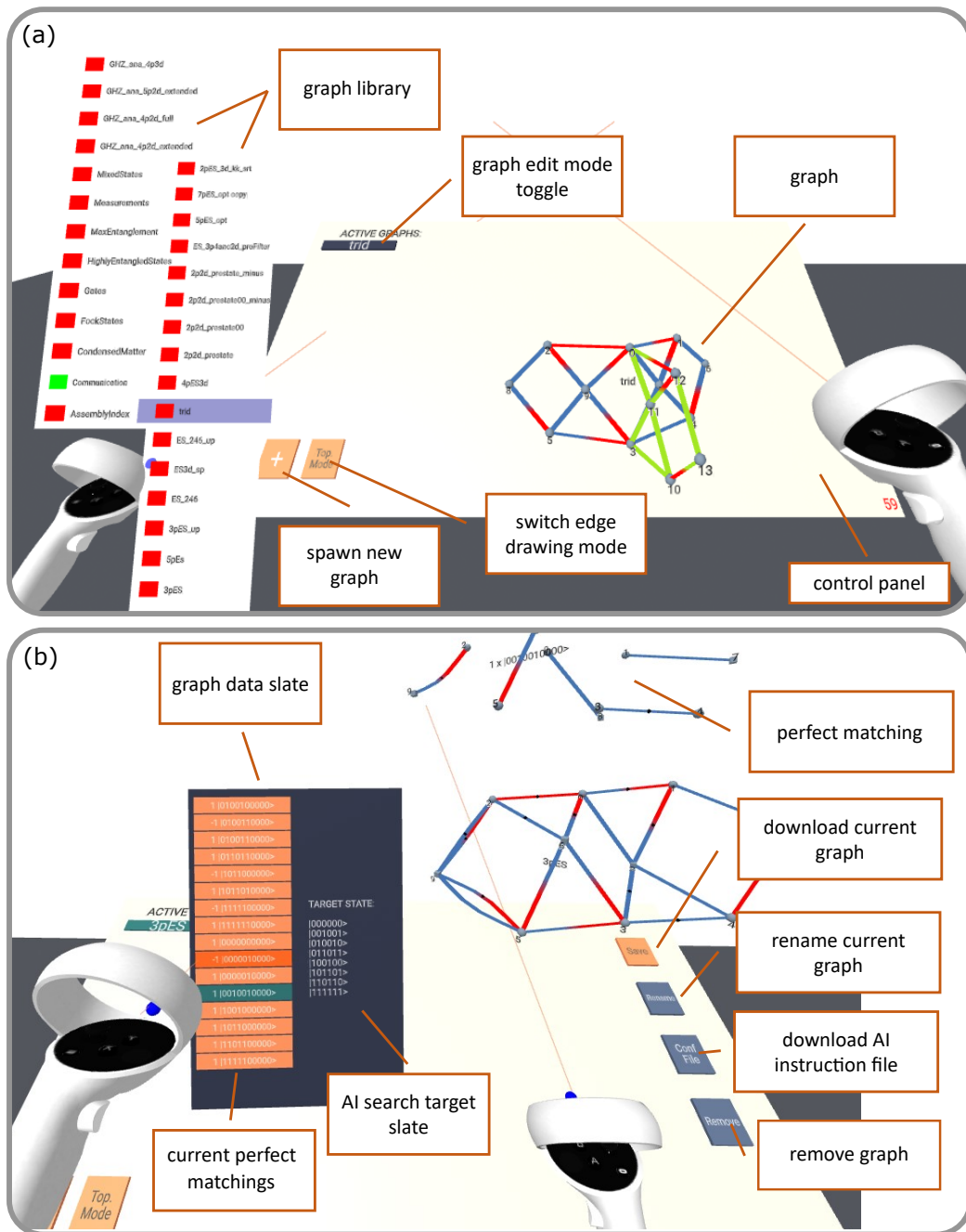


Figure 13: AriadneVR user interface (UI). **(a)** Base UI including the controller attached graph library and one spawned graph. For every spawned graph a toggle spawns enabling editing. The control panel includes two basic buttons for spawning a new one-vertex graph and switching the edit mode to colorless edges **(b)** UI with a graph active under investigation. On the right of the control panel, buttons appear for graph saving, renaming or deleting, and the creation of PyTheus instruction files based on the current geometry. The movable data slate lists all current perfect matchings with pressable switches, controlling the spawning of perfect-matching subgraphs, as well as the original target state of the graph. For better quality, these screenshots are taken from an emulator on a PC not from within a Headset.

defined structure.

4.2.2 Exploration Features:

Beyond visualization, the tool allows functional modification of graphs. Via controller interaction, the user can spawn and delete vertices, as well as add and remove edges to or from the graph. Thereby the underlying experiment can be freely modified. As an example, the process of adding an edge to a graph is demonstrated in Figure 14.

These changes are live-tracked on the aforementioned data slate showing the current state of the graph in its perfect matchings. Through this interactivity, a user can explore the effects of changes to the graph structure. On top of modifying existing graphs, the user can build up completely new ones if desired. Thus the user can use intuitive drawing type interactions to test their hypothesis about graph mechanism or attempt the design of new experiments utilizing their obtained experience.

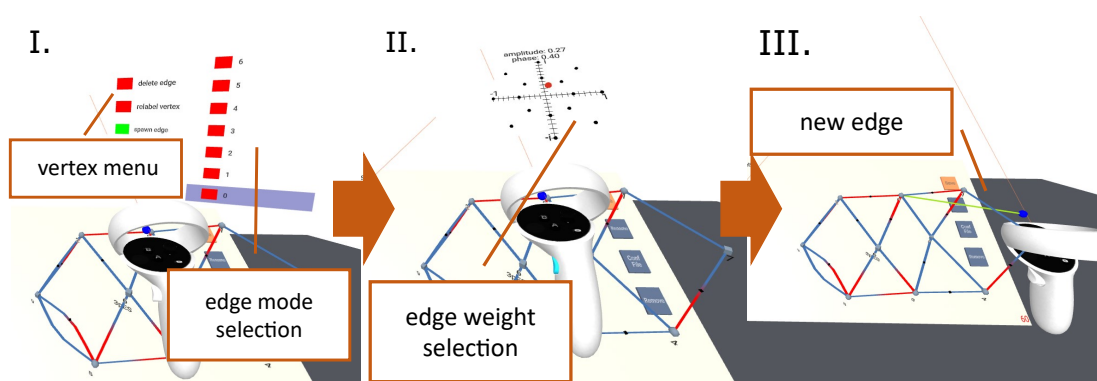


Figure 14: Graph editing on the example of edge spawning. **I.** Interaction with a vertex spawns a menu for selecting between adding or removing an edge or relabeling the vertex. Choosing add edge allows color selection. **II.** After color selection a menu appears for setting complex edge weight via a red selector movable on an Euler plane via the controller thumbsticks. **III.** After setting the weight the edge appears connected to the controller until placed on a second vertex finalizing the editing process. For better quality, these screenshots are taken within an emulator on a PC not from within a Headset.

The above-described graph editing is the basis for another feature, defining search geometries in VR. Via a toggle (see Figure 13 (a)), the user can switch the edge drawing behavior to forgo colors and weights to draw black edges. AriadneVR can then produce a template of a PyTheus search instruction file, where the search geometry is restricted to the current graph. Black connections are unspecified and hence represent the full set of possible edges between two vertices in the initial graph. Any colored edges in the template are taken as is and completely define the permitted connectivity between two vertices.

The template is downloaded to the VR device and can then be used to start PyTheus searches. It will require further input in other search parameters, for example, the target state of the search, as the classical keyboard and PC setup is far more suitable for typing than VR controllers. Optimization then happens as

normal on all edges of the created initial graph. With this feature, a user can employ their obtained structural understanding to propose geometries of generalizations and then have PyTheus confirm or reject those hypotheses. Thus, the size of the search space can be reduced using human intuition, making the search for generalizations more efficient. This constitutes a human-in-the-loop workflow, where VR analysis assists the AI task. This process is demonstrated on examples in section 5.2.

With the provided interactivity, AriadneVR matures from a visualization aid to an interactive graph exploration tool, that can serve as an experimental playground for a researcher to study the graph representation of quantum optics.

5 Virtual Reality-augmented Discovery

AriadneVR has been used to inspect and analyze selected graphs produced by PyTheus, most published in reference [12]. Special emphasis was placed on the PyTheus solutions for efficient entanglement swapping, as a generalization mechanism for the 2-dimensional 3-pair solution was discovered early in this work based solely on the graph geometry. This case will be discussed in detail in section 5.3. The first step in potential discovery is to search for candidates for interpretable graph structures, discussed in section 5.1. Such discovered structure is then used in section 5.2 to guide PyTheus towards discovering generalizations using the corresponding feature of AriadneVR.

5.1 Interpretable Structures

As described in section 4, AriadneVR features a variety of interactive features meant to ease graph analysis. A 2D desktop-based tool can be imagined with the same type of graph editing features. One of the key differences between AriadneVR and such an imaginary tool is the stereoscopic rendering provided by the VR headset. This allows for the illusion of viewing graphs as physical 3D objects and by extension provides intuitive access to structure. Previous discoveries made with the graph encoding, such as the extension mechanism of *W*-state-graphs [112] and Dicke-state graphs [12] and the *Hyperedge Assembly by Linear Optics* (Halo)-concept [14] (see. Sec. 3.3) heavily rely on structural properties of the involved graphs, rotational cylindrical symmetry in the case of *W*-states and hyperedge creation via subgraphs in the case of Halo. Hence, a reasonable approach to making discoveries is to search for structures in graphs, and subsequently to try to find useful interpretations of these structures.

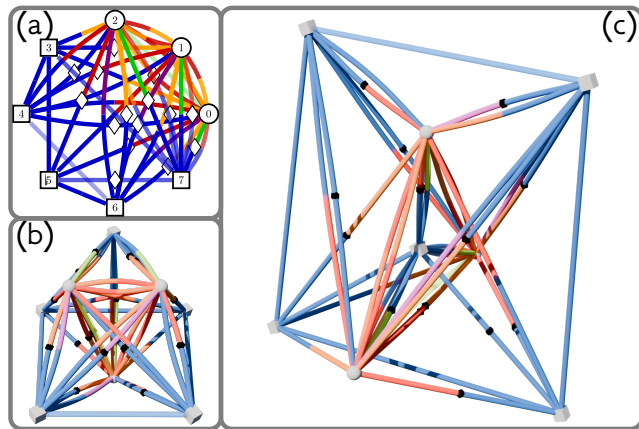
5.1.1 Graphs with candidate structures for interpretability

A large portion of graphs discovered by PyTheus yield visually accessible structure upon basic inspection and manual sorting. In this context *visually accessible* refers to the presence of some order in the graph, be it revealing symmetries or other structural features, like being able to define groupings of vertices based on properties like vertex type or connectivity. It is not a strictly definable term and the quality and appeal of any sorting is subjective to the user. Here, selected graphs showcase the potential for improvement in visual accessibility to graph features provided by AriadneVR. In comparison, the standard PyTheus circular 2D ordering is also shown, together with the 3D ordering produced by the sorting algorithm in preprocessing. This comparison is not made to illustrate the advantages of 3D sortings over 2D sortings, as the standard PyTheus structure is not optimal for most graphs. Rather, it is to show the readability advantage offered by structured sortings in general, be they 2D or 3D. To underline this point the discovered 3D sortings were used to draw optimized 2D representations. AriadneVR offers an intuitive way to discover these arrangements. Vertices are movable with two hands in 3D space, edge crossing can be easily eliminated and the structure can be investigated from all sides. Any 2D desktop-based tool would suffer from a lack of 3D perspective and less efficient input methods, and cannot eliminate edge crossings in non-planar graphs. All 3D-graphs are shown as renderings made with

Blender 4.0⁸ for better visuals.

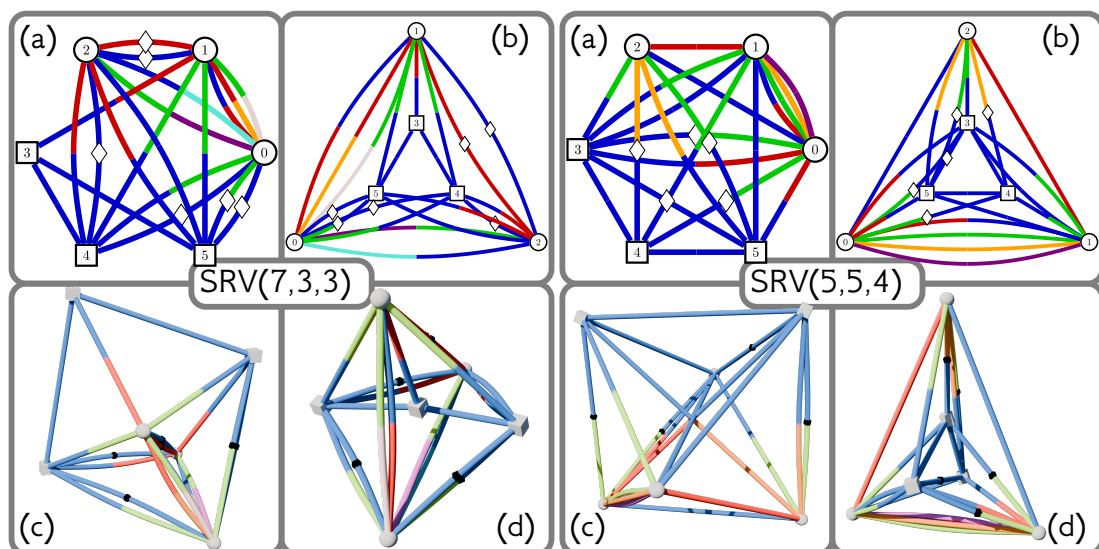
Ground State of the 3-Particle Spin-2 AKLT-Model:

This graph belongs to solid-state physics-related states discoverable by PyTheus [12]. It describes the ground state of the Affleck-Kennedy-Lieb-Tasaki (AKLT) Model for 3 particles of spin 2 [120]. Its graph has a striking structure accessible in 3D. In the chosen arrangement, the vertices 0-3 are sorted on a triangle perpendicular to the ancilla plane. The connectivity of the vertices is highly symmetric with respect to the ancilla and state vertex planes. No similarly accessible 2D sorting could be built from the 3D structure in this case. **(a):** Standard 2D sorting. **(b)** Algorithmic 3D sorting. **(d)** Manual 3D sorting. This graph is not part of the Ref. [12] but was discovered as part of the same work and is available on the PyTheus GitHub graph library [121].

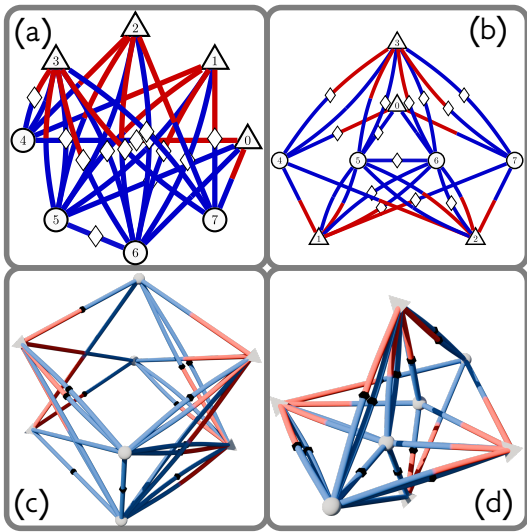


The connectivity of the vertices is highly symmetric with respect to the ancilla and state vertex planes. No similarly accessible 2D sorting could be built from the 3D structure in this case. **(a):** Standard 2D sorting. **(b)** Algorithmic 3D sorting. **(d)** Manual 3D sorting. This graph is not part of the Ref. [12] but was discovered as part of the same work and is available on the PyTheus GitHub graph library [121].

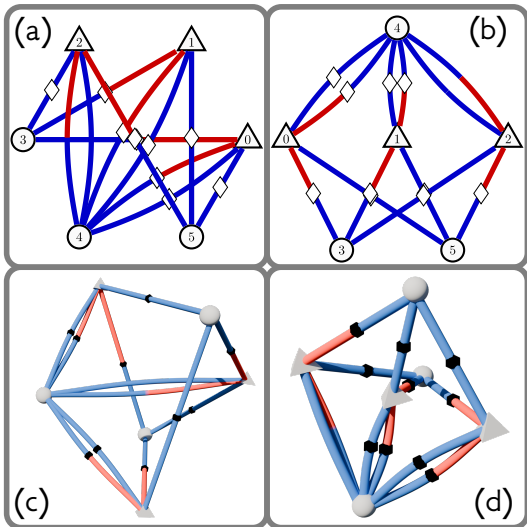
States with specific Schmidt Rank Vector With Pytheus a variety of new 3-particle states with specific SRV were discovered [12], among others for SRV(5,5,4) (graph 26 in Ref. [12]) and SRV(7,3,3) (graph 29 in Ref. [12]). Both are depicted below. Both graphs show symmetry in their structures both in connectivity and for specific edges in edge coloring, which is not immediately accessible in both their standard 2D and the automated 3D sorting. **(a):** Standard 2D sorting. **(b)** Manually optimised 2D sorting based on the manual 3D sorting. **(c)** Algorithmic 3D sorting. **(d)** Manual 3D sorting.



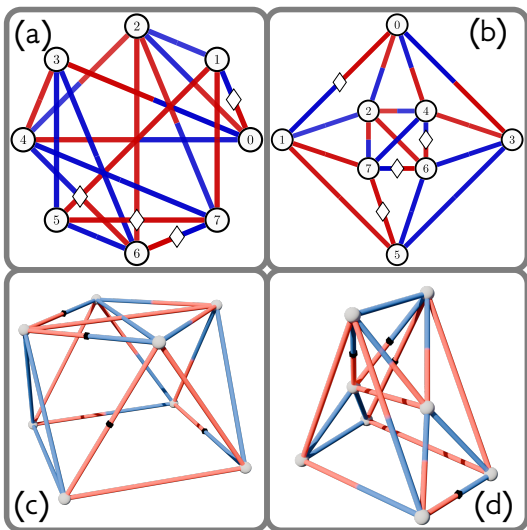
⁸<https://www.blender.org/>



Yeo-Chua State Measurement: This graph depicts a measurement setup of the Yeo-Chua state discovered with PyTheus (graph 84 in Ref. [12]), a highly entangled state used for teleportation of arbitrary two-qubit-states [12, 122]. **(a):** Standard 2D sorting. **(b)** Manually optimised 2D sorting based on the manual 3D sorting. **(c)** Algorithmic 3D sorting. **(d)** Manual 3D sorting. It is an example of revealed symmetry by sorting vertices based on their vertex type, in this case placing detectors on a line and the incoming photon nodes in a square arrangement perpendicular to the line.



2D GHZ State Measurement: The here depicted graph shows a measurement scheme of a 2-dimensional 3-particle GHZ-state. **(a):** Standard 2D sorting. **(b)** Manually optimised 2D sorting based on the manual 3D sorting. **(c)** Algorithmic 3D sorting. **(d)** Manual 3D sorting. It serves as an example of a graph with accessible standard sorting that can still be improved via manual 3D sorting. As the AKLT graph, this graph was discovered by the authors of Ref. [12] and is available on the PyTheus GitHub [121].



Maximizing Entanglement: This graph represents a state PyTheus found when maximizing entanglement for all possible 3 qubit partitions of an 8 qubit system (graph 43 in Ref. [12]). It displays maximal entanglement in 48 of 56 possible partitions [12]. It serves as an example for a graph where manual sorting, despite removing edge crossings, does not yield significant benefit over the sorting already achieved by the Kamada-Kawai-Algorithm. **(a):** Standard 2D sorting. **(b)** Manually optimised 2D sorting based on the algorithmic 3D sorting. **(c)** Algorithmic 3D sorting. **(d)** Manual 3D sorting.

5.1.2 Visualizing Loops of Zero-Contribution

Many graphs exhibit destructive interference to produce the desired output state. This requires two distinct perfect matchings contributing the same term to the output state but with opposite signs. This interference can lead to interesting frustrated multi-photon interference effects as demonstrated by recent experiments [113, 114]. One way this can be constructed is by forming subgraphs containing loops of even edge count, but negative accumulated weight along the loop:

$$\begin{array}{c} \bullet \\ | \\ \bullet \end{array} \begin{array}{c} \bullet \\ | \\ \bullet \end{array} - \begin{array}{c} \bullet \quad \bullet \\ \text{---} \\ \bullet \quad \bullet \end{array} = \begin{array}{c} \bullet \quad \bullet \\ \text{---} \\ \bullet \quad \bullet \end{array} = |\text{vac}\rangle$$

A loop only classifies as such, if both incident edges have the same color at any vertex of the loop, and any vertices not contained have one incident edge each. The former requirement ensures, that the loop subgraph has two perfect matchings representing the same state. The latter is required to guarantee, that the two perfect matchings of the loop generate two perfect matchings of the full graph. Destructive interference occurs, when one perfect matching of the loop has the same weight with the opposite sign of the other, meaning that the total sign of all edges along the loop is negative.

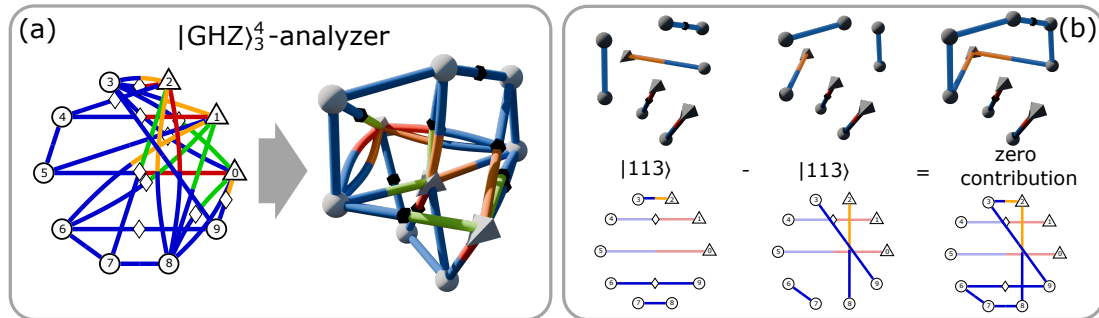


Figure 15: Illustration for loops of zero contribution. **(a)** 2D and 3D graphs for the 4-dimensional GHZ analyzer (graph. 81 in Ref. [12]). **(b)** Example for a loop of zero contribution, drawn both in 3D and 2D. Edges belonging to the loop are highlighted. Figure adapted from Ref. [15].

Such loops are present in many graphs and can be intuitively discovered through visualizing and superimposing the 3D models of two canceling perfect matchings in virtual space. An example of a graph achieving all its destructively interfering terms via loops is the four-dimensional GHZ-state analyzer (graph 81 in Ref. [12]). It distinguishes the $|\text{GHZ}\rangle_3^4$ -state (see. eqn. 2.3) from other incoming 4-dimensional GHZ-states. A 3D model of the graph and one example of a loop is shown in Figure 15.

Without a doubt, such loops are also easily visualized in 2D representations of graphs, either via the same technique of superimposing perfect matchings in 2D or by studying canceling pairs of matchings long enough⁹. However, this feature was independently discovered by chance in VR, while sorting canceling pairs of

⁹In fact one of my supervisors Sören Arlt was aware they existed when I reported the find I made in VR to him.

matchings of a graph creating the $|GHZ\rangle_6^3$ -state with 2 ancillae. It shows, how by only being able to grab and move subgraphs, a new understanding of the graph representation can be obtained by the user, even without looking specifically for it. This is a case, where the 3D nature of VR is less relevant than the interactivity the platform provides. The easy accessibility of loops of zero contribution through VR is one of the results of this thesis published in the preprint related to this work Ref. [15].

5.1.3 Discovery of a 2-dimensional Halo in a 3-particle 5-dimensional GHZ-state graph

Through interactive visualization, a 2-dimensional Halo (see. Sec. 3.3) is found in the known graph for a 3-particle 5-dimensional GHZ-state generation (graph 2 in Ref. [12]). Upon further inspection, it reveals to be closely related to a known Halo graph of the 4-dimensional 4-particle GHZ-state creation [14]. Henceforth, the graphs are labeled with their state labels $|GHZ\rangle_3^5$ and $|GHZ\rangle_4^4$ according to equation (2.3). 2D representations of both graphs are depicted in Figure 16. Both graphs require ancillary detectors and involve eight photons in total.

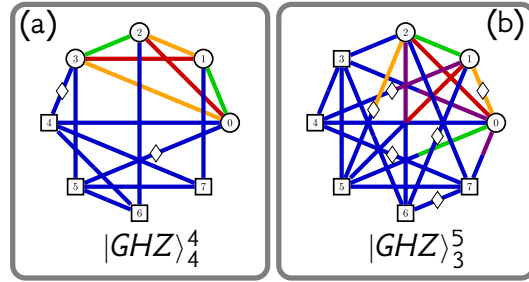


Figure 16: 2D GHZ graphs in a circular layout. (a) $|GHZ\rangle_4^4$ (b) $|GHZ\rangle_3^5$.

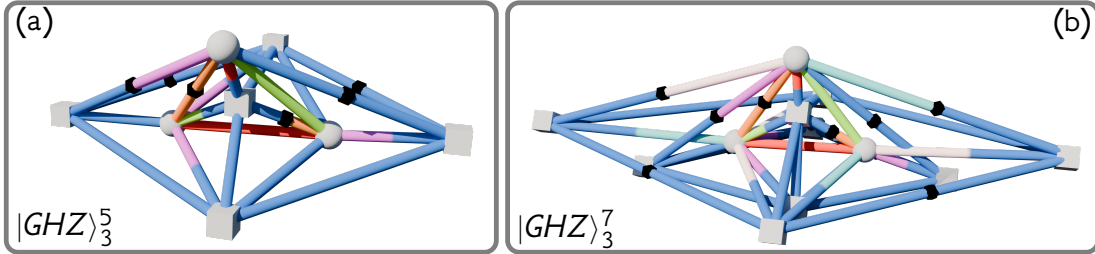


Figure 17: Symmetrized sortings for $|GHZ\rangle_3^5$ and $|GHZ\rangle_3^7$. (a) $|GHZ\rangle_3^5$, with one 2-dimensional Halo and 8 total photons (b) $|GHZ\rangle_3^7$, with two 2-dimensional Halos and 12 total photons. (For exact definition of the graph see App. A.1).

The $|GHZ\rangle_3^5$ contains a two-dimensional Halo, mimicking a source of the state $|0000\rangle + |4440\rangle$. This was discovered from the arrangement shown in Figure 17(a). The outer four ancillae constitute the Halo for central graph of three state vertices and one ancilla. Its Halo nature is confirmed by the construction of a 12-photon graph producing the state $|GHZ\rangle_3^7$ by copying the outer structure of $|GHZ\rangle_3^5$ once more, corresponding to adding two more hyperedges. The result is depicted in Figure 17(b).

By comparing this graph to $|GHZ\rangle_4^4$ it is revealed, that they share the same

core structure, a 4-vertex tetrahedron, with a differing outside vertex structure on top (See. Figure 18(a) I and V). The core geometry is the well-known 4-particle 3-dimensional GHZ state graph [110] (see. Fig. 4) with a slight modification in the case of the $|\text{GHZ}\rangle_3^5$. It reduces the particle count by declaring one corner of the tetrahedron as ancilla by flipping all its modes to mode 0 (blue). This find is also published in the preprint related to this thesis Ref. [15].

Upon further inspection prompted by this similarity, removing the edges corresponding to one dimension of the hyperedge of $|\text{GHZ}\rangle_3^5$ and moving into a similar arrangement reveals that both graphs $|\text{GHZ}\rangle_4^4$ and the reduced Graph $|\text{GHZ}\rangle_3^4$ share the same geometry. Hence, by explorative analysis in VR, it is shown, that $|\text{GHZ}\rangle_3^5$ is not only a Halo graph but can also be derived from $|\text{GHZ}\rangle_4^4$ with a few simple modifications. This process is depicted in Figure 18 both in the abstract hyperedge picture and with appropriate 3D sortings.

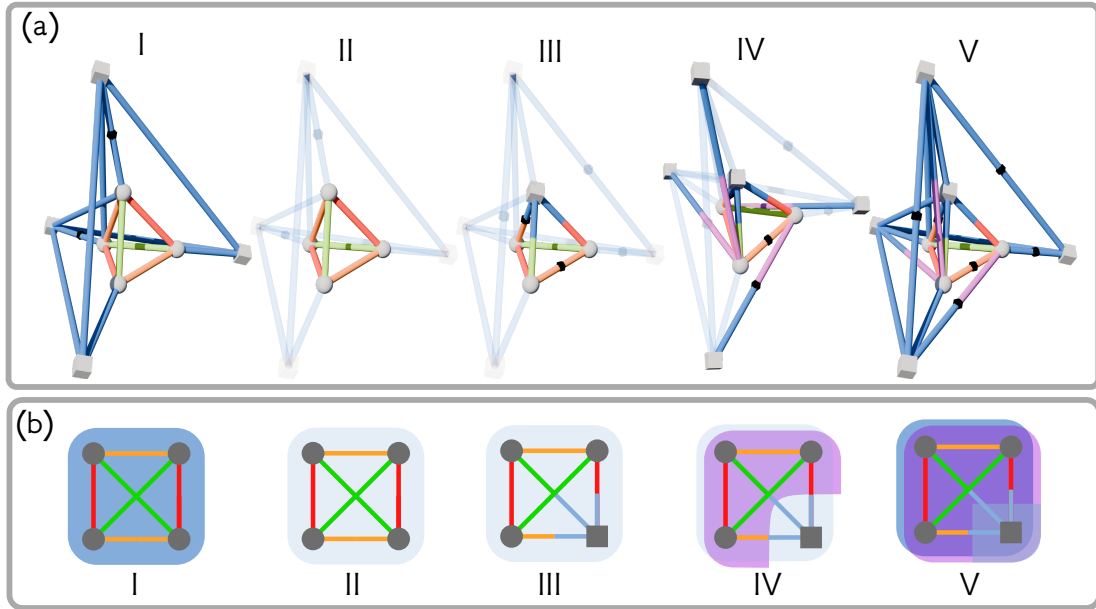


Figure 18: Transformation of the $|\text{GHZ}\rangle_4^4$ -graph to the $|\text{GHZ}\rangle_3^5$ -graph. **(a)** 3D visualisation of steps I-V **(b)** Abstracted hyperedge visualisation of steps I-V. Modifications on edge structures are drawn with reduced opacity for steps II-IV. **I:** Initial $|\text{GHZ}\rangle_4^4$ graph with 4 ancillae providing the 0-mode hyperedge. **II:** Core $|\text{GHZ}\rangle_3^4$ -graph highlighted within the $|\text{GHZ}\rangle_4^4$ graph. **III:** Core $|\text{GHZ}\rangle_3^4$ -graph highlighted within a $|\text{GHZ}\rangle_3^4$ -graph resulting from a mode flip to 0 of all incident modes on one vertex of graph II. **IV:** Insertion of 4 0-4 mode edges into graph III between the core and outside structure to construct the second dimension of the hyperedge providing mode 4. **V:** Final $|\text{GHZ}\rangle_3^5$ -graph with two 4 vertex hyperedges of mode coloring 0-0-0-0 and 4-4-4-0 utilizing the same ancillae detectors and connections for the last shared 0-mode photon creation processes. Differences in edge weight signs between the steps are due to freedom during optimization and are inconsequential to the final output of the graph.

5.1.4 Conclusions on Finding Interpretable Structures

The section above focuses on the discovery of structure in PyTheus graphs. Although this can be done automatically with graph sorting algorithms with acceptable results in some cases, VR allows the user to rely on natural perception skills to find ideal structures easily. Whether these structures are interpretable or not, depends on the background knowledge of the user. Here, two cases are presented where structural features can be interpreted. Loops of zero contribution are a general feature of the graph representation, whereas the relation between the two GHZ graphs is more specific. It is of note, that multiple useful sortings for a single graph can exist, depending on the feature they highlight. An example is the $|GHZ\rangle_3^5$ state. Its first sorting (see. Fig. 17(a)) nicely reveals a symmetric two-dimensional Halo structure. However, this sorting somewhat obscures its full relation to the $|GHZ\rangle_4^4$ graph, highlighted better in the arrangement of Fig. 18. The similarities of the core structure are clear in both, but the relation between the Halos is not obvious from 17. This arrangement in turn, hides the symmetry of the 2-dimensional Halo. More relations like this one are likely present in already discovered PyTheus graphs, only waiting to be discovered. As demonstrated here, AriadneVR could be used as an efficient tool to scan for these. In conclusion, AriadneVR has shown itself to be very helpful in finding structures in graphs useful for domain experts with background knowledge about the underlying physics. However, interacting with the structure can also lead a user to a general insights of the graph representation, independent of the specific represented physics and background, as in the case of loops of zero contribution.

5.2 Steering PyTheus by Geometry Based Constriction of the Search Space

In some cases, graph geometry immediately hints at possible generalization patterns. Example cases for this are the graphs of Dicke-states [12], W-states [112] both with or without single photon sources, and multipair entanglement swapping [12] (discussed in detail in section 5.3).

All of these graphs have shown themselves to be scalable to high photon counts based on simple graph construction rules based on patterns observed in the initial examples.

As demonstrated in section 5.1 many graphs can exhibit patterns or symmetries. However, especially for more complex graphs, it can still be a difficult task to manually generalize to higher dimensions and photon counts due to the large combinatorial space of possibilities involved. Every edge added on top of a pre-existing edge adds another full set of perfect matchings, i.e. terms contributing to the output state. In d -dimensional graphs, every connection added can be one or a set of d^2 edge possibilities without taking edge weight into account. With these simple considerations, it is easy to see how quickly the possibilities for graph extension can become incomprehensible, even if one is restricted by a hypothesis for an extension pattern. This is especially troubling in cases where there are not multiple initial scaling examples available to derive a pattern from. The search for such examples can be made more efficient by applying constraints to a graph. This is a routine part of PyTheus optimization to encode wanted features of the target state or graph type [12].

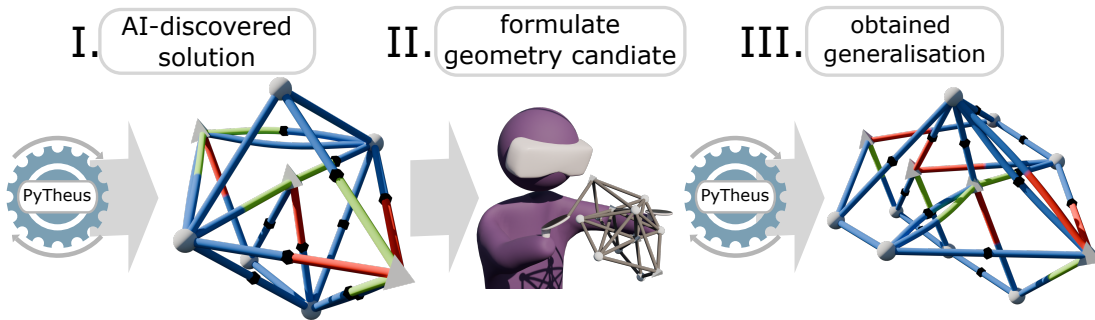


Figure 19: *Workflow of steering PyTheus searches based on geometries extrapolated from pre-existing PyTheus solutions. I. First, Pytheus finds a solution for an experimental task. II. Second, AriadneVR is used to discover a graph structure that is then used to define a search geometry for a generalization. III. Third, Pytheus optimizes this restricted geometry to discover a generalization. The depicted graphs are from the GHZ analyzer discussed in section 5.2.2. Figure adapted from the preprint related to this work Ref. [15].*

For example, as mentioned previously, in measurement or gate graphs, where input and/or output vertices exist, no connections are allowed between different input/output vertices. However, constraints can be motivated by other considerations. One possibility is proposing the geometry for a graph extension based on a previous example. Incidentally, this is how the first extension of the 3-pair entanglement swapping graph (see section 5.3) from three to five pairs was discovered. As described in section 4 AriadneVR has inbuilt support for formulating this type of geometry hypothesis by allowing the user to draw edges with undefined colors, and automatically creating constrained PyTheus instruction file templates. As an interactive VR platform, it is well suited for discovering possible geometries in graphs, as demonstrated in section 5.1, and defining their extension through intuitive drawing. This approach combines the strengths of PyTheus, efficiently searching a combinatorial space for a solution, and the strengths of human perception, recognizing patterns in complex structures, to more efficiently search for new solutions based on existing ones, constituting a case of a human-in-the-loop application. VR here acts as a platform serving the data to the human in a format well accessible to them. The thusly created workflow is visualized in Figure 19. However, it also has two obvious drawbacks, firstly, it is only justified while computing cost is a relevant factor in optimization, secondly, the human-defined search geometries can be too restrictive, excluding potentially interesting results from the search space, even if solutions are still found within.

5.2.1 2-dim. GHZ-state analyzers

As mentioned above, the method of searching on geometries constrained by human intuition or ideas predates the existence of the corresponding feature of AriadneVR. The first case where AriadneVR was actively used for defining search spaces is the discovery of GHZ-state analyzers in two dimensions based on the graph showcased in section 5.1.1. This case serves as a proof of concept as there are already schemes for extrapolating particle numbers of two-dimensional

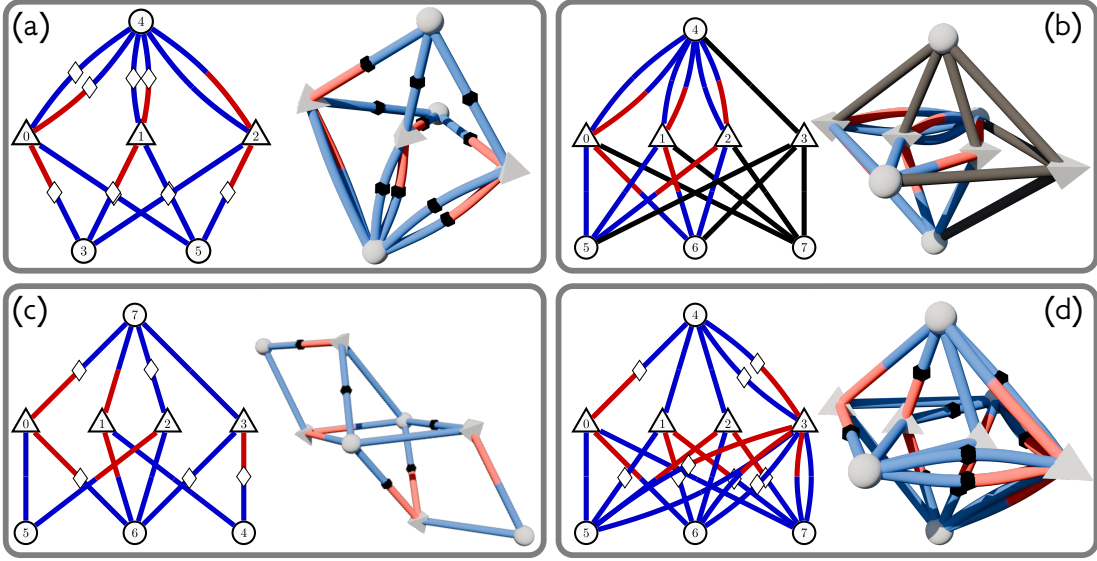


Figure 20: Graph Geometry-based search for particle number extension of a two-dimensional GHZ-state analyzer. **(a)** PyTheus discovered base graph in 2D and 3D. **(b)** Generalised search geometry for a four-particle GHZ-analyzer in 2D and 3D. Black edges represent all possible edges present in the initial graph for the search. **(c)** Smallest graph resulting from the search featuring only twelve edges, however, not mapping on the search geometry. **(d)** 20 edge graph search result mapping perfectly on the proposed geometry candidate. The search geometry and results are available as edge lists in App. A.2.

GHZ-state analyzers to arbitrary size [123, 124]. Such GHZ-states are defined for any particle number n as follows (also see. equation (2.3)):

$$|GHZ\rangle_n^2 = \frac{1}{\sqrt{2}} \left(\bigotimes_1^n |0\rangle + \bigotimes_1^n |1\rangle \right) \quad (5.1)$$

The state normalization is omitted in the graphs presented here to allow for edge weights of ± 1 .

The base graph for this search is an analyzer for the simplest GHZ-state (see equation (5.1) for $n=3$), and is depicted in Figure 20(a). It shows a clear pattern of having every detector connected to every input node and no ancillary connections between the detectors. Vertex 4 features double edges.

A logical extension to increase the particle count of this analyzer is to add one input node and one detector while keeping the connectivity pattern observed in the base graph, as depicted in Figure 20(b). Searching on this topology is successful and leads to a variety of graphs with perfect fidelity, meaning they can perfectly distinguish the four-particle GHZ-state versus other incoming states. Two of those results are of note. Firstly, the sparsest graph found by the search containing of twelve edges. It is depicted in Figure 20(c), a highly symmetric solution that does not coincide with the intended geometry extension. However, the search also revealed another solution, shown in Figure 20(d). This graph features the

connectivity of the assumed generalization. The exception is the placement of the double edges. In this result, one input node (vertex 3) has double edges to all detectors instead of one detector to all input nodes. It confirms that the proposed geometry is a correct extension of 2-dimensional analyzers, even if it does not represent the sparsest possible one.

5.2.2 3-dim. GHZ-state analyzers

This section describes the method on a more complex example, the three-particle three-dimensional GHZ-state analyzer depicted in Figure 21(a) (graph. 80 from Ref. [12]). Even though it is highly symmetric, it has a more complex geometry than the two-dimensional analyzer and requires five detectors, meaning on top of the three incoming photons two ancillary photons are created within the analyzer to allow full distinction of the desired GHZ state (see also equation (2.3)):

$$|GHZ\rangle_3^3 = |000\rangle + |111\rangle + |222\rangle. \quad (5.2)$$

Contrary to the two-dimensional analyzer search described in section 5.2.1, a four-particle geometry extension is not as clear. This is in part because a larger analyzer likely requires additional ancillae whose connectivity to the already existing detectors is not clear. Hence, instead of simply expanding the current graph with some additional nodes and edges, a completely new template is drawn featuring four input nodes and eight detectors, resulting in four ancillary photons.

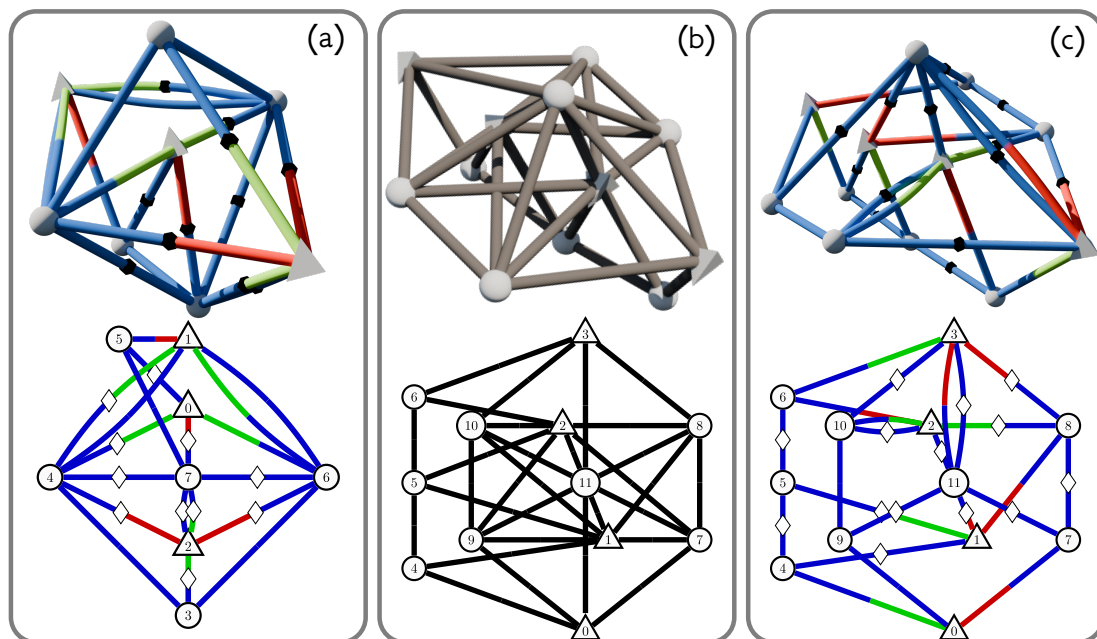


Figure 21: Graph Geometry-based search for particle number extension of a three-dimensional three-particle GHZ-state analyzer. **(a)** PyTheus discovered base graph in 2D and 3D. **(b)** Generalised search geometry for a 4 particle GHZ-analyzer in 2D and 3D. **(c)** Successful search result of a four-particle three-dimensional GHZ-state analyzer, based on the search geometry, however displaying sparser connectivity than the geometry. The search geometry and results are available as edge lists in App. A.3

The template is shown in Figure 21(b) and follows the general geometry of the three-particle analyzer in Figure 21(a), but is relatively loosely restricted on some detectors. It also makes no assumptions on the coloring of any edge.

This approach hence does not constrain the search space nearly as much as the approach of the two-dimensional analyzer, where most edges of the search geometry have defined colors. However, it still reduces the edge count of the initial graph by 50 compared to an unconstrained search from 124 down to 74 edges. This graph is then optimized for the four-particle extension of the state in equation (5.2) $|GHZ\rangle_4^3$. The search is successful and results in the graph shown in Figure 21(c). Unfortunately, this result is not nearly as symmetric as the base three-particle graph of the search. This is presumably due to the loose restrictions on some of the nodes in the template, not enforcing symmetries. This is evident in the fact, that the search turned up a variety of solutions featuring slight variations in connectivity patterns. A set of more restrictive geometries has been tried but led to no search success. The result of this search does not easily hint at generalizations to more particles. Hence, deriving a scalable three-dimensional GHZ-state analyzer would require more study and further searches. These graphs are significantly harder to project on 2D than the two-dimensional analyzers and turn out much less legible, as seen in Figure 21. Here, the three-dimensional environment assists substantially in extracting and defining geometries.

5.2.3 Conclusions on Steered PyTheus Discoveries

In this section structure discovered in VR is used to generalize tasks using PyTheus. It is achieved by restricting the search space based on geometric intuition by defining initial graphs. Although this has merit on its own, as it makes the search more efficient, the ultimate goal is to obtain the necessary examples to be able to derive schemes for generalization that do not require the AI search anymore. For the 3D analyzer, this is an unresolved challenge using the obtained result. Potentially a general extension pattern could be derived from the discovered solutions, but this has not been achieved within the scope of this thesis. Presumably, further study of the 3-particle analyzer is needed to design a better search geometry for search results that form better examples for extension patterns. However, as many graphs exhibit structure, as demonstrated in the previous section, it can be conjectured, that this method has a lot of potential to find generalizations of complex graphs. This can be done before the required understanding for manual extension is obtained, which could then be made easier through more available examples. This section is showing that it is possible to discover new results based on intelligently defined search spaces. AriadneVR plays a crucial part in this process, both in finding and rationalizing structure in the first place, and then defining the search geometry.

5.3 Resource Efficient Multipair Entanglement Swapping

The main work of this thesis was performed in analyzing and constructing graphs of entanglement swapping experiments. An introduction to the topic of entanglement swapping is given in section 2.2. The relevant graphs non-locally create multiple maximally entangled photon pairs. Hence, they are generally referred to as *multipair* entanglement swapping. Pair photons are not allowed to share

photon sources, to ensure non-local creation. This is fundamentally different from *multiparticle* entanglement swapping [32], a process where a single entangled multiphoton state is obtained, briefly touched upon in section 2.2. This allows more than two parties to share an entangled state, contrary to multipair entanglement swapping where two parties share many entangled states. This analysis was not exclusively performed in Virtual Reality but describes a case augmenting the process with the tool. During analysis, AriadneVR was repeatedly used for visualization and to confirm or reject hypotheses about the underlying mechanisms of graphs by testing them with quick graph edits. For simplicity the weights of all graphs in this section are 1 or -1, leading to states that are not normalized. This includes the Bell states mentioned.

5.3.1 Three Pair Entanglement Swapping

The basis for this analysis is a graph discovered by PyTheus and published in reference [12] as graph 77. The total state of the graph is:

$$\begin{aligned} & |\Phi^+\rangle_{a_0,b_0} |\Phi^+\rangle_{a_1,b_1} |\Phi^+\rangle_{a_2,b_2} \\ & \otimes |0000\rangle_{c_0,c_1,c_2,c_3} \end{aligned} \quad (5.3)$$

Here, the subscript labels are chosen to align with its depiction in Figure 22. The Vertices can be grouped into three parties, named by popular convention Alice, Charlie, and Bob, labeled with $a_0 - a_2$, $b_0 - b_2$, and $c_0 - c_3$ respectively. In the graph layout of Figure 22, horizontally aligned vertices belong to the same party, top row being Alice, mid row Charlie, and bottom row Bob. Vertically aligned photons form a non-locally generated Bell-pair (a_j, b_j) . As there are no edges between Alice's vertices and Bob's vertices, none of Alice's photons originate in the same sources as Bob's. The four photons Charlie detects are classified as ancillae. Hence, this graph non-locally creates three entangled photon pairs shared between Alice and Bob in the Bell state $|\Phi^+\rangle$ while requiring the measurement of four additional photons by Charlie.

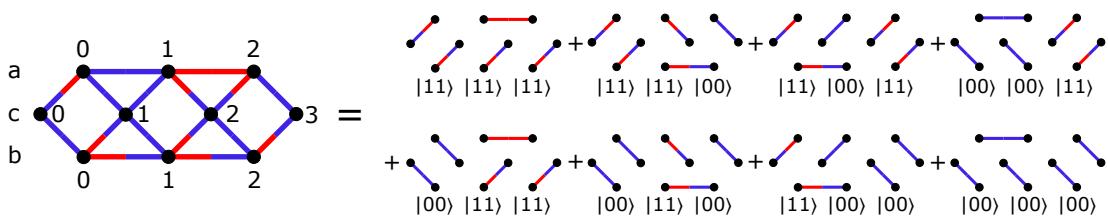


Figure 22: Graph 77 from reference [12] with its eight perfect matchings. Detectors are split into three parties, Alice, Charlie, and Bob, and labeled $a_0 - a_2$ for Alice, $c_0 - c_3$ for Charlie, and $b_0 - b_2$ for Bob. Post-selected on coincidence in all detectors every pair a_j, b_j is in the state $|\Phi^+\rangle$. Ancilla states are omitted from the ket-labels of the perfect matchings.

For regular entanglement swapping the number of ancillary photons per pair is two (see. Sec. 2.2), meaning for three pairs six ancillary photons are required. This increased efficiency is the reason this graph is of interest. As already touched upon in section 3.2, resource efficiency in terms of total photon number is crucial for the scalability of these experiments due to the stochastic nature of the involved

photon emission processes. Essentially, the more crystals have to emit at the same time for an experiment to succeed the less likely it is to happen in a specified time frame. Beyond this practical advantage, there is also the fundamental question about the origins of this increased efficiency. The central questions are firstly, how this increased efficiency is realized, i.e. how is the non-local correlation brought into existence without an independent messenger particle per pair photon, and secondly, if there are fundamental differences beyond the increased efficiency between this version and the standard version of entanglement swapping.

5.3.2 Generalization to Arbitrary Odd Pairs

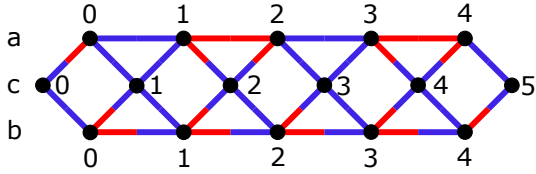


Figure 23: Five pair entanglement swapping. Detectors are split into three parties, Alice, Charlie, and Bob, and labeled $a_0 - a_4$ for Alice, $c_0 - c_5$ for Charlie, and $b_0 - b_4$ for Bob. Post-selected on coincidence in all detectors every pair a_j, b_j is in the state $|\Phi^+\rangle$.

Automatic sorting using the KK-algorithm [118] produced the chain-like structure in the graph displayed in Figure 22. The authors of Ref. [12] then used this knowledge to search for five-pair entanglement swapping based on an extension of the chain, very similar to the method showcased in the previous section. This search has proven successful and revealed the graph shown in Figure 23. This graph creates the state:

$$\begin{aligned} & |\Phi^+\rangle_{a_0, b_0} |\Phi^+\rangle_{a_1, b_1} \\ & |\Phi^+\rangle_{a_2, b_2} |\Phi^+\rangle_{a_3, b_3} |\Phi^+\rangle_{a_4, b_4} \quad (5.4) \\ & \otimes |000000\rangle_{c_0, c_1, c_2, c_3, c_4, c_5} \end{aligned}$$

It non-locally generates five Bell pairs requiring six additional photons. With this second example, the experimental scheme can be generalized to arbitrary odd-pair numbers by observing and extending the pattern in the graph. This pattern consists of three subgraphs, depicted in Figure 24. These subgraphs are now analyzed in detail to understand how they contribute to the total output state.

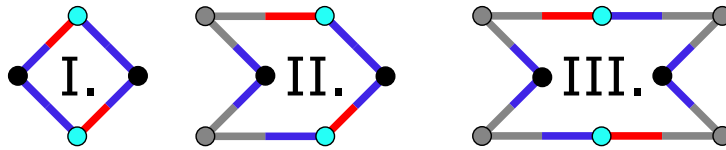


Figure 24: The three subgraphs contributing to the multipair entanglement swapping graphs. Ancillas are black, Bell pair photons cyan, and additional photons grey. All grey edges can be colored arbitrarily, but all incident edges at the grey vertex need to share the same color. The three graphs all create one non-local Bell-pair at the cyan vertices. **I.** 1-pair swapping (graph 76 in Ref. [12]). **II.** 1-pair swapping with two additional photons. **III.** 1-pair swapping with four additional photons

Subgraph I: Single Pair Entanglement Swapping The first of these subgraphs (see Fig. 24 I.) is actually a known one-pair entanglement swapping graph (graph

76 in Ref. [12]). It generates a single Bell pair using two ancillae. It appears at every even-numbered pair of the three and five-pair solutions.

$$\Rightarrow |\Phi^+\rangle_{a_0,b_0} |00\rangle_{c_0,c_1} \quad (5.5)$$

In Ref. [12] this graph is presented as entanglement swapping without requiring initial Bell states due to it only requiring sources of correlated pairs. However, the graph can also be understood to represent a slightly modified version of the standard entanglement swapping experiment using two initial Bell states (see. Sec. 2.2). It can be reached by applying filters and half-wave plates after the beam splitter in the standard setup. This transformation is depicted in Figure 25. PyTheus finds the transformed graph instead of the original one due to it firstly assuming ancillae to always be in mode 0, and secondly due to the smaller size of the filtered graph. This is a remarkable example of how efficient the graph representation is in encoding vastly different setups in the same graph. One must note, that the transformation in Figure 25 is not unique, and constructing an entanglement swapping experiment in this manner is not advisable as the filters reduce the count rate of coincidence events by a factor of two. Regardless this interpretation is chosen here for its close relation to the original entanglement swapping experiments. As shown in Figure 25 this interpretation also allows for the definition of pre-swap states, which is not possible in the Path Identity version. This is due to the fact, that the Path Identity version does not require a Bell-state measurement to non-locally entangle the pair photons. It creates the entanglement after all sources through the path alignment.

The derivation of the transformation is as follows. The initial state is chosen as (See. Fig. 25 I.):

$$|\Phi^+\rangle_{b_0,c_0} |\Psi^-\rangle_{a_0,c_1} \quad (5.6)$$

To avoid complex weights a dielectric beam splitter is inserted:

$$\hat{c}_0^\dagger \rightarrow \hat{c}_0^\dagger + \hat{c}_1^\dagger \quad (5.7a)$$

$$\hat{c}_1^\dagger \rightarrow \hat{c}_1^\dagger - \hat{c}_0^\dagger \quad (5.7b)$$

Normalization has been omitted to keep all edge weights to ± 1 . Thus the

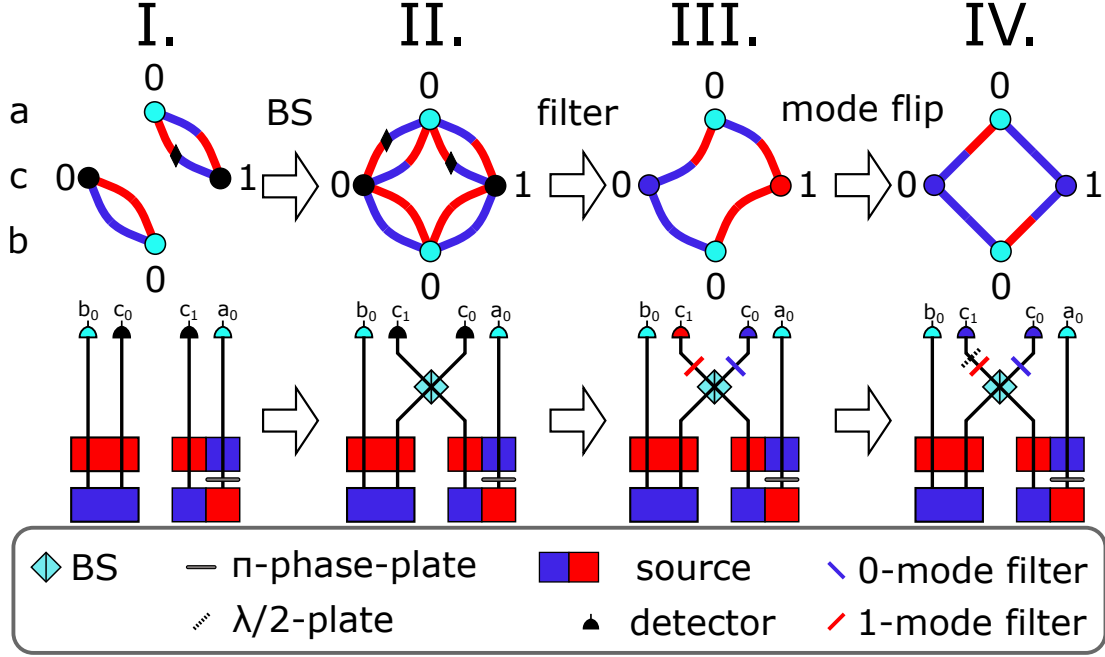


Figure 25: Graph and Setup transformation from standard entanglement swapping to subgraph I. **I.** Pre-swap graph, see eq. (5.6). Cyan vertices represent the pair to be swapped. **II.** Insertion of a beam splitter (BS) on the ancilla paths with subsequent coincidence measurement performs the swap to the state $|\Phi^+\rangle_{a_0, b_0} |\Psi^-_{c_0, c_1}\rangle$. **III.** Insertion of a 0-mode filter after the beam-splitter deletes excess edges in the graph, projecting on 1 term of the ancilla Bell state. **IV.** Finally, flipping the mode of the c_1 ancilla via a $\lambda/2$ -plate to reach the first subgraph of the multipair solutions. See eq. (5.5).

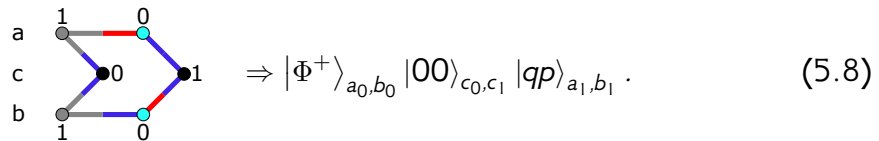
state evolves to:

$$\begin{aligned}
& \left[\hat{b}_{0,0}^\dagger (\hat{c}_{0,0}^\dagger + \hat{c}_{1,0}^\dagger) + \hat{b}_{0,1}^\dagger (\hat{c}_{0,1}^\dagger + \hat{c}_{1,1}^\dagger) \right] \\
& \otimes \left[\hat{a}_{0,0}^\dagger (\hat{c}_{1,1}^\dagger - \hat{c}_{0,1}^\dagger) - \hat{a}_{0,1}^\dagger (\hat{c}_{1,0}^\dagger - \hat{c}_{0,0}^\dagger) \right] |\text{vac}\rangle \\
& = \left[\hat{b}_{0,0}^\dagger \hat{a}_{0,0}^\dagger (\hat{c}_{0,0}^\dagger \hat{c}_{1,1}^\dagger - \hat{c}_{0,0}^\dagger \hat{c}_{0,1}^\dagger + \hat{c}_{1,0}^\dagger \hat{c}_{1,1}^\dagger - \hat{c}_{0,1}^\dagger \hat{c}_{1,0}^\dagger) \right. \\
& \quad - \hat{b}_{0,1}^\dagger \hat{a}_{0,0}^\dagger \left((\hat{c}_{0,1}^\dagger)^2 - (\hat{c}_{1,1}^\dagger)^2 \right) \\
& \quad + \hat{b}_{0,0}^\dagger \hat{a}_{0,1}^\dagger \left((\hat{c}_{0,0}^\dagger)^2 - (\hat{c}_{1,0}^\dagger)^2 \right) \\
& \quad \left. - \hat{b}_{0,1}^\dagger \hat{a}_{0,1}^\dagger (\hat{c}_{0,1}^\dagger \hat{c}_{1,0}^\dagger - \hat{c}_{0,1}^\dagger \hat{c}_{0,0}^\dagger + \hat{c}_{1,1}^\dagger \hat{c}_{1,0}^\dagger - \hat{c}_{1,1}^\dagger \hat{c}_{0,0}^\dagger) \right] |\text{vac}\rangle \\
& = \left[\hat{b}_{0,0}^\dagger \hat{a}_{0,0}^\dagger (\hat{c}_{0,0}^\dagger \hat{c}_{1,1}^\dagger - \hat{c}_{0,1}^\dagger \hat{c}_{1,0}^\dagger) - \hat{b}_{0,1}^\dagger \hat{a}_{0,1}^\dagger (\hat{c}_{0,1}^\dagger \hat{c}_{1,0}^\dagger - \hat{c}_{1,1}^\dagger \hat{c}_{0,0}^\dagger) \right] |\text{vac}\rangle \\
& = |\Phi^+\rangle_{a_0, b_0} |\Psi^-\rangle_{c_0, c_1}
\end{aligned}$$

For all creation operators, the second index specifies the mode number. Here, the 4-fold coincidence criterion is used to omit terms with two photons in the output paths of the beam splitter. In the graph representation, the beam splitter

copies the connectivities of the input nodes (see. Sec. 3.2) resulting in the graph of Figure 26 II. Filtering the output ports to $|01\rangle$ and subsequently flipping the mode of the photon in path c_1 leads to the output state of equation (5.5) and subgraph I.

Subgraph II: Two Additional Photons The second subgraph (see Fig. 24 II.) appears in two variants at the ends of the chain of the three and five-pair solution. The left-hand closure is a mirrored version of the graph depicted in the figure. It creates a non-local Bell pair with two ancillae plus two additional particles in an arbitrary product state $|qp\rangle$:



$$\Rightarrow |\Phi^+\rangle_{a_0, b_0} |00\rangle_{c_0, c_1} |qp\rangle_{a_1, b_1} \cdot \quad (5.8)$$

Upon inspection of the multipair graphs in Figures 22 and 23 it is evident that the graph shares the edges connecting the Bell pair photons a_0, b_0 to the ancilla c_1 with the first subgraph (see. fig 24 I.). Potential pre-swap states for this solution are more complicated. They can be reverse-engineered from the consideration, that as subgraph I and subgraph II are superimposed in the final graph, there should be a setup where the measurement setup on the ancillae paths is the same. I.e. there should exist a transformation from an initial state to the desired output state that involves the passage of a beam splitter and subsequent filters plus a mode flip. It is likely that other setups with different pre-swap states exist that still allow the superposition of the graphs.

It is possible to find pre-swap states by reversing the transformation that was applied in the derivation of subgraph I. The process from the pre-swap state to the output state is shown in Figure 26. The pre-state for this subgraph is fundamentally different from the one of subgraph I, as it does not have a well-defined particle count. To perform swapping, both Alice's pre-state and Bob's pre-state require one photon of Charlie in their pre-state. Thereby the total number of vertices in each pre-state is three, meaning no perfect matchings exist. As the final graph after the swap has six photons, the most general initial state for one party is the superposition of all two and all four-photon states. This superposition can lead to the final selection of either two photons from Bob's, and four from Alice's pre-state, or vice versa. However, with the knowledge that the final output is post-selected for coincidence in all detectors, this can be significantly simplified to all two and four-photon states that have one photon in the a and b paths and up to two photons in the ancilla paths c . This uses the knowledge of the intended beam splitter transformation on the c -paths during the transformation. The photon number in a and b paths is unchanged by this transformation, and as the beam splitter is lossless, the only way to obtain two output photons is by considering all possible combinations of inputs 0-2, 1-1, and 2-0. With these considerations, the initial state represented by the Graph shown in Figure 26 I. is:

$$\left[|1q\rangle_{a_0, a_1} - |q0\rangle_{a_1, c_1} |\Psi^-\rangle_{a_0, c_1} \right] \otimes \left[|0p\rangle_{b_0, b_1} + |p0\rangle_{b_1, c_0} |\Phi^+\rangle_{b_0, c_0} \right] \quad (5.9)$$

This state is now propagated through the same beam splitter given by equations (5.7). However, only 2+4 or 4+2 particle number combinations from equation

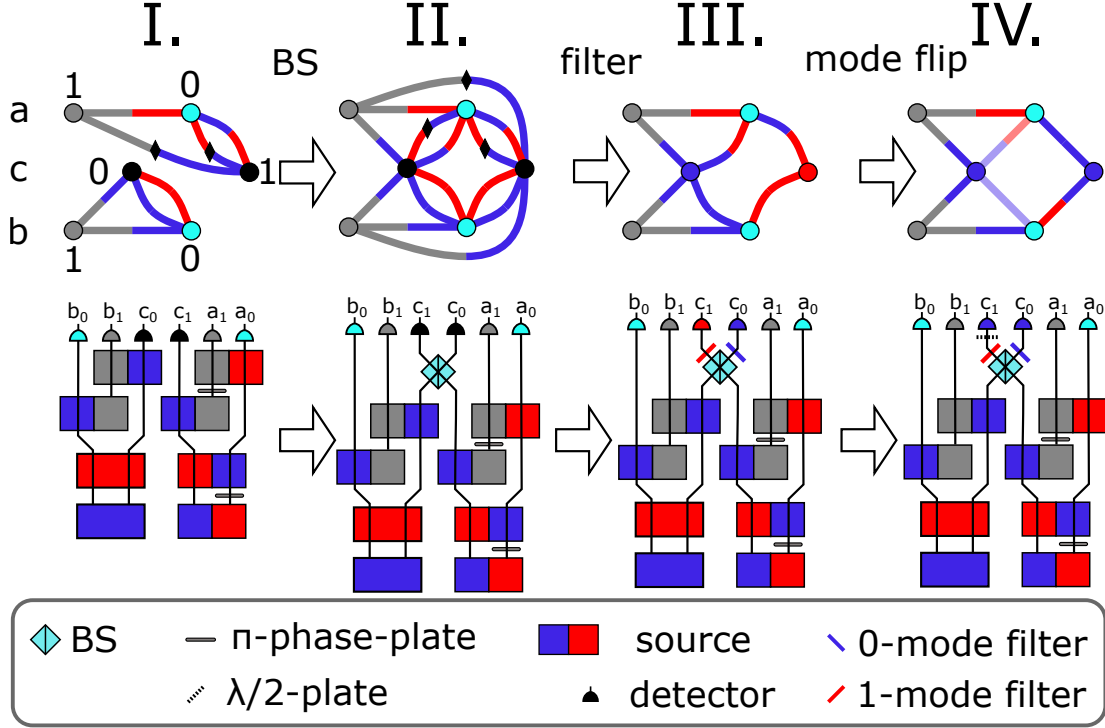


Figure 26: Graph and setup transformation for subgraph II. Cyan vertices represent the pair to be swapped. **I.** Initial state with not fully defined particle count (see. eq. (5.9)). **II.** Beam splitter (BS) insertion with subsequent coincidence detection projects the ancillae into the Bell state $|\Phi^+\rangle$, completing the swap. **III.** Filtering the ancilla Bell state on $|01\rangle$ reduces the edge count. **IV.** Insertion of a $\lambda/2$ plate to turn the ancilla state into $|00\rangle$. 6-fold coincidence post-selection now leads to the state given by eq. (5.8). Greyed-out edges do not contribute.

(5.9) are relevant as 2+2 or 4+4 cannot lead to 6-fold coincidence. Hence, the incident state on the beam splitter is:

$$|1q\rangle_{a_0, a_1} |p0\rangle_{b_1, c_0} |\Phi^+\rangle_{b_0, c_0} - |0p\rangle_{b_0, b_1} |q0\rangle_{a_1, c_1} |\Psi^-\rangle_{a_0, c_1}, \quad (5.10)$$

or, switching to creation operators for convenience:

$$\begin{aligned} & \hat{b}_{0,0}^\dagger \hat{b}_{1,p}^\dagger \left[\hat{a}_{1,q}^\dagger \hat{a}_{0,1}^\dagger (\hat{c}_{1,0}^\dagger)^2 - \hat{a}_{1,q}^\dagger \hat{a}_{0,0}^\dagger \hat{c}_{1,0}^\dagger \hat{c}_{1,1}^\dagger \right] \\ & + \hat{a}_{0,1}^\dagger \hat{a}_{1,q}^\dagger \left[\hat{b}_{1,p}^\dagger \hat{b}_{0,0}^\dagger (\hat{c}_{0,0}^\dagger)^2 + \hat{b}_{1,p}^\dagger \hat{b}_{0,1}^\dagger \hat{c}_{0,0}^\dagger \hat{c}_{0,1}^\dagger \right] |\text{vac}\rangle \end{aligned} \quad (5.11)$$

The quadratic terms of ancilla creators have the same prefactors, hence factorize. Their contribution after the beam splitter defined by equations (5.7) is:

$$(\hat{c}_{0,0}^\dagger)^2 + (\hat{c}_{1,0}^\dagger)^2 \rightarrow (\hat{c}_{0,0}^\dagger + \hat{c}_{1,0}^\dagger)^2 + (\hat{c}_{1,0}^\dagger - \hat{c}_{0,0}^\dagger)^2 = 2(\hat{c}_{0,0}^\dagger)^2 + 2(\hat{c}_{1,0}^\dagger)^2$$

Thus, no coincidence in the output paths of the beam splitter can come from these terms. Removing the corresponding terms from equation (5.11) the state becomes:

$$- \left[\hat{b}_{0,0}^\dagger \hat{b}_{1,p}^\dagger \hat{a}_{1,q}^\dagger \hat{a}_{0,0}^\dagger \hat{c}_{1,0}^\dagger \hat{c}_{1,1}^\dagger \right] + \left[\hat{a}_{0,1}^\dagger \hat{a}_{1,q}^\dagger \hat{b}_{1,p}^\dagger \hat{b}_{0,1}^\dagger \hat{c}_{0,0}^\dagger \hat{c}_{0,1}^\dagger \right] |\text{vac}\rangle \quad (5.12)$$

Therefore, after the beam splitter the final state is derived as follows:

$$\begin{aligned}
& -\hat{b}_{0,0}^\dagger \hat{b}_{1,p}^\dagger \hat{a}_{1,q}^\dagger \hat{a}_{0,0}^\dagger \hat{c}_{1,0}^\dagger \hat{c}_{1,1}^\dagger |\text{vac}\rangle + \hat{a}_{0,1}^\dagger \hat{a}_{1,q}^\dagger \hat{b}_{1,p}^\dagger \hat{b}_{0,1}^\dagger \hat{c}_{0,0}^\dagger \hat{c}_{0,1}^\dagger |\text{vac}\rangle \\
\rightarrow & -\hat{b}_{0,0}^\dagger \hat{b}_{1,p}^\dagger \hat{a}_{1,q}^\dagger \hat{a}_{0,0}^\dagger (\hat{c}_{1,0}^\dagger - \hat{c}_{0,0}^\dagger) (\hat{c}_{1,1}^\dagger - \hat{c}_{0,1}^\dagger) |\text{vac}\rangle \\
& + \hat{a}_{0,1}^\dagger \hat{a}_{1,q}^\dagger \hat{b}_{1,p}^\dagger \hat{b}_{0,1}^\dagger (\hat{c}_{0,0}^\dagger + \hat{c}_{1,0}^\dagger) (\hat{c}_{0,1}^\dagger + \hat{c}_{1,1}^\dagger) |\text{vac}\rangle \\
= & \hat{b}_{0,0}^\dagger \hat{b}_{1,p}^\dagger \hat{a}_{1,q}^\dagger \hat{a}_{0,0}^\dagger (\hat{c}_{1,0}^\dagger \hat{c}_{0,1}^\dagger + \hat{c}_{0,0}^\dagger \hat{c}_{1,1}^\dagger) |\text{vac}\rangle \\
& + \hat{a}_{0,1}^\dagger \hat{a}_{1,q}^\dagger \hat{b}_{1,p}^\dagger \hat{b}_{0,1}^\dagger (\hat{c}_{0,0}^\dagger \hat{c}_{1,1}^\dagger + \hat{c}_{1,0}^\dagger \hat{c}_{0,1}^\dagger) |\text{vac}\rangle \\
= & |qp\rangle_{a_1, b_1} |\Phi^+\rangle_{a_0, b_0} |\Psi^+\rangle_{c_0, c_1}
\end{aligned}$$

The corresponding graph is shown in Figure 26 II. After applying filters and the mode flip as before the correct desired subgraph is obtained producing the state of equation (5.8). In the setup present here, the crucial step for the creation of the non-local Bell state is the destructive interference of coincidence contributions from the quadratic creator terms in equation (5.11). This interference removes the processes where the a_0 and b_0 photons are in opposite modes.

Interestingly, the final contributing terms correspond to cases, where Charlie's photons come either both from Alice or both from Bob. In the 1-pair cases, Charlie always receives one from each, although it is unknown which by whom. However, performing swapping with two extra photons comes at a cost. In the standard 1-pair case, once Charlie successfully measures his Bell state, and communicates this result to one party, e.g. Alice, they together can confirm a successful swap. I.e. once Alice has confirmed the existence of her photon, Charlie and Alice can herald Bob's photon state because they know it is entangled to Alice's photon. In the scheme described in this paragraph, Alice and Charlie, cannot guarantee the existence of Bob's photons, even if all their measurements have been successful. Hence, they don't know if Alice's photon is part of an entangled pair or not.

Technically, the graph obtained via the beam splitter transformation has two extra edges, not part of subgraph II, as shown in Figure 26 II-IV. They do not contribute to perfect matchings of the graph, however, they show how subgraph I can be automatically embedded in subgraph II when following this derivation method. This is by design of the initial state. By loosening the conditions on the initial state to also include terms where the paths a_1 and b_1 do not contain a photon, but all others do, the pre-state is amended by the term of the pre-state of subgraph I given by (5.6). Pre-filter, the combined post-swap state does not factorize, due to the different Bell states, however, post-filter, the output state can be written as

$$|\Phi^+\rangle_{a_0, b_0} |00\rangle_{c_0, c_1} \left(|\text{vac}\rangle_{a_1, b_1} + |qp\rangle_{a_1, b_1} \right). \quad (5.13)$$

Note, conditioning the graph on four-fold coincidence in the vertices a_0, b_0, c_0, c_1 leads to additional terms with two photons in a_1 and 0 in b_1 or vice versa omitted in equation (5.13). They are excluded, once n-fold coincidence post-selection is used on all vertices of the full swapping graph. Thus the output state of equation (5.13) describes the contribution of the state with the full graph, not the output of the structure taken independently.

Constructing 3-Pair Entanglement Swapping With the knowledge of the previous two paragraphs, one can already construct the state of the 3-pair graph from Figure 22, by taking two copies of the combination of subgraphs I and II and merging at the a_1, b_1 vertices. The 3-pair state can then be constructed as such:

$$\begin{aligned} & \left[|\Phi^+\rangle_{a_0, b_0} |00\rangle_{c_0, c_1} \left(|\text{vac}\rangle_{a_1, b_1} + |qp\rangle_{a_1, b_1} \right) \right] \\ \otimes & \left[|\Phi^+\rangle_{a_2, b_2} |00\rangle_{c_2, c_3} \left(|\text{vac}\rangle_{a_1, b_1} + |q'p'\rangle_{a_1, b_1} \right) \right] = \\ & |\Phi^+\rangle_{a_0, b_0} |\Phi^+\rangle_{a_2, b_2} \left(|\text{vac}\rangle_{a_1, b_1} + |qp\rangle_{a_1, b_1} + |q'p'\rangle_{a_1, b_1} + |qpq'p'\rangle_{a_1, b_1, a_1, b_1} \right) \\ & \otimes |0000\rangle_{c_0, c_1, c_2, c_3} \end{aligned}$$

By post-selecting on ten-photon coincidence the state transforms to:

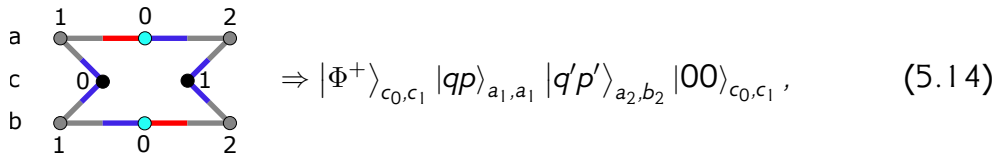
$$|\Phi^+\rangle_{a_0, b_0} |\Phi^+\rangle_{a_2, b_2} \left(|qp\rangle_{a_1, b_1} + |q'p'\rangle_{a_1, b_1} \right) |0000\rangle_{c_0, c_1, c_2, c_3}$$

Now one chooses $q = p = 0$; $q' = p' = 1$ to arrive at the output state of equation (5.5):

$$|\Phi^+\rangle_{a_0, b_0} |\Phi^+\rangle_{a_2, b_2} \left(|00\rangle_{a_1, b_1} + |11\rangle_{a_1, b_1} \right) |0000\rangle_{c_0, c_1, c_2, c_3}$$

One potential setup for the 3-pair graph consists of two copies of the setup shown in Figure 26 IV. The final output state is created by making the a_1, b_1 paths identical of both setups. Thus, the third entangled pair is created via the superposition of two additional photon pairs, created non-locally by two independent 1-pair entanglement swapping experiments. The two ancillae used in the 1-pair experiment are not able to create a fully independent entangled additional pair, but they can correlate two photons emitted by different pair sources as exhibited by subgraph II. Hence, the combined graph from subgraphs I and II acts as a 1-pair entanglement swapping setup, and simultaneously as a probabilistic non-local correlated pair source. Superimposing the emission of two such pairs then generates a third entangled pair. This pattern is visible in the perfect matchings of the 3-pair graph (See Fig. 22). All perfect matchings are combinations of perfect matchings from either subgraph I on the left end and subgraph II on the right end or vice versa. From the two perfect matchings of each subgraph this creates the eight matchings required for creating three entangled pairs.

Subgraph III: Four Additional Photons To generalize to arbitrary odd pair numbers subgraph III from Figure 24 is required. In essence, this is a further adaption to the second subgraph to create four additional photons instead of two. It appears in the center of the five-pair solution (see. Figure 23). Its state is given by:



$$\Rightarrow |\Phi^+\rangle_{c_0, c_1} |qp\rangle_{a_1, a_1} |q'p'\rangle_{a_2, b_2} |00\rangle_{c_0, c_1}, \quad (5.14)$$

with arbitrary q, p, q', p' . Thus it generates on non-local Bell pair and two additional non-local correlated photon pairs at the cost of two ancillae. This graph does not share edges with subgraph I, instead, subgraph I is fully embedded in it

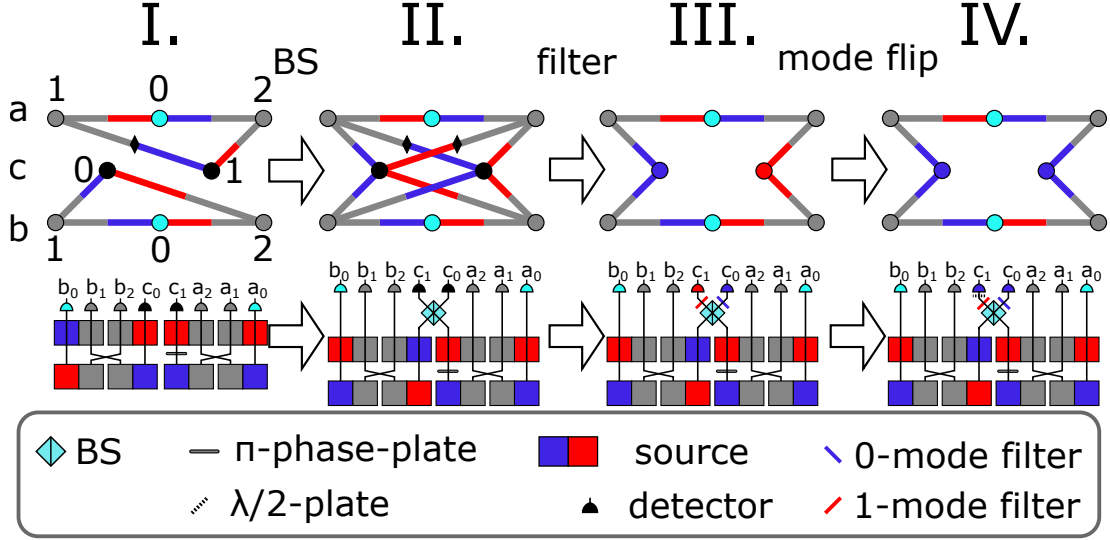


Figure 27: Graph and setup transformation for subgraph III. Cyan vertices represent the pair to be swapped. **I.** Initial state of two independent entanglement swapping experiments (see. eq. (5.15)). **II.** Insertion of a Beam Splitter (BS) in paths c_0 and c_1 with coincidence detection swaps entanglement to a_0, b_0 **III.** Filtering the ancilla Bell-state on $|01\rangle$ reduces the edge count. **IV.** Insertion of a $\lambda/2$ plate to turn the ancilla state into $|00\rangle$. 8-fold coincidence detection produces the state of eq. (5.14).

in the central piece of the chain of the 5-pair graph in Figure 23. Therefore, as with subgraph II a pre-swap state can be derived by reversing the transformation on the setup shown in Figures 25 and 26.

The obtained pre-swap state is given by the graph in Figure 27 I:

$$-|\Phi^-\rangle_{a_0, c_1} |qq'\rangle_{a_1, a_2} |\Psi^+\rangle_{b_0, c_0} |pp'\rangle_{b_1, b_2} \quad (5.15)$$

It consists of two modified copies of subgraph I, one shared by Charlie and Alice, and one by Charlie and Bob. The edge colors are modified to produce different Bell states to subgraph I in the a_0, c_1 and b_0, c_0 paths respectively. Hence, this pre-state can be interpreted as the result of two independent entanglement swapping experiments, where a_1, a_2 and b_1, b_2 are used as ancillae. As both c_0 and c_1 are each one partner of a Bell state, the insertion of the beam splitter just creates a regular entanglement swapping setup, although with different incident Bell states compared to subgraph I. As before, using the beam splitter defined by equations (3.15), the state transforms as such:

$$\begin{aligned} -|\Phi^-\rangle_{a_0, c_1} |\Psi^+\rangle_{b_0, c_0} &\rightarrow \left[-\hat{a}_{0,0}^\dagger (\hat{c}_{1,0}^\dagger - \hat{c}_{0,0}^\dagger) + \hat{a}_{0,1}^\dagger (\hat{c}_{1,1}^\dagger - \hat{c}_{0,1}^\dagger) \right] \\ &\otimes \left[\hat{b}_{0,0}^\dagger (\hat{c}_{0,1}^\dagger + \hat{c}_{1,1}^\dagger) + \hat{b}_{0,1}^\dagger (\hat{c}_{0,0}^\dagger + \hat{c}_{1,0}^\dagger) \right] |\text{vac}\rangle \\ &= \hat{a}_{0,0}^\dagger \hat{b}_{0,0}^\dagger (\hat{c}_{0,0}^\dagger \hat{c}_{1,1}^\dagger - \hat{c}_{1,0}^\dagger \hat{c}_{0,1}^\dagger) |\text{vac}\rangle \\ &\quad + \hat{a}_{0,1}^\dagger \hat{b}_{0,1}^\dagger (\hat{c}_{0,0}^\dagger \hat{c}_{1,1}^\dagger - \hat{c}_{1,0}^\dagger \hat{c}_{0,1}^\dagger) |\text{vac}\rangle \\ &= |\Phi^+\rangle_{a_0, b_0} |\Psi^-\rangle_{c_0, c_1} \end{aligned}$$

Here, the coincidence criterion was immediately used to omit terms producing two photons in one beam splitter output path. Re-adding the a_1, a_2 and b_1, b_2 product states returns the state of the graph depicted in Figure 27 II:

$$|\Phi^+\rangle_{a_0, b_0} |\Psi^-\rangle_{c_0, c_1} |pp'\rangle_{b_1, b_2} |qq'\rangle_{a_1, a_2} \quad (5.16)$$

Which, after applying the filter and wave plate on ancillae creates the desired output state of subgraph III (5.14).

As before with subgraph II, inserting subgraph I into III at the vertices, a_0, b_0, c_0, c_1 and then considering events where these have one photon each, and the a_1, b_1 and a_2, b_2 pairs have both 0 or both 1 photon leaves superpositions of an entangled pair at a_0, b_0 and 0, 2 or 4 additional photons spread over the a_1, b_1, a_2, b_2 modes.

$$\begin{aligned} |\Phi^+\rangle_{a_0, b_0} & \left(|\text{vac}\rangle_{a_1, b_1, a_2, b_2} \right. \\ & + |\text{vac}\rangle_{a_1, b_1} |q'p'\rangle_{a_2, b_2} \\ & + |qp\rangle_{a_1, b_1} |\text{vac}\rangle_{a_2, b_2} \\ & \left. + |qp\rangle_{b_1, b_1} |q'p'\rangle_{a_2, b_2} \right) |00\rangle_{c_0, c_1} \end{aligned} \quad (5.17)$$

Thus, subgraph III+I is a 1-pair entanglement swapping experiment, that simultaneously acts as either one or two probabilistic non-local photon pair sources. The two 1-pair terms originate from the fact that by generating the subgraph III+I subgraph II is found twice within (see. Figure 28). As with the subgraph II+I, equation (5.17) does not represent the heralded output state on vertices a_1, b_1, a_2, b_2 based on four-fold coincidence detection in vertices a_0, a_1, c_0, c_1 . This state has additional terms that are excluded once post-selection on all vertices in a full swapping graph.

N-pair entanglement swapping Similarly to the three particle case to obtain the mechanism for arbitrary odd pair numbers 2 copies of the superposition of subgraphs I and II and arbitrary many copies of the superposition of subgraphs I and III are merged in sequence, as shown in Figure 28, Using equation (5.13) and (5.17) the resulting superposition for $N = 3 + 2n$, $n \geq 0$ pairs can be described as such:

$$\begin{aligned} & |\Phi^+\rangle_{a_0, b_0} \left(|\text{vac}\rangle_{a_1, b_1} + |00\rangle_{a_1, b_1} \right) \\ & \bigotimes_{i=0}^{n-1} |\Phi^+\rangle_{a_{2i+2}, b_{2i+2}} \left(|\text{vac}\rangle_{a_{2i+1}, b_{2i+1}, a_{2i+3}, b_{2i+3}} \right. \\ & \quad + |11\rangle_{a_{2i+1}, b_{2i+1}} |\text{vac}\rangle_{a_{2i+3}, b_{2i+3}} + |\text{vac}\rangle_{a_{2i+1}, b_{2i+1}} |00\rangle_{a_{2i+3}, b_{2i+3}} \\ & \quad \left. + |11\rangle_{a_{2i+1}, b_{2i+1}} |00\rangle_{a_{2i+3}, b_{2i+3}} \right) \\ & \otimes |\Phi^+\rangle_{a_{N-1}, b_{N-1}} \left(|\text{vac}\rangle_{a_{N-2}, b_{N-2}} + |11\rangle_{a_{N-2}, b_{N-2}} \right) \end{aligned} \quad (5.18)$$

where the first and last lines are the superpositions of subgraphs I and II and the middle lines an arbitrary number of n superpositions of subgraphs I and III. The

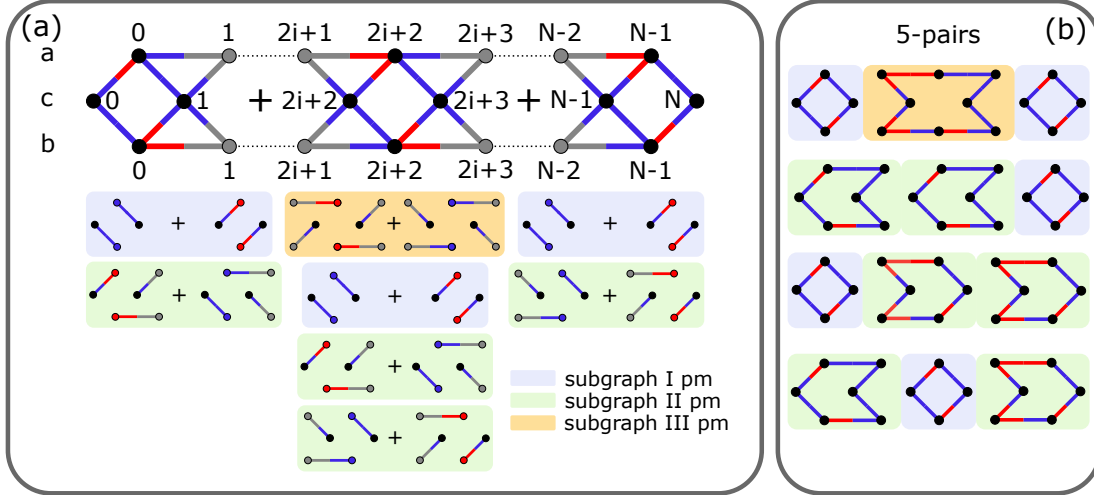


Figure 28: (a) Construction of the graph non-locally creating an arbitrary odd number of Bell pairs. The central element is repeated an arbitrary number of times adding two pairs per repetition to the base count of three. The perfect matchings (pm) can be constructed out of the pm of the three subgraphs. (b) Example for pm construction for five pairs with abbreviated matchings (see. Fig. 10(b)). As each subgraph has 2 pms each row represents $2^3 = 8$ pms, totaling to $5^2 = 32$ matchings of the five-pair graph. Code for generating any length chain is given in App. C.2.

correct output state is post-selected on coincidence on all detectors to arrive at the general multipair entanglement state:

$$\bigotimes_{i=0}^{N-1} |\Phi^+\rangle_{a_i, b_i} \bigotimes_{j=0}^N |0\rangle_{c_j} \quad (5.19)$$

Evidently, to non-locally create N bell pairs we require $N+1$ ancillae, in contrast to $2N$ ancillae when stacking standard one-pair entanglement swapping experiments. The extra Bell pairs are created at the joining odd-numbered vertices. Here two subgraphs meet and share two vertices. As those subgraphs also act as probabilistic correlated pair sources, if all detectors click, it is unknown which subgraph emitted the correlated pair, hence the state is a superposition of both options creating the Bell State $|\Phi^+\rangle$. As the subgraphs achieve the correlated pair emission non-locally. This Bell pair is also produced non-locally. This is mirrored in the perfect matchings of the total chain, where each elements either contributes via perfect matchings of subgraph I or II for the end pieces or I, II or III for the central pieces. This is visualized in Figure 28.

Conclusions on 2-dimensional Multipair Entanglement Swapping In summary, the additional entangled pairs stem from experiments that simultaneously perform entanglement swapping, and act as a non-local source for correlated photon pairs. By choosing the correct product states of each source and superimposing the output paths of the sources, entanglement is created. Thus, more entangled pairs can be created with fewer ancillae, but this has a cost, already touched upon while discussing subgraph II. For this theoretical argument, no loss in the entire system

and photon number resolving detectors is assumed for the swap of N pairs. As mentioned before, Charlie and one other party can herald the state of the third party in standard entanglement swapping. In the multipair scheme, this is not the case. Taking the 3-pair graph as an example, after measuring his four photons, Charlie can only confirm the existence of eight photons. Even with knowledge of e.g. Alice's successful measurement this number does not increase, as it is still possible that Bob has only one photon. To obtain full knowledge, Charlie requires confirmation from both Alice and Bob, that they have their full photon state, only then, is the swap confirmed to be complete. Thus, in the multipair scheme two parties can never herald the state of the third party as in standard entanglement swapping. With increasing pair count, Charlie obtains less and less information on his own, as the fraction of photons of the total state he measures decreases. In practice, when loss is present and if photon number resolving detectors are not used, confirmation from both parties is necessary also in the standard scheme, however in the multipair setup the requirement is fundamental. Additionally, the minimum amount of total photons required to be measured for N pairs is $3N$ in both cases, hence there is no increase in the information carried in each ancilla in the multipair scheme. This is because the existence of the last photon can always be heralded in any graph, due to the use of pair sources. The information not obtained by Charlie, but required to fully define the state is outsourced to measurements made by Alice and Bob.

Through graph analysis, a separable pre-swap state is discovered for each subgraph. Thus a clear method of how to create the non-local entanglement from local entanglement and correlations is shown for each subgraph. However, due to the required post-selection, these initial states don't combine to a fully separable initial state of the full graphs. The entangled pairs at the connections will already be created. This method is likely not unique but only one example of how this could be constructed. In the case of subgraph III, it is shown, that the output can be generated through three subsequent standard entanglement swapping experiments. Thus, the function of this subgraph is akin to a quantum repeater, where some of the ancillae are reused to generate extra photon pairs after they perform their swap. However, the same is not achieved for subgraph II.

5.3.3 Application to d-dimensional Entanglement Swapping

Initial Considerations D-dimensional entanglement swapping specifies the non-local creation of entangled states of more than two degrees of freedom. In this section, the created states are high-dimensional generalizations of the $|\Phi^+\rangle$ Bell state:

$$|\Phi_d^+\rangle = \sum_{j=0}^{d-1} |jj\rangle \quad (5.20)$$

As previously, the normalization constant of $\frac{1}{\sqrt{d}}$ is omitted to keep all edges to ± 1 for better clarity. Using the multipair scheme one can naturally create such high dimensional pairs at the vertices of the additionally created pairs where the subgraphs are joined. As these are created by the superposition of macroscopic, independent setups acting as pair sources, adding more setups adds new terms to

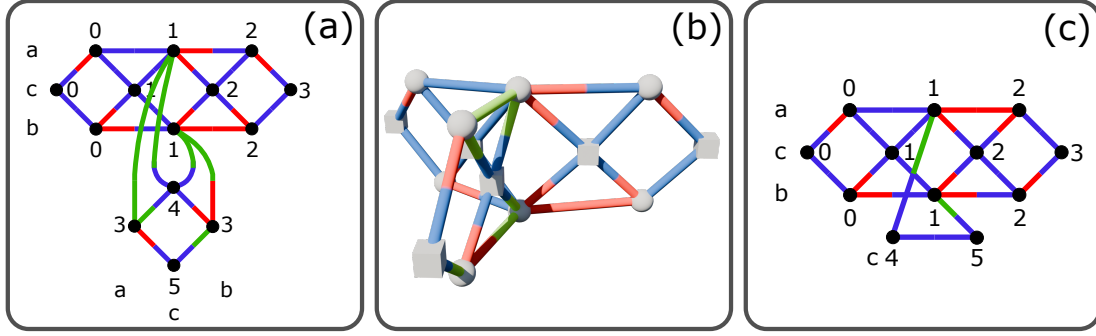


Figure 29: *Semi-3D entanglement swapping, (a) Graph creating three two-dimensional Bell states at pairs 0, 2, and 3 as well as one three-dimensional Bell state at pair 1 in 2D. (b) Same graph as (a) but in 3D highlighting the rotational symmetry. An edge list definition of this graph is available at App. A.4 (c) Graph creating two 2-dimensional Bell-pairs and one three-dimensional Bell-pair.*

the superposition. In graph form, this extension is simply done by adding another copy of subgraph II+I (see Fig. 24) at the intersecting pair with appropriately modified coloring. This process results in a graph shown in Figure 29 (a),(b) creating the state

$$|\Phi^+\rangle_{a_0,b_0} |\Phi^+\rangle_{a_2,b_2} |\Phi^+\rangle_{a_3,b_3} |\Phi_3^+\rangle_{a_1,b_1} |000000\rangle_{c_0,c_1,c_2,c_3,c_4,c_5} \quad (5.21)$$

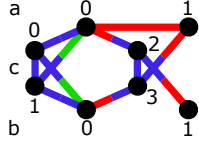
of three swapped 2-dimensional Bell pairs and one 3-dimensional Bell pair $|\Phi_3^+\rangle_{a_1,b_1}$. In 3D, the resulting graph is rotationally symmetric around the axis of the central pair a_1, b_1 , as shown in Figure 29(b). This works because the subgraph effectively acts as a non-local source of correlated photon pairs. Adding another copy just adds another non-local pair source to the central vertices. Thereby, is not required, to add a full third two-dimensional Bell pair to achieve the three-dimensional state, instead, one can also just add a minimal graph for such a non-local pair source, as shown in Figure 29(c). The added structure here is a simple form of a Halo as introduced in section 3.3 [14]. By adding more structures, either of the form in Figure 29(a) or in 29(c), the intersection pair can be generalized to arbitrary dimension. It is of note, that the latter Halo structure can support one additional dimension at no extra ancilla cost [14]. Hence, it has an advantage in ancillae if no extra two-dimensional pairs are desired. Using additional two-dimensional pairs one obtains the general state for d additional structures:

$$|\Phi_d^+\rangle_{a_d,b_d} \bigotimes_{i=0}^{d-1} |\Phi_{a_i,b_i}^+\rangle \bigotimes_{j=0}^{2(d-1)} |0\rangle_{c_j} \quad (5.22)$$

The central high-dimensional pair is created at no additional ancillae cost to the two ancillae required per two-dimensional pair.

True Multi-pair High-dimensional ES The base mechanism behind the ancillae advantage in the $2d$ -case is the creation of superposition in equation 5.13 between the creation of a nonlocal Bell-pair with and without an extra correlated photon

pair. Hence, to generalize to higher dimensions, a graph producing an analogous superposition for a d -dimensional Bell-pair is required. Using AriadneVR, such a graph was extracted from a PyTheus discovery swapping two 3-dimensional pairs with six ancillae (graph 78. in Ref. [12]). This graph and the extracted subgraph are shown in Figures 30(a) and (b). The extracted subgraph produces the state:



$$\Rightarrow |\Phi_3^+\rangle_{a_0,b_0} \left(|\text{vac}\rangle_{a_1,b_1} + |11\rangle_{a_1,b_1} \right) \bigotimes_{j=0}^3 |0\rangle_{c_j} \quad (5.23)$$

By then combining three such graphs in a trifold manner as described in the previous paragraph (see. Fig. 29) a graph creating four 3-dimensional pairs, at the ancilla cost of three pairs is obtained. Figure 30(c) shows the interplay between the resulting categories of perfect matchings for a simplified graph only containing two copies of the extracted subgraph in an abbreviated form as introduced in Figure 10(b). As in the 2-dimensional case, the full state is obtained through the superposition of either subgraph emitting one extra photon pair. Figure 30(d) shows the process of using the trifold construction method to arrive at the graph for 4-pair 3-dimensional entanglement swapping. This construction was performed within AriadneVR using its natural graph drawing capability. The resulting graph has 12 ancillae, four less than required when stacking four 3-dimensional 1-pair graphs. However, the 2-pair graph (see. Fig. 30) exhibits the same scaling. Two copies of this graph also create four pairs with 12 ancillae. The discovery of the 4-pair 3-dimensional entanglement swapping graph has been published in the preprint related to this work Ref. [15].

In principle, the 1-pair 3-dimensional graph contained in the extracted subgraph (see. Fig. 30(b)) can be replaced with higher dimensional subgraphs at the cost of two ancillae per two dimensions by copying the c_0 and c_1 ancillae structure. This generates higher dimensional entanglement swapping as described in Ref. [14]. By adding the appropriate number of extracted subgraphs the central pair can also be elevated to arbitrary dimension. Therefore, if the 1-pair subgraph requires n -ancillae an extra d -dimensional pair can be generated by superimposing d individual such subgraphs. However, this method results in reduced ancillae efficiency compared to stacking 1-pair graphs as only one free pair is obtained.

$$\frac{\#\text{ancillae superposition}}{\#\text{ancillae single pair stack}} = \frac{d \cdot n}{(d+1) \cdot n} = \frac{d}{d+1} \approx 1 \text{ for large } d$$

This discussion leaves the question, of what would be required to obtain the remarkable scaling of the required ancillae in the two-dimensional case. In essence, for the three-dimensional solution, analog graphs of the endpieces of the 2-dimensional chain (subgraph I + II) are available but an analog of the central piece of the chain (subgraph I + III) is missing. Such an analog would have to produce the following state,

$$|\Phi_d^+\rangle_{a_d,b_d} \bigotimes_{j=0}^{d-1} \left(|\text{vac}\rangle_{a_j,b_j} + |jj\rangle_{a_j,b_j} \right) \bigotimes_{i=0}^{n-1} |0\rangle_{c_i} \quad (5.24)$$

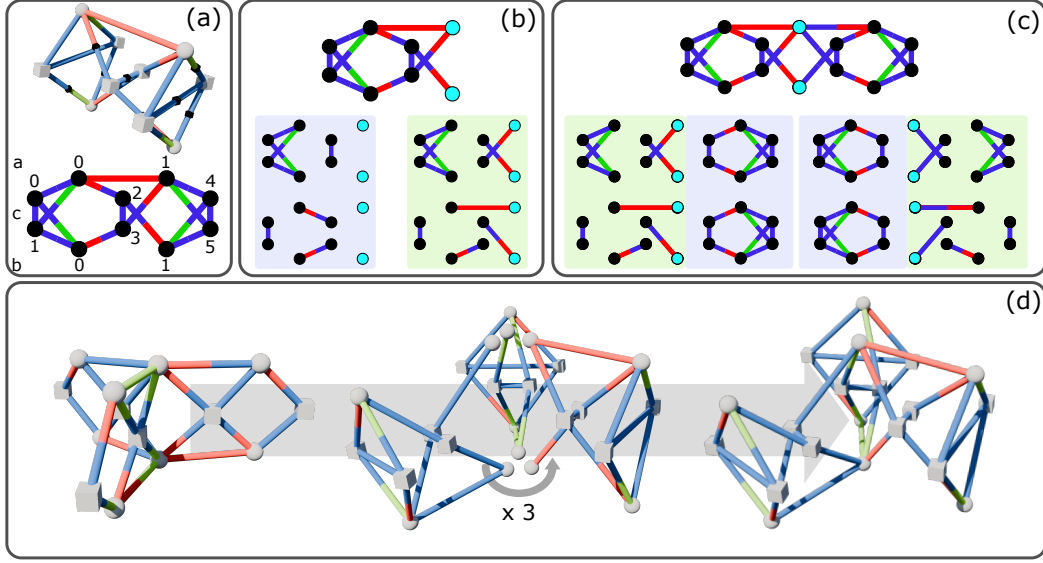


Figure 30: Derivation of full 3-dimensional multipair entanglement swapping for four pairs. **(a)** 2-pair 3-dimensional entanglement swapping graph in 2D and 3D (Graph 78 in Ref. [12]). **(b)** Extracted analog graph to the end-piece of the 2-dimensional entanglement swapping chain (see. Fig. 28). The open end is the a_1, b_1 detector pair from (a), highlighted in cyan. The graph has **internal** (blue) partial covers, creating 1 3-dimensional pair and **1-sided** (green) full covers, creating 1 pair and a product state $|11\rangle$ **(c)** First step towards the four pair graph, combining two copies of graph (b) to arrive at two 3-dimensional pairs and one 2-dimensional pair at the connection (cyan). **(d)** By copying the end piece (b) a third time as in Figure 29 full 4-pair 3-dimensional entanglement swapping is obtained. The final graph has 12 ancillae. Edge list definition available at App. A.5. Some matchings are abbreviated, see. Figure 10(b).

where n is the number of required ancillae and the product states $|ij\rangle_{a_j, b_j}$ and the d -dimensional Bell-pair are created non-locally. The detector index is chosen here to be the mode number for clarity not by necessity. Such a graph would swap one high-dimensional pair and create between 0 and d additional correlated photon pairs. These additional product states could then be used to construct additional high-dimensional entangled pairs resulting in a graph exhibiting a mesh-like structure, instead of a 1-D chain of the 2-dimensional case.

5.3.4 Generalising beyond ES graphs and the Connection to the Halo concept

As elaborated in the two previous sections, the ancillae advantage achieved by the entanglement swapping graphs relies on the core concept of graphs achieving a particular superposition of particle numbers:

$$|\psi\rangle \bigotimes_{j=0}^{n-1} (|\text{vac}\rangle + |\varphi_j\rangle_{\{P_j\}}) \quad (5.25)$$

They perform some core experimental tasks, generating state $|\psi\rangle$, but with the option of emitting up to n additional states $|\varphi_j\rangle$ emitted into the set of path

modes $\{P_j\}$. Ignoring the core task, they act as independent, probabilistic, non-local photon sources of the states $|\varphi_j\rangle$. This mechanism is closely related to how Halo-graphs [14] emit their correlated multiphoton states (see. sec. 3.3). By necessity, Halos function in a binary way, either, they are *off* and the original graph functions as it does before the addition of the Halo, or they are *on* where the original graph does not contribute and the Halo provides all photons to its vertices. All other contributions must interfere destructively or be excluded by post-selection. To be able to be "off" in this way a Halo needs an *internal* perfect-matching where all edges are confined to the vertices of the Halo and no vertices of the original graph (see. section 3.3). In the standard Halo picture, these vertices are classified as ancillae hence their state is of no concern and does not contribute to the desired output of the graph.

There are two crucial modifications to the Halo concept exhibited by equation (5.25). Firstly, not all detectors forming the source are considered ancillae. In the entanglement swapping case, two of the four photons of the structure are ancillae, and the other two form a Bell pair, which is part of the desired output state. In a sense, the internal structure of the Halo provides a useful contribution to the experiment beyond acting as a multi-photon source. The second modification is that structures described by equation (5.25) act as multiple independent sources, instead of one. Their function is not necessarily binary, i.e. *on* or *off*, they can independently provide anything between 0 and n photon states simultaneously. This is not the same as two-dimensional Halos, like they appear in single pair high-dimensional entanglement swapping [14] or in the $|GHZ\rangle_3^5$ graph (see section 5.1.3). Those structures cannot emit two states at the same time. They do not act as two independent sources of $|\varphi_0\rangle$ and $|\varphi_1\rangle$ but as a source for $|\varphi_0\rangle + |\varphi_1\rangle$. In the entanglement swapping case the two-sided structure in the center of the chain (see. Fig. 28) is an example of acting as two independent sources in accordance with equation (5.25). Its output state, neglecting the Bell pair, is $|\text{vac}\rangle + |00\rangle_{ab} + |11\rangle_{cd} + |00\rangle_{ab}|11\rangle_{cd}$ (see. equation (5.18)), meaning it can either emit nothing, two photons in states $|00\rangle_{ab}$ or $|11\rangle_{cd}$ or four photons in the state $|00\rangle_{ab}|11\rangle_{cd}$. Note that the path modes indicated by the subscript are different for each additionally emitted pair. In the following, each collection of path modes the source emits a state into is called a *facet*. The graph employed in the 2-dimensional chain has two 2-photon facets.

In summary, subgraphs described by equation (5.25) can be described as *multifaceted* Halos, structures acting as multiple independent photon sources, while always generating an internal state $|\psi\rangle$. This constitutes a meaningful extension of the Halo concept.

One way to construct multi-faceted Halos different from swapping graphs is to use 2-dimensional GHZ-states as the internal structure. Arbitrarily large, even-numbered GHZ-states can be created by alternating edge colors in polygon loops [112]. Adding one additional vertex per edge of such a polygon and connecting to the closest edges can create star-shaped graphs fulfilling equation 5.25. Essentially, these stars represent the placement of a smaller GHZ-type-polygon graph with a larger one. The resulting graph structure is a straight generalization of the central graph sub-graph of the two-dimensional entanglement

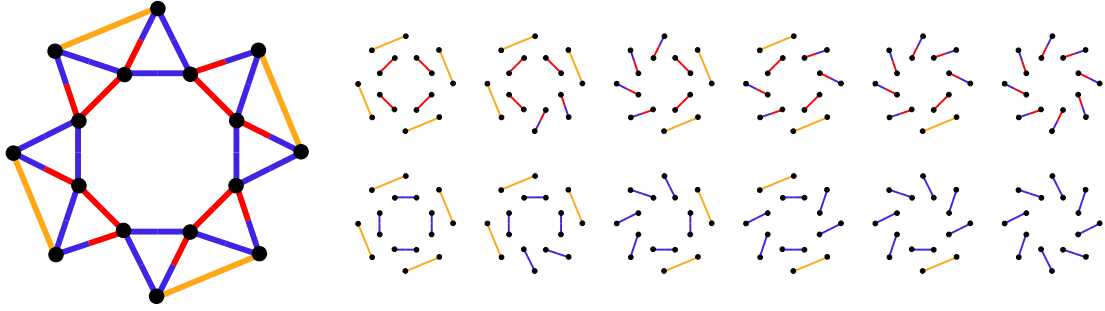


Figure 31: Example of a generalization of a multi-faceted Halo according to equation (5.26) with its perfect matchings. Orange edges are not part of the Halo but represent the simplest form of graph the four facets of the halo can be attached to. The top row of perfect matchings always generates the state $|1\rangle^{\otimes 8}$ internally, the bottom row $|0\rangle^{\otimes 8}$. From left to right the perfect matchings emit between zero and four additional pairs of photons in the state $|00\rangle$. Perfect matchings obtainable by rotation from the depicted ones are omitted.

swapping chain. An example is depicted in Figure 31 for a GHZ state of size 8. For a GHZ-state of size n as the central graph, the result features up to $\frac{n}{2}$ additional emitted photon pairs:

$$|GHZ\rangle_2^n \bigotimes_{j=0}^{\frac{n}{2}-1} (1 + |q_j p_j\rangle) \quad (5.26)$$

This means it acts as a Halo with $\frac{n}{2}$ facets. Different types of facets can be created from this graph, if instead of using pairwise groupings with normal edges, hyperedges are employed. This way one can construct e.g. a multi-faceted Halo with one 4-photon facet and two 2-photon facets by connecting four neighboring outer vertices of the graph depicted in Figure 31 with one hyperedge instead of two normal edges. Note that by construction, all facets are non-local, i.e. no photons of the facet share a pair source, this is a general feature of Halos. Such faceted Halos however have a significant caveat. Their function as correlated pair sources breaks down if the individual facets are not independent. For example, the central two-facet piece of the chain used in entanglement swapping cannot be repeated and then closed. i.e. the last facet connected to the first facet. Similarly, in the example using the GHZ-state presented above, edges between different facets can destroy the independent pair source nature of each facet.

5.3.5 On the Contribution of Ariadne VR

As mentioned in the introduction to this section, AriadneVR has been used repeatedly during this analysis process, to visualize and modify the presented graphs. The most important contribution during the early stages of analyzing the 2-dimensional swapping graphs was rejecting hypotheses for increasing the efficiency even further. Here, it provided an easy way to draw new or modified graphs and inspect their perfect matchings to check the result. However, none of these experiments proved successful. Hence, in the 2-dimensional case, the relevant contribution that enabled the discovery of the scaling method was achieved by the

KK-sorting algorithm, providing the necessary insight to set PyTheus on a search for the extension of the chain.

In the subsequent analysis of the search for pre-swap states AriadneVR did not play a large role. However, after the fundamental concept of using entanglement swapping experiments as pair sources were extracted, AriadneVR was used to generalize the central pair to three dimensions and to generalize the concept to three-dimensional graphs. When trying to construct multi-faceted Halos AriadneVR again was used for hypothesis testing of corresponding designs.

In Summary, in this particular example of VR-augmented quantum graph analysis, the interactivity provided by AriadneVR played a larger role than the 3D visualization. This is unsurprising, as the majority of graphs studied are planar. Hence, it is likely, that many of the discoveries presented here could have very well been executed without the use of virtual reality. AriadneVR probably made it easier and faster, through intuitive graph sorting and drawing especially when search the higher dimensional swapping solutions for analogs to the concepts extracted from the two-dimensional case. However, no quantitative analysis of the impact has been performed so this conclusion is subjective. Regardless, the lessons learned from this analysis prompted the implementation of the ability to define search templates for PyTheus searches into the tool. Another potentially useful extension that has not made it into the tool is the implementation of vertex operations, i.e. being able to apply matrix transformations on vertices. This would allow graphical computations and might increase the usefulness of AriadneVR substantially for the tasks of understanding underlying experimental processes behind a graph output, as was done through the search for pre-swap states.

6 Summary

This thesis work consists of the development of a custom webVR tool AriadneVR for the analysis of artificially discovered quantum optics experiments in graph form as well as the subsequent application of the tool in a variety of tasks around extracting insights from artificial scientific discoveries. It constitutes a new explorative application of an infrequently used method, Virtual Reality, to an emerging modern research challenge, understanding scientific results that have not been created by humans but AI.

The results consist of a demonstration of discoverable structures in a variety of graph-encoded experiments achieving a variety of different tasks. Beyond it showcases two examples where structural features discovered in VR proved themselves to be interpretable and gave the user new understanding of the graph representation as a whole as in the case of loops of zero contribution or a specific experiment as in the case of $|GHZ\rangle_3^5$ state creation. In the latter example, the obtained understanding could be used to design a new $|GHZ\rangle_3^7$ state-creation experiment utilizing Halos, a known concept from the graph representation. AriadneVR was used in attempts to increase the efficiency of searches for new experiments based on discovered geometries of related examples. This led to the successful discovery of GHZ-analyzer experiments in graph form for four particles in two and three

dimensions. The main result of this thesis is the study of AI-discovered resource-efficient entanglement swapping. Through mathematical and graphical analysis, the underlying mechanism of creating experiments that simultaneously act as probabilistic pair sources could be extracted. The understanding gained through this could then be reapplied to higher dimensional cases of entanglement swapping, using AriadneVR to identify the necessary analog structures in AI solutions and build generalizations. The fundamental concept behind the mechanism could be linked to the Halo concept, constituting a meaningful extension of the relevant ideas.

Thus, this thesis represents a successful application of VR for the artificial scientific discovery process, selected results have been published in Ref. [15].

7 Conclusion and Outlook

In conclusion, VR can be successfully applied to artificial scientific discoveries in the context of graph-encoded quantum experiments. It shines most in visual and structural analysis, but also provides a suitable interaction platform for finding and testing hypotheses relating to such graphs. It is also an efficient way to design graph structures, which can be used to manually generalize concepts or to design initial graphs for further AI searches. In the analysis of multi-pair entanglement swapping, VR was mainly used once the generalization of concepts to more complicated structures became relevant. Here, AriadneVR facilitated the discovery of analog structure that could then be used to manually generalize to a new solution. Thus, judging from the results of this work, VR as represented by AriadneVR is most useful at the beginning and end of the conceptualization of graph-encoded experiments. In the beginning, it facilitates the discovery of structure which can be used as a starting point for generalization (manually or by AI searches). Such generalization can then inspire conceptualization. At the end of conceptualization is the reapplication of the concepts to other use cases, where VR assists greatly through identifying the extracted concepts in other solutions or building abstractions. In the presented example of conceptualization, VR played less of a role during the actual extraction of the concept. One reason for this is the planar structure of the involved graph being sufficient for generalization. VR can be expected to take a bigger role in concept extraction based on more complex graphs. As mentioned, including more computational features like beam splitter graph transformations, controlled by simple touch interaction might greatly increase the use of the AriadneVR for computational analysis. Further application in different conceptualization attempts would shine further light on the use of such features and the further potential of VR for such analysis. In general, AriadneVR can be improved by streamlining the workflow. Useful additions for this purpose are some features often present in modern VR tools (e.g. in Narupa for interactive molecular simulations [71]) like support for collaborative analysis and access to advanced computing APIs. For AriadneVR, direct access to the PyTheus API could be very useful, especially for testing hypotheses and defining search spaces. This could be combined with more advanced UI for efficient and full control over PyTheus queries. In addition, implementing collaborative analysis would allow multiple researchers to experiment on graphs. Currently, AriadneVR

is also very restricted in the size of graphs it can display. This is in part by design, as it is an untethered application, however, considerable improvements are still possible through optimizing the employed rendering techniques.

As graphs are an abstract data representation, they are not tied to quantum optics experiments. Thus, the techniques employed in this thesis can in principle also be used for other graph-based tasks, and AriadneVR can potentially be extended to interact to include them [15]. An example could be graph-based quantum circuit design [125], especially with the graphs created by ZX-calculus [126].

This work represents a step towards an improved human-computer interface for understanding artificial intelligence. With increase in hardware capability, it seems likely that applications like this one will become more standard in the future.

References

- [1] 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 Alán Aspuru-Guzik. "On scientific understanding with artificial intelligence". In: *Nature Reviews Physics* 4.12 (Dec. 2022), pp. 761–769. doi: [10.1038/s42254-022-00518-3](https://doi.org/10.1038/s42254-022-00518-3).
- [2] Hanchen Wang, Tianfan Fu, Yuanqi Du, Wenhao Gao, Kexin Huang, Ziming Liu, Payal Chandak, Shengchao Liu, Peter Van Katwyk, Andreea Deac, Anima Anandkumar, Karianne Bergen, Carla P. Gomes, Shirley Ho, Pushmeet Kohli, Joan Lasenby, Jure Leskovec, Tie-Yan Liu, Arjun Manrai, Debora Marks, Bharath Ramsundar, Le Song, Jimeng Sun, Jian Tang, Petar Veličković, Max Welling, Linfeng Zhang, Connor W. Coley, Yoshua Bengio, and Marinka Zitnik. "Scientific discovery in the age of artificial intelligence". In: *Nature* 620.7972 (Aug. 2023), pp. 47–60. doi: [10.1038/s41586-023-06221-2](https://doi.org/10.1038/s41586-023-06221-2).
- [3] Alex Davies, Petar Veličković, Lars Buesing, Sam Blackwell, Daniel Zheng, Nenad Tomašev, Richard Tanburn, Peter Battaglia, Charles Blundell, András Juhász, Marc Lackenby, Geordie Williamson, Demis Hassabis, and Pushmeet Kohli. "Advancing mathematics by guiding human intuition with AI". In: *Nature* 600.7887 (Dec. 2021), pp. 70–74. doi: [10.1038/s41586-021-04086-x](https://doi.org/10.1038/s41586-021-04086-x).
- [4] Keith T. Butler, Daniel W. Davies, Hugh Cartwright, Olexandr Isayev, and Aron Walsh. "Machine learning for molecular and materials science". In: *Nature* 559.7715 (July 2018), pp. 547–555. doi: [10.1038/s41586-018-0337-2](https://doi.org/10.1038/s41586-018-0337-2).
- [5] Daniel C. Elton, Zois Boukouvalas, Mark D. Fuge, and Peter W. Chung. "Deep learning for molecular design—a review of the state of the art". In: *Molecular Systems Design & Engineering* 4.4 (2019), pp. 828–849. doi: [10.1039/C9ME00039A](https://doi.org/10.1039/C9ME00039A).
- [6] Benjamin Sanchez-Lengeling and Alán Aspuru-Guzik. "Inverse molecular design using machine learning: Generative models for matter engineering". In: *Science* 361.6400 (July 27, 2018), pp. 360–365. doi: [10.1126/science.aat2663](https://doi.org/10.1126/science.aat2663).
- [7] Carrie Arnold. "Inside the nascent industry of AI-designed drugs". In: *Nature Medicine* 29.6 (June 1, 2023), pp. 1292–1295. doi: [10.1038/s41591-023-02361-0](https://doi.org/10.1038/s41591-023-02361-0).
- [8] Carla Rodríguez, Sören Arlt, Leonhard Möckl, and Mario Krenn. *XLuminA: An Auto-differentiating Discovery Framework for Super-Resolution Microscopy*. Oct. 12, 2023. doi: [10.48550/arXiv.2310.08408](https://doi.org/10.48550/arXiv.2310.08408).
- [9] Mario Krenn, Mehul Malik, Robert Fickler, Radek Lapkiewicz, and Anton Zeilinger. "Automated Search for new Quantum Experiments". In: *Physical Review Letters* 116.9 (Mar. 4, 2016), p. 090405. doi: [10.1103/PhysRevLett.116.090405](https://doi.org/10.1103/PhysRevLett.116.090405).
- [10] P. A. Knott. "A search algorithm for quantum state engineering and metrology". In: *New Journal of Physics* 18.7 (July 2016), p. 073033. doi: [10.1088/1367-2630/18/7/073033](https://doi.org/10.1088/1367-2630/18/7/073033).
- [11] Mario Krenn, Jakob S. Kottmann, Nora Tischler, and Alán Aspuru-Guzik. "Conceptual Understanding through Efficient Automated Design of Quantum Optical Experiments". In: *Physical Review X* 11.3 (Aug. 26, 2021), p. 031044. doi: [10.1103/PhysRevX.11.031044](https://doi.org/10.1103/PhysRevX.11.031044).
- [12] 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". In: *Quantum* 7 (Dec. 12, 2023), p. 1204. doi: [10.22331/q-2023-12-12-1204](https://doi.org/10.22331/q-2023-12-12-1204).
- [13] Dang Minh, H. Xiang Wang, Y. Fen Li, and Tan N. Nguyen. "Explainable artificial intelligence: a comprehensive review". In: *Artificial Intelligence Review* 55.5 (June 1, 2022), pp. 3503–3568. doi: [10.1007/s10462-021-10088-y](https://doi.org/10.1007/s10462-021-10088-y).
- [14] Sören Arlt, Carlos Ruiz-Gonzalez, and Mario Krenn. *Digital Discovery of a Scientific Concept at the Core of Experimental Quantum Optics*. Oct. 18, 2022. doi: [10.48550/arXiv.2210.09981](https://doi.org/10.48550/arXiv.2210.09981).
- [15] Philipp Schmidt, Sören Arlt, Carlos Ruiz-Gonzalez, Xuemei Gu, Carla Rodríguez, and Mario Krenn. *Virtual Reality for Understanding Artificial-Intelligence-driven Scientific Discovery with an Application in Quantum Optics*. Feb. 20, 2024. doi: [10.48550/arXiv.2403.00834](https://doi.org/10.48550/arXiv.2403.00834).
- [16] Mario Krenn, Jonas Landgraf, Thomas Foesel, and Florian Marquardt. "Artificial intelligence and machine learning for quantum technologies". In: *Physical Review A* 107.1 (Jan. 3, 2023), p. 010101. doi: [10.1103/PhysRevA.107.010101](https://doi.org/10.1103/PhysRevA.107.010101).
- [17] Carl A Fogarty, Aoife M Harbison, Amy R Dugdale, and Elisa Fadda. "How and why plants and human N-glycans are different: Insight from molecular dynamics into the "glycoblocks" architecture of complex carbohydrates". In: *Beilstein Journal of Organic Chemistry* 16 (Aug. 21, 2020), pp. 2046–2056. doi: [10.3762/bjoc.16.171](https://doi.org/10.3762/bjoc.16.171).
- [18] Michael J. Hartmann and Giuseppe Carleo. "Neural-Network Approach to Dissipative Quantum Many-Body Dynamics". In: *Physical Review Letters* 122.25 (June 28, 2019), p. 250502. doi: [10.1103/PhysRevLett.122.250502](https://doi.org/10.1103/PhysRevLett.122.250502).
- [19] A. Zanella, C. M. Harrison, S. Lenzi, J. Cooke, P. Damsma, and S. W. Fleming. "Sonification and sound design for astronomy research, education and public engagement". In: *Nature Astronomy* 6.11 (Nov. 2022), pp. 1241–1248. doi: [10.1038/s41550-022-01721-z](https://doi.org/10.1038/s41550-022-01721-z).

- [20] Mario Krenn, Manuel Erhard, and Anton Zeilinger. "Computer-inspired quantum experiments". In: *Nature Reviews Physics* 2.11 (Nov. 2020), pp. 649–661. doi: [10.1038/s42254-020-0230-4](https://doi.org/10.1038/s42254-020-0230-4).
- [21] Zhaoyou Wang, Taha Rajabzadeh, Nathan Lee, and Amir H. Safavi-Naeini. "Automated Discovery of Autonomous Quantum Error Correction Schemes". In: *PRX Quantum* 3.2 (Apr. 4, 2022), p. 020302. doi: [10.1103/PRXQuantum.3.020302](https://doi.org/10.1103/PRXQuantum.3.020302).
- [22] Colin P. Williams and Alexander G. Gray. "Automated Design of Quantum Circuits". In: *Quantum Computing and Quantum Communications*. Ed. by Colin P. Williams. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 1999, pp. 113–125. doi: [10.1007/3-540-49208-9_8](https://doi.org/10.1007/3-540-49208-9_8).
- [23] Manuel Erhard, Mario Krenn, and Anton Zeilinger. "Advances in high-dimensional quantum entanglement". In: *Nature Reviews Physics* 2.7 (July 2020), pp. 365–381. doi: [10.1038/s42254-020-0193-5](https://doi.org/10.1038/s42254-020-0193-5).
- [24] Xiao-Min Hu, Yu Guo, Bi-Heng Liu, Chuan-Feng Li, and Guang-Can Guo. "Progress in quantum teleportation". In: *Nature Reviews Physics* 5.6 (June 2023), pp. 339–353. doi: [10.1038/s42254-023-00588-x](https://doi.org/10.1038/s42254-023-00588-x).
- [25] Lynden K. Shalm, Evan Meyer-Scott, Bradley G. Christensen, Peter Bierhorst, Michael A. Wayne, Martin J. Stevens, Thomas Gerrits, Scott Glancy, Deny R. Hamel, Michael S. Allman, Kevin J. Coakley, Shellee D. Dyer, Carson Hodge, Adriana E. Lita, Varun B. Verma, Camilla Lambrocco, Edward Tortorici, Alan L. Migdall, Yanbao Zhang, Daniel R. Kumor, William H. Farr, Francesco Marsili, Matthew D. Shaw, Jeffrey A. Stern, Carlos Abellán, Waldimar Amaya, Valerio Pruneri, Thomas Jennewein, Morgan W. Mitchell, Paul G. Kwiat, Joshua C. Bienfang, Richard P. Mirin, Emanuel Knill, and Sae Woo Nam. "Strong Loophole-Free Test of Local Realism". In: *Physical Review Letters* 115.25 (Dec. 16, 2015), p. 250402. doi: [10.1103/PhysRevLett.115.250402](https://doi.org/10.1103/PhysRevLett.115.250402).
- [26] Marissa Giustina, Marijn A. M. Versteegh, Sören Wengerowsky, Johannes Handsteiner, Armin Hochrainer, Kevin Phelan, Fabian Steinlechner, Johannes Kofler, Jan-Åke Larsson, Carlos Abellán, Waldimar Amaya, Valerio Pruneri, Morgan W. Mitchell, Jörn Beyer, Thomas Gerrits, Adriana E. Lita, Lynden K. Shalm, Sae Woo Nam, Thomas Scheidl, Rupert Ursin, Bernhard Wittmann, and Anton Zeilinger. "Significant-Loophole-Free Test of Bell's Theorem with Entangled Photons". In: *Physical Review Letters* 115.25 (Dec. 16, 2015), p. 250401. doi: [10.1103/PhysRevLett.115.250401](https://doi.org/10.1103/PhysRevLett.115.250401).
- [27] M. Żukowski, A. Zeilinger, M. A. Horne, and A. K. Ekert. "'Event-ready-detectors' Bell experiment via entanglement swapping". In: *Physical Review Letters* 71.26 (Dec. 27, 1993), pp. 4287–4290. doi: [10.1103/PhysRevLett.71.4287](https://doi.org/10.1103/PhysRevLett.71.4287).
- [28] Jian-Wei Pan, Dik Bouwmeester, Harald Weinfurter, and Anton Zeilinger. "Experimental Entanglement Swapping: Entangling Photons That Never Interacted". In: *Physical Review Letters* 80.18 (May 4, 1998), pp. 3891–3894. doi: [10.1103/PhysRevLett.80.3891](https://doi.org/10.1103/PhysRevLett.80.3891).
- [29] Davide Bacco, Jacob F. F. Bulmer, Manuel Erhard, Marcus Huber, and Stefano Paesani. "Proposal for practical multidimensional quantum networks". In: *Physical Review A* 104.5 (Nov. 23, 2021), p. 052618. doi: [10.1103/PhysRevA.104.052618](https://doi.org/10.1103/PhysRevA.104.052618).
- [30] Daniel M. Greenberger, Michael A. Horne, Abner Shimony, and Anton Zeilinger. "Bell's theorem without inequalities". In: *American Journal of Physics* 58.12 (Dec. 1, 1990), pp. 1131–1143. doi: [10.1119/1.16243](https://doi.org/10.1119/1.16243).
- [31] Jian-Wei Pan, Dik Bouwmeester, Matthew Daniell, Harald Weinfurter, and Anton Zeilinger. "Experimental test of quantum nonlocality in three-photon Greenberger–Horne–Zeilinger entanglement". In: *Nature* 403.6769 (Feb. 2000), pp. 515–519. doi: [10.1038/35000514](https://doi.org/10.1038/35000514).
- [32] S. Bose, V. Vedral, and P. L. Knight. "Multiparticle generalization of entanglement swapping". In: *Physical Review A* 57.2 (Feb. 1, 1998), pp. 822–829. doi: [10.1103/PhysRevA.57.822](https://doi.org/10.1103/PhysRevA.57.822).
- [33] Daniel Collins, Nicolas Gisin, Noah Linden, Serge Massar, and Sandu Popescu. "Bell Inequalities for Arbitrarily High-Dimensional Systems". In: *Physical Review Letters* 88.4 (Jan. 10, 2002), p. 040404. doi: [10.1103/PhysRevLett.88.040404](https://doi.org/10.1103/PhysRevLett.88.040404).
- [34] Feiran Wang, Manuel Erhard, Amin Babazadeh, Mehul Malik, Mario Krenn, and Anton Zeilinger. "Generation of the complete four-dimensional Bell basis". In: *Optica* 4.12 (Dec. 20, 2017), pp. 1462–1467. doi: [10.1364/OPTICA.4.001462](https://doi.org/10.1364/OPTICA.4.001462).
- [35] Manuel Erhard, Mehul Malik, Mario Krenn, and Anton Zeilinger. "Experimental Greenberger–Horne–Zeilinger entanglement beyond qubits". In: *Nature Photonics* 12.12 (Dec. 2018), pp. 759–764. doi: [10.1038/s41566-018-0257-6](https://doi.org/10.1038/s41566-018-0257-6).
- [36] Jaroslav Kysela, Manuel Erhard, Armin Hochrainer, Mario Krenn, and Anton Zeilinger. "Path identity as a source of high-dimensional entanglement". In: *Proceedings of the National Academy of Sciences* 117.42 (Oct. 20, 2020), pp. 26118–26122. doi: [10.1073/pnas.2011405117](https://doi.org/10.1073/pnas.2011405117).
- [37] Jueming Bao, Zhaorong Fu, Tanumoy Pramanik, Jun Mao, Yulin Chi, Yingkang Cao, Chonghao Zhai, Yifei Mao, Tianxiang Dai, Xiaojiong Chen, Xinyu Jia, Leshi Zhao, Yun Zheng, Bo Tang, Zhihua Li, Jun Luo, Wenwu Wang, Yan Yang, Yingying Peng, Dajian Liu, Daoxin Dai, Qiongyi He, Alif Laila Muthali, Leif K. Oxenløwe, Caterina Vigliar, Stefano Paesani, Huili Hou, Raffaele Santagati, Joshua W. Silverstone, Anthony Laing, Mark G. Thompson, Jeremy L. O'Brien, Yunhong Ding, Qihuang Gong, and Jianwei Wang. "Very-large-scale integrated quantum graph photonics". In: *Nature Photonics* 17.7 (July 2023), pp. 573–581. doi: [10.1038/s41566-023-01187-z](https://doi.org/10.1038/s41566-023-01187-z).

- [38] Wen-Bo Xing, Xiao-Min Hu, Yu Guo, Bi-Heng Liu, Chuan-Feng Li, and Guang-Can Guo. "Preparation of multiphoton high-dimensional GHZ states". In: *Optics Express* 31.15 (July 17, 2023), pp. 24887–24896. doi: [10.1364/OE.494850](https://doi.org/10.1364/OE.494850).
- [39] J. Calsamiglia and N. Lütkenhaus. "Maximum efficiency of a linear-optical Bell-state analyzer". In: *Applied Physics B* 72.1 (Jan. 1, 2001), pp. 67–71. doi: [10.1007/s003400000484](https://doi.org/10.1007/s003400000484).
- [40] John Calsamiglia. "Generalized measurements by linear elements". In: *Physical Review A* 65.3 (Feb. 5, 2002), p. 030301. doi: [10.1103/PhysRevA.65.030301](https://doi.org/10.1103/PhysRevA.65.030301).
- [41] Yi-Han Luo, Han-Sen Zhong, Manuel Erhard, Xi-Lin Wang, Li-Chao Peng, Mario Krenn, Xiao Jiang, Li Li, Nai-Le Liu, Chao-Yang Lu, Anton Zeilinger, and Jian-Wei Pan. "Quantum Teleportation in High Dimensions". In: *Physical Review Letters* 123.7 (Aug. 15, 2019), p. 070505. doi: [10.1103/PhysRevLett.123.070505](https://doi.org/10.1103/PhysRevLett.123.070505).
- [42] Xiao-Min Hu, Chao Zhang, Bi-Heng Liu, Yu Cai, Xiang-Jun Ye, Yu Guo, Wen-Bo Xing, Cen-Xiao Huang, Yun-Feng Huang, Chuan-Feng Li, and Guang-Can Guo. "Experimental High-Dimensional Quantum Teleportation". In: *Physical Review Letters* 125.23 (Dec. 2, 2020), p. 230501. doi: [10.1103/PhysRevLett.125.230501](https://doi.org/10.1103/PhysRevLett.125.230501).
- [43] John R. Stowers, Maximilian Hofbauer, Renaud Bastien, Johannes Griessner, Peter Higgins, Sarfarazhusain Farooqui, Ruth M. Fischer, Karin Nowikovsky, Wulf Haubensak, Iain D. Couzin, Kristin Tessmar-Raible, and Andrew D. Straw. "Virtual reality for freely moving animals". In: *Nature Methods* 14.10 (Oct. 2017), pp. 995–1002. doi: [10.1038/nmeth.4399](https://doi.org/10.1038/nmeth.4399).
- [44] Stefano Ferretti, Silvio Bianchi, Giacomo Frangipane, and Roberto Di Leonardo. "A virtual reality interface for the immersive manipulation of live microscopic systems". In: *Scientific Reports* 11.1 (Apr. 7, 2021), p. 7610. doi: [10.1038/s41598-021-87004-5](https://doi.org/10.1038/s41598-021-87004-5).
- [45] Rebecca K. Walters, Ella M. Gale, Jonathan Barnoud, David R. Glowacki, and Adrian J. Mulholland. "The emerging potential of interactive virtual reality in drug discovery". In: *Expert Opinion on Drug Discovery* 17.7 (July 3, 2022), pp. 685–698. doi: [10.1080/17460441.2022.2079632](https://doi.org/10.1080/17460441.2022.2079632).
- [46] David Kuřák, Pere-Pau Vázquez, Tobias Isenberg, Michael Krone, Marc Baaden, Jan Byška, Barbora Kozlíková, and Haichao Miao. "State of the Art of Molecular Visualization in Immersive Virtual Environments". In: *Computer Graphics Forum* 42.6 (2023), e14738. doi: [10.1111/cgf.14738](https://doi.org/10.1111/cgf.14738).
- [47] Ivan E. Sutherland. "The Ultimate Display". In: Proceedings of IFIP Congress. 1965.
- [48] Benjamin Rau, Florian Frieß, Michael Krone, Christoph Müller, and Thomas Ertl. "Enhancing visualization of molecular simulations using sonification". In: 2015 IEEE 1st International Workshop on Virtual and Augmented Reality for Molecular Science (VARMS@IEEEVR). Mar. 2015, pp. 25–30. doi: [10.1109/VARMS.2015.7151725](https://doi.org/10.1109/VARMS.2015.7151725).
- [49] Tifanie Bouchara and Matthieu Montès. "Immersive sonification of protein surface". In: 2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW). Mar. 2020, pp. 380–383. doi: [10.1109/VRW50115.2020.00082](https://doi.org/10.1109/VRW50115.2020.00082).
- [50] Dangxiao Wang, Yuan Guo, Shiyi Liu, Yuru Zhang, Weiliang Xu, and Jing Xiao. "Haptic display for virtual reality: progress and challenges". In: *Virtual Reality & Intelligent Hardware*. Haptic Interaction 1.2 (Apr. 1, 2019), pp. 136–162. doi: [10.3724/SP.J.2096-5796.2019.0008](https://doi.org/10.3724/SP.J.2096-5796.2019.0008).
- [51] Ivan E. Sutherland. "A head-mounted three dimensional display". In: *Proceedings of the December 9-11, 1968, fall joint computer conference, part 1*. AFIPS '68 (Fall, part 1). New York, NY, USA: Association for Computing Machinery, Dec. 9, 1968, pp. 757–764. doi: [10.1145/1476589.1476686](https://doi.org/10.1145/1476589.1476686).
- [52] Jie Yuan, Sohail S. Hassan, Jiaojiao Wu, Casey R. Koger, René R. Sevag Packard, Feng Shi, Baowei Fei, and Yichen Ding. "Extended reality for biomedicine". In: *Nature Reviews Methods Primers* 3.1 (Mar. 2, 2023), pp. 1–16. doi: [10.1038/s43586-023-00198-y](https://doi.org/10.1038/s43586-023-00198-y).
- [53] Carolina Cruz-Neira, Daniel J. Sandin, Thomas A. DeFanti, Robert V. Kenyon, and John C. Hart. "The CAVE: audio visual experience automatic virtual environment". In: *Communications of the ACM* 35.6 (June 1992), pp. 64–72. doi: [10.1145/129888.129892](https://doi.org/10.1145/129888.129892).
- [54] Frederick P. Brooks, Ming Ouh-Young, James J. Batter, and P. Jerome Kilpatrick. "Project GROPEHaptic displays for scientific visualization". In: *ACM SIGGRAPH Computer Graphics* 24.4 (Sept. 1, 1990), pp. 177–185. doi: [10.1145/97880.97899](https://doi.org/10.1145/97880.97899).
- [55] Elif Hilal Korkut and Elif Surer. "Visualization in virtual reality: a systematic review". In: *Virtual Reality* 27.2 (June 1, 2023), pp. 1447–1480. doi: [10.1007/s10055-023-00753-8](https://doi.org/10.1007/s10055-023-00753-8).
- [56] Mohamed El Beheiry, Sébastien Doutreligne, Clément Caporal, Cécilia Ostertag, Maxime Dahan, and Jean-Baptiste Masson. "Virtual Reality: Beyond Visualization". In: *Journal of Molecular Biology* 431.7 (Mar. 29, 2019), pp. 1315–1321. doi: [10.1016/j.jmb.2019.01.033](https://doi.org/10.1016/j.jmb.2019.01.033).
- [57] Caroline Stefani, Adam Lacy-Hulbert, and Thomas Skillman. "ConfocalVR: Immersive Visualization for Confocal Microscopy". In: *Journal of Molecular Biology* 430.21 (Oct. 19, 2018), pp. 4028–4035. doi: [10.1016/j.jmb.2018.06.035](https://doi.org/10.1016/j.jmb.2018.06.035).
- [58] Rensu P. Theart, Ben Loos, and Thomas R. Niesler. "Virtual reality assisted microscopy data visualization and colocalization analysis". In: *BMC Bioinformatics* 18.2 (Feb. 15, 2017), p. 64. doi: [10.1186/s12859-016-1446-2](https://doi.org/10.1186/s12859-016-1446-2).

- [59] Yimin Wang, Qi Li, Lijuan Liu, Zhi Zhou, Zongcai Ruan, Lingsheng Kong, Yaoyao Li, Yun Wang, Ning Zhong, Renjie Chai, Xiangfeng Luo, Yike Guo, Michael Hawrylycz, Qingming Luo, Zhongze Gu, Wei Xie, Hongkui Zeng, and Hanchuan Peng. "TeraVR empowers precise reconstruction of complete 3-D neuronal morphology in the whole brain". In: *Nature Communications* 10.1 (Aug. 2, 2019), p. 3474. doi: [10.1038/s41467-019-11443-y](https://doi.org/10.1038/s41467-019-11443-y).
- [60] Birte S. Steiniger, Henriette Pfeffer, Simone Gaffling, and Oleg Lobachev. "The human splenic microcirculation is entirely open as shown by 3D models in virtual reality". In: *Scientific Reports* 12.1 (Oct. 1, 2022), p. 16487. doi: [10.1038/s41598-022-19885-z](https://doi.org/10.1038/s41598-022-19885-z).
- [61] Thomas Blanc, Mohamed El Beheiry, Clément Caporal, Jean-Baptiste Masson, and Bassam Hajj. "Genuage: visualize and analyze multidimensional single-molecule point cloud data in virtual reality". In: *Nature Methods* 17.11 (Nov. 2020), pp. 1100–1102. doi: [10.1038/s41592-020-0946-1](https://doi.org/10.1038/s41592-020-0946-1).
- [62] Alexander Spark, Alexandre Kitching, Daniel Esteban-Ferrer, Anoushka Handa, Alexander R. Carr, Lisa-Maria Needham, Aleks Ponjavic, Ana Mafalda Santos, James McColl, Christophe Letierrier, Simon J. Davis, Ricardo Henriques, and Steven F. Lee. "vLUME: 3D virtual reality for single-molecule localization microscopy". In: *Nature Methods* 17.11 (Nov. 2020), pp. 1097–1099. doi: [10.1038/s41592-020-0962-1](https://doi.org/10.1038/s41592-020-0962-1).
- [63] Antoni Sagristà, Stefan Jordan, Thomas Müller, and Filip Sadlo. "Gaia Sky: Navigating the Gaia Catalog". In: *IEEE Transactions on Visualization and Computer Graphics* 25.1 (Jan. 2019), pp. 1070–1079. doi: [10.1109/TVCG.2018.2864508](https://doi.org/10.1109/TVCG.2018.2864508).
- [64] E. Baracaglia and F. P. A. Vogt. "E0102-VR: Exploring the scientific potential of Virtual Reality for observational astrophysics". In: *Astronomy and Computing* 30 (Jan. 1, 2020), p. 100352. doi: [10.1016/j.ascom.2019.100352](https://doi.org/10.1016/j.ascom.2019.100352).
- [65] Jordy Davelaar, Thomas Bronzwaer, Daniel Kok, Ziri Younsi, Monika Mościbrodzka, and Heino Falcke. "Observing supermassive black holes in virtual reality". In: *Computational Astrophysics and Cosmology* 5.1 (Nov. 19, 2018), p. 1. doi: [10.1186/s40668-018-0023-7](https://doi.org/10.1186/s40668-018-0023-7).
- [66] Will Usher, Pavol Klacansky, Frederick Federer, Peer-Timo Bremer, Aaron Knoll, Jeff Yarch, Alessandra Angelucci, and Valerio Pascucci. "A Virtual Reality Visualization Tool for Neuron Tracing". In: *IEEE Transactions on Visualization and Computer Graphics* 24.1 (Jan. 2018), pp. 994–1003. doi: [10.1109/TVCG.2017.2744079](https://doi.org/10.1109/TVCG.2017.2744079).
- [67] Rubén Jesús García-Hernández and Dieter Kranzlmüller. "NOMAD VR: Multiplatform virtual reality viewer for chemistry simulations". In: *Computer Physics Communications* 237 (Apr. 1, 2019), pp. 230–237. doi: [10.1016/j.cpc.2018.11.013](https://doi.org/10.1016/j.cpc.2018.11.013).
- [68] Kevin C. Cassidy, Jan Šefčík, Yogindra Raghav, Alexander Chang, and Jacob D. Durrant. "ProteinVR: Web-based molecular visualization in virtual reality". In: *PLoS Computational Biology* 16.3 (Mar. 31, 2020). Ed. by Dina Schneidman-Duhovny, e1007747. doi: [10.1371/journal.pcbi.1007747](https://doi.org/10.1371/journal.pcbi.1007747).
- [69] Kui Xu, Nan Liu, Jingle Xu, Chunlong Guo, Lingyun Zhao, Hong-Wei Wang, and Qiangfeng Cliff Zhang. "VRmol: an integrative web-based virtual reality system to explore macromolecular structure". In: *Bioinformatics* 37.7 (May 17, 2021), pp. 1029–1031. doi: [10.1093/bioinformatics/btaa696](https://doi.org/10.1093/bioinformatics/btaa696).
- [70] Laura J. Kingsley, Vincent Brunet, Gerald Lelais, Steve McCloskey, Kelly Milliken, Edgardo Leija, Stephen R. Fuhs, Kai Wang, Edward Zhou, and Glen Spraggon. "Development of a virtual reality platform for effective communication of structural data in drug discovery". In: *Journal of Molecular Graphics and Modelling* 89 (June 1, 2019), pp. 234–241. doi: [10.1016/j.jmgm.2019.03.010](https://doi.org/10.1016/j.jmgm.2019.03.010).
- [71] Michael B. O'Connor, Simon J. Bennie, Helen M. Deeks, Alexander Jamieson-Binnie, Alex J. Jones, Robin J. Shannon, Rebecca Walters, Thomas J. Mitchell, Adrian J. Mulholland, and David R. Glowacki. "Interactive molecular dynamics in virtual reality from quantum chemistry to drug binding: An open-source multi-person framework". In: *The Journal of Chemical Physics* 150.22 (June 10, 2019), p. 220901. doi: [10.1063/1.5092590](https://doi.org/10.1063/1.5092590).
- [72] Alex Zhavoronkov, Bogdan Zagribelnyy, Alexander Zhebrak, Vladimir Aladinskiy, Victor Terentiev, Quentin Vanhaelen, Dmitry S. Bezrukov, Daniil Polykovskiy, Rim Shayakhmetov, Andrey Filimonov, Michael Bishop, Steve McCloskey, Edgardo Leija, Deborah Bright, Keita Funakawa, Yen-Chu Lin, Shih-Hsien Huang, Hsuan-Jen Liao, Alex Aliper, and Yan Ivanenkov. *Potential Non-Covalent SARS-CoV-2 3C-like Protease Inhibitors Designed Using Generative Deep Learning Approaches and Reviewed by Human Medicinal Chemist in Virtual Reality*. May 19, 2020. doi: [10.26434/chemrxiv.12301457.v1](https://doi.org/10.26434/chemrxiv.12301457.v1).
- [73] Sebastian Pirch, Felix Müller, Eugenia Iofinova, Julia Pazmandi, Christiane V. R. Hütter, Martin Chiettoni, Celine Sin, Kaan Boztug, Iana Podkosova, Hannes Kaufmann, and Jörg Menche. "The VRNetzer platform enables interactive network analysis in Virtual Reality". In: *Nature Communications* 12.1 (Apr. 23, 2021), p. 2432. doi: [10.1038/s41467-021-22570-w](https://doi.org/10.1038/s41467-021-22570-w).
- [74] B. Laha, K. Sensharma, J. D. Schiffbauer, and D. A. Bowman. "Effects of Immersion on Visual Analysis of Volume Data". In: *IEEE Transactions on Visualization and Computer Graphics* 18.4 (Apr. 2012), pp. 597–606. doi: [10.1109/TVCG.2012.42](https://doi.org/10.1109/TVCG.2012.42).
- [75] C. Ware and G. Franck. "Viewing a graph in a virtual reality display is three times as good as a 2D diagram". In: *Proceedings of 1994 IEEE Symposium on Visual Languages*. Proceedings of 1994 IEEE Symposium on Visual Languages. Oct. 1994, pp. 182–183. doi: [10.1109/VL.1994.363621](https://doi.org/10.1109/VL.1994.363621).

- [76] Patrick Millais, Simon L. Jones, and Ryan Kelly. "Exploring Data in Virtual Reality: Comparisons with 2D Data Visualizations". In: *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18: CHI Conference on Human Factors in Computing Systems. Montreal QC Canada: ACM, Apr. 20, 2018, pp. 1–6. doi: [10.1145/3170427.3188537](https://doi.org/10.1145/3170427.3188537).
- [77] Benjamin Bach, Ronell Sicut, Johanna Beyer, Maxime Cordeil, and Hanspeter Pfister. "The Hologram in My Hand: How Effective is Interactive Exploration of 3D Visualizations in Immersive Tangible Augmented Reality?" In: *IEEE Transactions on Visualization and Computer Graphics* 24.1 (Jan. 2018), pp. 457–467. doi: [10.1109/TVCG.2017.2745941](https://doi.org/10.1109/TVCG.2017.2745941).
- [78] Michael O'Connor, Helen M. Deeks, Edward Dawn, Oussama Metatla, Anne Roudaut, Matthew Sutton, Lisa May Thomas, Becca Rose Glowacki, Rebecca Sage, Philip Tew, Mark Wonnacott, Phil Bates, Adrian J. Mulholland, and David R. Glowacki. "Sampling molecular conformations and dynamics in a multiuser virtual reality framework". In: *Science Advances* 4.6 (June 29, 2018), eaat2731. doi: [10.1126/sciadv.aat2731](https://doi.org/10.1126/sciadv.aat2731).
- [79] C. D. Porter, J. R. H. Smith, E. M. Stagar, A. Simmons, M. Nieberding, C. M. Orban, J. Brown, and A. Ayers. "Using virtual reality in electrostatics instruction: The impact of training". In: *Physical Review Physics Education Research* 16.2 (Sept. 3, 2020), p. 020119. doi: [10.1103/PhysRevPhysEducRes.16.020119](https://doi.org/10.1103/PhysRevPhysEducRes.16.020119).
- [80] Dörte Sonntag and Oliver Bodensiek. "How mixed reality shifts visual attention and success in experimental problem solving". In: *Physical Review Physics Education Research* 18.2 (July 21, 2022), p. 023101. doi: [10.1103/PhysRevPhysEducRes.18.023101](https://doi.org/10.1103/PhysRevPhysEducRes.18.023101).
- [81] P. Schlummer, A. Abazi, R. Borkamp, J. Lauströer, R. Schulz-Schaeffer, C. Schuck, W. Pernice, S. Heusler, and D. Laumann. "Seeing the unseen—enhancing and evaluating undergraduate polarization experiments with interactive Mixed-Reality technology". In: *European Journal of Physics* 44.6 (Sept. 2023), p. 065701. doi: [10.1088/1361-6404/acf0a7](https://doi.org/10.1088/1361-6404/acf0a7).
- [82] Jonas R. Schmid, Moritz J. Ernst, and Günther Thiele. "Structural Chemistry 2.0: Combining Augmented Reality and 3D Online Models". In: *Journal of Chemical Education* 97.12 (Dec. 8, 2020), pp. 4515–4519. doi: [10.1021/acs.jchemed.0c00823](https://doi.org/10.1021/acs.jchemed.0c00823).
- [83] André Lanrezac, Nicolas Férey, and Marc Baaden. "Wielding the power of interactive molecular simulations". In: *WIREs Computational Molecular Science* 12.4 (2022), e1594. doi: [10.1002/wcms.1594](https://doi.org/10.1002/wcms.1594).
- [84] Ulrik Günther, Kyle I. S. Harrington, Raimund Dachsel, and Ivo F. Sbalzarini. "Bionic Tracking: Using Eye Tracking to Track Biological Cells in Virtual Reality". In: *Computer Vision – ECCV 2020 Workshops*. Ed. by Adrien Bartoli and Andrea Fusiello. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2020, pp. 280–297. doi: [10.1007/978-3-030-66415-2_18](https://doi.org/10.1007/978-3-030-66415-2_18).
- [85] Silvia Amabilino, Lars A. Bratholm, Simon J. Bennie, Alain C. Vaucher, Markus Reiher, and David R. Glowacki. "Training Neural Nets To Learn Reactive Potential Energy Surfaces Using Interactive Quantum Chemistry in Virtual Reality". In: *The Journal of Physical Chemistry A* 123.20 (May 23, 2019), pp. 4486–4499. doi: [10.1021/acs.jpca.9b01006](https://doi.org/10.1021/acs.jpca.9b01006).
- [86] Silvia Amabilino, Lars A. Bratholm, Simon J. Bennie, Michael B. O'Connor, and David R. Glowacki. "Training atomic neural networks using fragment-based data generated in virtual reality". In: *The Journal of Chemical Physics* 153.15 (Oct. 15, 2020), p. 154105. doi: [10.1063/5.0015950](https://doi.org/10.1063/5.0015950).
- [87] Martin Bellgardt, Christian Scheiderer, and Torsten W. Kuhlen. "An Immersive Node-Link Visualization of Artificial Neural Networks for Machine Learning Experts". In: 2020 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR). Dec. 2020, pp. 33–36. doi: [10.1109/AIVR50618.2020.00015](https://doi.org/10.1109/AIVR50618.2020.00015).
- [88] Zhuoyue Lyu, Jiannan Li, and Bryan Wang. "Alive: Interactive Visualization and Sonification of Neural Networks in Virtual Reality". In: 2021 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR). Nov. 2021, pp. 251–255. doi: [10.1109/AIVR52153.2021.00057](https://doi.org/10.1109/AIVR52153.2021.00057).
- [89] Časlav Brukner, Marek Żukowski, and Anton Zeilinger. "The Essence of Entanglement". In: *Quantum Arrangements: Contributions in Honor of Michael Horne*. Ed. by Gregg Jaeger, David Simon, Alexander V. Sergienko, Daniel Greenberger, and Anton Zeilinger. Cham: Springer International Publishing, 2021, pp. 117–138. doi: [10.1007/978-3-030-77367-0_6](https://doi.org/10.1007/978-3-030-77367-0_6).
- [90] Werner Vogel, Dirk-Gunnar Welsch, and Sascha Wallentowitz. *Quantum Optics An Introduction*. Berlin: WILEY-VCH Verlag Berlin, 2001.
- [91] A. Einstein, B. Podolsky, and N. Rosen. "Can Quantum-Mechanical Description of Physical Reality Be Considered Complete?" In: *Physical Review* 47.10 (May 15, 1935), pp. 777–780. doi: [10.1103/PhysRev.47.777](https://doi.org/10.1103/PhysRev.47.777).
- [92] J. S. Bell. "On the Einstein Podolsky Rosen paradox". In: *Physics Physique Fizika* 1.3 (Nov. 1, 1964), pp. 195–200. doi: [10.1103/PhysicsPhysiqueFizika.1.195](https://doi.org/10.1103/PhysicsPhysiqueFizika.1.195).
- [93] John F. Clauser, Michael A. Horne, Abner Shimony, and Richard A. Holt. "Proposed Experiment to Test Local Hidden-Variable Theories". In: *Physical Review Letters* 23.15 (Oct. 13, 1969), pp. 880–884. doi: [10.1103/PhysRevLett.23.880](https://doi.org/10.1103/PhysRevLett.23.880).

- [94] Anton Zeilinger. "A Foundational Principle for Quantum Mechanics". In: *Foundations of Physics* 29.4 (Apr. 1, 1999), pp. 631–643. doi: [10.1023/A:1018820410908](https://doi.org/10.1023/A:1018820410908).
- [95] Āslav Brukner and Anton Zeilinger. "Operationally Invariant Information in Quantum Measurements". In: *Physical Review Letters* 83.17 (Oct. 25, 1999), pp. 3354–3357. doi: [10.1103/PhysRevLett.83.3354](https://doi.org/10.1103/PhysRevLett.83.3354).
- [96] Junghee Ryu, Changhyoung Lee, Marek Źukowski, and Jinhyoung Lee. "Greenberger-Horne-Zeilinger theorem for N qudits". In: *Physical Review A* 88.4 (Oct. 2, 2013), p. 042101. doi: [10.1103/PhysRevA.88.042101](https://doi.org/10.1103/PhysRevA.88.042101).
- [97] Jay Lawrence. "Rotational covariance and Greenberger-Horne-Zeilinger theorems for three or more particles of any dimension". In: *Physical Review A* 89.1 (Jan. 9, 2014), p. 012105. doi: [10.1103/PhysRevA.89.012105](https://doi.org/10.1103/PhysRevA.89.012105).
- [98] C. K. Hong, Z. Y. Ou, and L. Mandel. "Measurement of subpicosecond time intervals between two photons by interference". In: *Physical Review Letters* 59.18 (Nov. 2, 1987), pp. 2044–2046. doi: [10.1103/PhysRevLett.59.2044](https://doi.org/10.1103/PhysRevLett.59.2044).
- [99] Charles H. Bennett, Gilles Brassard, Claude Cr epeau, Richard Jozsa, Asher Peres, and William K. Wootters. "Teleporting an unknown quantum state via dual classical and Einstein-Podolsky-Rosen channels". In: *Physical Review Letters* 70.13 (Mar. 29, 1993), pp. 1895–1899. doi: [10.1103/PhysRevLett.70.1895](https://doi.org/10.1103/PhysRevLett.70.1895).
- [100] Anders J. E. Bjerrum, Jonatan B. Brask, Jonas S. Neergaard-Nielsen, and Ulrik L. Andersen. "Proposal for a long-distance nonlocality test with entanglement swapping and displacement-based measurements". In: *Physical Review A* 107.5 (May 23, 2023), p. 052611. doi: [10.1103/PhysRevA.107.052611](https://doi.org/10.1103/PhysRevA.107.052611).
- [101] Sofiane Merkouche, Val erian Thiel, Alex O. C. Davis, and Brian J. Smith. "Heralding Multiple Photonic Pulsed Bell Pairs via Frequency-Resolved Entanglement Swapping". In: *Physical Review Letters* 128.6 (Feb. 11, 2022), p. 063602. doi: [10.1103/PhysRevLett.128.063602](https://doi.org/10.1103/PhysRevLett.128.063602).
- [102] Mario Krenn, Armin Hochrainer, Mayukh Lahiri, and Anton Zeilinger. "Entanglement by Path Identity". In: *Physical Review Letters* 118.8 (Feb. 23, 2017), p. 080401. doi: [10.1103/PhysRevLett.118.080401](https://doi.org/10.1103/PhysRevLett.118.080401).
- [103] Armin Hochrainer, Mayukh Lahiri, Manuel Erhard, Mario Krenn, and Anton Zeilinger. "Quantum indistinguishability by path identity and with undetected photons". In: *Reviews of Modern Physics* 94.2 (June 21, 2022), p. 025007. doi: [10.1103/RevModPhys.94.025007](https://doi.org/10.1103/RevModPhys.94.025007).
- [104] X. Y. Zou, L. J. Wang, and L. Mandel. "Induced coherence and indistinguishability in optical interference". In: *Physical Review Letters* 67.3 (July 15, 1991), pp. 318–321. doi: [10.1103/PhysRevLett.67.318](https://doi.org/10.1103/PhysRevLett.67.318).
- [105] Lucien Hardy. "Source of photons with correlated polarisations and correlated directions". In: *Physics Letters A* 161.4 (Jan. 6, 1992), pp. 326–328. doi: [10.1016/0375-9601\(92\)90554-Y](https://doi.org/10.1016/0375-9601(92)90554-Y).
- [106] Paul G. Kwiat, Edo Waks, Andrew G. White, Ian Appelbaum, and Philippe H. Eberhard. "Ultrabright source of polarization-entangled photons". In: *Physical Review A* 60.2 (Aug. 1, 1999), R773–R776. doi: [10.1103/PhysRevA.60.R773](https://doi.org/10.1103/PhysRevA.60.R773).
- [107] Gabriela Barreto Lemos, Victoria Borish, Garrett D. Cole, Sven Ramelow, Radek Lapkiewicz, and Anton Zeilinger. "Quantum imaging with undetected photons". In: *Nature* 512.7515 (Aug. 2014), pp. 409–412. doi: [10.1038/nature13586](https://doi.org/10.1038/nature13586).
- [108] Dmitry A. Kalashnikov, Anna V. Paterova, Sergei P. Kulik, and Leonid A. Krivitsky. "Infrared spectroscopy with visible light". In: *Nature Photonics* 10.2 (Feb. 2016), pp. 98–101. doi: [10.1038/nphoton.2015.252](https://doi.org/10.1038/nphoton.2015.252).
- [109] Inna Kviatkovsky, Helen M. Chrzanowski, and Sven Ramelow. "Mid-infrared microscopy via position correlations of undetected photons". In: *Optics Express* 30.4 (Feb. 14, 2022), pp. 5916–5925. doi: [10.1364/OE.440534](https://doi.org/10.1364/OE.440534).
- [110] Mario Krenn, Xuemei Gu, and Anton Zeilinger. "Quantum Experiments and Graphs: Multiparty States as Coherent Superpositions of Perfect Matchings". In: *Physical Review Letters* 119.24 (Dec. 15, 2017), p. 240403. doi: [10.1103/PhysRevLett.119.240403](https://doi.org/10.1103/PhysRevLett.119.240403).
- [111] Xuemei Gu, Manuel Erhard, Anton Zeilinger, and Mario Krenn. "Quantum experiments and graphs II: Quantum interference, computation, and state generation". In: *Proceedings of the National Academy of Sciences* 116.10 (Mar. 5, 2019), pp. 4147–4155. doi: [10.1073/pnas.1815884116](https://doi.org/10.1073/pnas.1815884116).
- [112] Xuemei Gu, Lijun Chen, Anton Zeilinger, and Mario Krenn. "Quantum experiments and graphs. III. High-dimensional and multiparticle entanglement". In: *Physical Review A* 99.3 (Mar. 25, 2019), p. 032338. doi: [10.1103/PhysRevA.99.032338](https://doi.org/10.1103/PhysRevA.99.032338).
- [113] Kaiyi Qian, Kai Wang, Leizhen Chen, Zhaohua Hou, Mario Krenn, Shining Zhu, and Xiao-song Ma. "Multiphoton non-local quantum interference controlled by an undetected photon". In: *Nature Communications* 14.1 (Mar. 17, 2023), p. 1480. doi: [10.1038/s41467-023-37228-y](https://doi.org/10.1038/s41467-023-37228-y).
- [114] Lan-Tian Feng, Ming Zhang, Di Liu, Yu-Jie Cheng, Guo-Ping Guo, Dao-Xin Dai, Guang-Can Guo, Mario Krenn, and Xi-Feng Ren. "On-chip quantum interference between the origins of a multi-photon state". In: *Optica* 10.1 (Jan. 20, 2023), pp. 105–109. doi: [10.1364/OPTICA.474750](https://doi.org/10.1364/OPTICA.474750).

- [115] Chao Zhang, Yun-Feng Huang, Bi-Heng Liu, Chuan-Feng Li, and Guang-Can Guo. "Spontaneous Parametric Down-Conversion Sources for Multiphoton Experiments". In: *Advanced Quantum Technologies* 4.5 (2021), p. 2000132. doi: [10.1002/qute.202000132](https://doi.org/10.1002/qute.202000132).
- [116] A. V. Belinskii and D. N. Klyshko. "Two-photon optics: diffraction, holography, and transformation of two-dimensional signals". In: *Soviet Journal of Experimental and Theoretical Physics* 78 (Mar. 1, 1994), pp. 259–262.
- [117] Min Jiang, Shunlong Luo, and Shuangshuang Fu. "Channel-state duality". In: *Physical Review A* 87.2 (Feb. 13, 2013), p. 022310. doi: [10.1103/PhysRevA.87.022310](https://doi.org/10.1103/PhysRevA.87.022310).
- [118] Tomihisa Kamada and Satoru Kawai. "An Algorithm for Drawing General undirected Graphs". In: *INFORMATION PROCESSING LETTERS* 31.1 (1989).
- [119] Gabor Csardi and Tamas Nepusz. "The Igraph Software Package for Complex Network Research". In: *InterJournal Complex Systems* (Nov. 30, 2005), p. 1695.
- [120] Ian Affleck, Tom Kennedy, Elliott H. Lieb, and Hal Tasaki. "Valence Bond Ground States in Isotropic Quantum Antiferromagnets". In: *Condensed Matter Physics and Exactly Soluble Models: Selecta of Elliott H. Lieb*. Ed. by Bruno Nachtergaele, Jan Philip Solovej, and Jakob Yngvason. Berlin, Heidelberg: Springer, 2004, pp. 253–304. doi: [10.1007/978-3-662-06390-3_19](https://doi.org/10.1007/978-3-662-06390-3_19).
- [121] Artificial Scientist Lab. [github/artificial-scientist-lab/PyTheus](https://github.com/artificial-scientist-lab/PyTheus). Jan. 25, 2024. URL: <https://github.com/artificial-scientist-lab/PyTheus> (visited on 03/09/2024).
- [122] Ye Yeo and Wee Kang Chua. "Teleportation and Dense Coding with Genuine Multipartite Entanglement". In: *Physical Review Letters* 96.6 (Feb. 15, 2006), p. 060502. doi: [10.1103/PhysRevLett.96.060502](https://doi.org/10.1103/PhysRevLett.96.060502).
- [123] Jian-wei Pan and Anton Zeilinger. "Greenberger-Horne-Zeilinger-state analyzer". In: *Physical Review A* 57.3 (Mar. 1, 1998), pp. 2208–2211. doi: [10.1103/PhysRevA.57.2208](https://doi.org/10.1103/PhysRevA.57.2208).
- [124] Jun Qian, Xun-Li Feng, and Shang-Qing Gong. "Universal Greenberger-Horne-Zeilinger-state analyzer based on two-photon polarization parity detection". In: *Physical Review A* 72.5 (Nov. 8, 2005), p. 052308. doi: [10.1103/PhysRevA.72.052308](https://doi.org/10.1103/PhysRevA.72.052308).
- [125] Jakob S. Kottmann. "Molecular Quantum Circuit Design: A Graph-Based Approach". In: *Quantum* 7 (Aug. 3, 2023), p. 1073. doi: [10.22331/q-2023-08-03-1073](https://doi.org/10.22331/q-2023-08-03-1073).
- [126] Ross Duncan, Aleks Kissinger, Simon Perdrix, and John van de Wetering. "Graph-theoretic Simplification of Quantum Circuits with the ZX-calculus". In: *Quantum* 4 (June 4, 2020), p. 279. doi: [10.22331/q-2020-06-04-279](https://doi.org/10.22331/q-2020-06-04-279).

A Discovered Graphs

For transparency, this section lists all graphs, that constructed or discovered in the results of this thesis. The format is chosen compatible with the PyTheus Library. I.e. a dictionary of edge entries with (vertex 0, vertex 1, color 0, color 1): weight.

A.1 3-particle-7-dimensional GHZ-state

7 dimensional 3 particle GHZ-state graph of Figure 17(b).

```
{(0, 1, 3, 3): (-1),
(0, 2, 1, 1): 1,
(0, 3, 2, 0): 1,
(0, 4, 0, 0): 1,
(0, 5, 4, 0): 1,
(0, 7, 2, 0): 1,
(1, 2, 4, 4): 1,
(1, 4, 2, 0): -1,
(1, 5, 1, 0): 1,
(1, 6, 0, 0): -1,
(2, 3, 0, 0): 1,
(2, 5, 3, 0): -1,
(2, 6, 2, 0): 1,
(2, 7, 0, 0): 1,
(3, 4, 0, 0): 1,
(3, 5, 0, 0): 1,
(3, 6, 0, 0): 1,
(4, 7, 0, 0): -1,
(5, 7, 0, 0): 1,
(6, 7, 0, 0): -1,
(8, 1, 0, 5): -1,
(8, 2, 0, 6): 1,
(8, 10, 0, 0): 1,
(9, 0, 0, 5): 1,
(9, 1, 0, 6): -1,
(9, 11, 0, 0): -1,
(10, 0, 0, 6): 1,
(10, 2, 0, 5): 1,
(10, 5, 0, 0): 1,
(10, 9, 0, 0): 1,
(11, 0, 0, 6): 1,
(11, 2, 0, 5): 1,
(11, 5, 0, 0): 1,
(11, 8, 0, 0): -1}
```

A.2 2dim.-GHZ-analyzer search:

A.2.1 Search Geometry

99 is the mode number used to indicate that an edge is representative of full connectivity. (i.e. black edge colors in AriadneVR). Note that the edge weights are irrelevant here, as this is the initial starting point of an optimization during which edge weights are randomly initialized.

```
{(0, 4, 0, 0):1.0,
(0, 4, 1, 0):1.0,
(0, 5, 0, 0):1.0,
(0, 6, 1, 0):1.0,
(1, 4, 0, 0):1.0,
(1, 4, 1, 0):1.0,
(1, 5, 0, 0):1.0,
(1, 6, 1, 0):1.0,
(2, 4, 0, 0):1.0,
(2, 4, 1, 0):1.0,
(2, 5, 1, 0):1.0,
(2, 6, 0, 0):1.0,
(3, 7, 99, 99):1.0,
(1, 7, 99, 99):1.0,
(0, 7, 99, 99):1.0,
(2, 7, 99, 99):1.0,
(3, 5, 99, 99):1.0,
(3, 6, 99, 99):1.0,
(3, 4, 99, 99):1.0}
```

A.2.2 12-edge analyzer

See Figure 20(c).

```
{(0, 5, 0, 0): 1.0,
(0, 6, 1, 0): -1.0,
(0, 7, 1, 0): -1.0,
(1, 4, 0, 0): 1.0,
(1, 6, 1, 0): 1.0,
(1, 7, 1, 0): 1.0,
(2, 5, 1, 0): 1.0,
(2, 6, 0, 0): 1.0,
(2, 7, 0, 0): -1.0,
(3, 4, 1, 0): -1.0,
(3, 6, 0, 0): -1.0,
(3, 7, 0, 0): 1.0}
```

A.2.3 20-edge analyzer

See Figure 20(d).

```
{(0, 4, 1, 0): -1.0,
(0, 5, 0, 0): 1.0,
(0, 6, 1, 0): 1.0,
(0, 7, 0, 0): 1.0,
(1, 4, 0, 0): 1.0,
(1, 5, 0, 0): -1.0,
(1, 6, 1, 0): -1.0,
(1, 7, 1, 0): -1.0,
(2, 4, 0, 0): 1.0,
(2, 5, 1, 0): 1.0,
(2, 6, 0, 0): 1.0,
(2, 7, 1, 0): -1.0,
(3, 4, 0, 0): -1.0,
(3, 4, 1, 0): -1.0,
(3, 5, 0, 0): 1.0,
(3, 5, 1, 0): -1.0,
(3, 6, 0, 0): -1.0,
(3, 6, 1, 0): 1.0,
(3, 7, 0, 0): 1.0,
(3, 7, 1, 0): 1.0}
```

A.3 3dim.-GHZ-analyzer search:

A.3.1 Search Geometry:

99 represents full allowed connectivity. Note, that due to this being a measurement graph with input nodes 0-3 additional restrictions apply when turning this search geometry into the initial graph of 74 edges. Edge weights are meaningless for initial search geometries.

```
{(0, 4, 99, 99): 1.0,
(0, 7, 99, 99): 1.0,
(0, 9, 99, 99): 1.0,
(0, 11, 99, 99): 1.0,
(1, 4, 99, 99): 1.0,
(1, 5, 99, 99): 1.0,
(1, 7, 99, 99): 1.0,
(1, 8, 99, 99): 1.0,
(1, 9, 99, 99): 1.0,
(1, 10, 99, 99): 1.0,
(1, 11, 99, 99): 1.0,
(2, 5, 99, 99): 1.0,
(2, 6, 99, 99): 1.0,
(2, 7, 99, 99): 1.0,
(2, 8, 99, 99): 1.0,
(2, 9, 99, 99): 1.0,
(2, 10, 99, 99): 1.0,
(2, 11, 99, 99): 1.0,
(3, 6, 99, 99): 1.0,
(3, 8, 99, 99): 1.0,
(3, 10, 99, 99): 1.0,
(3, 11, 99, 99): 1.0,
(4, 5, 99, 99): 1.0,
(5, 6, 99, 99): 1.0,
(7, 8, 99, 99): 1.0,
(7, 11, 99, 99): 1.0,
(8, 11, 99, 99): 1.0,
(9, 10, 99, 99): 1.0,
(9, 11, 99, 99): 1.0,
(10, 11, 99, 99): 1.0}
```

A.3.2 4-particle 3-dim. GHZ-state analyzer

Selected successful search result for the 4-particle 3d-analyzer. (see. Fig. 21)

```
{(0, 4, 2, 0): 1.0,  
(0, 7, 1, 0): 1.0,  
(0, 9, 0, 0): 1.0,  
(1, 11, 1, 0): -1.0,  
(1, 4, 0, 0): -1.0,  
(1, 5, 2, 0): -1.0,  
(1, 8, 1, 0): 1.0,  
(2, 10, 0, 0): -1.0,  
(2, 10, 2, 0): 1.0,  
(2, 11, 0, 0): -1.0,  
(2, 6, 1, 0): 1.0,  
(2, 8, 2, 0): -1.0,  
(3, 10, 0, 0): -1.0,  
(3, 11, 0, 0): -1.0,  
(3, 11, 1, 0): 1.0,  
(3, 6, 2, 0): 1.0,  
(3, 8, 1, 0): -1.0,  
(4, 5, 0, 0): -1.0,  
(5, 6, 0, 0): -1.0,  
(7, 11, 0, 0): -1.0,  
(7, 8, 0, 0): -1.0,  
(9, 10, 0, 0): 1.0,  
(9, 11, 0, 0): -1.0}
```

A.4 Semi-3d-entanglement-swapping

See Figure 29(b) Note that the four non-local photon pairs of this state are at vertices 0,3 1,4 2,5 and 10,12. 0,3 is the 3-dimensional pair. 1,4 and 2,5 are in $|00\rangle + |11\rangle$ and 10,12 in $|11\rangle + |22\rangle$. This is due to this graph being drawn by hand on top of a 3-pair graph before convenient vertex relabeling was implemented in AriadneVR.

```
{(0, 1, 0, 1): 1,  
(0, 2, 1, 1): 1,  
(0, 7, 0, 0): 1,  
(0, 9, 1, 0): 1,  
(0, 12, 2, 1): 1,  
(1, 6, 0, 0): 1,  
(1, 7, 1, 0): 1,  
(2, 8, 0, 0): 1,  
(2, 9, 1, 0): 1,  
(3, 4, 0, 0): 1,  
(3, 5, 1, 0): 1,  
(3, 7, 0, 0): 1,  
(3, 10, 2, 2): 1,  
(3, 11, 2, 0): 1,  
(4, 6, 1, 0): 1,  
(4, 7, 0, 0): 1,  
(8, 5, 0, 1): 1,  
(9, 3, 0, 1): 1,  
(9, 5, 0, 0): 1,  
(10, 11, 2, 0): 1,  
(10, 13, 1, 0): 1,  
(11, 0, 0, 2): 1,  
(12, 11, 1, 0): 1,  
(13, 12, 0, 2): 1}
```

A.5 4-pair-3dimensional entanglement swapping:

See Figure 30.

```
{(0, 2, 1, 1): 1,  
(0, 10, 2, 0): 1,  
(0, 11, 0, 0): 1,  
(0, 16, 1, 0): 1,  
(1, 10, 0, 0): 1,  
(1, 11, 2, 0): 1,  
(1, 17, 1, 0): 1,  
(2, 17, 1, 0): 1,  
(3, 5, 0, 0): 1,  
(3, 16, 1, 0): 1,  
(7, 3, 2, 2): 1,  
(8, 3, 0, 0): 1,  
(8, 4, 0, 0): 1,  
(8, 9, 0, 0): 1,  
(9, 2, 0, 0): 1,  
(9, 5, 0, 0): 1,  
(10, 11, 0, 0): 1,
```

```

(12, 4, 0, 2): 1,
(12, 5, 0, 1): 1,
(13, 4, 0, 1): 1,
(13, 5, 0, 2): 1,
(13, 12, 0, 0): 1,
(14, 3, 0, 2): 1,
(14, 6, 0, 2): 1,
(14, 15, 0, 0): 1,
(15, 2, 0, 2): 1,
(15, 7, 0, 2): 1,
(16, 17, 0, 0): 1,
(18, 6, 0, 1): 1,
(18, 7, 0, 0): 1,
(18, 19, 0, 0): 1,
(19, 6, 0, 0): 1,
(19, 7, 0, 1): 1}

```

B AriadneVR Documentation

Here, a brief overview over the software structure of AriadneVR is provided. As mentioned in section 4, AriadneVR is a Javascript application built in A-Frame and Three.js. A-Frame code is structured into script components, that are attached to HTML entities. Three.js is used as the underlying 3D library responsible for displaying the scene. AriadneVR is a collection of custom components together with an HTML template, as well as a JavaScript-port of the algorithm for computing perfect matchings included in the PyTheus library. Figure 32 shows the control-scheme for the software.

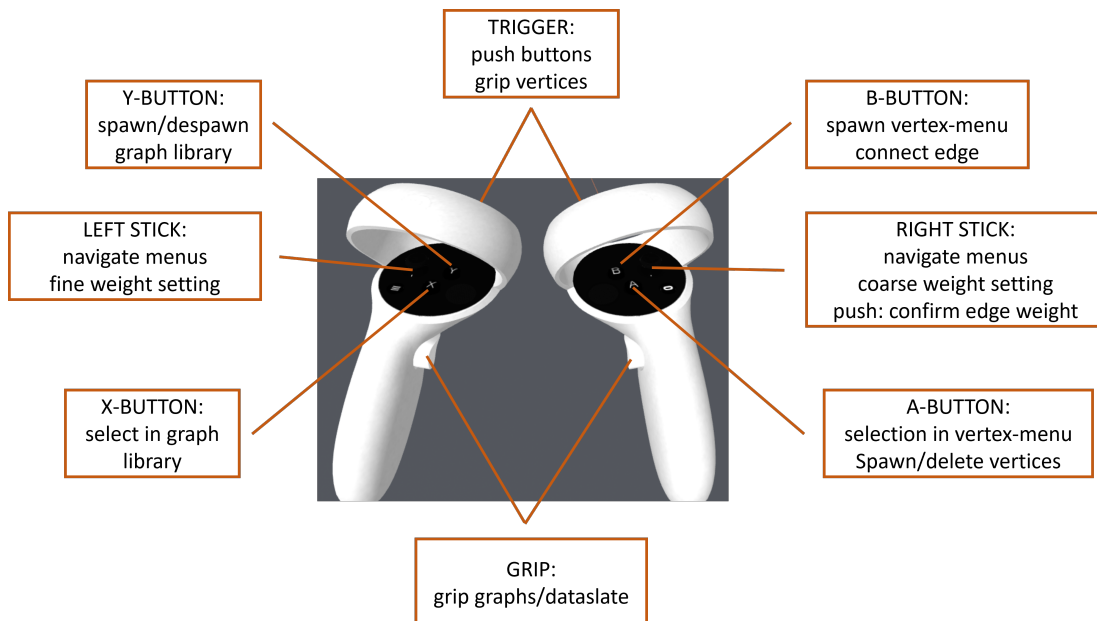


Figure 32: *AriadneVR control-scheme.*

A-Frame and related open-source community components are used as specified in the Readme on GitHub¹⁰. See the A-Frame documentation¹¹. for further detail on component structure GitHub Copilot¹², an AI-based coding assistant was used

¹⁰<https://github.com/artificial-scientist-lab/AriadneVR>

¹¹<https://aframe.io/docs/1.4.0/introduction/>

¹²<https://docs.github.com/de/copilot/copilot-individual/about-github-copilot-individual>

during development for faster auto-completion and generation of documentation strings. Figure 33 shows a schematic overview of the code structure.

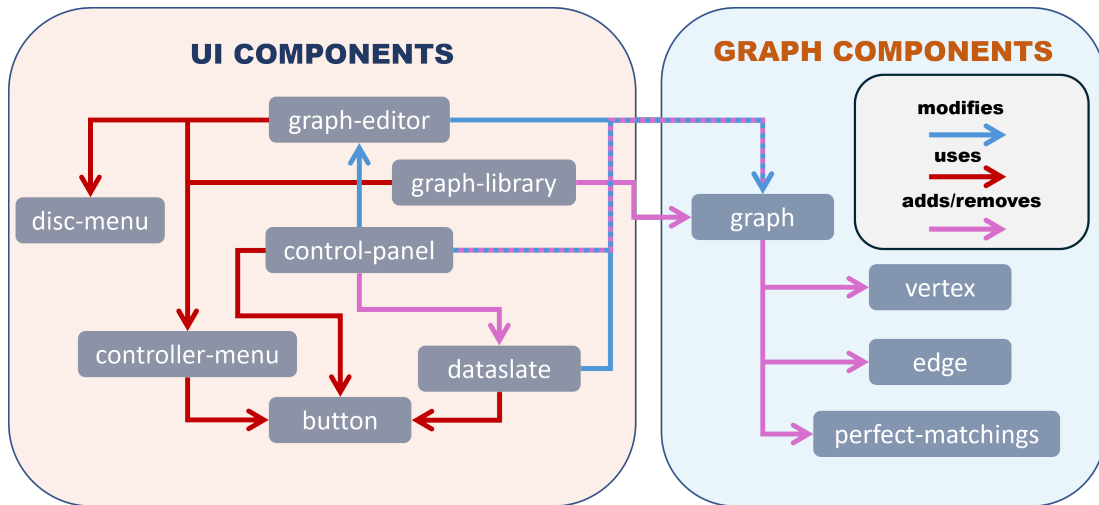


Figure 33: Schematic of the component interactions of the custom components of AriadneVR

B.1 UI-components

This section groups UI components from low-level to high-level.

B.1.1 button

Description: The Button component can be attached to entities with a geometry to turn them into virtual buttons, used in menus, panels, and slates. It is attached by higher-level components. Buttons can be pushed by the API (in menus) or by controllers via raycaster-interactions. The component controls the physical feedback of the button geometry element in color and position

Schema:

Property	Type	Default	Description
event	string	"none"	event the button emits on click.
clickEvent	string	"triggerdown"	event the button listens for to clicked.
clickable	boolean	false	Flag deciding whether the button is API pressed or Physical.
springLoaded	boolean	false	Flag deciding whether the button stays pushed or not.
raycastable	boolean	false	Flag turning the button raycastable.

Events:

Event	Detail		Description
schema.event	button:	component-instance	Event fired on button activation. Event title set by schema.
	name:	button entity id	
	pushed:	button status	

Listeners:

Event	Listener-Description
"raycaster-intersected"	Set when the button is set clickable, adds the "hovered"-state and changes the color of the button to indicate the hovering of the caster.
"raycaster-intersected-cleared"	Set when the button is set clickable, resets the action of the prev. listener or indicates the "pushed" state via coloring.
schema.clickEvent	Listener label set by schema, Emits the event of the button and causes physical feedback of the button if applicable.

B.1.2 controller-menu

Description: The controller menu is a collection of buttons on a 2D panel attached to controller models. It is navigated with a selector and supports 1 layer of submenus. The selector is navigated with the joystick and the menu is spawned and despawned with custom buttons.

Schema:

Property	Type	Default	Description
buttons	string	-	List of first layer buttons of the menu
subMenus	string	"[]"	List of lists, submenus of the menu
spawnKey	string	"abuttondown"	Key for spawning/despawning the menu
actionKey	string	"bbuttondown"	Key to press buttons/spawn submenus

Events:

Event	Detail		Description
"spawnkeypressed"	name:	main	Event fired when the custom spawn key set in the schema is pressed.

Listeners:

Event	Listener-Description
schema.spawnKey	Triggers toggling the menu.
"spawnkeypressed"	Toggles the menu.
schema.actionKey	Presses buttons/spawns submenus.
"thumbstickmoved"	Moves the selector across menus.

The component also adds Listeners for each button representing a sub-menu, listening for that buttons event and spawning the corresponding sub-menu. The "thumbstickmoved" and schema.actionKey listeners are dynamically added and removed during moving and if the menu is invisible.

B.1.3 disc-menu

Description: The disc-menu component specifically exists to provide the euler-plane weight selection menu for edge spawning during graph editing. It is spawned during the edge editing process.

Schema:

Property	Type	Default	Description
radius	number	0.05	Radius of the menu disc
type	string	weightSelector	currently unused, potential label to modify the behavior for other tasks than weight setting.

Events:

Event	Detail	Description	
"weightSelected"	amplitude:	Current menu state amplitude	Fired when the weight selection is finished.
	phase:	Current menu state phase	

Listeners:

Event	Listener-Description
"thumbsticktouchstart"	Enables moving of the selector
"thumbsticktouchend"	Disables moving of the selector
"thumbstickdown"	Confirms weight selection.
"thumbstickmoved"	Reads off the current thumbstick value.

Depending on which thumbstick is moved, the selector behavior is different. The right-hand thumbstick is responsible for coarse weight setting, allowing weights of amplitude 0.5 or 1 and phases in $\frac{\pi}{4}$ steps. The left-hand thumbstick allows full freedom for weight selection.

B.1.4 dataslate

Description: The dataslate component builds the dataslate object for a specific graph. It is assigned by the controlpanel component. It controls spawning of perfect-matchings for a graph.

Schema:

Property	Type	Default	Description
graph	string	-	Entity ID of the graph the slate belongs to

Listeners:

Event	Listener-Description
"thumbsticktouchstart"	Adds the scrolling state to the slate.
"thumbsticktouchend"	Removes the scrolling state from the slate.
schema.actionKey	Presses buttons/spawns submenus.
"thumbstickmoved"	Adjusts the scroll speed.
"PMsUpdated"	Updates the slate if the graph is changed.
"togglePM_{i}"	Triggers the corresponding graph-component to spawn/despawn PM_{i}, one listener for each PM.

The component creates entities with the button-component per perfect matching and listens to their events for spawning/despawning via the togglePM_{i} listener where *i* is the pm-index.

B.1.5 controlpanel

Description: The controlpanel component attaches to the central UI element of the scene, the 2D plane in front of the user. It controls behavior like file downloads, which graph is currently being edited, graph removal, renaming, and allows to spawn a fresh graph of a single vertex.

Events:

Event	Detail	Description
"topologyModeEnabled"	-	Fired if the top Mode button is pressed, controls the edge spawning behavior.
"topologyModeDisabled"	-	Fired if the top Mode button is released, controls edge spawning behaviour.
"graphactive"	graph active graph-component	Fired if a graph is set active via pushing the named button, all edits, saves, etc. relate to the active graph.
"graphinactive"	-	Fired if a graph is set inactive via pushing the named button. Editing is now no longer possible and the graph control buttons vanish

Listeners:

Event	Listener-Description
"graphbuilt"	Spawns buttons belonging to the graph.
"saveTo"	Listens for the saving button press. Initiates saving.
"selectGraph"	Listens for the graph named buttons to enable editing and spawn the dataslate and control-buttons
"removeGraph"	Listens for the removal button press. Removes the graph.
"newGraph"	Listens for the new graph button, spawns a new graph consisting of a single vertex.
"rename"	Listens for the rename button, renames the current graph.
"topMode"	Listens for the top-Mode button, changes edge drawing behavior.
"genConfFile"	Listens for the conf-file button, initiates the download of a conf. File.
"superkeyboardinput"	Dynamically added and removed during renaming, saving and template drawing to process keyboard input.

B.1.6 graph-library

Description: This component manages the spawning of graphs from the library of graph files included on the host platform. It is attached to the left-hand controller. It attaches a controller-menu component to the controller allowing the spawning of buttons via button press.

Listeners:

Event	Listener-Description
<graph>	Spawns the corresponding graph.

One graph listener is added per graph in the library. The listener listens to the corresponding button of the controller menu for the event carrying the name of the graph

B.1.7 graph-editor

Description: The graph-editor component is attached to the right hand controller and handles editing functionality like adding and removing vertices as well as drawing and removing edges. During editing cycles it assigns controller-menu and disk-menu components depending on the cycle. Editing functionality is only enabled once a graph is set active via the controlpanel component.

Schema:

Property	Type	Default	Description
vertexSpawner	string	"abuttondown"	Buttonevent for toggling vertices.
editKey	string	"bbuttondown"	Buttonevent for editing a vertex, i.e. spawning/removing edges or relabeling.

Events:

Event	Detail	Description
"graphedit"	type:	edit type
	vertex:	vertex corresponding to the edit
	edge:	edge corresponding to the edit
	graph:	graph corresponding to the edit
		Component makes the edited graph fire this event upon an edit to trigger updates.

Listeners:

Event	Listener-Description
"graphactive"	Enables editing upon setting a graph as active.
"graphinactive"	Disables editing upon setting a graph as inactive.
"topologyModeEnabled"	Changes edge drawing behavior to forgo colors.
"topologyModeDisabled"	Resets to normal edge drawing behavior.
schema.vertexSpawner	Adds/Removes vertices based on the hover state on button press
"hover-start"	Enabled only when editing is enabled. Adds the hovering state if hovering over a vertex.
"hover-end"	Enabled only when editing is enabled. Removes the hovering state if hovering over a vertex.
"stateadded"	Listens only for the hovering state. Switches schema.editKey behaviour between spawning and edit menu and connecting a spawned edge.
"stateremoved"	Listens only for the hovering state. Removes editKey behaviour.

The component also dynamically adds and removes listeners corresponding to the menus spawned during and editing process. Those listen to the corresponding button events for setting edge colors, new vertex labels, or edge deletion identifiers.

B.2 Graph-components

This section groups components relating to constructing and managing graphs.

B.2.1 edge

Description: The edge component is responsible for drawing and managing edge geometry over the lifecycle of a graph, as well as storing edge data. It is assigned either by the graph component during construction or edits. Every edge of a graph has its own instance of the edge component. Edge components are attached to the same entity as the graph component.

Schema:

Property	Type	Default	Description
graph	selector	-	selector for the graph entity the edge belongs to
numId	number	-	numerical edge id
edgeConfig	array	-	edge configuration according to PyTheus convention [vertex 1, vertex 2, color 1, color 2].
siblingConfig	array	[0, 0, [" "]]	[siblingCount, siblingIndex, [siblingNames]] array for edge geometry configuration in case of multi edges.]
phase	number	0	Edge weight phase in units of pi
amplitude	number	1	Edge weight amplitude

Siblings are edges between the same vertices with different colors, the initial array is computed in pre-processing. Sibling count is the total number of siblings, siblingIndex is the index of this edge in the set of siblings, and siblingNames are string identifiers of the entity arguments (e.g. edge__0). During initialization, the edge-config array entries are replaced with pointers to the relevant vertex components. The same happens to siblingNames.

B.2.2 vertex

Description: The vertex component turns entities into graph vertices. These are indented as child-entities to the parent graph entity. It is required to initialize them as separate entities to allow for them to be grabbed individually. Otherwise, they fulfill a similar purpose to edge components, storing vertex data.

Schema:

Property	Type	Default	Description
graph	selector	-	Selector for the graph entity the vertex belongs to.
numId	number	-	Numerical vertex ID.
color	string	-	Vertex color.
position	array	-	3D coordinates of the vertex.
geometry	string	-	Vertex shape (sphere, cube, or tetrahedron)
edges	array	-	List of connecting edge numerical ids
neighbours	array	-	List of neighbour vertex numerical ids

Both edges and neighbors are initialized as lists of IDs, but get replaced with pointers to the components during graph assembly.

Events:

Event	Detail	Description		
"vertexgrabstart"	<table border="1"> <tr> <td>vertex:</td> <td>name identifier of this vertex THREE.Object3D</td> </tr> </table>	vertex:	name identifier of this vertex THREE.Object3D	Perfect Matchings have vertex geometry copies and are listening for this event to start tracing position updates.
vertex:	name identifier of this vertex THREE.Object3D			
"vertexgrabend"	<table border="1"> <tr> <td>vertex:</td> <td>name identifier of this vertex THREE.Object3D</td> </tr> </table>	vertex:	name identifier of this vertex THREE.Object3D	Perfect Matchings have vertex geometry copies and are listening for this event to stop tracing position updates.
vertex:	name identifier of this vertex THREE.Object3D			

Listeners:

Event	Listener-Description
"stateadded"	Checks for addition the grabbed-state to cause attached edges to update their geometries and fire the "vertexgrabstart" event.
"stateremoved"	Checks for removal of the grabbed state to stop attached edges updating and to fire the "vertexgrabend" event.

B.2.3 perfect-matching

Description: The perfect matching component constructs the subgraph of each perfect-matching. It does not use its own vertex and edge components but copies the respective THREE.js objects. It is assigned by the graph-component entity through a trigger through the dataslate-component.

Schema:

Property	Type	Default	Description
graph	selector	-	Selector for the graph entity the perfect matching belongs to.
pmConfig	array	-	List of edges contained in the pm. (numerical identifiers)

Listeners:

Event	Listener-Description
"vertexgrabstart"	Triggers the geometry copies to trace the respective positions of the grabbed vertex.
"vertexgrabend"	Stops the geometry copies from tracing positions.

B.2.4 graph

Description: The graph-component turns an entity into a graph. It manages the construction of the model, assigns the relevant vertex and edge sub-components, and stores the graph-data. It also manages graph updates caused by editing the graph and computes its perfect-matchings.

Schema:

Property	Type	Default	Description
name	string	"graph"	graph name
graphData	string	-	stringified json file containing all relevant graph information.
history	string	-	stringified json containing the optimization history. This is currently unused.

Events:

Event	Detail	Description
"pmdecomposed"	graph: graph entity	Emitted on spawning of a perfect matching
"pmremoved"	graph: graph entity	Emitted on removal of a perfect matching
"graphbuilt"	graph: graph entity	emitted once graph construction is complete
"PMSUpdated"	graph: graph component	Emitted every time the graph is edited, e.g. via vertex addition.

Listeners:

Event	Listener-Description
"graphedit"	Listens for graph edit events to update the geometry
"graphbuilt"	deprecated listener
"grab-end"	Listens for grab events on this graph to updated the collision-mesh upon moving of vertices.

C Code snippets

This section includes useful code snippets for transparency and ease of verifying the presented results:

C.1 computing perfect matchings:

This snippet computes perfect matchings of a graph. Set into the graph variable.

```

from pytheus import fancy_classes as fc
from pytheus import help_functions as hf

graph = {
    (0, 1, 1, 1) : 1,
    (1, 2, 0, 0) : 1,
    (2, 3, 1, 1) : 1,
    (3, 0, 0, 0) : 1
}

graph = fc.Graph(graph)
graph.getState()
print(hf.readableState(graph.state))
print(graph.perfect_matchings)

```

C.2 generating 2d-entanglement swapping graphs:

This code constructs a according to Fig. 28. The function builds the graph, the bottom code segment checks the result for accuracy. Note that the generation of the target is memory intensive for large pair counts.

```

import numpy as np
import itertools
from pytheus import help_functions as hf
from pytheus import fancy_classes as fc

def construct2D_ES_chain(n):
    """Builds the graph for the n-pair 2d entanglement swapping chain. The
       first n photons are Alice, next n Bobs, then n+1 Charlies.

    Args:
        n (int): odd integer specifying pair count.

    Returns:
        dict: dictionary with edge tuples as keys
    """
    a = np.arange(n)
    b = np.arange(n, 2*n)

```



```

c = np.arange(2*n, 2*n + n+1)

# build the left end of the chain
left_end = {
    (a[0], a[1], 0, 0): 1,
    (a[0], c[0], 1, 0): 1,
    (a[0], c[1], 0, 0): 1,
    (a[1], c[1], 0, 0): 1,
    (b[0], c[0], 0, 0): 1,
    (b[0], c[1], 1, 0): 1,
    (b[0], b[1], 1, 0): 1,
    (b[1], c[1], 0, 0): 1
}

# build the right end of the chain
right_end = {
    (a[n-2], a[n-1], 1, 1): 1,
    (a[n-2], c[n-1], 1, 0): 1,
    (a[n-1], c[n-1], 1, 0): 1,
    (a[n-1], c[n], 0, 0): 1,
    (b[n-2], b[n-1], 1, 0): 1,
    (b[n-2], c[n-1], 1, 0): 1,
    (b[n-1], c[n-1], 0, 0): 1,
    (b[n-1], c[n], 1, 0): 1
}

left_end.update(right_end) #update the chain

# build the required number of middle elements and add to the chain
for i in range((n-3)//2):
    left_end.update({
        (a[2*i+1], a[2*i+2], 1, 1):1,
        (a[2*i+1], c[2*i+2], 1, 0):1,
        (a[2*i+2], c[2*i+2], 1, 0):1,
        (a[2*i+2], c[2*i+3], 0, 0):1,
        (a[2*i+2], a[2*i+3], 0, 0):1,
        (a[2*i+3], c[2*i+3], 0, 0):1,
        (b[2*i+1], b[2*i+2], 1, 0):1,
        (b[2*i+1], c[2*i+2], 1, 0):1,
        (b[2*i+2], c[2*i+2], 0, 0):1,
        (b[2*i+2], c[2*i+3], 1, 0):1,
        (b[2*i+2], b[2*i+3], 1, 0):1,
        (b[2*i+3], c[2*i+3], 0, 0):1,
    })
return left_end
n = 9 #define number of pairs, must be odd

# built the kets for the n-pair entangled state to check the result
target = [np.unique(list(itertools.permutations(comb)), axis=0) for comb in
    itertools.combinations_with_replacement([0, 1], n)]
target_kets = []
for comb in target:
    for perm in comb:
        target_kets.append([*perm, *perm])
target_kets = sorted(target_kets)

```

```

# build the graph object to compute the state from the graph
ES_np = construct2D_ES_chain(n)
ES_np = fc.Graph(ES_np)
ES_np.getState()
state = hf.readableState(ES_np.state) #get a readable state
state_kets = sorted(np.array(ES_np.state.kets)[:,:,:1][:,:2*n].tolist()) #get a
    sorted list of kets

print(f"number_of_unique_kets:_{len(list(state.keys()))}_{2*n}")
print(f"alice_and_bob_share_all_photon_states_in_all_kets?:_{np.array_equal(np
    .array(ES_np.state.kets)[:,:,:1][:,:n],np.array(ES_np.state.kets)
    [:,:,:1][:,n:2*n])}")
print(f"correct_state?:_{np.array_equal(state_kets,target_kets)}")

```