Parallel Adaptive Monte Carlo Integration and Vector-Boson Scattering at the Large Hadron Collider

DISSERTATION
zur Erlangung des Grades eines Doktors
der Naturwissenschaften

vorgelegt von

Simon Braß, M. Sc.

eingereicht bei der
Naturwissenschaftlich-Technischen Fakultät
der Universität Siegen

Siegen 2019

1. Gutachter und Betreuer: Prof. Dr. Wolfgang Kilian, Universität Siegen

2. Gutachter: PD Dr. Tobias Huber, Universität Siegen

Datum der mündlichen Disputation: 5. Juni 2019

Zusammenfassung

Das Standard-Modell der Teilchenphysik hat sich als verlässliche Theorie für die Beschreibung der Wechselwirkung zwischen elementaren Teilchen erwiesen. Mit der Entdeckung des Higgs-Bosons am Large Hadron Collider (LHC) sind alle theoretisch-vorhergesagten Teilchen des Standard-Modells beobachtet worden. Damit sieht sich die Teilchenphysik neuen Herausforderungen gegenübergestellt, wie die weitere Validierung des Standard-Modells voranzutreiben oder nach neuer Physik jenseits des Standard-Modells zu suchen. Beides ist durch präzisere und umfangreichere Messungen und experimentelle Daten-Analysen am LHC oder anderen zukünftigen Kollider-Experimenten möglich.

Monte-Carlo–Event-Generatoren sind ein wichtiges Werkzeug für Messungen und experimentelle Daten-Analysen, sowie für theoretische Vorhersagen für Kollider-Experimente. Sie basieren auf der "acceptance-rejection"-Methode für die Event-Erzeugung mit einer vorgegebenen Wahrscheinlichkeitsverteilung im Konfigurationsraum. Diese wird ergänzt durch die Methoden der Monte-Carlo–Integration des d-dimensionalen Phasenraumes mit $d = 3n - 4$ Freiheitsgraden eines n-Teilchen Endzustands, wofür die Anwendung von klassischen, numerischen Integrationsregeln bei großen $n > 4$ von Nachteil sind. Dabei sind Monte-Carlo–Integration und Event-Erzeugung durch das sogenannte „importance sampling" eng miteinander verbunden. Die Anwendung von iterativen und adaptiven Monte-Carlo–Algorithmen für numerischen Integration erlaubt es uns die Effizienz der Event-Erzeugung mit einer vorhergehenden Monte Carlo-Integration zu optimieren.

Wir stellen die Parallelisierung des doppelt-adaptiven Monte-Carlo–Algorithmus Vamp unter der Benutzung der Paradigmen aus OpenMP und MPI vor, mit Rücksichtnahme auf die Minimierung des Kommunikationsbedarfs und einer Methode der Effizienz-Verbesserung der Parallelisierung durch einen statischen Last- und Arbeitsverteiler. Wir haben eine neue Implementation dieses doppelt-adaptiven Algorithmus, Vamp2, für die adaptive Monte Carlo-Integration und Event-Erzeugung geschrieben, welcher als Teil des Monte Carlo Event-Generators Whizard verfügbar ist. Ingesamt können wir eine Verbesserung der Integrationslaufzeit für typische Anwendungsfälle an Kollider-Experimenten im Rahmen von Whizard um die Ordnung 10 erreichen, sodass sich die Rechenzeit von Tagen oder Wochen auf Stunden reduziert.

Wir verwenden die parallelisierte Integration in einer ersten Anwendung im elektroschwachen und Higgs-Bereich des Standard-Modells für hohe Energien an der TeV-Skala, welche am LHC erreichbar sind. Hierzu betrachten wir die Rolle von transversal-polarisierten Eichbosonen und dem Higgs-Boson in Vektorboson-Streuung (VBS), dass heißt, der Streuung von W und Z Boson in Paaren zu W, Z und Higgs.

In einem "bottom-up"-Ansatz der effektiven Feldtheorie des Standard-Modells können wir auf modell-unabhängige Weise anomale Beiträge durch mögliche neue Physik mit dimension-acht Operatoren modellieren. Dabei stellen wir Relationen für die dimension-acht Operatoren für verschiedene Darstellungen des Higgs-Feldes auf, um Vergleiche zwischen unseren Resultaten und existierenden Studien zu ermöglichen.

Es zeigt sich, dass eine Beschreibung der VBS-Prozesse bei diesen hohen Energie durch die effektiven Feldtheorie die Unitarität der Streumatrix verletzt. Durch die Anwendung des "T-matrix" Unitaritätsverfahren können wir die Unitarität der Theorie und damit das verletzte Grundprinzip der Quantenfeldtheorie wiederherstellen. Die dadurch definierten vereinfachten Modelle erlauben uns quantitative Aussagen über (typische) Szenarien neuer Physik zu studieren und mögliche zusätzliche Event-Raten über die Standard-Modell Vorhersagen hinaus durch Obergrenzen abzuschätzen. Davon können wir die Sensitvität auf Resultate für neue Physik am LHC abstecken.

Abstract

The Standard Model of particle physics has proved to be a reliable theory for the description of the interaction between elementary particles. With the discovery of the Higgs boson at the LHC, all predicted particles of the Standard Model have been observed. As a result, particle physics faces new challenges such as further validation of the Standard Model or the search for New Physics beyond the Standard Model. More precise and more comprehensive measurements and experimental data analyses at the LHC or other future collider will shed light on this.

Monte Carlo event generators are an indispensable tool for measurements and experimental data analyses, as well as for theoretical predictions at collider experiments. They are based on the acceptance-rejection method for event generation with a given probability distribution in configuration space. This is supplemented by the methods of Monte Carlo integration of the d -dimensional phase space with $d = 3n-4$ degrees of freedom of an n-particle final state, for which the application of classical, numerical integration rules for large $n > 4$ are disadvantageous. In particular, Monte Carlo integration and event generation are closely linked by the so-called importance sampling. The application of iterative and adaptive Monte Carlo algorithms for numerical integration allows us to optimize the efficiency of event generation with a previous Monte Carlo integration.

We present the parallelization of the doubly-adaptive Monte Carlo algorithm VAMP using the OpenMP and MPI paradigms, under special consideration of minimizing communication needs and a method to improve the efficiency of parallelization by a static load balancer. We have written a new implementation of the doubly-adaptive algorithm, VAMP2, for adaptive Monte Carlo integration and event generation, which is available as part of the Monte Carlo event generator WHIZARD. We reached an overall improvement of the integration run-time of WHIZARD for typical use cases at collider experiments by order 10, reducing possible computation times from days or weeks to hours.

In a first application, we employ the parallelized integration in the electroweak and Higgs sector of the Standard Model for high energies at the TeV scale accessible by the LHC. For this purpose, we consider the role of transverse-polarized gauge bosons and the Higgs boson in vector boson scattering (VBS), that is, the scattering of W and Z bosons in pairs to W, Z, and Higgs.

In a bottom-up approach of the effective field theory of the Standard Model, we model anomalous contributions by possible new physics in a model-independently way with dimension-eight operators. We provide relations for the dimension-eight operators for different representations of the Higgs field to allow comparisons between our results and existing studies.

A description of the VBS processes by the effective field theory at such high energies violates the unitarity of the scattering matrix. By applying the T-matrix unitarity projection we can restore the unitarity of the theory and thus the first principles of quantum field theory. The simplified models, defined in this way, allow us to study quantitative statements about (typical) scenarios of New Physics and estimate possible event rates beyond the standard model predictions by upper limits. From this we can pinpoint the sensitivity on results for New Physics at the LHC.

"Study hard what interests you the most in the most undisciplined, irreverent and original manner possible." (Richard P. Feynman)

Acknowlegdements

I would like to thank Prof. Dr. Wolfgang Kilian for giving me the opportunity to perform my PhD under his supervision. I was offered the chance to elaborate and present my own ideas and visions in a scientific way with your helpful assistance. Furthermore, I am grateful for your comments on drafts of this thesis.

I would like to thank PD Dr. Tobias Huber for helpful discussions on the technical details of writing a thesis and to be the second referee of this thesis.

I would like to thank Prof. Dr. Markus Risse and Prof. Dr. Thorsten Ohl for joining my thesis committee.

I would like to thank my colleague Dr. Jan Piclum for the intense proofreading of my thesis and lots of help, preventing me from submitting a linguistic disaster. Furthermore, I would like to thank my office partner Nils Kreher for encouraging discussions and his support during the final fine-tuning of this thesis.

I would like to thank my former office partner and computer administration colleague Dr. Rebecca Klein as well as my present computer administration colleague Dr. Jan Piclum for your elusive support regarding the administration of our institute's computer system.

I would like to thank my former office partner and colleague Christian Fleper. After your unexpected departure from theoretical particle physics, I was able to continue your project thanks to your excellent preliminary work; without I would not haven been able to complete the project in the given short amount of time of my last thesis year. In particular, I would like to thank you for providing the implementation and fruitful notes on the project.

I would like to thank the other WHIZARD collaborators Prof. Dr. Thorsten Ohl, Dr. Christian Weiß, Dr. Bijan Chokoufé Nejad, Dr. Jürgen Reuter, Dr. Marco Sekulla, Vincent Rothe and Pascal Stienemeier for enlighting discussions on my WHIZARD-related projects during this thesis. In particular, I would like to express my gratitude towards Dr. Jürgen Reuter for several invitations to Deutsches Elektron-Synchrotron (DESY) for prolonged discussions on my and other related projects in WHIZARD.

I would like to express my gratitude towards the former and present colleagues of the Theoretical Particle Physics Institute at University Siegen; it was a great pleasure to be part of such a stunning work group. In particular, I would like to thank Dr. Paul Moch, Dr. Keri Vos and Dr. Jan Piclum for the many entertaining game evenings.

Last, but not least, I would like to thank my family, especially, my parents Bernd and Marianne Braß, my brother Tobias Braß and Jessica Reuter for your incredible support and the solidarity you have given me during my time as PhD student. Without you, I certainly would not have made it.

Contents

List of Figures

List of Tables

Preface

After a decade-long search in a series of experiments, a new area in particles physics has opened with the launch of the LHC. The tremendous success of particle physics, both in terms of experiment and theory, is based on the predictive power of the SM, which evolved over the last century. In 2012, at the end of the first run of the LHC at center-of-mass energies of 7 TeV and 8 TeV, the discovery of a light Higgs-like state with a mass of 125 GeV was reported by the two major LHC experiments, ATLAS [1] and CMS [2]. With that, all particles predicted by the SM have been observed and their parameters determined. However, with the discovery of the Higgs, particle physics faces new questions in the energy range above 1 TeV, especially with regard to the electroweak and the Higgs sector: Is the Higgs sector minimal or non-minimal? Is it strongly or weakly interacting? Are the first principles of quantum field theory (QFT) validated or invalidated?

New experimental data will deepen our understanding. Especially, with the upcoming third run phase of the LHC, as well as with the proposed High-Luminosity LHC and High-Energy LHC at 27 TeV, experimental observations will be achieved with an ever increasing amount of data and precision. Experimental analyses will be able to advance into less-tested areas of the SM and probe electroweak physics and the Higgs sector, particularly, trilinear and quartic couplings of Higgs and weak gauge bosons are one of the more interesting candidates.

In order to compare experimental data with theoretical predictions, Monte Carlo event generators [3] are an indispensable tool of elementary particle physics for both experiment and theory. They supply us with well-founded theoretical predictions which can be compared to experimental data, or help us to develop experiments such as those at the LHC or planned future high-energy colliders, such as International Linear Collider (ILC), Compact Linear Collider (CLIC) or Future Circular Collider (FCC).

The comparison of theory and experiment requires a two-step approach, on the one hand the generation of particle-level event samples, resulting in a set of particle species and four-momenta, and on the other hand, the simulation of the detector response. In order to generate particle-level events, Monte Carlo event generators rely on the well-proven methods of QFT and perturbation theory. Starting at high energies, where perturbation theory has its validity, the hard scattering process for partonic observables, such as cross sections and decay rates, and partonic event samples are computed. In subsequent steps, these are then evolved via parton showers, subsequent decays and non-perturbative models for hadronization to the energy and length scales of the particle-level events as observed in a detector.

We focus our efforts on calculations regarding hard scattering processes, which involve elementary SM particles, e.g. gluons, quarks, leptons, electroweak bosons and the Higgs boson, and the photon. The large number and complexity of scattering events recorded at detectors such as ATLAS or CMS call for a matching computing power for the simulation

and the efficient use of existing computing resources. By parallelizing the simulations, these resources can be better utilized.

Typical processes for the LHC are $2 \to 2$ or $2 \to 3$ scatterings, where resonant particles in the final state are subsequently decayed and additional jets are added by a parton shower. For these processes, analytical expressions for both the matrix element and the phase space are often known, at leading order of the perturbative series. For those the application of parallelization can be done in a trivial way on a single machine with a multi-core CPU, operating with nearly optimal efficiency. However, multi-leg processes, which are also probed by the LHC experiments, with $2 \to n$, where $n \geq 4$, involve more and more complex expressions where automation is the method of choice in order to generate associated matrix elements and phase-space expressions. An example for such processes is given by those involving vector-boson fusion or vector-boson scattering, where the simulation handles elementary $n = 6$, 8 and 10 processes.

With the beginning of LHC automated leading-order calculations were more or less the state-of-the-art method for Monte-Carlo event generators. However, with the growing needs for precision in data analysis, and planned future high-energy and high-luminosity lepton and hadron colliders, automation is shifting towards next-to-leading or even higher orders. Computational simplification, as for leading order, can not be used anymore at next-to-leading due to separation of production and decay or signal and background processes, making the computation even more complicated and prolonged. Here, we restrict ourselves to improvements for multi-leg processes at leading-order, as an extension to next-leading order seems to be trivial from a purely technical point of view.

Summarizing, the increased need for higher-precision predictions involve more complicated processes (more particles in the final state or next-to-leading order, parton shower and hadronization), and require therefore longer computation times, which grows with the complexity in orders of magnitude: from hours to days, from days to weeks or months. Hence, optimization and parallelization play a crucial role in order to fulfill these requirements [4] and to keep up in the race between experiment and theory.

We describe a new approach to efficient parallelization for adaptive Monte Carlo integration and event generation, implemented within WHIZARD [5] Monte Carlo integration and event-generation program. The approach combines independent evaluation on separate processing units with asynchronous communication via MPI-3.1 with internally parallelized loops distributed on multiple cores via OpenMP.

For such computation tasks, the particle physics community has different systems of computing resources at hand, mainly grid and high-perfomance computing (HPC)-based systems. These differ in their respective philosophy, as well as the corresponding hardware. Grid-based systems are used to process enormous numbers of serial computing tasks simultaneously, which do not require node-communication during run-time. Whereas HPC systems are explicitly designed for computations that require node-communication during run-time, these systems are better equipped in terms of computing and network performance, but with less computing nodes. Both systems have their valid application in the area of particle physics, and the answer, which system will assert itself in the future, is still pending. We are pushed towards HPC due the adaptive nature of the applied Monte Carlo integration algorithms which require substantial communication after each iteration.

With the Higgs as a portal to possible new physics beyond the Standard Model

long-standing puzzles of particle physics, e.g. dark matter, might be solved. However, new physics scenarios have to respect data from current and previous experiments. Therefore, we expect new physics to manifest themselves primarily in interactions of massive electroweak gauge bosons, namely, $\mathrm{W}^{\pm}, \mathrm{Z}$, and the Higgs boson.

Processes of the type $VV \to VV$, where $V = \mathrm{W}, \mathrm{Z}, \mathrm{H}$, known as vector-boson scattering (VBS) and associated or Higgs-pair production in vector-boson fusion (VBF), are a sensitive probe of electroweak physics and the Higgs sector. VBF and VBS are a prime application for the parallelization of WHIZARD, [6, 7] as the integration and event generation is time-consuming due to the complex matrix element and high-dimensionality of phase space. Deviations from the SM can be parameterized, based on Lorentz and gauge invariance, in a convenient, methodical way using an effective field theory (EFT). In a bottom-up approach, the SM as low-energy theory can be extended to energy scales above 1 TeV as Standard Model Effective Field Theory (SMEFT) [8–11]. The ATLAS and CMS experiment at the LHC have measured VBS processes as a signal embed in partonic process of the type $qq \to VVqq$, where q is any light quark. Numerical results [12–16] have been presented in the form of limits on parameters of the SMEFT.

Hints to new physics may arise as anomalous contributions to trilinear or quartic gauge couplings. The usual applications of the SMEFT restrict themselves to dimension-six operators. A useful parameterization regarding quartic gauge coupling, in particular, independently of the trilinear gauge couplings, requires dimension-eight effective operators, the second order of the low-energy expansion beyond the SM. In recent works [17–19], possible deviations from the SM have been studied in VBS processes, but confined to the longitudinal scattering modes of W and Z bosons. The numerical results of the non-SM interactions of the longitudinal scattering have shown that for the level of deviations that can be detected by the LHC experiments, the unitarity bounds as dictated by the unitarity of the S-matrix are *always* violated in the high-energy range, if a naive SMEFT calculation is attempted. At first glance, a model-independent prescription of new physics beyond the SM that covers the accessible parameter range becomes impossible. But, reasonable assumptions on new physics, e.g. a global high-energy symmetry and that fermions decouple from new physics, and a unitarization projection lead to unitarity constraints that limit the level of possible excess above the SM prediction. It is then possible to devise simplified models that both satisfy unitary over the whole energy range and smoothly match onto the SMEFT parameterization at low energy.

We study deviations from the SM in VBS processes that involve transverse-polarized W and Z bosons, and also consider Higgs bosons in the final state. We provide relations for the dimension-eight operators for different Higgs field definitions in order to compare our results with previous studies. For the purpose of an exemplary study we have compared a class of continuum models, which merely extrapolate the SMEFT expansion into asymptotically strong interactions with models that describe single resonance with specific quantum number assignments as weakly-interacting ones.

The thesis is structured as follows, in chapter 1 we introduce the theoretical foundations of the Standard Model as well as the basics of Monte Carlo integration and event generation. We complete the introduction with a description of the multi-purpose Monte Carlo event generator WHIZARD and new contributions due to this thesis.

In chapter 3, we introduce the improved Monte Carlo integration algorithm and discuss possible ways of optimization and parallelization. We illustrate variance-reducing

techniques of adaptive Monte Carlo integration and the application of mappings for an efficient parameterization of the phase-space. Furthermore, we highlight the different parallelization paradigms and their applicability with regard to the parallelization of the adaptive algorithm within the Monte Carlo event generator WHIZARD.

This is followed in chapter chapter 3 by a discussion of the new implementation, VAMP2, and the technical details of the parallelization implementation of the adaptive Monte Carlo algorithms. Furthermore, we discuss the run-time improvements due to our parallelization regarding the integration of typical processes within the WHIZARD.

In chapter 4, we introduce the extension of the Standard Model as an effective theory and present new interactions for the electroweak and Higgs sectors. We explain the basic assumptions, which ends in the definition of the low energy operators. From this we construct unitary models which either exhibit a strongly-interacting continuum in the high energy range or receive a resonance and are otherwise parameterized likewise the VBS amplitudes to all energies. These models yield then the maximally allowed number of events consistent with quantum field theory in the VBS channel, matched to the low-energy SMEFT with specific values for the operator coefficients. For the implementation, we utilize the symmetry of the new interaction in order to determine the eigenmodes of the quasi-elastic $2 \to 2$ scattering, thus, to diagonalize the amplitudes for all vector-boson states in isospin-spin eigenamplitudes. We present numerical results and plots for selected parameter sets and final states. To conclude, we discuss the relevance of our results for future analysis at the LHC or other future colliders.

We conclude and summarize the thesis in chapter 5.

1 Theoretical Foundations

In this chapter, we outline the basics of particle physics and the connection between experiment and theory. There is a wide-spread textbook literature on particle physics.[1] In this short introduction based on [20, 24] we introduce the basic concepts of QFT and modern particle physics, which will then lead us to the necessity of computational automation and the questions of optimization and parallelization.

The chapter is structured into the presentation of the Standard Model of particle physics section 1.1, the formulation of the observables of theory and experiment in section 1.2. We introduce the concept of Monte Carlo integration in section 1.3 and event generation in section 1.4, and discuss their natural relation to each other. In section 1.5, we introduce the multi-purpose Monte Carlo event generator WHIZARD around which this thesis revolves.

1.1 The Standard Model of Particle Physics

The description of the fundamental forces and properties of nature at smallest scales is given by the SM of particle physics. The SM has gradually evolved over the second half of the last century and has so far withstood every experimental test. The most important points in this development were the emergence of quantum field theory in the form of quantum electrodynamics, the introduction of renormalization of infinities, the description of the strong interaction within the Yang-Mills theory of non-Abelian local gauge symmetries, the unification of electromagnetic and weak gauge theories in the Glashow–Salam–Weinberg gauge theory of the electroweak interactions, the spontaneous electroweak symmetry breaking introduced by the Higgs mechanism in order to generate masses for the all bosons and fermions (except neutrinos) and the prediction of a third generation of quarks by Kobayashi and Maskawa in order to solve the weak CP-violation, and thus the emergence of flavor physics with the description of the Cabibbo-Kobayashi-Maskawa (CKM)-matrix.

The interactions between particles are described by local gauge symmetries, where quantum fields transform locally under elements of the gauge groups which are dependent on the space time. The connection between two field configurations at different space times is given by the covariant derivative, which contains the gauge fields as minimal couplings. The interaction between field configurations is therefore propagated through the gauge fields. The gauge group of the SM is

$$\mathcal{G}_{\mathrm{SM}} = \mathrm{SU}(3)_{\mathrm{color}} \times \mathrm{SU}(2)_{\mathrm{L}} \times \mathrm{U}(1)_{\mathrm{Y}}. \tag{1.1}$$

The $\mathrm{SU}(3)_{\mathrm{color}}$ is the symmetry of the strong interaction with coupling constant g_{s}; this theory of Quantum Chromodynamics (QCD) involves three colors and has eight mediators,

[1] Our recommendations being [20–24].

the gluons. The $\mathrm{SU(2)_L}$ symmetry of the weak interaction with coupling constant g has three gauge fields and the $\mathrm{U(1)_Y}$ is the symmetry of the weak hypercharge with coupling constant g' with one gauge boson. All gauge bosons are vector particles of spin 1 having three polarization states, in the massive case, or two polarization states, in the massless case. After spontaneous electroweak symmetry breaking, the Higgs potential acquires a non-zero vacuum expectation value and the fermions acquire masses through Yukawa couplings to the Higgs field. Rotating the gauge fields of the broken $\mathrm{SU(2)_L} \times \mathrm{U(1)_Y}$ subgroup with the Weinberg angle θ_w into the mass eigenbasis gives the observable, massive gauge bosons $\mathrm{W}^\pm, \mathrm{Z}$ and the massless photon. The massless Goldstone bosons with spin 0, which appear as result of the spontaneous symmetry breaking of the Higgs field, are eaten by the weak gauge bosons, in unitary gauge. Thus, with the gauge bosons gaining their masses, the unphysical Goldstone bosons reappear as longitudinal polarizations of the massive weak gauge bosons. The SM gauge group reduces then to the symmetry of low-energy, $\mathrm{SU(3)_{color}} \times \mathrm{U(1)_{em}}$. We remark that the interplay between Higgs and weak gauge bosons plays an important role for high-energy predictions around the TeV scale for the weak interactions. An intricate cancellation between Higgs and longitudinal polarization of the weak gauge bosons hinders amplitudes to rise unphysically strong with the energy. Without the Higgs, amplitudes proportional to the longitudinal polarization of the weak gauge bosons would violate the unitary bound imposed by QFT and, therefore, would leave the applicable range of the perturbative expansion of the SM.

From the experimental side, the major predictions of the standard model were confirmed. Nevertheless, the keystone of the SM, the Higgs boson, was only confirmed with the discovery of a Higgs-like light state at 125 GeV, independently within the ATLAS and CMS experiments at the LHC in 2012 [1, 2]. With that, a period of precision measurements of SM parameters as well as the different trilinear and quartic couplings of Higgs and electroweak vector-bosons has started with current studies, e.g. [25], at the LHC, or is planned for future colliders.

With the finding of such a Higgs-like state, we can continue our description of the SM with a linear representation of the Higgs field. For the future, the Higgs will play a crucial role at the LHC.

1.2 Formulation of Phase Space, Cross Section and Decay

The outcome of such collision experiments as run at the LHC, e.g. ATLAS and CMS experiments, or any other experiment at any planned future collider like CLIC, FCC or ILC, can be described by means of cross sections. For a quantum mechanical process, the cross-section, based on the classical definition, is a measure of the probability P of interaction (given in units of area)

$$\mathrm{d}\sigma = \frac{1}{T}\frac{1}{\Phi}\,\mathrm{d}P, \tag{1.2}$$

where T is the time for the experiment and Φ is the incoming flux. The cross section is then directly related to the measured events N in the detector of an experiment by the luminosity of the accelerator

$$\mathrm{d}N = \mathcal{L}\,\mathrm{d}\sigma. \tag{1.3}$$

Detector effects such as covered solid angle, trigger efficiency, etc. must be taken into account in the analysis of the experiment to compare experimental data and theoretical predictions leading to the so-called fiducial cross section.

On the other hand, we can formulate the theoretical foundation more profoundly. We describe the scattering of an initial state $|i\rangle$ to the final state $\langle f|$ by a scattering matrix $\mathcal{S}$. The probability for a transition $\langle f|\mathcal{S}|i\rangle$ is directly connected (in a Lorentz-invariant way) to the matrix element $\mathcal{M}$ by

$$\langle f|\mathcal{S}-\mathbb{1}|i\rangle = \mathrm{i}\,(2\pi)^4\,\delta^4\left(\sum_i q_i + \sum_f p_f\right)\mathcal{M}, \tag{1.4}$$

where the initial state $|i\rangle$ and the final state $\langle f|$ are to be taken at times $t = \mp\infty$ and $\mathcal{M} \equiv \langle f|\mathcal{M}|i\rangle$, for ease of notation. The translation into an observable is given by the cross section for a scattering of $2 \to n$ with the matrix element $\langle p_1, \dots, p_n|\mathcal{S}|q_1, q_2\rangle$, or a decay $1 \to n$ with the matrix element $\langle p_1, \dots, p_n|\mathcal{S}|q_1\rangle$, into n particles with momenta $p_1, \dots, p_n$, respectively, where $n \geq 2$. Following the definition of [24], the differential cross section is given by

$$\mathrm{d}\sigma = \frac{(2\pi)^4}{4\sqrt{(q_1\cdot q_2)^2 - m_1^2 m_2^2}}\,|\mathcal{M}|^2\;\mathrm{d}\Phi_n\left(q_1+q_2; p_1, \dots, p_n\right). \tag{1.5}$$

The decay rate of a particle in its rest frame into n bodies is given by

$$\mathrm{d}\Gamma = \frac{(2\pi)^4}{2M}\,|\mathcal{M}|^2\;\mathrm{d}\Phi_n\left(q_1; p_1, \dots, p_n\right). \tag{1.6}$$

The Lorentz-invariant n-body phase space is given by

$$\mathrm{d}\Phi_n\left(Q; p_1, \dots, p_n\right) = \delta^4(Q - \sum_{i=1}^{n} p_i)\prod_{i=1}^{n}\frac{\mathrm{d}^3 p_i}{(2\pi)^3 2E_i}, \tag{1.7}$$

where $Q = q_1 + q_2$ for the scattering and $Q = q_1$ for the decay. In general, arbitrary analytical expressions for the construction of phase space exist for $n = 2$ and 3. For higher multiplicities algorithms in closed form exist for flat population[2] of the phase space [26, 27]. For arbitrary phase-space distributions with high particle multiplicities, such closed algorithms are not available. The standard approach is to decompose the phase-space into Lorentz-invariant two-body phase spaces, which are individually parameterized and kinematically linked by a series of subsequent Lorentz boosts [28].

The transition matrix is connected to the fields of the QFT by the Lehmann–Symanzik–Zimmermann reduction formula. Time-ordered perturbation theory allows us then to compute the matrix element in eq. (1.4) as an expansion order by order in one or more coupling constants g_i, as long as $g_i \ll 1$. In general, these couplings are sensitive to the energy scale; this fact is known as running of the coupling. An important example of the running of a coupling is QCD with its asymptotic freedom. The coupling strength of QCD

[2] A corresponding flat-populated phase space is filling with four-momenta associated with a constant weight, which can be proved for massless particles in a closed analytical form. For massive particles, this only applies approximately, when the energy is large compared to the invariants of the phase space [26].

becomes strong for small energy scales below $\Lambda_{\mathrm{QCD}} \sim 250\,\mathrm{MeV}$, thus, at large length scale. Conversely, the strong interaction becomes weak at high energies, thus, at small length scales. This allows for a perturbative calculation of the elementary interactions at the LHC or other high-energy collider.

Within the theoretical framework of perturbation theory, we can express the matrix element $\mathcal{M}$ of particle interactions, and therefore, scattering processes and decays, to any order pictorially with Feynman diagrams. Feynman diagrams can then be translated with so-called Feynman rules[3] into mathematical expressions. For processes, involving non-elementary particles in the initial-state like proton-proton collision, non-perturbative effects (in the length scale of the protons) need to be separated from the perturbative expansion of the hard process at high energies. Hence, the non-perturbative part is factorized from the perturbative part. In order to restore the full hadronic process, the hard partonic process is convoluted with parton distribution functions to the level of compound particles. However, these measures are not sufficient for the complete theoretical description of the final state. To describe the physics from the energy scale of the hard partonic process down to the energy scales of hadrons, the unstable final-state particles need to decay to soft gluons (and photons) in the form of a parton shower. And finally, due to color confinement of QCD, all color-charged particles are hadronized. Since this part can not be done perturbatively, it has to be modeled in some way.[4]

Instead of writing complicated formulas from scratch (or more accurately from diagrams), we can handle diagrams representing different processes automatically and translate them to mathematical formulas, which results in fully-automated matrix element generators and associated program frameworks, the so-called Monte Carlo event generators.

To summarize, from theory side in particle physics, strong predictions for an experiment and for the adjustment of the detectors are required. As we discussed above, the computation of (differential) cross sections and decays can be completely automated for leading order processes[5]. The total cross section or decay rate can be determined by integration of the matrix element over the total Lorentz-invariant phase space. Differential cross sections and decay rates can be determined from event samples.

1.3 Monte Carlo Integration

The phase-space integration over eqs. (1.5) and (1.6) are typical for particle physics. They have a complicated integrand and even more complicated integration regions dictated by momentum conservation and on-shell condition. The high-dimensionality, e.g. for the cross section $d = 3n - 4$, where typically $n = 2, ..., 8, 10$, forces us to divert from classical cubature rules and to use Monte Carlo methods in order to evaluate the integral. We outline the relevant information given in section 3.1 of [30].

[3] These Feynman rules can be derived from a Lagrangian, as known from classical mechanics, which is manifestly Lorentz-invariant.

[4] These models should be sufficiently motivated by theory and confirmed by experiment These calculations are then mostly done in a non-analytical, numerical fashion [29].

[5] In principle this is also true for next-leading-order processes, but in this work we deal exclusively with leading-order processes.

The integral I for an integrand $f : \Omega \to \mathbb{R}$, e.g. a real-valued squared matrix-element over the compact d-dimensional phase-space manifold Ω, is given by

$$I_\Omega[f] = \int_\Omega \mathrm{d}\mu(p) f(p). \tag{1.8}$$

The coordinates p represent four-momenta and possible extra integration variables for structure function parameters or initial-state radiation. Four-momentum conservation, on-shell conditions, (user-defined) cuts and weight factors are considered to be part of the phase-space manifold and the measure $\mathrm{d}\mu(p)$. For the numerical evaluation a phase-space parameterization has to be chosen. A phase-space parameterization is defined as a bijective mapping ϕ from the subset U of the d-dimensional unit hypercube, $U \subset (0,1)^d$, onto Ω with Jacobian $\phi' = \left|\frac{\mathrm{d}\phi}{\mathrm{d}x}\right|$,

$$p = \phi(x), \quad \mathrm{d}\mu(p) = \phi'(x)\mathrm{d}\mu(x) = \phi'(x)\rho_\phi(x)\mathrm{d}^d x, \tag{1.9}$$

where the map with density $\phi'(x)\rho_\phi(x)$ maps to the canonical measure on $\mathbb{R}^d$. In addition, the integration over the unit hypercube can be arbitrarily continued beyond the fiducial phase space Ω as long as $\mathrm{d}\mu(x) = 0$ holds there.

A general overview of Monte Carlo Theory and its application is given in [31]. In stochastic, the integral $I_\Omega[f]$ matches the expectation value of $f(x)$

$$E[f] = \int_\Omega f(p)\mathbb{1}_\Omega \,\mathrm{d}\mu(p) = I_\Omega[f], \tag{1.10}$$

for a uniform probability density distribution $\mathbb{1}_\Omega(x)$. Therefore, the integral I_Ω can be estimated by a finite sum

$$E_N[f] = \langle f \rangle_N = \frac{1}{N}\sum_{i=1}^{N} f(\phi(x_i))\phi'(x_i)\rho_\phi(x_i), \tag{1.11}$$

with the points $x_i \in U$ distributed according to a uniform distribution and N the number of random number configurations for which the integrand has been evaluated, the *calls*. The estimate in (1.11) is for independent random points statistically distributed according to a Gaussian (or normal) distribution[6] around the true value $I_\Omega[f]$. The statistical error of (1.11) is given by the square root of the variance estimate,

$$V_N[f] = \frac{N}{N-1}\left(\langle f^2\rangle_N - \langle f\rangle_N^2\right), \tag{1.12}$$

which can be computed alongside (1.11). Furthermore, the definition of the accuracy a of the sampling is defined in the context of

$$\frac{\Delta I}{I} = \frac{a}{\sqrt{N}}, \tag{1.13}$$

where we find that "the accuracy a depends on the actual variance of the effective integrand" [30]

$$f_\phi(x) \equiv f(\phi(x))\phi'(x)\rho_\phi(x). \tag{1.14}$$

[6] Law of large numbers.

For later reference, we will refer to a single evaluation of (1.14) as *event* with associated weight $w_i = f_\phi(x_i)$.

Variance-reducing methods can be applied to improve the overall scaling behavior of the variance beyond $\propto 1/N$, or to compensate ill-behaving integrands. The two methods of choice are either stratified sampling, where the integrand is more frequently evaluated at points where the contributions to variance are large, or importance sampling, which evaluates the integrand at points where the absolute value of the integrand is large in magnitude. We will introduce these terminologies and their application in more detail in sections 2.1 and 2.2.

1.4 Event Generation

In eq. (1.5) and later on, we introduced the definition of fiducial cross section using the notion of an *event*, which we did not clarify further. Following the definition in [23], an event refers to an individual interaction of colliding beams at a particle collider. Further machinery of detector systems and algorithms using a wide range of methods and technologies is then used to reconstruct the properties of the primary interacting particles, thus, the hard-scattering process. In simple terms, an experimental setup tries to connect a signal "in the different detector system back to the Feynman diagram responsible for the interaction" [23, p.22]. The properties of the detector setup and the detector precision dictate the possible observables and the comprehensible event properties. On the other hand, for the purpose of a (Monte Carlo) simulation, we *can* define the event properties, as long as they are physically meaningful observables. For our considerations,[7] we reduce the definition of an individual event to the bare minimum (along the lines of [32]): a set of momenta, particle types, spins and charges.

Experimentally, events are produced at a particle collider, they automatically obey the laws of physics, and appear therefore with the correct probability. In order to predict events theoretically, we have to understand the probability of an event. For completeness, we give a brief overview of the concepts of probability theory used in particle physics.

The probability density $f(x;\theta)$ is a non-negative function and can depend on some arbitrary parameters θ. The probaility density is normalized,

$$\int f(x;\theta)\,\mathrm{d}x = 1. \tag{1.15}$$

We can interpret $f(x)\,\mathrm{d}x$ as the probability to find an event x in the interval $[x, x+\mathrm{d}x]$. The cumulative distribution function $F(a)$ is the probability that $x \leq a$,

$$F(a) = \int_{-\infty}^{a} f(x)\,\mathrm{d}x, \tag{1.16}$$

$$P(a < x \leq b) = F(b) - F(a). \tag{1.17}$$

In particular, we have the important relation:

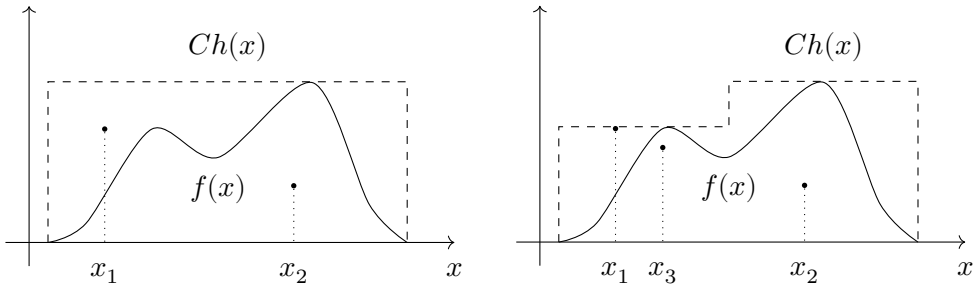
$$\frac{F(x)}{\mathrm{d}x} = f(x). \tag{1.18}$$

[7] We limit ourself to the generation of complete partonic events, without any application of a parton shower, hadronization or detector effects.

In the following, we consider how to generate an F-distributed random number from a uniform random variable. Let $u \in [0,1]$, and a be distributed according to $f(x)$, such that, $u = F(a)$ holds. If we can find the inverse distribution function, then we can relate a uniformly distributed random variable to an F-distributed random variable, $a = F^{-1}(u)$. Unfortunately, the inverse distribution is only known for a rather small set of functions. In particular, for a probability density given by a matrix element, it is in most cases too complicated or even impossible to provide a unique inverse. Therefore, classically the acceptance-rejection method is used, where the actual but complicated probability distribution f is enveloped in a simpler, invertible probability distribution h. Usually, the envelope is given by a uniform distribution or the sum of multiple uniform distributions, and we find,

$$f(x) \leq Ch(x), \quad |f(x)| < C \wedge C > 1. \tag{1.19}$$

Given the constant C, we generate x according to $h(x)$ and calculate $f(x)$. We accept x under the condition that $\frac{f(x)}{Ch(x)} \leq u$,[8] where $u \in [0,1]$, else we reject x.



(a) Rectangular envelope $Ch(x)$. x_1 gets rejected and x_2 accepted.

(b) Adapted envelope $Ch(x)$. x_3 becomes more likely accepted.

Figure 1.1: Illustration of the acceptance-rejection method. We show an arbitrary probability density function $f(x)$ and two examples of an envelope shape: rectangular and adapted. Random points are chosen inside the upper bounding function $Ch(x)$ and are accepted if the coordinate lies below $f(x)$, else rejected.

The acceptance-rejection algorithm populates the area (speaking in terms of a graph, see fig. 1.1) below $Ch(x)$ with points and only accepts those points that lie below $f(x)$. These points are distributed with the probability $f(x)$, as required. The efficiency ε of the acceptance is defined as the ratio of the area of $f(x)$ over $Ch(x)$, hence, $\varepsilon = \frac{1}{C}$. We can directly translate the efficiency into computation time. Let us have N points distributed according to $f(x)$, then we need to compute on average N/ε points for the acceptance-rejection method, resulting in an additional $1/\varepsilon$ computation time in contrast to the direct inverse distribution method. Therefore, we want the efficiency to be as close as possible to one; an optimal choice of $h(x)$ will approximate the form $f(x)$ as well as possible (and feasible). With the acceptance-rejection method, we now have the tool to generate physical events by unweighting, such that the events occur with the frequency specified by the corresponding matrix element of a process.

[8] For the actual computation, the test $f(x) \leq uCh(x)$ is numerically stable against division by zero.

In the previous section, we have already introduced the concept of Monte Carlo integration, which is based on sampling of the integrand function at uniformly distributed random points. However, instead of a uniform distribution we can make use of a well-adapted probability distribution matching the form of the integrand function, the so-called importance sampling, which we will introduce in more detail in chapter 2.

Here, we want to point out the important connection between Monte Carlo integration and event generation: The same probability distribution can be used for both, the integration and event simulation. Hence, with the above knowledge we find a natural approach to an efficient event generation. First, we choose or adapt a probability distribution, starting with a uniformly distributed one, according to the integrand function *during* integration. Then, using the previously adapted probability distribution for the actual simulation of the physical events with the acceptance-rejection method, we can vastly improve the efficiency with a well-adapted probability distribution, assuming that the maximum event weight $w_{\max}$, as in eq. (1.14), can be estimated sufficiently well during integration. The efficiency ε of the acceptance (also known as unweighting efficiency), is the ratio of the average event weight over the maximum weight,

$$\varepsilon = \frac{\langle w \rangle_N}{w_{\max}}. \tag{1.20}$$

In summary, this approach has two advantages: We get the integrated cross section or decay width, and we gain an optimization of efficiency for the event simulation from adapting the probability distribution during integration for the event generation on the fly, resulting in an overall reduced computation time for the event generation.

1.5 Multi-Purpose Monte Carlo Event Generator: WHIZARD

The multi-purpose event generator Whizard [5, 33, 34] is a multi-channel Monte Carlo integration and event generation program. The theoretical framework is given by leading-order perturbative theory supplied in a fully automated way by the matrix-element generator Omega. Beyond that, support for next-to-leading order (NLO) automation within the Frixione–Kunszt–Signer (FKS) subtraction scheme has been implemented [35, 36], but is still under validation. Initial studies dealt with NLO QCD corrections for top processes and decays [37–39], and resummation with POWHEG matching [40–42]. However, Whizard masters the classical tasks of a Monte Carlo event generator, such as calculation of cross sections and other observables, generation of event samples, for hard scattering processes and decays of particles with multiplicities of up to 10 particles at high-energy colliders. In addition, Whizard supports hadronic scattering processes, including the convolution with PDF, e.g. with LHAPDF [43], and provides support for the calculation of Bremsstrahlung, QED initial-state radiation (ISR) and beam polarization[9] making Whizard the standard simulation tool for various linear collider studies [44–49].

Compared to other Monte Carlo generators, Whizard plays a minor role in LHC physics, where other programs such as Herwig [50, 51], Madgraph [52] and Sherpa [53] have distinguished themselves and have focused their efforts on improving QCD precision

[9] In general, it supports any combination of lepton and hadron beams.

for SM processes. In contrast, WHIZARD traditionally targeted beyond the Standard Model (BSM) physics, [7, 17, 54–58], and plays an important role in the feasibility studies for future colliders, such as the ILC in Japan and CLIC and FCC at European Organization for Nuclear Research (CERN).

Program Structure and Workflow

In the following we describe roughly the typical run of WHIZARD in order to determine the total cross section of a process and to generate event samples. This knowlegde will later prove as essential for understanding the implementation of parallelization and will play an important role in evaluating the efficiency of the parallelization. WHIZARD combines Monte Carlo integration and event generation in a single framework, where phase-space parameterization for each process and the various add-on treatments for event generation play an important role.[10]

WHIZARD is written in an object-oriented and modular way using the programming language Fortran 2008. Most components can be extended in a modular fashion. Examples for this approach are various interfaces for external programs for event formats (HepMC, LCIO, etc.), phase-space parameterization or the different integrators. An important point of the Fortran 2008 support is the native interoperability with `C`-interfaces and a detailed prescription for accessing the program environment and operating system.

The development of WHIZARD is test-driven: most modules are covered by unit tests against unwanted changes, and the general functionality of WHIZARD is secured by functional tests. In particular, the latest technologies such as continuous integration based on external tools like Gitlab and Docker, have proved their worth during development. My personal contribution to WHIZARD, which is not governed in detail by this thesis, are major refactoring, fixing and validation efforts for the NLO QCD production of WHIZARD. Further contributions are the implementation and validation of a PYTHIA-8 `C/C++`-interface on a single-event basis allowing for internal matching of showered events. And the implementation of the flat phase-space parameterization RAMBO proposed by [27] and an algorithmically improved version [26], where I implemented the latter. One major part of this thesis evolves around the Monte Carlo integration and event generation, for which I provided a new implementation of the adaptive Monte Carlo integrator as well as an implementation of a random-number generator for parallel application to provide correlation-free random numbers for the parallelization setup.

WHIZARD is steered by the user through the domain-specific scripting language SINDARIN, which allows the user to specify the physics model, scattering or decay processes both for integration or event generation, physical parameters and run parameters. In addition, WHIZARD offers advanced control structures such as if-clauses or scan loops. The SINDARIN command can be submitted either through a file or via command line to WHIZARD. We restrict our considerations to the case of steering by file, where the instructions of the file are processed in a top-down approach.

The workflow of WHIZARD is structured as follows. A WHIZARD run starts by setting up the physics model and the physical parameters, initializing the random number

[10] For a more detailed description of the features of WHIZARD we refer to the WHIZARD manual, or for the details of the implementation to the WHIZARD documentation, which are delivered as part of the WHIZARD package [59].

generator (RNG) with a corresponding seed. Then, the processes are initialized and the source code for the matrix-element is generated by, OMEGA[11] after which the matrix element source code is compiled and dynamically linked. The phase-space configuration is generated and written to a file. What follows is the adaptive integration pass, in which an importance sampling is performed and adapted, and a final integration pass for the total cross section. Thereafter, the previously adapted importance sampling is used to efficiently sample events. WHIZARD offers options for adding shower and hadronization with internal and external tools on a single event basis. The so-generated events are then exported to an appropriate event file format on disk, e.g. LHEF or HepMC to name the most prominent ones. The program structure is also reflected in large parts of the source code in a one-to-one mapping to the program flow.

In summary, a WHIZARD run consists of three parts: (i) initialization of physics model and parameters, and setup of the process and phase space, both involving the writing and reading of files, (ii) adaptive integration and final integration pass for the actual integration result, both involve writing a grid-file, which then holds the adaptation- and integration results, and (iii) the actual event generation, which is now as efficient as possible through the previous adaptive integration, which then allows to generate individual events and to save them on disk.

[11] The source code for the matrix elements are automatically provided by the tree-level matrix-element generator OMEGA, NLO matrix elements can be provided via the standardized BLHA interface of a one-loop provider such as RECOLA, OPENLOOPS or GOSAM.

2 Adaptive Monte Carlo Integration

In this chapter, we introduce the multi-channel approach for integration [60] and the adaptive Monte Carlo algorithms of VEGAS [61, 62] published as part of [30]. Furthermore, we outline the requirement for the optimization and the parallelization of the adaptive Monte Carlo methods with the paradigms of MPI and OpenMP within the framework of the event generator WHIZARD.

The concept of parallelization is not new to high-energy particle physics as the calculation methods are generally very time-consuming. As we elaborated in sections 1.3 and 1.4, the number of sampling points N in (1.11) directly correlates to the reduction of the variance estimate and the computation time. Or otherwise stated, the more computation time one invests, the further the variance can be reduced by the dimension-independent Monte Carlo error scaling $\propto 1/\sqrt{N}$, given that the integrand is well-behaved. Since computation time is a finite resource and not every integrand is well-behaved,[1] further improvements of the Monte Carlo error scaling with variance-reducing methods such as stratified or importance sampling are required.

A major effort of the Monte Carlo event generator community is to advance the status of the parallelization [4] of the major Monte Carlo event generators. Most have already implemented parallelization features, e.g. SHERPA [53], MG5_aMC@NLO [52] as mentioned in [4], MATRIX [63] and MCFM [64]. However, the requirements for theory predictions in the Monte Carlo event generator area are ever increasing, e.g. due to the increased measurement accuracy at the LHC, as already stated in the preface. Until the end of 2018 the LHC run-II provided a combined $160\,\mathrm{fb}^{-1}$ of integrated luminosity of events to the general-purpose experiments ATLAS and CMS. In order to meet these requirements, there is a need for optimized computation methods, but also for an efficient use of computational resources.

These computational resources can be split among grid-based batch systems and HPC systems. Both are present at research institutes like CERN or DESY. Grid-based batch systems can run an enormous amount of serial programs on a large grid of computing nodes simultaneously, because they do not require communication between each other during run-time. On the other hand, HPC systems are explicitly designed for computing-intensive problems where parallelization requires explicit communication and a high throughput of data between the computing nodes.

We present the adaptive Monte Carlo integration algorithm VEGAS for (approximately) factorizable integrands based on the methods of importance and stratified sampling. In addition, we introduce the multi-channel approach and the combination of VEGAS and multi-channel approach, the VAMP algorithm [65] in section 2.4. In table 2.1 we compare the integrated cross sections for various scattering processes, e.g. for LHC or for a future

[1] In addition to momentum conservation and on-shell conditions, phase-space integration may depend on external constraints such as phase-space cuts, as well as very much on the exact analytic structure of the matrix element. In this sense it is well-defined, but certainly not always well-behaved.

high-energy lepton collider, done with a flat phase-space parameterization using RAMBO and the adaptive Monte Carlo integrator VEGAS and with the full machinery of the doubly-adaptive multi-channel Monte Carlo algorithm VAMP. We see that the integration results are compatible within their error estimates. The doubly-adaptive approach has a faster error convergence and its efficiency is of several orders of magnitude better than the one for the flat phase-space parameterization. Thus, the use of the much-more involved VAMP algorithm is therefore clearly justified for the integration as well as for the event generation due to the much higher efficiency. For the parallelization, the double adaptation and the multiple phase-space parameterization require a detailed prescription, due to their complexity, and non-trivial communication during parallelization. Therefore, we follow the philosophy of HPC in order to provide an efficiently parallel adpative Monte Carlo integration with MPI.

In the following, we introduce the optimization of the computation time by suitable sampling techniques, i.e. multi-channel, importance and stratified sampling in sections 2.1 and 2.2, as well as in section 2.5 we discuss the parallelization schemes on the basis of shared and distributed memory systems such as MPI or OpenMP. We explain in detail the different, aforementioned algorithms, the functionality of the VAMP integration. Furthermore, we discuss the details of possible optimization in the Monte Carlo event generator WHIZARD.

Process	$\sigma_{\text{multi-channel}}$/fb	Eff. %	σ_{flat}/fb	Eff. %
$e^+e^- \to \mu^+\mu^-$	106.3864(22)	41.22(0)	106.386 45(4)	42.69(0)
$e^+e^- \to \mu^+\mu^-\gamma$	28.788(10)	7.95(25)	28.91(14)	0.112(19)
$e^+e^- \to \mu^+\mu^-\gamma\gamma$	0.9049(6)	3.95(11)	0.907(5)	0.048(12)
$e^+e^- \to \mu^+\mu^-\gamma\gamma\gamma$	0.010 873(15)	1.93(6)	0.010 82(7)	0.024(14)
$jj \to W^-(\to e^-\bar{\nu}_e)j$	9.368(6) $\times 10^5$	2.9(4)	9.39(4) $\times 10^5$	0.038(27)
$jj \to W^-(\to e^-\bar{\nu}_e)jj$	2.8813(27) $\times 10^5$	2.40(17)	2.91(4) $\times 10^5$	0.008(4)
$jj \to W^-(\to e^-\bar{\nu}_e)jjj$	7.967(13) $\times 10^4$	1.03(5)	7.62(16) $\times 10^4$	$< 10^{-3}$
$gg \to W^-(\to e^-\bar{\nu}_e)qq$	2.4106(24) $\times 10^4$	1.27(16)	2.411(20) $\times 10^4$	0.020(9)
$gg \to W^-(\to e^-\bar{\nu}_e)qqg$	9.150(15) $\times 10^3$	0.81(7)	8.86(18) $\times 10^3$	0.004(5)
$gg \to W^-(\to e^-\bar{\nu}_e)qqgg$	2.366(15) $\times 10^3$	0.152(18)	1.54(9) $\times 10^3$	$< 10^{-3}$

Table 2.1: We compare the integration results between a flat phase-space parameterization using the RAMBO algorithm with single adaptive Monte Carlo integration and the doubly adaptive multi-channel Monte Carlo integration, which will be presented in section 2.4. Both are based on the VEGAS integration algorithm, which will be presented in section 2.1 and computed with the same number of sampling points. We see that the integration results are compatible within (at worst) a 2σ range and that the doubly adaptive approach has a better error convergence, especially for processes with higher multiplicities. Furthermore, we see that the respective efficiency, as defined in eq. (1.20), for the flat phase-space parameterization is by several orders of magnitude smaller than the efficiency of the multi-channel parameterization, which is of the order of a few percent instead of promille or less.

2.1 The VEGAS Algorithm: Importance Sampling

First, we reconsider the estimate expression of the integral $I_\Omega[f]$ in (1.11). Instead of evaluating the integrand at uniformly distributed random points r_i, a more fitting probability function $G(x)$ allows us to sample the integrand more densely in regions of greater interest, e.g. where f takes large values in magnitude, the so called *importance sampling*. Hence, let us have a bijective, factorizable mapping $U \mapsto G(U) : r \mapsto G(x)$, which maps a uniformly drawn random number $r_i \in U$ to a G-distributed random variable $x_i \in G(U)$ with $G(U) \equiv U$ and $x_i = G^{-1}(r_i)$. Applying the map to the integral, we have to account for the change in the integration variable by dividing by the Jacobian $g(x)$,

$$I_\Omega[f] = \int_U f_\phi(x)\,\mathrm{d}^d x = \int_{G(U)=U} \left.\frac{f_\phi(x)}{g(x)}\right|_{x=G^{-1}(r)} \mathrm{d}^d r. \tag{2.1}$$

For a finite sample with N events, the estimator for the integral and the variance are now given by

$$E_N[f] = \frac{1}{N}\sum_{i=1}^{N} \frac{f_\phi(x_i)}{g(x_i)}, \quad \text{with}\, x_i = G^{-1}(r_i), \tag{2.2}$$

$$V_N[f] = \frac{N}{N-1}\left(\sum_{i=1}^{N}\left(\frac{f_\phi(x_i)}{g(x_i)}\right)^2 - E_N[f]^2\right). \tag{2.3}$$

The integral itself does not change under the change of variable, but the variance does depend on the particular choice of $G(x)$.

The VEGAS algorithm proposes a particular choice for the mapping G for (nearly-)factorizable functions: For each dimension $k = 1, ..., d$ the interval $(0, 1)$ is divided into n_k bins B_{j_k} with bin width $\Delta x_{k,j_k} = x_{k,j_k+1} - x_{k,j_k}, j_k = 1, ..., n_k$, respecting

$$\sum_{j_k} \Delta x_{k,j_k} = 1, \quad 0 = x_{k,0} < ... < x_{k,n_k} = 1. \tag{2.4}$$

A one-dimensional probability distribution $g_k(x_k)$ is defined as constant over a bin B_{k,j_k} and equal to its inverse bin with, $1/\Delta x_{k,j_k}$. The overall probability distribution $g(x)$ is then given by

$$g(x) = \prod_k g_k(x_k) = \prod_k \frac{1}{n_k \Delta x_{k,j_k}}, \tag{2.5}$$

if $x_k \in B_{k,j_k}$, which is also constant in the multi-dimensional bin $\prod_k B_{k,j_k}$. It is positive-definite and satisfies

$$\int_U g(x)\,\mathrm{d}^d x = 1, \tag{2.6}$$

by construction, and fulfills the requirement for the mapping G. The binnings x_{j_k} define the so-called integration grid, which has $\sum_{k=1}^{d}(n_k - 1)$ free parameters subject to the following optimization: The VEGAS algorithm adjusts the size of the bins for each bin j_k of each axis k based on the size of the following measure

$$m_{j_k} = \left[\left(\frac{\omega_{j_k}}{\sum \omega_{j_k}} - 1\right)\frac{1}{\log \frac{\omega_{j_k}}{\sum \omega_{j_k}}}\right]^\alpha = \begin{cases} > 1 & \text{increase bin size} \\ = 1 & \text{keep bin size} \\ < 1 & \text{decrease bin size} \end{cases}, \tag{2.7}$$

respecting the overall normalization. The measure is chosen such that it suppresses the adaptation for small weights ω_{j_k} avoiding rapid destabilizing changes of the bin widths. This behavior can be tuned with the free parameter α which is set to a value between 1 and 2. For values of α closer to 1, the suppression of small values is enhanced, for values closer to or equal to 2 the suppression is damped. The default setting of WHIZARD is $\alpha = 1.5$. The individual (squared) bin weights are defined as

$$\omega_{j_k}^2 = (\langle f_i \rangle \Delta x_i)^2 = \sum_{(x_i)_k \in B_{k,j_k}} \frac{f^2(x_i)}{g^2(x_i)} \rho(x_i), \tag{2.8}$$

marginalizing over all integration dimensions $k' \neq k$ when refining the bins along a dimension k. The bin weights are chosen such that the probability distribution g approaches the form of the integrand. A proof that the choice of weights actually leads to an optimal minimization of the variance for an factorizable integrand and a factorizable probability distribution as given in appendix A.1 using the method of Lagrange multipliers. As additional safety measurement against destabilizing grid changes, we average the individual bin weights over the nearest neighbors

$$w_{j_k}^2 \to w_{j_k}'^2 = \begin{cases} \frac{1}{2}\left(w_{1_k}^2 + w_{2_k}^2\right) & j_k < 3 \\ \frac{1}{3}\left(w_{j_k-1}^2 + w_{j_k}^2 + w_{j_k+1}^2\right) & 3 \leq j_k < n_k - 1 \\ \frac{1}{2}\left(w_{n_k-1}^2 + w_{n_k}^2\right) & j_k \geq n_k - 1 \end{cases}, \tag{2.9}$$

as proposed in the original VAMP implementation. The bin weights $w_{j_k}^2$ are collected as by-product while sampling events during integration.

For the event generation, we select a point component by component, which we draw from a uniformly in x-space selected bin B_{j_k}, and compute the associated probability by eq. (2.5). The event is then accepted or rejected due to the acceptance ratio

$$r_{\text{VEGAS}} = \frac{f(x)}{g(x)} \Big/ \max_x \frac{f(x)}{g(x)} \,. \tag{2.10}$$

We see again, that the efficiency is directly sensitive to the resemblance of the probability distribution to the integrand. In particular, we note that only a sufficiently well-adapted integration grid leads to an optimized efficiency of event generation. However, this requires, firstly, that the integrand should approximately factorize in its variables, and, secondly, that the integration grid should be filled and adapted with a sufficient number of sampling points. The choice of the latter is subject to the user's experience.

Furthermore, we want to examine the connection between the integration grid and the probability distribution more precisely, and the practical impact on the ratio $f(x)/g(x)$. The properties of the probability distribution in eq. (2.5) are clearly specified. We have visualized the probability distribution given by an integration grid for an real-world application, the integration of the hadronic process $\mathrm{pp} \to \mathrm{W}^-(\to \mathrm{e}^-\bar{\nu}_\mathrm{e})j$ with dimension $d = 7$, 5 for the phase-space integral of the hard process and 2 for the convolution of the process with the PDF. We computed for each dimension the (flat) probability distribution $g_k(x_k)$ based on the position and width of each bin B_{k,j_k}. From that, we determined the

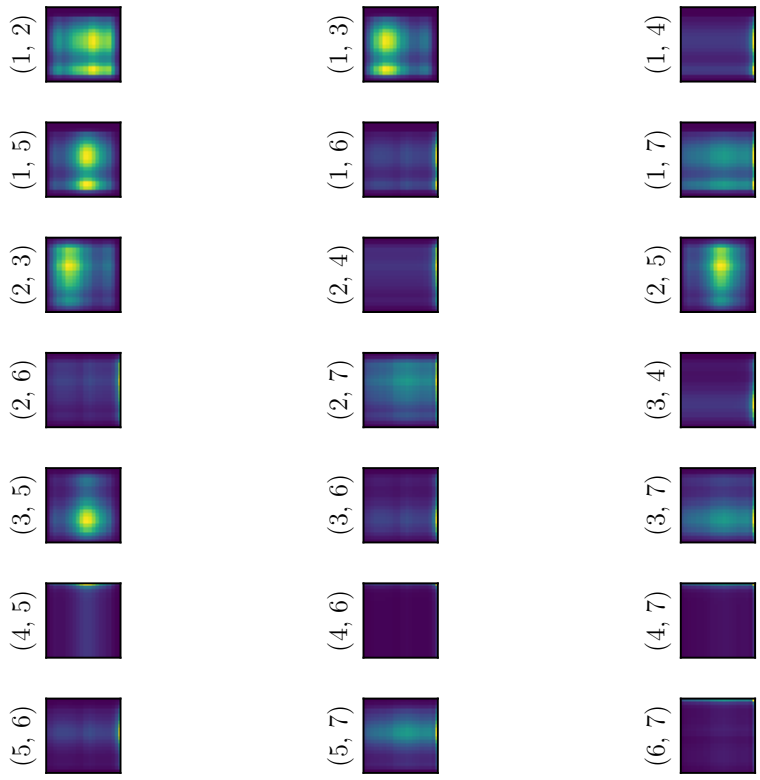


Figure 2.1: Example of an adapted grid. We show the projection of the probability distribution onto two integration dimensions $(0,1)^{\otimes 2}$, namely, the (k,k')-plane. The probability on the plane is given as the product of the probability distributions $p_{j_k}, p_{j_{k'}}$ for each bin B_{k,j_k}. The color coding follows: blue-ish for small p and yellow-ish for large p.

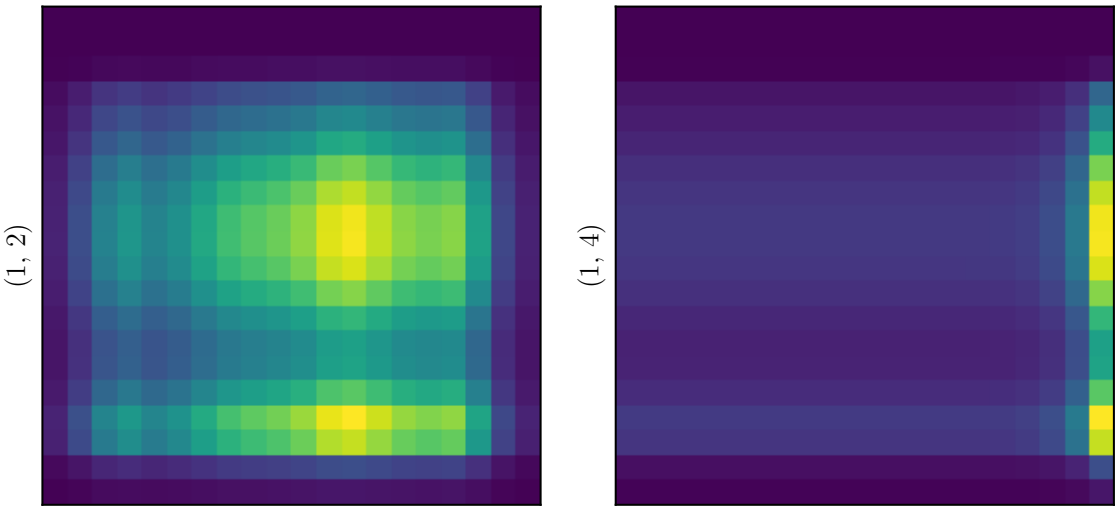


Figure 2.2: We show a zoomed version of fig. 2.1 for dimension $(1,2)$ and $(1,4)$.

two-dimensional probability distribution for dimensions (k, k') by marginalizing over all other dimensions. In fig. 2.1, all combinations of the tuple (k, k') of dimensions are plotted. We see that some combinations, e.g. $(1, 2), (2, 5), (3, 5)$ exhibit a unique structure *inside* the integration domain. On the other hand, as shown in $(1, 4)$ or $(2, 6)$, the adaptation leads to enhancements at the border of the integration domain. For better visibility, we consider the two zoomed cases: $(1, 2)$ and $(1, 4)$ in fig. 2.2. In the plot for $(1, 2)$, we see two substructures resolved in the middle of the integration domain, which are detected and adapted by the algorithm. In the plot for $(1, 4)$, we see that the algorithm has detected and adapted a structure at the border of the integration domain. The plots show that the probability distribution $g(x)$ behaves as expected.

However, the ratio $f(x)/g(x)$ exhibits a more interesting characteristic. In fact, the shape of $g(x)$ will eventually resemble a histogramed version of $f_\phi(x)$, with a saw-like profile along each integration dimension. Bins will narrow along slopes of high variation in f_ϕ, such that the ratio f_ϕ/g becomes bounded. The existence of such a bound is essential for unweighting events, since the unweighting efficiency ε scales with the absolute maximum of $f_\phi(x)/g(x)$ within the integration domain. Clearly, the value of this maximum can only be determined with some uncertainty since it relies on the finite sample x_i. The saw-like shape puts further limits on the achievable efficiency ε. Roughly speaking, each direction with significant variation in f_ϕ reduces ε by a factor of two.

2.2 The VEGAS Algorithm: (Pseudo-)Stratified Sampling

On average, sampling a function at completely random points lets the Monte Carlo integration converge proportional to $\alpha/\sqrt{N}$ independently of the number of dimensions.[2] As previously introduced, the convergence behavior can be improved with the methods of importance sampling. Instead of drawing random numbers uniformly, one draws them either from a cleverly chosen probability distribution or from an adaptively generated or improved probability distribution.

Another approach is to distribute the random numbers less randomly [66], or to ensure an even distribution among the integration domain, the so called stratified sampling.[3] The integration volume is subdivided into disjoint sub-volumes. Then for each sub-volume, the estimators for the integral and the variance are determined independently in the usual Monte Carlo way with a fixed, overall sampling number. The total integral and the total variance can then be easily determined from the individual results.

In general, stratified sampling has an equally-well convergence behavior compared to importance sampling. But choosing the sub-volumes and the sampling number of each wisely can prove to handle some integrals better, as well as some less well than importance sampling. Nevertheless, it does not only accommodate for large contributions of magnitude in the integrands, but also where the integrand is varying rapidly. Applying an adaptation prescription, e.g. using the information on the variances of the sub-volumes

[2] However, the actual convergence will still depend on the behavior of the integrand.

[3] Quasi-Monte Carlo methods [67, 68] go beyond this approach by distributing the quasi-random numbers with low discrepancy as evenly as possible among the integration domain. This approach could be used for the phase-space integration, but not for event generation as such an approach breaks with the laws of quantum mechanics.

to vary their size, one can find that the best convergence occurs when the sub-volumes are chosen in a way that the variance becomes the same for the different sub-volumes [60].

The disadvantage of stratified sampling is the requirement for an unpractically large number of total sampling points. As a short example: A subdivision of the axis $[0, 1]$ into n_{mid} cells with s sample points each, would result in d dimensions in a total sample number $N = s \cdot n^d$, which easily exceeds practical values. If we set $d = 10$, and $n_{\text{mid}} = 20$, then we get already 1.024×10^{13} sub-volumes, which need to be sampled s-times needing disproportionately much computing time in practice.

Although stratified sampling is disadvantageous in high dimensions, it is advantageous for low dimensions. The VEGAS (pseudo-)stratified approach allows us to apply the benefits of stratified sampling to high dimensions without the need of large sampling numbers and for switching to genuine stratified sampling for low dimensions.

We impose a binning along all coordinates axes producing n_r^d cells. Within each cell, we sample the integrand precisely s times at distinctive points, $s \geq 2$. The number of cells per axis is chosen such, that the total number of calls, $N = s \cdot n_r^d$, is still practical. For example, $s = 2, d = 10$, and limiting the sampling number to $N \approx 10^7$, we obtain $n_r \approx 4.676$. In each cell the sampling points are drawn randomly from a uniform distribution. In the genuine stratified mode for low dimensions, the VEGAS algorithm iteratively adapts the binning according to the variances per cell r_k

$$\omega_{r_k} = \langle f^2 \rangle_{r_k} - \langle f \rangle_{r_k}^2. \tag{2.11}$$

For high dimensions, VEGAS uses a pseudo-stratified approach. We also introduce an equidistant binning in r-space, but superimpose the integration grid in x-space. In each cell in r-space we sample precisely two points by selecting them uniformly in the r cell and map it point-wise in one of the n_x^d bins of x-space. During adaptation, the cells in r-space are unaffected and their distribution stays uniform.

In the implementation, the binning in x-space equals the binning in r-space, the integration grid itself is stratified. For the pseudo-stratified sampling, VEGAS applies a point-wise injective mapping from r-space to x-space, thus, the integration grid is stratified in an superficial way. An overview of the implementation of the VEGAS algorithm is given in the activity diagram in appendix A.2.

As short remark on the applicability of integration grids adapted by stratified sampling during event generation: For the integration both sampling methods can be used interchangeably. For the event generation we only use the probability density based on the adapted integration grid (see section 1.4 for the details of Monte Carlo event generation). The actual adaptation method is identical for both methods and it differs just in the choice of the bin weights. Therefore, both sampling methods conserve the criterions in eq. (2.4) and eq. (2.6), giving us a valid probability distribution. Concerning both method, the integration grid can be used safely in context of the event generation with the acceptance-rejection method.

2.3 The Multi-Channel Approach

In fig. 2.3 we show a simple example of electron-position annihilation into a quark-anti quark pair and an additional gluon. For the simple case, that $q^2 < M_{\text{Z}}$, we can find the

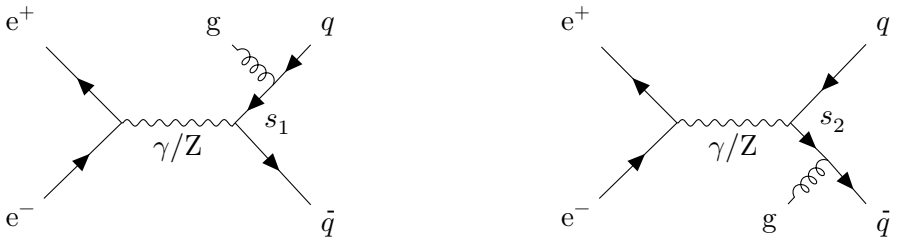


Figure 2.3: Example Multi-Channel $e^-e^+ \to q\bar{q}g$ above the Z production threshold.

averaged and summed matrix element by hand. It is proportional to the propagators of the photon q and the quark/antiquark with momenta $s_{1,2} = p_{1,2} + k$,

$$\frac{1}{4}\sum |\mathcal{M}|^2 \propto \frac{1}{q^2(p_1k)(p_2k)}, \tag{2.12}$$

where lepton and quark masses are neglected, $q^2 \gg m_\ell^2, m_q^2$. The fully-reduced matrix element has only one denominator, which is already fully factorized (in a single mapping). Above the threshold, the matrix element gains additional terms from the Z propagator and the chiral structure of the interaction vertex of Z and e^+, e^-, which then does not factorize in a single mapping. However, for importance sampling using VEGAS, we have so far assumed that the matrix element, or differential cross section, is factorizable in its variables, which as we see in general does not always apply.

A typical differential cross section exhibits singularities[4] in different phase-space regions, best accounted for with different sets of phase-space variables. The basic idea is to generate the singular structures in different mappings of the random numbers drawn from a unit hypercube. Therefore, let us introduce the multi-channel ansatz following [60]. We introduce K phase-space channels as respective mappings of the unit hypercube U to the phase-space manifold Ω, $\phi_c : U \to \Omega$ with corresponding coordinates x_c, Jacobian $|\phi_c'(x_c)|$ and densities $\rho_\phi(x_c)$ such, that $\mathrm{d}\mu(p) = |\phi_c'(x_c)|\rho_\phi(x_c)\,\mathrm{d}^d x_c$. Furthermore, given the partition of unity with the channel weights $\alpha_c \in \mathbb{R}^+$, which satisfy

$$0 \le \alpha_c \le 1, \quad \sum_{c=1}^{K} \alpha_c = 1, \tag{2.13}$$

we can introduce an intermediate function

$$h(p) = \sum_c \alpha_c h_c(p), \quad h_c(p) = \left|\phi_c'(\phi_c^{-1}(p))\right|^{-1}, \tag{2.14}$$

consisting of the Jacobian of all channels. We can then rewrite the integral expression, where we pull back from Ω to U,

$$\begin{aligned} I_\Omega[f] &= \int_\Omega f(p)\,d\mu(p) = \int_\Omega \sum_c \alpha_c \frac{f(p)}{h(p)} h_c(p)\,d\mu(p) \\ &= \sum_c \alpha_c \int_U \frac{f(\phi_c(x_c))}{h(\phi_x(x_c))} \rho_{\phi_c}(x_c)\,\mathrm{d}^d x_c. \end{aligned} \tag{2.15}$$

[4]For the ease of readability, we simply refer to both peaks and genuine singularities as "singularities".

The mappings ϕ_c are then chosen in such a way, that each singularity is canceled at least by one Jacobian $|\phi_c'|$. If we assume, that the Jacobians $|\phi_c'|$ tend sufficiently fast to zero in the vicinity of their respective singularity, then the effective integrand

$$f_c^h(x_c) = \frac{f(\phi_c(x_c))}{h(\phi_c(x_c))}\rho_{\phi_c}(x_c), \tag{2.16}$$

reduces approximately at a singularity $\tilde{x}_c$ to

$$f_c^h(\tilde{x}_c) \approx \frac{f(\phi_c(\tilde{x}_c))}{\alpha_c|\phi_c'(\tilde{x}_c)|}. \tag{2.17}$$

Each effective integrand, which depends on all weights α_c and Jacobians $|\phi_c'|$, is then integrated in its associated phase-space channel, and the total integral can be obtained, by summing the α_c weighted results.

The usual Monte Carlo method can then be applied,

$$E_N[f] = \sum_c \alpha_c \frac{1}{N_c} \sum_{i_c=1}^{N_c} f_c^h(x_{c,i_c}), \tag{2.18}$$

where the total number of samples N is distributed among the channels, $N = \sum_c N_c$. A particular choice for the weights is to correlate them and the number of samples N_c, $N_c = \alpha_c N$, thus

$$E_N[f] = \frac{1}{N} \sum_c \sum_{i_c=1}^{N_c} f_c^h(x_{c,i_c}). \tag{2.19}$$

For each channel, the points x_{c,i_c} can be taken independently as uniformly distributed random numbers. Or, one may apply stratified sampling within each channel.

The channel weights α_c are free parameters and the integral does not depend on the actual choice of the channel weights and/or the mappings, in contrast to the variance. Therefore, the choice and the method of optimization of the weights play an important role for minimization of the variance. The optimization rule, which minimizes the variance based on the channel weights, can be derived by the method of Lagrange multipliers (see appendix A.3). The result is that a minimum[5] of the variance is achieved when all channel variances become equal.

At first, we may start the weight optimization from a uniform distribution, hence, $\alpha_c = 1/K$, and adapt them iteratively. The channel weights are updated by

$$\alpha_c \to \frac{\alpha_c V_c^\beta}{\sum_c \alpha_c V_c^\beta}, \tag{2.20}$$

with an independent parameter β [60],[6] where the V_c denote the channel variances.

We want to note that the update of the weights in eq. (2.20) following the argument in section 2.3 does not necessarily converge to a global fixed point, but may only achieve

[5] It is not possible to argue that the optimization will lead to an global minimum, but at least to *a* minimum.

[6] We choose $\beta = 1/4$, however, the parameter can be changed by the user.

a minor level of local optimization. Nonetheless, the update prescription will focus on channels with large contributions to the variance. Furthermore, practical application shows the update prescription works rather well. In order to dampen statistical fluctuations, the free parameter β can be adjusted to smaller values. Furthermore, it may be useful to implement a safe-guard for when a channel weight becomes small and, therefore, the corresponding number of samples N_c is getting small. In that case, we may either choose to switch off the channel, $\alpha_c = 0$, or introduce a threshold value for N_c. However, the choice of such a safe-guard is up to the implementation, e.g. in WHIZARD a lower threshold value for N_c is enforced by reweighting the channels weights α_c accordingly such that the condition $\sum_c \alpha_c = 1$ is conserved.

2.4 Doubly-Adaptive Multi-Channel Integration: VAMP

Connecting the multi-channel ansatz with VEGAS leads us to the VAMP algorithm, introduced in [65]. In a similar way as before, we introduce K bijective mappings G_c of the unit hypercube onto itself $U \to G_c(U) = U$. Furthermore, we constrain $G_c(x)$ with the Jacobian $g_c(x_c) = G_c'(x_c)$ and the normalization condition

$$\int_U g_c(x_c)\,\mathrm{d}^d x_c = 1, \tag{2.21}$$

for all channels. We can then identify $g_c(x_c)$ as the probability distribution given in eq. (2.5) by VEGAS, and chain G_c with the channel mappings ϕ_c in eq. (2.14). The integral expression can then be reformulated,

$$I_\Omega[f] = \sum_c \alpha_c \int_U f_c^g(x_c)|_{x_c = G_c^{-1}(r_c)}\,\mathrm{d}^d r_c, \tag{2.22}$$

where the modified effective integrand for the channel c is given by

$$f_c^g(x_c) = \frac{f(\phi_c(x_c))}{g(\phi_c(x_c))}\rho_{\phi_c}(x_c). \tag{2.23}$$

The overall probability density, replacing $h(p)$ in eq. (2.16), is

$$g(p) = \sum_c \alpha_c \frac{g_c(x_c)}{\phi_c'(x_c)}|_{x_c = \phi_c(p)}. \tag{2.24}$$

The variance can be reduced in a two-fold way: By choosing a suitable mapping ϕ_c, but with fixed Jacobain ϕ_c', and also by the tune-able g_c distributions. In regions, where the contribution of g_c gets large in magnitude, we find that eq. (2.23) reduces approximately to

$$f_c^g(x_c) \approx \frac{f_{\phi_c}(x_c)}{g_c(x_c)}, \tag{2.25}$$

in analogy to eq. (2.14).

The integral estimate for the combined approach is given by

$$E_N[f] = \frac{1}{N}\sum_c \sum_{i_c=1}^{N_c} f_c^g x_{c,i_c}, \quad x_{c,i_c} = G_c^{-1}(r_{c,i_c}). \tag{2.26}$$

The variance can then be minimized by optimizing the free parameters independently, thus, the channel weights α_c accordingly to eq. (2.20) and the respective channel grids by eq. (2.7).

We want to note that the multi-channel approach can scale the number of channels, and therefore, integration grids, up arbitrarily.[7] Normally, the increasing number of free parameters of the optimization will then face a constant number of sampling points, which can result in insufficiently filled integration grids. Such critical grids can then adapt inefficiently and spoil the combined ansatz. In order to improve the adaptation in such cases, we define equivalences for channels that share a common structure. These equivalences allow adaptation information, i.e. the individual bin weights w_{j_k} of each axis, to be averaged over several channels, which in turn improves the statistics of the adaptation. Such an equivalence maps the individual bin weights of a channel c' onto the current channel c together with the permutation of the integration dimensions d, $\pi : c \mapsto c' : k \mapsto \pi(k)$, and the type of mapping. The different mappings used in the algorithm are:

$$\textbf{identity} \qquad w^c_{j_k} \to w^c_{j_k} + w^{c'}_{j_{\pi(k)}}, \tag{2.27}$$

$$\textbf{invert} \qquad w^c_{j_k} \to w^c_{j_k} + w^{c'}_{d-j_{\pi(k)}}, \tag{2.28}$$

$$\textbf{symmetric} \qquad w^c_{j_k} \to w^c_{j_k} + \frac{1}{2}\left(w^{c'}_{d-j_{\pi(k)}} + w^{c'}_{d-j_{\pi(k)}}\right), \tag{2.29}$$

$$\textbf{invariant} \qquad w^c_{j_k} \to 1. \tag{2.30}$$

Interesting applications are those that require very high statistics. Channel equivalences have been shown to play a crucial role in sampling correctly particularly the less densely populated regions of phase space, e.g. in vector-boson scattering [6, 7, 17, 19, 49, 56, 57] or in BSM simulations with a huge number of phase space channels [55, 69–71]. Constructing these channel equivalences is part of the phase-space algorithm, but a detailed explanation is out of scope of this thesis.

We employ two steps for the event generation after the multi-channel integration: Firstly, we select a channel c by the probability p_{channel} on $\sup_c \sum_{c'=1}^{c} \alpha_{c'} \leq p_{\text{channel}}$ and, secondly, we generate and accept an event in the respective channel c,

$$r'_c = \frac{f(\phi_c(x_c))}{h(\phi_c(x_c)} \Big/ \max_{c',x_{c'}} \frac{f(\phi_{c'}(x_{c'}))}{h(\phi_{c'}(x_{c'}))}\,, \tag{2.31}$$

where $\max_{c,x} f(\phi_c(x_c))/h(\phi_c(x_c))$ denotes the overall maximum weight of *all* channels. However, to simplify the implementation where we want to generate an event in the selected channel c with VEGAS, we have to replace the total maximum weight with the individual maximum channel weights. In order to do so, we reweight the channel weights (only for event generation) with the individual channel maximum weights over the overall maximum weight,

$$\alpha_c \to \frac{\alpha_c \max_{x_c} f(\phi_c(x_c))}{\max_{c,x_c} h(\phi_c(x_c))} \Big/ \sum_c \frac{\alpha_c \max_{x_c} f(\phi_c(x_c))}{\max_{c,x_c} h(\phi_c(x_c))}\,, \tag{2.32}$$

[7] The actual number of channels is either up to an algorithm or user input, and is in particular neither restricted by the multi-channel ansatz, nor by the VEGAS.

from where the dependency on the overall maximum weight then drops. After selecting a channel with the reweighted channel weights, we can proceed to generate and accept an event in the selected channel,

$$r_c = \frac{f(\phi_c(x_c))}{h(\phi_c(x_c)} \Big/ \max_{x_c} \frac{f(\phi_c(x_c))}{h(\phi_c(x_c))} \,, \tag{2.33}$$

with the same procedure as in VEGAS.

2.5 Optimization and Parallelization

Regarding particle-physics applications, we are met with the ever increasing demands on the capabilities of modern Monte-Carlo event generators, e.g. on the precision in data analysis at the LHC and planned future high-energy and high-luminosity lepton and hadron colliders. Evolving from simple $2 \to 2$ or $2 \to 3$ processes, where the amplitudes are hard-coded and already optimized, e.g. Pythia 6 [72] or its successor Pythia 8 [29], the state-of-the-art simulations for LHC have been made with automated multi-leg tree-level processes, which are capable of simulating $2 \to n$ scattering processes, where $n = 6$, 8 and 10. However, data analyses with higher precision at the major experiments of the LHC, such as ATLAS or CMS, require often to go beyond multi-leg tree-level processes, hence, computations shift towards automated NLO (or even higher orders) processes for integration and event generation at fixed-order and a corresponding matching routine, e.g. POWHEG method [73]. Compared to multi-leg tree-level calculations, automated NLO computations are currently limited to smaller multiplicities $n \leq 5$, but they introduce (computing-intense) terms such as the real (with multiplicity plus one) and a virtual correction terms, as well as numerical stabilizing subtraction terms, which are used to subtract the occurring divergences on the individual terms, separately[8]. Combined with the expanding possibilities of multi-processing resources and high-performance computing systems, we are forced to examine the possibilities of optimization and/or parallelization of our computation tools.

We can employ two approaches: reducing the actual central-processing unit (CPU) time spent by optimization or reducing the wall time by parallel evaluation of the computation tasks.

Optimization

First, we want to discuss the possibility of optimizing computational methods. The optimization of the Monte Carlo integration and event generation has been discussed thoroughly in the previous sections. Next, we consider the optimization of the matrix element computation, as well as the phase-space generation: The number of Feynman graphs contributing to a physical process grows approximately factorially with the number of outgoing particles.

The generation of efficient code for the matrix-element computation is covered by specialized matrix-element generators, e.g OMEGA for leading order (LO), [34], or RECOLA for LO/NLO, [74, 75].

[8]The divergences cancel in a full analytical calculation of both real and virtual corrections, but need to be regularized in a numerical approach.

OMEGA [76, 77] constructs multi-leg tree-level amplitudes not directly from Feynman diagrams as it proves to be that the direct Feynman diagram approach introduces many redundant terms. Instead, OMEGA utitlizes *One Particle Off-Shell Wave* (1POW) functions

$$W(x; q_1, ..., q_m, p_1, ..., p_n) = \langle \phi(q_1), ..., \phi(q_m); \text{out} | \Phi(x) | \Phi(p_1), ..., \phi(p_n); \text{in} \rangle \,, \qquad (2.34)$$

in order to express sums of Feynman diagrams as 1POWs. Further reduction of the algebraic expression of the scattering amplitudes is achieved with help of direct acyclical graphs (DAGs), e.g. where expression of the form $ab + ac$ are reduced to $a(b + c)$ saving one multiplication. In this form, the computation of the matrix elements can be put in its most optimized form, resulting in an exponential scaling law.

Further improvements of OMEGA are made with regard to the form of the matrix element as machine interpretable code. OMEGA can either provide the matrix element as Fortran source code which then needs to be compiled or in the form of a self-provided byte-code syntax which is then interpreted by the OMEGA virtual-machine [78, 79]. The byte-code based matrix-element computation performs comparable to the Fortran-based and compiled matrix element. The compilation time of the Fortran code can consume a considerable amount of run-time and can exceed the available memory for multi-leg processes with large $n = 8$ and 10. In contrast, the byte-code does neither require a compilation as it is interpreted just-in-time nor does the interpreter demand vast amounts of memory.[9] Thus, the OMEGA virtual-machine pushes the boundary for an optimized computation of high-multiplicity multileg tree-level processes further.

Regarding the multi-channel phase-space parameterization: At tree level, the analytical structure of the matrix elements is dominated by particle propagators going on-shell.[10] From the Feynman diagrams we can then construct a phase-space configuration using particle information, i.e. their masses. However, with a one-to-one translation of the Feynman graphs into integration channels, the number of channels would grow factorially, too. Additionally, if the multi-channel approach is combined with the adaptive binning of VEGAS, the number of channels is then multiplied by the number of grid parameters, which considerably complicates the overall optimization of integration grids as it would require a tremendous number of sampling to cover all of the free parameters efficiently. Furthermore, the calculation of the channel Jacobians, and the lookup of the grid probabilities, consume a considerable fraction of the overall computation time, rising again with the number of channels. In order to limit the number of integration channels, a heuristic algorithm searches for dominant parameterization [80] or [5, see the WHIZARD manual] whose description goes beyond the scope of this thesis.

The original implementation of the phase-space generator of WHIZARD, which constructs the channel paramerterizations from an internal construction of relevant Feynman graphs, is called *wood*, and uses the internal model information to find resonant propagators, actual mass values in order to identify collinear and soft singularities and to map mass edges, as well the process energy. However, the internal construction of Feynman graphs, although implemented recursively, is inefficient and memory-consuming. Therefore, a new algorithm *wood2* has been designed, which relies on the already optimized

[9] For the actual matrix-element computation the memory footprint of the byte-code interpreted matrix-element is slightly larger than for the Fortran-based, compiled matrix element.

[10] In addition, logarithmic radiation of massless particles or t-channel diagrams and further contribute.

DAG representation of the Feynman graphs from OMEGA. It applies the elimination and simplification algorithm of *wood* on the DAG represented Feynman graphs, and combines them with the corresponding numerical values for the masses and the colliding energy. The new algorithm is on average 100 times faster as can be seen in fig. 2.4.

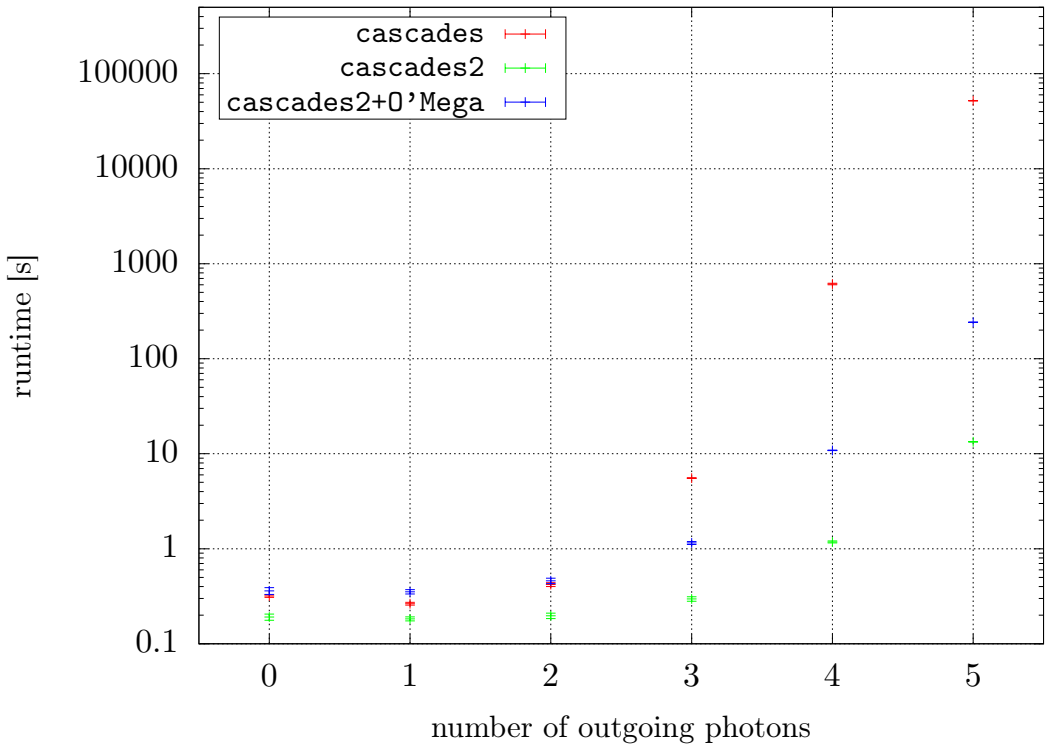


Figure 2.4: Comparison of the phase-space configuration generators: *wood* and *wood2* [80]. The *wood* algorithm is based on the `cascades` implementation, whereas the *wood2* is based on `cascades2`. The run-time for the phase-space configuration is shown for both. Furthermore, we compare them with the run-time of `cascades2` and the preceding, necessary OMEGA run. Taken from the master thesis [80]. The English-translated version of the plot was provided by Manuel Utsch.

A final remark: The particular choice of the channel parameterization does not change the limit $E_\Omega[f] \to I_\Omega[f]$, but it affects the variance and it may improve or slow down the convergence of the integral estimate.

Parallelization

So far, we have sufficiently exhausted the possibilities of optimization, but the computation tools are still bound to the resources of a single CPU machine. Further reduction of the computation time can be reached with parallelization: The distribution of the computation on different computing resources, e.g. CPUs with multiple cores or machines of a high-performance computing cluster. The basic methods of parallel evaluation can be divided

among two paradigms: shared memory and distributed memory, where also a combination of both approaches is possible.

Shared memory systems allow multiple processes to share memory addresses allowing processes to access or manipulate data of other processes in the same place and avoiding copies of memory. Due to the design of shared memory, race conditions can occur where several processes try to access or manipulate data at the same memory address. A typical application of the shared-system paradigm is threading: A process spawns threads, which are confined to their respective parent process, and allows computation tasks to run concurrently on the threads, where the parent process and threads share memory. There exist several application programming interfaces (APIs) with the better-known being the POSIX thread standard or the OpenMP standard.

For independent processes, each operating locally on data, access to remote data requires communication. Such distributed memory systems require sophisticated communication, which depend on the physical side crucially on the network topology and scalability. However, distributed memory systems allow significantly higher scalability of computational resources, as opposed to shared-memory systems that rely on particular hardware and software to scale accordingly, which are mostly bound to a single machine with multiple-core CPUs.

In the following, we give an overview of the different protocols, in particular their strategy, native language support, embedding into the programs, and hardware dependence.[11]

1. Message-passing interface (MPI) [81, 82] is a standard associated to the distributed-memory paradigm. It introduces the concepts of abstract workers and processes, such that the high-level implementation is independent of the detailed nature of the underlying hard- and software. Each worker is assigned a process running the same program concurrently. Remote data requires communication which is introduced in a set of communication procedures and message handling. Furthermore, in MPI-2 the notion of parallel input/ouput (I/O) was introduced, mostly supported by file formats like the hierarchical file format, version 5 (HDF5). In MPI-3, the standard introduces non-blocking communication for all communicative procedures allowing for concurrent communication and computation. It provides native Fortran 2008 support and the implementation of the APIs is deferred to a library, e.g. OpenMPI, MPICH, or Intel MPI. The standard does not provide any parallelization concepts, only the framework for the communication, the concept is up to the respective developer. The MPI protocol is particularly well suited for HPC clusters with many independent, but interconnected CPUs with a well designed network.

2. Open multi-processing (OpenMP) [see 83, 84] is a shared-memory based parallelization model that uses threads. OpenMP allows during program execution to switch between sequential and parallel stages of the program flow using a master-slave approach, where the master coordinates the distribution of computing resources and the synchronization between the stages. A disadvantage of the shared-memory model is that the execution of atomic instructions, such as read and write access

[11] We are particularly interested in the Fortran support, which is the language of choice for our numerical applications.

to memory, can lead to race-conditions and deadlocks of the program flow. The standard provides an explicit parallelization concept, which can be implemented via directives in the source code by the developer, which is then realized automatically by an OpenMP-compatible compiler.

3. Multithreading with the POSIX Thread Standard [85] is based on operating system features and does not evolve itself around a sophisticated parallelization concept. It requires a sophisticated implementation by the developer realizing higher concepts of parallelization on its own, for example, resource manager must be derived.

4. Coarrays [e.g. see 86, 87] are an approach to native parallelism in the Fortran 2008 standard. Its semantics combine the concepts of MPI and OpenMP, hence, it provides concepts for independent workers, where data can be accessed in shared-memory fashion or requires communication for remote data. However, the support by different Fortran compiler is rather poor.

5. CUDA [88] is a software-layer that gives direct access to graphical processing units (GPU) as highly valuable multiprocessing units for parallelization of floating-point computation with a small memory footprint.

The current strategy for the parallel evaluation within WHIZARD involves MPI and OpenMP, either separately or in combination. We do not make use of the other protocol, as they are either insufficiently supported by compilers (Coarrays) or are superseeded by an easier paradigm (POSIX), or our problems do not fit on to the prerequisites for a protocol, e.g. CUDA.

An example for existing parallelization efforts within WHIZARD is the parallelization of the matrix-element computation, which is based on the concurrent evaluation of helicity amplitudes with OpenMP multi-threading.

We rely on MPI for the parallel implementation of the integrator VAMP2 (see section 3.1), however, minor parts can be parallelized with OpenMP, e.g. computation of the overall probability in eq. (2.24), or the parallel evaluation of the helicity amplitudes of the OMEGA amplitudes.

Ahmdal's and Gustafson's laws

We assume that we can separate a program into a parallelizable fraction with run-time T_p, which can be performed concurrently by n workers, and a serial fraction with run-time T_s, the same for all workers. Additionally, we make the communication $T_c(n)$ time explicit, which we expect to vary on the used algorithm and to be dependent on the number of workers. The total computation time can then be written as

$$T(n) = T_s + \frac{T_p}{n} + T_c(n). \tag{2.35}$$

We expect the communication time to vanish for $n = 1$ and to increase monotonically in n, e.g. $T_c \sim \log(n)$ or $T_c \sim (n-1)^\alpha, \alpha > 0$.

The *speedup* factor of the parallelization is given by

$$f(n) = \frac{T(1)}{T(n)} = \frac{T_s + T_p}{T_s + \frac{T_p}{n} + T_c(n)}. \tag{2.36}$$

In the ideal case, that T_s and T_c vanish,

$$f(n) = n, \tag{2.37}$$

and as long we can neglect T_c, the speedup is bounded by

$$f(n) \leq \left(1 + \frac{T_p}{T_s}\right) \Bigg/ \left(1 + \frac{T_p}{n \cdot T_s}\right) \leq 1 + \frac{T_p}{T_s}, \tag{2.38}$$

but in practice we can neglect neither T_s nor T_c. Eventually communication increases so much that the speedup vanishes,

$$f(n) \approx \frac{T_s + T_p}{T_c(n)} \to 0, \tag{2.39}$$

limiting the number of workers that we can efficiently employ for a given task.

We can summarize the challenges of the parallelization effort: Firstly, we want to increase the fraction T_s/T_p that determines the critical behavior for large n, by parallelizing all computing-intense parts of the program as thoroughly as possible. For example, if the serial fraction is 0.1 %, then we already reach a speedup plateau at around $n = 100$ workers. Secondly, we have to make sure that the communication time T_c remains negligible until we reach this saturation point either by choosing a communication algorithm with a low power of n in $T_c(n)$, or by reducing communication where applicable and preventing communication-related blocking by applying non-blocking communication in combination with a feasible parallelization model.

The effectiveness for the parallelization is determined by benchmarks, where we examine the computation time of a fixed problem for a varying number of workers. We then compare the speedups of the benchmark with the ideal prediction of the speedup in Ahmdal's law [89]. For this, we neglect the communication time and write the time executed by the parallelization part as fraction $p \cdot T$ of the total time $T = T_s + T_p$ of the serial process. The speedup in eq. (2.36) translates into

$$f(n) = \frac{1}{(1-p) + \frac{p}{n}} \overset{n\to\infty}{\longrightarrow} \frac{1}{1-p}. \tag{2.40}$$

Again, in the limit of infinite resources, the speedup reaches a plateau given exactly by the serial portion $s = 1 - p$.

The pessimistic view of Ahmdal's law can be contrasted with Gustafson's Law [90]. The main critique point is that in practice the parallel fraction T_p is not independent of n. Furthermore, in practice, no one would compare the run-time of a fixed problem with varying number of processes, except for academic research. In fact, we expect the problem and the number of processes to scale together, as a user would tune a parallel run. So, if we compare the run-time of a parallel evaluation $T_s + T_p$, and compute the run-time of a serial run as $T_s + n \cdot T_p$, we find that the speedup is given by

$$g(n) = \frac{T_s + n \cdot T_p}{T_s + T_p} = n + (1-n) \cdot \underbrace{\frac{T_s}{T_s + T_p}}_{\equiv s = (1-p)}. \tag{2.41}$$

Compared to Ahmdal's law, the scaled speedup depends linearly on the serial time, with a moderate slope $1 - n$. It is thus much easier to achieve an efficient parallelization than suggested by Ahmdal's law.

However, we want to assess the worst case of our parallelization. Therefore, we resort to Ahmdahl's law in order to estimate a lower bound on the speedup regarding a fixed-size problem. In a real application, the computational resources should be adapted to the present (integration) problem resulting in a more efficient parallelization on average due to being a scaled-size problem. We close this discussion with the memorable statement that "the speedup should be measured by scaling the problem, not by fixing the problem size" [90].

2.6 The Message-Passing Interface (MPI) Standard

In section 2.5, we briefly mentioned the message-passing interface standard. In the following, we give a short introduction to shed more light on the relevant terminology for our implementation, as well as the standard itself.

The MPI standard is very general in its applications, it specifies a large amount of procedures, types and memory-, process- and error-related management of different purposes. However, we limit the wide range of functionality, which obscures a clear view on the problem of parallelization and unnecessarily complicates the problem itself, to the absolute minimum, which we need for the implementation of our parallelization. For example, we do not use the MPI shared-memory capabilities or the MPI process management for the implementation of a dynamic server-client model. For clarity in the discussion of implementation later on, we refer to MPI processes as *worker* in order to not confuse them with physical processes (in context of WHIZARD).

For MPI programs, the standard specifies that the processes are autonomous, in principle each running its own code, and run in a MIMD[12] style [cf. 82, p. 20]. The definition of communicators [82, pp. 223–257] allows to abstract processes from the underlying hardware and network topology, and to separate communication of different program parts or libraries. The latter is achieved by the introduction of communication context securing that sending and receiving messages in a given one does not interfere with messages in other contexts. Communicators group processes (a so called process group) as ordered set with assigned ranks (labels) $0, ..., n - 1$. The MPI standard predefines the `MPI_COMM_WORLD` which contains all processes known at the initialization of a MPI program (done by `MPI_INIT`) in a linearly ordered fashion.

In most cases, the underlying architecture of the hardware and the network infrastructure is not reflected by the linearly ordering of the process group, therefore, the standard defines a process topology mechanism [82, pp. 289–332] which allows to order the processes in such a way that the communication pattern is reflected in a virtual topology, and thereby the communication is optimized. The virtual topologies can also be exploited by the operating system to improve the distribution of processes to the hardware, and it must not be confused with a hardware topology. Importantly, a virtual topology is introduced principally to optimize communication by adjusting the process ranking through the

[12] Multiple instruction, multiple data. In practice, MIMD have a number of processes running asynchronously and independently.

virtual topology to match the communication pattern. The application of such topologies is useful, for example, in the case where the next neighbor needs to exchange information, e.g. solving partial differential equations, in contrast, during Monte Carlo sampling there is no need for communication, only before and after, for the adaptation step. A specific profiling of the run-time could reveal special topological structures in the communication which might offer potential for further improvement, but for the time being, we neglect the optimization through topologies. Lastly, all newly created communicators inherit from the process group and context definition of `MPI_COMM_WORLD`.

Messages are passed between sender(s) and receiver(s) inside a communicator or between communicators where the following communication modes are available:

non-blocking A non-blocking procedure returns directly after initiating a communication process. The status of communication must be checked by the user.

blocking A blocking procedure returns after the communication process has completed.

point-to-point A point-to-point procedure communicates between a single receiver and single sender.

collective A collective procedure communicates with the complete process group. Collective procedures must appear in the same order on all processes.

The standard distinguishes between blocking and non-blocking point-to-point or collective communications. A conciser program flow and an increased speedup are advantages of non-blocking over blocking communication.

In order to ease the startup of a parallel application, the standard specifies the startup command `mpiexec`. However, we recommend the de-facto standard startup command `mpirun` which is in general part of a MPI-library. In this way, the user does not have to bother with the quirks of the overall process management and inter-operability with the operating system, as this is then covered by `mpirun`. Furthermore, most MPI-libraries support interfaces to cluster software, e.g. Slurm, Torque, HTCondor.

In addition, MPI supports parallel input/output (I/O) [82, pp. 491–560]. The standard defines a high-level interface supporting partitioning of file data among processes and a collective interface supporting complete transfers of global data structures between process memories and files. In addition, the standard allows for further improvement of the efficiency by supporting asynchronous I/O, strided access and control over physical file layout on storage device. Although the standard provides a more abstract approach than a low-level approach with basic tools from POSIX standard, the complexity of handling parallel I/O is still enormous. Well-defined file formats such as the HDF5 support parallel I/O based on the MPI standard. We point out that an implementation of parallel I/O for the self-defined file formats of Whizard and Vamp2 is meaningless, firstly, due to their arbitrary definition and that such an implementation would require large amount of development time and personal resources. Therefore, we should fall back on external, well-defined file formats.

In summary, we do not use process and (shared-)memory management, topologies, parallel I/O and the advanced error handling of MPI.

3 Parallelization of the Adaptive Monte Carlo Integration

In this chapter, we present our results on the implementation of the parallel adaptive multi-channel Monte Carlo methods using MPI which was published as part of [30]. The results are presented within the framework of the event generator WHIZARD.

In the previous chapter 2, we have introduced the adaptive multi-channel Monte Carlo method VAMP and the parallelization paradigms of MPI and OpenMP. In this chapter, we reason in detail the need for a modern re-implementation as the VAMP2 integrator in the Monte Carlo event generator WHIZARD. The advantage of the new implementation VAMP2 over VAMP is the complete object-oriented programming approach, the dynamic memory management, the inclusion of arbitrary random number generators (as available in WHIZARD), whereas VAMP is limited to its own supplied random number generator. Most important, the new implementation has a working parallelization scheme based on the third MPI standard, which allows for a strong-typed interface to any MPI-3.1 compatible library and the application of non-blocking collective communication. However, most benefits are based on the application of more modern Fortran 2008 compared to Fortran 90 and the more comprehensive documentation of the source code.

We discuss the applications of random number generators in parallelized applications and introduce the RNGstream [91] which allows us to generate random numbers without any need of communication. We then present in detail the parallelization of the adaptive Monte Carlo algorithms for the integration, VEGAS and VAMP, respectively, as well as their implementation in VAMP2 and WHIZARD with a simple and rather naive approach for load-balancing between the two algorithms. The load balancing is necessary for a efficient parallelization as we need to cover the full range of multi-channel and worker configurations for the parallel integration, e.g. from single to $\mathcal{O}(1000)$ channels in the integration with several to hundreds of workers for the parallelization. In addition, we present new results, that were not part of [30], namely, a static load-balancer to further advance the efficiency of our parallelization effort.

3.1 A New Implementation: VAMP2

The implementation of the VAMP algorithm is packaged as part of WHIZARD [59], but is also available as standalone package. VAMP ist written as standalone library which can be interfaced by any Fortran 90 compatible program. Furthermore, we refer out of simplicity to the library itself as VAMP integrator. The original implementation was written by Thorsten Ohl and has been slightly extended over time to the additional requirements of WHIZARD, e.g. the addition of channel equivalences. The implementation has been written in Fortran 90 and utilizes an object-oriented approach as well as some concepts of functional programming, as far as Fortran 90 allows for such programming constructs. Memory management for dynamically allocatable arrays is done with Fortran

pointers. The latter do not necessarily force the compiler to implement dynamic arrays in a very efficient way. In general, pointers are used basically for different task, but in Fortran 90 they were exploited[1] to provide allocatable memory (as in languages like C). The programmer alone was responsible for the correct handling of allocated memory, i.e. referencing and explicit deallocation.

Furthermore, the implementation of VAMP, which in itself is fully functional, is only in parts well documented and many of the key details of the algorithm are lost in structure and incomprehensible ways of the source code, which make maintenance and extensibility a rather tedious task. In addition, there existed also a duplicate version of VAMP called VAMPI, which implemented a parallel version of the VAMP algorithm using a first version of the MPI standard. Unfortunately, this version was not kept up-to-date and is no longer functional in the sense that it is not compatible anymore with the current version of the VAMP implementation. Again, the documentation is insufficient, moreover, the structure of the source code is confusing and in parts incomprehensible. This is not a direct criticism on the VAMP package, but it was an insurmountable obstacle that we encountered trying to parallelize the original VAMP algorithm. These and the need to understand the algorithm at all depths have led us to write a new implementation.

We implement the VAMP algorithm from scratch in Fortran 2008 as VAMP2, with all supported public features of VAMP, as part of the WHIZARD source code. For the implementation of VEGAS, we followed the concepts of the original source code of Lepage [61] and the `C` version in the GNU Scientific Library [92].

VAMP2 is designed as independent library and is fully programmed in an object-orientated fashion based on the object-orientated features of Fortran 2008, where we have defined various objects: containers for the configuration parameters, for the integration results and for the integration grid itself and as well as a separate data types for the integrators. The new implementation explicitly distinguishes between the two integration algorithms, VEGAS and VAMP, and we implement them as separate integration modules, which leaves us with two encapsulated integration entities, i.e. two integration libraries to which we refer again as integrators, respectively.[2] The object-oriented approach allows us to reuse the VEGAS integrator as an encapsulated object in the VAMP2 integration module, this reflects the natural dependence of the two algorithms. Furthermore, we either extended different structures of VEGAS by inheritance to VAMP2 (for the configuration and result containers), or define a new data type for the VAMP2 integrator. All arrays are dynamically treated as allocatable, which ensures that all arrays are contiguous and are always correctly allocated and deallocated. This allows the compiler to make certain assumptions for further optimization of the compiled code. Therefore, we expect a small (but hardly measurable) speed advantage due to the use of allocatable versus pointer arrays, as well as with full object-oriented typing.

The benefit of the Fortran 2008 implementation, besides the typical advantages due to object-oriented programming and allocatables, is that we can implement a direct `C`-interface (and thus an interface for `C++`) based on the Fortran 2008 standard, which

[1] The Fortran 90 Standard introduced pointers. Before, there was no way for a Fortran programmer to do any type of memory operation during run-time.

[2] For the sake of simplicity, we refer to the specific implementation of a single integration module as an integrator.

makes the integrator accessible to a broader user base.[3]

VAMP2 is shipped as part of the WHIZARD source code available in [5].[4] A future standalone packaging is planned with an additional `C++` interface allowing a variety of external tools to access the VAMP2 integrator and most importantly to make the here presented parallelization feature accessible to the public.

3.2 Random Numbers and Parallelization

The backbone of the afore-mentioned Monte Carlo method is the underlying generation of random numbers which allows us to compute an integral without detailed knowledge about the integrand with a convergence approximately to $\alpha/\sqrt{N}$ to the true value, where α denotes the contribution to the variance due to the integrand. Without loss of generality, we have always assumed that a "black box"-generator produced a sequence of uniform random numbers in the open interval $(0, 1)$ with appropriate "random" properties.

The quality of random numbers such as uniformity or correlation have an important impact on the integration result and the variance of the Monte Carlo approach. A deviation from the uniformity without corresponding compensation of the integration measure can lead to a wrong estimate of the integration results and the variance. Whereas correlations in the random numbers have no effect on the integration result itself, with them we may underestimate or overestimate the variance as our assumption of independent random numbers for the calculation of the variance in eq. (1.12) is no longer valid. Furthermore, in both cases, such "bad" random numbers can not be used for event generation. Quantum mechanics predicts that the events occur independently of each other and "randomly" (according to the underlying dynamics), these principles are broken by such bad random numbers.

In order to respect these basic principles of quantum mechanics, an RNG could utilize a physical process to generate random numbers such as radioactive decay or Brownian molecular motion, known from thermodynamics. [5] However, we require for our purpose that the computation of an integral or event generation is reproducible which is a clear disadvantage of a *genuine* RNG as is it can not reproduce a once generated sequence of random numbers. Furthermore, a genuine RNG may require additional hardware with its own quirks and requirements. [6] Therefore, we use a pseudo-random number generator (pRNG) which computes based on a deterministic algorithm a sequence of uniform random numbers from a starting point (seed) with a large periodicity. Thus, a pRNG generates *pseudo*-random numbers, not genuine random numbers. Still, we expect the pseudo-random numbers to appear random enough, that is, the statistical expectations of the pseudo-random numbers conform to those of a random system, e.g. they have no common structure or global correlations. For more detailed introduction into the topic of random numbers we refer to [93, 94].

[3] The high-energy particle physics community is clearly biased towards `C++` applications.

[4] The details on the source code and the documentation can be found in `src/vegas/vegas.nw` in the WHIZARD package release.

[5] Such a setup exists in reality, see Cloudflare: A wall of lava lamps produces a random pattern of ascending and descending wax blobs which are turned into floating point data by a video recorder.

[6] The measurement of a physical process is still an experiment, and as such needs attention and knowledge from user side and a very careful design.

These requirements can be tested with different test suites based on statistical principles and other methods. A simple example of such a randomness test is the χ^2-test or uniform binning test. Assume we divide the interval $[0, 1]$ into M bins of equal length and generate N random numbers inside that interval. We can then determine the number of random numbers n_i in each bin i. The mean over many trials for n_i is Poisson-distributed, hence, $\langle n_i \rangle = N/M$ and the expected variance is N/M. With the constraint $\Sigma_i^M n_i = N$ for the M values n_i, we see that we have a total of $(M-1)$ degrees of freedom and we obtain for $\chi^2 = M-1$. Even simple random number generators should satisfy this rather simple statistical test. In addition, every random number generator should endure a battery of tests: The TestU01 [95] library implements in `ANSI C` several tests for empirical randomness. A very extensive collection of tests can be found in the Die Hard suite [96], also known as Die Hard 1, which contains, e.g. the squeeze test, the overlapping sums test, the parking lot test, the craps test, and the runs test [97]. There is also a more modern version of this test suite, the Die Harder or Die Hard 2 [98], which contains e.g. the Knuthran [94] and the RanLux [99, 100]. Naturally, such a pRNG should add as little as possible overhead in the form of computation time.

The simplest set of pRNGs is given by the class of linear congruence generator (LCG)

$$X_{n+1} = (aX_n + c) \bmod m, \tag{3.1}$$

with sequence X_n, multiplier a, modulus m and increment c, which fail for any set of parameters every of the aforementioned statistical tests. Every LCG suffers from the so called Marsaglia effect [101] where points composed of n random numbers, $(r_i, ..., r_{i+n})$, are distributed along $(n-1)$-dimensional hyperplanes.

For actually correct example of a better random generator, we introduce the TAO random number generator proposed by [94] and provided by the Vamp package. The generator is based on a lagged Fibonacci sequence of states X_n,

$$X_n = (X_{n-k} + X_{n-l}) \bmod 2^{30}, \tag{3.2}$$

with lags $k = 100$ and $l = 37$ computing portable 30-bit integer numbers. The computation requires a reservoir of at least k random numbers in advance. For higher efficiency a reservoir larger than 1000 states is needed. The TAO random number generator passes the Die Hard tests. The implementation suffers from its integer arithmetic which is less efficient on modern CPUs than floating point arithmetic as it is faster and can be put in pipelines allowing terser computation.

For the parallelization of Monte Carlo integration and event generation, we have to consider the detailed handling of random numbers. First, in order to utilize an RNG like the TAO generator we have to either communicate each random number before or during sampling. Both are time-consuming, spoiling parallelization efficiency. Or we have to prepare them or, at least, guarantee independent sequences of random numbers of different instances of RNG by initializing each sequence with different seeds. For the latter, we have to assume that each independent seed leads to a separate sequence of random numbers. [7] However, there is the unlikely possibility that the sequences will

[7] The assumption can be seen to be valid only in the case that the length of each stream is much smaller than the periodicity of the pRNG.

overlap, for example if one seed is part of the other sequence.[8] In such a case, random numbers would be drawn and evaluated twice, resulting in a false result. Therefore, it is hardly feasible or even impossible to ensure for all combinations of seeds and number of workers such independence.

The inconvenience of slower integer arithmetic on modern CPUs, and the dubious independence renders the TAO random number unpractical for the parallelization.

RNGstream [91] allows to generate multiple independent streams[9] of random numbers and makes it possible to access these streams with $\mathcal{O}(1)$ operations. The underlying backbone generator is the combined multiple-recursive generator, referred to as `MRG32k3a`, which manipulates two components each of order 3 with respect to the simultaneously handled random numbers numbers. The state of the generator after n steps is given by the recurrence relation,

$$x_{1,n} = (1403580 \times x_{1,n-2} - 810728 \times x_{1,n-3}) \bmod 4294967087, \tag{3.3}$$

$$x_{2,n} = (527612 \times x_{2,n-1} - 1360589 \times x_{2,n-3}) \bmod 4294944443, \tag{3.4}$$

with the initial seed $\vec{x}_{i,0} = (x_{i,-2}, x_{i,-1}, x_{i,0})^{\mathrm{T}} \in \{1, 2\}$. The output u_n of the RNG is defined by

$$z_n = (x_{1,n} - x_{2,n}) \mod 4294967087, \tag{3.5}$$

$$u_n = \begin{cases} z_n/4294967088 & \text{if } z_n > 0, \\ 4294967087/4294967088 & \text{if } z_n = 0. \end{cases} \tag{3.6}$$

The resulting period of the RNGstream is approximately 2^{191}. The RNGstream passes the TestU01 and Die Hard suites. Furthermore, the RNG passes the spectral test up to (at least) 45 dimensions, which requires that the point set $T_n \equiv \{(u_0, ..., u_{n-1}) | (x_{1,0}, x_{1,1}, x_{1,2})^{\mathrm{T}} \in \mathbb{Z}^3_{m_1}, (x_{2,0}, x_{2,1}, x_{2,2})^{\mathrm{T}} \in \mathbb{Z}^3_{m_2}\}$ covers the n-dimensional unit hypercube very uniformly, making it an ideal candidate for high multiplicity phase-space integrals of particle physics.

The sequence of random numbers is divided into disjoint streams of length 2^{127}, which are then further divided into substreams of length 2^{76}. Each stream or substream can be accessed by repeated application of the transition function $x_n = T(x_{n-1})$. We rewrite the transition function as a matrix multiplication on a vector, making the linear dependence clear, $x_n = T \times x_{n-1}$. In that form, the transition function can be precomputed and stored for n steps, $T_n = T^n$, using the power of modular arithmetic. Accessing a stream or substream is of the same cost as computing a single advance of the RNG. The exact choice of the advance parameters for the streams and substreams is an intricate problem and we refer to [91].

We have rewritten the RNGstream `C++` implementation, which uses floating point arithmetic, in Fortran 2008 in Whizard. The new RNG ist available for the Vamp2 integrator, but unfortunetaly not for the original Vamp integrator as the TAO RNG is hard-coded for it.

[8] The assumption of independent seeds does not hold here, but there is no way to test this a-priori.

[9] We change our notation here to streams instead of sequences of random numbers accordingly to the RNGstream terminology.

In context of parallel evaluation, the RNGstream allows us to identify each channel with a stream and each cell in r-space with a substreams in advance. As the numerical properties of the random number are then fixed for both serial and parallel evaluation, it will also fix (with further constraints, see later) the numerical properties of the integration. This gives us a powerful tool to test and verify the integrator at hand as serial and parallel results must coincide completely. As such, it gives us a strong tool to test and verify the parallelized version of the integrator.

3.3 Parallelization of VEGAS

A consistent description of the parallelization of the VEGAS implementation [102] was given in [103].

In the following we discuss why a trivial parallelization of VEGAS due to the adaptive algorithm fails. We then discuss how we can distribute the integration using the stratified sampling properties of VEGAS, first, to ensure that the integration is done properly (like in a serial run) and, second, that the adaptation does not fail. For the purpose of illustration we consider an integration over a sharp Gaussian curve, which may resemble a narrow particle resonance.

The most trivial way to parallelize VEGAS would be to distribute N samples among p workers such that each worker uses N/p samples to compute the integral. *After* sampling the result can be obtained as weighted average where the weights correspond to the errors of the individual computations. We have p grids which are adapted each with N/p sampling points, and thus in comparison to a single grid with N samples are less well adapted. In particular, interesting structures may be barely or not at all recognized by VEGAS which in return can lead to a grid that is adapted too coarse in some regions. In the worst case, such a misadaptation can lead to a false result and a correspondingly underestimated error. An example of this is given in the third column of table 3.1, where the first five iterations give completely inconsistent results as the algorithm does not get a hold on the sharp Gaussian function.

However, the method can be improved by synchronizing the p grids after each iteration. The adaptation information is collected on a single worker and the grid is adapted there, in order to minimize numerical instability. In such a way, we adapt a single grid with the full information of N samples. But, as we can see in the fourth column in table 3.1, the errors of each iteration are still larger than those of the serial run.

It.	Calls	Trivial parallelized	Parallelized+Sync.	Unparallelized
1	$5000 \cdot 16$	$4.2 \cdot 10^{-36} \pm 2.2 \cdot 10^{-33}$	$0.000\,086\,4 \pm 0.000\,069\,0$	$0.537\,686\,4 \pm 0.537\,683\,0$
2		$1.2 \cdot 10^{-39} \pm 3.5 \cdot 10^{-33}$	$1.032\,088 \pm 0.041\,060$	$0.993\,894 \pm 0.013\,399$
3		$1.9 \cdot 10^{-35} \pm 5.0 \cdot 10^{-33}$	$1.000\,428 \pm 0.000\,283$	$1.000\,008 \pm 0.000\,067$
4		$6.0 \cdot 10^{-34} \pm 5.0 \cdot 10^{-33}$	$0.999\,933 \pm 0.000\,143$	$0.999\,724 \pm 0.000\,080$
5		$5.7 \cdot 10^{-20} \pm 5.7 \cdot 10^{-20}$	$0.998\,949 \pm 0.000\,187$	$1.001\,166 \pm 0.001\,102$
6		$0.998\,361 \pm 0.000\,492$	$1.000\,323 \pm 0.000\,980$	$0.999\,365 \pm 0.000\,603$
7		$0.996\,855 \pm 0.000\,635$	$1.000\,582 \pm 0.000\,676$	$1.000\,288 \pm 0.000\,295$
8		$0.997\,329 \pm 0.001\,107$	$0.997\,636 \pm 0.000\,311$	$1.000\,052 \pm 0.000\,127$
9		$0.993\,738 \pm 0.002\,539$	$0.997\,168 \pm 0.000\,160$	$1.000\,017 \pm 0.000\,088$
10	$20000 \cdot 16$	$0.992\,781 \pm 0.001\,604$	$0.998\,939 \pm 0.001\,081$	$1.000\,019 \pm 0.000\,068$
11		$0.993\,782 \pm 0.001\,661$	$0.999\,631 \pm 0.000\,827$	$1.000\,016 \pm 0.000\,057$
12		$0.993\,844 \pm 0.001\,446$	$0.999\,705 \pm 0.000\,355$	$0.999\,937 \pm 0.000\,046$
13		$0.994\,918 \pm 0.001\,192$	$0.999\,693 \pm 0.000\,325$	$0.999\,940 \pm 0.000\,040$
14		$0.996\,218 \pm 0.001\,213$	$0.999\,611 \pm 0.000\,316$	$0.999\,957 \pm 0.000\,035$
15		$0.997\,030 \pm 0.001\,070$	$0.999\,720 \pm 0.000\,255$	$0.999\,995 \pm 0.000\,021$

Table 3.1: Taken from [103], we compare 15 iterations of a trivial parallelized and trivial parallized-synchronised VEGAS with 16 workers integrating a sharp Gaussian with an unparallelized run. Equal numbers of function calls were sampled in each run.

The numerical properties of the adaptation have been fixed so far with the synchronization of the grids using a single worker. However, all workers sample the full unit hypercube. On the other hand, the unparallelized run distributes all samples evenly by means of the super-imposed stratification. In order to fix the numerical properties of the sampling itself, we make use of the super-imposed stratification of the unit hypercube into many sub-hypercubes which we can then distribute among the workers.

In principle, Monte Carlo sampling does not require intermediate communication during an iteration, except for synchronization calls at the beginning or end of an iteration for synchronizing the grids. But we have to ensure that the sub-hypercube each are only assigned once to a worker. A direct one-to-one assignment that would communicate via a broadcast-gather-approach each per hypercube introduces a significant overhead that spoils the efficiency of the parallelization.

Therefore, it is more convenient for a worker to calculate more than one hypercube before communicating its results. Let r be the number of fractions the unit hypercube is equally split into and k the number of hypercubes $\{G_i\}$. In order to minimize the communication time, we require that $r \ll k$. On the other hand, the problem of dynamic load balancing can be circumvented by requiring for the number of workers p and number of fractions r: $p \ll r$. We arrive at the overall constraint,

$$p \ll r \ll k. \tag{3.7}$$

We can implement a static load balancer respecting the constraint in eq. (3.7) using an algorithmic approach. First, we slice the d-dimensional hypercubes r_i up into two sub-cubes $r_i^{\parallel}$ and $r_i^{\perp}$ with dimensions $d_{\parallel}$ and $d_{\perp}$. The sub-cubes $r_{i,\perp}$ are distributed among the workers where each worker evaluates a d-dimensional hypercube $r_i = r'_{i,\parallel} \otimes r'_{i,\perp}$. Here, the $x = x_1 \otimes x_2$ denotes an outer product between the two sub-cubes x_i such that $d = d_1 + d_2$. We refer to the sub-cubes in $r^{\parallel}$-space as parallel sub-space as they are evaluated concurrently by all workers. The complementary sub-space is referred to as orthogonal sub-space, as $d = d_{\parallel} + d_{\perp}$ holds. In order to impose the load balancing, it is suggested to choose $d_{\parallel} = \lfloor d/2 \rfloor$ and $d_{\perp} = \lceil d/2 \rceil$, which should satisfy eq. (3.7).

A last remark, VEGAS is only parallelizable if there is an equidistant binning in the super-imposed stratification space, otherwise stated, $n_r \geq 2$. However, the number of equidistant binnings per dimension depends on the number of sampling points N,

$$n_r = \sqrt[d]{N/2}, \tag{3.8}$$

or, vice versa,

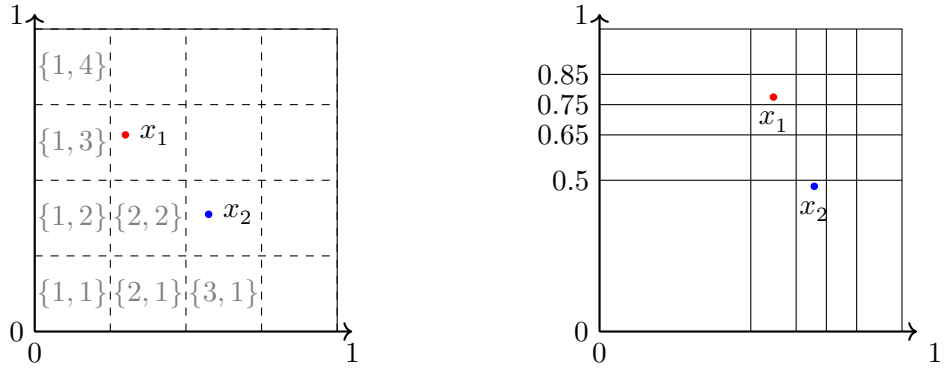
$$N = 2n_r^d. \tag{3.9}$$

Requiring at least $n_r = 2$, eq. (3.8) sets a lower bound for the number of sampling points, $N \geq 2^{d+1}$, which becomes important later for the combination of VEGAS and multi-channel parallelization as the number of sampling points per channel is subject to adaptation.

Implementation

In the VEGAS implementation, each hypercube in r-space is tagged with a unique coordinate,

$$\vec{c}_i = (c_{i,1}, ..., c_{i,d})^{\mathrm{T}}, \quad 1 \leq c_{i,j} \leq n_r, c_{i,j} \in \mathbb{N}, \tag{3.10}$$



(a) Super-imposed stratified grid in r-space. **(b)** Integration grid in x-space.

Figure 3.1: We compare the super-imposed stratified grid in r-space with the actual integration grid in x-space. For illustration, we have chosen $n_r = 4$ for the uniformly binned stratified grid and $n = 5$ for the adaptive integration grid, and we entered the first coordinates for the cells in the super-imposed stratified grid. The integration grid is adapted and has a pronounced structure in the area $[0.65, 0.85]^{\otimes 2}$. We see how different points in r-space are mapped by eq. (A.9) into x-space.

as shown in the example in fig. 3.1. The algorithm cycles over the hypercubes in r-space by incrementing the coordinate $\vec{c}_i$ of the latest evaluated hypercube r_i, as shown in fig. 3.2, beginning at $\vec{c}_0 = (1, 1, ..., 1)^{\mathrm{T}}$. We split the coordinate $\vec{c}_i$ up into parallel and orthogonal coordinates, $\vec{c}_i^{\,\parallel}, \vec{c}_i^{\,\perp}$, with $\vec{c}_i = \vec{c}_i^{\,\parallel} \otimes \vec{c}_i^{\,\perp}$,

$$\vec{c}_i^{\,\parallel} = (c_{i,d_\perp+1}, ..., c_{i,d})^{\mathrm{T}}, \tag{3.11}$$

$$\vec{c}_i^{\,\perp} = (c_{i,1}, ..., c_{i,d_\perp})^{\mathrm{T}}. \tag{3.12}$$

For the purpose of the parallelization, we can then split up the loop over the hypercubes (cells) of the r-space into an inner loop over $r^{\parallel}$-space, which is evaluated by all workers, and an outer loop over $r^{\perp}$-space. Each worker only descends into the inner loop, if the corresponding hypercube with coordinate $\vec{c}_i$ is assigned, if not, the inner loop is skipped and we advance the state of the random number generator by the number of samples in the hypercube $\vec{c}_i$. After sampling, the results can be collected from the individual workers. Since we distribute the computation on the basis of the decomposition of r-space, the full integral can be retrieved by summing all individual results in the r_i hypercubes and the variance as the sum of squared values in the r_i hypercubes, from which we can then determined with eq. (2.2) the actual variance. The same applies to the distribution values of r_i in eq. (2.7), which can be summed up binwise in each dimension. The detailed explanation of the implementation will be discussed in section 3.5.

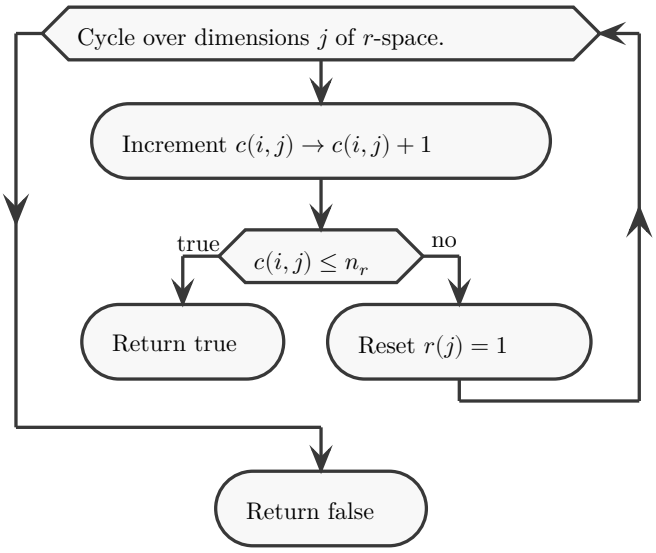


Figure 3.2: The increment algorithm for the hypercube coordinates works as follows: Beginning with dimension $j = 1$, the j-th coordinate component is incremented by one, $c_{i,j} \to c_{i,j} + 1$. If for the coordinate component $c_{i,j} \leq n_r$ holds, then the algorithm will terminate and return the `true` value. However, if the coordinate component $c_{i,j}$ overflows, $c_{i,j} > n_r$, then the coordinate component $c_{i,j}$ will be reset to one and j will be incremented by one, $j \to j+1$. The increment algorithm then starts over again until it terminates or all coordinate components overflow. In that case, the algorithm terminates with the `false` value as the hypercube has been successfully cycled over.

3.4 Combined Parallelization of VEGAS and VAMP2

In contrast to VEGAS, we can parallelize the multi-channel integration directly over the individual channels as they are independent of eachother for the duration of the sampling, only the adaptation requires simultaneously all channel results, as can be seen from eq. (2.26). However, the approach is limited by the number of applicable workers N, since effective parallelization[10] can only occur for $N \leq N_c$. On the other hand, for the case, $N \geq N_c$, we propose to use the spare workers to parallelize individual channels in VEGAS. Combining both approaches, however, requires a detailed examination of the parallelizability of the individual channels in VEGAS, and the computational requirements of individual channels. In the end, we have to find a balance between the two parallelizations, so that the resources can be used optimally.

In general, for standard multi-particle problems the number of phase-space channels is rather large, typically exceeding $10^3 ... 10^4$. We can then assign one or more channels to each worker. In some calculations, the matrix-element is computing-intensive but the number of phase-space channels is small, e.g. NLO virtual matrix elements. In that case, we can parallelize over single grids. We assign to each worker a separate slice of the n_r^d cells of the stratification space. In principle, for the simplest case of $n_r = 2$, we can

[10] If we have more workers available than we distribute tasks, we lose efficiency for the remaining works in terms of Ahmdal's law.

exploit up to 2^d computing nodes. On the other hand, parallelization over r-space is only meaningful *when* $n_r \geq 2$. Especially when we take into account that n_r changes between different iterations as the number of calls $N_{i,c}$ depends on the multi-channel weights α_i.

In summary, we need to consider the following possibilities:

- $N_c \geq N$, apply full multi-channel parallelization;
- all channels parallelizable, apply full VEGAS parallelization;
- mixed mode parallelization with a load balancer between VEGAS and multi-channel parallelization.

Before we turn our attention to the load balancing process between VEGAS and multi-channel parallelization, we discuss the implementation of the previously described parallelization approaches.

3.5 Implementation Details of the MPI Parallelization

Firstly, we use the encapsulation of the integrator as a library. Starting from a top-down approach that the integrator's code is embedded as a (static or shared) library in a main program, we distinguish between the interface data and run-time data. The interface data are passed down from the interface by the calling program and are the same for all parallel instances, requiring no communication. The run-time data arise in the integrator's library and may differ during execution, therefore, requiring communication. Secondly, we expect that the workers are running identical code up to different communication-based code branches. This is ensured by the overall worker setup which is externally coordinated by the MPI-library provided process manager `mpirun`,[11] and the applied master-slave model. The master-slave model allows us to conserve the numerical properties of the integrator, in the form that the master collects the individual results and adaptation information of the slaves (and itself), computes the overall result and performs the adaptation of weights and grids as the adaptation of the grids is numerically very sensitive to the smallest changes. Together with the previously introduced RNGstream we can transfer the complete numerical properties of the serial to the parallel execution, where the latter reproduces the former.

At the beginning of an iteration pass of VAMP2, we broadcast the current grid setup and the channel weights from the master to all slave workers. For this purpose, the MPI protocol defines collective procedures, e.g. `MPI_BCAST` for broadcasting data from one process to all other processes inside the communicator, as shown in fig. 3.3. The multi-channel formulas eq. (2.22) and eq. (2.24) force us to communicate each grid[12] to every worker. The details of an efficient communication algorithm and its implementation is part of the actual MPI implementation (most notably the OpenMPI and MPICH libraries) and no concern for us. After we have communicated the grid setup using the collective procedure `MPI_BCAST`, we let each worker sample over a predefined set of

[11] The executable is started by `mpirun` in N identical instances, hence, N worker.

[12] The grid type holds information on the binning x_i, the number of dimensions, the integration boundaries and the Jacobian.

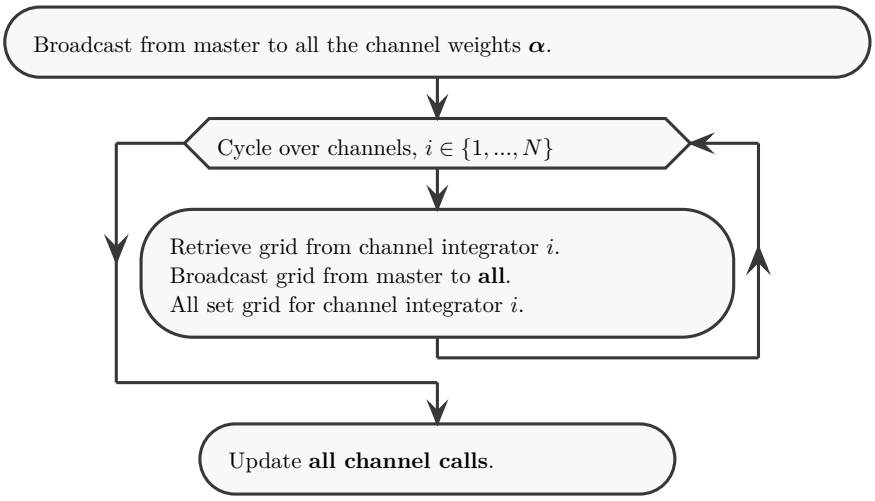


Figure 3.3: Setup of the grids for the multi-channel parallelization. Before we broadcast the individual grids from the master to all workers, we broadcast the channel weights in a non-blocking fashion. After all grids have been broadcasted, we update the configuration of the channels due to the changes of weights and grids.

phase-space channels. Each worker skips its non-assigned channels and advances the stream of random numbers to the next substream such as it would have used them for sampling.

If a channel is VEGAS, we will defer the parallelization to VEGAS either using a naive or more sophisticated static load-balancing approach. For the naive load balancer, we spawn an `MPI_BARRIER` waiting for all other workers to finish their computation until they synchronize at the barrier and start with parallelization of the channel over its grid. Alternatively, we use the static load balancer, which makes a more specific choice regarding parallelizable grids, based on the channel weights and the number of workers.

In order to employ the VEGAS parallelization from section 3.3 we divide the r-space into a parallel subspace $r_\parallel$ with dimension $d_\parallel = \lfloor d/2 \rfloor$ over which we distribute the workers. We define the left-over space $r_\perp = r \backslash r_\parallel$ as perpendicular space with dimension $d_\perp = \lceil d/2 \rceil$. Assigning to each worker a subspace $r_{\parallel,i} \subset r_\parallel$, the worker samples $r_{\parallel,i} \otimes r_\perp$. For the implementation we split the loop over the cells in r-space into an outer parallel loop and an inner perpendicular loop. In the outer parallel loop the implementation descends in the inner loop only when worker and corresponding subspace match, if not, we advance the state of the random number generator by the number of sample points in $r_{\parallel,i} \otimes r_\perp$, where i is the skipped outer loop index. The overall procedure is schematically shown in fig. 3.4.

After sampling over the complete r-space the results of the subsets are collected. All results are collected by reducing[13] them by an operator, e.g. `MPI_SUM` or `MPI_MAX` with `MPI_REDUCE`. The application of such a procedure from a MPI library is in general more efficient than a self-written procedure. We implemented all communication calls as non-

[13]Reduction here is meant as a concept from functional programming where data reduction is done by reducing a set of numbers into a smaller set of numbers via a function or an operator.

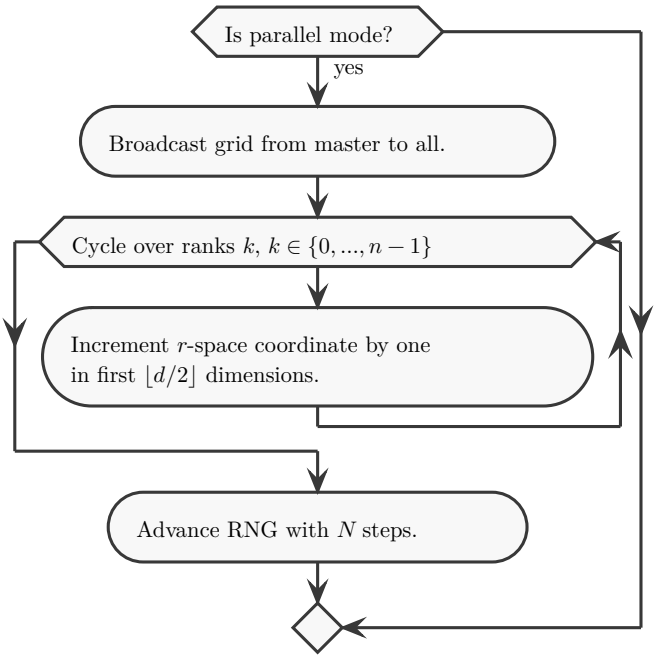


Figure 3.4: Preparation of the VEGAS parallelization. When the channel is VEGAS-parallelizable, we then synchronize the grid firstly in the VEGAS integrator. After that, we set the correct start coordinate for the respective worker and advance the state of the RNGstream accordingly.

blocking, i.e. the called procedure will directly return after setting up the communication. The communication itself is done in the background, e.g. by an additional communication thread. The details are provided in the applied MPI library. To ensure the completion of communication a call to `MPI_WAIT` has to be done.

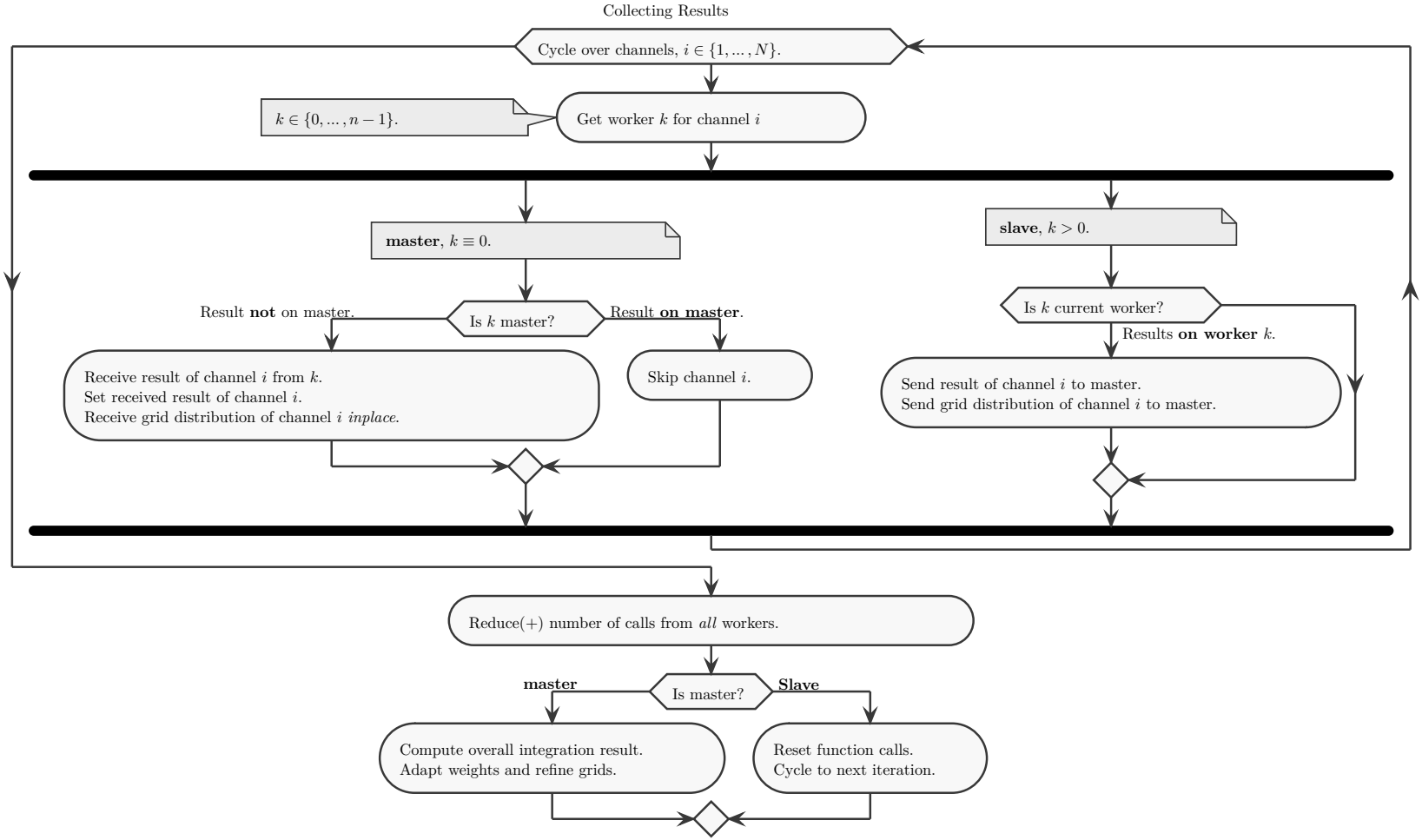


Figure 3.5: Execution branches for master and slave. We iterate over the channels and determine for each channel the assigned worker which has the integration results and adaptation information (grid distribution). On the master, the results and the adaptation information are received from the respective slave. On the slave worker, the assigned channel is sent to the master. At the end, the number of function calls is collected on the master.

When all channels have been sampled, we collect the results from every channel and combine them to the overall estimate and variance. We apply a master/slave chain of communication where each slave sends its results to the master as shown in fig. 3.5. For this purpose the master worker and the slave worker execute different parts of the code. The computation of the final results of the current pass is then exclusively done by the master worker. Additionally, the master writes the results to a VAMP2 grid-file in case the computation is interrupted and needs to be restarted after the latest iteration (adding extra serial time to WHIZARD runs).

When possible, we let objects directly communicate by Fortran 2008 type-bound procedures, e.g. the main VEGAS grid object, `vegas_grid_t` has `vegas_grid_broadcast`. The latter broadcasts all relevant grid information which is not provided by the API of the integrator. We have to send the number of bins to all processes before the actual grid binning happens, as the size of the grid array is larger than the actual number of bins requested by VEGAS.[14]

Further important explicit implementations are the two combinations of type-bound procedures `vegas_send_distribution` for sending and for receiving `vegas_receive_distribution`, and furthermore `vegas_result_send`/`vegas_result_receive` which are needed for the communication steps involved in `VAMP2` in order to keep the VEGAS integrator objects encapsulated (i.e. preserve their `private` attribute).

Beyond the inclusion of non-blocking collective communication we choose as a minimum prerequisite the major version 3 of MPI for better interoperability with Fortran and its conformity to the Fortran 2008 `+ TS19113` (and later) standard [82, Sec. 17.1.6]. This, e.g., allows for MPI-derived type comparison as well as asynchronous support (for I/O or non-blocking communication).

Blocking

A final note on the motivation for the usage of non-blocking procedures. Classic (i.e. serial) Monte Carlo integration exhibits no need for in-sampling communication in contrast to classic application of parallelization, e.g. solving partial differential equations. For the time being, we still use non-blocking procedures in VEGAS for future optimization, but in a more or less blocking fashion, as most non-blocking procedures are followed by an `MPI_WAIT` procedure. However, the multi-channel ansatz adds sufficient complexity, as each channel itself is an independent Monte Carlo integration. A typical use case is the collecting of already sampled channels while still sampling the remaining channels as it involves the largest data transfers in the parallelization setup. Here, we could benefit most from non-blocking communication. To implement these procedures as non-blocking necessitates a further refactoring of the present multi-channel integration in WHIZARD, because in that case the master worker must not perform any kind of calculation but should only coordinate communication. A further constraint to demonstrate the impact of turning many of our still blocking communication into a non-blocking one is the fact that at the present moment, there do not exist any free and open-source profilers compliant

[14] The size of the grid array is set to a pre-defined or user-defined value. If the implementation switches to stratified sampling, the number of bins is adjusted to the number of cells and, hence, does not necessarily match the size of the grid array

with the MPI 3.1 standard that support Fortran 2008. Therefore, we have to postpone the opportunity to show the possibility of completely non-blocking communication in our setup.

3.6 MPI Process Balancing

In the following, we outline the requirements of load balancing for the combined parallelization by introducing an appropriate notation and discuss the condition of balance between both parallelization modes.

We define the set $C \equiv \{1, ..., N_c\}$ as indices of the multi-channel sum. The index set V for the VEGAS parallelizable channels is given as $V \equiv \{i \in C | n_r \geq 2, \text{ for the i-th channel}\}$, where n_r is the number of (equal) binnings in r-space (see section 2.2). The index set $\bar{C} \equiv C/V$ gives the remaining channels, which can be trivially parallelized. The number of elements in a set is given by the cardinality, denoted by $|\cdot|$.

First of all, we compare the computation time $\mathcal{C}(i, 1)$ of a single-worker channel with the computation time $\mathcal{C}(i, N)$ for an N-parallel channel, parallelized within VEGAS. We find the following inequality

$$\mathcal{C}(i, 1) \leq \mathcal{C}(i, N) \cdot N + \mathcal{C}_{\parallel}(i, N), \tag{3.13}$$

where we introduced the additional overhead of the parallelization as $\mathcal{C}_{\parallel}(i, N)$. The latter incorporates the additional execution time of code introduced for the coordination of the parallelization and the additional execution time by the parallelization itself, e.g. time of communication procedures. We remark that the communication time depends crucially on the number of workers, the actual MPI library implementation and other external factors like network bandwidth or cluster workload. For further considerations, we ignore these effects as they are beyond of the scope of this thesis.[15] In a next step, we consider eq. (3.13) in the combined parallelization. The overall composition of the communication time changes. First, both multi-channel and VEGAS parallelization require the same setup, distributing the integration grids and channel weights, and after sampling, all integration results and adaptation information must be collected. For the VEGAS parallelization, the results as well as the adaptation information are firstly collected on a single worker. Generally, this worker is not the master rank, so we explicitly separate the communication time $\tilde{\mathcal{C}}_{\parallel}$ within VEGAS from the overall communication time. The rest of the communication is identical for both sides, then during the sampling itself no communication is further required. The design goal for the load balancing is then to ensure that the total computation time for the multi-channel parallelized channels is equal to the total computation time for the VEGAS-parallelized channels. We re-express this statement as

$$\sum_{i \in \bar{C}} \mathcal{C}(i, 1) = \sum_{i \in V} \mathcal{C}(i, N_{i,\text{worker}}) + \tilde{\mathcal{C}}_{\parallel}(i, N_{i,\text{worker}}), \tag{3.14}$$

where we have neglected the additional execution times for collecting the overall results or organizational code, since these times are applied equally to both sides.

[15] We can not improve or change those negative effects by the implementation as those are an issue to be solved by the user.

Naive Load Balacing

First, we introduce a naive approach using all workers in `MPI_COMM_WORLD`. We let each worker sample over a predefined set of phase-space channels. Each worker skips its non-assigned channels but advances the stream of random numbers to the next substream as it would have used them for sampling. However, if we can defer the parallelization to VEGAS, we spawn a barrier waiting for all other workers to finish their computation, and then start with the parallelization of the channel over its grid. The results for multi-channel parallelized channels $\bar{C}$ are firstly stored on their assigned ranks, and must be communicated to the master worker. On the other hand, in this setup the results for VEGAS parallelized grids are already stored on the master removing the additional communication for collecting results during post sampling $\mathcal{C}_{\text{post}} = \mathcal{C}_{\parallel}(i, N_{\text{Worker}})$.

The pre-sampling setup stays unchanged, and the balance in eq. (3.14) changes,

$$\sum_{i \in \hat{\mathcal{C}}} \mathcal{C}(i, 1) = \sum_{i \in V} \mathcal{C}(i, N_{i,\text{Worker}}). \tag{3.15}$$

In summary, per default, we parallelize over phase-space multi-channels, but prefer single-grid parallelization for the case that the number of cells in r-space is $n_r > 1$.

The advantage of this approach is that we save the extra communication time by collecting the results of the VEGAS parallelized channels, and that both the conceptual approach and the implementation are very simple (and understandable).

Static Load Balancing

In order to estimate the computation time for a grid, we use the channel weights, since these are proportional to the number of samples N_i per channel. Furthermore, we assume that all sampling points cost on average the same computation time. As an estimate, the average behavior should be sufficient, but we keep in mind that the execution time for different points can fluctuate very much, as the sampling depends, besides the matrix elements computation, on phase-space cuts, phase-space mappings, and search for grid probabilities. Each can reject further evaluation of a sampling point. For example, the sampling can be prematurely terminated if the cuts were not met.

Based on eq. (3.13), the multi-channel parallelization seems to perform more efficiently with regard to the VEGAS parallelization and, therefore, it is the favorable mode of parallelization for $N \leq N_c$. For increasing number of workers, we need then to model the resource requirements between the multi-channel parallelization and the additional VEGAS-parallelization smoothly, e.g. without oversubscribing resources to either of both as this would spoil the efficiency.[16] In order to persuade such a smooth modeling between modes, we introduce the following steps: Firstly, we define the criterions on the selection of VEGAS parallel grids depending on the number of channels and workers. Secondly, we introduce the static load balancing and describe the mechanism to create communicators in separate contexts with `MPI_CREATE_COMM_GROUP`.

The necessary criterion for grid parallelization is that it can only be decomposed in r-space and *only when* $n_r \geq 2$. Furthermore, we require $\alpha_i \geq \alpha_{\min}(N_c, N_{\text{Worker}})$ in order

[16] Such a behavior can be countered by a dynamic load balancer, which is much more involved to implement and is under current research.

to include the computational expense of the channels, as $\alpha_i \propto N_i$. The minimum weight $\alpha_{\min}$ is a function of the number of channels N_c and number of workers N_{Worker} which we define as

$$\alpha_{\min}(N_c, N_{\text{Worker}}) = N_c^{1-\frac{1}{\gamma}} \cdot N_{\text{Worker}}^{-\gamma}, \quad N_c \leq N_{\text{Worker}}, 1 \leq \gamma < 2. \tag{3.16}$$

The minimum weight models a continuous transition between the modes for fixed N_c and increasing N_{Worker}, letting us count channels with a weight larger than an average weight $1/N_c$ as parallelizable.

After communicating the integration grids and channel weights, we initialize the static load balancer for each integration pass of VAMP2. In addition to the number of channels and workers, the static load balancer requires the number of cells in r -space. The balancer decides according to the aforementioned rules which mode is to be applied and which channel is evaluated by which worker or subset of workers, as shown in fig. 3.6. Since all information is known *a priori* and identical on all workers, the balancer does not require any communication for the initialization. After the number of workers has been set for both parallelization modes, workers are assigned accordingly. For the multi-channel parallelization, we assign workers to one or more channels, in acyclical manner. For the VEGAS parallelization, we assign as many workers as possible (limited only by the number of cells $n_r^{d,\|}$) to the individual grids. In order to do so, we iterate over the parallel grids and determine the largest possible number of available workers. These are then marked as occupied for the time being. The remaining workers are then distributed among the next grids, until there are either too few available workers or no more available at all. In that case, we overflow the assignment by taking the leftover available worker and start again with the first worker. The remaining worker are then marked all again as available and we repeat the procedure until all grids have workers assigned.

We separate the communication between the two modes by creating a communicator for the channel parallelization and creating a communicator for each parallelizable grid. Thus, the communication for each evaluation is separated. With that we can use the existing infrastructure with only slight adjustments. Instead of `MPI_COMM_WORLD` the respective communicator is then used.

The MPI standard defines two procedures using process groups to create communicators: `MPI_COMM_CREATE` and `MPI_COMM_CREATE_GROUP`, the latter is an addition of MPI-3. `MPI_CREATE_COMM` creates a new communicator from a parent communicator, typically `MPI_COMM_WORLD`, and a subset of the parent communicator's process group, which can be generated using the `MPI_GROUP_*` procedures. All workers in the parent communicator must then call `MPI_CREATE_COMM`. This constraint is removed in `MPI_COMM_CREATE_GROUP`, where only the workers in the specified subgroup have to execute `MPI_COMM_CREATE_GROUP` in order to create the communicator. Thus, for our implementation, we can dynamically generate the grid communicators from the assigned subgroup of workers from the parent-communicator in the integration loop of VAMP2, without bothering the channel workers or other non-assigned grid workers with a call to `MPI_CREATE_COMM`, which would introduce a non-trivial blocking of the integration loop.

For the multi-channel parallelization, the results for each channel are on their assigned worker. For the parallelized grids, the results are collected on the first worker, to which we refer as slave master, in the grid's communicator. In the end, the results can be collected

in the same way as for the naive balancer in the `MPI_COMM_WORLD` communicator. For this, the balancer keeps track of the relationship between the ranks in our self-created communicators with the ranks in `MPI_COMM_WORLD`.

A first version of the implementation is part of the current development branch of WHIZARD. Unfortunately, we can not show any results as the implementation has been more difficult to handle than expected. Furthermore, the communication in the new implementation is not stable. We could not reliably reproduce the same integration results for different numbers of workers. So, we defer a detailed discussion of benchmark results to a future work. Only when the numerical properties are restored, the benchmarks are meaningful.

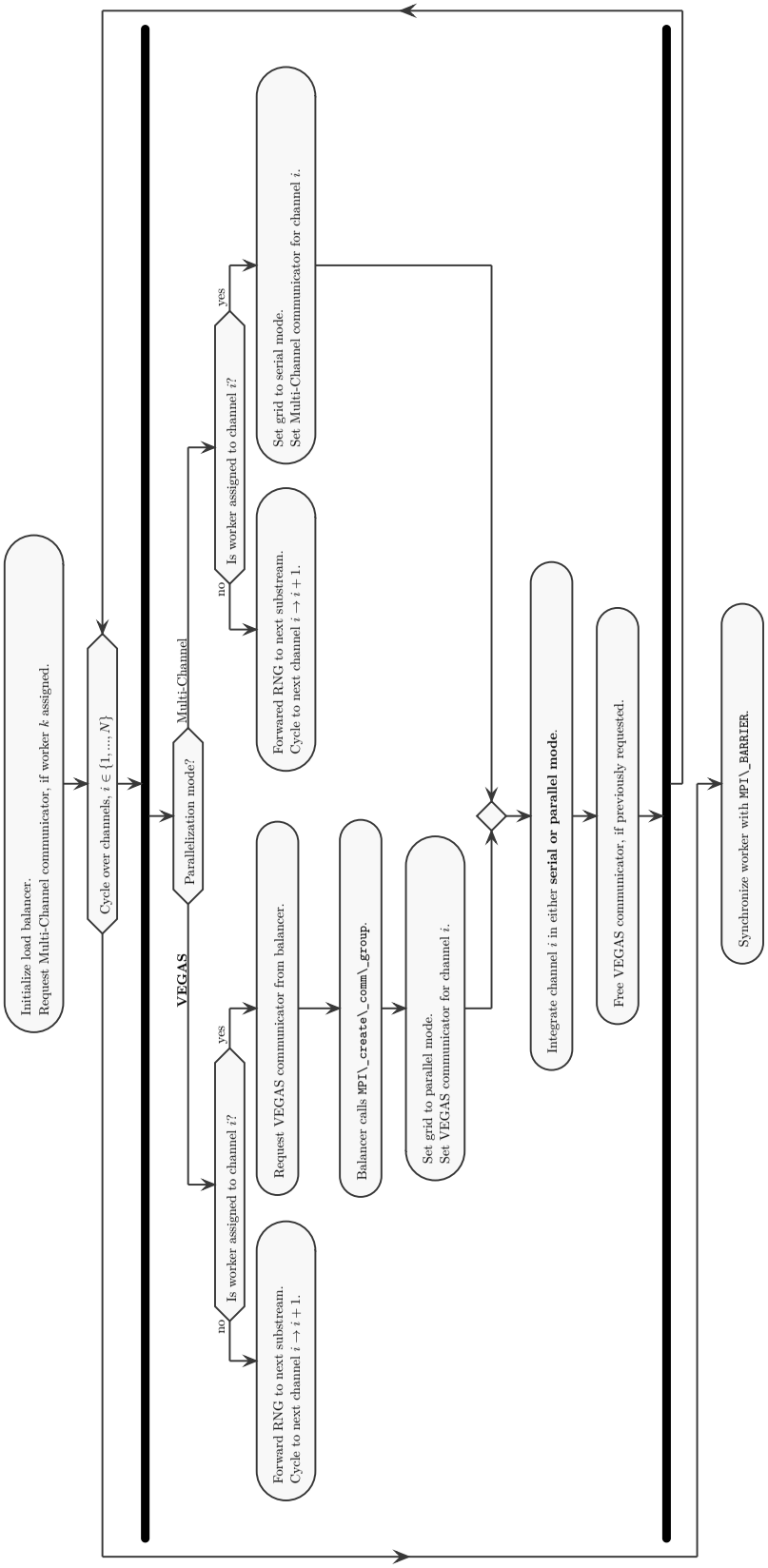


Figure 3.6: Parallelized multi-channel integration loop. After preparing the parallel integration run, we either descend into the VEGAS integrator for parallelization or distribute the channels among the available workers, depending on the parallelization mode of the respective channel.

Profiling

For efficient parallelization, it is important to identify the critical points that prevent a more effective use of computation resources. In general, for serial applications there is the possibility of profiling, which can be applied on parallel programs only with specialized tools, too. We discuss some minor technical details of the fine tuning of the new integrator and in some points of WHIZARD.

In order to profile a program, for example to determine the number of calls or the (total) run-time of procedures, the compiler offers the possibility to generate additional annotations as well as code in the program, which measure different program parameters and write them into a file. This file can then be evaluated using a profiler such as `gprof`. Unfortunately, this approach only supports statically linked programs. Since WHIZARD and its libraries, as well as the external libraries, e.g. the MPI library, are all dynamically linked and loaded, we need another tool.

For the profiling of a dynamically linked program, the dynamic linker and loader `ld-linux.so` (or `ld.so`)[17] supports since version 2.1 of `glibc` profiling of a single shared object. In order to activate the profiling, one passes the environment variable `LD_PROFILE = <shared object name>` to the program to be executed. The names of the shared objects (shared libraries) required by a program or shared object can be listed with `ldd <program>`. The output of the profiling data can be found under `$LD_PROFILE_OUTPUT/$LD_PROFILE.profile` where the environment variable `$LD_PROFILE_OUTPUT` steers the output directory. The program `sprof` can then evaluate the previously generated profiling data providing a flat-profile, a call graph or call pairs, steerable by commandline options. A disadvantage of the `ld.so` profiling method is that only one profile of a single shared object can be created at once, so that several runs of a program are needed to profile all relevant shared objects. An advantage of the aforementioned tools is that they are all free software and they are open source, which is relevant for the reproducibility of scientific results. It allows other researchers to check and verify our proceeding as open source offers a transparent view of the tool's internals, making results more comprehensible. The free license removes the boundaries of a dubious (and possible open science hostile) license. On the other hand, we have to question the usability of the free tools, as they are only of limited use for us.

Therefore, we inevitably chose to use a commercial, non-free and closed-source solution from Intel,[18] VTune [104]. VTune supports directly the profiling of shared-libraries and the evaluation of the interaction between different shared-libraries, which can not be done in a straight-forward manner with `sprof`. Firstly, we used VTune on the serial version of WHIZARD and found that, for example, the debug statements used in the new integrator have caused a significant I/O, even when not activated. Since these were called each time a phase space point was evaluated, this considerably slowed down the run-time of the new implementation, doubling or tripling the run time of the integration. We could easily solve this problem by rewriting the debug statements so that they do not unnecessarily waste resources unless explicitly enabled. Secondly, we could speed up the search of the

[17] "The programs ld.so and ld-linux.so* find and load the shared objects (shared libraries) needed by a program, prepare the program to run, and then run it." from `man ld.so`.

[18] Although Intel offers a free license for open source projects, but explicitly excluding paid developers even if they are funded by the public sector.

grid probabilities in eq. (2.26), by using a linear search instead of a binary search on the grids. This may seem unintuitiv at first, but has the following reason, the processor is trying to predict which code will be executed in the future to introduce a low-level optimization. This can lead to mispredictions, e.g. for `if`-clauses a wrong branching may be predicted. Then, the processor has to revert its state back, and thus wasting time. This concerns in particular the binary search, although this is mathematically of order $\mathcal{O}(n \log n)$. But the processor optimization can slow down considerably its execution time for a small number of bins per axis, e.g. $n \leq 20$. Profiling has shown that a linear search for $n \leq 20$ gives an advantage over the binary search of 1 % to 10 %, depending on the number of involved channels.

We were able to use VTune for minor parallel profiling. We could show that for a small number of workers, the naive load balancer behaves comparatively well. Unfortunately, we were not able to test a higher number of workers because the corresponding tool was not / is not available on the HPC of the University of Siegen. In a subsequent work, the static-load balancer could be tuned accordingly, or an efficient pull-push system could be established by profiling.

3.7 Embedding the Parallel Integration into WHIZARD

We have done the parallelization of the adaptive integrator in an encapsulated fashion. Furthermore, we did not propose any major changes to the interface of the integrator due to the parallelization. We may embed the parallelized integrator into WHIZARD without major precautions. But it turned out that a straightforward embedding of the parallel integrator is not possible. Therefore, we first need to familiarize ourselves with the structure and program flow of WHIZARD with respect to the integration setup in more detail.

The program flow of WHIZARD is steered on user-side by the domain-specific language `Sindarin`[19] where instructions are executed in their specified order. Here, we restrict ourselves to the execution of the `integrate` command that instructs WHIZARD to integrate a (specific) process according to the previous settings. However, before the actual process has to setup, WHIZARD carries out several tasks: The matrix element source code has to be generated by a call to OMEGA, from which WHIZARD compiles the process library, which is then dynamically linked by the main executable,[20] and the phase-space parameterization has to be carried out either by the antiquated `cascades` module or the optimized-version `cascades2`. WHIZARD can skip individual steps of the process setup if the corresponding task has been carried out, e.g. by testing respective files. All these parts of process setup involve heavy I/O, especially writing to the filesystem. In order to enable file input/output (I/O), in particular to allow the setup of a process, without user intervention, we implement the well-known master-slave model. Additionally, we require WHIZARD to operate parallely on a distributed filesystem, e.g. network filesystems as they are usually supported by HPC clusters. The master worker, specified by rank 0, is allowed

[19] WHIZARD can either read statements from one (or more nested) file(s) or directly interpret input in a command line mode.

[20] For the sake of completeness, OMEGA can also provide bytecode that does not need to be compiled, which is then executed by the OMEGA virtual machine [78].

to setup the matrix-element library and to provide the phase-space configuration (or to write the grid files of the integrator) as those are involved with heavy I/O operations. The other workers function solely as slave workers supporting only integration and event generation. Therefore, the slave workers have to wait during the setup phase of the master worker. We implement this dependence via a blocking call to `MPI_BCAST` for all slaves while the master is going through the setup steps. As soon as the master worker has finished the setup, the master starts to broadcast a simple logical which completes the blocked communication call of the slaves allowing the execution of the program to proceed. The slaves are then allowed to load the matrix-element library and read the phase-space configuration file in parallel.

The slave setup adds a major contribution to the serial time, mainly out of our control as the limitation of the parallel setup of the slave workers are imposed by the underlying filesystem and/or operating system, since all the workers try to read the files simultaneously. We expect that the serial time is increased at least by the configuration time of WHIZARD without building and making the matrix-element library and configuring the phase-space. Therefore, we expect the configuration time at least to increase linearly with the number of workers.

In comparison, we want to discuss the possibility for genuine[21] parallel I/O with regard to configuration, grid and event files. As explained in section 2.6, the MPI standard since version 2 defines a parallel I/O system. Despite the high level of abstraction reached in the specification, it is still a very tedious task to develop and implement appropriate routines and procedures specific to a given file format. Existing file formats, e.g. the HDF5, specifically developed for scientific use, are well-defined, and in most cases support parallel I/O by interfacing a MPI library. With such a file format with parallel I/O support, an advantage could arise if all workers could write or read the grid file of the integrator in parallel, as this involves the major serial time in our approach. However, we would have to loosen our assumption about conserving the numerical properties of the integrator by allowing slaves to refine grid (after each pass). In principle, we could save a small but not negligible amount of communication time, for example by not having to collect all the results on the master and then writing them to the grid file. But, to what extent such an approach may prove to be more effective, can be only checked in detail with a corresponding profiling tool, as discussed in section 3.6.

3.8 Benchmark of VEGAS and VAMP2

In order to assess the efficiency of our parallelization, we compare the two modes, the traditional serial VAMP implementation and our new parallelized implementation, VAMP2. We restrict ourselves to measuring the efficiency of the parallel integration, which, in contrast to parallel event generation, requires non-trivial communication.

For the latter, we limited our efforts to extending capabilities of the event generation to use the RNGstream algorithm in order to secure independent random numbers among the workers and automatically split the output of events into multiple files for each worker,

[21] In our approach, we assume that the underlying filesystem handles the distribution of files among all workers. This is not the most efficient but is easy to implement. Therefore, we deferred the task of parallel I/O to a distributed filesystem.

respectively, as we do not make use of parallel I/O. In this version, the event generation does not require any kind of communication and, therefore, we skip a detailed discussion of the efficiency of parallel event generation, as such a discussion would too much depend upon environmental factors of the used cluster like the capabilities of the underlying filesystem etc.

In the following we outline the hard- and software setup for the two benchmark scenarios: (i) a pure MPI setup, where we determine the run-time and speedup for a fixed problem over a growing number of workers, (ii) a hybrid approach with MPI + OpenMP, where we combine thread-based parallelization with OpenMP with inter-node communication based on MPI. Again, we determine the run-time and speedup, where fix the number of nodes, but with increasing number of threads per node.

The benchmark processes are run on the high-performance cluster of the University of Siegen (Hochleistungrechner Universität Siegen, HoRUS). The cluster consist of 34 Dell PowerEdge C6100 containing each 4 computing nodes with 2 multi-core CPUs. The CPUs are Intel Xenon X5650 with 6 cores each 2.7 GHz and 128 MiB cache. The nodes are connected by gigabit ethernet.

On the software side, the HoRUS cluster is managed by the batching and allocation system Slurm 17.02.02 for easy source distribution. We use Gfortran 7.3.0 as OpenMP-compatible compiler for Whizard and the MPI-library OpenMPI 3.1.3. For the benchmark we let the MPI-library provided run manager `mpirun` coordinate the MPI-specific setup of Whizard within Slurm. We measure the run-time with the GNU time command tool and for the final results we average over three independent runs. The overall computation of a Whizard run is measured, including the complete process setup with the matrix-element generation and phase-space configuration. We expect that the setup gives rise to the major part of the serial computation time of Whizard, and, the I/O operations of the multi-channel integrator, which save the grids after each integration iteration as safe-guard against premature termination.

We benchmark, see fig. 3.7, over N_{CPU} in powers of 2, as it is quasi-standard. Given the architecture of the HorUS cluster with its double hex-cores, benchmarking in powers of 6 would maybe be more appropriate for the MPI-only measurements. We apply a node-specific setup for the measurement of the hybrid parallelization. Each CPU can handle up to six threads without any substantial throttling. For the measurement of the hybrid parallelization we apply a node-specific setup. Each CPU can handle up to six threads without oversubscribing its resources. We operate over $N_{\text{Threads}} = \{1, 2, 3, 6\}$ with either fixed overall number of involved cores/MPI worker, $N_{\text{Worker}} = 60, 30, 20, 10$, with results shown in fig. 3.9, or with fixed number of MPI workers $N_{\text{Worker}} = 20$, with results shown in fig. 3.9.

Benchmark Processes

The benchmark processes are selected such that they include a wide range of Standard Model physics, e.g. QCD, electroweak and Higgs physics, to cover typical applications of Whizard, representative for both hadron and electron-position collider. Such a selection is not unambiguous, as it depends, among other things, on the details of physics to be examined as well as the collider type self, and should only be understand as representive class of processes.

In the following, we study different processes at different levels of complexity in order to investigate the scaling behavior of our parallel integration algorithm. The process $e^+e^- \to \mu^+\mu^-$ at energies below the Z resonance has only one phase-space channel (s-channel photon exchange) and its integration is adapted over one grid. Parallelization is done in VEGAS over stratification space. The process $e^+e^- \to \mu^+\mu^-\mu^+\mu^-\nu_\mu\bar{\nu}_\mu$ with its complicated vector-boson interactions gives rise to $\mathcal{O}(3000)$ phase-space channels. With overall $\mathcal{O}(10^6)$ number of calls for the process, each phase-space channel is sampled (in average) by $\mathcal{O}(10^2)$ calls suppressing the stratification space of all grids. Therefore, parallelization is done over the more coarse phase-space channel loop. The last process we investigate, $e^+e^- \to \mu^+\mu^-\mu^+\mu^-$, has $\mathcal{O}(100)$ phase-space channels where we expect for some grids a distinct stratification space (at least two cells per dimension) allowing WHIZARD to switch between VEGAS and multi-channel parallelization. All but the first trivial examples are taken from [5] mimicking real-world application for a proton-proton collider. The results for the full integration, thus, times for both adaptation and final iterations for the results of the integration are shown in fig. 3.7.

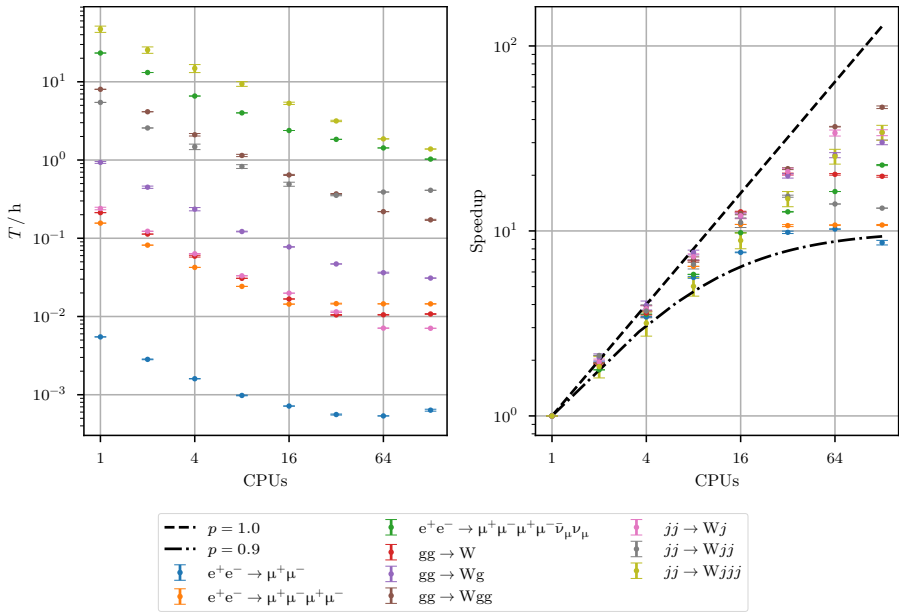


Figure 3.7: We show the overall computation for the adaptive integration processes for process with differing complexity, each for a different number of participating CPUs (left panel). The number of CPUs is chosen as a power of 2. In the right panel, we plot the speedup of the processes and compare them to ideal of Ahmdal's Law with parallel fraction of 1.0 (dashed) and 0.9 (dash-dotted). For the sake of better readibility, we labeled the processes with their partonic content. However, the computation involves the complete process with PDF convolution, but without shower and hadronization.

Furthermore, we are interested in the behavior for increasing complexity of a single process, e.g. increasing (light) flavor content of processes with multiple jets. For the two processes, $jj \to W^-(\to e^-\bar{\nu}_e) + \{j, jj\}$ we increase the number of massless quark flavors in the content of the jets. The results in fig. 3.8 indicate that for a single final-state jet more

flavor content (and hence more complicated matrix elements) lead to lower speedups. For two (and more) final state jets the speedups increases with the multiplicity of light quarks in the jet definition. This means that possibly for smaller matrix elements there is a communication overhead when increasing the complexity of the matrix element, while for the higher multiplicity process and many more phase-space channels, improvement in speedup can be achieved.

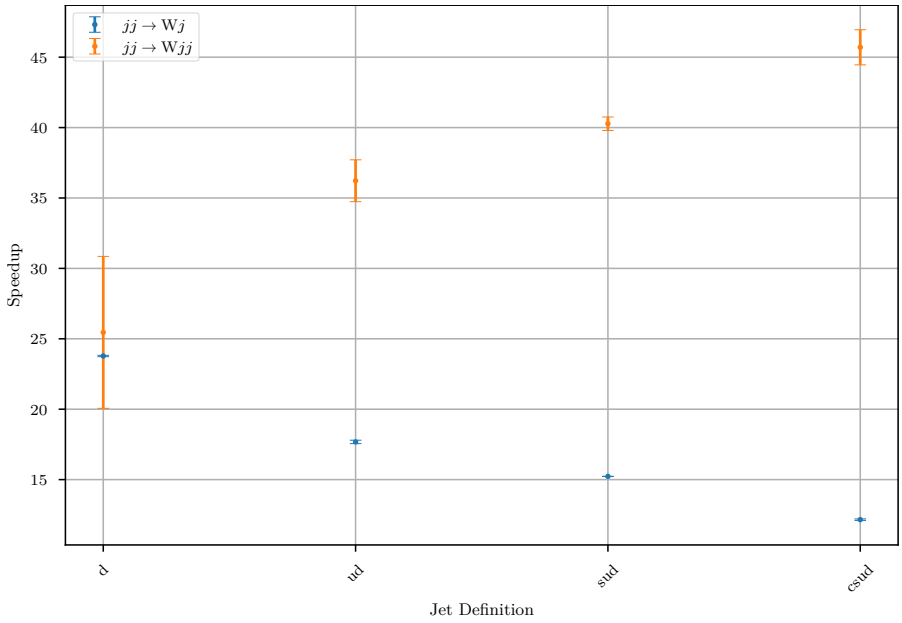


Figure 3.8: We show the speedup for a jet process with increasing flavor content and a fixed numbers of CPUs of 60.

We repeat the measurement for the two processes for the hybrid parallelization, once with a variable number of MPI workers and a fixed number of MPI workers, but each time with an increasing number of threads. The results are shown in fig. 3.10, for variable number of workers, and, for fixed number of workers in fig. 3.9.

Coming back to fig. 3.7 showing the results of the benchmark measurement for MPI: The process $e^+e^- \to \mu^+\mu^-$ saturates for $N > 32$. The serial run-time of WHIZARD is dominating for that process with its two-dimensional integration measure (without beam structure functions) where the Monte Carlo integration is anyways inferior to classical integration techniques. The process $e^+e^- \to \mu^+\mu^-\mu^+\mu^-$ showing mixed multi-channel and VEGAS parallelization, however, also saturates for $N > 32$. Going beyond that, the multi-channel parallelizable process $e^+e^- \to \mu^+\mu^-\mu^+\mu^-\nu_\mu\bar{\nu}_\mu$ achieves a higher speedup but with decreasing slope. The overall speedup plot indicates a saturation beginning roughly at $N > 32$ where serial time and communication start to dominate. We conclude that WHIZARD embarks a parallelization fraction higher than at least 90 % for MPI.

We point out that these results are obtained for a fixed-size problem and the speedup was determined due to Ahmdal's law in section 2.5. However, we expect for real applications where the number of workers is chosen according to complexity and evaluation time of problem, a higher and more favorable speedups of $\mathcal{O}(100)$, in accordance to Gustafson's law. This can be seen from the results in fig. 3.8, where the computation time of the

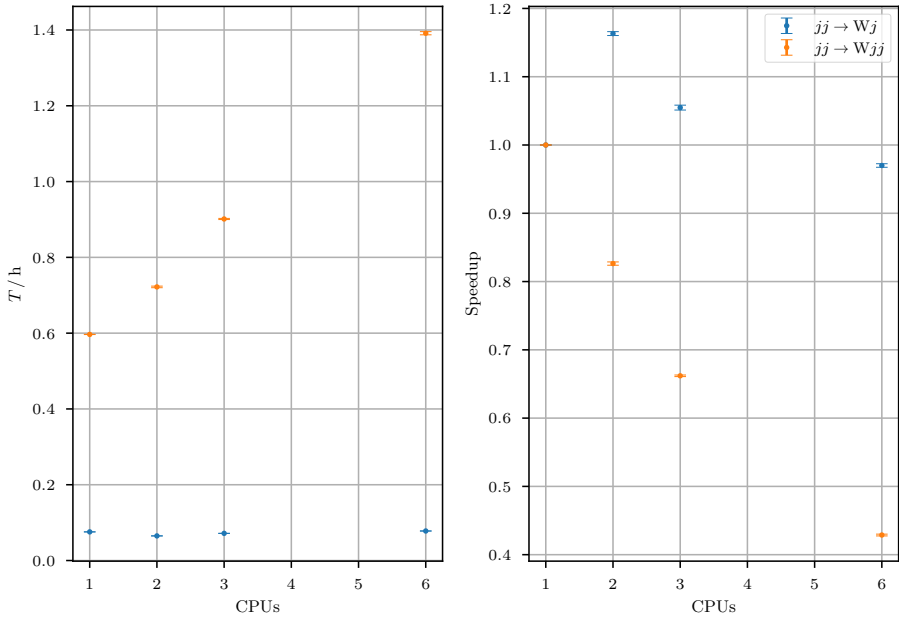


Figure 3.9: We show the speedup for a fixed number of MPI worker with increasing number of OpenMP threads.

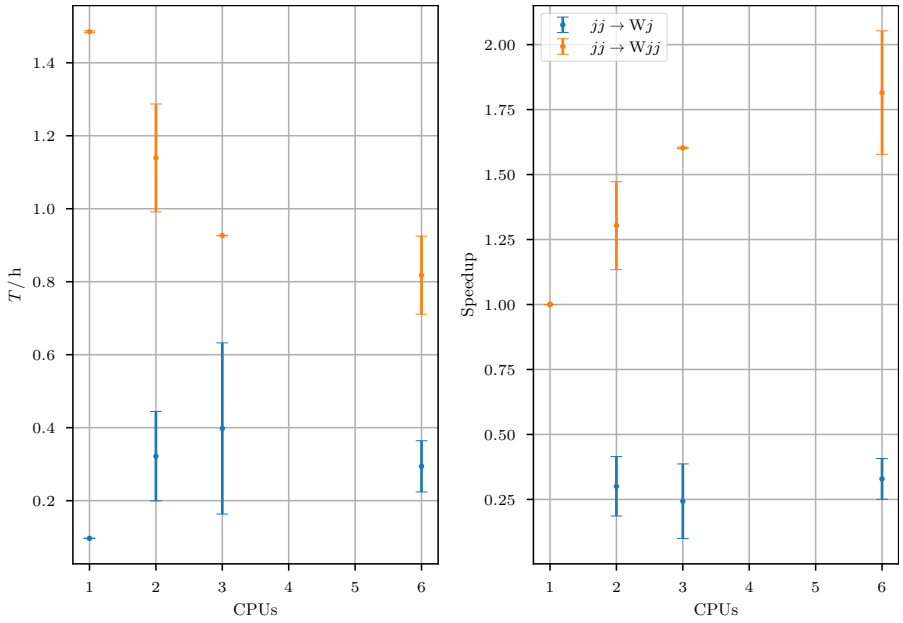


Figure 3.10: We show the speedup at an overall fixed number of cores (60) involved in the parallelization. We distribute the cores among MPI and OpenMP parallelization. For the latter we have to respect the node structure where each node consists of two CPUs each with 6 cores handling up to 6 threads without any performance penalty.

matrix element increases with the jet content, which then allows for a linearly increased speedup.

In section 4.6 we present tables that show the actual physical run-times for the different processes under consideration.

3.9 Future Improvements

To summarize, we recapitulate our reasoning about the low-level parallelization of the adaptive Monte Carlo algorithms and give an outlook about possible improvements.

We have decided against the use of a high-level API parallelization package, which would provide efficient and well-tested communication concepts and structures for computing-intense application, based on MPI. Instead, we chose a self-written, low-level implementation parallelization of the adaptive Monte Carlo algorithms with hands directly on MPI. We made this decision with a certain naivety such that we assumed that a low-level implementation would provide a higher efficiency and would be simple enough to implement. Both aspects have proven to be much more complicated than anticipated.

We see the effect that the run-time is saturated by the communication time for larger numbers of workers, e.g. $N \geq 30$. However, the naive load-balancing approach yields a surprisingly good parallel efficiency. A detailed profiling, especially with regard to larger numbers of workers, could reveal possible bottlenecks in the implementation. In addition, we have given a detailed description for (static) load balancing. Unfortunately, we could not complete the implementation in the time scale of this thesis [22] Therefore, we refrained from showing benchmark results for it, but we expect the static load balancer to further reduce the run-time and, thus, increase the parallel efficiency. Furthermore, the static load balancer could be extended by a server-client mechanism to collect the integration and adaptation results. In this model, a worker would act as a permanent receiver, which would collect the results of the other clients in a non-blocking fashion. However, this approach would be only apply well for larger numbers of worker where the omission of a single worker would not significantly impact the parallel efficiency.

In order to improve the efficiency of the parallelization, a detailed profiling should reveal bottlenecks or hot-spots of the parallel execution. The MPI standard explicitly defines a profiling interface to collect the contribution of individual communication routines to the overall run-time. In example, this profiling interfaces is utilized in the publicly-funded performance tool Scalasca [105], but at time of writing this thesis it did not support the Fortran 2008 bindings of the MPI standard which we applied. Other tools such as `mpiP` [106] are not longer actively developed and do not support all Fortran 2008 bindings. Commercial general-purpose solutions such as VTune [104] or TotalView [107] are hardly an alternative for free scientific work due to their expensive and "non-free" license agreements, as well as their closed-source code. Therefore, we tried avoiding to use them to profile the parallelization, which could have revealed unnecessary calls, redundant messages or inefficient load balancing.

Perhaps, the use of a high-level parallelization system could provide a better parallel efficiency due to more mature concepts and well-tested structures. For example, a dedicated

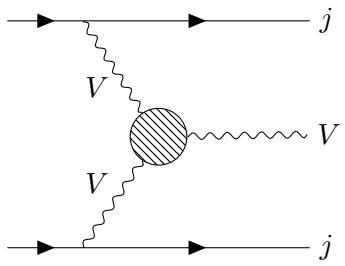
[22] In principle, the implementation of the static load balancer is fully functional, but it randomly yields incorrect results which need further investigations.

library for dynamic load balancing like the Zoltan parallel service library [108] or the Asynchronous Dynamic Load Balancer (ADLB) [109], which even provides Fortran 90 bindings. Further enhancements could be made directly in the use of more abstract and higher-leveled parallelization models such as provided by DIY [110], which are also based on MPI, but include appropriate load balancing, efficient input and output to/from parallel storage systems as well as the assignment of data into domains and their allocation to MPI processes.

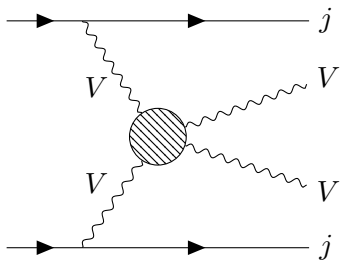
Concluding, we underestimated the complexity of parallelizing an algorithm, especially the combination of two algorithms such as the multi-channel approach and VEGAS. The parallel implementation of these turned out to be very challenging and I could only address this in very general and overviewing way in this thesis.[23] Nevertheless, we achieved our goal to provide an efficiently parallelized adaptive Monte Carlo integration.

[23] I refer the interested reader to the source code of the new integrator in WHIZARD [5] for a detailed reading on the *actual* implementation hurdles.

4 Transverse Modes and Higgs Bosons in Electroweak Vector-Boson Scattering at the LHC



(a) (Anomalous) Trilinear gauge interaction in vector-boson fusion (VBF).



(b) (Anomalous) Quartic gauge interaction in vector-boson scattering (VBS).

Figure 4.1: Self-interactions of the electroweak gauge bosons and Higgs boson, $V = \mathrm{W}, \mathrm{Z}$ and H, are predicted by the local, non-Abelian SM gauge group, further contributions due to new physics beyond the SM can be modeled by local contributions with a EFT. Those interactions are collectively represented by the blob(s).

The discovery of the Higgs in 2012 marks an important step toward a detailed understanding of nature at the smallest length scales. The success of the SM of particle physics is undoubted, but nevertheless, questions arise about new physics beyond the SM.

We want to extend our understanding of the SM beyond the TeV range, which is accessible at the LHC. For the search for new physics, we anticipate possible deviations from the SM to appear within the electroweak and Higgs sector. The details of the Higgs sector are currently under probe as well as trilinear and quartic couplings of electroweak gauge bosons, see fig. 4.1, which so far have not been observed at the LHC. However, a recent ATLAS analysis [25] claims evidence for the production of three massive vector bosons.

For our scenarios, we limit our investigations to quasi-elastic $2 \to 2$ scattering processes involving electroweak gauge bosons and the Higgs, in order to probe (anomalous) quartic gauge couplings. Deviations due to physics beyond the SM are weakly constrained by available experimental data. However, as a consequence, we facilitate our predictions with assumptions, firstly, that light fermions do not couple directly to the dynamics of new physics, since this would have a direct impact, e.g., on flavor physics. Secondly, that the local gauge symmetry of the SM retains its relevance beyond the TeV range, which is backed by present data at LHC. Lastly, that the new interactions are not charged under $\mathrm{SU}(3)_{\mathrm{color}}$, hence, they do not carry color, as it would considerably impact QCD at high energies.

Under these assumptions, we consider light fermion currents as spectator fields, e.g. for embedding the vector-boson scattering (VBS) processes in partonic processes, and focus completely on bosonic SM multiplet currents that probe new physics. However, heavy fermions $(\mathrm{t}, \mathrm{b}, \tau)$ should be taken into account as corrections due to their non-neglible masses, but we do not consider processes with those particles in the final-state.[1]

In any case, new physics that is coupled to SM bosons, will manifest itself in anomalous scattering matrix elements of those bosons, and should become accessible in high-energy VBS. As a common feature of this class of models, we expect the scattering matrix to be self-contained and complete in terms of SM bosons and eventual new-physics states, to a good approximation.

In this chapter, we present our results, which were published as [7], on predictions on VBS and associated or Higgs-pair production in vector-boson fusion (VBF) processes at LHC, see fig. 4.1, using the framework of the Standard Model Effective Field Theory (SMEFT) for energy ranges beyond the TeV scale.

The chapter is organized as follows, in section 4.1 we introduce the framework of the SMEFT and discuss the structure of the new physics in the electroweak and Higgs sector and state the underlying assumptions, particularly: (i) the effective field theory Lagrangian, (ii) the resulting dimension-eight operators and their Wilson coefficients, (iii) application and validity of the custodial symmetry, (iv) comparison of the scalar operators for different representations of the Higgs field, (v) and embedding the $VV \to VV$ process into a partonic process at the LHC. In section 4.2, we discuss the existence of a unitary bound on general scattering amplitudes following from first principles of QFT and the implications for the EFT approach. From the interactions given by the dimension-eight operator, we can determine the eigenamplitudes of the quasi-elastic $2 \to 2$ scatterings using custodial symmetry, which allows us to diagonalize the amplitudes for all vector-bosons modes in section 4.3, from which we construct unitary models exhibiting a strongly interacting continuum. These models give us the maximally allowed number of events that are consistent with QFT for a given high-energy VBS process matched to a low-energy SMEFT prediction with specific Wilson coefficients. Furthermore, we discuss simplified models, which contain a resonance and similarly parameterize the VBS amplitudes. In section 4.5 we outline the production setup for computing the models within the Monte Carlo event generator Whizard and the matrix-element generator Omega. For both models we present in section 4.6 numerical results and plots for selected Wilson coefficients and final states. We conclude in section 4.7 with a discussion of our results and their relevance to future analyses at the LHC or other future colliders.

At this point, I want to remark that the analytical expressions for the diagonalized amplitudes were calculated by Christian Fleper, whose work I continued after he left particle physics on short notice. My relevant part in the calculation of those amplitudes was to cross-check and validate the formulas and the implementation in Omega, i.e. the model `SSC2`. My major contribution to [7] was to combine his code with Whizard and to compute the actual results, as well as presenting them.

[1] Furthermore, they do not contribute to the initial-state at the LHC, except for the b-quark whose mass is still small compared to the TeV scale.

4.1 Standard Model Effective Field Theory

In the absence of a particular preferred theory for BSM physics, we resort to a bottom-up approach based on a formulation in an effective field theory (EFT) [111]. This allows us to describe deviations from the SM in a model-independent fashion, or if there is evidence for a deviation from the SM, it could give us an indication of its origins. So far, the only groundbreaking discovery of the LHC is a Higgs-like state [1, 2], no further new elementary particles or sufficiently significant deviations from the SM have been observed. Therefore, the focus of collider physics has shifted towards a detailed study of the electroweak symmetry breaking and the relationship with the Higgs boson. One of the most fundamental processes of the electroweak interaction is the electroweak gauge boson scattering, which is currently studied at the LHC [13, 15, 16, 112, 113] and will be a key process at the high-luminosity LHC or at future high-energy accelerators such as FCC [114] (pp-collider), ILC [115] or CLIC [116] (both e^+e^--collider). The strong suppression of electroweak VBS by the Higgs at the high center-of-mass energies is of particular interest [117, 118]. A intricate cancellation between the amplitudes involving quartic gauge coupling (QGC), triple gauge coupling (TGC) and Higgs exchange results in scattering amplitudes for longitudinally-polarized weak gauge bosons that do not grow with the energy, and, for a light Higgs boson, respect the upper bounds dictated by unitarity. We would expect that deviations from the SM indicate a new interaction in the electroweak symmetry breaking (EWSB) sector.

Individual contributions to the VBS scattering amplitudes are dominated by Goldstone bosons, which are directly related to the scattering amplitudes of longitudinally polarized W and Z [119, 120]. These classes of processes have become accessible at the LHC, [121, 122], requiring also a strong theory-side description of them.

The phenomenological descriptions of VBS processes, e.g. for data analyses, should smoothly interpolate between the low-energy behavior, which is well-defined by the SM and its parameters, and all possible asymptotic high-energy behaviors. The latter must thereby remain consistent with the universal principles of QFT. In the following, we will introduce and discuss the necessary steps that allow a corresponding parameterization of the asymptotic high-energy behavior, as well as interpolate between the low and high energy.

Effective Field Theory Lagrangian

Following the introduction from [123] for a genuine aQGC model with an elementary Higgs in linear realization, where the SM Higgs boson refers to a light electroweak scalar doublet. The effective theory, where the $\mathrm{SU(2)_L} \times \mathrm{U(1)_Y}$ gauge symmetry is linearly realized, can be expressed as

$$\mathcal{L}_{\mathrm{SMEFT}} = \mathcal{L}_{\mathrm{SM}} + \sum_{n=5}^{\infty} \frac{f_n}{\Lambda^{n-4}} \mathcal{O}_n, \tag{4.1}$$

where the dimension-n operators $\mathcal{O}_n$ involve SM fields with Wilson coefficients f_n (which allow for variations of importance of the individual operators) and a characteristic scale Λ, at which heavy fields are integrated out. We have made the contribution of the SM, the dimension-four operators, explicit.

We will neglect contributions by $n = 5$ as there is only one operator, see [124], which violates lepton number conservation with $\Delta L = \pm 2$, but can be used to introduce Majorana masses for neutrinos, see [125]. Furthermore, we skip dimension-seven operators as they violate both baryon- and lepton-baryon number conservation, too, see [126]. In general, odd-dimensional terms are neglected out of the assumption that baryon and lepton numbers still need to be conserved in the effective theory, and possible violations, if there are any, are expected to arise at a very large energy scale, e.g. the Planck scale, [127].

With the linear Higgs representation, deviations from the SM predictions for Higgs couplings and trilinear and quartic gauge boson couplings first appear at dimension $n = 6$, as introduced in [8–11]. AQGC are already introduced at the level of dimension-six operators, but are directly related to anomalous trilinear gauge coupling (aTGC).[2] In particular, the aQGC are not independent of changes in the Higgs couplings in aTGC. Furthermore, the tensor and Lorentz structure introduced by the dimension-six operators is not general enough to cover all possible aQGC. However, if we want to describe aQGC independently and separately from aTGC, we have to increase the order of the expansion to dimension-eight operators. That dimension-eight operators are necessary, can be seen from the pure modification to QGC: gauge bosons can be derived either from a field strength tensor or the covariant derivative acting on the Higgs. Both expressions have dimension two in the canonical power-counting. We needed a total of four terms of field strength tensors and covariant derivative acting on the Higgs, in order to find all possible contributions to the aQGC.

We can then truncate the infinite series of operators in eq. (4.1), and we arrive at

$$\mathcal{L}^{(8)}_{\text{SMEFT}} = \mathcal{L}_{\text{SM}} + \frac{1}{\Lambda^2}\mathcal{L}_{\text{dim.-6}} + \frac{1}{\Lambda^4}\mathcal{L}_{\text{dim.-8}} \subset \mathcal{L}_{\text{SMEFT}}. \tag{4.2}$$

The EFT field content is given by fermions, gluons, vector gauge boson and the $\text{SU}(2)_\text{L}$ Higgs doublet $\mathbf{\Phi}$ which in the linear representation consists of the physical Higgs boson and the three Goldstone boson $w^\pm, w^3$.[3] For the electroweak gauge bosons, we take the notation from [125]

$$\boldsymbol{D}_\mu = \partial_\mu \mathbf{\Phi} - \mathrm{i}g\mathbf{W}_\mu \mathbf{\Phi} - \mathrm{i}g'\mathbf{\Phi}\mathbf{B}_\mu, \tag{4.3}$$

$$\mathbf{W}_{\mu\nu} = \partial_\mu \mathbf{W}_\nu - \partial_\nu \mathbf{W}_\mu - \mathrm{i}g[\mathbf{W}_\mu, \mathbf{W}_\nu], \tag{4.4}$$

$$\mathbf{B}_{\mu\nu} = \partial_\mu \mathbf{B}_\nu - \partial_\nu \mathbf{B}_\nu, \tag{4.5}$$

with

$$\mathbf{W}_\mu = W^a_\mu \frac{\tau^a}{2}, \quad \mathbf{B}_\mu = B_\mu \frac{\tau^3}{2}. \tag{4.6}$$

The $\mathbf{W}_\mu$ fields transform locally as $\mathbf{W}_\mu \to U^\dagger_\text{L} \mathbf{W}_\mu U_\text{L}$ and the $\mathbf{B}_\mu$ field transforms covariantly only under a $U(1)_\text{R}$ subgroup of $\text{SU}(2)_\text{R}$ which explicitly breaks the global symmetry of the linear Higgs. After EWSB, the global symmetry becomes the approximate custodial symmetry $\text{SU}(2)_\text{C}$ which is also broken by the hypercharge symmetry $\text{U}_\text{Y}(1)$.

In our definition of the Higgs multiplets, we deviate from [7] where the Higgs field is introduced as 2×2 Hermitian matrix field. We follow [123, 127] where the Higgs field is

[2] Therefore, we would expect to see deviations from the SM first in TGC.

[3] We focus solely on the bosonic SM multiplets and, therefore, do not write fermions and gluons explicitly.

given as a doublet field $\mathbf{\Phi}$. In a side-by-side comparison, the Higgs doublet field $\mathbf{\Phi}$ and the 2×2 Hermitian matrix Higgs field $\mathbf{H}$ are given in unitary gauge by

$$\mathbf{\Phi} = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ v + h \end{pmatrix}, \tag{4.7}$$

$$\mathbf{H} = \frac{1}{2} \begin{pmatrix} v + h & 0 \\ 0 & v + h \end{pmatrix}, \tag{4.8}$$

where the components h, v are the physical Higgs and vacuum expectation value of the Higgs potential. We show later how this representation of the Higgs doublet connect to the representation as 2×2 Hermitian matrix Higgs field (see section 4.1).

Dimension-Eight Operators

Pure aQGC are first introduced by dimension-eight operators. In the case of dimension-six operators, modifications to aQGC would also lead to changes in aTGC, and one would expect to see deviations from SM predictions there first. As already discussed, we need in total a combination of four terms of field strength tensors and covariant derivatives acting on the Higgs, i.e. $\boldsymbol{D}_\mu \mathbf{\Phi}, \mathbf{W}_{\mu\nu}$ or $\mathbf{B}_{\mu\nu}$, for constructing dimension-eight operators. Since the number of open Lorentz indices is only one or two, there are only three possibilities [123, 128] to combine them.

In detail, we construct the operators from currents that probe new, (possibly) non-local dynamics. The simplest Higgs-field currents are given by

$$\begin{aligned} J_{\mathrm{H}}^{(2)} &= \left[\mathbf{\Phi}^\dagger \mathbf{\Phi}\right], \\ J_{\mathrm{H}}^{(4)} &= \left[\left(\boldsymbol{D}_\mu \mathbf{\Phi}\right)^\dagger \left(\boldsymbol{D}^\mu \mathbf{\Phi}\right)\right], \\ J_{\mathrm{H},\mu\nu}^{(4)} &= \left[\left(\boldsymbol{D}_\mu \mathbf{\Phi}\right)^\dagger \left(\boldsymbol{D}_\nu \mathbf{\Phi}\right)\right]. \end{aligned} \tag{4.9}$$

The gauge-field tensors can be combined as

$$J_{\mathrm{W}}^{(4)} = g^2 \,\mathrm{tr}\left[\mathbf{W}_{\mu\nu}\mathbf{W}^{\mu\nu}\right], \qquad J_{\mathrm{W},\mu\nu}^{(4)} = g^2 \,\mathrm{tr}\left[\mathbf{W}_{\mu\rho}\mathbf{W}^{\rho\nu}\right], \tag{4.10}$$

$$J_{\mathrm{B}}^{(4)} = g^2 \,\mathrm{tr}\left[\mathbf{B}_{\mu\nu}\mathbf{B}^{\mu\nu}\right], \qquad J_{\mathrm{B},\mu\nu}^{(4)} = g^2 \,\mathrm{tr}\left[\mathbf{B}_{\mu\rho}\mathbf{B}^{\rho\nu}\right]. \tag{4.11}$$

These terms are electroweak singlets. Non-singlet currents can be likewise constructed.

In the so-called Eboli basis [123, 128], we have operators that couple only to the covariant derivative and the Higgs field, the scalar operators,

$$\mathcal{L}_{S,0} = F_{S,0} \left[\left(\boldsymbol{D}_\mu \mathbf{\Phi}\right)^\dagger \boldsymbol{D}_\nu \mathbf{\Phi}\right] \left[\left(\boldsymbol{D}^\mu \mathbf{\Phi}\right)^\dagger \boldsymbol{D}^\nu \mathbf{\Phi}\right], \tag{4.12}$$

$$\mathcal{L}_{S,1} = F_{S,1} \left[\left(\boldsymbol{D}_\mu \mathbf{\Phi}\right)^\dagger \boldsymbol{D}^\mu \mathbf{\Phi}\right] \left[\left(\boldsymbol{D}_\nu \mathbf{\Phi}\right)^\dagger \boldsymbol{D}^\nu \mathbf{\Phi}\right], \tag{4.13}$$

$$\mathcal{L}_{S,2} = F_{S,2} \left[\left(\boldsymbol{D}_\mu \mathbf{\Phi}\right)^\dagger \boldsymbol{D}_\nu \mathbf{\Phi}\right] \left[\left(\boldsymbol{D}^\nu \mathbf{\Phi}\right)^\dagger \boldsymbol{D}^\mu \mathbf{\Phi}\right], \tag{4.14}$$

field-strength tensor and covariant derivatives, the mixed operators,

$$\mathcal{L}_{M,0} = -g^2 F_{M,0} \left[(\boldsymbol{D}_\mu \boldsymbol{\Phi})^\dagger (\boldsymbol{D}^\mu \boldsymbol{\Phi})\right] \operatorname{tr}\left[\mathbf{W}_{\nu\rho}\mathbf{W}^{\nu\rho}\right], \tag{4.15}$$

$$\mathcal{L}_{M,1} = -g^2 F_{M,1} \left[(\boldsymbol{D}_\mu \boldsymbol{\Phi})^\dagger (\boldsymbol{D}^\rho \boldsymbol{\Phi})\right] \operatorname{tr}\left[\mathbf{W}_{\nu\rho}\mathbf{W}^{\nu\mu}\right], \tag{4.16}$$

$$\mathcal{L}_{M,2} = -g^2 F_{M,2} \left[(\boldsymbol{D}_\mu \boldsymbol{\Phi})^\dagger (\boldsymbol{D}^\mu \boldsymbol{\Phi})\right] \operatorname{tr}\left[\mathbf{B}_{\nu\rho}\mathbf{B}^{\nu\rho}\right], \tag{4.17}$$

$$\mathcal{L}_{M,3} = -g^2 F_{M,3} \left[(\boldsymbol{D}_\mu \boldsymbol{\Phi})^\dagger (\boldsymbol{D}^\rho \boldsymbol{\Phi})\right] \operatorname{tr}\left[\mathbf{B}_{\nu\rho}\mathbf{B}^{\nu\mu}\right], \tag{4.18}$$

$$\mathcal{L}_{M,4} = -g g' F_{M,4} \operatorname{tr}\left[\left(\boldsymbol{D}_\mu \boldsymbol{\Phi}\right)^\dagger \boldsymbol{W}_{\nu\rho} \left(\boldsymbol{D}^\mu \boldsymbol{\Phi}\right) \mathbf{B}^{\nu\rho}\right], \tag{4.19}$$

$$\mathcal{L}_{M,5} = -g g' F_{M,5} \operatorname{tr}\left[\left(\boldsymbol{D}_\mu \boldsymbol{\Phi}\right)^\dagger \boldsymbol{W}_{\nu\rho} \left(\boldsymbol{D}^\rho \boldsymbol{\Phi}\right) \mathbf{B}^{\nu\mu}\right], \tag{4.20}$$

$$\mathcal{L}_{M,7} = -g^2 F_{M,7} \operatorname{tr}\left[(\boldsymbol{D}_\mu \boldsymbol{\Phi})^\dagger \mathbf{W}_{\nu\rho} \mathbf{W}^{\nu\mu} (\boldsymbol{D}^\rho \boldsymbol{\Phi})\right]. \tag{4.21}$$

only field-strength tensor, which couple the gauge-field currents to themselves, the tensor operators,

$$\mathcal{L}_{T,0} = g^4 F_{T,0} \operatorname{tr}\left[\mathbf{W}_{\mu\nu}\mathbf{W}^{\mu\nu}\right] \operatorname{tr}\left[\mathbf{W}_{\alpha\beta}\mathbf{W}^{\alpha\beta}\right], \tag{4.22}$$

$$\mathcal{L}_{T,1} = g^4 F_{T,1} \operatorname{tr}\left[\mathbf{W}_{\alpha\nu}\mathbf{W}^{\mu\beta}\right] \operatorname{tr}\left[\mathbf{W}_{\mu\beta}\mathbf{W}^{\alpha\nu}\right], \tag{4.23}$$

$$\mathcal{L}_{T,2} = g^4 F_{T,2} \operatorname{tr}\left[\mathbf{W}_{\alpha\mu}\mathbf{W}^{\mu\beta}\right] \operatorname{tr}\left[\mathbf{W}_{\beta\nu}\mathbf{W}^{\nu\alpha}\right], \tag{4.24}$$

$$\mathcal{L}_{T,5} = g^4 F_{T,5} \operatorname{tr}\left[\mathbf{W}_{\mu\nu}\mathbf{W}^{\mu\nu}\right] \operatorname{tr}\left[\mathbf{B}_{\alpha\beta}\mathbf{B}^{\alpha\beta}\right], \tag{4.25}$$

$$\mathcal{L}_{T,6} = g^4 F_{T,6} \operatorname{tr}\left[\mathbf{W}_{\alpha\nu}\mathbf{W}^{\mu\beta}\right] \operatorname{tr}\left[\mathbf{B}_{\mu\beta}\mathbf{B}^{\alpha\nu}\right], \tag{4.26}$$

$$\mathcal{L}_{T,7} = g^4 F_{T,7} \operatorname{tr}\left[\mathbf{W}_{\alpha\mu}\mathbf{W}^{\mu\beta}\right] \operatorname{tr}\left[\mathbf{B}_{\beta\nu}\mathbf{B}^{\nu\alpha}\right], \tag{4.27}$$

$$\mathcal{L}_{T,8} = g'^4 F_{T,8} \operatorname{tr}\left[\mathbf{B}_{\mu\nu}\mathbf{B}^{\mu\nu}\right] \operatorname{tr}\left[\mathbf{B}_{\alpha\beta}\mathbf{B}^{\alpha\beta}\right], \tag{4.28}$$

$$\mathcal{L}_{T,9} = g'^4 F_{T,9} \operatorname{tr}\left[\mathbf{B}_{\alpha\mu}\mathbf{B}^{\mu\beta}\right] \operatorname{tr}\left[\mathbf{B}_{\beta\nu}\mathbf{B}^{\nu\alpha}\right]. \tag{4.29}$$

We have adopted the naming convention of[127–129] and redundant interactions have been removed to arrive at a linearly independent set: In [128], the two additional operators,

$$\mathcal{L}_{T,3} = \operatorname{tr}\left[\mathbf{W}_{\alpha\mu}\mathbf{W}^{\mu\beta}\mathbf{W}^{\nu\alpha}\right] \mathbf{B}_{\beta\nu}, \tag{4.30}$$

$$\mathcal{L}_{T,4} = \operatorname{tr}\left[\mathbf{W}_{\alpha\mu}\mathbf{W}^{\alpha\mu}\mathbf{W}^{\beta\nu}\right] \mathbf{B}_{\beta\nu}, \tag{4.31}$$

have been defined. Both operators vanish identically. For $\mathcal{L}_{T,3}$, the trace is symmetric under permutations of indices β and ν, whereas the field-strength tensor $B_{\beta\nu}$ is anti-symmetric, hence, the product vanishes. For $\mathcal{L}_{T,4}$, the trace itself vanishes. Furthermore, the operator

$$\mathcal{L}_{M,6} = \left[(\boldsymbol{D}_\mu \boldsymbol{\Phi})^\dagger \mathbf{W}_{\beta\nu} \mathbf{W}^{\beta\nu} \boldsymbol{D}^\mu \boldsymbol{\Phi}\right], \tag{4.32}$$

is linearly-dependent on $\mathcal{L}_{M,0}$ by the relation $\mathcal{L}_{M,6} = \frac{1}{2}\mathcal{L}_{M,0}$.

Furthermore, the additional scalar operator $\mathcal{L}_{S,2}$ has been introduced [123, 130], as opposed to in [7, 128]. The third scalar operator is important in order to relate the dimension-eight operators eq. (4.12) of the Eboli base with the operators of the electroweak chiral Lagrangian [125] (see appendix B.1 for an overview) and to the basis definition used in [7]. As we will see later on, the implementation of WHIZARD's scalar operators was originally based on the electroweak chiral Lagrangian with a non-linearly realized symmetry. We show how the coefficients of the canonical dimension-eight operators can be mapped to the coefficients of the scalar operators of chiral theory and which role the operator $\mathcal{L}_{S,2}$ plays, in section 4.1. Before that, we will focus on the underlying symmetries and their extension for our models.

Custodial Symmetry

In the following we introduce the approximate custodial symmetry $\mathrm{SU}(2)_\mathrm{C}$ [131–133], which will later play a crucial role in the analytical derivation of the unitarization corrections of the EFT regarding VBS.

In eq. (4.12) we have introduced the dimension-eight operators in the Eboli basis[128], with a linearly-realized Higgs boson as Higgs doublet $\mathbf{\Phi}$. In the following, we present the convention for the Higgs field and the corresponding changes to the dimension-eight operators used in our study [17].

Following along the lines of [125, pp. 28–29] we extend the (local) symmetry of the standard model, $\mathrm{SU}(2)_\mathrm{L} \times \mathrm{U}(1)_\mathrm{Y}$, to the global symmetry $\mathrm{SU}(2)_\mathrm{L} \times \mathrm{SU}(2)_\mathrm{R}$, where for the Higgs sector the relation $\mathrm{U}(1)_\mathrm{R} \subset \mathrm{SU}(2)_\mathrm{R}$ should apply. For a Higgs multiplet $\mathbf{H}$,[4] the following transformation law applies,

$$\mathbf{H} \to U_\mathrm{L} \mathbf{H} U_\mathrm{R}^\dagger, \tag{4.33}$$

where $U_\mathrm{L} \in \mathrm{SU}(2)_\mathrm{L}$ and $U_\mathrm{R} \in \mathrm{SU}(2)_\mathrm{R}$. The vector bosons are assumed to transform as singlets under $\mathrm{SU}(2)_\mathrm{R}$. It may be desirable to assign fermions to $\mathrm{SU}(2)_\mathrm{R}$ doublets (as already realized for $\mathrm{SU}(2)_\mathrm{L}$), but this is not a valid symmetry of the fermions. This symmetry requires assumptions that are not realized in nature: the masses of the up- and down-like quarks have to be the same and the CKM matrix must be trivial and with that, also flavor physics! However, the CKM matrix is not trivial, and the quark masses are not the same, e.g. the top quark mass is two orders of magnitude larger than the mass of the bottom (and all other quark masses).

In addition, the weak interaction breaks the $\mathrm{SU}(2)_\mathrm{R}$ symmetry as the hypercharge boson B transforms only under $\mathrm{U}(1)_\mathrm{R}$, which is not consistent with the $\mathrm{SU}(2)_\mathrm{R}$ symmetry. However, the coupling of the hypercharge boson B is proportional to $\sin\theta_\mathrm{w}$,[5], which is not a large parameter, and the fermions play for aQGC interaction only a role in one-loop corrections. Thus, we neglect these interactions by assuming the $\mathrm{SU}(2)_\mathrm{R}$ is approximately realized.

In the limit where up and down masses are equal, and we have $e \to 0, \sin\theta_\mathrm{w} \to 0$ and $e/\sin\theta_\mathrm{w}$ fixed, the $\mathrm{SU}(2)_\mathrm{R}$ is exactly realized in the SM Lagrangian. In this limit, the electroweak symmetry breaking brings the global symmetry down to the custodial symmetry $\mathrm{SU}(2)_\mathrm{C}$, under which the Higgs multiplet transforms as,

$$\mathbf{H} \to U_\mathrm{C} \mathbf{H} U_\mathrm{C}^\dagger, \tag{4.34}$$

with $U_\mathrm{C} \in \mathrm{SU}(2)_\mathrm{C}$. Under this $\mathrm{SU}(2)_\mathrm{C}$ transformation, to which we will refer later as custodial isospin, the W and Z vector fields form a mass-degenerate triplet. We will use the latter to simplify aQGC amplitudes of vector-boson scattering with so-called isospin amplitudes [134].

For the Higgs field, the matrix notation is chosen as it manifestly represents the larger global symmetry $\mathrm{O}(4) \sim \mathrm{SU}(2)_\mathrm{L} \times \mathrm{SU}(2)_\mathrm{R}$, which becomes after the EWSB the

[4] As earlier mentioned, we refer to the Higgs doublet as $\mathbf{\Phi}$, and to the Higgs multiplet in general as $\mathbf{H}$.

[5] The Weinberg angle θ_w is introduced in order to rotate the weak boson W^0 and the hypercharge boson B into mass eigenstates, i.e. the massive Z and the massless photon, after electroweak symmetry breaking. The value is given as $\sin^2\theta_\mathrm{w}(M_\mathrm{Z}) = 0.231\,26 \pm 0.000\,05$ in the $\overline{\mathrm{MS}}$ scheme, [24].

approximate custodial symmetry $\mathrm{SU}(2)_\mathrm{C}$. The Higgs multiplet, written in the form of a 2×2 Hermitian matrix, is then given as

$$\boldsymbol{H} = \frac{1}{2}\begin{pmatrix} v + h - \mathrm{i}w^3 & -\mathrm{i}\sqrt{2}w^+ \\ -\mathrm{i}\sqrt{2}w^- & v + h + \mathrm{i}w^3 \end{pmatrix}, \tag{4.35}$$

where the components $h, w^\pm, w^3$ are the physical Higgs boson and the three unphysical Goldstone scalars, respectively. v denotes the numerical Higgs vacuum expectation value (vev), $v = 246\,\mathrm{GeV}$ [24]. The (actual) Higgs field is just a unit matrix multiplied by $v + h$, while components with the Goldstone boson correspond to the Pauli matrices τ_i.

The three transformations left, right and custodial, $U_\mathrm{L} \in \mathrm{SU}(2)_\mathrm{L}, U_\mathrm{R} \in \mathrm{SU}(2)_\mathrm{R}$ and $U_\mathrm{C} \in \mathrm{SU}(2)_\mathrm{C}$, act as

$$\boldsymbol{H} \to U_\mathrm{L}\boldsymbol{H}, \quad \boldsymbol{H} \to \boldsymbol{H}U_\mathrm{R}^\dagger, \quad \boldsymbol{H} \to U_\mathrm{C}\boldsymbol{H}U_\mathrm{C}^\dagger, \tag{4.36}$$

respectively. With the exception of $\tau^{1,2} \in \mathrm{SU}(2)_\mathrm{C}$, the global symmetries of $\mathrm{SU}(2)_\mathrm{L}$ and $\mathrm{SU}(2)_\mathrm{R}$ are realized as local gauge symmetries. The Higgs field decomposes under the custodial transformation into a singlet and triplet, which corresponds to the physical Higgs and the Goldstone bosons. Conversely, under $\mathrm{SU}(2)_\mathrm{L}$ gauge transformations, the two columns of the Higgs matrix transform independently as the conventional complex doublet $\mathbf{H}$ in eq. (4.7) and its charge conjugate. In unitary gauge, the Goldstones "are eaten" by the electroweak gauge bosons when the latter become massive. The longitudinal polarizations of the electroweak gauge bosons are thus provided by the Goldstone bosons. The Higgs field matrix then reduces to,

$$H = \frac{1}{2}(v + h)\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{4.37}$$

The plain SM lagrangian, which can be interpreted as the lowest order of the EFT expansion, with only the kinematic terms of the bosons and the Higgs potential, reads

$$\begin{aligned}\boldsymbol{\mathcal{L}}_\mathrm{SM} \supset -\frac{1}{2}\,\mathrm{tr}[\mathbf{W}_{\mu\nu}\mathbf{W}^{\mu\nu}] - \frac{1}{2}\,\mathrm{tr}[\mathbf{B}_{\mu\nu}\mathbf{B}^{\mu\nu}] \\ + \mathrm{tr}[\left(D_\mu \boldsymbol{H}\right)^\dagger D^\mu \boldsymbol{H}] + \mu^2\,\mathrm{tr}[\boldsymbol{H}^\dagger\boldsymbol{H}] + \frac{\lambda}{2}\left(\mathrm{tr}[\boldsymbol{H}^\dagger\boldsymbol{H}]\right)^2.\end{aligned} \tag{4.38}$$

For the precise definition of higher-dimensional operators, the free parameters have to be expressed in terms of observables, order by order. The free parameters are defined by observable quantities,

$$g = 2\frac{m_\mathrm{W}}{v}, \tag{4.39}$$

$$g' = 2\frac{\sqrt{m_\mathrm{Z}^2 - m_\mathrm{W}^2}}{v}, \tag{4.40}$$

$$\mu^2 = \frac{1}{2}m_\mathrm{H}^2, \tag{4.41}$$

$$\lambda = \frac{m_\mathrm{H}}{v^2}, \tag{4.42}$$

applicable to the operator expansion at tree level and beyond in eq. (4.2), with the particle masses and the Higgs vacuum expectation value v as fixed input. Importantly, the definition of g and g' fixes the covariant field strength and the covariant derivative of the Higgs that we apply for the construction of higher-dimensional operators.

With the Higgs matrix notation we can rewrite the operators such that the custodial symmetry is represented manifestly. Since the product $(\boldsymbol{D}_\mu \mathbf{H})^\dagger \boldsymbol{D}_\nu \mathbf{H}$ is a matrix, we take the trace operation over the monomial. The replacement $[(\boldsymbol{D}_\mu \mathbf{\Phi})^\dagger \boldsymbol{D}_\nu \mathbf{\Phi}] \to \mathrm{tr}[(\boldsymbol{D}_\mu \mathbf{H})^\dagger \boldsymbol{D}_\nu \mathbf{H}]$ in the operators with $\mathbf{H}$ in the matrix notation yields the basis definition of our study [7], which we will use in the following.

Relation of Scalar Operators between Doublet and Matrix Notation

For the electroweak chiral Lagrangian with a light Higgs boson the following operators were introduced in [17] analogously to the scalar operators of the Eboli basis in eq. (4.12),

$$\mathcal{L}_{S,0} = \hat{F}_{S,0}\, \mathrm{tr}\left[(\boldsymbol{D}_\mu \mathbf{H})^\dagger \boldsymbol{D}_\nu \mathbf{H}\right] \mathrm{tr}\left[(\boldsymbol{D}^\mu \mathbf{H})^\dagger \boldsymbol{D}^\nu \mathbf{H}\right], \tag{4.43}$$

$$\mathcal{L}_{S,1} = \hat{F}_{S,1}\, \mathrm{tr}\left[(\boldsymbol{D}_\mu \mathbf{H})^\dagger \boldsymbol{D}^\mu \mathbf{H}\right] \mathrm{tr}\left[(\boldsymbol{D}_\nu \mathbf{H})^\dagger \boldsymbol{D}^\nu \mathbf{H}\right], \tag{4.44}$$

where $\mathbf{H}$ refers to the 2×2 Hermitian Higgs matrix field, see for the definition eq. (4.8). In order to compare the operator sets, we find that that the monomials of the operators from the scalar operators eq. (4.12) and eq. (4.43) are given by [135],

$$\begin{aligned}\left[(\boldsymbol{D}_\mu \mathbf{\Phi})^\dagger \boldsymbol{D}_\nu \mathbf{\Phi}\right]_{\text{Eboli}} &= \frac{1}{2}(\partial_\mu h)(\partial_\nu h) + M_{\mathrm{W}}^2 \mathrm{W}^-{}_\mu \mathrm{W}^+{}_\nu \left(1+\frac{h}{v}\right)^2 + \frac{M_{\mathrm{Z}}^2}{2}\mathrm{Z}_\mu \mathrm{Z}_\nu \left(1+\frac{h}{v}\right)^2 \\ &\quad + \frac{\mathrm{i}M_{\mathrm{Z}}}{2}(\mathrm{Z}_\mu(\partial_\nu h) - \mathrm{Z}_\nu(\partial_\mu h))\left(1+\frac{h}{v}\right), \end{aligned} \tag{4.45}$$

$$\begin{aligned}\mathrm{tr}\left[(\boldsymbol{D}_\mu \mathbf{H})^\dagger \boldsymbol{D}_\nu \mathbf{H}\right]_{\text{chiral}} &= \frac{1}{2}(\partial_\mu h)(\partial_\nu h) + \frac{M_{\mathrm{W}}^2}{2}(\mathrm{W}^-{}_\mu \mathrm{W}^+{}_\nu + \mathrm{W}^-{}_\nu \mathrm{W}^+{}_\mu)\left(1+\frac{h}{v}\right)^2 \\ &\quad + \frac{M_{\mathrm{Z}}^2}{2}\mathrm{Z}_\mu \mathrm{Z}_\nu \left(1+\frac{h}{v}\right)^2. \end{aligned} \tag{4.46}$$

Comparing them, it follows,

$$\mathrm{tr}\left[(\boldsymbol{D}_\mu \mathbf{H})^\dagger \boldsymbol{D}_\nu \mathbf{H}\right]_{\text{chiral}} = \frac{1}{2}\left(\left[(\boldsymbol{D}_\mu \mathbf{\Phi})^\dagger \boldsymbol{D}_\nu \mathbf{\Phi}\right]_{\text{Eboli}} + \left[(\boldsymbol{D}_\nu \mathbf{\Phi})^\dagger \boldsymbol{D}_\mu \mathbf{\Phi}\right]_{\text{Eboli}}\right). \tag{4.47}$$

Hence, the operator coefficients are related by [129]

$$\frac{16}{v^4}\alpha_4 = \hat{F}_{S,0} = F_{S,0} + F_{S,2}, \tag{4.48}$$

$$\frac{16}{v^4}\alpha_5 = \hat{F}_{S,1} = F_{S,1}, \tag{4.49}$$

where we added for completeness the coefficients α_4 and α_5 of the electroweak chiral Lagrangian (see appendix B.1). This applies both to the non-linear formulation without Higgs boson and with Higgs boson of the electroweak chiral theory. In the former case only the relations between aQGC are taken into account, since no vertices with Higgs

boson appear then in the operators $\mathcal{L}_4$ and $\mathcal{L}_5$. Furthermore, custodial $\mathrm{SU}(2)_\mathrm{C}$ isospin conservation requires then $F_{S,0} = F_{S,2}$, hence, $\hat{F}_{S,0} = 2F_{S,0}$.

For the case, $F_{S,0} \neq F_{S,2}$, the previous subset of operators does not suffice to mutually model the coefficients, neither in the chiral Lagrangian nor in the Eboli-operator basis. For this, the operators that mix the custodial symmetry $\mathrm{SU}(2)_\mathrm{C}$ breaking operator $T = \Sigma \frac{\tau^3}{2} \Sigma^\dagger$ with V_μ become necessary [125, 136],

$$\mathcal{L}_6 = \alpha_6 \,\mathrm{tr}\left[V_\mu V_\nu\right] \mathrm{tr}\left[TV^\mu\right] \mathrm{tr}\left[TV^\nu\right], \tag{4.50}$$

$$\mathcal{L}_7 = \alpha_7 \,\mathrm{tr}\left[V_\mu V^\mu\right] \mathrm{tr}\left[TV_\nu\right] \mathrm{tr}\left[TV^\nu\right]. \tag{4.51}$$

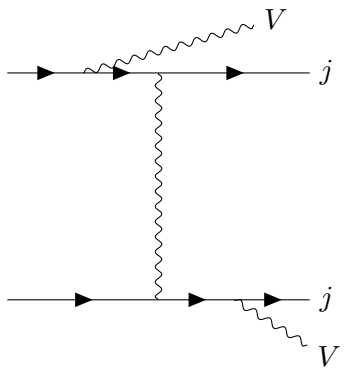
However, we can neglect the operator $\mathcal{L}_7$. For the linear representation of the Higgs within the electroweak chiral Lagrangian, where we assign dimension one to Σ in chiral counting, the dimension of the operator changes to 10 [125, p. 35] and, therefore, is beyond our considerations. The relations for the aQGC vertices for $F_{S,0} \neq F_{S,2}$ are given in appendix B.2, including the analytical expressions for the contributions of the operators to the aQGC.

With the above definition we can briefly discuss the comparability of the different definitions of operators. We already have the relations eq. (4.48) between the scalar operator coefficients. The question is whether these relations can be applied to obtain genuinely equivalent results from different operator sets, with equivalence in the sense that if we set the operator coefficients correspondingly in relation, then the physical observables are equal for all the definitions at LO. This is especially true if the sets lead to the same Feynman rules for all vertices that deviate from the SM.

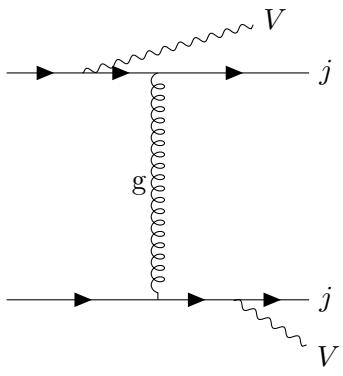
This is the case with the additional $\mathcal{L}_{S,2}$ operator as we can write down the relations between the quartic gauge boson vertices in the same way. Still, without $\mathcal{L}_{S,2}$ one can find vertex-specific conversion rules [135, p. 61] as the vertices for $\mathrm{W^+W^-W^+W^-}, \mathrm{W^+W^-ZZ}$ and ZZZZ contain at least two identical particles. This means that, after symmetrizing, the two vertices $\mathrm{W^+W^-W^+W^-}$ and $\mathrm{W^+W^-ZZ}$ can be generated from Feynman rules with two Lorentz structures each, instead of three Lorentz structures as one may naively assume. Therefore, it is sufficient to have two operators with two different Lorentz structures from which all possible Lorentz structures can then be generated for a single vertex. For the fully symmetric vertex ZZZZ, there is only one Lorentz structure for the scalar operators.

We note, however, that this is only a provisional solution and one should instead include the additional operator $\mathcal{L}_{S,2}$ for a correct comparison. Under certain circumstances, the latter may even be mandatory, for example in the VBS-production process $\mathrm{pp} \to \mathrm{W^+W^-}jj$: For the partonic subprocess $\mathrm{ud} \to \mathrm{udW^+W^-}$, both, $\mathrm{W^+W^-W^+W^-}$ and $\mathrm{W^+W^-ZZ}$, vertices contribute, in which case the operator sets can only be compared when $\mathcal{L}_{S,2}$ is included.

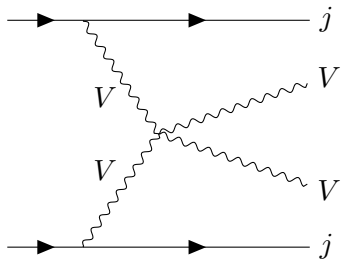
In summary, we were able to show that the coefficients of the chiral scalar operators and the scalar operator of the Eboli basis can be modeled mutually, when isospin is conserved, or broken, and for the latter, by picking up the isospin-breaking $\mathcal{L}_6$ operator in the chiral basis. The connection for the operators in eq. (4.43) is only done in the isospin-conserved case, since here the $\mathrm{SU}(2)_\mathrm{C}$ conservation is required explicitly.



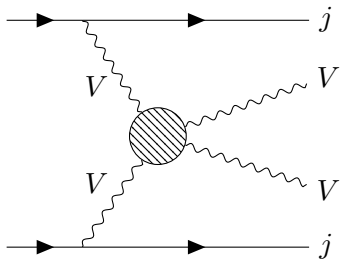
(a) Vector boson emission.



(b) QCD-induced background.



(c) Quartic gauge interaction from the SM.



(d) Anomalous quartic gauge interaction.

Figure 4.2: Examples of Feynman diagrams contributing to vector-boson scattering where the anomalous quartic coupling is modeled by the dimension-eight operators of SMEFT and all other interactions originate from the SM.

Embedding the VBS Matrix Element

The main goal of our study is the discussion of aQGC, which are already realized in $2 \to 2$ vector boson scattering by the dimension-eight operators in eq. (4.12), see table 4.1 for their contributions to different VBS processes. However, such scatterings are not directly observable at LHC, in the sense that one can not directly measure VV final states, but only the decay products of the vector bosons. Furthermore, at the LHC the measurement of the decay products is complicated, since hadronic decay products are indistinguishable from the large QCD background. Therefore, only leptonic decay products are taken into consideration. The situation for the initial state at the LHC is similar as vector bosons are produced by emissions off quark currents. Thus, we can not directly observe $VV \to VV$ scattering. These processes are embedded in pp-processes, where the production of the vector bosons comes from quark currents $J^{\mu}_{\mathrm{p}\to jV}$, and the final state consists of leptonic decay products, written as decay currents $J^{\mu}_{V\to f\bar{f}}$. Furthermore, these VBS processes are strongly suppressed by the PDF dependence of the quark currents in comparison to gluon-induced processes, since gluon-PDFs give the dominant contribution from proton-proton scattering at high energies.

A wide variety of Feynman diagrams [24] contribute to the transition amplitude for the embedded vector-boson scattering $\mathrm{pp} \to VVjj$, see fig. 4.2. In particular, contributions come from vector-boson emission from the quark currents, trilinear or quartic couplings,

	$\mathcal{L}_{S,i}$	$\mathcal{L}_{M,i}$		$\mathcal{L}_{T,i}$		
$i \in$	$\{0,1,2\}$	$\{0,1,7\}$	$\{2,3,4,5\}$	$\{0,1,2\}$	$\{5,6,7\}$	$\{8,9\}$
WWWW	✓		✓	✓		
WWZZ	✓	✓	✓	✓	✓	
ZZZZ	✓	✓	✓	✓	✓	✓
WWZγ		✓	✓	✓	✓	
WW$\gamma\gamma$		✓	✓	✓	✓	
ZZZγ		✓	✓	✓	✓	✓
ZZ$\gamma\gamma$		✓	✓	✓	✓	✓
Z$\gamma\gamma\gamma$				✓	✓	✓
$\gamma\gamma\gamma\gamma$				✓	✓	✓

Table 4.1: Contributions of the different dimension-eight operators of the Eboli basis to the aQGC vertices, taken from [129].

and QCD-induced backgrounds.

For our consideration, we limit ourselves to embedding the aQGC from the dimension-eight operators as quasi-elastic $VV \to VV$ scattering process into the quark currents, but neglect the leptonic decay of the vector bosons. This assumption is justified since we are only interested in the general high-energy behavior and (possible) violations of the unitarity of the scattering matrix by the dimension-eight operators. In particular, we do not want to concern ourselves with the prediction of fiducial cross sections, which we postpone to a future work.

Proceeding as in [57], we restrict ourselves to on-shell $2 \to 2$ vector boson scattering, whose kinematics can be fully described with the Mandelstam variables s, t and u (see appendix B.3 for their definition). The application of symmetries, in our case the (approximate) custodial symmetry $\mathrm{SU}(2)_\mathrm{C}$ (see section 4.1), and crossing relations allows us to cast the different amplitudes of the vector-boson scatterings in a compact analytical form, as proposed in [134], and will be given in section 4.3.

In this form, we obtain a single master amplitude (from a single process), which is related by the isospin symmetry and crossing relations to the amplitudes of all other processes. We can then decompose all amplitudes (using the master amplitude) into partial-wave amplitudes by partial-wave decomposition. This allows us to diagonalize the scattering matrix in a bottom-up approach, which is a prerequisite for the application of the T-matrix unitarization [17, 130] prescription (which we introduce in section 4.2). In this form, we can unitarize the respective eigenamplitudes of the EFT for aQGC with an analytical method [17, 57, 130, 137, 138]. Compared to previous studies [17, 19, 130], which investigated only scattering of longitudinally polarized particles, where the decomposition into partial waves is done for a spinless transition (e.g. using Legendre polynomials instead of Wigner D-functions), we examine all 81 helicity transitions from initial to final state.[6]

After determining the spin eigenamplitudes, we neglect helicity-dependent terms of

[6] 9 helicity states for the initial state $|VV\rangle$ and 9 helicity states for the final state $\langle VV|$, hence, 81 transitions.

$\mathcal{O}(s^4)$, which provide only subleading contributions and we thus neglect them in a first approximation. The unitarized spin eigenamplitudes are then transformed back to the isospin basis and the unitarity corrections are applied to the Feynman rules for the VV $\to$ VV processes as s-dependent form factors. These form factors imply on-shell evaluation. For the later implementation, we neglect this prerequisite and evaluate the s-dependent unitarization form factor with off-shell momenta.

We have to match the space-like momenta of the incoming vector bosons to s. We do this by setting $s = (q_1 + q_2)^2$, where q_1, q_2 are the space-like momenta of the two incoming vector bosons. The effect is negligible, as long as the mismatch between off-shell and on-shell amounts to subleading terms of order m_{W}^2/s and q_i^2/s. For VBS, the kinematics $|q_i^2| \sim m_{\mathrm{W}}^2$ dominate the cross section, but phase-space regions also contribute where terms proportional to q_i^2/s become leading. However, we limit ourselves to observables inclusive in q_i^2, where these terms are (mostly) of subleading order. A detailed description for handling off-shell $VV \to VV$ subprocesses for any momenta, as well as a complete helicity treatment (without neglecting higher terms in the partial-wave decomposition) is given in [139] for the Monte Carlo generator VBFNLO with the specifically developed T_u-matrix unitarization method.

Custodial symmetry dictates that we neglect contributions of external and internal photons, since these explicitly break the custodial symmetry. Furthermore, we ignore the Coulomb singularity [140] for charged particles, i.e. charged-W, since the forward region, where the Coulomb singularity occurs, is cut out in an experimental analysis [122], and the effect for off-shell particles is rather small compared to on-shell particles [141].

4.2 Unitarization

The predictive power of the EFT introduced in section 4.1 breaks down for high energies larger than the masses of the electroweak gauge bosons, which was demonstrated in [17, 19, 57, 130]. Break-down is meant in the sense that in the asymptotic limit of high center-of-mass energies the EFT predicts more events than are allowed by the unitarity of the S-matrix, and thus violates conservation of probability. In particular, we know that the SM provides an intricate cancellation of amplitudes with gauge boson interactions and associated Higgs exchange for its high-energy behavior [117, 118]. Therefore, any high-energy completion of the SM consistent with the basic assumptions of the EFT approach should not predict unitarity-violating event distributions [17].

Various methods have been proposed: cut-off functions, energy-dependent form-factors applied to couplings or unitarization prescriptions acting on the scattering matrix $\boldsymbol{\mathcal{S}}$ itself, which all suppress the unphysical high-energy behavior of the effective theory. Since we want to model the high-energy behavior without a dependence on a cut-off or form-factor in a bottom-up approach, we make use of the unitarization prescription, in our case, the T-matrix unitarization proposed by (Kilian, Reuter, Ohl and Sekulla in) [17].

In the following, we derive the most important properties from the unitarity of the scattering matrix, and outline the derivation of the K-matrix unitarization scheme [142, 143] and its drawbacks, along the lines of [144]. Furthermore, we introduce the T-matrix unitarization from [17, 130] and briefly discuss the requirement to diagonalize the scattering matrix $\boldsymbol{\mathcal{S}}$ in order to calculate unitarized amplitudes of the EFT.

The transition amplitude for an initial state $|i\rangle$ becoming a final state $\langle f|$ is given by the $\mathcal{S}$-matrix operator

$$\mathcal{S}_{fi} = \langle f|\mathcal{S}|i\rangle\,, \tag{4.52}$$

which under time reversal becomes

$$\mathcal{S}^{\dagger}_{if} = \langle i|\mathcal{S}^{\dagger}|f\rangle\,. \tag{4.53}$$

Splitting off the non-interacting part of the $\mathcal{S}$-matrix and denoting the interacting part as $\mathcal{T}$-matrix, the scattering matrix is then written as

$$\mathcal{S} = \mathbf{I} + \mathrm{i}\mathcal{T}, \tag{4.54}$$

where $\mathbf{I}$ denotes the identity operator. The unitarity of the $\mathcal{S}$-matrix incorporating the conservation of probability

$$\mathcal{S}^{\dagger}\mathcal{S} = \mathcal{S}\mathcal{S}^{\dagger} = \mathbf{I}, \tag{4.55}$$

can then be used to relate the imaginary part of the $\mathcal{T}$-matrix to the total cross section of a related $2 \to 2$ forward scattering process by the virtue of the optical theorem, see [20, pp. 453–455] for a detailed derivation.

From the unitarity condition, we can also derive an upper bound for the size of an arbitrary scattering amplitude. For this consideration, we limit ourselves, for the sake of simplicity, to $2 \to 2$ scattering processes of the form $ab \to cd$. The differential cross section for such a scattering is given in correspondence to eq. (1.5) by

$$\frac{\mathrm{d}\sigma}{\mathrm{d}\Omega} = \frac{1}{(8\pi)^2 s}\frac{q_f}{q_i}\left|\mathcal{M}_{fi}\right|^2 = \left|f_{fi}(\Omega)\right|^2, \tag{4.56}$$

with the solid angle $\Omega = (\theta, \phi)$, the center-of-mass energy $\sqrt{s}$ and the invariant squared matrix-element $\mathcal{M}_{fi}$ where f and i denote the final and initial states, and the respective momenta q_f and q_i. The scattering amplitude $f_{fi}(\Omega)$ can then be expressed in terms of partial-wave amplitudes with Wigner D-functions [21, 145]

$$f_{fi}(\Omega) = \frac{1}{q_i}\sum_J (2J+1)\mathcal{T}^J_{fi}\, d^J_{\lambda,\lambda'}(\phi, \theta, 0), \tag{4.57}$$

where $\lambda = (\lambda_a - \lambda_b)$ and $\lambda' = (\lambda_c - \lambda_d)$ in terms of the helicities of the involved particles in the scattering process. Integrating over the full solid angle, the total cross section for the partial-wave J is then given by

$$\sigma^J_{fi} = \frac{4\pi}{q_i^2}(2J+1)\left|\mathcal{T}^J_{fi}(s)\right|^2. \tag{4.58}$$

We can identify the initial and final states with total spin J and the z-component M of J (in a coordinate system fixed in the overall center-of-mass frame). The states are then defined as

$$|i\rangle = |\{a, b\}; J, M, \lambda_a, \lambda_b\rangle\,, \tag{4.59}$$

$$|f\rangle = |\{c, d\}; J, M, \lambda_c, \lambda_d\rangle\,, \tag{4.60}$$

where $\{a, b\}$ and $\{c, d\}$ denote additional quantum numbers to fully specify the initial and final states, respectively. The normalization is then $\langle f|i\rangle = \delta_{fi}$ due to overall angular momentum conservation. Assuming that we can then diagonalize the $\mathcal{T}$-matrix with a unitary transformation U, e.g. into partial waves.[7] We find

$$\mathcal{S}^\dagger \mathcal{S} = \left(\mathbf{I} + \frac{\mathrm{i}}{2} U \mathcal{T}' U^\dagger\right)^\dagger \left(\mathbf{I} + \frac{\mathrm{i}}{2} U \mathcal{T}' U^\dagger\right) = \mathbf{I}, \tag{4.61}$$

where $\mathcal{T} = U\mathcal{T}'U^\dagger$. Equation (4.61) can be reformulated with the complex eigenvalues[8] $(\mathcal{T})_{ij} = 2t_{ij}\delta_{ij} \equiv 2t_j$ of the $\mathcal{T}$-matrix resulting in the Argand circle condition

$$\left|t_j - \frac{\mathrm{i}}{2}\right| = \frac{1}{2}, \tag{4.62}$$

which has a simple geometrical interpretation (see fig. 4.3). The complex eigenvalues of the transition operator $\mathcal{T}$ will lie on a circle with radius $\frac{1}{2}$ around the center $\frac{\mathrm{i}}{2}$ in the complex plane. Conversely, we can read off of eq. (4.62) an upper bound for tree-level amplitudes, e.g. for the partial-wave amplitudes $a^J_{\lambda,\lambda'}$ of spin J and helicity differences of in- and outgoing particles λ, λ' in eq. (4.58)

$$\left|\mathrm{Re}\, a^J_{\lambda,\lambda'}\right| \leq \frac{1}{2}. \tag{4.63}$$

We can use the unitary condition to introduce a unitarization prescription, we get from eq. (4.55) and eq. (4.54)

$$\mathcal{S}^\dagger \mathcal{S} = \mathbf{I} + \mathrm{i}\left(\mathcal{T} - \mathcal{T}^\dagger\right) + \mathcal{T}^\dagger \mathcal{T} = \mathbf{I}, \tag{4.64}$$

where from unitarity of the $\mathcal{S}$-matrix also follows that the $\mathcal{T}$-matrix is normal, thus, $\mathcal{T}^\dagger\mathcal{T} = \mathcal{T}\mathcal{T}^\dagger$. Assuming that the inverse $\mathcal{T}^{-1}$ exists,[9] multiplying eq. (4.64) from left with $\mathcal{T}^{-1}$ and from right with $\left(\mathcal{T}^\dagger\right)^{-1}$ we find

$$\begin{aligned} &(\mathcal{T}^{-1})^\dagger - \mathcal{T}^{-1} = \mathrm{i}\mathcal{T} \\ \Leftrightarrow &(\mathcal{T}^{-1} + \frac{\mathrm{i}}{2}\mathbf{I})^\dagger = \mathcal{T}^{-1} + \frac{\mathrm{i}}{2}\mathbf{I}. \end{aligned} \tag{4.65}$$

K-matrix scheme

Using eq. (4.65) we can then define the inverse of the K-matrix operator by,

$$K^{-1} \equiv \mathcal{T}^{-1} + \frac{\mathrm{i}}{2}\mathbf{I}, \tag{4.66}$$

and from eq. (4.65), we see that $K = K^\dagger$ holds. Furthermore, from invariance of time reversal of the $\mathcal{S}$ and $\mathcal{T}$-matrix operators, it follows that the K-matrix operator is symmetric

[7] In general, the overall approach is not limited to a $2 \to 2$ scattering process.

[8] We introduced a factor of 2 for convenience.

[9] In general, the assumption is not trivial. But we refrain from any profound reasoning, since we use the assumption only for the introduction of the K-matrix unitarization in a specific case where the assumption is valid, see later.

and furthermore, we may choose the K-matrix to be real. Multiplying eq. (4.66) by K from left and $\mathcal{T}$ from right (and vice versa), we find

$$\mathcal{T} = K + \frac{\mathrm{i}}{2} K \cdot \mathcal{T} = K + \frac{\mathrm{i}}{2} \mathcal{T} \cdot K, \tag{4.67}$$

from which directly follows that K and $\mathcal{T}$ commutate, $[K, \mathcal{T}] = 0$. Solving for $\mathcal{T}$, we arrive at

$$\mathcal{T} = \frac{K}{\mathbf{I} - \frac{\mathrm{i}}{2} K}, \tag{4.68}$$

where $\mathbf{I}/(\mathbf{I} - \frac{\mathrm{i}}{2}K) \equiv (\mathbf{I} - \frac{\mathrm{i}}{2}K)^{-1}$, and the relation to the $\mathcal{S}$-matrix operator is given by

$$\mathcal{S} = \frac{\mathbf{I} + \frac{\mathrm{i}}{2} K}{\mathbf{I} - \frac{\mathrm{i}}{2} K}. \tag{4.69}$$

The K-matrix in eq. (4.69) was introduced by Heitler [137] and Schwinger [138] as Cayley transform of the complete unitary scattering operator $\mathcal{S}$. For a perturbatively calculable theory, the K-matrix can be determined perturbatively as long as $\mathcal{T} - 2\mathrm{i}$ is non-singular [17]. From eq. (4.68) we see that K and $\mathcal{T}$ are identical at lowest order of the perturbation series.

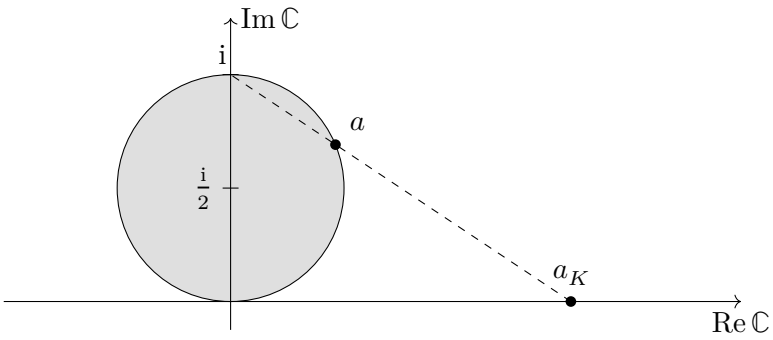


Figure 4.3: Stereographic projection of an unitary scattering amplitude on the Argand circle to the real-valued K-matrix amplitude a_K.

If we assume that the $\mathcal{T}$-matrix is diagonalizable, i.e. with the eigenvalues $t = 2a_J$ for the special case of the partial wave decomposition in eq. (4.58), then the $n \times n$-matrix $K \in \mathbb{R}^{n \times n}$ is given by,

$$K_{ij} = \sum_n \mathcal{T}_{in} (\mathbf{I} + \frac{\mathrm{i}}{2} \mathcal{T})^{-1}_{nj} = \sum_n 2a_i \left(1 + \mathrm{i}a_j\right)^{-1} \delta_{in} \delta_{nj} = 2 \frac{a_i}{1 + \mathrm{i}a_j} \delta_{ij}. \tag{4.70}$$

The corresponding eigenvalue a_K of the K-matrix can then be related to an amplitude a "as a sterographic projection from the Argand circle onto the real axis" [130] (see fig. 4.3),

$$a_K = \frac{a}{1 + \mathrm{i}a}. \tag{4.71}$$

In general, the diagonalization of the $\mathcal{S}$-matrix may seem infeasible or even impossible. In that case, it was proposed to revert the logic to compute the K-matrix from the scattering

matrix by [142, 143, 146–148]. Here, the K-matrix is interpreted as an incomplete, but calculable approximation of the true scattering matrix $\boldsymbol{S}$. Thus, the $\boldsymbol{\mathcal{T}}$-matrix can be calculated as a non-perturbative completion of this approximation from a diagonalized K-matrix by inverting eq. (4.71),

$$a = \frac{a_K}{1 - \mathrm{i}a_K}. \tag{4.72}$$

For the case of a non-diagonalizable K-matrix, we refer to [17, 130].

T-matrix scheme

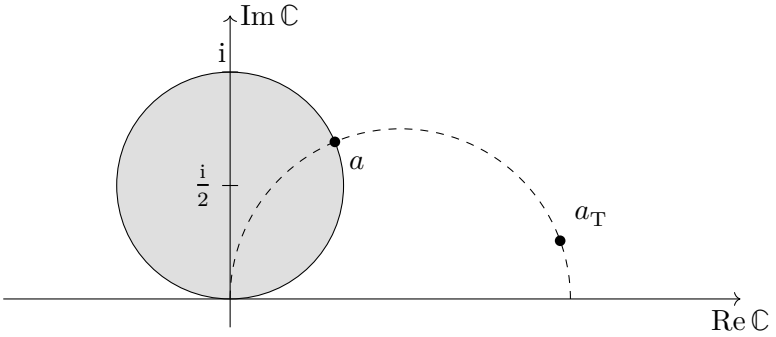


Figure 4.4: Thales projection of a unitary scattering amplitude on the Argand circle to the complex-valued, (possibly non-perturbative) T-matrix amplitude a_T.

The disadvantage of the K-matrix prescription is that the K-matrix implies the existence of a perturbative expansion of the underlying theory. Furthermore, it must be calculated as an intermediate step and the K-matrix itself does not imply any further (physical) use. This intermediate step can become complicated and cumbersome for some processes. Therefore, direct T-matrix unitarization was introduced in [17, 130]

$$\boldsymbol{\mathcal{T}} = \frac{1}{\mathrm{Re}\,\boldsymbol{\mathcal{T}}_0^{-1} - \frac{\mathrm{i}}{2}\mathbf{I}}, \tag{4.73}$$

with a (potentially) unitary-violating $\boldsymbol{\mathcal{T}}_0$-matrix and the unitarized $\boldsymbol{\mathcal{T}}$-matrix (see fig. 4.4). If we let a_J be the eigenamplitudes of the partial-wave decomposition of an arbitrary $\boldsymbol{\mathcal{T}}_0$-matrix, e.g. from section 4.1, then the unitarized amplitude $a_{\mathrm{T},j}$ in the T-matrix unitarization is given by

$$a_{\mathrm{T},J} = \frac{1}{\mathrm{Re}\,\frac{1}{a_J} - \mathrm{i}}. \tag{4.74}$$

After diagonalizing the respective $\boldsymbol{\mathcal{T}}$-matrix, we can then directly apply the unitarization prescription onto the eigenamplitudes.

4.3 Isospin-Spin Amplitudes

We outline the approach to the unitarization of $2 \to 2$ processes as presented in [17, 57, 130]. We revisit the most important steps in order to compute the isospin-spin eigenbasis of the quartic gauge couplings as well as their unitarization. We show schematically the back transformation into isospin amplitudes for the actual implementation in the Monte Carlo event generator WHIZARD.[10]

In the approximate custodial symmetry $\mathrm{SU}(2)_\mathrm{C}$, the three vector bosons $\mathrm{W}^+, \mathrm{W}^-$ and Z form a (mass-)degenerate triplet state.[11] From this, we can build $\mathrm{SU}(2) \times \mathrm{SU}(2)$ states $|VV\rangle$. The latter can be transformed to the isospin-eigenbasis through application of Clebsch–Gordon decomposition (see appendix B.4 for the details). In addition, we can fully describe the kinematics of a $2 \to 2$ scattering with the Mandelstam variables s, t and u. Along with crossing relations (see appendix B.3) and the former isospin algebra, we can express all quasi-elastic scattering amplitudes, where $\mathrm{W}^+, \mathrm{W}^-$ and Z are involved, by a single master amplitude, analogous to [134].

For the transverse interactions with the dimension-eight operators $\boldsymbol{\mathcal{L}}_{T,0/1/2}$ and for the mixed dimension-eight operators $\boldsymbol{\mathcal{L}}_{M,0/1/7}$ we define the master amplitude

$$\begin{aligned}
A(s,t,u;\boldsymbol{\lambda}) \equiv \mathcal{A}(W^+_{\lambda_1} W^-_{\lambda_2} \to Z_{\lambda_3} Z_{\lambda_4}) = \\
&- 2g^4 \left(F_{T,0} + \frac{1}{4} F_{T,2}\right) \delta_{\lambda_1,\lambda_2} \delta_{\lambda_3,\lambda_4} s^2 \\
&- g^4 \left(F_{T,1} + \frac{1}{2} F_{T,2}\right) \left(\delta_{\lambda_1,-\lambda_3} \delta_{\lambda_2,-\lambda_4} t^2 + \delta_{\lambda_1,-\lambda_4} \delta_{\lambda_2,-\lambda_3} u^2\right) \\
&+ \frac{1}{2} g^4 F_{T,2} \left(\delta_{\lambda_1,\lambda_2} \delta_{\lambda_3,\lambda_4} \delta_{\lambda_1,-\lambda_3}\right) \left(t^2 + u^2\right) \\
&+ \frac{1}{16} g^2 \left(8F_{M,0} - 2F_{M,1} + F_{M,7}\right) s^2 \left(\delta_{\lambda_1,\lambda_2} \delta_{\lambda_3,0} \delta_{\lambda_4,0} - 16 \delta_{\lambda_3,\lambda_4} \delta_{\lambda_1,0} \delta_{\lambda_2,0}\right) \\
&+ \frac{1}{16} g^2 \left(2F_{M,1} + F_{M,7}\right) \left(s^2 - t^2 - u^2\right) \left(\delta_{\lambda_1,-\lambda_2} \delta_{\lambda_3,0} \delta_{\lambda_4,0} - \delta_{\lambda_3,-\lambda_4} \delta_{\lambda_1,0} \delta_{\lambda_2,0}\right) \\
&+ \frac{1}{16} g^2 F_{M,7} \left(\left(\delta_{\lambda_1,-\lambda_3} \delta_{\lambda_2,0} \delta_{\lambda_4,0} - \delta_{\lambda_2,-\lambda_4} \delta_{\lambda_1,0} \delta_{\lambda_3,0}\right) \left(s^2 - u^2\right)\right. \\
&+ \left.\left(\delta_{\lambda_2,-\lambda_3} \delta_{\lambda_1,0} \delta_{\lambda_4,0} - \delta_{\lambda_1,-\lambda_4} \delta_{\lambda_2,0} \delta_{\lambda_3,0}\right) \left(s^2 - t^2\right)\right).
\end{aligned} \tag{4.75}$$

where $\boldsymbol{\lambda} \equiv (\lambda_1, \lambda_2, \lambda_3, \lambda_4)$ for ease of the notation. Using the custodial $\mathrm{SU}(2)_\mathrm{C}$ isospin-decomposition rules given in appendix B.4, the scattering amplitudes for the mixed and transverse operators are given by

$$\mathcal{A}(W^+_{\lambda_1} W^+_{\lambda_2} \to W^+_{\lambda_3} W^+_{\lambda_4}) = \quad \mathcal{A}_2(s,t,u;\boldsymbol{\lambda}), \tag{4.76}$$

$$\mathcal{A}(W^+_{\lambda_1} W^-_{\lambda_2} \to W^+_{\lambda_3} W^-_{\lambda_4}) = \frac{1}{3}\mathcal{A}_0(s,t,u;\boldsymbol{\lambda}) + \frac{1}{2}\mathcal{A}_1(s,t,u;\boldsymbol{\lambda}) + \frac{1}{6}\mathcal{A}_2(s,t,u;\boldsymbol{\lambda}), \tag{4.77}$$

$$\mathcal{A}(W^+_{\lambda_1} W^-_{\lambda_2} \to Z_{\lambda_3} Z_{\lambda_4}) = \frac{1}{3}\mathcal{A}_0(s,t,u;\boldsymbol{\lambda}) - \frac{1}{3}\mathcal{A}_2(s,t,u;\boldsymbol{\lambda}), \tag{4.78}$$

[10] We skip the details of the implementation due to their purely technical nature, and refer the interested reader to the previous studies [57, 130].

[11] Since we restrict ourself to the high-energy range, we neglect the masses of W, Z and H (where applicable).

$$\mathcal{A}(W^+_{\lambda_1}Z_{\lambda_2}\to W^+_{\lambda_3}Z_{\lambda_4}) = \frac{1}{2}\mathcal{A}_1(s,t,u;\boldsymbol{\lambda}) + \frac{1}{2}\mathcal{A}_2(s,t,u;\boldsymbol{\lambda}), \tag{4.79}$$

$$\mathcal{A}(Z_{\lambda_1}Z_{\lambda_2}\to Z_{\lambda_3}Z_{\lambda_4}) = \frac{1}{3}\mathcal{A}_0(s,t,u;\boldsymbol{\lambda}) + \frac{2}{3}\mathcal{A}_2(s,t,u;\boldsymbol{\lambda}). \tag{4.80}$$

The decomposition into isospin-spin eigenamplitude by an expansion of isospin eigenamplitudess into Wigner D-functions [145] $d^J_{\lambda,\lambda'}(\theta)$ with $\lambda = \lambda_1 - \lambda_2$ and $\lambda^{'} = \lambda_3 - \lambda_4$ yields

$$\mathcal{A}_{IJ}(s;\boldsymbol{\lambda}) = \int_{-s}^{0}\frac{\mathrm{d}t}{s}\mathcal{A}_I \cdot d^J_{\lambda,\lambda'}\left(\arccos\left(1+2\frac{t}{s}\right)\right), \tag{4.81}$$

where we matched the angle θ with the Mandelstam variable t, $\cos\theta = \frac{2t}{s}+1$. The full expressions for the Wigner D-functions are given in appendix B.5.

The isospin eigenamplitudes $\mathcal{A}_I$ are given by

$$\begin{aligned}\mathcal{A}_0(s,t,u;\boldsymbol{\lambda}) =& 3\mathcal{A}(s,t,u;\lambda_1,\lambda_2,\lambda_3,\lambda_4) + \mathcal{A}(t,s,u;-\lambda_4,\lambda_2,\lambda_3,-\lambda_1)\\ &+ \mathcal{A}(u,t,s;\lambda_1,-\lambda_4,\lambda_3,-\lambda_2),\end{aligned} \tag{4.82}$$

$$\mathcal{A}_1(s,t,u;\boldsymbol{\lambda}) = \mathcal{A}(t,s,u;-\lambda_4,\lambda_2,\lambda_3,-\lambda_1) + \mathcal{A}(u,t,s;\lambda_1,-\lambda_4,\lambda_3,-\lambda_2), \tag{4.83}$$

$$\mathcal{A}_2(s,t,u;\boldsymbol{\lambda}) = \mathcal{A}(t,s,u;-\lambda_4,\lambda_2,\lambda_3,-\lambda_1) + \mathcal{A}(u,t,s;\lambda_1,-\lambda_4,\lambda_3,-\lambda_2). \tag{4.84}$$

For the purpose of constructing a minimal unitary projection, it is sufficient to determine a set of master amplitudes which capture the leading term proportional to s^2 for each spin-isospin channel, uniformly for all individual helicity combinations. The implied over-compensation of some helicity channels that are subleading at high energy, is within the scheme dependence that is inherent in the unitary projection. The simplified helicity-independent expressions are

$$\mathcal{A}_{00}(s) = -\frac{3}{2}g^4\left[4F_{T,0}-2F_{T,1}+F_{T,2}\right]s^2 + \frac{3}{16}g^2\left[8F_{M,0}+2F_{M,1}+F_{M,7}\right]s^2, \tag{4.85}$$

$$\mathcal{A}_{01}(s) = -\frac{1}{32}g^2\left[4F_{M,0}+F_{M,1}-3F_{M,7}\right]s^2, \tag{4.86}$$

$$\mathcal{A}_{02}(s) = -\frac{1}{10}g^4\left[4F_{T,0}-2F_{T,1}+F_{T,2}\right]s^2 + \frac{1}{160}g^2\left[4F_{M,0}+F_{M,1}+F_{M,7}\right]s^2, \tag{4.87}$$

$$\mathcal{A}_{10}(s) = 0, \tag{4.88}$$

$$\mathcal{A}_{11}(s) = \frac{1}{6}g^4F_{T,2}s^2 - \frac{1}{32}g^2\left[4F_{M,0}+F_{M,1}-3F_{M,7}\right]s^2, \tag{4.89}$$

$$\mathcal{A}_{12}(s) = \frac{1}{5}g^4\left[-2F_{T,0}+F_{M,1}-3F_{M,7}\right]s^2 + \frac{1}{160}g^2\left[4F_MM,0+F_{M,1}+F_{M,7}\right]s^2, \tag{4.90}$$

$$\mathcal{A}_{20}(s) = 0, \tag{4.91}$$

$$\mathcal{A}_{21}(s) = -\frac{1}{32}g^2\left[4F_{M,0}+F_{M,1}-3F_{M,7}\right]s^2, \tag{4.92}$$

$$\mathcal{A}_{22}(s) = -\frac{1}{10}g^4\left[4F_{T,0}-2F_{T,1}+F_{T,2}\right]s^2 + \frac{1}{160}g^2\left[4F_{M,0}+F_{M,1}+F_{M,7}\right]s^2. \tag{4.93}$$

We list the complete set of master amplitudes with their dependence on the operator coefficients in table 4.2 for the transverse operators and in table 4.3 for the mixed operators.

				j						
i		0			1			2		$\boldsymbol{\lambda}$
0	-6	-2	$-\frac{5}{2}$	0	0	0	0	0	0	$(+,+,+,+)$
	0	0	0	0	0	0	$-\frac{2}{5}$	$-\frac{4}{5}$	$-\frac{1}{2}$	$(+,-,+,-)$
	0	0	0	0	0	0	$-\frac{2}{5}$	$-\frac{4}{5}$	$-\frac{1}{2}$	$(+,-,-,+)$
	$-\frac{22}{3}$	$-\frac{14}{3}$	$-\frac{11}{6}$	0	0	0	$-\frac{2}{15}$	$-\frac{4}{15}$	$-\frac{1}{30}$	$(+,+,-,-)$
1	0	0	0	0	0	0	0	0	0	$(+,+,+,+)$
	0	0	0	0	0	0	$\frac{2}{5}$	$-\frac{1}{5}$	0	$(+,-,+,-)$
	0	0	0	0	0	0	$-\frac{2}{5}$	$\frac{1}{5}$	0	$(+,-,-,+)$
	0	0	0	$\frac{2}{3}$	$-\frac{1}{3}$	$\frac{1}{6}$	0	0	0	$(+,+,-,-)$
2	0	-2	-1	0	0	0	0	0	0	$(+,+,+,+)$
	0	0	0	0	0	0	$-\frac{2}{5}$	$-\frac{1}{5}$	$-\frac{1}{5}$	$(+,-,+,-)$
	0	0	0	0	0	0	$-\frac{2}{5}$	$-\frac{1}{5}$	$-\frac{1}{5}$	$(+,-,-,+)$
	$-\frac{4}{3}$	$-\frac{8}{3}$	$-\frac{1}{3}$	0	0	0	$-\frac{2}{15}$	$-\frac{1}{15}$	$-\frac{1}{30}$	$(+,+,-,-)$
	c_0	c_1	c_2	c_0	c_1	c_2	c_0	c_1	c_2	

Table 4.2: Coefficients of the isospin-spin amplitudes calculated with eq. (4.81) for the transversal operators $\mathcal{L}_{T,i}$ depending on the helicity of the incoming and outgoing particles. The isospin-spin amplitudes are given by $\mathcal{A}_{ij}(s;\boldsymbol{\lambda}) = (c_0 F_{T_0} + c_1 F_{T_1} + c_2 F_{T_2}) g^4 s^2$.

Unitarization Corrections

The T-matrix projection (as introduce in section 4.2) is applied to the simplified helicity-independent eigenamplitudes where only one coupling (mixed or transverse) is active. With the latter, one arrives at a closed form expression. In contrast, when using an arbitrary number of non-zero couplings or using explicitly helicity-dependent amplitudes, one has to resort to a numerical evaluation of the T-matrix projection [139]. For the spin-isospin eigenamplitudes $\mathcal{A}_{IJ}$, the unitarization projection is given as

$$\hat{\mathcal{A}}_{IJ} = \frac{1}{Re\frac{1}{\mathcal{A}_{IJ}} - \frac{\mathrm{i}}{32\pi}}. \tag{4.94}$$

The unitarization correction is split off of the naive EFT amplitude $\mathcal{A}_{IJ}$

$$\Delta\mathcal{A}_{IJ} = \hat{\mathcal{A}}_{IJ} - \mathcal{A}_{IJ}, \tag{4.95}$$

reverting the corrections of the isospin-spin amplitudes back to the isospin basis[12]

$$\Delta\mathcal{A}_I(s,t,u;\boldsymbol{\lambda}) = \sum_{J=0}^{2}(2J+1)\Delta\mathcal{A}_{IJ}\, d^J_{\lambda,\lambda'}(\cos\theta)\Big|_{\cos\theta=\frac{2t}{s}+1}, \tag{4.96}$$

[12]We have placed the factor of 32π into the unitarization projection in eq. (4.94).

	j									
i	0			1			2			$\boldsymbol{\lambda}$
0	$\frac{3}{2}$	$-\frac{3}{8}$	$\frac{3}{16}$	0	0	0	0	0	0	$(+,+,0,0)$
	0	0	0	0	0	0	0	$\frac{1}{20}\sqrt{\frac{3}{2}}$	$\frac{1}{40}\sqrt{\frac{3}{2}}$	$(+,-,0,0)$
	0	0	0	$-\frac{1}{8}$	$\frac{1}{32}$	$\frac{7}{192}$	$\frac{1}{40}$	$-\frac{1}{160}$	$\frac{3}{320}$	$(+,0,-,0)$
	0	0	0	$\frac{1}{8}$	$-\frac{1}{32}$	$-\frac{7}{192}$	$\frac{1}{40}$	$-\frac{1}{160}$	$\frac{3}{320}$	$(+,0,0,-)$
	0	0	0	0	$\frac{1}{12}$	$\frac{1}{24}$	0	0	0	$(+,0,+,0)$
	0	0	0	0	$-\frac{1}{12}$	$-\frac{1}{24}$	0	0	0	$(+,0,0,+)$
1	0	0	0	0	0	$\frac{1}{24}$	0	0	0	$(+,+,0,0)$
	0	0	0	0	0	0	0	0	0	$(+,-,0,0)$
	0	0	0	$-\frac{1}{8}$	$\frac{1}{32}$	$\frac{1}{96}$	$\frac{1}{40}$	$-\frac{1}{160}$	$\frac{1}{160}$	$(+,0,-,0)$
	0	0	0	$-\frac{1}{8}$	$\frac{1}{32}$	$\frac{1}{96}$	$-\frac{1}{40}$	$\frac{1}{160}$	$-\frac{1}{160}$	$(+,0,0,-)$
	0	0	0	0	$\frac{1}{12}$	$\frac{1}{24}$	0	0	0	$(+,0,+,0)$
	0	0	0	0	$\frac{1}{12}$	$\frac{1}{24}$	0	0	0	$(+,0,0,+)$
2	0	0	0	0	0	0	0	0	0	$(+,+,0,0)$
	0	0	0	0	0	0	0	0	0	$(+,-,0,0)$
	0	0	0	$-\frac{1}{8}$	$\frac{1}{32}$	$-\frac{1}{24}$	$\frac{1}{40}$	$-\frac{1}{160}$	0	$(+,0,-,0)$
	0	0	0	$\frac{1}{8}$	$-\frac{1}{32}$	$\frac{1}{24}$	$\frac{1}{40}$	$-\frac{1}{160}$	0	$(+,0,0,-)$
	0	0	0	0	$\frac{1}{12}$	$\frac{1}{24}$	0	0	0	$(+,0,+,0)$
	0	0	0	0	$-\frac{1}{12}$	$-\frac{1}{24}$	0	0	0	$(+,0,0,+)$
	c_0	c_1	c_2	c_0	c_1	c_2	c_0	c_1	c_2	

Table 4.3: Coefficients of the isospin-spin amplitudes calculated with eq. (4.81) for the mixed operators $\boldsymbol{\mathcal{L}}_{M,i}$ depending on the helicity of the incoming and outgoing particles. The isospin spin amplitudes are given by $\boldsymbol{\mathcal{A}}_{ij}(s;\boldsymbol{\lambda}) = (c_0 F_{M_0} + c_1 F_{M_1} + c_2 F_{M_7}) g^2 s^2$.

one arrives with counter-terms for the different scattering channels. Using $\cos\theta = \frac{2t}{s} + 1$ and $t = -(u+s)$, for the massless case, we can express the isospin eigenamplitudes in the Mandelstam variables s, t and u. We arrive for the mixed operators with $F_{M,i} \neq 0, F_{T,i} = 0$

at

$$
\begin{aligned}
&\Delta\mathcal{A}_I(s,t,u;\boldsymbol{\lambda}) = \\
&\quad\left(\Delta\mathcal{A}_{I0}(s) - \frac{5}{2}\Delta\mathcal{A}_{I1}(s)\frac{t-u}{s} + \frac{15}{2}\Delta\mathcal{A}_{I2}(s)\left(\frac{t-u}{s}\right)^2\right)\delta_{\lambda_1,+}\delta_{\lambda_2,+}\delta_{\lambda_3,0}\delta_{\lambda_4,0} \\
&+\left(\frac{5\sqrt{6}}{4}\Delta\mathcal{A}_{I2}(s) - \frac{5\sqrt{6}}{4}\Delta\mathcal{A}_{I2}(s)\left(\frac{t-u}{s}\right)^2\right)\delta_{\lambda_1,+}\delta_{\lambda_2,-}\delta_{\lambda_3,0}\delta_{\lambda_4,0} \\
&+\left(\Delta\mathcal{A}_{I1}(s) - \Delta\mathcal{A}_{I1}(s)\frac{t-u}{s} - 5\Delta\mathcal{A}_{I2}(s)\left(\frac{t-u}{s}\right)^2\right)\delta_{\lambda_1,+}\delta_{\lambda_2,0}\delta_{\lambda_3,-}\delta_{\lambda_4,0} \\
&+\left(\Delta\mathcal{A}_{I1}(s) + \Delta\mathcal{A}_{I1}(s)\frac{t-u}{s} + 5\Delta\mathcal{A}_{I2}(s)\left(\frac{t-u}{s}\right)^2\right)\delta_{\lambda_1,+}\delta_{\lambda_2,0}\delta_{\lambda_3,0}\delta_{\lambda_4,-} \\
&+\left(\Delta\mathcal{A}_{I1}(s) + \Delta\mathcal{A}_{I1}(s)\frac{t-u}{s} + 5\Delta\mathcal{A}_{I2}(s)\left(\frac{t-u}{s}\right)^2\right)\delta_{\lambda_1,+}\delta_{\lambda_2,0}\delta_{\lambda_3,+}\delta_{\lambda_4,0} \\
&+\left(\Delta\mathcal{A}_{I1}(s) - \Delta\mathcal{A}_{I1}(s)\frac{t-u}{s} - 5\Delta\mathcal{A}_{I2}(s)\left(\frac{t-u}{s}\right)^2\right)\delta_{\lambda_1,+}\delta_{\lambda_2,0}\delta_{\lambda_3,0}\delta_{\lambda_4,+},
\end{aligned}
\tag{4.97}
$$

and analogously, for the transverse operators with $F_{M,i} = 0, F_{T,i} \neq 0$ we obtain

$$
\begin{aligned}
&\Delta\mathcal{A}_I(s,t,u;\boldsymbol{\lambda}) = \\
&\quad\left(\Delta\mathcal{A}_{I0}(s) - \frac{5}{2}\Delta\mathcal{A}_{I2}(s) + 2\Delta\mathcal{A}_{I1}(s)\frac{t-u}{s} + 3\Delta\mathcal{A}_{I1}(s)\left(\frac{t-u}{s}\right)^2\right)\delta_{\lambda_1,+}\delta_{\lambda_2,+}\delta_{\lambda_3,+}\delta_{\lambda_4,+} \\
&+\left(\frac{5}{4}\Delta\mathcal{A}_{I2}(s) + 5\Delta\mathcal{A}_{I2}(s)\frac{t-u}{s} + 5\Delta\mathcal{A}_{I2}(s)\left(\frac{t-u}{s}\right)^2\right)\delta_{\lambda_1,+}\delta_{\lambda_2,-}\delta_{\lambda_3,+}\delta_{\lambda_4,-} \\
&+\left(\frac{5}{4}\Delta\mathcal{A}_{I2}(s) + 5\Delta\mathcal{A}_{I2}(s)\frac{t-u}{s} + 5\Delta\mathcal{A}_{I2}(s)\left(\frac{t-u}{s}\right)^2\right)\delta_{\lambda_1,+}\delta_{\lambda_2,-}\delta_{\lambda_3,-}\delta_{\lambda_4,+} \\
&+\left(\Delta\mathcal{A}_{I0}(s) - \frac{5}{2}\Delta\mathcal{A}_{I2}(s) + 2\Delta\mathcal{A}_{I1}(s)\frac{t-u}{s} + 3\Delta\mathcal{A}_{I1}(s)\left(\frac{t-u}{s}\right)^2\right)\delta_{\lambda_1,+}\delta_{\lambda_2,+}\delta_{\lambda_3,-}\delta_{\lambda_4,-}.
\end{aligned}
\tag{4.98}
$$

Off-shell Implementation

In the following, we briefly outline the procedure, introduced in [57], for constructing the generic off-shell parameterization of VBS corresponding to the appropriate unitarized on-shell isospin amplitude in eqs. (4.97) and (4.98). The Feynman rules have been derived from the Lagrangian in eq. (4.12) with the help of Mathematica [149] and the Feynrules package [150, 151]. We replace the dependence on the constant prefactors, i.e. $F_{T/M,i}$, by form factors which only depend on s. Compared to the Feynman rules for the scalar operators, as implemented in [17, 130], the Feynman rules for the aQGC introduced by the mixed and transverse-only operators, that contain currents $\mathbf{W}_{\mu\nu}\mathbf{W}^{\alpha\beta}$, depend on the momenta entering/leaving the vertex which come from derivatives acting on the gauge fields. Correspondingly new Lorentz structures have been introduced to Omega.[13]

The unitarization corrections for a given $F_{T/M,i}$ parameter are added in such a way that we match Lorentz structures of the naive Feynman rules with the terms s^2, t^2, and u^2, see appendix B.3. The unitarization corrections are then multiplied as form

[13] See sections: Scalar4-Dim8 Couplings, Mixed Scalar2-Vector2-Dim8 Couplings, Transversal Gauge4 Dim8 Couplings and Mixed Gauge4 Dim8 Couplings of the Omega manual [5].

factors to the corresponding parameters and Lorentz structures. This approach explicitly breaks the crossing symmetry, which, however, is already broken as a consequence of the unitarization scheme for $2 \to 2$ amplitudes. For the OMEGA implementation all possible orientations of the time arrow are implemented for a given vertex as alternatives. The actual orientation is then determined by OMEGA during embedding the quasi-elastic scattering in the actual physical process.

The model `SSC2`, which only incorporated the T-matrix unitarized dimension-eight $\mathcal{L}_{S,1/2}$, has been extended, such that the mixed operators $\mathcal{O}_{M,0/1/7}$ and the transverse operators $\mathcal{O}_{T,0/1/2}$ are implemented supporting the naive EFT and adding unitarization corrections as form factor modified Feynman rules.

Including the Higgs Boson

The operators $\mathcal{L}_{M,1/2/7}$ contribute to Higgs interactions with the vector gauge besons, including the amplitudes for the processes $\mathrm{W^+W^-} \to \mathrm{HH}$ and $\mathrm{ZZ} \to \mathrm{HH}$. The Higgs contribution can be obtained from the corresponding Z contribution with the replacement: $\mathrm{Z} \to \mathrm{Z} + \mathrm{i}h$, [130]. The interaction leads to the following amplitude relations

$$\begin{aligned} &\mathcal{A}(\mathrm{W^\pm W^\mp} \to \mathrm{ZZ}) \to \mathcal{A}(\mathrm{W^\pm W^\mp} \to \mathrm{HH}) = \mathcal{A}(\mathrm{ZZ} \to \mathrm{HH}), \\ &\mathcal{A}(\mathrm{W^\pm Z} \to \mathrm{W^\pm Z}) \to \mathcal{A}(\mathrm{W^\pm H} \to \mathrm{W^\pm H}). \end{aligned} \tag{4.99}$$

These amplitudes contain isospin contributions $I = 1$ and $I = 2$, although the Higgs is a scalar field. This can be explained by the fact that the low energy custodial $\mathrm{SU(2)_C}$ symmetry is a subgroup of the larger, high-energy $\mathrm{SU(2)_L} \times \mathrm{SU(2)_R}$ chiral symmetry after EWSB. The Higgs amplitudes should therefore be considered in context of $\mathrm{SU(2)_L} \times \mathrm{SU(2)_R}$ chiral symmetry, such that single contributions by higher isospin quantum numbers do not occur alone.[14]

4.4 Simplified Resonance Models

In this section, we include resonance models which give rise to distinctive features in the energy spectrum. In the previous section, we only considered continuum models where we smoothly interpolate between the low-energy SMEFT and high-energy unitary saturation.

A resonance is the saturation of an elastic channel for a finite energy, visible as a peak in the spectrum, and the amplitude drops off after the peak, and may then rise again for higher energies. For example, such a behavior can be observed for charm mesons, where one finds resonances below the threshold for $\mathrm{c\bar{c}}$ production and a continuum above [152]. In the following section, we use the approach of [19, 57, 130], where models were considered in the context of VBS, and couplings of resonances with longitudinal gauge bosons were studied via the scalar current $J_{\mathrm{H}}^{(4)}$ from section 4.1. We extend the allowed coupling to transverse bosons. For this we take a single scalar which couples to the current $J_{\mathrm{W}}^{(4)}$ from eq. (4.10).

For high energies, the K-matrix and T-matrix unitarization of an amplitude leads to a saturation which can formally be regarded as a resonance at infinity [57, 130].

[14] For further details, [see 130, p. 68].

However, this high energy behavior of the amplitude differs from resonances with finite mass where the highest absolute value is reached at an energy equal to the resonance's mass. If we can not reach these energies experimentally, we will only observe a rise of the amplitude. In this case, the description of the resonance coincides with the description of the dimension-eight operators in eq. (4.12). Therefore, we study exemplary models with single resonances with specific quantum numbers and compare them with the class of continuum models in section 4.1 that model merely the SMEFT expansion into asymptotically strong interactions.

There are several models for the extension of the Higgs sector, which effectively lead to a similar phenomenology. Firstly, we expect that such a resonance couples both to longitudinal and transverse vector boson polarizations. However, BSM models with new physics coupled to the transverse polarization of the electroweak gauge bosons are constrained by data [153]. Other models include extra dimensions [154, 155], or indirect couplings due to loop contributions [156–158].

Mostly, such resonant models with an additional scalar resonance introduce a heavy particle, [7]. In the composite Higgs models couplings to transverse gauge bosons can be mediated either by technipions [159] or heavy fermions [160, 161]. If the mass scale introduced by a new heavy particle is beyond the energy range of the LHC, the corresponding model can then be parameterized in an EFT expansion. As an example, recent studies [153, 162–164] for the diphoton excess [165, 166] used an EFT approach to investigate the effect of a possible diphoton resonance. In general, vector boson resonances were explicitly investigated in WZ scattering, see [167]. For models with heavy resonance like in the Little Higgs [168, 169], the coefficients of SMEFT were calculated, see [170].

However, we do not refer to a specific model, but provide a (largely) model-independent method to describe a scalar resonance scenario as we are interested in the exemplary behavior of a single resonance to generally show the interpolation between the continuum model and the resonance model, which has been already demonstrated for the scalar case [19, 130].

The Lagrangian (taken from [19]) for the isoscalar-scalar σ is extended by couplings to the electroweak (scalar) currents $J_{\mathrm{W}}^{(4)}$ and $J_{\mathrm{B}}^{(4)}$,

$$\mathcal{L}_\sigma = -\frac{1}{2}\sigma\left(m_\sigma^2 - \partial^2\right)\sigma + \sigma\left(J_{\sigma\parallel} + J_{\sigma\perp}\right), \tag{4.100}$$

$$J_{\sigma\parallel} = F_{\sigma\mathrm{H}}\,\mathrm{tr}\left(\left(D_\mu \boldsymbol{H}\right)^\dagger \left(D^\mu \boldsymbol{H}\right)\right), \tag{4.101}$$

$$J_{\sigma\perp} = g^2 F_{\sigma\mathrm{W}}\,\mathrm{tr}\left(\boldsymbol{W}_{\mu\nu}\boldsymbol{W}^{\mu\nu}\right) + g'^2 F_{\sigma\mathrm{B}}\,\mathrm{tr}\left(\boldsymbol{B}_{\mu\nu}\boldsymbol{B}^{\mu\nu}\right), \tag{4.102}$$

and has then three independent coupling parameters $\$F_{\sigma\mathrm{H}}, F_{\sigma\mathrm{W}}$ and $F_{\sigma\mathrm{B}}$. In the low-energy limit, integrating out the scalar resonance by the replacement $\frac{1}{s-m^2+\mathrm{i}\varepsilon} \to \delta(s-m^2)$, we obtain the SMEFT Lagrangian with the following nonzero coefficients of the dimension-eight operators at LO:

$$F_{S,0} = \frac{F_{\sigma\mathrm{H}}^2}{2m_\sigma^2}, \tag{4.103}$$

$$F_{M,0} = -\frac{F_{\sigma\mathrm{H}}F_{\sigma\mathrm{W}}}{m_\sigma^2}, \tag{4.104}$$

$$F_{M,2} = -\frac{F_{\sigma\mathrm{H}} F_{\sigma\mathrm{B}}}{m_\sigma^2}, \tag{4.105}$$

$$F_{T,0} = \frac{F_{\sigma\mathrm{W}}^2}{2m_\sigma^2}, \tag{4.106}$$

$$F_{T,5} = \frac{F_{\sigma\mathrm{W}} F_{\sigma\mathrm{B}}}{2m_\sigma^2}, \tag{4.107}$$

$$F_{T,8} = \frac{F_{\sigma\mathrm{B}}^2}{2m_\sigma^2}. \tag{4.108}$$

The relation between coupling constants to the electroweak currents and the resonance mass m_σ is given by the width of the scalar resonance,

$$\begin{aligned}\Gamma(m_\sigma) = \int \mathrm{d}\Omega \frac{|\vec{p}|}{32\pi^2 m_\sigma^2} \Big(&\left|\mathcal{M}_{\sigma\to \mathrm{W^+W^-}}\right|^2 + \frac{1}{2}\left|\mathcal{M}_{\sigma\to \mathrm{ZZ}}\right|^2 \\ &+\frac{1}{2}\left|\mathcal{M}_{\sigma\to \mathrm{HH}}\right|^2 + \left|\mathcal{M}_{\sigma\to \mathrm{Z}\gamma}\right|^2 + \frac{1}{2}\left|\mathcal{M}_{\sigma\to \gamma\gamma}\right|^2 \Big),\end{aligned} \tag{4.109}$$

with $|\vec{p}| = m_\sigma/2$. The masses of the electroweak gauge bosons are neglected in the kinematics of the phase space as $m_\sigma \gg m_V$.

We only consider the contribution of $F_{\sigma\mathrm{W}}$ and set the parameters $F_{\sigma\mathrm{H}}, F_{\sigma\mathrm{B}}$ to zero, as $F_{\sigma\mathrm{H}}$ has been covered in [19], and contributions by $F_{\sigma\mathrm{B}}$ break custodial symmetry. Then the resonance width becomes

$$\Gamma_{\mathrm{W}}(m_\sigma) = \frac{3m_\sigma^3}{16\pi} g^4 F_{\sigma\mathrm{W}}^2 \left(1 + \mathcal{O}\left(1/m_\sigma^2\right)\right). \tag{4.110}$$

In particular, by fixing the mass of the scalar m_σ we can directly connect the resonance width $\Gamma_{\sigma\mathrm{W}}$ and the coupling strength $F_{\sigma\mathrm{W}}$.

In the low-energy limit, only $\mathcal{L}_{T,0}$ contributes. Therefore, we can compare the analogous distributions of the continuum model and the resonance model with no further adjustment, since both evolve from the same low-energy limit. The free parameter for the continuum model is $F_{T,0}$, the free parameter for the resonance model is the decay width eq. (4.110) which depends on two parameters, the scalar mass and the scalar-W coupling strength. The parameters have to be chosen in such a way, that eq. (4.103) is fulfilled, in order to compare them.

Isospin-Spin Amplitudes

We use the same approach as in section 4.3 to determine the eigenamplitudes for all polarizations by decomposition with Wigner D-functions. Again, we limit ourselves to the operators $\mathcal{L}_{\mathcal{T},0/1/2}$ and $\mathcal{L}_{T,0/1/2}$, since these conserve custodial symmetry.

Already in [19], it was shown that the kinematic functions for the unitarized amplitudes for resonances are not simple powers of s, but contain logarithms and pole-like rational functions of s in general. For the isospin-spin amplitudes in the case of an isoscalar-scalar

resonance as given in table 4.4, we define the following kinematic functions:

$$X(s,m) = \frac{3s^2}{s-m^2}, \tag{4.111}$$

$$\mathcal{S}_0(s,m) = 2m^2 + 2\frac{m^4}{s}\log\left(\frac{m^2}{s+m^2}\right) - s, \tag{4.112}$$

$$\mathcal{S}_1(s,m) = 4\frac{m^4}{s} + 6m^4\left(2m^2+s\right)\log\left(\frac{m^2}{s+m^2}\right) + \frac{s}{3}, \tag{4.113}$$

$$\mathcal{S}_2(s,m) = 6\frac{m^4}{s^2}\left(2m^2+s\right) + 2\frac{m^4}{s^3}\left(6m^4+6m^2s+s^2\right)\log\left(\frac{m^2}{s+m^2}\right), \tag{4.114}$$

$$\tilde{\mathcal{S}}_2(s,m) = \frac{m^2}{3} - \frac{m^4}{2s} + \frac{m^6}{s^2} + \frac{m^8}{s^3}\log\left(\frac{m^2}{s+m^2}\right) - \frac{s}{4}. \tag{4.115}$$

	j			
i	0	1	2	$\boldsymbol{\lambda}$
0	$X(s,m)$	0	0	$(+,+,+,+)$
	0	0	$\tilde{\mathcal{S}}_2(s,m)$	$(+,-,+,-)$
	0	0	$\tilde{\mathcal{S}}_2(s,m)$	$(+,-,-,+)$
	$X(s,m)+\mathcal{S}_0(s,m)$	0	$\mathcal{S}_2(s,m)$	$(+,+,-,-)$
1	0	0	0	$(+,+,+,+)$
	0	0	$-\tilde{\mathcal{S}}_2(s,m)$	$(+,-,+,-)$
	0	0	$\tilde{\mathcal{S}}_2(s,m)$	$(+,+,+,+)$
	0	$\mathcal{S}_1(s,m)$	0	$(+,+,-,-)$
2	0	0	0	$(+,+,+,+)$
	0	0	$\tilde{\mathcal{S}}_2(s,m)$	$(+,-,+,-)$
	0	0	$\tilde{\mathcal{S}}_2(s,m)$	$(+,-,-,+)$
	$\mathcal{S}_0(s,m)$	0	$\mathcal{S}_2(s,m)$	$(+,+,-,-)$

Table 4.4: Coefficients c of the isospin-spin amplitudes calculated with eq. (4.81) for the isoscalar-scalar resonance $\mathcal{L}_{\sigma W}$ depending on the helicity of the incoming and outgoing particles. The isospin spin amplitudes are given by $\mathcal{A}_{ij}(s;\boldsymbol{\lambda}) = cg^4s^2$.

4.5 Production Setup

In the following we describe the implementation of the isospin-spin amplitudes in WHIZARD. We outline the setup for computing the cross sections for the different processes in order to assess the properties of the dimension-eight theory and unitarization. Furthermore, we justify the need for parallelization, as well as explain the overall steering of WHIZARD for the production.

Implementation

The implementation of the diagonalized isospin-spin eigenamplitude and the T-matrix unitarization follows [17, 57, 130]. For the operators $\mathcal{L}_{T,i}$ and $\mathcal{L}_{M,i}$ form-factor modified Feynman rules that allow unitarized and non-unitarized evaluation of the dimension-eight operators, are implemented for the processes $W^\pm W^\pm \to W^\pm W^\pm$, $W^\pm W^\mp \to W^\pm W^\mp$, $W^\pm W^\mp \to ZZ$ and $ZZ \to ZZ$ in the model `SSC2` in `O'Mega`, which is available in WHIZARD. The details of the implementation can be found in `targets_Kmatrix_2.ml` where the actual expressions for the corrected isospin-spin eigenamplitudes, dependent on the Mandelstam variables s, t and u, are implemented.

Although the unitarization corrections for the isospin-spin eigenamplitudes are expressed for on-shell evaluation in a $2 \to 2$ VBS process, we evaluate the VBS subprocess off-shell, embedded by quark currents $j_{p\to Vj}$. Hence, we set as Mandelstam variable $\hat{s} = q_1 + q_2$, where q_1 and q_2 correspond to the off-shell momenta of the two vector bosons involved.

The unitarization corrections can be switched off with the Sindarin parameter `fkm = 0`, or conversely, with `fkm = 1` switched on. The model parameters $F_{T,i}$ and $F_{M,i}$ can be set via the Sindarin parameters `ft0`, `ft1`, `ft2`, `fm0`, `fm1` and `fm7` in TeV^{-4}. For the coefficient $F_{\sigma W}$, the Sindarin parameter `gkm_st` has been introduced. The width and mass of the scalar can be set with the mass parameter `mkm_s` and the width parameter `wkm_s`.

Processes

In the continuum model, we first discuss the same-sign vector boson process $pp \to W^+W^+jj$ and the Higgs production process $pp \to HHjj$ in proton-proton collision at $\sqrt{s} = 14\,\mathrm{TeV}$. For the simplified resonance model, we consider the process $pp \to ZZjj$. The first stands out for its characteristic same-sign signature for the leptons from the W decays. The second is very difficult to distinguish from the background, but depends on the triple Higgs coupling, which has yet to be measured. The Higgs production is already considered by a gauge-invariant dimension-six operator [127], but the precision of the prediction can be (significantly) improved by the next order in the SMEFT expansion, thus, by dimension-eight operators.

A short remark on the explicit VV final state of the processes. The final states is not directly measurable due to background, e.g. QCD-induced same-signed W boson pairs, top quark production, Drell-Yan and WZ background [15], and the fact of finite-energy jet resolution. However, we do not compute the fiducial (differential) cross section where we would have to consider the further decay of the vector bosons, or of the Higgs, in final state leptons. We are only interested in the overall physical behavior between low-energy theory and high-energy saturation, hence, we neglect this step as it only obscures further insights.

Firstly, we consider the contributions of the longitudinal-transverse mixed operators $\mathcal{L}_{M,0/1/7}$ to the processes, considering the contribution of each operator one at a time by setting all coefficients to zero, except the active one. We choose three distinctive values, $F_{M,i} = \{2, 10, 50\}\mathrm{TeV}^4$. We present the differential cross section in the invariant diboson mass M_{VV}, which coincides with the effective energy $\sqrt{\hat{s}}$ of the VBS subprocess.

Secondly, we repeat the procedure for the transverse operators $\mathcal{L}_{T,0/1/2}$ with $F_{T,i} = \{2, 10, 50\}\text{TeV}^{-4}$.

Cuts

We calculate the full hadronic differential cross sections, where we use the PDF set `CTEQ6L` [171] at $\sqrt{s} = 14\,\text{TeV}$, which is set by `beams = p, p => pdf_builtin` as default within WHIZARD. Furthermore, we use the standard parameters of WHIZARD, where the electroweak parameters are determined by the G_F scheme through the measured values of $G_\text{F}, m_\text{W}, m_\text{Z}$ and m_H.[15] In addition, we apply the standard cuts, see eq. (4.116), for VBS processes from [2]. In the recent analysis [15] a slightly changed set of VBS cuts has been introduced, which is used in [139].

$$\begin{aligned} M_{jj} &= 500\,\text{GeV}, \\ \Delta\eta_{jj} &> 2.4 \wedge |\eta_j| < 4.5, \\ p_\text{T}^j &> 20\,\text{GeV}. \end{aligned} \tag{4.116}$$

An example for a full Sindarin script is given in appendix B.6.

Computation

For the determination of the differential cross sections in the invariant mass of the diboson final state, which coincides with the effective energy $\sqrt{\hat{s}}$ for the VBS subprocess, we have completely resorted to the computation of total cross sections. We compute the total cross section with an additional cut on the invariant diboson mass $M_{VV} \in [M-50, M+50]\,\text{GeV}$ with the scan parameter M in 100 GeV steps in the range from 100 GeV to 8 TeV. The resulting total cross sections can then be plotted against the invariant diboson mass in bins of $[M-50, M+50]\,\text{GeV}$, this gives us the desired differential cross sections. Therefore, we have to compute a total of 89 individual total cross sections for each configuration.

Typically, one would prefer to generate unweighted events, put them in a histogram in the invariant diboson mass and normalize the histogram to the total cross section of the process. However, initial trials have shown that the differential cross section varies by order of magnitude(s) over the entire energy range, and especially in the tail of the distribution at 8 TeV we expect large errors due to small number of events per histogram bin.[16] This would require a huge number of events and a corresponding chain of tools to circumvent storage of so many event samples. With the total cross-section method, we can control the errors of each bin separately, which was very important for us in order to examine possible subleading errors due to contributions from the mismatch of the off-shell/on-shell evaluation of the isospin-spin eigenamplitude, as well as, errors in the general implementation of the amplitudes.

After our validation, however, the actual event generation should be preferred because the approach with the total cross-section is time consuming and would not have been

[15] Although G_F is a low-energy constant, we use it to fix the SM couplings at LO, thus, without further concern regarding the running of those couplings, except for the QCD coupling.

[16] The error of the bin scales with $\sqrt{N}$, as the number of events are Poisson-distributed.

possible without parallelization, since the calculation of a single point takes on average 4 to 8 days, which we could reduce to some hours.

The computations with WHIZARD, version 2.6.3, were carried out at the high-performance cluster Maxwell at DESY[17]. We used the parallel and doubly-adaptive multi-channel integrator VAMP2 from chapter 3, as well as the RNGStream introduced in section 3.2. For production runs, we used the hybrid parallelization of MPI and OpenMP, where each worker was assigned two OpenMP threads. With the parallelization, it was possible to reduce the calculation of a single cross section from several days to mere hours.

Summarizing, we could reach a relatively small error in the tail of the distribution by computing the total cross section bin-wise, but only with the help of the parallelized Monte Carlo integrator VAMP2.

Simplified Resonance Models

For the simplified resonance model we consider the process $\mathrm{pp} \to \mathrm{ZZ}jj$. We compute the differential cross section for the invariant-mass distribution of the ZZ final state for a scalar resonance with mass $m_\sigma = 1\,\mathrm{TeV}$ and different couplings $F_{\sigma\mathrm{W}} = \{2, 4.72, 10\}\,\mathrm{TeV}^{-1}$. These values correspond to the anomalous quartic couplings $F_{T,i} = \{2, 10, 50\}\,\mathrm{TeV}^{-4}$, where the scalar resonance has been integrated out as in eq. (4.103). The respective decay widths Γ_W are computed from eq. (4.110), see table 4.5.

$F_{\sigma\mathrm{W}}/\mathrm{TeV}^{-1}$	$\Gamma_\mathrm{W}(m_\sigma)/\mathrm{GeV}$
2	43.395(32)
4.72	219.69(16)
10	1084.9(8)

Table 4.5: The decay widths Γ_W are given for the proposed values of $F_{\sigma\mathrm{W}}$. We used as input for the parameter $g = 2m_\mathrm{W}\sqrt{\sqrt{2}G_\mathrm{F}}$ with m_W and G_F taken from [24].

The width Γ_W for the coupling $F_{\sigma\mathrm{W}} = 10\,\mathrm{TeV}^{-1}$ is of the order of the scalar mass $m_\sigma \sim 1\,\mathrm{TeV}$. A distinction between mass and width becomes meaningless in the context of $\Gamma_{\sigma\mathrm{W}} \gtrsim m_\sigma$. Therefore, a prescription as Breit–Wigner resonance is not appropriate anymore [20, p. 463] as the interaction becomes strong.

4.6 Results

In the following we present the predictions from our calculations for the (strongly coupled) continuum models from section 4.1 and the simplified resonances models from section 4.4.

[17] Deutsches Elektron Synchroton, `https:\\www.desy.de`.

(Strongly coupled) Continuum Models

First, we consider the strongly-interacting continuum models. But before discussing the results of our calculations, we outline our expectations with respect to our initial assumption about unitarity, gauge invariance and minimal flavor violation. The presented models describe all possible interactions and interaction strengths which may be observed in future VBS measurements, their robust predictions provide an indication of the sensitivities for collider experiments.

Except for the standard model, none of the aforementioned models is UV-complete, which means in particular that we may expect different behavior of the results in the asymptotic limit, for example, that inelastic channels, i.e. resonances, may appear as final states. However, given our initial assumptions, we do not expect that they exceed the event rate in our considered sector of the strongly-interacting continuum model.

To proceed, we compare the SM predictions, with the unitarized predictions as well as with the naive predictions of the SMEFT. We consider individual operators one at a time by setting only the respective coefficients to nonzero values. For completeness, we have listed in table 4.6 the contributions of the different isospin-spin channels of transverse and mixed operators to the VBS processes.

Process	$\mathcal{L}_{M,0/1/7}$	$\mathcal{L}_{T,0/1/2}$	Higgs?
$W^\pm W^\pm \to W^\pm W^\pm$	$\Delta\mathcal{A}_{21}, \Delta\mathcal{A}_{22}$	$\Delta\mathcal{A}_{22}$	
$W^\pm W^\mp \to W^\pm W^\mp$	$\Delta\mathcal{A}_{00}, \Delta\mathcal{A}_{01}, \Delta\mathcal{A}_{02}, \Delta\mathcal{A}_{11}, \Delta\mathcal{A}_{12}, \Delta\mathcal{A}_{21}, \Delta\mathcal{A}_{22}$	$\Delta\mathcal{A}_{00}, \Delta\mathcal{A}_{02}, \Delta\mathcal{A}_{11}, \Delta\mathcal{A}_{12}, \Delta\mathcal{A}_{22}$	
$W^\pm W^\mp \to ZZ$	$\Delta\mathcal{A}_{00}, \Delta\mathcal{A}_{01}, \Delta\mathcal{A}_{02}$	$\Delta\mathcal{A}_{00}, \Delta\mathcal{A}_{02}$	✓
$W^\pm Z \to W^\pm Z$	$\Delta\mathcal{A}_{11}, \Delta\mathcal{A}_{12}, \Delta\mathcal{A}_{21}, \Delta\mathcal{A}_{22}$	$\Delta\mathcal{A}_{11}, \Delta\mathcal{A}_{12}, \Delta\mathcal{A}_{22}$	✓
$ZZ \to ZZ$	$\Delta\mathcal{A}_{00}, \Delta\mathcal{A}_{01}, \Delta\mathcal{A}_{02}, \Delta\mathcal{A}_{21}, \Delta\mathcal{A}_{22}$	$\Delta\mathcal{A}_{00}, \Delta\mathcal{A}_{02}, \Delta\mathcal{A}_{22}$	

Table 4.6: We list the different contribution of the isospin-spin eigenamplitude to the different processes for the longitudinal-transverse mixed and purely transverse operators. Furthermore, we assign a ✓ to processes whose eigenamplitude decomposition is the same for the respective Higgs-associated process.

Mixed operators

We show the results for $pp \to W^+W^+jj$ in fig. 4.5 and for $pp \to HHjj$ in fig. 4.6, within the continuum model with one nonzero coefficient at a time for the longitudinal-transverse mixed operators with parameters $F_{M,0/1/7}$, respectively. We made the particular choice for the parameters, $F = \{2, 10, 50\}\,\mathrm{TeV}^{-4}$. Solid lines show the distributions over the invariant mass of the final state VV/HH pair coinciding (in our setup) with the effective energy $\sqrt{\hat{s}}$ of the embedded VBS (sub-)process. For comparison, the black solid line shows the SM prediction and the dashed lines show the naive predictions of SMEFT, without unitarization, respectively.

We spot around 1 TeV that both the unitarized solid and naive SMEFT dashed curves rise commonly from the SM curve, predicting higher event rates. After a 1 TeV to 2 TeV wide transition region, the unitarized amplitude of the W^+W^+ approximates an asymptotic differential cross section that is an order of magnitude larger than the SM prediction. For the HH final state, the differential cross section is enhanced by two orders of magnitude. For both processes, the dashed curves predict an huge amount of events. Such an overestimate suggests a sensitivity to the model parameters which is unphysical.

To summarize, regardless of the chosen parameterization, the solid unitarized curves approaches an asymptotic differential cross section. This asymptotic limit corresponds to a maximally-strong interaction, which saturates in the unitarity limit. Therefore, the high-energy prediction is mainly controlled by the unitarization requirement [17], however, as the differential cross section falls off asymptotically the maximally allowed event rate is given. The remaining parameter dependence is confined to the transition region, beyond which we can read off the maximally allowed event rate from the saturated quasi-elastic amplitude corresponding to the individual isospin-spin eigenamplitudes, see table 4.6.

Transverse operators

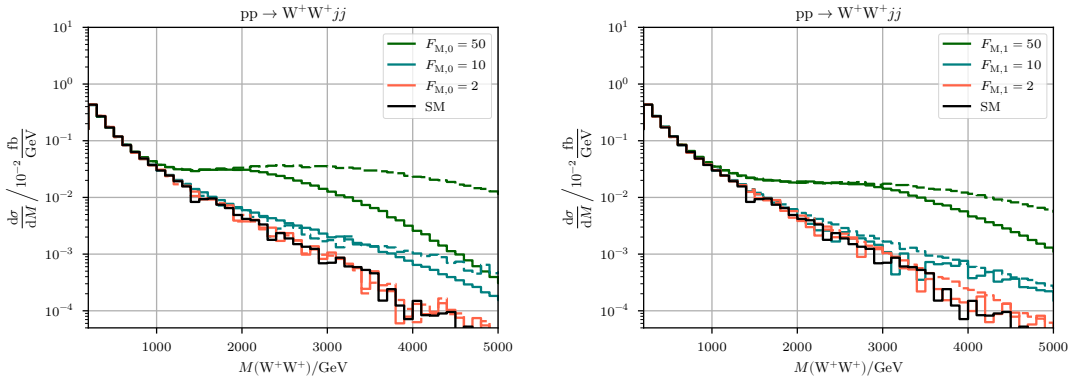
In fig. 4.7 we show the results of the $pp \to W^+W^+jj$ for the transverse operators with parameters $F_{T,0/1/2}$, respectively. The purely transverse operators do not contribute to any aQGC involving the Higgs, thus, the HH channel is not affected.

We spot again around 1 TeV that both the unitarized solid and naive SMEFT dashed curves rise commonly from the SM curve, predicting higher event rates. After a 1 TeV to 2 TeV wide transition region, the unitarized amplitude, solid curve, of the W^+W^+ process approximates an asymptotic differential cross section. The plots indicate an enhancement of the purely transverse operators over the SM prediction by two orders of magnitude.

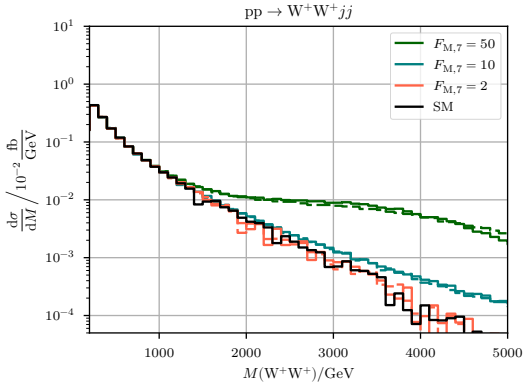
Again, the naive SMEFT result overestimates the possible event rates. We may interpret the result of the unitarized SMEFT in such a way that the purely transverse operators are much more sensitive to new physics than the longitudinal-transverse mixed operators. This may be accounted for in data analyses.

Simplified resonance models

In fig. 4.8 we see that for the two parameters $F_{\sigma W} = 2\,\mathrm{TeV}^{-1}$ and $4.72\,\mathrm{TeV}^{-1}$ the resonance peaks are well aligned above 1 TeV, but for $10\,\mathrm{TeV}^{-1}$ there is a significant deviation of the resonance position. For our further consideration, we exclude the curves



(a) $\mathcal{L}_{M,0}$: pp → W⁺W⁺jj **(b)** $\mathcal{L}_{M,1}$: pp → W⁺W⁺jj



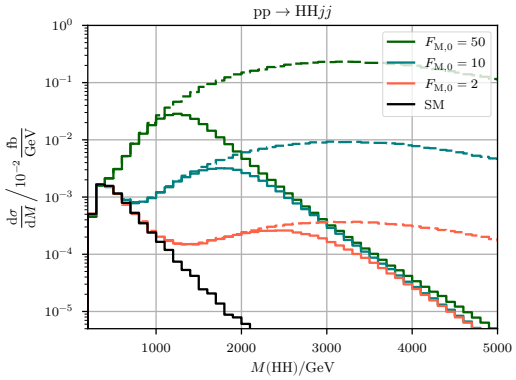
(c) $\mathcal{L}_{M,7}$: pp → W⁺W⁺jj

Figure 4.5: Differential cross section for the diboson invariant mass for the process pp → W⁺W⁺jj. The solid black line shows the SM differential cross section, the greenish, blueish and redish lines the cross sections with anomalous couplings $F_{M,i} = \{2, 10, 50\}\,\text{TeV}^{-4}$, respectively. Solid: unitarized; dashed: naive result. Cuts like in eq. (4.116).

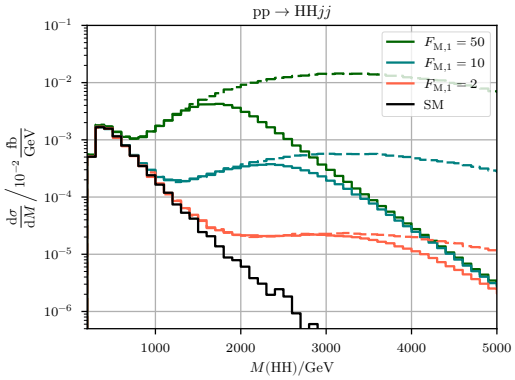
for this parameter, since the resonance peak, independent of whether the unitarization projection was applied or not, does not align at 1 TeV, where the propagator for the scalar acquire an imaginary part. We attribute this to the width $\Gamma_{\sigma\text{W}}$ being of the order of the scalar mass m_σ (see page 89), thus, we enter the regime of a strong interaction,[18] in contradiction to our goal studying weakly interacting resonance models.

Despite that, we observe that the resonance peaks becomes narrower and more pronounced by the unitarization projection. We attribute this to the fact that we have neglected the subleading terms in the calculation of the resonance width, which, however, are taken into account in the unitarity projection. Furthermore, in the high-energy limit, the amplitude is suppressed as result of a saturated isospin-spin eigenamplitude with s^2 terms originating from a derivative coupling. This derivative coupling is a typical

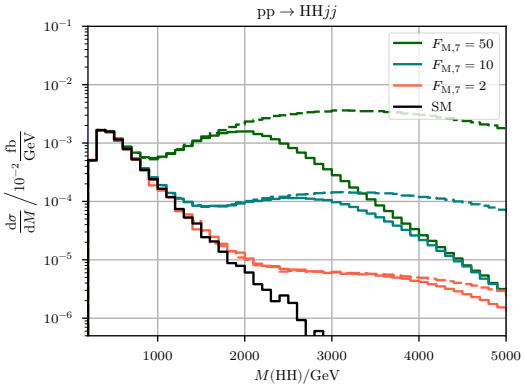
[18] This can also be seen by comparing the non-unitarized curve for $F_{\sigma\text{W}} = 10\,\text{TeV}^{-1}$ with the results from the strongly interacting continuum models in the previous section.



(a) $\mathcal{L}_{M,0}$: pp → HHjj



(b) $\mathcal{L}_{M,1}$: pp → HHjj

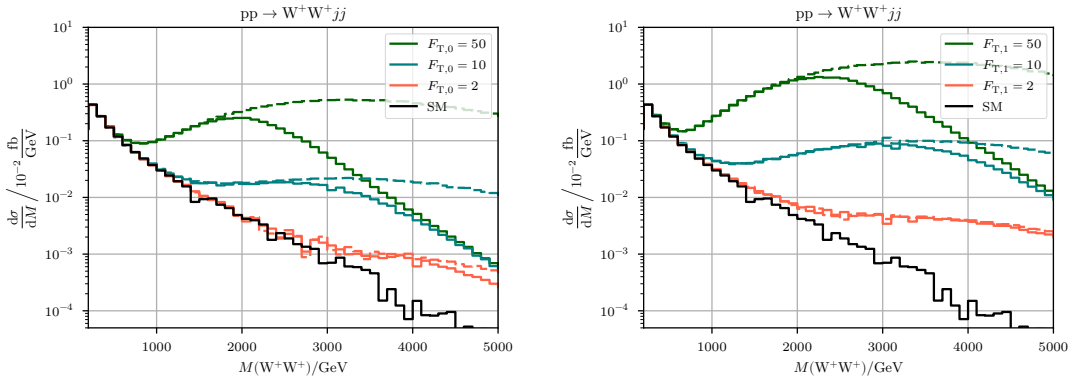


(c) $\mathcal{L}_{M,7}$: pp → HHjj

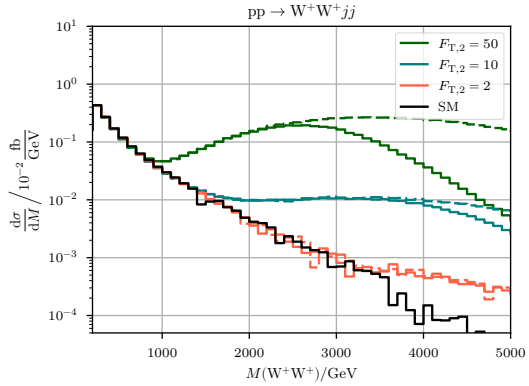
Figure 4.6: Differential cross section for the diboson invariant mass for the process pp → HHjj. The solid black line shows the SM differential cross section, the greenish, blueish and redish lines the cross sections with anomalous couplings $F_{M,i} = \{2, 10, 50\}\,\text{TeV}^{-4}$, respectively. Solid: unitarized; dashed: naive result. Cuts like in eq. (4.116).

feature of strong interactions and is a necessary property of a resonance of higher spin. Conversely, this means that saturation from unitarization plays an important role for the correct description. Therefore, the use of T-matrix unitarization, which allows for arbitrary complex amplitudes and non-perturbative models in contrast to the K-matrix unitarization, is indispensable to cover the entire kinematic range, in particular, as the amplitude becomes complex in the region of the resonance (or strongly interacting).

In fig. 4.9 we compare the continuum model with the matching scalar resonance from eq. (4.103), with and without unitarization projection. For the small coupling, the resonance behaves almost like an elementary particle, and there is almost no difference between the unitarized and non-unitarized amplitudes. In addition, the resonance peak is not approximated by the continuum model as the resonance rises far before the strongly interacting rise of the continuum model around 2 TeV. In addition, the high-energy behavior of the continuum model does not match that of the scalar resonance model,



(a) $\mathcal{L}_{T,0}$: pp → W^+W^+jj **(b)** $\mathcal{L}_{T,1}$: pp → W^+W^+jj



(c) $\mathcal{L}_{T,2}$: pp → W^+W^+jj

Figure 4.7: Differential cross section for the diboson invariant mass for the process pp → W^+W^+jj. The solid black line shows the SM differential cross section, the greenish, blueish and redish lines the cross sections with anomalous couplings $F_{T,i} = \{2, 10, 50\}\,\text{TeV}^{-4}$, respectively. Solid: unitarized; dashed: naive result. Cuts like in eq. (4.116).

whether unitarized or not. We may conclude that it may be more interesting to search experimentally for a weakly coupled resonance in VBS based on such a simplified model as the sigma model eq. (4.100), than to find deviations by SMEFT Wilson coefficients. We conclude that the simplified model allows us to smoothly interpolate between strongly and weakly interacting models. Possibly, the interpolation may leave the range of application of the perturbation series.

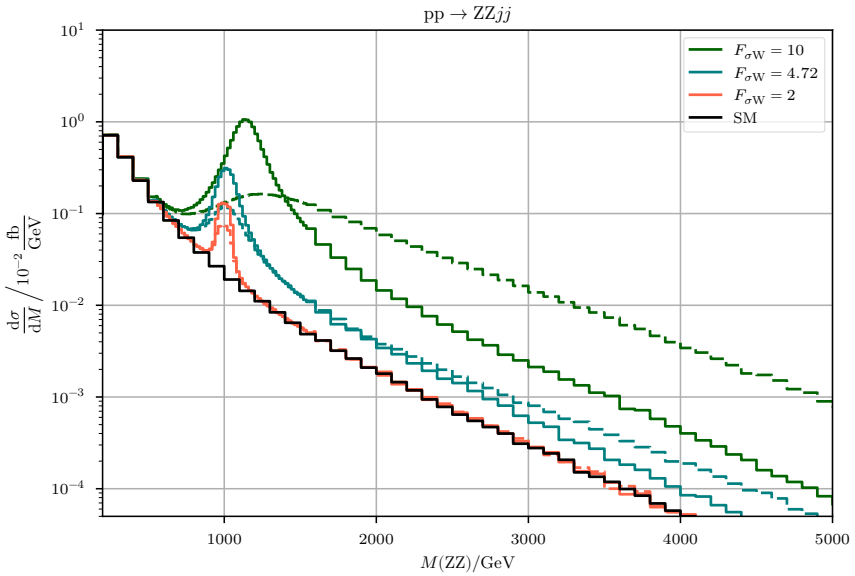


Figure 4.8: Differential cross section for the diboson invariant mass for the process pp $\to$ ZZjj. The solid black line shows the SM differential cross section, the greenish, blueish and redish lines the cross sections with a scalar resonance with mass $m_\sigma = 1\,\text{TeV}$ and coupling of $F_{\sigma W} = \{2, 4.72, 10\}\,\text{TeV}^{-1}$, respectively. Solid: unitarized; dashed: naive result. Cuts like in eq. (4.116).

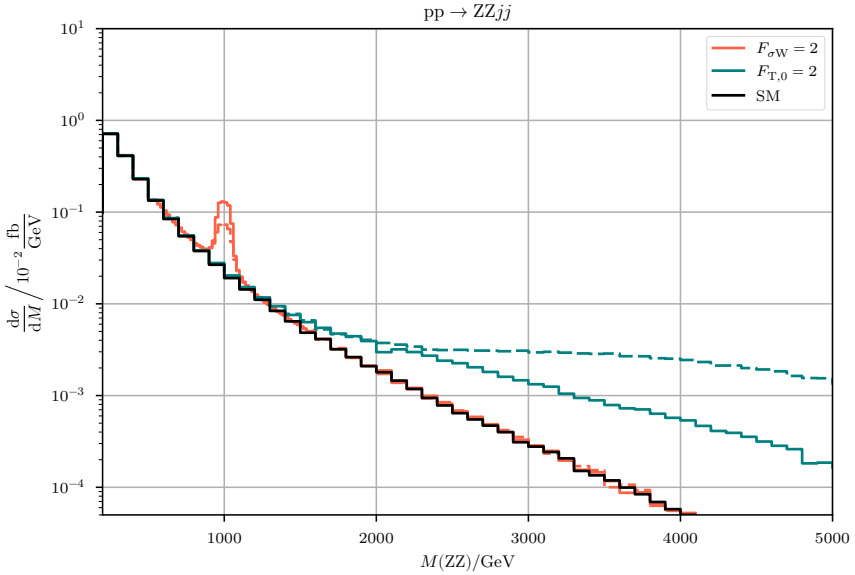


Figure 4.9: Differential cross section for the diboson invariant mass for the process pp $\to$ ZZjj. The solid black line shows the SM differential cross section, the blueish lines show an anomalous coupling $F_{T,i} = 2\,\text{TeV}^{-4}$ and the redish lines show a scalar resonance with mass $m_\sigma = 2\,\text{TeV}^{-1}$. Solid: unitarized; dashed: naive result. Cuts like in eq. (4.116).

4.7 Discussion

In summary, we find that (i) the low energy theory dictated by the SMEFT smoothly interpolates to the high-energy prediction of the unitary saturation, (ii) a violation of the unitary bound is given for *any* dimension-eight operator, (iii) the T-matrix unitarization scheme preserves the smooth interpolation between low and high energy prediction up to the saturation of the unitary bound, (iv) the here probed simplified scalar resonance allows for a smooth interpolation between weakly and strongly interacting models, where the unitarization of the resonance leads to (Dyson) resummation of the resonance propagator.

SMEFT provides us with a well-defined framework for theoretical predictions at the LHC for the energy below the TeV scale. However, the SMEFT predicts unphysically large event rates for high energies, since amplitudes from the dimension-eight operators grow with energy. Thus, they violate the unitary bound, which happens independently of the choice of the Wilson coefficients, unless they are so small that their predictions are no longer distinguishable from SM predictions. We have studied the behavior for VBS processes at the LHC, and our findings, in this regard, are consistent with the study [139]. We found that the SMEFT expansion does not provide a systematic expansion or meaningful description of the complete dataset.

We want to specify this "failure" of the naive SMEFT in more detail: SMEFT does not allow for a meaningful description of the considered processes that can be viewed as model-independent. However, this failure should not be understood in the sense of a quantum field theoretical failure of a valid (but truncated) expansion of a UV-complete model, the SM. It rather expresses the practical problem of the tension between a size of operator coefficients experimentally detectable at the LHC and the limitations from the unitary bounds.

Therefore, a physical interpretation is meaningful only in the context of a unitarization, as already presented in [17, 57, 130]. To estimate the contribution through individual parameters, this is an indispensable tool. Due to the minimal T-matrix projection, the saturation of the elastic channels provides us with a maximum limitation on the achievable parameter sensitivity. Especially with regard to the determination of the parameter sensitivity, this is important as in the unitarized model contribution is by an order of magnitude weaker than a naive determination[19] lets us expect. However, the description of the non-SM behavior is no longer model-independent within the chosen unitarization scheme, as it dictates more or less the behavior in the asymptotic saturation. For comparison with experimental data without a unitarization scheme, kinematic regions would have to be removed that would violate unitarity in the comparison model, thus one would lose valuable information. We conclude, therefore, that a well-defined, but model-dependent, i.e. dependent on the unitarization scheme, parameterization is possible without losing the SMEFT as a systematic description of the low-energy region.

For the energy range that is accessible at the LHC, we have provided predictions, from which we can read off the expected range of event rates and distributions, for particular models. If one observes an excess over these event rates, it could hint to a possibility that

[19]The naive description predicts more events than are physical, so a fit against experimental data tends to constrain the Wilson coefficients more than physically meaningful. By contrast, the unitarized description predicts physical event rates that are smaller, making the bounds on the Wilson coefficients less stringent [139].

assumptions are not realized in nature. Though, such violations would strongly point to certain new physics scenarios, e.g. direct involvement of fermions into new, possibly Higgs-scenario interactions (in contrast, we only regarded them as spectators to new physics), or that the gauge symmetry of the SM is only valid for energies below the TeV scale, or that the four-dimensional QFT loses its universal validity. However, the current success of the SM in the low energy-data make this scenario seem rather unlikely. Hence, we believe that the quantitative results obtained within the framework of unitary simplified models reliably exhaust the range that can be expected from real data.

In addition, we have extended our predictions by the processes W^+W^-, W^+Z and ZZ (see appendix B.7), but these are less accessible at the LHC. However, they are necessary if one wants to assess several anomalous couplings, like given in our model, so it is necessary to increase the number of (free) observables in a global fit of all parameters.

4.8 Future Improvements

Our results were only generated with variation of a single Wilson coefficient. However, these results do not provide realistic predictions, despite our sophisticated initial assumptions such as enforced unitarity, gauge symmetry, and minimal flavor violation, which was introduced by the assumption of custodial $SU(2)_C$ symmetry. A more realistic approach would take several parameters into account at the same time as we would expect in a real process. In order to reveal the correlations between the parameters and their sensitivity to collider experiments, a global fit is necessary.

5 Conclusion and Outlook

In this thesis we introduced an approach to parallel adaptive Monte Carlo integration and with its help we studied possible beyond the SM contributions to VBS processes using SMEFT at the LHC.

Monte Carlo Event Generators are an essential tool for the simulation of event-distributions of elementary processes for experimental analyses at present and future colliders to study the SM and search for New Physics beyond the SM. Increased demand on high multiplicity and high precision in the detection of signal and background processes call for appropriate computing resources. One major contribution to the computation time is the multi-dimensional phase-space integration as well as determinig a most efficient sampling for (unweighted) event generation automatically.

We have described an efficient Monte Carlo algorithm that applies an automatic iterative adaptation to integration and event generation, and their respective parallelization. We outlined the two major parallelization concepts, MPI and OpenMP, and discussed their applicability. We reviewed the generation of random numbers in parallel application and how it helps to avoid unnecessary communication for random numbers using the RNGstream.

The parallel evaluation is based on the paradigm of the MPI standard in conjunction with OpenMP multi-threading. For the concrete realization, the algorithm has been implemented within the framework of the multi-purpose event generator Whizard. The new code constitutes a replacement for the Vamp adaptive multi-channel integrator and makes active use of modern features in the current MPI-3.1 standard. Our initial tests for a variety of benchmark physics processes demonstrate a speedup by a factor > 10 with respect to serial evaluation. The best results have been achieved by sole MPI parallelization.

We were able to show that, in general, hybrid parallelization with OpenMP and MPI leads to a speedup which is comparable to sole MPI parallelization. However, combining both approaches is beneficial for tackling memory-intense processes, such as 8- or 10-particle processes. Depending on particular computing resources, the latter approach allows for a more efficient use of the locally available memory at a computing node. In the hybrid approach, Whizard is parallelized on individual multi-core nodes via OpenMP multi-threading, while distinct computing nodes communicate with each other via MPI. The setup of the system allows for sufficient flexibility to make optimal use of both approaches for a specific problem.

The initial tests pointed to further possibilities for improvement. For this purpose, we have proposed an improved load balancing in comparison to a first, naive load balancing approach. Furthermore, a server/client structure should give the freedom to re-allocate and assign computing workers dynamically during a computing task, and thus make a more efficient use of the available resources. Further speedup can be expected from removing various remaining blocking communications and replacing them by non-blocking communication while preserving the integrity of the calculation. Finally, we note that the

algorithm shows its potential for calculations that spend a lot of time in matrix-element evaluation.

A first application of the new implementation and the parallelization were the simulations for [6]. Beyond that, the parallelization proves to be a keystone in the present validation of the NLO automation in WHIZARD [172] since the evaluation of NLO matrix elements is much more time-consuming than for LO ones. For instance, in some tests of NLO QCD processes we found that the time required for integration could be reduced from the order of a week down to a few hours.

We have demonstrated a further application of the VAMP2 implementation and its parallelization in the search for new physics in the electroweak and Higgs sectors in the TeV range accessible at the LHC. For this purpose, we studied transverse modes and Higgs bosons in vector-boson scattering as quasi-elastic scattering of the type VV $\to$ VV to investigate new physics as anomalous contributions in quartic couplings. We use a bottom-up effective field theory approach to study new, unknown physics in a model-independent framework. We consider the Standard Model effective field theory (SMEFT), which preserves the local gauge symmetry of the Standard Model. In order to assess anomalous quartic gauge couplings independently of anomalous trilinear gauge couplings in the SMEFT, we had to deviate from the usual first order approach of SMEFT with dimension-six operators to the next order with dimension-eight operators. We consider transverse and mixed, longitudinal-transverse operators for the aQGC, in the corresponding vector-boson polarizations.

We imposed global chiral symmetry $\mathrm{SU}(2)_\mathrm{L} \times \mathrm{SU}(2)_\mathrm{R}$ as extension of the local gauge symmetry of the SM which becomes the global custodial $\mathrm{SU}(2)_\mathrm{C}$ symmetry under electroweak symmetry breaking. For the definition of the dimension-eight operators we departed from the representation of the Higgs field as doublet in the Eboli-Basis to the Hermitian matrix representation, which is manifestly invariant under chiral and custodial symmetry. We have derived a full set of relations between the scalar operators in our base, the Eboli base and the chiral electroweak Lagrangian, both in the case of conserved custodial symmetry, as well as broken custodial symmetry. These relations prove to be necessary in comparison of experimental analyses and predictions of Wilson coefficients, since different parameterization are in use.

Naive SMEFT prescriptions of the dimension-eight operators for the isospin-spin eigenamplitudes within the sensitivity of the LHC do not respect first principles of QFT, i.e. they violate bounds dictated by the unitarity of the $\mathcal{S}$-matrix. However, an exclusion of high-invariant masses of the vector-boson scattering system would remove valuable information neglecting the interesting high energy region for possible new physics contributions. Deviating from the model-independent paradigm of the EFT, we introduce an (arbitrary) unitary projection to model the unitary saturation in the high-energy region. We apply the T-matrix unitarization projection in order to provide a consistent description of high-energy matched to low energy EFT covered by the SMEFT. For the valid region of the EFT, the T-matrix unitarization leaves the theoretical prediction unchanged and only effects for unitarity violating terms and can be applied in absence of a non-perturbative theory.

We constructed unitary projections yielding simplified models to assess prospects for new physics contribution. In the context of these simplified models, we made quantitative statements for future measurements. We have extended previous works by including

transverse vector-boson polarization modes together with final-state Higgs bosons. In particular, we studied strongly-interacting continuum models where we smoothly interpolated the low-energy SMEFT to the high-energy unitary saturation by applying T-matrix projection. In addition, we considered a simplified resonance model with one scalar resonance coupled to a transverse gauge boson. For both sets of models, we obtained parameter-dependent upper limits for the event rates of all processes for the full energy ranges.

A comparison with data from a future experimental analysis should reveal a hierarchy and correlations between the individual parameters of the dimension-eight operators. Furthermore, the semi-numerical approach is cumbersome and requires some substantial simplifications; future improvements should involve a full numerical approach.

In conclusion, this thesis proposed a parallelization to an efficient, adaptive Monte Carlo integration to face increasing demands on computational complexity and to utilize available computing resources. In a prime application of this, we constructed and studied simplified unitary models based on dimension-eight operators of SMEFT for the search of new physics in vector-boson scattering at the LHC. During this thesis, the newly-implemented, parallelized Monte Carlo integrator Vamp2 and the simplified models have been added for further phenomenological studies at LHC or other future colliders to the public available Monte Carlo event generator Whizard.

A Parallel Adaptive Monte Carlo Integration

A.1 The VEGAS Algorithm: Optimization

We prove in full-length the optimization formulae for a factorizable probability density function, which was only proved schematically in [62].

We want to minimize the effective variance $\left\langle \frac{f^2}{g^2} \right\rangle_p - \left\langle \frac{f}{p} \right\rangle_p^2$ under the probability distribution p. The square of the expected value of f is per definition independent of p. However, the particular choice of p affects the expected value of f^2,

$$\left\langle \frac{f^2}{g^2} \right\rangle_p = \int_0^1 \frac{f^2(x)}{p^2(x)} \, \mathrm{d}P(x) = \int_0^1 \frac{f^2(x)}{p(x)} \, \mathrm{d}x, \tag{A.1}$$

where we used $\mathrm{d}P = p(x)\mathrm{d}x$. The probability distribution has to be normalized and factorizable,

$$p(x) = \prod_{k=1}^{d} p_k(x_k) \quad \wedge \quad \int_0^1 p_k(x_k) \, \mathrm{d}x_k = 1. \tag{A.2}$$

Using the method of Lagrange multipliers, we can connect the factorization and normalization condition with (A.1)

$$\frac{\partial}{\partial p_k} \left\{ \left(\prod_{k=1}^{d} \int_0^1 \mathrm{d}x_k \right) \frac{f^2(x_1, ..., x_d)}{p_1(x_1) \cdot ... \cdot p_d(x_d)} + \sum_{k=1}^{d} \lambda_k \int_0^1 p_k(x_k) \, \mathrm{d}x_k \right\} = 0. \tag{A.3}$$

We exchange derivation and integration. We insert a "one" expression in the λ_k-dependent term $1 = \prod_{k \neq k'} \int \mathrm{d}x_k p_k(x_k)$,

$$\int_0^1 \mathrm{d}x_k \left\{ \frac{-1}{p_k^2(x_k)} \left(\prod_{k \neq k'} \int_0^1 \mathrm{d}x_k \right) \frac{f^2(x_1, ..., x_d)}{\prod_{k \neq k'} p_k(x_k)} + \lambda_k p_k(x_k) \right\} = 0. \tag{A.4}$$

Taking the derivative with respect to x_k and inserting (A.2), we find the optimized probability distribution

$$p_k(x_k) = \frac{1}{\sqrt{\lambda_k}} \left(\prod_{k \neq k'} \int_0^1 \mathrm{d}x_{k'} \frac{f^2(x_1, ..., x_d)}{p(x_1, ..., x_d)} \right)^{\frac{1}{2}}. \tag{A.5}$$

The normalization is given by the derivative of λ_k, instead of p_k, in (A.3),

$$\sqrt{\lambda_k} = \int_0^1 \mathrm{d}x_k \left(\prod_{k \neq k'} \int_0^1 \mathrm{d}x_{k'} \frac{f^2(x_1, ..., x_d)}{p(x_1, ..., x_d)} \right)^{\frac{1}{2}}. \tag{A.6}$$

The optimization weights are then chosen according to (A.5),

$$\omega_{j_k} \propto \tilde{\Delta} \left(\frac{f(x_j)}{p(x_j)} \right)^2, \; (x_j)_k \in B_{k,j_k} \qquad \text{(importance sampling)}, \tag{A.7}$$

$$\omega_{j_k} \propto \frac{1}{N_r} \sum_{i=1}^{N_r} \tilde{\Delta} \left(\frac{f(x_i)}{p(x_i)} \right)^2, \; (x_i)_k \in B_{k,j_k} \qquad \text{(stratified sampling)}. \tag{A.8}$$

A.2 The VEGAS algorithm: Implementation

The VEGAS algorithm is implemented according to the diagram in fig. A.1. We iterate over the d-dimensional equally-binned hypercube, i.e. the coordinate c_i^r of r-space. Each cell in r space is then sampled n_r times. Each time, the coordinates in r-space are mapped to a bin of the integration grid, see fig. 3.1. Using a random number $r \in (0, 1)$, the map from stratified to integration space is given by

$$c_i^g(c_i^r) = \text{int} \left(\frac{c_i^r - 1 + r}{n_r} \cdot n_{\text{bins}} \right) + 1, \tag{A.9}$$

where int converts its argument to the nearest integer. A random point x in the d-dimensional bin c_i^g is then determined by

$$x = \left(\left(\frac{c_i^r - 1 + r}{n_r} \cdot n_{\text{bins}} + 1 \right) - c_i^g \right) \cdot \Delta x_i, \tag{A.10}$$

and the bin volume (inversely proportional to the probability) is determined. Depending on the integration mode, the contribution to the adaptation is updated in the inner loop, for importance sampling, or outside the loop, for stratified sampling. The cumulative result for the integral and the variance is updated at the end.

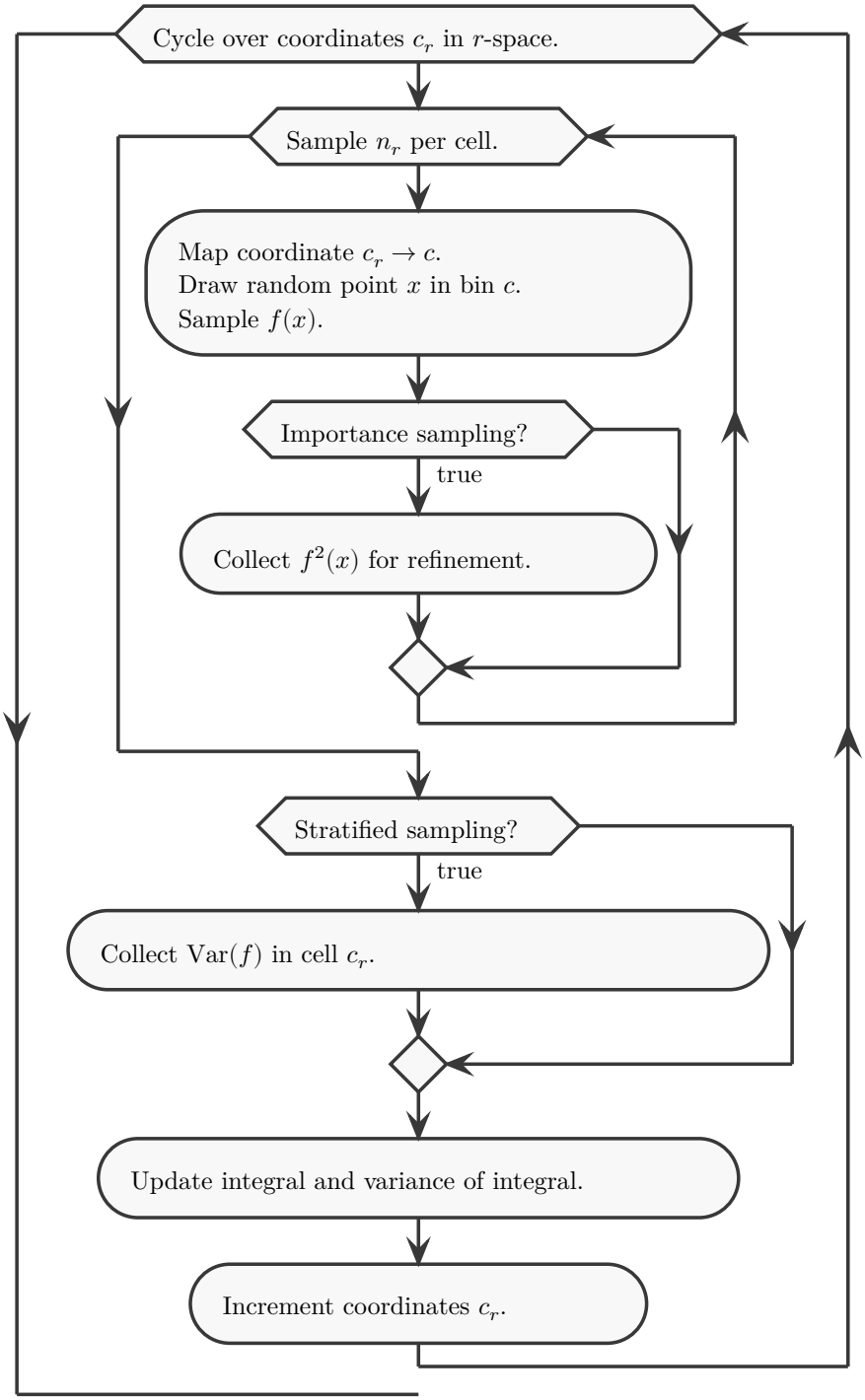


Figure A.1: An overview over the performed steps during integration in the VEGAS integrator.

A.3 Multi-Channel Weight Optimization

The channel weights are free parameters, which are constrained only due to eq. (2.13). Following [60], we outline the derivation of the optimization of the channel weights.

The integral expression in eq. (2.15) is independent of the choice of the channel weights. On the other hand, the total variance $\left(W(\vec{\alpha}) - I^2\right)/N$, with

$$W(\vec{\alpha}) = \int_\Omega \frac{f^2(p)}{h(p)}\,\mathrm{d}\mu(p), \tag{A.11}$$

does explicitly depend on the channel weights. Therefore, we can minimize the total variance by optimizing the channel weights α_c. We use the method of Lagrange multipliers, and find the auxiliary functions

$$\mathcal{L}(\vec{\alpha}, \lambda) = -W(\vec{a}) + \lambda\left(\sum_c \alpha_c - 1\right). \tag{A.12}$$

We find that we reach a minimum $\bar{\alpha}$, when all channel variances are equal, as $\forall c : W_c = -\lambda \equiv W(\bar{\alpha})$ dictates, where we identify

$$W_c \equiv -\frac{\partial}{\partial\alpha_c} W(\vec{a}) = \int h_c(p)(f_c^h)^2. \tag{A.13}$$

as a single channel variance. We verify that $\bar{\alpha}$ gives us indeed a minimum, by expanding $\alpha_c = \bar{\alpha}_c + \beta_c$, $\beta_c \ll \alpha_c$ and $\sum_c \beta_c$. We find that

$$W(\vec{\alpha}) = W(\bar{\alpha}) + \frac{1}{2}\int \frac{f^2(p)}{h^3(p)}\left(\sum_c \beta_c h_c(p) + \mathcal{O}(\beta_c)\right), \tag{A.14}$$

stating that $W(\bar{\alpha})$ is one minimum. However, we can not state that $\bar{\alpha}$ is a global minimum, nor how we can reach such a minimum.

A.4 Benchmark Results

We list the results of our time measurements for the benchmark runs in tables A.1 to A.4. The measurements provide the basis for our conclusions on the actual performance of the parallelized program.

The numbers have to be interpreted with some care; there are sizable fluctuations in the measured run time for long-running processes which can be attributed to a somewhat messy computing environment. All data have been obtained from runs on a university-wide cluster which suffers from high and variable usage by other research groups. A precise measurement of benchmark runs would require an exclusive allocation of computing resources, which was not available.

N_{CPU}	$T(jj \to \text{W}j)\text{s}^{-1}$	$T(jj \to \text{W}jj)\text{s}^{-1}$	$T(jj \to \text{W}jjj)\text{s}^{-1}$
1	865.1 ± 31.9	$19\,621.3 \pm 0.1$	$169\,679.3 \pm 15\,590.2$
2	442.4 ± 2.4	9248.4 ± 18.0	$91\,824.3 \pm 8660.0$
4	228.0 ± 1.8	5320.4 ± 429.1	$53\,487.4 \pm 6279.5$
8	119.3 ± 0.5	2974.3 ± 166.3	$33\,808.9 \pm 2346.9$
16	71.6 ± 0.2	1768.2 ± 105.4	$19\,114.2 \pm 637.3$
32	41.3 ± 0.8	1273.5 ± 16.5	$11\,371.1 \pm 229.0$
64	25.6 ± 0.2	1403.2 ± 1.3	6707.4 ± 0.1
128	25.5 ± 0.0	1477.5 ± 2.1	4976.0 ± 0.1

Table A.1: Computation time of $jj \to \text{W}^-(\to \text{e}^-\bar{\nu}_\text{e}) + nj$ processes.

N_{CPU}	$T(\text{gg} \to \text{W}q\bar{q})\text{s}^{-1}$	$T(\text{gg} \to \text{W}q\bar{q}\text{g})\text{s}^{-1}$	$T(\text{gg} \to \text{W}q\bar{q}\text{gg})\text{s}^{-1}$
1	765.3 ± 1.5	3356.7 ± 93.8	$28\,853.2 \pm 103.2$
2	407.2 ± 0.8	1617.7 ± 54.0	$14\,908.5 \pm 0.1$
4	212.4 ± 5.5	848.0 ± 40.9	7573.8 ± 276.2
8	110.8 ± 0.1	439.0 ± 4.4	4118.6 ± 137.1
16	60.3 ± 0.4	279.1 ± 2.1	2321.3 ± 36.1
32	37.7 ± 0.4	169.0 ± 0.4	1329.0 ± 14.7
64	37.8 ± 0.4	130.7 ± 1.9	788.1 ± 1.0
128	38.8 ± 0.4	111.5 ± 0.7	618.8 ± 9.9

Table A.2: Computation time of $\text{gg} \to \text{W}^-(\to \text{e}^-\bar{\nu}_\text{e})q\bar{q} + n\text{g}$ processes.

N_{CPU}	$T(\text{e}^-\text{e}^+ \to \mu^-\mu^+)\text{s}^{-1}$	$T(\text{e}^-\text{e}^+ \to \mu^-\mu^+\mu^-\mu^+)\text{s}^{-1}$	$T(\text{e}^-\text{e}^+ \to \mu^-\mu^+\mu^-\mu^+\nu_\mu\bar{\nu}_\mu)\text{s}^{-1}$
1	19.8 ± 0.1	561.9 ± 0.1	$83\,927.3 \pm 0.4$
2	10.2 ± 0.1	294.0 ± 0.4	$47\,290.9 \pm 19.5$
4	5.8 ± 0.1	152.7 ± 0.3	$23\,667.4 \pm 19.3$
8	3.5 ± 0.1	87.3 ± 0.0	$14\,415.6 \pm 27.4$
16	2.6 ± 0.1	51.8 ± 0.0	8583.3 ± 0.4
32	2.0 ± 0.1	52.6 ± 0.6	6613.4 ± 26.4
64	1.9 ± 0.1	52.2 ± 0.2	5138.1 ± 0.1
128	2.3 ± 0.1	52.2 ± 0.3	3697.5 ± 25.2

Table A.3: Computation time of the leptonic processes.

process	N_{CPU}	$T(\{\text{d}\})\text{s}^{-1}$	$T(\{\text{u},\text{d}\})\text{s}^{-1}$	$T(\{\text{u},\text{d},\text{s}\})\text{s}^{-1}$	$T(\{\text{u},\text{d},\text{s},\text{c}\})\text{s}^{-1}$
$jj \to \text{W}j$	1	1589.1±3.0	2189.3±8.3	2730.9±1.2	3440.6±11.2
	60	66.8±0.0	123.8±0.7	179.3±0.1	283.1±0.8
$jj \to \text{W}jj$	1	11 075.0±2347.7	29 405.0±1206.9	49 151.5±585.6	91 677.3±2502.7
	60	435.1±0.3	811.7±0.4	1220.4±0.8	2006.0±0.5

Table A.4: Computation time over increasing flavor content. The upper two lines are for the process $jj \to \text{W}j$, the lower two for the process $\text{pp} \to \text{W}jj$, respectively. The second column gives the number of CPU cores, the following columns are the results for an increasing number of massless quark flavors in the initial state and jets, growing from one (d) to four $(\text{d},\text{u},\text{s},\text{c})$.

B Vector-Boson Scattering at the LHC

B.1 Electroweak Chiral Lagrangian

In order to compare results for WWWW, WWZZ and ZZZZ calculated from the scalar operators from the non-linear EFT,[1] with the dimension-eight scalar operators in eq. (4.12) of the linear Higgs representation,[2] the most important points from the chiral electroweak Lagrangian are introduced. We use them then in section 4.1 for comparison of the operators from the linear representations, i.e. doublet or matrix notation, with the electroweak chiral theory.

Following among the lines of [57, 125], we introduce the non-linear representation of electroweak symmetry breaking using a matrix-valued auxiliary field $\mathbf{\Sigma}$, which transforms as

$$\mathbf{\Sigma} \to U_{\mathrm{L}} \mathbf{\Sigma} U_{\mathrm{R}}^{\dagger}, \tag{B.1}$$

under local $SU(2)_{\mathrm{L}} \times U(1)_{\mathrm{Y}}$ electroweak gauge symmetry.

The $\mathbf{\Sigma}$ matrix can be parameterized in a Higgs-less scenario by

$$\mathbf{\Sigma}(x) = \exp\left(\frac{-\mathrm{i}}{v}\mathbf{w}(x)\right), \tag{B.2}$$

with the scalar triplet $\mathbf{w} = w^a \frac{\tau^a}{2}$ and the Pauli matrices τ^a. The scalar triplet refers to the Goldstone bosons, which vanish in unitary gauge leading to $\Sigma = 1$.

For the QGC relevant operators, respecting custodial symmetry $\mathrm{SU}(2)_{\mathrm{C}}$ and conserving CP, are [136, 173–175],

$$\mathcal{L}_4 = \alpha_4 \,\mathrm{tr}\left[V_\mu V_\nu\right]^2, \tag{B.3}$$

$$\mathcal{L}_5 = \alpha_5 \,\mathrm{tr}\left[V_\mu V^\mu\right]^2, \tag{B.4}$$

where α_4 and α_5 are dimensionless coefficients. With further definitions,

$$\mathbf{V}_\mu = \mathbf{\Sigma}\left(\boldsymbol{D}_\mu \mathbf{\Sigma}\right) = -\left(\boldsymbol{D}_\mu \Sigma\right)\Sigma^\dagger, \tag{B.5}$$

$$\boldsymbol{D}_\mu \mathbf{\Sigma} = \partial_\mu \mathbf{\Sigma} + \mathrm{i}g\frac{\tau^a}{2} W_\mu^a \mathbf{\Sigma} - \mathrm{i}g' \Sigma B_\mu \frac{\tau^3}{2}, \tag{B.6}$$

[1] As implemented in Whizard [17, 57, 130].

[2] As implemented in Vbfnlo [123, 128, 139] or MadGraph [52] (the latter using the convention of Feynrules [150, 151]).

we find [135],

$$\begin{aligned}\mathcal{L}_4 = \alpha_4 \Bigg(& 8\frac{M_{\mathrm{W}}^4}{v^4}\left((\mathrm{W}^-\cdot\mathrm{W}^-)(\mathrm{W}^+\cdot\mathrm{W}^+) + (\mathrm{W}^-\cdot\mathrm{W}^+)(\mathrm{W}^-\cdot\mathrm{W}^+)\right) \\ & + 16\frac{M_{\mathrm{W}}^2 M_{\mathrm{Z}}^2}{v^4}(\mathrm{W}^-\cdot\mathrm{Z})(\mathrm{W}^+\cdot\mathrm{Z}) + 4\frac{M_{\mathrm{Z}}^4}{v^4}(\mathrm{Z}\cdot\mathrm{Z})(\mathrm{Z}\cdot\mathrm{Z})\Bigg) \end{aligned} \tag{B.7}$$

$$\begin{aligned}\mathcal{L}_5 = \alpha_5 \Bigg(& 16\frac{M_{\mathrm{W}}^4}{v^4}\left((\mathrm{W}^-\cdot\mathrm{W}^+)(\mathrm{W}^-\cdot\mathrm{W}^+)\right) \\ & + 16\frac{M_{\mathrm{W}}^2 M_{\mathrm{Z}}^2}{v^4}(\mathrm{W}^-\cdot\mathrm{W}^+)(\mathrm{Z}\cdot\mathrm{Z}) + 4\frac{M_{\mathrm{Z}}^4}{v^4}(\mathrm{Z}\cdot\mathrm{Z})(\mathrm{Z}\cdot\mathrm{Z})\Bigg). \end{aligned} \tag{B.8}$$

For the Higgs field, the matrix notation is chosen, which manifestly represents the larger global symmetry $\mathrm{O}(4) \sim \mathrm{SU}(2)_{\mathrm{L}} \times \mathrm{SU}(2)_{\mathrm{R}}$, which becomes after the EWSB the approximate custodial $\mathrm{SU}(2)_{\mathrm{C}}$ symmetry. The Higgs multiplet is then written in the form of a 2×2 matrix,

$$\boldsymbol{H} = \frac{1}{2}\begin{pmatrix} v + h - \mathrm{i}w^3 & -\mathrm{i}\sqrt{2}w^+ \\ -\mathrm{i}\sqrt{2}w^- & v + h + \mathrm{i}w^3 \end{pmatrix}, \tag{B.9}$$

with the three Goldstone bosons w^a. A simple relation holds between Higgs matrix field $\mathbf{H}$ and auxiliary field $\boldsymbol{\Sigma}$ [17],

$$\mathbf{H} \to \frac{1}{2}(v + h)\,\Sigma, \tag{B.10}$$

with the physical Higgs field h. A new set of scalar operators has been defined analogously to the dimension-eight scalar operators in eq. (4.12) for the chiral Lagrangian [17],

$$\mathcal{L}_{S,0} = \hat{F}_{S,0}\,\mathrm{tr}\left[(\boldsymbol{D}_\mu\mathbf{H})^\dagger \boldsymbol{D}_\nu\mathbf{H}\right]\mathrm{tr}\left[(\boldsymbol{D}^\mu\mathbf{H})^\dagger \boldsymbol{D}^\nu\mathbf{H}\right], \tag{B.11}$$

$$\mathcal{L}_{S,1} = \hat{F}_{S,1}\,\mathrm{tr}\left[(\boldsymbol{D}_\mu\mathbf{H})^\dagger \boldsymbol{D}^\mu\mathbf{H}\right]\mathrm{tr}\left[(\boldsymbol{D}_\nu\mathbf{H})^\dagger \boldsymbol{D}^\nu\mathbf{H}\right], \tag{B.12}$$

the coefficients $F_{S,0}$ and $F_{S,1}$ are of mass dimension -4. With the covariant derivative,

$$\boldsymbol{D}_\mu = \partial_\mu - \mathrm{i}gW_\mu^a\frac{\tau^a}{2} + \mathrm{i}g'B_\mu\frac{\tau^3}{2}. \tag{B.13}$$

and, the parametrization of the 2×2 Higgs doublet is given in unitary gauge by

$$\mathbf{H} = \frac{v + h}{2}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \tag{B.14}$$

A full equivalence can then be found between the chiral Lagrangian in eq. (B.3) and eq. (B.11) with Higgs and the scalar dimension-8 operators eq. (4.12).

B.2 Comparison of Scalar Operators

In the following we give the contributions of the scalar dimension-eight operators to the quartic gauge couplings [135]. First, the contributions of the dimension-eight operators of

the Eboli-basis eq. (4.12),

$$\mathcal{L}_{S,0}\big|_{4\text{ gauge}} = M_{\mathrm{W}}^4(\mathrm{W}^-\cdot\mathrm{W}^-)(\mathrm{W}^+\cdot\mathrm{W}^+) + M_{\mathrm{W}}^2M_{\mathrm{Z}}^2(\mathrm{W}^-\cdot\mathrm{Z})(\mathrm{W}^+\cdot\mathrm{Z}) + \frac{1}{4}M_{\mathrm{Z}}^4(\mathrm{Z}\cdot\mathrm{Z})^2, \tag{B.15}$$

$$\mathcal{L}_{S,1}\big|_{4\text{ gauge}} = M_{\mathrm{W}}^4(\mathrm{W}^+\cdot\mathrm{W}^-)(\mathrm{W}^+\cdot\mathrm{W}^-) + M_{\mathrm{W}}^2M_{\mathrm{Z}}^2(\mathrm{W}^-\cdot\mathrm{W}^+)(\mathrm{Z}\cdot\mathrm{Z}) + \frac{1}{4}M_{\mathrm{Z}}^4(\mathrm{Z}\cdot\mathrm{Z})^2, \tag{B.16}$$

$$\mathcal{L}_{S,2}\big|_{4\text{ gauge}} = M_{\mathrm{W}}^4(\mathrm{W}^+\cdot\mathrm{W}^-)(\mathrm{W}^+\cdot\mathrm{W}^-) + M_{\mathrm{W}}^2M_{\mathrm{Z}}^2(\mathrm{W}^-\cdot\mathrm{Z})(\mathrm{W}^+\cdot\mathrm{Z}) + \frac{1}{4}M_{\mathrm{Z}}^4(\mathrm{Z}\cdot\mathrm{Z})^2. \tag{B.17}$$

Second, the contributions of the dimension-8 operators eq. (B.11),

$$\begin{aligned}\mathcal{L}_{S,0}\big|_{4\text{ gauge}} &= \hat{F}_{S,0}\left(\frac{M_{\mathrm{W}}^4}{2}\left((\mathrm{W}^-\cdot\mathrm{W}^-)(\mathrm{W}^+\cdot\mathrm{W}^+) + (\mathrm{W}^-\cdot\mathrm{W}^+)(\mathrm{W}^-\cdot\mathrm{W}^+)\right)\right.\\ &\quad \left. + M_{\mathrm{W}}^2M_{\mathrm{Z}}^2(\mathrm{W}^-\cdot\mathrm{Z})(\mathrm{W}^+\cdot\mathrm{Z}) + \frac{1}{2}M_{\mathrm{Z}}^4(\mathrm{Z}\cdot\mathrm{Z})^2\right), \qquad \text{(B.18)}\\ \mathcal{L}_{S,0}\big|_{4\text{ gauge}} &= \hat{F}_{S,1}\left(M_{\mathrm{W}}^4(\mathrm{W}^-\cdot\mathrm{W}^-)(\mathrm{W}^+\cdot\mathrm{W}^+) + M_{\mathrm{W}}^2M_{\mathrm{Z}}^2(\mathrm{W}^-\cdot\mathrm{W}^+)(\mathrm{Z}\cdot\mathrm{Z})\right. \qquad \text{(B.19)}\\ &\quad \left. + \frac{1}{4}M_{\mathrm{Z}}^4(\mathrm{Z}\cdot\mathrm{Z})^2\right). \qquad \text{(B.20)}\end{aligned}$$

Additionally, the custodial-symmetry breaking operator $\mathcal{L}_7$ [125] gives contributions to,

$$\mathcal{L}_6 = \alpha_6\left(4\frac{M_{\mathrm{W}}^2M_{\mathrm{Z}}^2}{v^4}(\mathrm{W}^+\cdot\mathrm{Z})(\mathrm{W}^-\cdot\mathrm{Z}) + 2M_{\mathrm{Z}}^4(\mathrm{Z}\cdot\mathrm{Z})^2\right). \tag{B.21}$$

For the case of $F_{S,0} \neq F_{S,1}$, the operator coefficients for $\mathcal{L}_{S,0/1/2}$ and $\mathcal{L}_{4/5/6}$ can be related by,

$$\alpha_4 = \frac{v^4}{296}\frac{23F_{S,0} + 14F_{S,2}}{\Lambda^4}, \tag{B.22}$$

$$\alpha_5 = \frac{v^4}{592}\frac{-11F_{S,0} + 37F_{S,1} + 11F_{S,2}}{\Lambda^4}, \tag{B.23}$$

$$\alpha_6 = \frac{7v^4}{148}\frac{-F_{S,0} + F_{S,2}}{\Lambda^4}. \tag{B.24}$$

In the limit of the custodical symmetry, when $F_{S,0} = F_{S,2}$, we arrive at

$$\alpha_4 = \frac{v^4}{16}\hat{F}_{S,0} = \frac{v^4}{16}\frac{F_{S,0} + F_{S,2}}{\Lambda^4}, \tag{B.25}$$

$$\alpha_5 = \frac{v^4}{16}\hat{F}_{S,1} = \frac{v^4}{16}\frac{F_{S,1}}{\Lambda^4}, \tag{B.26}$$

which is the relation mentioned in section 4.1.

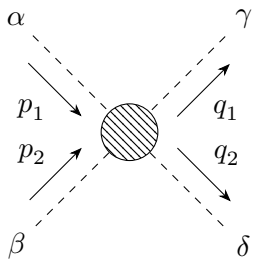


Figure B.1: $2 \to 2$ scattering. Definition of the incoming momenta p_1, p_2 and outgoing momenta q_1, q_2.

B.3 Mandelstam Variables

For $2 \to 2$ scattering process with incoming momenta p_1, p_2 and outgoing momenta q_1, q_2, the Mandelstam variables [134] are defined as

$$s = (p_1 + p_2)^2 = (q_1 + q_2)^2, \tag{B.27}$$
$$t = (p_1 - q_1)^2 = (p_2 - q_2)^2, \tag{B.28}$$
$$u = (p_1 - q_2)^2 = (p_2 - q_1)^2, \tag{B.29}$$
$$s + t + u = m_{p_1}^2 + m_{p_2}^2 + m_{q_1}^2 + m_{q_2}^2. \tag{B.30}$$

We use the crossing symmetry relations for $2 \to 2$ scattering:

$$\left.\begin{aligned}(p_1, \alpha), (p_2, \beta) &\to (q_1, \gamma), (q_2, \delta), \\ (-q_1, \gamma), (-q_2, \delta) &\to (-p_1, \alpha)(-p_2, \beta)\end{aligned}\right\} \quad \text{s-channel,} \tag{B.31}$$

$$\left.\begin{aligned}(p_1, \alpha), (-q_1, \gamma) &\to (q_2, \delta)(-p_2, \beta), \\ (p_2, \beta), (-q_2, \delta) &\to (q_1, \gamma)(-p_1, \alpha)\end{aligned}\right\} \quad \text{t-channel,} \tag{B.32}$$

$$\left.\begin{aligned}(p_1, \alpha), (-q_2, \delta) &\to (q_1, \gamma)(-p_2, \beta), \\ (p_2, \beta), (-q_1, \gamma) &\to (q_2, \delta)(-p_1, \alpha)\end{aligned}\right\} \quad \text{u-channel.} \tag{B.33}$$

The complete amplitude decomposes in three independent, invariant functions in s, t, u

$$\mathcal{A}_{\alpha\beta\gamma\delta}(s,t,u) = A(s,t,u)\delta_{\alpha\beta}\delta_{\gamma\delta} + B(s,t,u)\delta_{\alpha\gamma}\delta_{\beta\delta} + C(s,t,u)\delta_{\alpha\delta}\delta_{\gamma\beta}. \tag{B.34}$$

Every amplitude has to respect the crossing symmetry, which lets us relate the amplitudes A, B and C to each other

$$B(s,t,u) = A(t,s,u), \tag{B.35}$$
$$C(s,t,u) = A(u,s,t). \tag{B.36}$$

We can utilize the massless limit of the Mandelstam variables, $s = -(t+u)$, to reexpress,

$$\frac{t-u}{s} = \frac{t^2 - u^2}{s^2}, \tag{B.37}$$
$$\left(\frac{t-u}{s}\right)^2 = 4\frac{t^2}{s^2} + 2\frac{u^2}{s^2} + \frac{s^2}{s^2}, \tag{B.38}$$

and we can therefore write all counter-terms in section 4.3 explicitly in s^2, t^2 and u^2. In the massless limit, the Mandelstam variables are connected to the $2 \to 2$ scattering momenta by

$$(p_1 \cdot p_2)(q_1 \cdot q_2) = \frac{s^2}{4}, \tag{B.39}$$

$$(p_1 \cdot q_1)(p_2 \cdot q_2) = \frac{t^2}{4}, \tag{B.40}$$

$$(p_1 \cdot q_2)(p_2 \cdot q_1) = \frac{u^2}{4}. \tag{B.41}$$

The form factors are then extracted from the isospin amplitudes eqs. (4.97) and (4.98) by substituting

$$\frac{s^2}{s^2} \to \frac{4}{s^2} g_{\alpha\beta} g_{\gamma\delta}, \tag{B.42}$$

$$\frac{t^2}{s^2} \to \frac{4}{s^2} g_{\alpha\gamma} g_{\beta\delta}, \tag{B.43}$$

$$\frac{u^2}{s^2} \to \frac{4}{s^2} g_{\alpha\delta} g_{\beta\gamma}. \tag{B.44}$$

B.4 Isospin Algebra

We use the isospin formalism for spin-1 particles. The three vector bosons W^+, W^- and Z form an isospin triplet, equivalent to the Goldstone Boson triplet in [130],

$$|\mathrm{W}^+\rangle = -\,|1,+1\rangle\,, \tag{B.45}$$

$$|\mathrm{Z}\rangle = \quad |1,0\rangle\,, \tag{B.46}$$

$$|\mathrm{W}^-\rangle = \quad |1,-1\rangle\,. \tag{B.47}$$

Then, we can rewrite the amplitude for $p_1 p_2 \to p_3 p_4$ scattering using Glebsch–Gordon decomposition, $1 \otimes 1 = \mathbf{3} + \mathbf{2} + \mathbf{1}$, in terms of isospin amplitudes $\langle j_1, j_2; J, M_J | T | j_3, j_4; K, M_K \rangle$

$$\begin{aligned}\mathcal{A}\left(p_1 p_2 \to p_3 p_4\right) &= \langle j_1, m_1; j_2, m_2 | T | j_3, m_3; j_4, m_4 \rangle \\ &= \sum_{\substack{J, M_J, \\ K, M_K}} \langle j_1, m_1; j_2, m_2 | j_1, j_2; J, M_J \rangle \langle j_3, j_4; K, M_K | j_3, m_3; j_4, m_4 \rangle \\ &\qquad \times \langle j_1, j_2; J, M_J | T | j_3, j_4; K, M_K \rangle\,.\end{aligned} \tag{B.48}$$

With the relations from appendix B.3, we can express all amplitudes by a single master amplitude. In our case, we choose the master amplitude to be

$$\mathcal{A}\left(\mathrm{W}^+\mathrm{W}^- \to \mathrm{ZZ}\right) = A(s,t,u), \tag{B.49}$$

where we matched the case $M = 0$ to a amplitude A.

As an example, we take at closer look at $\mathrm{W}^+\mathrm{W}^- \to \mathrm{ZZ}$ scattering,

$$\begin{aligned}\mathcal{A}\left(\mathrm{W}^+\mathrm{W}^- \to \mathrm{ZZ}\right) &= -\,\langle 1,1;1,-1|T|1,0;1,0\rangle \\ &= \sum_{\substack{J, M_J, \\ K, M_K}} \langle 1,1;1,-1|1,1;J,M_J\rangle \langle 1,1;K,M_K|1,1;0,0\rangle \langle 1,1;J,M_J|T|1,1;K,M_K\rangle\,.\end{aligned} \tag{B.50}$$

Letting the scattering operator T act on $|j_1, j_2; J, M\rangle$ and using orthogonality between isospin states[3] we arrive at

$$\begin{aligned}
\mathcal{A}\left(\mathrm{W^+W^-} \to \mathrm{ZZ}\right) &= \sum_{J,M} \langle 1,1;1,-1|1,1;J,M\rangle \langle 1,1;J,M|1,1;0,0\rangle \mathcal{A}_{J,M} \\
&= \langle 1,1;1,-1|1,1;J=0,M=0\rangle \langle 1,0;1,0|1,1;J=0,M=0\rangle \mathcal{A}_{J=0,M=0} \\
&+ \langle 1,1;1,-1|1,1;J=1,M=0\rangle \langle 1,0;1,0|1,1;J=1,M=0\rangle \mathcal{A}_{J=1,M=0} \\
&+ \langle 1,1;1,-1|1,1;J=2,M=0\rangle \langle 1,0;1,0|1,1;J=2,M=0\rangle \mathcal{A}_{J=2,M=0} \\
&= \sqrt{\frac{1}{3}} \cdot \left(-\frac{1}{3}\right) \mathcal{A}_0 + \sqrt{\frac{1}{2}} \cdot 0 \mathcal{A}_1 + \sqrt{\frac{1}{6}} \cdot \sqrt{\frac{2}{3}} \mathcal{A}_2 = \frac{1}{3}\mathcal{A}_2 - \frac{1}{3}\mathcal{A}_0,
\end{aligned} \tag{B.51}$$

where we used that contributions from the doublet or singlet representation do not contribute. We add the Higgs as complex correction to the Z with the replacement $\mathrm{Z} \to \mathrm{Z} + \mathrm{i}h$

Following the above definitions and using the Condon-Shortley conventions on the phases, we can write down the complete basis, which is given by

$$|2, \pm 2\rangle = |\mathrm{W^\pm W^\pm}\rangle\,, \tag{B.52}$$

$$|2, \pm 1\rangle = \frac{1}{\sqrt{2}} \left(|\mathrm{ZW^\pm}\rangle + |\mathrm{W^\pm Z}\rangle\right), \tag{B.53}$$

$$|2, \quad 0\rangle = \frac{1}{\sqrt{6}} \left(|\mathrm{W^+W^-}\rangle + 2\,|\mathrm{ZZ}\rangle + |\mathrm{W^+W^-}\rangle\right), \tag{B.54}$$

$$|1, \pm 1\rangle = \frac{1}{\sqrt{2}} \left(|\mathrm{W^\pm Z}\rangle - |\mathrm{ZW^\pm}\rangle\right), \tag{B.55}$$

$$|1, \quad 0\rangle = \frac{1}{\sqrt{2}} \left(|\mathrm{W^+W^-}\rangle - |\mathrm{W^-W^+}\rangle\right), \tag{B.56}$$

$$|0, \quad 0\rangle = \frac{1}{\sqrt{3}} \left(|\mathrm{W^+W^-}\rangle - |\mathrm{ZZ}\rangle + |\mathrm{W^-W^+}\rangle\right). \tag{B.57}$$

The inversion of the basis reads,

$$|\mathrm{W^\pm W^\pm}\rangle = |2, \pm 2\rangle\,, \tag{B.58}$$

$$|\mathrm{W^\mp W^\pm}\rangle = \frac{1}{\sqrt{6}}\,|2,0\rangle \mp \frac{1}{\sqrt{2}}\,|1,0\rangle + \frac{1}{\sqrt{3}}\,|0,0\rangle\,, \tag{B.59}$$

$$|\mathrm{W^\pm Z}\rangle = \frac{1}{\sqrt{2}} \left(|2,1\rangle \pm |1,\pm 1\rangle\right), \tag{B.60}$$

$$|\mathrm{ZW^\pm}\rangle = \frac{1}{\sqrt{2}} \left(|2,1\rangle \mp |1,\pm 1\rangle\right), \tag{B.61}$$

$$|\mathrm{ZZ}\rangle = \frac{2}{\sqrt{6}}\,|2,0\rangle - \frac{1}{\sqrt{3}}\,|0,0\rangle\,. \tag{B.62}$$

[3] $\langle j_1, j_2; J, M_J | j_3, j_4; K, M_K\rangle = \delta_{j_1 j_3} \delta_{j_3 j_4} \delta_{JK} \delta_{M_J M_K}$.

B.5 Wigner D-functions

The Wigner D-functions $d^j_{\lambda,\lambda'}$ for $j = \{0, 1, 2\}$ are given by [24],

$$d^0_{0,0} = 1, \tag{B.63}$$

$$d^1_{0,0} = \cos\theta, \qquad d^1_{1,1} = \frac{1+\cos\theta}{2}, \qquad d^1_{1,-1} = \frac{1-\cos\theta}{2}, \tag{B.64}$$

$$d^2_{0,0} = \left(\frac{3}{2}\cos^2\theta - \frac{1}{2}\right), \tag{B.65}$$

$$d^2_{1,1} = \frac{1+\cos\theta}{2}(2\cos\theta - 1), \quad d^2_{1,0} = -\sqrt{\frac{3}{2}}\sin\theta\cos\theta, \quad d^2_{1,-1} = \frac{1-\cos\theta}{2}(2\cos\theta + 1), \tag{B.66}$$

$$d^2_{2,2} = \left(\frac{1+\cos\theta}{2}\right)^2, \qquad d^2_{2,1} = -\frac{1+\cos\theta}{2}\sin\theta, \quad d^2_{2,0} = \frac{6}{4}\sin^2\theta, \tag{B.67}$$

$$d^2_{2,-1} = -\frac{1-\cos\theta}{2}\sin\theta, \qquad d^2_{2,-2} = \left(\frac{1-\cos\theta}{2}\right)^2. \tag{B.68}$$

B.6 Sindarin File

We show an example Sindarin file for the production runs of WHIZARD for our results in section 4.6 and appendix B.7. In particular, we choose the Sindarin file for the process $pp \to HHjj$, see listing B.1. The example exhibits a wide range of features of the Sindarin scripting language: model selection, process definition, beams and PDF, integration and RNG methods, scan loop with run ids as well as formatted output.

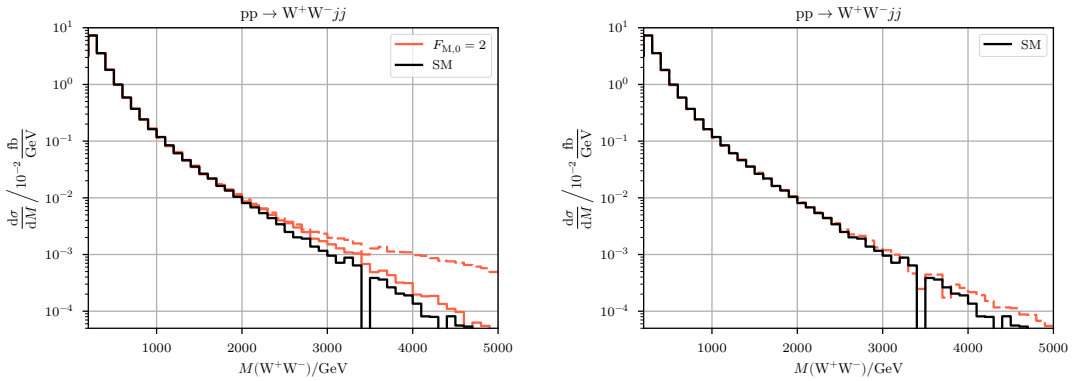
```
! Set RNG and seed.
$integration_method = "vamp2"
$rng_method = "rng_stream"
seed = 1961991
model = SSC_2
alias q = u:d:s:c:U:D:S:C:g
process signal = q,q => H,H,q,q {$restrictions =
 ↪  "!Rsigma:Rphi0:Rphis0:Rphi+:Rphi++:Rphi-:Rphi--
 ↪  :Rf:Rt0:Rts0:Rt+:Rt++:Rt-:Rt--"}
compile
ms = 0
mc = 0
relative_error_goal = 0.02
beams = p, p => pdf_builtin
fs0 = 0
fs1 = 0
ft0 = 0
ft1 = 0
ft2 = 0
fm0 = 2
fm1 = 0
eft_h = 1
fkm = 0
sqrts = 14 TeV
$out_file = "ppHHjj_fm0_0.dat"
open_out ("ppHHjj_fm0_0.dat")
scan real M_cut = (0 GeV => 8 TeV /+ 100 GeV) {
  $run_id = sprintf "%e" (M_cut)
  cuts = all M > 500 GeV [q,q]
     and all Pt > 20 GeV [q]
     and all -4.5 < Eta < 4.5 [q]
     and all abs(Rap) > 2.4 [q,q]
     and all M_cut < M < (M_cut + 100 GeV) [H,H]
  integrate (signal) { iterations = 20:300000:"gw", 5:150000 }
  show(integral(signal))
  printf " %f %E %E " (((M_cut+(50 GeV))), integral(signal)/(100 GeV),
   ↪  error(signal)/(100 GeV))
}
close_out ("ppHHjj_fm0_0.dat")
```

Listing B.1: Example Sindarin file for the process pp $\to$ HHjj.

B.7 Additional Numerical Results

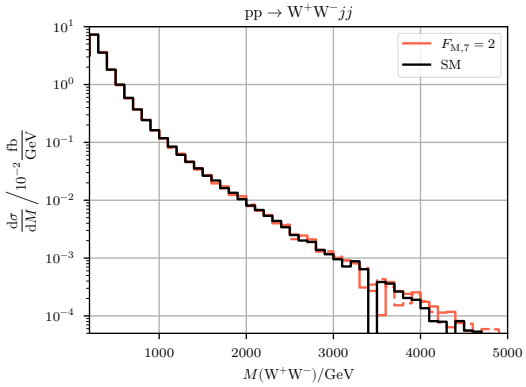
In this section, we display results for the invariant-mass distribution of the LHC processes $pp \to W^+W^-jj, ZZjj$, and W^+Zjj in figs. B.2 to B.6 which supplement the results for the W^+W^+ and HH channels in section 4.6. For all processes, we present the SM distribution together with the corresponding distribution of the continuum simplified model, one free parameter varied at a time, with a universal parameter value of $2\,\mathrm{TeV}^{-4}$.

Longitudinal-Transverse Mixed Operators



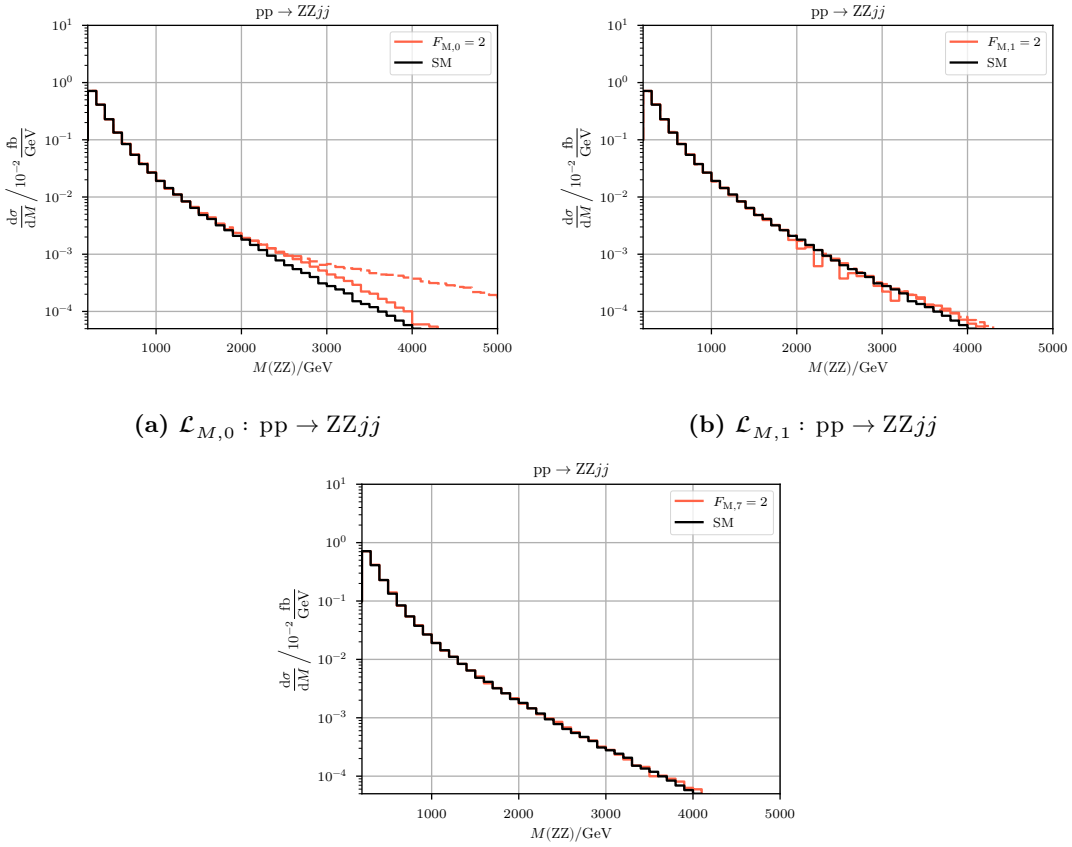
(a) $\mathcal{L}_{M,0}$: $pp \to W^+W^-jj$

(b) $\mathcal{L}_{M,1}$: $pp \to W^+W^-jj$



(c) $\mathcal{L}_{M,7}$: $pp \to W^+W^-jj$

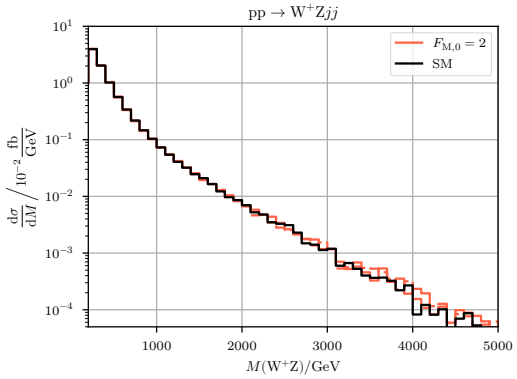
Figure B.2: Differential cross section for the diboson invariant mass for the process $pp \to W^+W^-jj$. The solid black line shows the SM differential cross section and the redish lines the cross sections with anomalous couplings $F_{M,i} = \{2, 10, 50\}\mathrm{TeV}^{-4}$, respectively. Solid: unitarized; dashed: naive result. Cuts like in eq. (4.116).



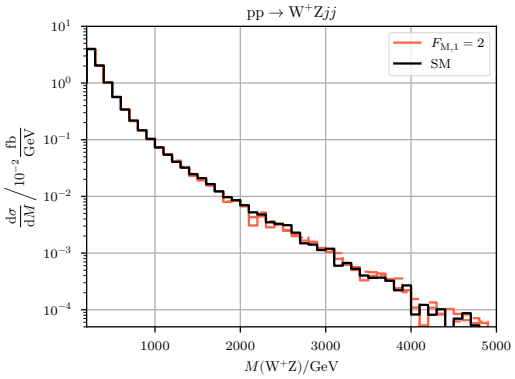
(a) $\mathcal{L}_{M,0}$: pp → ZZjj **(b)** $\mathcal{L}_{M,1}$: pp → ZZjj

(c) $\mathcal{L}_{M,7}$: pp → ZZjj

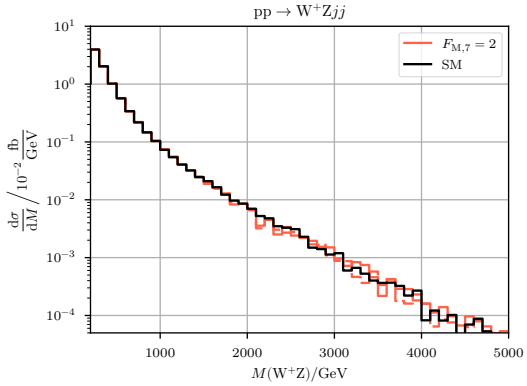
Figure B.3: Differential cross section for the diboson invariant mass for the process pp → ZZjj. The solid black line shows the SM differential cross section and the redish lines the cross sections with anomalous couplings $F_{M,i} = \{2, 10, 50\}\text{TeV}^{-4}$, respectively. Solid: unitarized; dashed: naive result. Cuts like in eq. (4.116).



(a) $\mathcal{L}_{M,0}$: pp → W⁺Zjj



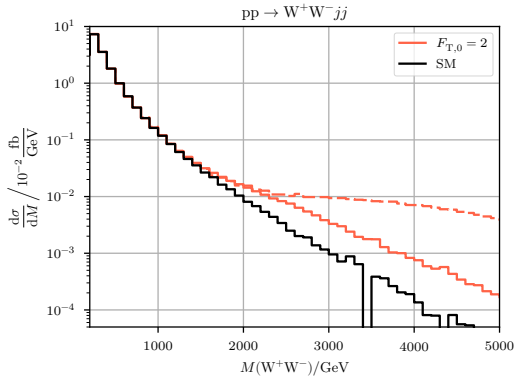
(b) $\mathcal{L}_{M,1}$: pp → W⁺Zjj



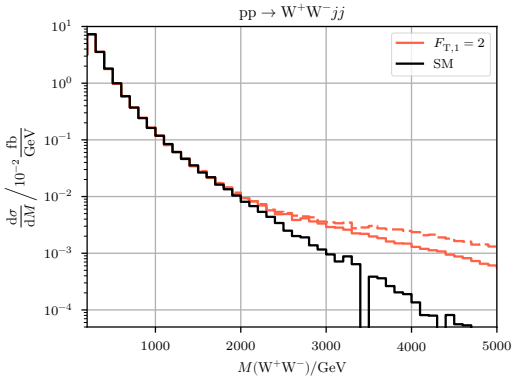
(c) $\mathcal{L}_{M,7}$: pp → W⁺Zjj

Figure B.4: Differential cross section for the diboson invariant mass for the process pp → W⁺Zjj. The solid black line shows the SM differential cross section and the redish lines the cross sections with anomalous couplings $F_{M,i} = \{2, 10, 50\}\text{TeV}^{-4}$, respectively. Solid: unitarized; dashed: naive result. Cuts like in eq. (4.116).

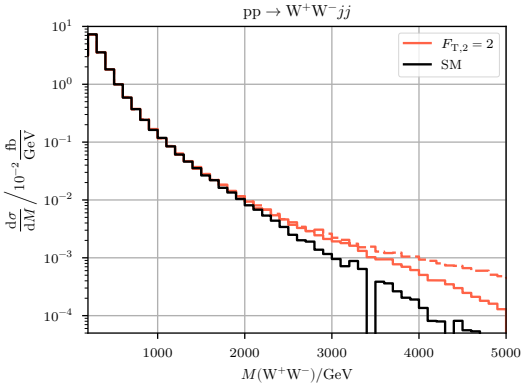
Transverse Operators



(a) $\mathcal{L}_{M,0}$: pp → W^+W^-jj

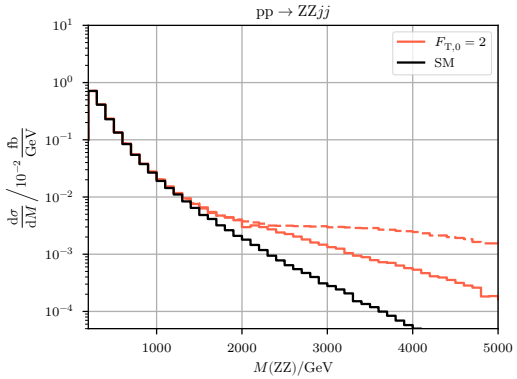


(b) $\mathcal{L}_{M,1}$: pp → W^+W^-jj

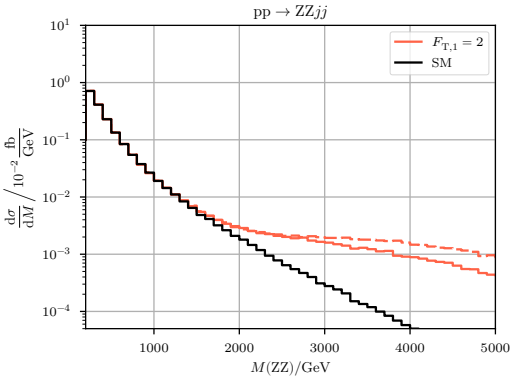


(c) $\mathcal{L}_{M,7}$: pp → W^+W^-jj

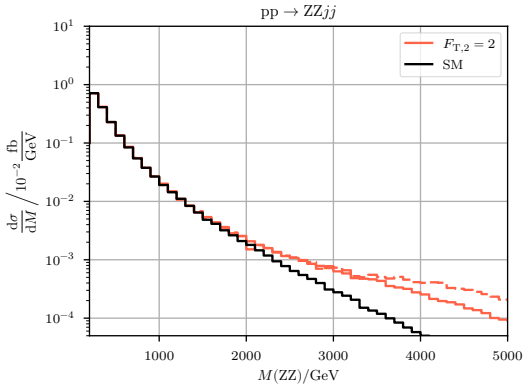
Figure B.5: Differential cross section for the diboson invariant mass for the process pp → W^+W^-jj. The solid black line shows the SM differential cross section and the redish lines the cross sections with anomalous couplings $F_{M,i} = \{2, 10, 50\}\text{TeV}^{-4}$, respectively. Solid: unitarized; dashed: naive result. Cuts like in eq. (4.116).



(a) $\mathcal{L}_{M,0}$: pp → ZZjj



(b) $\mathcal{L}_{M,1}$: pp → ZZjj



(c) $\mathcal{L}_{M,7}$: pp → ZZjj

Figure B.6: Differential cross section for the diboson invariant mass for the process pp → ZZjj. The solid black line shows the SM differential cross section and the redish lines the cross sections with anomalous couplings $F_{M,i} = \{2, 10, 50\}\text{TeV}^{-4}$, respectively. Solid: unitarized; dashed: naive result. Cuts like in eq. (4.116).

Bibliography

[1] Georges Aad et al. "Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC". In: *Phys. Lett.* B716 (2012), pp. 1–29. DOI: 10.1016/j.physletb.2012.08.020. arXiv: 1207.7214 [hep-ex].

[2] Serguei Chatrchyan et al. "Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC". In: *Phys. Lett.* B716 (2012), pp. 30–61. DOI: 10.1016/j.physletb.2012.08.021. arXiv: 1207.7235 [hep-ex].

[3] Andy Buckley et al. "General-purpose event generators for LHC physics". In: *Phys. Rept.* 504 (2011), pp. 145–233. DOI: 10.1016/j.physrep.2011.03.005. arXiv: 1101.2599 [hep-ph].

[4] *Physics Event Generator Computing Workshop*. CERN, Nov. 26–28, 2018.

[5] Wolfgang Kilian, Thorsten Ohl, and Jürgen Reuter. "WHIZARD: Simulating Multi-Particle Processes at LHC and ILC". In: *Eur. Phys. J.* C71 (2011), p. 1742. DOI: 10.1140/epjc/s10052-011-1742-y. arXiv: 0708.4233 [hep-ph].

[6] Alessandro Ballestrero et al. "Precise predictions for same-sign W-boson scattering at the LHC". In: *Eur. Phys. J.* C78.8 (2018), p. 671. DOI: 10.1140/epjc/s10052-018-6136-y. arXiv: 1803.07943 [hep-ph].

[7] Simon Braß et al. "Transversal Modes and Higgs Bosons in Electroweak Vector-Boson Scattering at the LHC". In: (2018). arXiv: 1807.02512 [hep-ph].

[8] W. Buchmuller and D. Wyler. "Effective Lagrangian Analysis of New Interactions and Flavor Conservation". In: *Nucl. Phys.* B268 (1986), pp. 621–653. DOI: 10.1016/0550-3213(86)90262-2.

[9] Kaoru Hagiwara et al. "Low-energy constraints on electroweak three gauge boson couplings". In: *Phys. Lett.* B283 (1992), pp. 353–359. DOI: 10.1016/0370-2693(92)90031-X.

[10] Kaoru Hagiwara et al. "Low-energy effects of new interactions in the electroweak boson sector". In: *Phys. Rev.* D48 (1993), pp. 2182–2203. DOI: 10.1103/PhysRevD.48.2182.

[11] B. Grzadkowski et al. "Dimension-Six Terms in the Standard Model Lagrangian". In: *JHEP* 10 (2010), p. 085. DOI: 10.1007/JHEP10(2010)085. arXiv: 1008.4884 [hep-ph].

[12] Daniel R. Green, Patrick Meade, and Marc-Andre Pleier. "Multiboson interactions at the LHC". In: *Rev. Mod. Phys.* 89.3 (2017), p. 035008. DOI: 10.1103/RevModPhys.89.035008. arXiv: 1610.07572 [hep-ex].

[13] Morad Aaboud et al. "Search for anomalous electroweak production of WW/WZ in association with a high-mass dijet system in pp collisions at $\sqrt{s} = 8$ TeV with the ATLAS detector". In: *Phys. Rev.* D95.3 (2017), p. 032001. DOI: 10.1103/PhysRevD.95.032001. arXiv: 1609.05122 [hep-ex].

[14] Marco Sekulla et al. "Effective Field Theory and Unitarity in Vector Boson Scattering". In: *PoS* LHCP2016 (2016), p. 052. DOI: 10.22323/1.276.0052. arXiv: 1610.04131 [hep-ph].

[15] Albert M. Sirunyan et al. "Observation of electroweak production of same-sign W boson pairs in the two jet and two same-sign lepton final state in proton-proton collisions at $\sqrt{s} = 13$ TeV". In: *Phys. Rev. Lett.* 120.8 (2018), p. 081801. DOI: 10.1103/PhysRevLett.120.081801. arXiv: 1709.05822 [hep-ex].

[16] Albert M. Sirunyan et al. "Measurement of vector boson scattering and constraints on anomalous quartic couplings from events with four leptons and two jets in proton–proton collisions at $\sqrt{s} = 13$ TeV". In: *Phys. Lett.* B774 (2017), pp. 682–705. DOI: 10.1016/j.physletb.2017.10.020. arXiv: 1708.02812 [hep-ex].

[17] Wolfgang Kilian et al. "High-Energy Vector Boson Scattering after the Higgs Discovery". In: *Phys. Rev.* D91 (2015), p. 096007. DOI: 10.1103/PhysRevD.91.096007. arXiv: 1408.6207 [hep-ph].

[18] Jan Kalinowski et al. "Same-sign WW scattering at the LHC: can we discover BSM effects before discovering new states?" In: *Eur. Phys. J.* C78.5 (2018), p. 403. DOI: 10.1140/epjc/s10052-018-5885-y. arXiv: 1802.02366 [hep-ph].

[19] Wolfgang Kilian et al. "Resonances at the LHC beyond the Higgs boson: The scalar/tensor case". In: *Phys. Rev.* D93.3 (2016), p. 036004. DOI: 10.1103/PhysRevD.93.036004. arXiv: 1511.00022 [hep-ph].

[20] Matthew D. Schwartz. *Quantum Field Theory and the Standard Model*. Cambridge University Press, 2014. ISBN: 1107034736, 9781107034730. URL: http://www.cambridge.org/us/academic/subjects/physics/theoretical-physics-and-mathematical-physics/quantum-field-theory-and-standard-model.

[21] C. Itzykson and J. B. Zuber. *Quantum Field Theory*. International Series In Pure and Applied Physics. New York: McGraw-Hill, 1980. ISBN: 9780486445687, 0486445682. URL: http://dx.doi.org/10.1063/1.2916419.

[22] Michael E. Peskin and Daniel V. Schroeder. *An Introduction to quantum field theory*. Reading, USA: Addison-Wesley, 1995. ISBN: 9780201503975, 0201503972. URL: http://www.slac.stanford.edu/~mpeskin/QFT.html.

[23] Mark Thomson. *Modern particle physics*. New York: Cambridge University Press, 2013. ISBN: 9781107034266. URL: http://www-spires.fnal.gov/spires/find/books/www?cl=QC793.2.T46::2013.

[24] M. Tanabashi et al. "Review of Particle Physics". In: *Phys. Rev. D* 98 (3 Aug. 2018), p. 030001. DOI: 10.1103/PhysRevD.98.030001. URL: https://link.aps.org/doi/10.1103/PhysRevD.98.030001.

[25] Georges Aad et al. "Evidence for the production of three massive vector bosons with the ATLAS detector". In: (2019). arXiv: 1903.10415 [hep-ex].

[26] Simon Plätzer. "RAMBO on diet". In: (2013). arXiv: 1308.2922 [hep-ph].

[27] R. Kleiss, W. James Stirling, and S. D. Ellis. "A New Monte Carlo Treatment of Multiparticle Phase Space at High-energies". In: *Comput. Phys. Commun.* 40 (1986), p. 359. DOI: 10.1016/0010-4655(86)90119-0.

[28] Keijo Kajantie and Eero Byckling. *Particle kinematics*. Wiley-Interscience, 1973.

[29] Torbjörn Sjöstrand et al. "An Introduction to PYTHIA 8.2". In: *Comput. Phys. Commun.* 191 (2015), pp. 159–177. DOI: 10.1016/j.cpc.2015.01.024. arXiv: 1410.3012 [hep-ph].

[30] Simon Braß, Wolfgang Kilian, and Jürgen Reuter. "Parallel Adaptive Monte Carlo Integration with the Event Generator WHIZARD". In: *Eur. Phys. J.* C (2019). arXiv: 1811.09711 [hep-ph]. In press.

[31] F. James. "Monte Carlo Theory and Practice". In: *Rept. Prog. Phys.* 43 (1980), p. 1145. DOI: 10.1088/0034-4885/43/9/002.

[32] E. Boos et al. "Generic user process interface for event generators". In: *Physics at TeV colliders. Proceedings, Euro Summer School, Les Houches, France, May 21-June 1, 2001*. 2001. arXiv: hep-ph/0109068 [hep-ph]. URL: http://lss.fnal.gov/archive/preprint/fermilab-conf-01-496-t.shtml.

[33] Wolfgang Kilian et al. "New Developments in WHIZARD Version 2.6". In: *International Workshop on Future Linear Collider (LCWS2017) Strasbourg, France, October 23-27, 2017*. 2018. arXiv: 1801.08034 [hep-ph].

[34] Mauro Moretti, Thorsten Ohl, and Jürgen Reuter. "O'Mega: An Optimizing matrix element generator". In: (2001), pp. 1981–2009. arXiv: hep-ph/0102195 [hep-ph].

[35] C. Weiss et al. "Automated NLO QCD Corrections with WHIZARD". In: *PoS* EPS-HEP2015 (2015), p. 466. DOI: 10.22323/1.234.0466. arXiv: 1510.02666 [hep-ph].

[36] Bijan Chokoufé Nejad et al. "Matching NLO QCD Corrections in WHIZARD with the POWHEG scheme". In: *PoS* EPS-HEP2015 (2015), p. 317. DOI: 10.22323/1.234.0317. arXiv: 1510.02739 [hep-ph].

[37] Jürgen Reuter et al. "Top Physics in WHIZARD". In: *Proceedings, International Workshop on Future Linear Colliders (LCWS15): Whistler, B.C., Canada, November 02-06, 2015*. 2016. arXiv: 1602.08035 [hep-ph].

[38] Christian Weiss. "Top quark physics as a prime application of automated higher-order corrections." PhD thesis. Hamburg: DESY, 2017. DOI: 10.3204/PUBDB-2017-07541.

[39] Fabian Bach et al. "Fully-differential Top-Pair Production at a Lepton Collider: From Threshold to Continuum". In: *JHEP* 03 (2018), p. 184. DOI: 10.1007/JHEP03(2018)184. arXiv: 1712.02220 [hep-ph].

[40] Jürgen Reuter et al. "QCD NLO with Powheg matching and top threshold matching in WHIZARD". In: *PoS* RADCOR2015 (2016), p. 088. DOI: 10.22323/1.235.0088. arXiv: 1601.02459 [hep-ph].

[41] Jürgen Reuter et al. "Automation of NLO processes and decays and POWHEG matching in WHIZARD". In: *J. Phys. Conf. Ser.* 762.1 (2016), p. 012059. DOI: 10.1088/1742-6596/762/1/012059. arXiv: 1602.06270 [hep-ph].

[42] Bijan Chokoufé Nejad. "Scrutinizing the Top Quark at Lepton Colliders with Higher Orders". PhD thesis. Hamburg: DESY, 2017. DOI: 10.3204/PUBDB-2017-07624.

[43] Andy Buckley et al. "LHAPDF6: parton density access in the LHC precision era". In: *Eur. Phys. J.* C75 (2015), p. 132. DOI: 10.1140/epjc/s10052-015-3318-8. arXiv: 1412.7420 [hep-ph].

[44] J. Reuter, W. Kilian, and M. Sekulla. "Simplified Models for Vector Boson Scattering at ILC and CLIC". In: *International Workshop on Future Linear Colliders (LCWS13) Tokyo, Japan, November 11-15, 2013*. 2014. arXiv: 1403.7392 [hep-ph].

[45] W. Kilian et al. "WHIZARD 2.2 for Linear Colliders". In: *International Workshop on Future Linear Colliders (LCWS13) Tokyo, Japan, November 11-15, 2013*. 2014. arXiv: 1403.7433 [hep-ph].

[46] H. Abramowicz et al. "Top-Quark Physics at the CLIC Electron-Positron Linear Collider". In: (2018). arXiv: 1807.02441 [hep-ex].

[47] J. de Blas et al. "The CLIC Potential for New Physics". In: (2018). DOI: 10.23731/CYRM-2018-003. arXiv: 1812.02093 [hep-ph].

[48] P. Roloff et al. "The Compact Linear e^+e^- Collider (CLIC): Physics Potential". In: (2018). arXiv: 1812.07986 [hep-ex].

[49] Christian Fleper et al. "Scattering of W and Z Bosons at High-Energy Lepton Colliders". In: *Eur. Phys. J.* C77.2 (2017), p. 120. DOI: 10.1140/epjc/s10052-017-4656-5. arXiv: 1607.03030 [hep-ph].

[50] M. Bahr et al. "Herwig++ Physics and Manual". In: *Eur. Phys. J.* C58 (2008), pp. 639–707. DOI: 10.1140/epjc/s10052-008-0798-9. arXiv: 0803.0883 [hep-ph].

[51] Johannes Bellm et al. "Herwig 7.0/Herwig++ 3.0 release note". In: *Eur. Phys. J.* C76.4 (2016), p. 196. DOI: 10.1140/epjc/s10052-016-4018-8. arXiv: 1512.01178 [hep-ph].

[52] J. Alwall et al. "The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations". In: *JHEP* 07 (2014), p. 079. DOI: 10.1007/JHEP07(2014)079. arXiv: 1405.0301 [hep-ph].

[53] T. Gleisberg et al. "Event generation with SHERPA 1.1". In: *JHEP* 02 (2009), p. 007. DOI: 10.1088/1126-6708/2009/02/007. arXiv: 0811.4622 [hep-ph].

[54] W. Kilian, D. Rainwater, and J. Reuter. "Pseudo-axions in little Higgs models". In: *Phys. Rev.* D71 (2005), p. 015008. DOI: 10.1103/PhysRevD.71.015008. arXiv: hep-ph/0411213 [hep-ph].

[55] Kaoru Hagiwara et al. "Supersymmetry simulations with off-shell effects for CERN LHC and ILC". In: *Phys. Rev.* D73 (2006), p. 055005. DOI: 10.1103/PhysRevD.73.055005. arXiv: hep-ph/0512260 [hep-ph].

[56] M. Beyer et al. "Determination of New Electroweak Parameters at the ILC - Sensitivity to New Physics". In: *Eur. Phys. J.* C48 (2006), pp. 353–388. DOI: 10.1140/epjc/s10052-006-0038-0. arXiv: hep-ph/0604048 [hep-ph].

[57] Ana Alboteanu, Wolfgang Kilian, and Jürgen Reuter. "Resonances and Unitarity in Weak Boson Scattering at the LHC". In: *JHEP* 11 (2008), p. 010. DOI: 10.1088/1126-6708/2008/11/010. arXiv: 0806.4145 [hep-ph].

[58] So Young Shim. "Beyond the SM searches at the LHC: Little Higgs models and Effective Field Theory". PhD thesis. Hamburg: DESY, 2018. DOI: 10.3204/PUBDB-2018-01793.

[59] Jürgen Reuter, Wolfgang Kilian, and Thorsten Ohl. *The WHIZARD Event Generator*. URL: http://whizard.hepforge.org/.

[60] Ronald Kleiss and Roberto Pittau. "Weight optimization in multichannel Monte Carlo". In: *Comput. Phys. Commun.* 83 (1994), pp. 141–146. DOI: 10.1016/0010-4655(94)90043-4. arXiv: hep-ph/9405257 [hep-ph].

[61] G. Peter Lepage. *VEGAS - an adaptive multi-dimensional integration program.* Tech. rep. CLNS-447. Ithaca, NY: Cornell Univ. Lab. Nucl. Stud., Mar. 1980. URL: http://cds.cern.ch/record/123074.

[62] G. Peter Lepage. "A New Algorithm for Adaptive Multidimensional Integration". In: *J. Comput. Phys.* 27 (1978), p. 192. DOI: 10.1016/0021-9991(78)90004-9.

[63] Massimiliano Grazzini, Stefan Kallweit, and Marius Wiesemann. "Fully differential NNLO computations with MATRIX". In: *Eur. Phys. J.* C78.7 (2018), p. 537. DOI: 10.1140/epjc/s10052-018-5771-7. arXiv: 1711.06631 [hep-ph].

[64] John M. Campbell, R. Keith Ellis, and Walter T. Giele. "A Multi-Threaded Version of MCFM". In: *Eur. Phys. J.* C75.6 (2015), p. 246. DOI: 10.1140/epjc/s10052-015-3461-2. arXiv: 1503.06182 [physics.comp-ph].

[65] Thorsten Ohl. "Vegas revisited: Adaptive Monte Carlo integration beyond factorization". In: *Comput. Phys. Commun.* 120 (1999), pp. 13–19. DOI: 10.1016/S0010-4655(99)00209-X. arXiv: hep-ph/9806432 [hep-ph].

[66] Il'ya Meerovich Sobol'. "On the distribution of points in a cube and the approximate evaluation of integrals". In: *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki* 7.4 (1967), pp. 784–802.

[67] Josef Dick, Frances Y. Kuo, and Ian H. Sloan. "High-dimensional integration: the quasi-Monte Carlo way". In: *Acta Numerica* 22 (2013), pp. 133–288.

[68] Frances Y. Kuo and Dirk Nuyens. "A Practical Guide to Quasi-Monte Carlo Methods". In: (2016).

[69] Jürgen Reuter and Daniel Wiesler. "Distorted mass edges at LHC from supersymmetric leptoquarks". In: *Phys. Rev.* D84 (2011), p. 015012. DOI: 10.1103/PhysRevD.84.015012. arXiv: 1010.4215 [hep-ph].

[70] Niklas Pietsch et al. “Extracting Gluino Endpoints with Event Topology Patterns”. In: *JHEP* 07 (2012), p. 148. DOI: 10.1007/JHEP07(2012)148. arXiv: 1206.2146 [hep-ph].

[71] Jürgen Reuter and Daniel Wiesler. “A Fat Gluino in Disguise”. In: *Eur. Phys. J.* C73.3 (2013), p. 2355. DOI: 10.1140/epjc/s10052-013-2355-4. arXiv: 1212.5559 [hep-ph].

[72] Torbjörn Sjöstrand, Stephen Mrenna, and Peter Z. Skands. “PYTHIA 6.4 Physics and Manual”. In: *JHEP* 05 (2006), p. 026. DOI: 10.1088/1126-6708/2006/05/026. arXiv: hep-ph/0603175 [hep-ph].

[73] Stefano Frixione, Paolo Nason, and Carlo Oleari. “Matching NLO QCD computations with Parton Shower simulations: the POWHEG method”. In: *JHEP* 11 (2007), p. 070. DOI: 10.1088/1126-6708/2007/11/070. arXiv: 0709.2092 [hep-ph].

[74] Stefano Actis et al. “RECOLA: REcursive Computation of One-Loop Amplitudes”. In: *Comput. Phys. Commun.* 214 (2017), pp. 140–173. DOI: 10.1016/j.cpc.2017.01.004. arXiv: 1605.01090 [hep-ph].

[75] Ansgar Denner, Jean-Nicolas Lang, and Sandro Uccirati. “Recola2: REcursive Computation of One-Loop Amplitudes 2”. In: *Comput. Phys. Commun.* 224 (2018), pp. 346–361. DOI: 10.1016/j.cpc.2017.11.013. arXiv: 1711.07388 [hep-ph].

[76] Thorsten Ohl. “O’mega: An Optimizing matrix element generator”. In: *AIP Conf. Proc.* 583.1 (2002), pp. 173–175. DOI: 10.1063/1.1405295. arXiv: hep-ph/0011243 [hep-ph].

[77] Thorsten Ohl. “O’Mega & WHIZARD: Monte Carlo event generator generation for future colliders”. In: *AIP Conf. Proc.* 578.1 (2001), pp. 638–641. DOI: 10.1063/1.1394396. arXiv: hep-ph/0011287 [hep-ph].

[78] Bijan Chokoufé Nejad, Thorsten Ohl, and Jürgen Reuter. “Simple, parallel virtual machines for extreme computations”. In: *Comput. Phys. Commun.* 196 (2015), pp. 58–69. DOI: 10.1016/j.cpc.2015.05.015. arXiv: 1411.3834 [physics.comp-ph].

[79] J. Reuter, Bijan Chokoufé Nejad, and T. Ohl. “Making extreme computations possible with virtual machines”. In: *J. Phys. Conf. Ser.* 762.1 (2016), p. 012071. DOI: 10.1088/1742-6596/762/1/012071. arXiv: 1602.07242 [physics.comp-ph].

[80] Manuel Utsch. “Parametrisierung des Phasenraums im Monte-Carlo-Eventgenerator WHIZARD”. Master’s thesis. Unversität Siegen, Apr. 2018.

[81] David W. Walker et al. “MPI: A Standard Message Passing Interface”. In: *Supercomputer* 12 (1996), pp. 56–68.

[82] *MPI: A Message-Passing Interface Standard Version 3.1*. June 2015. URL: https://www.mpi-forum.org/docs/mpi-3.1/mpi31-report.pdf.

[83] Robit Chandra et al. “Parallel Programming in OpenMP”. In: *NY.: Morgan Kaufmann Publishers* (2008).

[84] *OpenMP Application Programming Interface - Version 5.0*. Nov. 2018. URL: https://www.openmp.org/wp-content/uploads/OpenMP-API-Specification-5.0.pdf.

[85] David R Butenhof. *Programming with POSIX threads*. Addison-Wesley Professional, 1997.

[86] Anton Shterenlikht et al. "Fortran 2008 Coarrays". In: *SIGPLAN Fortran Forum* 34.1 (Apr. 2015), pp. 10–30. ISSN: 1061-7264. DOI: 10.1145/2754942.2754944. URL: http://doi.acm.org/10.1145/2754942.2754944.

[87] Robert W. Numrich. *Parallel Programming with Co-arrays: Parallel Programming in Fortran*. Chapman and Hall/CRC, 2018.

[88] David Kirk et al. "NVIDIA CUDA software and GPU parallel computing architecture". In: *ISMM*. Vol. 7. 2007, pp. 103–104.

[89] Gene M. Amdahl. "Validity of the Single Processor Approach to Achieving Large Scale Computing Capabilities". In: *Proceedings of the April 18-20, 1967, Spring Joint Computer Conference*. AFIPS '67 (Spring). Atlantic City, New Jersey: ACM, 1967, pp. 483–485. DOI: 10.1145/1465482.1465560. URL: http://doi.acm.org/10.1145/1465482.1465560.

[90] John L. Gustafson. "Reevaluating Amdahl's Law". In: *Commun. ACM* 31.5 (May 1988), pp. 532–533. ISSN: 0001-0782. DOI: 10.1145/42411.42415. URL: http://doi.acm.org/10.1145/42411.42415.

[91] Pierre L'Ecuyer et al. "An object-oriented random-number package with many long streams and substreams". In: *Operations research* 50.6 (2002), pp. 1073–1075.

[92] Brian Gough. *GNU scientific library reference manual*. Network Theory Ltd., 2009.

[93] Melissa E. O'Neill. *PCG: A Family of Simple Fast Space-Efficient Statistically Good Algorithms for Random Number Generation*. Tech. rep. HMC-CS-2014-0905. Claremont, CA: Harvey Mudd College, Sept. 2014.

[94] Donald E. Knuth. *The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1997. ISBN: 0-201-89684-2.

[95] Pierre L'Ecuyer and Richard Simard. "Testu01: A C Library for Empirical Testing of Random Number Generators". In: *ACM Trans. Math. Softw.* 33.4 (Aug. 2007), 22:1–22:40. ISSN: 0098-3500. DOI: 10.1145/1268776.1268777. URL: http://doi.acm.org/10.1145/1268776.1268777.

[96] George Marsaglia. "The Marsaglia Random Number CDROM including the Diehard Battery of Tests of Randomness". In: (1995). URL: https://web.archive.org/web/20160125103112/http://stat.fsu.edu/pub/diehard/.

[97] A. Wald and J. Wolfowitz. "On a Test Whether Two Samples are from the Same Population". In: *Ann. Math. Statist.* 11.2 (June 1940), pp. 147–162. DOI: 10.1214/aoms/1177731909. URL: https://doi.org/10.1214/aoms/1177731909.

[98] Robert G. Brown, Dirk Eddelbüttel, and David Bauer. In: (2018). URL: http://webhome.phy.duke.edu/~rgb/General/dieharder.php.

[99] Martin Luscher. "A Portable high quality random number generator for lattice field theory simulations". In: *Comput. Phys. Commun.* 79 (1994), pp. 100–110. DOI: 10.1016/0010-4655(94)90232-1. arXiv: hep-lat/9309020 [hep-lat].

[100] Lev N. Shchur and Paolo Butera. "The RANLUX generator: Resonances in a random walk test". In: *Int. J. Mod. Phys.* C9 (1998), pp. 607–624. DOI: 10.1142/S0129183198000509. arXiv: hep-lat/9805017 [hep-lat].

[101] George Marsaglia. "RANDOM NUMBERS FALL MAINLY IN THE PLANES". In: *Proceedings of the National Academy of Sciences* 61.1 (1968), pp. 25–28. ISSN: 0027-8424. DOI: 10.1073/pnas.61.1.25. eprint: https://www.pnas.org/content/61/1/25.full.pdf. URL: https://www.pnas.org/content/61/1/25.

[102] G. Peter Lepage. "VEGAS: AN ADAPTIVE MULTIDIMENSIONAL INTEGRATION PROGRAM". In: (1980).

[103] Richard Kreckel. "Parallelization of adaptive MC integrators". In: *Computer Physics Communications* 106.3 (Nov. 1997), pp. 258–266. DOI: 10.1016/s0010-4655(97)00099-4. URL: https://doi.org/10.1016%2Fs0010-4655%2897%2900099-4.

[104] Rama Kishan Malladi. "Using Intel® VTune™ Performance Analyzer Events/Ratios & Optimizing Applications". In: *https://software.intel.com* (2009).

[105] Markus Geimer et al. "The SCALASCA performance toolset architecture". In: *Concurrency and Computation: Practice and Experience* 22 (Apr. 2010), pp. 702–719. DOI: 10.1002/cpe.1556.

[106] Jeffrey S. Vetter and Michael O. McCracken. "Statistical scalability analysis of communication operations in distributed applications". In: *ACM SIGPLAN Notices* 36.7 (2001), pp. 123–132.

[107] *TotalView for HPC*. URL: https://www.roguewave.com/products-services/totalview.

[108] K. Devine et al. "Zoltan data management services for parallel dynamic applications". In: *Computing in Science Engineering* 4.2 (Mar. 2002), pp. 90–96. ISSN: 1521-9615. DOI: 10.1109/5992.988653.

[109] Ewing L. Lusk, Steve C. Pieper, Ralph M. Butler, et al. "More scalability, less pain: A simple programming model and its implementation for extreme computing". In: *SciDAC Review* 17.1 (2010), pp. 30–37.

[110] Tom Peterka et al. "Scalable parallel building blocks for custom data analysis". In: *IEEE Symposium on Large Data Analysis and Visualization (LDAV 2011)*. IEEE. 2011, pp. 105–112.

[111] H. Georgi. "Effective field theory". In: *Ann. Rev. Nucl. Part. Sci.* 43 (1993), pp. 209–252. DOI: 10.1146/annurev.ns.43.120193.001233.

[112] Vardan Khachatryan et al. "Study of vector boson scattering and search for new physics in events with two same-sign leptons and two jets". In: *Phys. Rev. Lett.* 114.5 (2015), p. 051801. DOI: 10.1103/PhysRevLett.114.051801. arXiv: 1410.6315 [hep-ex].

[113] Morad Aaboud et al. "Measurement of $W^{\pm}W^{\pm}$ vector-boson scattering and limits on anomalous quartic gauge couplings with the ATLAS detector". In: *Phys. Rev.* D96.1 (2017), p. 012007. DOI: 10.1103/PhysRevD.96.012007. arXiv: 1611.02428 [hep-ex].

[114] Michelangelo Mangano. "Physics at the FCC-hh, a 100 TeV pp collider". In: *CERN Yellow Rep. Monogr.* 3 (2017). DOI: 10.23731/CYRM-2017-003. arXiv: 1710.06353 [hep-ph].

[115] Philip Bambade et al. "The International Linear Collider: A Global Project". In: (2019). arXiv: 1903.01629 [hep-ex].

[116] T. K. Charles et al. "The Compact Linear Collider (CLIC) - 2018 Summary Report". In: *CERN Yellow Rep. Monogr.* 1802 (2018), pp. 1–98. DOI: 10.23731/CYRM-2018-002. arXiv: 1812.06018 [physics.acc-ph].

[117] Benjamin W. Lee, C. Quigg, and H. B. Thacker. "Weak Interactions at Very High-Energies: The Role of the Higgs Boson Mass". In: *Phys. Rev.* D16 (1977), p. 1519. DOI: 10.1103/PhysRevD.16.1519.

[118] Benjamin W. Lee, C. Quigg, and H. B. Thacker. "The Strength of Weak Interactions at Very High-Energies and the Higgs Boson Mass". In: *Phys. Rev. Lett.* 38 (1977), pp. 883–885. DOI: 10.1103/PhysRevLett.38.883.

[119] Steven Weinberg. "Physical Processes in a Convergent Theory of the Weak and Electromagnetic Interactions". In: *Phys. Rev. Lett.* 27 (1971), pp. 1688–1691. DOI: 10.1103/PhysRevLett.27.1688.

[120] Steven Weinberg. "General Theory of Broken Local Symmetries". In: *Phys. Rev.* D7 (1973), pp. 1068–1082. DOI: 10.1103/PhysRevD.7.1068.

[121] Georges Aad et al. "Evidence for Electroweak Production of $W^{\pm}W^{\pm}jj$ in pp Collisions at $\sqrt{s} = 8$ TeV with the ATLAS Detector". In: *Phys. Rev. Lett.* 113.14 (2014), p. 141803. DOI: 10.1103/PhysRevLett.113.141803. arXiv: 1405.6241 [hep-ex].

[122] CMS Collaboration. "Vector boson scattering in a final state with two jets and two same-sign leptons". In: (2014).

[123] O. J. P. Éboli and M. C. Gonzalez–Garcia. "Classifying the bosonic quartic couplings". In: *Phys. Rev.* D93.9 (2016), p. 093013. DOI: 10.1103/PhysRevD.93.093013. arXiv: 1604.03555 [hep-ph].

[124] Steven Weinberg. "Baryon and Lepton Nonconserving Processes". In: *Phys. Rev. Lett.* 43 (1979), pp. 1566–1570. DOI: 10.1103/PhysRevLett.43.1566.

[125] Wolfgang Kilian. "Electroweak interactions: Summary". In: *AIP Conf. Proc.* 578.1 (2001), pp. 503–510. DOI: 10.1063/1.1394369. arXiv: hep-ph/0012243 [hep-ph].

[126] Steven Weinberg. "Varieties of Baryon and Lepton Nonconservation". In: *Phys. Rev.* D22 (1980), p. 1694. DOI: 10.1103/PhysRevD.22.1694.

[127] Celine Degrande et al. “Monte Carlo tools for studies of non-standard electroweak gauge boson interactions in multi-boson processes: A Snowmass White Paper”. In: *Proceedings, 2013 Community Summer Study on the Future of U.S. Particle Physics: Snowmass on the Mississippi (CSS2013): Minneapolis, MN, USA, July 29-August 6, 2013*. 2013. arXiv: 1309.7890 [hep-ph]. URL: http://www.slac.stanford.edu/econf/C1307292/docs/submittedArxivFiles/1309.7890.pdf.

[128] O. J. P. Eboli, M. C. Gonzalez-Garcia, and J. K. Mizukoshi. “p p —> j j e+- mu+- nu nu and j j e+- mu-+ nu nu at O(alpha(em)**6) and O(alpha(em)**4 alpha(s)**2) for the study of the quartic electroweak gauge boson vertex at CERN LHC”. In: *Phys. Rev.* D74 (2006), p. 073005. DOI: 10.1103/PhysRevD.74.073005. arXiv: hep-ph/0606118 [hep-ph].

[129] M. Baak et al. “Working Group Report: Precision Study of Electroweak Interactions”. In: *Proceedings, 2013 Community Summer Study on the Future of U.S. Particle Physics: Snowmass on the Mississippi (CSS2013): Minneapolis, MN, USA, July 29-August 6, 2013*. 2013. arXiv: 1310.6708 [hep-ph]. URL: http://www.slac.stanford.edu/econf/C1307292/docs/EnergyFrontier/Electroweak-19.pdf.

[130] Marco Sekulla. “Anomalous couplings, resonances and unitarity in vector boson scattering”. PhD thesis. Siegen U., 2015. URL: http://dokumentix.ub.uni-siegen.de/opus/volltexte/2016/979/index.html.

[131] D. A. Ross and M. J. G. Veltman. “Neutral Currents in Neutrino Experiments”. In: *Nucl. Phys.* B95 (1975), pp. 135–147. DOI: 10.1016/0550-3213(75)90485-X.

[132] M. J. G. Veltman. “Second Threshold in Weak Interactions”. In: *Acta Phys. Polon.* B8 (1977), p. 475.

[133] P. Sikivie et al. “Isospin Breaking in Technicolor Models”. In: *Nucl. Phys.* B173 (1980), pp. 189–207. DOI: 10.1016/0550-3213(80)90214-X.

[134] Geoffrey F. Chew and Stanley Mandelstam. “Theory of low-energy pion pion interactions”. In: *Phys. Rev.* 119 (1960), pp. 467–477. DOI: 10.1103/PhysRev.119.467.

[135] Michael Rauch. “Vector-Boson Fusion and Vector-Boson Scattering”. In: (2016). arXiv: 1610.08420 [hep-ph].

[136] Anthony C. Longhitano. “Low-Energy Impact of a Heavy Higgs Boson Sector”. In: *Nucl. Phys.* B188 (1981), pp. 118–154. DOI: 10.1016/0550-3213(81)90109-7.

[137] W. Heitler. “The influence of radiation damping on the scattering of light and mesons by free particles. I”. In: *Mathematical Proceedings of the Cambridge Philosophical Society* 37.3 (1941), pp. 291–300. DOI: 10.1017/S0305004100021770.

[138] Julian Schwinger. “Quantum Electrodynamics. I. A Covariant Formulation”. In: *Phys. Rev.* 74 (10 Nov. 1948), pp. 1439–1461. DOI: 10.1103/PhysRev.74.1439. URL: https://link.aps.org/doi/10.1103/PhysRev.74.1439.

[139] Genessis Perez, Marco Sekulla, and Dieter Zeppenfeld. “Anomalous quartic gauge couplings and unitarization for the vector boson scattering process $pp \to W^+W^+jjX \to \ell^+\nu_\ell\ell^+\nu_\ell jjX$”. In: *Eur. Phys. J.* C78.9 (2018), p. 759. DOI: 10.1140/epjc/s10052-018-6230-1. arXiv: 1807.02707 [hep-ph].

[140] Andrei D. Sakharov. "Interaction of the electron and the positron in pair production". In: *Soviet Physics Uspekhi* 34.5 (May 1991), pp. 375–377. DOI: 10.1070/pu1991v034n05abeh002492.

[141] D. Bardin, W. Beenakker, and A. Denner. "The Coulomb singularity in off-shell W-pair production". In: *Physics Letters B* 317.1 (1993), pp. 213–217. ISSN: 0370-2693. DOI: https://doi.org/10.1016/0370-2693(93)91595-E. URL: http://www.sciencedirect.com/science/article/pii/037026939391595E.

[142] Suraj N. Gupta. *Quantum electrodynamics*. New York, 1977.

[143] Michael S. Chanowitz. "Quantum corrections from nonresonant W W scattering". In: *Phys. Rept.* 320 (1999), pp. 139–146. DOI: 10.1016/S0370-1573(99)00077-0. arXiv: hep-ph/9903522 [hep-ph].

[144] S. U. Chung et al. "Partial wave analysis in K-matrix formalism". In: *Annalen der Physik* 507.5 (1995), pp. 404–430. DOI: 10.1002/andp.19955070504. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/andp.19955070504. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/andp.19955070504.

[145] Eugene Paul Wigner. "Gruppentheorie und ihre Anwendung auf die Quantenmechanik der Atomspektren". In: (1931).

[146] Suraj N. Gupta. "Theory of longitudinal photons in quantum electrodynamics". In: *Proc. Phys. Soc.* A63 (1950), pp. 681–691. DOI: 10.1088/0370-1298/63/7/301.

[147] Suraj N. Gupta, James M. Johnson, and Wayne W. Repko. "W, Z and Higgs scattering at SSC energies". In: *Phys. Rev.* D48 (1993), pp. 2083–2096. DOI: 10.1103/PhysRevD.48.2083. arXiv: hep-ph/9307239 [hep-ph].

[148] Micheal S. Berger and Michael S. Chanowitz. "Strong W+ W+ scattering at the SSC". In: *Phys. Lett.* B263 (1991), pp. 509–516. DOI: 10.1016/0370-2693(91)90497-E.

[149] Wolfram Research Inc. *Mathematica, Version 11.3*. Champaign, IL, 2018.

[150] Neil D. Christensen and Claude Duhr. "FeynRules - Feynman rules made easy". In: *Comput. Phys. Commun.* 180 (2009), pp. 1614–1641. DOI: 10.1016/j.cpc.2009.02.018. arXiv: 0806.4194 [hep-ph].

[151] Adam Alloul et al. "FeynRules 2.0 - A complete toolbox for tree-level phenomenology". In: *Comput. Phys. Commun.* 185 (2014), pp. 2250–2300. DOI: 10.1016/j.cpc.2014.04.012. arXiv: 1310.1921 [hep-ph].

[152] Simon Braß, Gudrun Hiller, and Ivan Nisandzic. "Zooming in on $B \to K^* \ell\ell$ decays at low recoil". In: *Eur. Phys. J.* C77.1 (2017), p. 16. DOI: 10.1140/epjc/s10052-016-4576-9. arXiv: 1606.00775 [hep-ph].

[153] Dario Buttazzo, Admir Greljo, and David Marzocca. "Knocking on new physics' door with a scalar resonance". In: *Eur. Phys. J.* C76.3 (2016), p. 116. DOI: 10.1140/epjc/s10052-016-3970-7. arXiv: 1512.04929 [hep-ph].

[154] Steven B. Giddings and Hao Zhang. "Kaluza-Klein graviton phenomenology for warped compactifications, and the 750 GeV diphoton excess". In: *Phys. Rev.* D93.11 (2016), p. 115002. DOI: 10.1103/PhysRevD.93.115002. arXiv: 1602.02793 [hep-ph].

[155] Chengcheng Han et al. "The diphoton resonance as a gravity mediator of dark matter". In: *Phys. Lett.* B755 (2016), pp. 371–379. DOI: 10.1016/j.physletb.2016.02.040. arXiv: 1512.06376 [hep-ph].

[156] Lisa Randall and Raman Sundrum. "A Large mass hierarchy from a small extra dimension". In: *Phys. Rev. Lett.* 83 (1999), pp. 3370–3373. DOI: 10.1103/PhysRevLett.83.3370. arXiv: hep-ph/9905221 [hep-ph].

[157] Nima Arkani-Hamed, Savas Dimopoulos, and G. R. Dvali. "The Hierarchy problem and new dimensions at a millimeter". In: *Phys. Lett.* B429 (1998), pp. 263–272. DOI: 10.1016/S0370-2693(98)00466-3. arXiv: hep-ph/9803315 [hep-ph].

[158] Chao-Qiang Geng and Da Huang. "Note on spin-2 particle interpretation of the 750 GeV diphoton excess". In: *Phys. Rev.* D93.11 (2016), p. 115032. DOI: 10.1103/PhysRevD.93.115032. arXiv: 1601.07385 [hep-ph].

[159] Roman Pasechnik et al. "Chiral-symmetric technicolor with standard model Higgs boson". In: *Phys. Rev.* D88.7 (2013), p. 075009. DOI: 10.1103/PhysRevD.88.075009. arXiv: 1304.2081 [hep-ph].

[160] Yeo Woong Yoon et al. "Radiative decays of a singlet scalar boson through vectorlike quarks". In: *Phys. Rev.* D96.5 (2017), p. 055041. DOI: 10.1103/PhysRevD.96.055041. arXiv: 1705.05486 [hep-ph].

[161] Martin Bauer, Clara Hörner, and Matthias Neubert. "Diphoton Resonance from a Warped Extra Dimension". In: *JHEP* 07 (2016), p. 094. DOI: 10.1007/JHEP07(2016)094. arXiv: 1603.05978 [hep-ph].

[162] Roberto Franceschini et al. "What is the $\gamma\gamma$ resonance at 750 GeV?" In: *JHEP* 03 (2016), p. 144. DOI: 10.1007/JHEP03(2016)144. arXiv: 1512.04933 [hep-ph].

[163] Rick S. Gupta et al. "Interpreting a 750 GeV Diphoton Resonance". In: *JHEP* 07 (2016), p. 145. DOI: 10.1007/JHEP07(2016)145. arXiv: 1512.05332 [hep-ph].

[164] Jong Soo Kim et al. "A resonance without resonance: scrutinizing the diphoton excess at 750 GeV". In: *Phys. Lett.* B755 (2016), pp. 403–408. DOI: 10.1016/j.physletb.2016.02.041. arXiv: 1512.06083 [hep-ph].

[165] Georges Aad et al. "Search for high-mass diphoton resonances in pp collisions at $\sqrt{s} = 8$ TeV with the ATLAS detector". In: *Phys. Rev.* D92.3 (2015), p. 032004. DOI: 10.1103/PhysRevD.92.032004. arXiv: 1504.05511 [hep-ex].

[166] Vardan Khachatryan et al. "Search for diphoton resonances in the mass range from 150 to 850 GeV in pp collisions at $\sqrt{s} = 8$ TeV". In: *Phys. Lett.* B750 (2015), pp. 494–519. DOI: 10.1016/j.physletb.2015.09.062. arXiv: 1506.02301 [hep-ex].

[167] R. L. Delgado et al. "Production of vector resonances at the LHC via WZ-scattering: a unitarized EChL analysis". In: *JHEP* 11 (2017), p. 098. DOI: 10.1007/JHEP11(2017)098. arXiv: 1707.04580 [hep-ph].

[168] Nima Arkani-Hamed, Andrew G. Cohen, and Howard Georgi. "Electroweak symmetry breaking from dimensional deconstruction". In: *Phys. Lett.* B513 (2001), pp. 232–240. DOI: 10.1016/S0370-2693(01)00741-9. arXiv: hep-ph/0105239 [hep-ph].

[169] N. Arkani-Hamed et al. "The Littlest Higgs". In: *JHEP* 07 (2002), p. 034. DOI: 10.1088/1126-6708/2002/07/034. arXiv: hep-ph/0206021 [hep-ph].

[170] W. Kilian and J. Reuter. "The Low-energy structure of little Higgs models". In: *Phys. Rev.* D70 (2004), p. 015004. DOI: 10.1103/PhysRevD.70.015004. arXiv: hep-ph/0311095 [hep-ph].

[171] J. Pumplin et al. "New generation of parton distributions with uncertainties from global QCD analysis". In: *JHEP* 07 (2002), p. 012. DOI: 10.1088/1126-6708/2002/07/012. arXiv: hep-ph/0201195 [hep-ph].

[172] Bijan Chokoufé Nejad et al. "*work in progress*". In: (2019).

[173] Thomas Appelquist and Claude W. Bernard. "Strongly Interacting Higgs Bosons". In: *Phys. Rev.* D22 (1980), p. 200. DOI: 10.1103/PhysRevD.22.200.

[174] Anthony C. Longhitano. "Heavy Higgs Bosons in the Weinberg-Salam Model". In: *Phys. Rev.* D22 (1980), p. 1166. DOI: 10.1103/PhysRevD.22.1166.

[175] Thomas Appelquist and Guo-Hong Wu. "The Electroweak chiral Lagrangian and new precision measurements". In: *Phys. Rev.* D48 (1993), pp. 3235–3241. DOI: 10.1103/PhysRevD.48.3235. arXiv: hep-ph/9304240 [hep-ph].

[176] J. D. Hunter. "Matplotlib: A 2D graphics environment". In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95. DOI: 10.1109/MCSE.2007.55.

[177] Joshua P. Ellis. "TikZ-Feynman: Feynman diagrams with TikZ". In: *Computer Physics Communications* 210 (2017), pp. 103–123. ISSN: 0010-4655. DOI: https://doi.org/10.1016/j.cpc.2016.08.019. URL: http://www.sciencedirect.com/science/article/pii/S0010465516302521.

[178] Stéfan van der Walt, S. Chris Colbert, and Gaël Varoquaux. "The NumPy Array: A Structure for Efficient Numerical Computation". In: *Computing in Science & Engineering* 13.2 (2011), pp. 22–30. DOI: 10.1109/MCSE.2011.37. eprint: https://aip.scitation.org/doi/pdf/10.1109/MCSE.2011.37. URL: https://aip.scitation.org/doi/abs/10.1109/MCSE.2011.37.

Glossary

API application programming interface. 26, 58

aQGC anomalous quartic gauge coupling. xii, 63–65, 67, 69–72, 82, 92, 100

aTGC anomalous trilinear gauge coupling. 64, 65

BSM beyond the Standard Model. 9, 22, 63, 84

CERN European Organization for Nuclear Research. 9, 11

CKM Cabibbo-Kobayashi-Maskawa. 1, 67

CLIC Compact Linear Collider. xv, 2, 9, 63

CPU central-processing unit. 23, 25, 26

DAG directed acyclical graph. 24, 25

DESY Deutsches Elektron-Synchrotron. v, 11

EFT effective field theory. ix, xvii, 61–64, 67, 68, 72, 73, 80, 83, 84, 100, 109

EWSB electroweak symmetry breaking. 63, 64, 67, 83, 110

FCC Future Circular Collider. xv, 2, 9, 63

FKS Frixione–Kunszt–Signer. 8

HDF5 hierarchical file format, version 5. 26, 30, 53

HPC high-perfomance computing. xvi, 11, 12, 26, 52

ILC International Linear Collider. xv, 2, 9, 63

ISR initial-state radiation. 8

LCG linear congruence generator. 34

LHC Large Hadron Collider. ii, iii, xv–xviii, 2, 4, 8, 11, 23, 61–63, 71, 84, 97–101, 117

LO leading order. 23, 70, 84, 88, 100

MPI message-passing interface. ix, 11, 12, 26, 27, 29–32, 41–43, 46, 48, 51, 53, 54, 56–59, 89, 99

NLO next-to-leading order. 8–10, 23, 40, 100

PDF parton distribution function. ix, 8, 15, 55, 71, 115

pRNG pseudo-random number generator. 33, 34

QCD Quantum Chromodynamics. 1, 3, 4, 8, 9, 54, 61, 71, 72, 87, 88, 100

QFT quantum field theory. xv, 1–3, 62, 63, 98, 100

QGC quartic gauge coupling. 63, 64, 109

RNG random number generator. 9, 33–35, 115

SM Standard Model. ix–xi, xv, xvii, 1, 2, 9, 61–65, 67, 68, 70, 71, 73, 88, 90, 92–100, 117–121

SMEFT Standard Model Effective Field Theory. xvii, xviii, 62, 71, 83, 84, 87, 90, 92, 95, 97, 99–101

TGC triple gauge coupling. 63, 64

VBF vector-boson fusion. xvii, 61, 62

VBS vector-boson scattering. xvii, 61–63, 67, 70, 71, 73, 82, 83, 87, 88, 90, 92, 95, 97, 99

vev vacuum expectation value. 68