FERMILAB-CONF-23-104-T

Performance of modern color decompositions for standard candle LHC tree amplitudes

E Bothmann[1], JT Childers[2], W Giele[3], S Höche[3], J Isaacson[3], M Knobbe[1] and R Wang[2]

[1]Institut für Theoretische Physik, Georg-August-Universität Göttingen, 37077 Göttingen, Germany
[2]Argonne National Laboratory, Lemont, IL, 60439, USA
[3]Fermi National Accelerator Laboratory, Batavia, IL 60510, USA

E-mail: `max.knobbe@uni-goettingen.de`

Abstract. In the last decade, developments of matrix element and phase space generators have focused on providing good efficiency and maximal flexibility and automation for a wide range of physical processes. However, as recent studies have shown, they are a major bottleneck in the established Monte Carlo event generator toolchains. With the advent of the HL-LHC and ever rising precision requirements, future developments will need to focus on computational performance, especially at intermediate to large jet multiplicities. We present the novel BlockGen family of fast matrix element algorithms that are amenable for GPU acceleration, making use of modern, minimal color decompositions. Moreover, we discuss the performance achieved for standard candle processes such as V+jets and $t\bar{t}$+jets production.

1. Introduction

The physics program at high energy particle colliders crucially depends on the precise simulation of scattering events at particle level. These simulations are carried out by event generators, computer codes that link theory to experiment by the Monte Carlo method. Event generators are usually combinations of different modules, each encapsulating the physics at a given energy scale. The highest energy scale is given by the hard scattering process and the physics can be described by perturbation theory. The computations are fully automated at tree-level [1, 2, 3, 4, 5] and next-to-leading order [6, 7, 8, 9, 10, 11] in the strong coupling constant α_s. The number of particles participating in the scattering process is arbitrary in principle, but limited in practice due to the inherently exponential scaling of the computing algorithms.

Among the computationally most demanding Monte Carlo samples are those of background processes like V+Jets or top-antitop-pair production in association with a large number of associated jets. These processes contribute to the background of many current experimentally interesting processes while still having large cross-sections and thus they require excellent statistics [12]. For standard setups the computational bottleneck of the Monte-Carlo generation have been found to be the computation of tree-level matrix elements and the generation of phase-space points and their corresponding weights [13, 14]. Delivering good computational performance for these physical processes is one of the main goals of this work.

Due to the inherent parallel nature of event generation, one can make use of modern computational architectures like GPUs that are specialized for such applications. GPUs also become increasingly available in the HPC landscape. Among other things, direct applications of GPU accelerated computations in the Monte Carlo toolchain has been investigated for tree-level matrix elements [15, 16], loop-amplitudes [17, 18] and parton density functions [19]. However, a fully functional GPU accelerated Monte Carlo event generator is not yet available.

In these proceedings, we report on our efforts to deliver some of the missing pieces, namely the matrix element computation for more general amplitudes and the phase space generation.

2. Amplitude computations

Based on earlier studies [20, 21, 22], we chose Berends-Giele recursion [23] for the amplitude computations. The recursive nature yields a favorable scaling and in particular allows the computation of many-jet amplitudes. The amplitudes are accompanied by their respective QCD color factors, that can either be factorized via color-decompositions or embedded directly in the recursion using a technique called color-dressing [21, 24].

In a previous study [25] we explored different color decomposition and dressing methods in the simplified context of pure-gluon amplitudes. For GPU acceleration, we concluded that the most promising approach is to sum colors using a color-decomposition with a minimal basis. In the pure gluon case this basis can be obtained in the adjoint representation and has been know for some time [26, 27, 28]:

$$\mathcal{A}^{\lambda_1\ldots\lambda_n}_{a_1\ldots a_n}(p_1,\ldots,p_n) = \sum_{\vec{\sigma}\in S_{n-2}} (F^{a_{\sigma_2}}\ldots F^{a_{\sigma_{n-1}}})_{a_1 a_n}\, A^{\lambda_1\ldots\lambda_n}(p_1,p_{\sigma_2},\ldots,p_{\sigma_{n-1}},p_n)\,, \tag{1}$$

where $F^a_{bc} = if^{abc}$. The functions A are called color-ordered or partial amplitudes and are stripped of any color information, which is now contained purely in the color coefficients. In the following, we abbreviate $A(p_1,\ldots,p_n) := A(1,\ldots,n)$. If the partial amplitudes carry a helicity label they are often simply referred to as helicity amplitudes. The multi-index $\vec{\sigma}$ runs over all permutations S_{n-2} of the $(n-2)$ gluon indices $2,\ldots,n-1$. This basis has $(n-2)!$ elements and scales factorially with the number of gluons. The factorial scaling favors other algorithms for very large multiplicities but is still manageable for up to roughly six to eight outgoing particles.

For QCD amplitudes involving not only gluons but also quarks, the corresponding minimal basis has recently been developed, and it is shown that the elements of the basis can be described by Dyck words [29, 30]. A Dyck word is a set of brackets, e.g. (and), such that the number of opening brackets equals the number of closing brackets, and the number of opening brackets is always larger or equal the number of closing ones for any subset starting at the beginning of the Dyck word. For example for four characters and one type of brackets (and), there are two Dyck words:

$$(()),\ ()()\,, \tag{2}$$

and for six characters there are five:

$$((())),\ (()()),\ (())(),\ ()(()),\ ()()()\,. \tag{3}$$

Dyck words can also consist of different types of brackets, say ( ) and [ ]. In this case, there should never be an odd number of one type of brackets embraced by another bracket. In the four character example above, there are four Dyck words:

$$([]),\ [()],\ ()[],\ []()\,, \tag{4}$$

and for example [)(] is not a valid word. For the amplitude computation, every opening bracket represents a quark and every closing one an anti-quark. Different bracket types are understood as differently flavored quarks.

Similar to the gluon case in Eq. (1), the kinematic basis allows to keep two indices fixed and permute the others, this time using Dyck words. For purely fermionic amplitudes the basis can thus be written as:

$$\{A(1,2,\sigma) \,|\, \sigma \in \text{Dyck}\} \,. \tag{5}$$

For amplitudes involving gluons, we allow all possible gluon insertions and permutations in the Dyck word, but not within or in front of the initial, fixed fermion line. Finally, for multiple fermion lines with the same flavor, we allow all possible permutations of same-flavor quarks. In the following, we call this generated basis the Melia basis.

In the Melia basis, for n particles and k distinct quark-pairs, the number of amplitudes is [29, 30]:

$$\varkappa(n,k)_{\text{Melia}} = \frac{(n-2)!}{k!} \,, \tag{6}$$

which is a generalization of the all-gluon ($k=0$) and the single-quark-pair ($k=1$) basis.

For the color decomposition of the squared amplitude, we still need the corresponding color factors. They can be obtained in a closed form following the procedure given in Ref. [31] and proven in Ref. [32].

3. Phase space generation

As noted in [33], for the processes of interest here, we can achieve good performance with a relatively simple phase space generator as realized in MCFM [34, 35]. The core idea is based on the fact that the n-particle phase-space element:

$$\mathrm{d}\phi_n(a,b;1,...,n) = \left[\sum_{i=1}^{n} \frac{\mathrm{d}\vec{p}_i}{(2\pi)^3 2E_i}\right] (2\pi)^4 \delta\left(p_a + p_b - \sum_{i=1}^{n} p_i\right) \,, \tag{7}$$

for incoming particles a, b and outgoing particles $1,\dots,n$, can be decomposed into a product of lower-dimensional phase space elements via [36, 37]

$$\mathrm{d}\phi_n(a,b;1,...,n) = \mathrm{d}_{n-m+1}(a,b,\pi,m+1,...,n)\frac{\mathrm{d}s_\pi}{2\pi}\mathrm{d}\phi_m(\pi;1,...,m) \,, \tag{8}$$

with π being an intermediate particle with $s_\pi^2 = p_\pi^2$. Repeating this decomposition, the phase-space element can be understood as a single t-channel with a number of s-channels. This allows to map the structure of the matrix elements to different channels making use of the multi-channel technique. In this first implementation, we omit the s-channels and only study the performance of a single t-channel integrator. For the performance of the full integrator see [38].

4. Computational Performance

4.1. Matrix Element

In Fig. 1 we show the time required per matrix element computation for different algorithmic choices. We compare the currently used matrix-element generator COMIX for color summed and helicity summed configurations against our new matrix elements computed on a CPU and on a GPU. The current default setup of COMIX is color-sampled and helicity-summed, which is therefore the baseline for our ratio plot. We find that our new matrix element is faster for small multiplicities even on the CPU. Comparing the scaling with COMIX, we find that our new matrix element scales worse and we reach similar computational performance for five additional jets. This is to be expected, since the color-dressed approach of COMIX scales exponentially while the color sum implemented in BlockGen scales factorially. The GPU version scales similar to the CPU version but is roughly two orders of magnitude faster in the whole multiplicity range. Taking into account that the CPU version would usually be parallelized and run on multiple CPU core, this leaves a practical performance gain of roughly an order of magnitude.

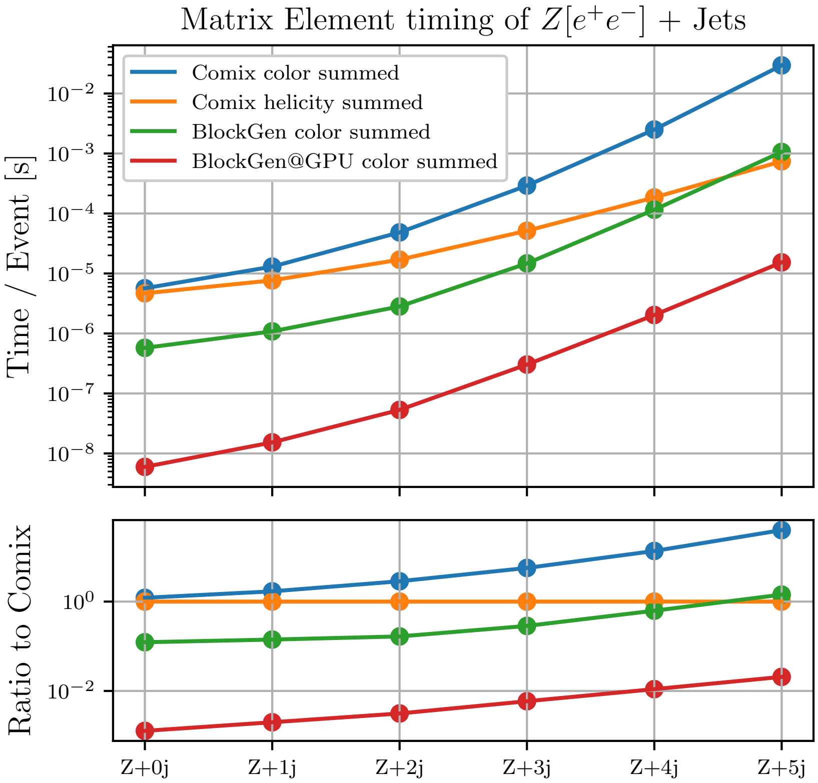

Figure 1. Comparison of different single-threaded CPU-based algorithms for $Z[e^+e^-]$+jets production. The CPU numbers are generated on an Intel® i3-8300 (3.70GHz), and the GPU numbers are generated on a NVIDIA® Tesla V100S.

4.2. Phase space

Table 1 shows a comparison of the number of points and computing time required to reach a predefined accuracy for the new phase space ("New PS") and the current Sherpa default ("Default PS", which is the recursive phase space as described in [1]). We test the production of leptonically decaying W^+, on-shell H production and pure QCD processes like $t\bar{t}$ and pure jet production. These processes are required for a large number of background analysis at the LHC. Furthermore, the mixture allows to probe the phase-space generator in most of the relevant scenarios. We observe that the new generator requires fewer points for most of the processes and in particular significantly reduces the computing time. The exception is the clearly s-channel dominated $t\bar{t}$-production where the new generator only yields small improvements. We expect this to improve with the addition of s-channel phase-space elements.

5. Outlook

We have presented preliminary results for our current effort to develop a novel matrix element generator using the minimal number of amplitudes while color-summing. We tested the computing performance for the production of Z-boson accompanied by a number of jets and found significant performance gains. Furthermore we have shown new tests for a relatively simple phase space generator that yields good results for the processes we are interested in. The simple structure is promising for portability and the possible performance on accelerators.

In the future, we plan to combine the different components to a fully functional GPU-accelerated Monte Carlo generator suitable to generate unweighted events. We plan to interface

Table 1. Runtimes and required number of points for given accuracy. All run on a Xeon® E5-2650v2, timings taken for the matrix element and the phase space, # points before cuts. The center-of-mass energy is $\sqrt{s} = 14$TeV and jets are defined using the anti-k_T algorithm with $p_{T,j} = 30$GeV. The scales are chosen to be $\mu_{R/F} = H'_T/2$. The Higgs-bosons are generated on shell.

Process / MC accu	Default PS Time	Default PS # pts	New PS Time	New PS # pts	Process / MC accu	Default PS Time	Default PS # pts	New PS Time	New PS # pts
W+1j / 1‰	4m 52s	10.3M	2m 32s	3.10M	$t\bar{t}$+0j / 1‰	4m 38s	3.15M	4m 0s	3.59M
W+2j / 3‰	17m 12s	5.52M	13m 52s	2.53M	$t\bar{t}$+1j / 3‰	3m 12s	1.38M	3m 4s	1.47M
W+3j / 1%	46m 24s	7.48M	20m 16s	1.15M	$t\bar{t}$+2j / 1%	11m 58s	1.47M	11m 20s	0.89M
H+1j / 1‰	2m 20s	1.83M	1m 36s	1.50M	2j / 1‰	12m 48s	2.98M	7m 44s	1.80M
H+2j / 3‰	4m 36s	2.32M	4m 4s	0.71M	3j / 3‰	22m 48s	6.80M	23m 12s	2.39M
H+3j / 1%	18m 12s	2.32M	12m 56s	0.63M	4j / 1%	1h 25m	6.95M	50m 24s	0.91M

these events to Sherpa via the existing HDF5 interface [14] for the remaining processing steps such as parton showering and hadronisation.

Acknowledgments

This research was supported by the Fermi National Accelerator Laboratory (Fermilab), a U.S. Department of Energy, Office of Science, HEP User Facility. Fermilab is managed by Fermi Research Alliance, LLC (FRA), acting under Contract No. DE–AC02–07CH11359. The work of S.H. and J.I. was supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Scientific Discovery through Advanced Computing (SciDAC) program, grant "HPC framework for event generation at colliders". E.B. and M.K. acknowledge support from BMBF (contract 05H21MGCAB). Their research is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 456104544; 510810461. This research used resources of the Argonne Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC02-06CH11357.

References

[1] Gleisberg T and Hoeche S 2008 *JHEP* **12** 039 (*Preprint* `0808.3674`)
[2] Cafarella A, Papadopoulos C G and Worek M 2009 *Comput. Phys. Commun.* **180** 1941–1955 (*Preprint* `0710.2427`)
[3] Mangano M L, Moretti M, Piccinini F, Pittau R and Polosa A D 2003 *JHEP* **07** 001 (*Preprint* `hep-ph/0206293`)
[4] Krauss F, Kuhn R and Soff G 2002 *JHEP* **02** 044 (*Preprint* `hep-ph/0109036`)
[5] Alwall J, Herquet M, Maltoni F, Mattelaer O and Stelzer T 2011 *JHEP* **06** 128 (*Preprint* `1106.0522`)
[6] Berger C F, Bern Z, Dixon L J, Febres Cordero F, Forde D, Ita H, Kosower D A and Maitre D 2008 *Phys. Rev. D* **78** 036003 (*Preprint* `0803.4180`)
[7] Actis S, Denner A, Hofer L, Lang J N, Scharf A and Uccirati S 2017 *Comput. Phys. Commun.* **214** 140–173 (*Preprint* `1605.01090`)
[8] Alwall J, Frederix R, Frixione S, Hirschi V, Maltoni F, Mattelaer O, Shao H S, Stelzer T, Torrielli P and Zaro M 2014 *JHEP* **07** 079 (*Preprint* `1405.0301`)
[9] Cascioli F, Maierhofer P and Pozzorini S 2012 *Phys. Rev. Lett.* **108** 111601 (*Preprint* `1111.5206`)
[10] Bevilacqua G, Czakon M, Garzelli M V, van Hameren A, Kardos A, Papadopoulos C G, Pittau R and Worek M 2013 *Comput. Phys. Commun.* **184** 986–997 (*Preprint* `1110.1499`)
[11] Cullen G, Greiner N, Heinrich G, Luisoni G, Mastrolia P, Ossola G, Reiter T and Tramontano F 2012 *Eur. Phys. J. C* **72** 1889 (*Preprint* `1111.2034`)
[12] Aad G *et al.* (ATLAS) 2022 *JHEP* **08** 089 (*Preprint* `2112.09588`)
[13] Bothmann E, Buckley A, Christidi I A, Gütschow C, Höche S, Knobbe M, Martin T and Schönherr M 2022 *Eur. Phys. J. C* **82** 1128 (*Preprint* `2209.00843`)

[14] Höche S, Prestel S and Schulz H 2019 *Phys. Rev.* **D100** 014024 (*Preprint* 1905.05120)

[15] Hagiwara K, Kanzaki J, Okamura N, Rainwater D and Stelzer T 2010 *Eur. Phys. J. C* **66** 477–492 (*Preprint* 0908.4403)

[16] Giele W, Stavenga G and Winter J C 2011 *Eur. Phys. J. C* **71** 1703 (*Preprint* 1002.3446)

[17] Yuasa F, Ishikawa T, Hamaguchi N, Koike T and Nakasato N 2013 *J. Phys. Conf. Ser.* **454** 012081

[18] Borowka S, Heinrich G, Jahn S, Jones S P, Kerner M and Schlenk J 2019 *Comput. Phys. Commun.* **240** 120–137 (*Preprint* 1811.11720)

[19] Carrazza S, Cruz-Martinez J M and Rossi M 2021 *Comput. Phys. Commun.* **264** 107995 (*Preprint* 2009.06635)

[20] Dinsdale M, Ternick M and Weinzierl S 2006 *JHEP* **03** 056 (*Preprint* hep-ph/0602204)

[21] Duhr C, Hoeche S and Maltoni F 2006 *JHEP* **08** 062 (*Preprint* hep-ph/0607057)

[22] Badger S, Biedermann B, Hackl L, Plefka J, Schuster T and Uwer P 2013 *Phys. Rev. D* **87** 034011 (*Preprint* 1206.2381)

[23] Berends F and Giele W 1988 *Nuclear Physics B* **306** 759–808 ISSN 0550-3213 URL https://www.sciencedirect.com/science/article/pii/0550321388904427

[24] Giele W, Kunszt Z and Winter J 2010 *Nucl. Phys. B* **840** 214–270 (*Preprint* 0911.1962)

[25] Bothmann E, Giele W, Hoeche S, Isaacson J and Knobbe M 2021 (*Preprint* 2106.06507)

[26] Berends F and Giele W 1987 *Nuclear Physics B* **294** 700–732 ISSN 0550-3213 URL https://www.sciencedirect.com/science/article/pii/0550321387906043

[27] Del Duca V, Frizzo A and Maltoni F 2000 *Nucl. Phys. B* **568** 211–262 (*Preprint* hep-ph/9909464)

[28] Del Duca V, Dixon L J and Maltoni F 2000 *Nucl. Phys. B* **571** 51–70 (*Preprint* hep-ph/9910563)

[29] Melia T 2013 *Phys. Rev. D* **88** 014020 (*Preprint* 1304.7809)

[30] Melia T 2013 *PoS* **RADCOR2013** 031

[31] Johansson H and Ochirov A 2016 *JHEP* **01** 170 (*Preprint* 1507.00332)

[32] Melia T 2015 *JHEP* **12** 107 (*Preprint* 1509.03297)

[33] Campbell J M, Höche S and Preuss C T 2021 *Eur. Phys. J. C* **81** 1117 (*Preprint* 2107.04472)

[34] Campbell J M, Ellis R K and Giele W T 2015 *Eur. Phys. J. C* **75** 246 (*Preprint* 1503.06182)

[35] Campbell J and Neumann T 2019 *JHEP* **12** 034 (*Preprint* 1909.09117)

[36] Figy T M and Giele W T 2018 *JHEP* **10** 203 (*Preprint* 1806.09678)

[37] Byckling E and Kajantie K 1969 *Nucl. Phys.* **B9** 568–576

[38] Bothmann E, Childers T, Giele W, Herren F, Hoeche S, Isaacsson J, Knobbe M and Wang R 2023 (*Preprint* 2302.10449)