

# The Matrix Element Method in the LHC era

Sébastien Wertz<sup>1,a</sup>

<sup>1</sup>Centre for Cosmology, Particle Physics and Phenomenology - CP3,  
Université catholique de Louvain, 2 Chemin du Cyclotron - Box L7.01.05,  
B-1348 Louvain-la-Neuve, Belgium

**Abstract.** The Matrix Element Method (MEM) is a powerful multivariate method allowing to maximally exploit the experimental and theoretical information available to an analysis. The method is reviewed in depth, and several recent applications of the MEM at LHC experiments are discussed, such as searches for rare processes and measurements of Standard Model observables in Higgs and Top physics. Finally, a new implementation of the MEM is presented. This project builds on established phase-space parametrisations known to greatly improve the speed of the calculations, and aims at a much improved modularity and maintainability compared to previous software, easing the use of the MEM for high-statistics data analyses.

## 1 Introduction

Multivariate techniques have become ever more ubiquitous in high energy physics, playing an important role at many levels, be it data collection, reconstruction or analysis. In the final steps of the latter, they allow isolating rare signals from backgrounds in searches for physics beyond the Standard Model (SM), searches for SM processes, or measurements of SM observables. Most of these techniques rely on Machine Learning (ML), whereby an algorithm is trained on a sample of (usually simulated) data. Boosted Decision Trees (BDT) and Artificial Neural Networks (ANN), which fall into this category, are widely used and straightforward to implement in an analysis. However, the discrimination that can be attained by these means depends on the size of the samples used in the training phase. In some cases, this can be a limiting factor and significantly restrict the reach of an analysis. As an alternative, the Matrix Element Method (MEM) has the advantage that the achieved signal-to-background discrimination does not depend on the amount of simulated events at hand. Unfortunately, the lack of standard and general implementations of the method and the fact that its use is computationally heavy have so far prevented it from being more widely adopted.

In the next section, the MEM will be introduced in more details. Its use at the LHC so far will then be reviewed. The last section will be dedicated to issues related to the implementation of the method, the review of the MADWEIGHT software, and the introduction of a new project: MoMEMTA.

---

<sup>a</sup>e-mail: [sebastien.wertz@uclouvain.be](mailto:sebastien.wertz@uclouvain.be)

## 2 The Matrix Element Method

The Neyman-Pearson lemma [1] states that the most powerful (in the sense of statistical power) discriminant between two simple hypotheses  $\alpha$  and  $\beta$  is given by the likelihood ratio between these two hypotheses,  $\Lambda(\mathbf{x}) = P(\mathbf{x}|\alpha)/P(\mathbf{x}|\beta)$  (or any invertible function of it). Machine Learning algorithms try to approximate this quantity using some limited knowledge of the probability density functions (PDF) for  $\mathbf{x}$  in the phase-space under these hypotheses. However, for the particular problem of discriminating between different processes in high-energy hadronic collisions, it is possible to compute the likelihood ratio from first principles, by relying on factorisation between the hard process and the soft initial- and final-state dynamics. Specifically, the likelihood for event  $\mathbf{x}$  under hypothesis  $\alpha$  is given by:

$$P(\mathbf{x}|\alpha) = \frac{1}{A_\alpha \sigma_\alpha} \int d\Phi_n(y) \frac{dx_1 dx_2}{x_1 x_2 s} f(x_1) f(x_2) |\mathcal{M}_\alpha(y, x_1, x_2)|^2 W(\mathbf{x}|y) \epsilon_\alpha(y) \quad (1)$$

where:

- $|\mathcal{M}_\alpha|^2$  is the squared matrix element under hypothesis  $\alpha$ . Sums and averages over final and initial states (spin, colour) are implied. In practice, leading order (LO) matrix elements are used.
- The transfer function  $W(\mathbf{x}|y)$  is the likelihood that the partonic configuration  $y$  ends up as the measured event  $\mathbf{x}$  in the detector. It models parton shower, hadronisation, and detector and reconstruction effects. It is normalised as  $\int_\Omega d\mathbf{x} W(\mathbf{x}|y) = 1$  (where  $\Omega$  stands for the experimental acceptance), while the total probability that  $y$  ends up as a selected event is included in the efficiency term  $\epsilon_\alpha(y)$ . Deriving the transfer function is in general a difficult task, which can be solved in practice by assuming independence between the different objects in the events, and independence between the objects' kinematical properties, i.e.:

$$W(\mathbf{x}|y) = \prod_i W_i(p_i^x | p_i^y) = \prod_i W_i^E(E_i^x | E_i^y) W_i^\theta(\theta_i^x | \theta_i^y) W_i^\phi(\phi_i^x | \phi_i^y) \quad (2)$$

For well-reconstructed objects such as charged leptons, it is further possible to assume a perfect resolution on angles or even energies, which has the beneficial effect of reducing the dimensionality of the integration (1):  $W_i(p_i^x | p_i^y) \simeq \delta(E^x - E^y) \delta(\theta^x - \theta^y) \delta(\phi^x - \phi^y)$ . Obtaining the remaining functions can then be achieved e.g. by fitting a given analytic form or by filling histograms using a simulated sample of events.

- $f$  and  $x_i$  are the Parton Distribution Functions and Bjorken- $x$ ;  $d\Phi_n(y)$  is the phase-space density term for the production of  $n$  final-state particles;  $s$  is the hadronic centre-of-mass energy. A sum over all initial state parton flavours contributing to the considered process is implied.
- The denominator  $A_\alpha \sigma_\alpha$ , which includes the total cross section of the process  $\alpha$  and the overall acceptance  $A_\alpha = \langle \epsilon_\alpha(y) \rangle$ , ensures that  $P$  is correctly normalised as a likelihood [2].

Going beyond plain signal-to-background discrimination, having access to the *per-event* likelihood  $P(\mathbf{x}|\alpha)$  allows computing the *sample* likelihood  $P(\mathcal{D}|\alpha) = \prod_{i \in \mathcal{D}} P(\mathbf{x}_i|\alpha)$ , where  $\mathcal{D}$  is the considered experimental dataset. This quantity enables measuring a physical parameter in the model at hand through a maximum-likelihood fit, or testing the data for discrete hypotheses using a Neyman-Pearson discriminant of the sample likelihoods. In either case, because of the approximations required to compute the likelihoods in a reasonable amount of time, one needs to calibrate the fit or derive the PDFs of the sample likelihood ratio using sets of simulated events.

### 3 Uses of the MEM at the LHC

Since the successful use of the MEM by the  $D\bar{\phi}$  and CDF experiments at the Tevatron[3–5], which will not be recounted here, the method has been implemented several times by the general-purpose LHC experiments ATLAS and CMS. The corresponding analyses are summarised in this section.

#### Searches for the $t\bar{t}$ process

A discovery of the production of Standard Model (SM) Higgs bosons in association with Top quark pairs ( $t\bar{t}$  process) will give access to direct measurements of the Top-Higgs coupling, a crucial parameter of the SM. Small signal-to-background (S/B) ratio and poor S/B discrimination render this discovery extremely challenging. Data collected so far have only allowed to set upper limits on the signal strength  $\mu = \sigma_{t\bar{t}h}/\sigma_{SM}$ , but it is expected that LHC Run II will allow an actual discovery of the process (due to a larger signal cross section at 13 TeV and increased integrated luminosity).

The ATLAS and CMS experiments have performed searches for the  $t\bar{t}$  process in many different final states. Some analyses featured MEM-based discriminants: on the one hand in final states with  $h \rightarrow b\bar{b}$  and semi-/fully-leptonic  $t\bar{t}$  decays [6–8], on the other in multi-lepton final states with  $h \rightarrow VV$  [9]. In these analyses, the MEM likelihoods were combined with non-kinematical information (such as quantities related to object reconstruction and identification) to further increase the sensitivity to the signal. In all cases, dedicated implementations of the MEM computation were used.

#### Search for $s$ -channel single Top quark production

$s$ -channel single Top is the rarest Top quark production mode, and was only evidenced recently at the Tevatron. While more difficult to detect at the LHC (due to a less-favourable initial state), it has been searched for, and upper limits on its cross section have been set by ATLAS [10] and CMS [11]. BDTs were used in both these analysis to enhance the sensitivity to the signal.

Recently, ATLAS has repeated the search for this process by re-analysing its 8 TeV dataset using updated calibrations and a Matrix Element technique [12]. ATLAS was able to substantially increase the expected/observed significance for the process in the background-only hypothesis, up from 1.4/1.3 $\sigma$  in [10] to 3.9/3.3 $\sigma$ . Again, an in-house implementation of the MEM was used.

#### Measurement of $t\bar{t}$ spin correlation in the $\mu$ +jets channel

In the SM, Top-quark pairs are produced spin-correlated, and decay quickly enough to imprint this correlation on their decay products. A measurement of this correlation constitutes a crucial test of the SM, and a way to search for indirect effects due to physics beyond the SM. CMS has measured [13] the correlation between pair-produced Top quarks in the  $\mu$ +jets channel using a MEM, yielding the most precise measurement of this quantity in this particular channel. Additionally, the MEM was used to perform hypothesis testing, measuring the compatibility of the data with the correlated or uncorrelated spins hypotheses. In this analysis, the MADWEIGHT software was used for the computation of the likelihoods.

#### Discovery of the Higgs boson and spin-parity measurements in the $h \rightarrow 4\ell$ channel

ME techniques (“MELA”) were used by CMS in the discovery of the SM Higgs boson, to increase the sensitivity of the search for the  $h \rightarrow ZZ^* \rightarrow 4\ell$  process [14]. Furthermore, similar tools were

used by both ATLAS and CMS to constrain the spin-parity properties of this newly discovered resonance [15–20]. Contrary to the three previous use cases presented this section, there is no need to compute integrals such as (1) here, since the final state in the process considered is completely and precisely identified: there are no invisible particles, and the transfer function can be approximated by a delta-function thanks to the precise measurement of the direction and energy of electrons and muons, reducing the integrals to simple evaluations of the matrix element on the measured events.

## 4 The MEM in practice

In order to use the MEM in an experimental analysis, one has to compute a large number of multi-dimensional integrals such as (1) in a reasonable time scale, which constitutes a difficult numerical problem. While there exist standard tools providing some of the building blocks needed in the computation (such as the matrix elements, the parton distribution functions, a numerical integration library, input/output libraries, . . .), it is in general non-trivial to find an efficient parametrisation of the phase space integrated in (1). Indeed, the integrands present variations of several orders of magnitude across the phase-space, featuring complicated peak structures due to propagators of intermediate particles in the matrix element, and transfer functions on well-reconstructed quantities. This poses a challenge for Monte-Carlo integration algorithms such as VEGAS [21], whose convergence rate critically depends on the integrand being flat and its peaks being separable. Finally, the phase-space density term  $d\Phi_n(y)$  in (1) contains a Dirac delta function (due to conservation of momentum between initial and final states), which cannot possibly be integrated numerically and requires at least some analytical manipulations of the integrand to be thought out.

The problem of efficiently generating an  $n$ -particle phase-space for the purpose of computing process cross sections is easily solved by applying recursive changes of variables allowing to align the peaks due to the propagators in the matrix elements with the integration grid axes. The peaks can then be removed analytically by applying further changes of variables using the inverse primitive of the Breit-Wigner distribution. However, this method is not suited for the MEM due to the additional presence of the transfer function, for which the enhancements directly depend on the final particles' momenta and therefore are already aligned with the integration grid in the standard phase-space parametrisation, written here in polar coordinates:

$$d\Phi_n(y) = \prod_{i=1}^n \frac{d(\cos \theta_i) d\phi_i |p_i|^2 dp_i}{2E_i(2\pi)^3} (2\pi)^4 \delta^4 \left( q_1 + q_2 - \sum_{i=1}^n p_i \right) \quad (3)$$

Clearly, aligning a propagator peak would inevitably spoil the alignment of some components of the transfer function. While this is not worrisome for badly measured quantities such as e.g. neutrino momenta or jet energies, it is highly undesirable in the case of lepton or jet angular variables, for which the transfer function is very sharp and often modelled as a Dirac delta.

### **MADWEIGHT**

While several process-specific implementations of the MEM exist (as can be seen from the previous section), the problem of automatically finding an efficient phase-space parametrisation and computing the required integrals has been solved in the MADWEIGHT package [22], shipped with the MADGRAPH5\_AMC@NLO [23] collection of tools for high-energy physics. MADWEIGHT uses leading-order matrix elements generated automatically by MG5\_AMC@NLO, which allows computing likelihoods for almost any user-specified hypothesis. The integration is performed using the VEGAS algorithm.

In MADWEIGHT, the standard phase-space parametrisation (3) is divided into so-called *blocks* (sets of variables), to each of which is applied a change of variables designed to align as many peaks as possible with the integration grid. At the very least, a *main block* is used to remove the Dirac delta of momentum conservation and align some of the propagator peaks. If need be, *secondary blocks* are chosen to remove further propagators. The requirement that the mappings defining the changes of variables be analytically invertible limits the number of possible main and secondary blocks, however there is no limit to the number of secondary blocks that can be used together. If the problem at hand features too many constraints (propagators, thin transfer functions), it might be impossible to remove all the peaks simultaneously. In those cases, MADWEIGHT resorts to a multi-channel Monte-Carlo integration [24]. The integral is computed several times using different parametrisations aligning different sets of peaks, while multiplying the integrands by weight functions chosen to be larger when the parametrisation is successful in removing the peaks, in a way similar to what is done in the event generation software MADEVENT [25]. The result is a weighted average of the different computations.

Though MADWEIGHT is general and automatic, it does not fully address the needs of real experimental analyses. Indeed, full automation prevents users from tweaking the phase-space parametrisation or the matrix element (for instance, to implement approximations), or from writing their own parametrisation in cases where the algorithm fails to find any efficient combination of main and secondary blocks. To boot, it uses inefficient and error-prone text files-based input/output. Finally, it is neither in development nor supported anymore.

## MoMEMTA

The previous considerations about MADWEIGHT have spawned us to launch a new project from scratch: MoMEMTA<sup>1</sup> (Modular Matrix Element Method implementation). The goal of this project is to provide a *general*, yet not *automatic* framework allowing to compute likelihoods under virtually any hypothesis, while leaving all freedom to the user to configure the integration or link the tool with their analysis code. MoMEMTA builds on widely used and well-maintained tools such as ROOT<sup>2</sup>, CUBA [26] (Monte-Carlo integration library) and LHAPDF [27]. It is modular in the sense that the user steers the integration by writing a small configuration file in the LUA<sup>3</sup> scripting language. The script describes the structure of the integrand through a set of *modules* whose inputs and outputs are linked together. Modules are implemented as C++ classes, making it easy for users to write additional modules tailored to their needs. MoMEMTA can be interfaced with both C++ and Python code.

As an example, consider the following integral to be computed:  $\int_{-\infty}^{\infty} \exp(-x^2) dx$ . Since the integration algorithms used by MoMEMTA integrate on an unit hypercube, it is first necessary to define a phase-space mapping  $\phi(y) : y \rightarrow x$ , in order to change variables from  $x$  to  $y \in ]0, 1[$ . Taking into account the Jacobian due to this change of variable, the integral now reads  $\int_0^1 \exp(-\phi(y)^2) |\phi'(y)| dy$ . In MoMEMTA, this integral could be computed by first defining a module responsible for the change of variable: given as input the “phase-space point”  $y \in ]0, 1[$ , it would return as outputs both  $\phi(y)$  and the jacobian  $|\phi'(y)|$ . Further modules would be responsible for squaring a number, taking its opposite, and computing its exponential. A final module would multiply two numbers (in this case, the exponential and the Jacobian) and finally, its result would be returned to MoMEMTA.

Similarly to the above simple example, complicated integrands such as (1) can be defined in MoMEMTA by splitting them into simple elements, such as parts of the phase-space parametrisation, transfer functions, matrix element, . . . , each of which is taken care of by a module. MoMEMTA ships

<sup>1</sup><https://momemta.github.io/>

<sup>2</sup><https://root.cern.ch>

<sup>3</sup><http://lua.org>

with a comprehensive set of modules covering the most common needs. In particular, it includes modules implementing MADWEIGHT's blocks (plus a new block that had not been identified in [22]), allowing to easily use highly efficient phase-space parametrisations.

As part of the MoMEM<sub>TA</sub> project, a plugin for MG5\_AMC@NLO has been developed. This plugin allows to easily export any leader-order matrix element generated by MG5\_AMC@NLO into a C++ format suitable for use in MoMEM<sub>TA</sub>. Naturally, use of other matrix elements is possible provided they are wrapped for MoMEM<sub>TA</sub>.

## 5 Conclusions

The Matrix Element Method has been described and its successful uses at the LHC so far have been reviewed. The MADWEIGHT software, able to efficiently and automatically evaluate likelihoods under almost any hypothesis, has been presented. Since it is no longer maintained and difficult to use in large scale analysis, we have launched the MoMEM<sub>TA</sub> project: a modular framework retaining MADWEIGHT's generality and effectiveness while being more adaptable to the user's needs.

## Acknowledgements

We warmly thank the organisers of the XIIth Quark Confinement and the Hadron Spectrum, in particular Tommaso Dorigo, for giving us the opportunity to present this work. Our gratitude goes to Sébastien Brochet, Alexandre Mertens, Alessia Saggio and Miguel Vidal for their substantial contributions, and to Pierre Artoisenet and Vincent Lemaître for their useful feedback. This work is supported through a FRIA grant by the FNRS.

## References

- [1] J. Neyman, E.S. Pearson, *Phil. Trans. R. Soc. Lond. A* **231**, 289 (1933)
- [2] I. Volobouev (2011), data-an/1101.2259
- [3] K. Kondo, *Journal of the Physical Society of Japan* **57**, 4126 (1988)
- [4] K. Kondo, *Journal of the Physical Society of Japan* **60**, 836 (1991)
- [5] V.M. Abazov et al. (D0), *Nature* **429**, 638 (2004), hep-ex/0406031
- [6] V. Khachatryan et al. (CMS), *Eur. Phys. J.* **C75**, 251 (2015), hep-ex/1502.02485
- [7] G. Aad et al. (ATLAS), *Eur. Phys. J.* **C75**, 349 (2015), hep-ex/1503.05066
- [8] V. Khachatryan et al. (CMS) (2016), CMS-PAS-HIG-16-038
- [9] V. Khachatryan et al. (CMS) (2016), CMS-PAS-HIG-16-022
- [10] G. Aad et al. (ATLAS), *Phys. Lett.* **B740**, 118 (2015), hep-ex/1410.0647
- [11] V. Khachatryan et al. (CMS) (2013), CMS-PAS-TOP-13-009
- [12] G. Aad et al. (ATLAS), *Phys. Lett.* **B756**, 228 (2016), 1511.05980
- [13] V. Khachatryan et al. (CMS), *Phys. Lett.* **B758**, 321 (2016), 1511.06170
- [14] S. Chatrchyan et al. (CMS), *Phys. Lett.* **B716**, 30 (2012), hep-ex/1207.7235
- [15] G. Aad et al. (ATLAS), *Phys. Rev.* **D91**, 012006 (2015), hep-ex/1408.5191
- [16] S. Chatrchyan et al. (CMS), *Phys. Rev.* **D89**, 092007 (2014), hep-ex/1312.5353
- [17] G. Aad et al. (ATLAS), *Eur. Phys. J.* **C75**, 476 (2015), [Erratum: *Eur. Phys. J.*C76,no.3,152(2016)], 1506.05669
- [18] G. Aad et al. (ATLAS) (2016), ATLAS-CONF-2016-079
- [19] V. Khachatryan et al. (CMS), *Phys. Lett.* **B759**, 672 (2016), 1602.04305

- [20] S. Chatrchyan et al. (CMS) (2016), CMS-PAS-HIG-16-033
- [21] G.P. Lepage, *Journal of Computational Physics* **27**, 192 (1978)
- [22] P. Artoisenet, V. Lemaître, F. Maltoni, O. Mattelaer, *J. High Energy Phys.* **2010** (2010), hep-ph/1007.3300
- [23] J. Alwall et al., *JHEP* **07**, 079 (2014), hep-ph/1405.0301
- [24] R. Kleiss, R. Pittau, *Comput. Phys. Commun.* **83**, 141 (1994), hep-ph/9405257
- [25] F. Maltoni, T. Stelzer, *JHEP* **02**, 027 (2003), hep-ph/0208156
- [26] T. Hahn, *Comput. Phys. Commun.* **168**, 78 (2005), hep-ph/0404043
- [27] A. Buckley et al., *Eur. Phys. J.* **C75**, 132 (2015), hep-ph/1412.7420