The Matrix Element Method at the LHC: status and prospects for Run II

Sébastien Wertz
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
E-mail: sebastien.wertz@uclouvain.be

Abstract. The Matrix Element Method (MEM) is a powerful multivariate method allowing to maximally exploit the experimental and theoretical information available to an analysis. Applications of the MEM at LHC experiments are discussed, such as searches for rare processes and measurements of properties of the Standard Model Higgs boson. The MadWeight software, allowing for a fast and automated computation of MEM weights for any user-specified process, is briefly reviewed. A new implementation of the MEM in the C++ language, MoMEMta, is presented. Building on MadWeight's tricks to accelerate the calculations, it aims at a much improved modularity and maintainability. Examples of this modularity are discussed: the possibility to compute several weights in parallel (propagation of systematic uncertainties), the Differential MEM (DMEM), and a novel way to search for non-resonant New Physics.

1. Introduction
The LHC has restarted last year at the record center-of-mass energy of 13 TeV and has already delivered an integrated luminosity of more than 2 fb\(^{-1}\). Over the next two years, it is expected that the ATLAS and CMS experiments will collect about 50 times more data. As during LHC Run I, physics signals can be very hard to extract out of the huge backgrounds, requiring the use of advanced multivariate techniques. While Machine-Learning (ML) techniques such as Boosted Decision Trees (BDT) and Artificial Neural Networks (ANN) have become increasingly popular, the Matrix Element Method (MEM), pioneered at the Tevatron [1, 2, 3], represents a powerful alternative. Its main advantage is that the achieved discrimination is not limited by the amount of available simulated Monte-Carlo events, which can be an issue in searches for rare processes.

2. The Matrix Element Method
Being able to compute the likelihood \(P(x|\alpha)\) to observe an event \(x\) in a detector under a certain theoretical hypothesis \(\alpha\) would give direct access to powerful tools such as:

- The Maximum likelihood fit of the sample likelihood \(\mathcal{L} = \prod_{i\in\text{data}} P(x_i|\alpha)\), allowing to measure a physical parameter in the model.
- The Neyman-Pearson discriminant \(P(x|\alpha)/P(x|\beta)\), the most powerful discriminant for hypothesis \(\alpha\) against hypothesis \(\beta\) [4], which can be used either for discrete hypothesis testing or as a discriminator in searches for rare processes.
This likelihood\(^1\) can actually be computed, as is obtained through the convolution of the theoretical likelihood for a partonic configuration \(y\) with so-called “transfer functions” and efficiency terms (defined below) linking \(y\) to the event \(x\):

\[
P(x|\alpha) = \frac{1}{A_\alpha \sigma_\alpha} \int d\Phi(y) \frac{dx_1 dx_2}{x_1 x_2 s} f(x_1) f(x_2) |M_\alpha(y, x_1, x_2)|^2 W(x|y) \epsilon_\alpha(y)
\]

where:

- \(M_\alpha\) is the matrix element (usually LO) under hypothesis \(\alpha\).
- The transfer function \(W(x|y)\) gives the probability density that the selected event \(y\) ends up as the measured event \(x\). It describes parton shower, hadronisation, and detector and reconstruction effects. It is normalised as \(\int dx W(x|y) = 1\). The probability that \(y\) ends up as a selected event is included in the efficiency term \(\epsilon_\alpha(y)\).
- \(f\) and \(x_i\) are the Parton Distribution Functions (PDFs) and Björkén-\(x\); \(d\Phi(y)\) is the phase-space density term; \(s\) is the hadronic center-of-mass energy.
- The denominator \(A_\alpha \sigma_\alpha\), including 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 \([5]\).

3. Use of the MEM at the LHC

Several analyses carried out by the ATLAS and CMS experiments during LHC Run I have successfully used the MEM and are briefly summarised here.

**Searches for the \(t\bar{t}\)h process**

ATLAS and CMS have performed multiple searches for this process, in different final states, using either ML and cut-and-count techniques \([6, 7, 8]\), the MEM \([9]\), or a combination of ML and MEM \([10]\). Both MEM analyses concerned the \(h \to b\bar{b}\) and semi-/fully-leptonic \(t\bar{t}\) final state. Dedicated implementations of the MEM computation were used.

**Search for \(s\)-channel single Top quark production**

This rare process at the LHC has been searched for using BDTs, and upper limits on its cross section have been set by ATLAS \([11]\) and CMS \([12]\). ATLAS has repeated the search \([13]\) for this process by re-analysing its 8 TeV dataset using updated calibrations and the MEM. ATLAS was able to increase the expected/observed significance for the process in the background-only hypothesis, up from 1.4/1.3\(\sigma\) in \([11]\) to 3.9/3.3\(\sigma\). Again, an in-house implementation of the MEM was used.

**Measurement of \(t\bar{t}\) spin correlation**

CMS has measured \([14]\) the correlation between pair-produced Top quarks in the \(\mu+\text{jets}\) channel using a MEM, yielding the most precise measurement of this quantity in this channel. Additionally, the MEM was used to perform hypothesis testing, measuring the compatibility of the data with the correlated or uncorrelated spins hypotheses. The MadWeight software was used for the computation of the likelihoods.

**Discovery of the Higgs boson and spin-parity measurements**

ME techniques (“MELA”) were used by CMS in the discovery of the SM Higgs boson, to increase the sensitivity of the search for the \(h \to ZZ^* \to 4l\) process \([15]\). Similar tools were used by both ATLAS and CMS to constrain the spin-parity properties of the Higgs boson \([16, 17]\). Contrary to the three previous use cases presented this section, there is no need to compute integrals such as \((1)\) here, but only to evaluate the matrix element itself on the measured events.

\(^1\) Called “weight” in general, to include use cases where there is no need for proper normalisation.
4. Implementation of the MEM

Any implementation of the MEM must include a common set of building blocks: matrix element evaluation, transfer functions, Monte-Carlo (MC) integrator, calls to PDFs, phase-space parametrisation, and input/output. While most of these are fairly simple to implement, it is in general non-trivial to parametrise the phase-space in an efficient way, since the integrand of (1) includes sharp peaks (propagators in the matrix element, transfer functions) and represents a challenge for adaptive MC integration algorithm.

As can be seen from the previous section, several dedicated implementations, suited to the analyses’ specific needs (input/output, reduced set of processes, ...), have been developed. On the other hand, there also exists a fully general code, able to compute weights under almost any user-specified hypothesis: MadWeight.

MadWeight

MadWeight [18] is part of the MadGraph5_aMC@NLO [19] software suite. It relies on Vegas [20] to compute the integrals and builds on smart phase-space mappings allowing to align the integrand’s peaks with the coordinate axes and analytically remove some of the peaks, greatly improving the convergence rate of the calculations (since Vegas assumes a separable weight function). While being extremely general and fully automatic, it is difficult to use in large-scale experimental analyses, due to inefficient input/output (text files), lack of active development, use of old Fortran syntax, and full automation preventing the user to implement useful tweaks or approximations. This observation has spawned us to launch the MoMEMta project.

MoMEMta

MoMEMta (Modular Matrix Element Method implementation) aims at a C++ framework allowing to compute weights under any desired hypothesis, keeping MadWeight’s highly efficient parametrisations while featuring a modular architecture enabling easy tailoring to the user’s specific needs (new features, interface to analysis code, ...). A set of modules (phase-space mappings, transfer functions, ...), as well as the possibility to quickly develop additional ones, will be provided. The code relies on Root 2 CUBA [21] (Monte-Carlo integration library) and LHAPDF 3, while the configuration of the modular structure is achieved using the Lua scripting language. Modules are implemented as C++ classes. Linking of the modules is achieved using a memory pool built at startup, yielding minimal overhead during the computation compared to the non-modular code. The present-day status of the project can be summarised as follows:

- A non-modular prototype has been written to ensure our understanding of the computation in a particular topology (fully-leptonic $t\bar{t}$ production) and was validated against MadWeight.
- The modular framework is being actively developed and the code (with a working proof-of-principle) is already freely available[4].
- A new MadGraph matrix element export module in the C++11 language has been created. It allows easier linking of the output code to a larger project, and the resulting code is significantly faster than the default C++ exporter’s. However, we would like to stress that MoMEMta is being designed to allow inclusion of any matrix element, not solely restricted to MadGraph’s format.

---

2 https://root.cern.ch
3 http://lua.org
4 The project can be retrieved at https://github.com/MoMEMta/MoMEMta
5. Example applications

This section discusses uses of the MEM that have not yet been applied to data analyses, and are targeted to be easily realised using the modular framework. The exploratory results presented here have been obtained using the non-modular prototype of MoMEMtA (while possible using MadWeight, it requires significant input by the authors).

Parallel weights & systematic uncertainties

One of the main challenges when relying on the MEM for an analysis is the evaluation of systematic uncertainties. While some of them can be handled through Monte-Carlo event reweighting (scale and PDF uncertainties, scale-factors, . . .), others require re-evaluations of the weights on slightly modified event contents (chiefly among them Jet Energy Scale and Resolution uncertainties - JES/JER), which affects the computing time needed for the analysis. However, for most events the object selection does not change, and evaluating these uncertainties only amounts to slight variations $E_{\text{rec}}^\pm$ of the event’s reconstructed jet energies $E_{\text{rec}}$. It is therefore possible to define a **vectorial** integrand,

$$
\begin{bmatrix}
  P^+ \\
  P \\
  P^-
\end{bmatrix}
= \int dE_{\text{gen}} |\mathcal{M}(E_{\text{gen}})|^2 \times
\begin{bmatrix}
  W(E_{\text{rec}}^+|E_{\text{gen}}) \\
  W(E_{\text{rec}}|E_{\text{gen}}) \\
  W(E_{\text{rec}}^-|E_{\text{gen}})
\end{bmatrix} \times \ldots,
$$

where the **Vegas** grid remains nearly optimal for all components (variations being small). Since only the transfer functions need to be re-evaluated on each phase-space point in (2), not the matrix element (by far the slowest part), this procedure can bring a significant speed-up.

Differential MEM

While (1) provides a way to obtain a per-event likelihood, it is also possible to compute a per-event **differential** likelihood w.r.t. a variable $Z$, as was first proposed in [23]:

$$
\frac{1}{P} \frac{\partial P(x)}{\partial Z} \bigg|_{Z_0} = \int d\Phi(y) \frac{dx_1 dx_2}{x_1 x_2 s} f(x_1) f(x_2) |\mathcal{M}(y)|^2 W(x|y) \delta(Z(y) - Z_0)
$$

In practice, this amounts to filling a histogram, for each event, with the values of the variable $Z(y)$ during the computation of the weights $P$. Stacking these histograms yields a distribution of variable $Z$. The procedure can prove useful in cases where ambiguities prevent proper reconstruction of kinematical variables, such as in the fully-leptonic decay of a $t\bar{t}$ pair.

**Figure 1.** Distribution of $m_\ell$ obtained on a simulated sample of fully-leptonic $t\bar{t}$ events. Blue: true distribution. Red: distribution obtained using DMEM.

**Figure 2.** Distributions obtained through DMEM when adding mixtures of $H \rightarrow t\bar{t}$ events to a sample of SM $t\bar{t}$ events ($m_H = 875$ GeV).
Search for higher-dimensional operators

If new physics is too heavy to be produced directly at the LHC, its effects may be parametrised through so-called effective operators giving rise to new couplings amongst SM particles [24, 25, 26, 27]. In the case of \( t\bar{t} \) production, considered here, the dominant effect is expected from the interference of dimension 6 operators with SM amplitudes. The experimental goal is then to extract a global fit of these operators’ couplings from data. While possible using cross-section and asymmetry measurements, we are investigating the potential of the MEM. Denoting by \( \hat{P}_i \) the “weight” obtained by integrating as in (1) the interference between operator \( i \) and the SM amplitude, and \( P_{t\bar{t}} \) the weight under the SM hypothesis, the variable

\[
D_i = \frac{\arctan(\log(|\hat{P}_i|/P_{t\bar{t}})) + \pi/2}{\pi}
\]

is equivalent to a Neyman-Pearson discriminant between the hypotheses “Operator \( i \) (with a certain coupling) interfering with \( t\bar{t} \) in the SM” and “SM \( t\bar{t} \) production”, while being conveniently bound between 0 and 1, and avoiding the need to normalise the weights as likelihoods.

Preliminary studies (in the fully-leptonic final state) were performed based on a set of 7 operators defined and implemented in [28, 29], using MadGraph5 _amc@NLO, Pythia8 [30, 31] and Delphes [32]. Event samples were generated by considering only the interference between the SM and one operator at a time, yielding a set of signals whose contributions are linear in the operators’ couplings, avoiding the need to rely on techniques such as clustering to reduce the number of samples. The constraining power of the \( D_i \) variables can then be probed by template-fitting the distributions obtained with the generated samples. While some gain over using simple kinematical variables is already visible when considering single operators in the \( t\bar{t} \), we are still in the process of investigating the use of these variables in a global fit.

6. Conclusions

The Matrix Element Method has been presented and its successful uses at the LHC so far have been reviewed. While the MadWeight software is able to efficiently and automatically compute weights under almost any hypothesis, it is no longer in development and is difficult to use in a large scale analysis. We have therefore started the MoMEMTa project, with the aim to build a modular framework retaining MadWeight’s generality and effectiveness while being more maintainable and adaptable to the user’s needs. Finally, novel use cases of the MEM, expected to be easily realised using MoMEMTa, have been presented.

Acknowledgments

We warmly thank the organisers of ACAT2016 for giving us the opportunity to present this work. Our gratitude goes to Alexandre Mertens and Sébastien Brochet for their substantial contributions, and to Pierre Artoisenet and Vincent Lemaitre for their useful feedback. This work is supported through a FRIA grant by the FNRS.

References

[1] Kondo K 1988 Journal of the Physical Society of Japan 57 4126-4140
[2] Kondo K 1991 Journal of the Physical Society of Japan 60 830-844
[3] Abazov V M et al. (DO) 2004 Nature 429 638-642 (Preprint hep-ex/0406031)
[4] Neyman J and Pearson E S 1933 Phil. Trans. R. Soc. Lond. A 231 289-337 ISSN 0264-3952
[5] Vebouev I 2011 (Preprint data-an/1101.2259)
[6] Aal G et al. (ATLAS) 2015 Phys. Lett. B749 519-541 (Preprint hep-ex/1506.05988)
[7] Aal G et al. (ATLAS) 2015 Phys. Lett. B740 222-242 (Preprint hep-ex/1409.3122)
[8] Khachatryan V et al. (CMS) 2014 JHEP 09 087 [Erratum: JHEP10,106(2014)] (Preprint hep-ex/1408.1682)
[9] Khachatryan V et al. (CMS) 2015 Eur. Phys. J. C75 251 (Preprint hep-ex/1502.02485)
[10] Aad G et al. (ATLAS) 2015 Eur. Phys. J. C75 349 (Preprint hep-ex/1503.08666)
[11] Aad G et al. (ATLAS) 2015 Phys. Lett. B740 118-136 (Preprint hep-ex/1410.0647)
[12] Khachatryan V et al. (CMS) 2013 CMS-PAS-TOP-13-009
[13] Aad G et al. (ATLAS) 2015 Submitted to Phys. Lett. B (Preprint hep-ex/1511.05980)
[14] Khachatryan V et al. (CMS) 2015 Submitted to Phys. Lett. B (Preprint hep-ex/1511.06170)
[15] Chatrchyan S et al. (CMS) 2012 Phys. Lett. B716 30–61 (Preprint hep-ex/1107.7236)
[16] Aad G et al. (ATLAS) 2015 Phys. Rev. D91 012006 (Preprint hep-ex/1408.5191)
[17] Chatrchyan S et al. (CMS) 2014 Phys. Rev. D89 092007 (Preprint hep-ex/1312.5353)
[18] Artisienet P, Lemaître V, Maltoni F and Mattelaer O 2010 J. High Energy Phys. 2010 (Preprint hep-ph/1007.3300)
[19] Alwall J et al. 2014 JHEP 07 079 (Preprint hep-ph/1405.0301)
[20] Lepage G P 1978 Journal of Computational Physics 27 192
[21] Hahn T 2005 Comput. Phys. Commun. 168 78–95 (Preprint hep-ph/0404043)
[22] Buckley A et al. 2015 Eur. Phys. J. C75 332 (Preprint 1412.7420)
[23] Pin A and Mattelaer O 2012 Nuovo Cim. C035N3 229 TOP2011 Conf. Rep.
[24] Weinberg S 1979 Phys. Rev. Lett. 43(21) 1566–1570
[25] Buchmüller W and Wyler D 1986 Nucl. Phys. B268 621–653
[26] Leung C N, Love S T and Rao S 1986 Zeitschrift für Physik C Particles and Fields 31 432–437 ISSN 1431-5858
[27] Gradkowski B, Ikrzyinski M, Misiek M and Rosiek J 2010 JHEP 10 085 (Preprint hep-ph/1008.4884)
[28] Zhang C and Willenbrock S 2011 Phys. Rev. D83 034006 (Preprint hep-ph/1008.3869)
[29] Degrande C, Gerard J M, Grojean C, Maltoni F and Servant G 2011 JHEP 03 125 (Preprint hep-ph/1010.6304)
[30] Sjostrand T, Mrenna S and Skands P Z 2006 JHEP 05 026 (Preprint hep-ph/0503175)
[31] Sjostrand T, Mrenna S and Skands P Z 2008 Comput. Phys. Commun. 178 852–867 (Preprint hep-ph/0710.3820)
[32] de Favereau J, Delaere C, Denin P, Giannanco A, Lemaître V, Mertens A and Selvaggi M 2014 J. High Energy Phys. 2014 (Preprint hep-ex/1307.6346)