CORE
Metadata, citation and similar papers at core.ac.uk
Provided by Springer - Publisher Connector

Published for SISSA by Springer
Received: August 25, 2015
Accepted: December 9, 2015
Published: January 7, 2016

OPTIMASS: a package for the minimization of kinematic mass functions with constraints

Won Sang Cho,[a,1] **James S. Gainer,**[b] **Doojin Kim,**[b] **Sung Hak Lim,**[a,c,1] **Konstantin T. Matchev,**[b] **Filip Moortgat,**[d] **Luc Pape**[d] **and Myeonghun Park**[e,f,g]

[a] *Center for Theoretical Physics of the Universe, Institute for Basic Science (IBS), Daejeon 305-811, Korea*
[b] *Department of Physics, University of Florida, Gainesville, FL 32611, U.S.A.*
[c] *Department of Physics, Korea Advanced Institute of Science and Technology, 335 Gwahak-ro, Yuseong-gu, Daejeon 305-701, Korea*
[d] *Physics Department, CERN, CH-1211 Geneva 23, Switzerland*
[e] *Asia Pacific Center for Theoretical Physics, San 31, Hyoja-dong, Nam-gu, Pohang 790-784, Korea*
[f] *Department of Physics, Postech, Pohang 790-784, Korea*
[g] *Kavli IPMU (WPI), The University of Tokyo, Kashiwa, Chiba 277-8583, Japan*

E-mail: hepkosmos@ibs.re.kr, jgainer137@gmail.com, immworry@ufl.edu, sunghak.lim@kaist.ac.kr, matchev@phys.ufl.edu, filip.moortgat@cern.ch, luc.pape@cern.ch, parc@ibs.re.kr

Abstract: Reconstructed mass variables, such as M_2, M_{2C}, $M_T^\star$, and M_{T2}^W, play an essential role in searches for new physics at hadron colliders. The calculation of these variables generally involves constrained minimization in a large parameter space, which is numerically challenging. We provide a C++ code, Optimass, which interfaces with the Minuit library to perform this constrained minimization using the Augmented Lagrangian Method. The code can be applied to arbitrarily general event topologies, thus allowing the user to significantly extend the existing set of kinematic variables. We describe this code, explain its physics motivation, and demonstrate its use in the analysis of the fully leptonic decay of pair-produced top quarks using M_2 variables.

Keywords: Supersymmetry, Hadron-Hadron scattering, Beyond Standard Model, top squark, Particle and resonance production

ArXiv ePrint: 1508.00589

[1] Corresponding author.

Open Access, © The Authors.
Article funded by SCOAP[3].

doi:10.1007/JHEP01(2016)026

JHEP01(2016)026

Contents

1 Introduction **1**

2 Review of constrained minimization **5**
- 2.1 The method of Lagrange multipliers 7
- 2.2 Penalty methods 9
- 2.3 Augmented Lagrangian Method 10

3 The OPTIMASS code **11**
- 3.1 Operational algorithm 12
  - 3.1.1 Step one: initialization 12
  - 3.1.2 Step two: unconstrained minimization with MINUIT 15
  - 3.1.3 Step three: ALM parameter adjustment 16
- 3.2 Validation 17
  - 3.2.1 Example one 17
  - 3.2.2 Example two 20

4 Calculating M_2 variables with OPTIMASS **21**
- 4.1 Introduction to M_2 21
- 4.2 Calculating M_2 for dilepton top events 24
- 4.3 Demonstration of OPTIMASS for one event 27

5 Conclusions **27**

A Unconstrained minimization with MINUIT **30**
- A.1 MIGRAD 30
- A.2 Simplex 31

B Installation and user instructions **32**

1 Introduction

The CERN Large Hadron Collider (LHC) has successfully completed the decades-long quest to discover the particles of the Standard Model (SM) by finding the Higgs Boson [1, 2]. The paramount question in the current Run 2 of the LHC is whether the LHC can reach the relevant energy scale to discover physics beyond the standard model (BSM). Popular frameworks for new physics such as Supersymmetry (SUSY) [3–6] and Universal Extra Dimensions (UED) [7, 8] predict:

JHEP01(2016)026

1. The presence of particles, such as neutralinos or KK-photons, that are not reconstructed in the detector, and are hence termed "invisible". In general, the production of these particles will lead to "missing transverse energy"[1] (MET) in an event.

2. Relatively complex decay topologies, in which pair-produced, generally colored, particles undergo several subsequent decays. Each "decay chain" thus produces one or more visible particles, as well as at least one invisible particle.

Searches for new particles that produce such long decay chains in combination with a MET signature are complicated by large backgrounds from $t\bar{t}$, W, and Z production, often with additional jets from initial state radiation (ISR). The severity of these backgrounds in multijet and multilepton channels increases with the collider energy. Even if a signal of such new physics is seen, the corresponding measurements of particle properties such as masses, couplings, and spins, are highly nontrivial [9–11]. Therefore, sophisticated procedures must be used to separate signal from background and to extract the quantum numbers of the new particles.

In a given event, one observes some number of "physics objects" which correspond to physical particles that have produced appropriate energy deposits in the tracker and calorimeters of the detector; we will refer to these objects as "visible particles". We shall denote their measured four-momenta by p_j^μ, where j is the visible particle label. At the same time, the existence of a non-vanishing MET in the event indicates the presence of some number of additional, invisible, particles with four-momenta q_k^μ, where k now labels the invisible particles. The individual momenta q_k^μ are not measured, and the only available piece of information is the missing transverse momentum

$$\vec{\not{p}}_T \equiv \sum_k \vec{q}_{Tk} = -\sum_j \vec{p}_{Tj}. \tag{1.1}$$

The MET, $\not{E}_T$, is then simply the magnitude of the missing transverse momentum vector:

$$\not{E}_T \equiv |\vec{\not{p}}_T|. \tag{1.2}$$

The essential question is how to take these sets of measured four-momenta, $\{p_j^\mu\}_a$, (one for each event a) and determine whether the events are produced purely by SM processes or whether new physics is at work. The standard procedure is to construct some kinematic variable, v, and compare the v distribution predicted by the SM to the data. In choosing a suitable variable, v, one typically follows one of these approaches:

1. *The variable, v, is an analytic function of some, but not all, of the measured momentum degrees of freedom.* This is the preferred approach in inclusive analyses, where one targets a specific subset of the event. For example, in an inclusive search for a visibly decaying resonance, X, one would select only the momenta of the hypothesized decay products $j_1, j_2, \ldots, j_n$, and form the invariant mass of the resonance X,

$$M_X \equiv \sqrt{(p_{j_1} + p_{j_2} + \ldots + p_{j_n})^2}, \tag{1.3}$$

leaving out all other objects in the event.[2] In the case of missing energy events,

[1] Or more precisely, missing transverse momentum.

[2] In forming the variable (1.3), one also ignores, e.g., additional angular information.

JHEP01(2016)026

the situation is much more complicated — we cannot reconstruct the mass of the resonance as in eq. (1.3), since we have not measured the momenta, q_k^μ, of the invisible decay products. Then, one typically tries to form a variable which correlates with the scale of M_X. Various candidates have been tried, including the transverse momentum of the hardest object of a given type (lepton, jet, etc.) [12, 13], the scalar p_T sum of the four hardest jets (or of all jets) [14], the jet multiplicity [15, 16], the "fat" jet mass [17], the "contransverse mass", M_{CT} [18–20], the lepton energy [21–23] or lepton energy ratios [24, 25], and many more. The advantage of such techniques is their simplicity and robustness — they do not involve too many theoretical assumptions, making them ideal for model-independent searches for new physics. At the same time, they appear to be suboptimal, since they do not utilize the full set of measured degrees of freedom, leading to a certain loss of information. It is also rather challenging to assign a proper physical meaning to a kinematic variable which only uses such partial information (for more detailed discussion, see refs. [14, 26, 27]).

2. *The variable, v, is an analytic function of some measured momentum degrees of freedom and the measured $\not{E}_T$.* The explicit inclusion of the measured $\not{E}_T$ in the definition of v was the next attempt to design a better performing class of variables. Perhaps the best known example is the W transverse mass [28, 29], where one identifies the transverse momentum of the missing neutrino with the measured $\vec{\not{p}}_T$. Other possibilities include the "effective mass", M_{eff} [14, 30], the $\sqrt{\hat{s}}_{\text{min}}$ variable [31–33], and the "razor" variables [34–37]. The outputs of neural nets, boosted decision trees, and other multivariate analyses [38], particularly those involving some form of machine learning, are also variables in this class. Incorporating the measured $\not{E}_T$, which is often a sensitive variable all by itself [39, 40], into the definition of the kinematic variable, v, is certainly a step in the right direction.

3. *The definition of the variable, v, involves all measured momentum degrees of freedom and the individual invisible momenta, q_k^μ.* Finally, one may construct the variable, v, so that from the onset it has explicit dependence on the individual invisible momenta, q_k^μ. The advantage of this approach is that one works with theoretically motivated quantities with clear physical meaning [26]. The obvious downside is that the individual invisible particle momenta, q_k^μ, are unknown, and something must be done to fix their values in the calculation. There are two possible alternatives:

   - *Integrate over all possible values of the invisible momenta.* Perhaps the simplest solution is to allow all possible values of the invisible momenta, q_k^μ, which are consistent with the measured missing transverse momentum (1.1), and compute the variable, v, as a suitably weighted average. A celebrated example of this approach is the Matrix Element Method (MEM) [41–43] which is finding increased use in hadron collider physics [44–53]. However, the method is often too computationally challenging, requiring novel ideas and approaches [54–56]. Additionally, it is generally difficult to incorporate "reducible" backgrounds, which

JHEP01(2016)026

consist of events where the reconstructed particles are misidentified and/or their momenta significantly mismeasured.

- *Use a physically motivated ansatz for the invisible momenta.* Alternatively, instead of considering all possible values of the set of invisible momenta, $\{q_k^\mu\}$, one could fix them by following some prescription specified in advance. With this approach, one gives up on trying to "guess" the correct values of the invisible momenta, and instead focuses on constructing a useful variable, v, whose properties can reveal important information about the underlying physics. Examples of such variables include the Cambridge M_{T2} variable [57, 58] and its variants [59–61], and the variables M_{2C} [62, 63], M_{CT2} [64, 65], $M_T^\star$ [66], M_{T2}^W [67], and $M_{\rm min}$ [68]. The variables in this class are often specified, not by analytic formulae, but by the algorithm used to calculate them.[3]

Our focus in this paper will be on the algorithmically specified variables from the very last category, which are known to possess several attractive features:

1. They are "maximally constraining" [26, 76] in the sense that, on an event per event basis, they provide the best possible lower bound on an invariant mass quantity of interest, such as the parent masses or the center-of-mass energy, $\sqrt{\hat{s}}$. This is particularly useful in cases where it is not possible to determine the actual values of that quantity due to incomplete event information.

2. Their kinematic distributions exhibit sharper endpoints which are easier to measure over the SM backgrounds [77], leading to a more precise determination of the new physics mass spectrum.

3. Certain measurements of their properties can be used as a self-consistency check on the assumed signal event topology [77, 78]. If the check fails, our conjecture about the event topology is falsified, thus narrowing down the allowed set of possibilities.

At the same time, such algorithmically defined variables are notoriously difficult to compute. The algorithmic procedure typically involves calculating a mass for a set of hypothesized particles in an event (possibly after projecting their momenta onto the transverse plane), and minimizing the value of that mass with respect to *all* invisible momenta, q_k^μ. What makes the problem particularly challenging, however, is the presence of additional non-linear mass-shell constraints. In essence, we are faced with a multidimensional constrained optimization problem, where our objective function is an energy function which is to be minimized. Given the importance of the maximally constraining invariant mass variables for new physics searches and measurements [26, 76–78], it is important to have publicly-available software for performing constrained minimizations to calculate kinematic variables from high energy collision data with sufficient generality and efficiency. The existing packages described in the literature are typically only applicable to a specific variable,

[3] For some of the variables, analytical formulas may exist in certain special cases [69–75], but not in general.

JHEP01(2016)026

e.g., M_{T2} [58, 79–81], or M_{T2}^{W} [67] and cannot be readily generalized to the whole class of on-shell constrained variables [76, 77]. The standard approach is to try to solve the constraining equations, thus reducing the unknown number of degrees of freedom (d.o.f.), then implement an unconstrained minimization over the remaining d.o.f. While this approach generally provides the most efficient algorithm for a specific event topology, it is not extendable to more general event topologies, where not all constraints can be simultaneously solved analytically.

In this paper, we describe an alternative approach that is sufficiently universal and flexible, and can be applied to arbitrarily general event topologies. The main idea is to use the Augmented Lagrangian Method (ALM) [82, 83], briefly described below in section 2.3. In this approach, the feasibility (i.e., the validity of the constraints) is ensured by penalizing infeasibility by adding "penalty terms" to the objective function, rather than by directly solving the constraining equations. The fact that the method does not require the solving of any constraints beforehand makes it very flexible and applicable to a very general class of event topologies. Of course, we still have to perform a standard unconstrained minimization, for which we can take advantage of any one of the many publicly available packages — we have chosen to use Minuit [84], which is widely popular in high energy physics. We also supply a comprehensive software package, Optimass,[4] in the form of a library, which interfaces with Minuit to perform the constrained minimization of a user-specified kinematic function using the ALM. Appendix B contains instructions on the installation and usage of Optimass.

The paper is organized as follows. In section 2, we review the general problem of constrained optimization with special emphasis on the motivation and the techniques used by Optimass, in particular the ALM. The relevant Minuit routines with which it interfaces are described in appendix A. Section 3 describes in detail the algorithm behind the Optimass package and presents several toy examples for its validation. In section 4 we briefly review the M_2 variables [26, 76, 77], and provide examples of their use, in the study of the fully leptonic decays of pair-produced top quarks. We use this as an opportunity to compare the results from Optimass with previous studies and known analytical calculations. Finally, section 5 is reserved for our conclusions and a brief discussion of the future of Optimass.

2 Review of constrained minimization

While many excellent textbooks and references discuss the optimization techniques that we will utilize (see, e.g., refs. [85–87] and references therein), we feel that a brief review of the elements of optimization theory relevant to the operation of Optimass will prove useful. We first note that while we will sometimes speak of "optimization" rather than "minimization": (i) in the calculation of kinematic variables we will be interested only in minimization, and (ii) the methods used to find a maximum are, up to obvious changes in the sign of certain parameters, identical to those used to find a minimum. So in what follows we will not worry much about this distinction.

[4] For OPTImization of MASS variables.

JHEP01(2016)026

The first issue that *will* concern us is the important division of minimization problems into two types: (i) constrained and (ii) unconstrained. In constrained minimization, we want to minimize an *objective function*,

$$f(\vec{x}), \tag{2.1}$$

subject to a set of m *constraints*

$$c_a(\vec{x}) = 0, \quad a = 1, \ldots, m, \tag{2.2}$$

where $\vec{x}$, in general, refers to a point in $\mathbb{R}^n$ formed from some unknown momentum degrees of freedom $x_1, x_2, \ldots, x_n$. In what follows, we shall assume that the number of constraints, m, is always less than the number of independent degrees of freedom, n, so that we are dealing with a true minimization problem.

If the constraints in eq. (2.2) are all independent, then the parameter space is effectively reduced from n dimensions to $n-m$ dimensions. Sometimes this reduction can be performed analytically. For example, consider an optimization problem in $\mathbb{R}^3$, subject to the constraint $x^2 + y^2 + z^2 = 1$ — then we should parameterize the two dimensional subspace that satisfies the constraints, i.e., the surface of the unit sphere, S_2, by the angles θ and φ in the standard way.

However, this reduction of dimensionality cannot always be performed analytically. A useful alternative procedure, therefore, is to turn the constrained minimization problem into the problem of an unconstrained minimization[5] of *a modified objective function*, $\tilde{f}(\vec{x})$, over the full, unconstrained, parameter space, $\mathbb{R}^n$. This new problem can then be solved by setting the gradient of some function equal to zero, or by searching for a local/global minimum using one of the many standard numerical algorithms conventionally used for this purpose.

When performing this unconstrained minimization iteratively, one develops an algorithm for finding the location of the minimum of $\tilde{f}(\vec{x})$, $\vec{x}^*$, which is also referred to as the *minimizer*. At each iteration, one starts with some initial estimate, $\vec{x}_k$, (typically taken to be the minimizer of the previous, $k-1^{\text{st}}$, iteration), then refining this estimate by obtaining a new minimizer, $\vec{x}_{k+1}$, in some prescribed way, until certain convergence criteria are met. Since we have not analytically solved the constraints (2.2), the estimates, $\vec{x}_k$, *will not*, in general, satisfy the constraints exactly. Following the standard mathematical terminology, we shall refer to values of $\vec{x}$ that satisfy the constraints in eq. (2.2) as *feasible*. The absolute value of $c_a(\vec{x})$ is then a measure of the *feasibility*.[6] Even though feasibility is not strictly guaranteed, the ultimate solution found by the method should nevertheless be such that the constraints are satisfied to within a required degree of numerical precision.

[5] Or possibly, a *series of* unconstrained minimization problems.

[6] A point $\vec{x}$ in the unknown momentum space $\mathbb{R}^n$ is feasible if it is an element of the feasible set Ω defined by

$$\Omega = \{\, \vec{x} \,|\, c_a(\vec{x}) \,=\, 0,\, a = 1, \ldots, m \}\,.$$

JHEP01(2016)026

In the remaining three subsections of this section, we discuss three possible ways to transform a constrained minimization problem into an unconstrained minimization problem, namely (i) the method of Lagrange multipliers (in section 2.1), (ii) penalty methods (in section 2.2), and (iii) the Augmented Lagrangian Method (in section 2.3). We shall see how penalty methods solve some of the problems associated with the use of Lagrange multipliers, while the ALM, in turn, resolves certain numerical issues related to the use of penalty methods. Of course we also must be able to solve the resulting unconstrained optimization problem; we discuss approaches to this challenge in appendix A.

2.1 The method of Lagrange multipliers

As noted above, in a generic constrained minimization problem we are looking for the minimum value of the target function, $f(\vec{x})$, subject to the constraints (2.2):

$$f(\vec{x}^{\,*}) \equiv \min_{x\in\mathbb{R}^n} f(\vec{x}) \quad \text{such that} \quad c_{a=1,\ldots,m}(\vec{x}^{\,*}) = 0. \tag{2.3}$$

Alternatively, one is trying to find the location of the minimizer, $\vec{x}^*$, in $\mathbb{R}^n$. We note that in practical applications of the maximally constraining invariant mass variables both the minimum value of the function, $f(\vec{x}^*)$, and the minimizer, $\vec{x}^*$, itself can serve a useful purpose. For example, in the M_{T2}-assisted on-shell (MAOS) reconstruction method, the minimizer is used to provide an ansatz for certain transverse components of the invisible momenta [88, 89]. Both the function, $f(\vec{x})$, and the constraints, $c_a(\vec{x})$, are assumed to be smooth[7] real-valued functions in $\mathbb{R}^n$.

The method of Lagrange multipliers provides necessary and sufficient conditions for finding local solutions of the minimization problem (2.3) above. In this method, we define a corresponding Lagrangian, henceforth denoted by $\mathcal{L}$, for an objective function, $f(\vec{x})$, and constraints, $c_a(\vec{x})$, by

$$\mathcal{L}(\vec{x}, \boldsymbol{\lambda}) = f(\vec{x}) - \sum_{a=1}^{m} \lambda_a\, c_a(\vec{x}), \tag{2.4}$$

where $\boldsymbol{\lambda} \equiv (\lambda_1, \lambda_2, \ldots, \lambda_m)$ is an m-component vector[8] of Lagrange multipliers, λ_a. We now describe the conditions that must be satisfied in this method in order to establish the existence of a local minimum at the proposed minimizer, $\vec{x}^*$.

First order condition (FOC). In unconstrained optimization, a *necessary* condition for the existence of a local minimum of f at $\vec{x}^*$ is that the gradient vector, $\nabla_x f(\vec{x}^*)$, vanishes. As this condition involves the gradient, we may term it a *first order condition*. In the method of Lagrange multipliers, an analogous condition holds for the existence of a local minimum. Here it is only necessary that the gradient of the objective function, f, at $\vec{x}^*$ is orthogonal to the surface defined by the constraints. This condition will hold if the

[7] In some cases, the objective function may depart from smoothness in a specific way; we discuss this point and related issues in sections 3.2.2 and 4.1, see also ref. [77].

[8] Throughout this paper we shall use the notation $\vec{v}$ for n-dimensional vectors in the space of independent variables, $\mathbb{R}^n$, and $\boldsymbol{v}$ to denote m-dimensional vectors in the space of constraints, $\mathbb{R}^m$.

JHEP01(2016)026

gradient of f, $\nabla_x f(\vec{x}^*)$, is an element of the vector space spanned by the gradient vectors of c_a, $\nabla_x c_a(\vec{x}^*)$. Thus the FOC is that there exists a Lagrange multiplier vector, $\boldsymbol{\lambda}^*$, in $\mathbb{R}^m$, such that at the point $(\vec{x}^*, \boldsymbol{\lambda}^*) \in \mathbb{R}^n \otimes \mathbb{R}^m$, the following conditions hold

$$\begin{cases} \nabla_\lambda \mathcal{L}(\vec{x}^*, \boldsymbol{\lambda}^*) = c_a(\vec{x}^*) = 0, \\ \nabla_x \mathcal{L}(\vec{x}^*, \boldsymbol{\lambda}^*) = \nabla_x f(\vec{x}^*) - \sum_a \lambda_a^* \nabla_x c_a(\vec{x}^*) = 0. \end{cases} \tag{2.5}$$

Therefore, at least at the level of the FOC, the problem of constrained optimization has been reformulated as an unconstrained optimization problem.

Second order condition (SOC). As is well-known, the condition that a first derivative vanishes is not sufficient to establish that there is an extremum at that point — one must also verify that the second derivative is, in the case of a minimum, positive. The extension of this idea to unconstrained optimization in many dimensions is to require that the *Hessian matrix*, defined for the objective function, f, by

$$H_{jk}(\vec{x}^*) = \frac{\partial^2 f(\vec{x}^*)}{\partial x_j \partial x_k}, \tag{2.6}$$

and evaluated at the prospective minimizer, $\vec{x}^*$, be positive definite.[9] This is the "second order condition" (SOC); that both the FOC and the SOC are satisfied is sufficient for the existence of a local minimum. As long as we are not at a boundary of the parameter space, these conditions are both necessary and sufficient.

We now consider the corresponding SOC for the Lagrange multiplier method, hoping that, as in the case of the FOC, we can obtain a condition analogous to the case of unconstrained minimization, i.e., a constraint involving the positive definiteness of some Hessian. We therefore evaluate the Hessian of the Lagrange function (2.4) with respect to $\vec{x}$:

$$(H_{\mathcal{L}})_{jk}(\vec{x}^*, \boldsymbol{\lambda}^*) \equiv \frac{\partial^2 \mathcal{L}(\vec{x}^*, \boldsymbol{\lambda}^*)}{\partial x_j \partial x_k} = \frac{\partial^2 f(\vec{x}^*)}{\partial x_j \partial x_k} - \sum_a \lambda_a^* \frac{\partial^2 c_a(\vec{x}^*)}{\partial x_j \partial x_k}. \tag{2.7}$$

To determine the SOC, we note that for $\vec{x}^*$ to be a minimizer (with Lagrange multiplier vector, $\boldsymbol{\lambda}^*$), we must have

$$\vec{d}^T H_{\mathcal{L}}(\vec{x}^*, \boldsymbol{\lambda}^*) \, \vec{d} > 0, \tag{2.8}$$

for any infinitesimal displacement, $\vec{d}$, from the proposed minimizer, $\vec{x}^*$, in a direction allowed by the constraints.

Thus the relevant condition is *not* the positive definiteness of the Hessian (2.7), but the positive definiteness of the restriction of the Hessian to the space allowed by the constraints. The fact that we must still use the constraints to see if a given stationary point (i.e., a point satisfying the FOC) is a minimum, means that we have failed in our mission to convert the constrained minimization problem to an unconstrained minimization problem. We therefore proceed to look for methods for which the SOC does not explicitly involve the constraints.

[9] In linear algebra, a symmetric $n \times n$ real matrix, M, is said to be positive definite if $\vec{z}^T M \vec{z}$ is positive for every non-zero column vector, $\vec{z}$, of n real numbers.

JHEP01(2016)026

2.2 Penalty methods

A natural approach to our problem of transforming a constrained optimization problem to an unconstrained optimization problem is to follow the example of the method of Lagrange multipliers in modifying the objective function, but to do so in a different way. Clearly, we would like to modify the function so that infeasibility incurs a penalty. One possible way to achieve this is via "convexification" of the geometry near the desired solution point, i.e., making sure that a solution to the constrained minimization problem is a local minimum of the transformed function, $\tilde{f}(\vec{x})$, even if it is *not* a local minimum of $f(\vec{x})$ in the absence of constraints. We now proceed to give an example of one such convexification approach.

In the so-called *penalty methods*, the original objective function, $f(\vec{x})$, is modified by the addition of a *penalty term*, i.e., a functional of $c_a(x)$ weighted by a positive penalty parameter, μ, so that the term vanishes when $c_a(\vec{x}) = 0$, but becomes large if $c_a(\vec{x}) \neq 0$ in the $\mu \to 0$ limit. While there are various penalty methods, which differ in the form of the penalty function, we consider the "Quadratic Penalty Method" (QPM) here, because of its connection with the ALM discussed below. In the QPM, the penalty term is chosen so that the modified function[10] under consideration is

$$P(\vec{x}\,;\mu) \equiv f(\vec{x}) + \frac{1}{2\mu}\sum_a c_a^2(\vec{x}). \tag{2.9}$$

In the course of the algorithm, the parameter, μ, will be reduced, as the desired properties of this function hold in the $\mu \to 0$ limit. The gradient of (2.9) is

$$\nabla P(\vec{x}\,;\mu) = \nabla f(\vec{x}) + \sum_a \frac{c_a}{\mu}\,\nabla c_a(\vec{x}), \tag{2.10}$$

which reproduces the gradient of the Lagrange function (2.5) if we take

$$\lambda_a(\mu) \;\equiv -\frac{c_a(\vec{x})}{\mu} \tag{2.11}$$

to be (the components of) the Lagrange multiplier vector, $\boldsymbol{\lambda}$. This shows that the necessary FOC for a minimum is the same as for the method of Lagrange multipliers, only the Lagrange multiplier vector is now determined for us. Since a minimization using a QPM should give the same value for the solution, $\vec{x}^*$, as the Lagrange multiplier method (which gives the solution $(\vec{x}^*, \boldsymbol{\lambda}^*)$), it follows that as we approach this solution, we must have

$$\lim_{\vec{x}\to\vec{x}^*}\left(-\frac{c_a(\vec{x})}{\mu}\right) = \lambda_a^*, \tag{2.12}$$

thus $|c_a(\vec{x})| \to 0$ as $\mu \to 0$.

In determining the SOC for the QPM, we note that the Hessian of the function in (2.9) is

$$H_P(\vec{x}^*)_{jk} = \frac{\partial^2 f(\vec{x}^*)}{\partial x_j \partial x_k} + \frac{1}{\mu}\sum_a \Big((\nabla c_a)_j(\nabla c_a)_k + c_a \nabla^2_{jk} c_a\Big)\Big|_{\vec{x}=\vec{x}^*}. \tag{2.13}$$

[10] From here on, we shall use a semicolon, ";", to separate the arguments of a function into two groups: independent variables, with respect to which an optimization is to be done, and fixed parameters.

JHEP01(2016)026

If $\vec{x}^*$ satisfies the constraints, then this expression simplifies to

$$H_P(\vec{x}^*)_{jk} = \frac{\partial^2 f(\vec{x}^*)}{\partial x_j \partial x_k} + \frac{1}{\mu}\sum_a (\nabla c_a)_j (\nabla c_a)_k \bigg|_{\vec{x}=\vec{x}^*}. \tag{2.14}$$

If we consider an infinitesimal displacement, $\vec{d}$, along the surface allowed by the constraints, then as

$$\vec{d}\cdot\nabla c_a(\vec{x}^*) = 0, \tag{2.15}$$

we find that

$$\vec{d}^{\,T}\, H_P(\vec{x}^*)\,\vec{d} = \vec{d}^{\,T}\, H_f(\vec{x}^*)\,\vec{d}, \tag{2.16}$$

where $H_f(\vec{x})$ is the Hessian (2.6) for the original objective function, i.e., $\partial_j\partial_k f(\vec{x})$.

On the other hand, for displacements, $\vec{d}'$, orthogonal to this surface, we obtain non-zero terms from both the objective function and the penalty term, i.e.

$$\vec{d}'^{\,T}\, H_P(\vec{x}^*)\,\vec{d}' = \vec{d}'^{\,T}\, H_f(\vec{x}^*)\,\vec{d}' + \frac{1}{\mu}\sum_a (\nabla c_a(\vec{x}^*)\cdot\vec{d}')^2, \tag{2.17}$$

where $\nabla c_a(x)\cdot\vec{d}'$ is now non-vanishing. In the $\mu\to 0$ limit, the second term will dominate. So if the Hessian is positive definite with respect to allowed displacements, it is automatically positive definite in general. Thus, we now have that in the limit of $\mu\to 0$, the sufficient condition that the stationary point, $\vec{x}^*$, be a local minimum is simply that the Hessian of the function (2.9) is positive definite. Thus we have succeeded in overcoming the limitation of the method of Lagrangian multipliers in that both the FOC and SOC are the same as in unconstrained minimization.

Unfortunately, all is not well when it comes to practical applications of the QPM. The basic problem is that in the $\mu\to 0$ limit the algorithm becomes too sensitive to small departures from feasibility, hence numerical instabilities may prevent convergence to the solution. In more formal language, small values of $\mu\to 0$ can result in severe *ill-conditioning* of the Hessian, since the rightmost term in (2.14), which dominates in the $\mu\to 0$ limit, is not invertible. Therefore, we will need to further modify the QPM, retaining the way in which it maps constrained optimization problems to unconstrained optimization problems, but reducing the relative importance given to feasibility in the $\mu\to 0$ limit. The solution, presented in the next subsection, is the Augmented Lagrangian Method (ALM).

2.3 Augmented Lagrangian Method

We seek a method which preserves the success of the QPM in generating FOC and SOC that correspond exactly to those obtained in unconstrained minimization, but which overcomes the difficulties encountered by the QPM in the $\mu\to 0$ limit. One approach is to introduce *augmented Lagrange multiplier terms*, leading to the following modified objective function:

$$\tilde{\mathcal{L}}(\vec{x}\,;\boldsymbol{\lambda},\,\mu) \equiv f(\vec{x}) - \sum_a \lambda_a c_a(\vec{x}) + \frac{1}{2\mu}\sum_a c_a^2(\vec{x}). \tag{2.18}$$

Note that $\boldsymbol{\lambda}$ is no longer determined numerically by the optimization procedure, but is instead a fixed vector, just like the penalty parameter, μ. As was the case for the QPM, (2.18)

JHEP01(2016)026

is used iteratively; unconstrained minimization of $\tilde{\mathcal{L}}(\vec{x}\,;\boldsymbol{\lambda},\,\mu)$ is performed for the chosen values of $\boldsymbol{\lambda}$ and μ in each step of the procedure. After optimizing $\tilde{\mathcal{L}}(\vec{x}\,;\boldsymbol{\lambda},\,\mu)$ to within some tolerance, new values of $\boldsymbol{\lambda}$ and μ are chosen; the process is repeated until the desired levels of optimality and feasibility are satisfied. This procedure is the ALM.

We must verify that the ALM indeed avoids the problems of the QPM. To do this, we note that

$$\nabla\tilde{\mathcal{L}}(\vec{x}\,;\boldsymbol{\lambda},\,\mu) = \nabla f(\vec{x}) - \sum_a \lambda_a \nabla c_a(\vec{x}) + \sum_a \frac{c_a}{\mu}\nabla c_a(\vec{x}), \tag{2.19}$$

while the Hessian is given by

$$H_{\tilde{\mathcal{L}}}(\vec{x})_{jk} = \frac{\partial^2 f(\vec{x})}{\partial x_j \partial x_k} - \sum_a \lambda_a \frac{\partial^2 c_a(\vec{x})}{\partial x_j \partial x_k} + \frac{1}{\mu}\sum_a \Big((\nabla c_a)_j (\nabla c_a)_k + c_a(\vec{x})\nabla^2_{jk} c_a(\vec{x}) \Big). \tag{2.20}$$

We note that (2.19) recovers the expression for the method of Lagrange multipliers (2.5) with the substitution

$$\lambda_a \to \lambda_a - \frac{c_a(\vec{x})}{\mu}, \tag{2.21}$$

which shows that the FOCs for optimality are the same as for the problem of unconstrained minimization, just as in the case of the QPM. It is also possible, albeit more challenging, to show that the SOC for a minimum is the same as in the unconstrained case; see refs. [82, 83], and [86, Proposition 4.2.3, Theorem 17.6, "Numerical optimization"] for details.

We also conclude from (2.21) that in the asymptotic limit

$$\lambda^*_a \to \lambda_a - \frac{c_a(\vec{x}^*)}{\mu}. \tag{2.22}$$

in analogy to (2.12). The point of the ALM is that now that we are free to choose both λ_a and μ, we can enforce (2.22) without taking the $\mu \to 0$ limit. We shall do this by choosing the value of λ_a in the $(k+1)^{\text{st}}$ iteration as follows

$$\lambda^{k+1}_a = \lambda^k_a - \frac{c_a(\vec{x}_k)}{\mu_k}; \tag{2.23}$$

As we do not take $\mu_k \to 0$, the Hessian defined in (2.20) should never become ill-conditioned. Hence the ALM avoids the major drawback of the QPM, while preserving its successes. We therefore implement the ALM in the Optimass code, as described in the following section.

3 The Optimass code

Having reviewed both unconstrained and constrained minimization, we now state precisely how Optimass accomplishes the task of minimizing a mass function with constraints. The basic algorithm is presented in figure 1 and is the main subject of this section.[11]

[11] Readers who are not interested in the details of the code may skip directly to the next section.

JHEP01(2016)026

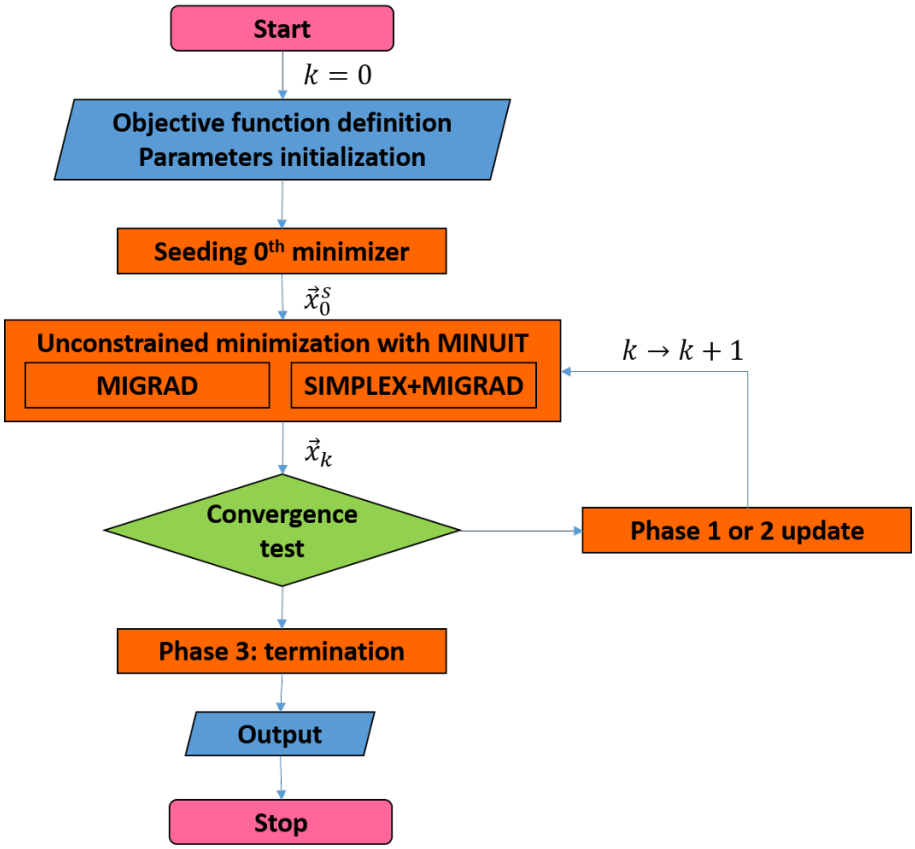

Figure 1. Flowchart of the minimization procedure in OPTIMASS.

3.1 Operational algorithm

3.1.1 Step one: initialization

At the onset, we must specify certain parameters. We first discuss the parameters associated with the optimality condition, followed by those related to the feasibility condition. Finally, we will discuss the penalty/Lagrange multiplier parameter, and the starting value for the minimizer.

Optimality condition. The test of optimality, i.e., whether the relevant sub-minimization procedure, has reached an acceptable local minimum in the kth iteration, should be performed in terms of some relevant geometric quantity. In OPTIMASS, we choose the MIGRAD algorithm of MINUIT (see appendix A.1), where the optimality criterion utilizes the so called "Estimated vertical Distance to the Minimum" (EDM) defined as follows

$$\text{EDM} \equiv \frac{1}{2}\left(\vec{\nabla}\tilde{\mathcal{L}}(\vec{x}_0)\right)^T \cdot H^{-1}(\vec{x}_0) \cdot \vec{\nabla}\tilde{\mathcal{L}}(\vec{x}_0) = \frac{1}{2}\Delta\vec{x}^T \cdot H(\vec{x}_0) \cdot \Delta\vec{x} \approx f(\vec{x}) - f(\vec{x}_0), \quad (3.1)$$

where

$$\Delta\vec{x} \equiv \vec{x} - \vec{x}_0 = -H(\vec{x}_0)^{-1} \cdot \vec{\nabla}\tilde{\mathcal{L}}(\vec{x}_0). \quad (3.2)$$

JHEP01(2016)026

Using the EDM (3.1), at each iteration k in OPTIMASS the optimality convergence test in MINUIT is performed by checking if the EDM is smaller than the optimality tolerance, ω_k:

$$\text{EDM} < \omega_k. \tag{3.3}$$

We have observed that simply setting ω_k to a constant, ω^*, suffices. By default in MIGRAD, this constant is set by the internal re-parametrization

$$\omega^* = 0.001 \cdot \text{TOLERANCE} \cdot \text{UP}. \tag{3.4}$$

In OPTIMASS, the optimality tolerance ω^* is controlled and set by the TOLERANCE parameter through the interface with MIGRAD.[12] Its default value is set to be

$$\text{TOLERANCE} = 0.1, \tag{3.5}$$

while the parameter UP is not used, it is set internally in OPTIMASS to UP = 1.

Feasibility condition. Throughout the whole minimization procedure, we test for feasibility by computing the quantity[13]

$$||c(\vec{x}_k)||^2 \equiv \sum_a c_a^2(\vec{x}_k) \tag{3.6}$$

in each iteration. Then, the test for feasibility is

$$||c(\vec{x}_k)|| < \eta_k, \tag{3.7}$$

where η_k denotes the feasibility tolerance in the kth iteration; this criterion evolves iteration-by-iteration, unlike the optimality parameter ω_k. Although η_k eventually approaches zero as $k \to \infty$, we set the final convergence criterion as follows:

$$||c(\vec{x}_k)|| < \eta^*, \tag{3.8}$$

where η^* denotes the terminal feasibility tolerance set to be

$$\eta^* = 0.001 \times M. \tag{3.9}$$

Here M is the appropriate typical mass scale associated with the target mass function, its value should be chosen depending on the specific physics process at hand. In the current version of OPTIMASS, the user is expected to provide the relevant fixed value of the scale

[12] The specific method to set this quantity is

MNMIGRAD::OPERATOR()(UNSIGNED INT MAXFCN, DOUBLE TOLERANCE),

where MAXFCN denotes maximum number of function calls after which the MINUIT minimization routine will be stopped even if it has not yet obtained satisfactory convergence to an acceptable minimum. The default value is MAXFCN = 5000.

[13] In eq. (3.6), we assume that all constraints c_a have the same mass dimension $dim(c_a) = dim(c) = 1$, otherwise each term in the r.h.s. should be raised to the appropriate power of $1/dim(c_a)$.

JHEP01(2016)026

M.[14] The rules for updating η_k will be described in section 3.1.3. The starting feasibility tolerance is initialized by

$$\eta_0 = \bar{\eta}\,(\min\,[\mu_0, \bar{\gamma}])^{\beta_\eta^0}\,. \tag{3.10}$$

Here the pre-factor $\bar{\eta}$ is given by

$$\bar{\eta} = \alpha \cdot \eta^*, \tag{3.11}$$

with some coefficient α chosen in the range

$$10 \lesssim \alpha \lesssim 1000, \tag{3.12}$$

μ_0 is the starting penalty parameter given in (3.16) below, while $\bar{\gamma} \in (0, 1)$, and our default for it is

$$\bar{\gamma} = 0.2. \tag{3.13}$$

By adjusting the coefficient α within the range (3.12), one can control the relative scale of the initial feasibility tolerance, η_0, to its terminal value, η^*. This in turn determines the relative number of iterations within each of the two regimes (phase-1 and phase-2) described in section 3.1.3 below. If α is too large, the ALM iterations in phase-1 terminate very quickly, and the majority of the ALM iterations are performed in the regime of phase-2, where the Lagrange-multipler driven evolution may not be efficient. On the other hand, if α is chosen to be too small, most of the ALM iterations will be done in the regime of phase-1, which only reduces the penalty parameter μ_k. In that case, OPTIMASS will not be able to take full advantage of the ALM method in avoiding the ill-conditioning as explained in section 2.2. We recommend that users test several different values of α, until the number of iterations in each phase is adequate and the results are stable.

Finally, the "tightening" parameters, $\beta_\eta^k \in (0, 1)$, for the feasibility constraint are set to be

$$\beta_\eta^0 = 0.5, \tag{3.14}$$

$$\beta_\eta^k = 0.3 \quad (k \geq 1). \tag{3.15}$$

Penalty and Lagrange multiplier parameters. The penalty parameter, μ, and Lagrange multiplier parameter vector, $\boldsymbol{\lambda}$, are updated in each iteration (the actual assignment rule will be explained below in section 3.1.3). Their starting values are

$$\mu_0 = 0.1 \tag{3.16}$$

$$\lambda_a^0 = 0 \quad (a = 1, \cdots, m). \tag{3.17}$$

For the reduction of the penalty parameter μ, OPTIMASS introduces another parameter, $\tau_\mu \in (0, 1)$, defined in (3.20) and by default set to

$$\tau_\mu = 0.5. \tag{3.18}$$

[14] One possibility is $M = [f(\vec{x}_0)]^{1/dim(f(\vec{x}))}$, where $dim(f(\vec{x}))$ is the mass dimension of $f(\vec{x})$.

JHEP01(2016)026

If the value of τ_μ is too small, the penalty parameter μ may decrease too quickly, causing strong convexification, which can lead to ill-conditioning. Conversely, if the value of τ_μ is too large, one can experience slow convergence and a premature transition to phase-2 (see section 3.1.3 below).

Initial minimizer and initial step size for MINUIT**.** The initial guess for the minimizer of the objective function, $\vec{x}_0^s$, (referred to as "seeding" in the flowchart of figure 1), is set by the invisible momentum configuration corresponding to $\hat{s}_{\rm min}$ [31], i.e., the invisible momenta are such that the total invariant mass in the event is minimized. In addition, the initial step size, $\Delta\vec{x}_0^s$, from from the $\vec{x}_0^s$ toward the final minimizer, $\vec{x}^*$, is another input parameter for the MINUIT initialization.[15]

3.1.2 Step two: unconstrained minimization with MINUIT

Once the initial parameters have been chosen, we use the ALM mapping of a constrained minimization problem to an unconstrained minimization problem and then perform the latter minimization with MINUIT (see appendix A). In the process of this minimization, we perform an appropriate adjustment of parameters in each step of the algorithm. The code has two different options for minimization with MINUIT in the $k^{\rm th}$ iteration:

1. MIGRAD:
   Using as a starting value the minimizer, $\vec{x}_{k-1}$, obtained in the previous iteration, MIGRAD searches for a minimizer, $\vec{x}_{k,1}$.

2. SIMPLEX *and* MIGRAD:
   Again using as a starting value the minimizer, $\vec{x}_{k-1}$, obtained in the previous iteration, SIMPLEX first finds a minimizer, $\vec{x}_{k,S}$, and then MIGRAD uses $\vec{x}_{k,S}$ as a starting value to find the final minimizer, $\vec{x}_{k,2}$.

Among the two possible answers, $\vec{x}_{k,1}$ and $\vec{x}_{k,2}$, we choose as our minimizer, $\vec{x}_k$, the one which gives a lower value for the objective function. One could repeat either algorithm 1 or algorithm 2 (or other minimization procedures) with various starting points, $\vec{x}_0$, to find a global minimum more accurately. However, we find that the combination of algorithms 1 and 2 described above is adequate to obtain accurate on-shell constrained M_2 values, while keeping the computational effort to a minimum. Interestingly, we find that algorithms 1 and 2 are complementary to each other. For example, in the M_2 calculations of section 4, we found that for the maximally constrained case of M_{2CC}, the final solution, $\vec{x}_k$, was given by the answer $\vec{x}_{k,1}$ from algorithm 1 ($\vec{x}_{k,2}$ from algorithm 2) in 83% (17%) of the events. This trend was reversed in the calculation of the minimally constrained case of M_{2XX}, where algorithm 1 (algorithm 2) supplied the final solution $\vec{x}_k$ in 19% (81%) of the events. This behavior can be understood as follows. MIGRAD relies on gradient information, thus

[15] The initial input values for $\vec{x}_0^s$ and $\Delta\vec{x}_0^s$ can be set via the method

MNUSERPARAMETERS::ADD(CONST CHAR* PAR-NAME, DOUBLE INIT-POINT, DOUBLE INIT-STEP-SIZE).

JHEP01(2016)026

it can be fast and accurate if the objective function is smooth and continuous. On the other hand, SIMPLEX does not require gradient information, and can handle more complicated functions (including "folds" and "creases"), although the ultimate accuracy is not as high. In the case of M_{2CC}, the objective function is convexified by the penalty terms, which makes the relevant geometry near the local minimum smooth and well-defined, thus we expect algorithm 1 by MIGRAD to outperform algorithm 2. However, it is known that the M_{2XX} objective function, "the maximum of the two invariant masses", develops a crease, on which the solution is found [77], and therefore one might expect algorithm 2 by SIMPLEX to work better for this case. The performance of OPTIMASS with regard to the M_2 variables will be discussed in more detail in section 4.

3.1.3 Step three: ALM parameter adjustment

Once the minimization routine has obtained a value of the minimizer, $\vec{x}_k$, we then evaluate the constraints at this minimizer, i.e., $c_a(\vec{x}_k)$. Depending on the value of the feasibility (3.6), we define three phases: "Phase 1", "Phase 2", and "Phase 3". While "Phase 3" is nothing but terminating the entire minimization procedure, the other two phases basically tighten the feasibility tolerance. Our tightening scheme is inspired by the LANCELOT package [90–92]. As mentioned earlier, the tolerance, ω_k, for the optimality condition (3.3) is *not* evolved: $\omega_k = \omega^*$.

Phase 1. The feasibility condition is far from satisfied:

$$||c(\vec{x}_k)|| > \max\left[\eta_k, \eta^*\right]. \tag{3.19}$$

In this case we put more weight on the penalty term by reducing μ_{k+1} for the next iteration. At the same time, the Lagrange multiplier vector, $\boldsymbol{\lambda}_k$, remains unchanged. In the next iteration, the starting value of the minimizer, $\vec{x}^s_{k+1}$, is set to be the minimizer obtained in the previous iteration, $\vec{x}_k$. Finally, the feasibility tolerance, η_{k+1}, is also evolved. The detailed updating rules for Phase 1 are thus

$$\mu_{k+1} = \tau_\mu \cdot \mu_k, \tag{3.20}$$

$$\lambda_a^{k+1} = \lambda_a^k, \tag{3.21}$$

$$\vec{x}^s_{k+1} = \vec{x}_k, \tag{3.22}$$

$$\eta_{k+1} = \bar{\eta}(\bar{\gamma} \cdot \mu_{k+1})^{\beta_\eta^{k+1}}. \tag{3.23}$$

Phase 2. The feasibility condition is converging, but insufficient to terminate the algorithm:

$$\eta^* < ||c(\vec{x}_k)|| < \eta_k. \tag{3.24}$$

In this case, we do not reduce the penalty parameter, μ_k, and instead adjust the values of the Lagrange multiplier vector by the rule in eq. (2.23). The starting value of the minimizer, $\vec{x}^s_{k+1}$, for the next iteration is set as in Phase 1. The feasibility tolerance is also modified.

JHEP01(2016)026

The detailed updating rules are given by

$$\mu_{k+1} = \mu_k, \tag{3.25}$$

$$\lambda_a^{k+1} = \lambda_a^k - \frac{c_a(\vec{x}_k)}{\mu_k}, \tag{3.26}$$

$$\vec{x}_{k+1}^s = \vec{x}_k, \tag{3.27}$$

$$\eta_{k+1} = \eta_k \cdot \mu_{k+1}^{\beta_\eta^{k+1}}. \tag{3.28}$$

Phase 3. In this phase, we have achieved sufficient feasibility:

$$||c(\vec{x}_k)|| < \eta^*. \tag{3.29}$$

We therefore break the sub-minimization loop and return $\vec{x}_k$ as the final value, $\vec{x}^*$, of the minimizer.

3.2 Validation

We now demonstrate the performance of the algorithm described in the previous subsection with two simple examples, for which one can also obtain analytic solutions for the minimizer, $\vec{x}^*$, and the Lagrange multiplier vector, $\boldsymbol{\lambda}^*$. The first example, considered in section 3.2.1, yields a well-defined solution at a unique global minimum. In the second example, treated in section 3.2.2, we find that the solution for the Lagrange multiplier vector is *not* well-defined because the relevant objective function is "folded" and is not differentiable at $\vec{x}^*$. The examples illustrate the evolution of the ALM parameters and demonstrate how the solution found in the kth iteration converges to the true value in terms of feasibility and optimality.

3.2.1 Example one

Our first example involves minimizing the objective function

$$f(x, y) = x + y, \tag{3.30}$$

over the usual plane, $\vec{x} \equiv (x, y)$, subject to the constraint

$$x^2 + y^2 - 1 = 0. \tag{3.31}$$

This constraint implies that our solution must lie on a unit circle centered at the origin. Clearly, the function (3.30) is minimized along the circle at the point

$$(x^*, y^*) = \left(-\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2} \right), \tag{3.32}$$

which is the global minimizer in this example.

The objective function, (3.30), is plotted in the left panel of figure 2. The locus of feasible points (i.e., the unit circle about the origin,) is shown in black. With the Lagrange multiplier method, one adds a Lagrange multiplier term to the objective function as in (2.4)

$$\mathcal{L}(x, y, \lambda) = x + y - \lambda\,(x^2 + y^2 - 1). \tag{3.33}$$

JHEP01(2016)026

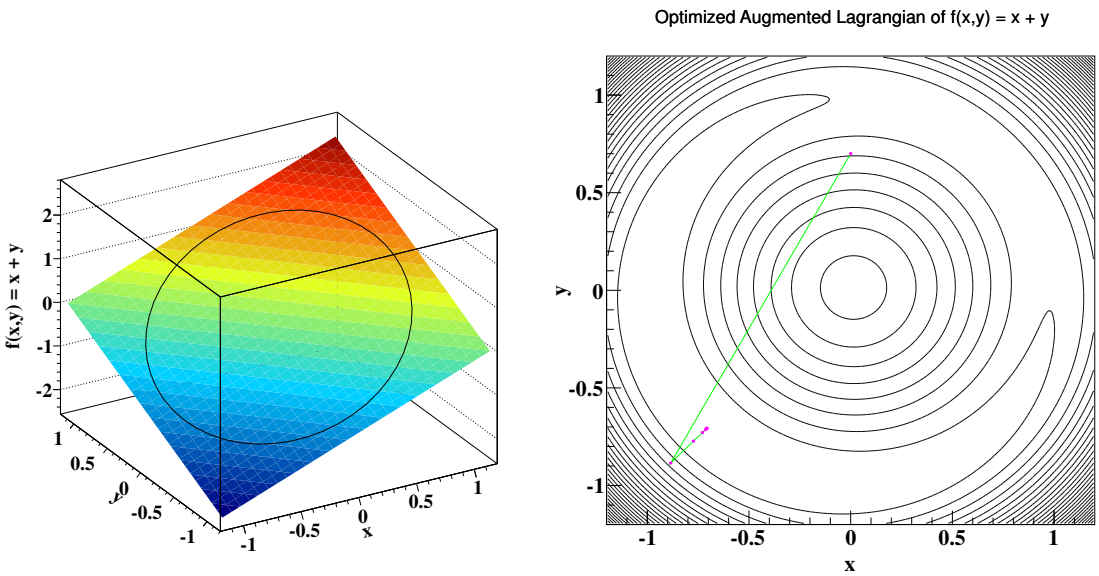

Figure 2. Test of the ALM for the objective function $f(x, y) = x + y$ subject to $x^2 + y^2 = 1$. (a) Plot of the objective function (color coded) and the constraint curve (in black). (b) Contour plot of the augmented Lagragian (3.38) for the last (fifth) iteration. The magenta points denote the minimizers, $\vec{x}_k$, found at each iteration.

There are two *stationary points* given by

$$(x^*,\, y^*\,,\, \lambda^*) = \pm\left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\,,\, \frac{\sqrt{2}}{2}\right). \tag{3.34}$$

The Hessian corresponding to the function (3.33) is

$$H_{\mathcal{L}} = \begin{pmatrix} -2\lambda & 0 \\ 0 & -2\lambda \end{pmatrix}, \tag{3.35}$$

and the condition for a minimum (i.e., that $H_{\mathcal{L}}$ should be positive definite) requires us to choose $\lambda < 0$, which selects the correct minimizer among the two stationary points (3.34):

$$(x^*,\, y^*\,,\, \lambda^*) = \left(-\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}\,,\, -\frac{\sqrt{2}}{2}\right), \tag{3.36}$$

confirming the earlier result (3.32).

We now wish to verify that the OPTIMASS algorithm reproduces this solution. After running the code, we obtain

$$(x^*, y^*; \lambda^*) = (-0.707106, -0.707106; -0.707180), \tag{3.37}$$

which is consistent with (3.36) ($\sqrt{2}/2 = 0.707107\ldots$). We note that this convergence only required five steps, suggesting that the minimum was found relatively easily. The right

JHEP01(2016)026

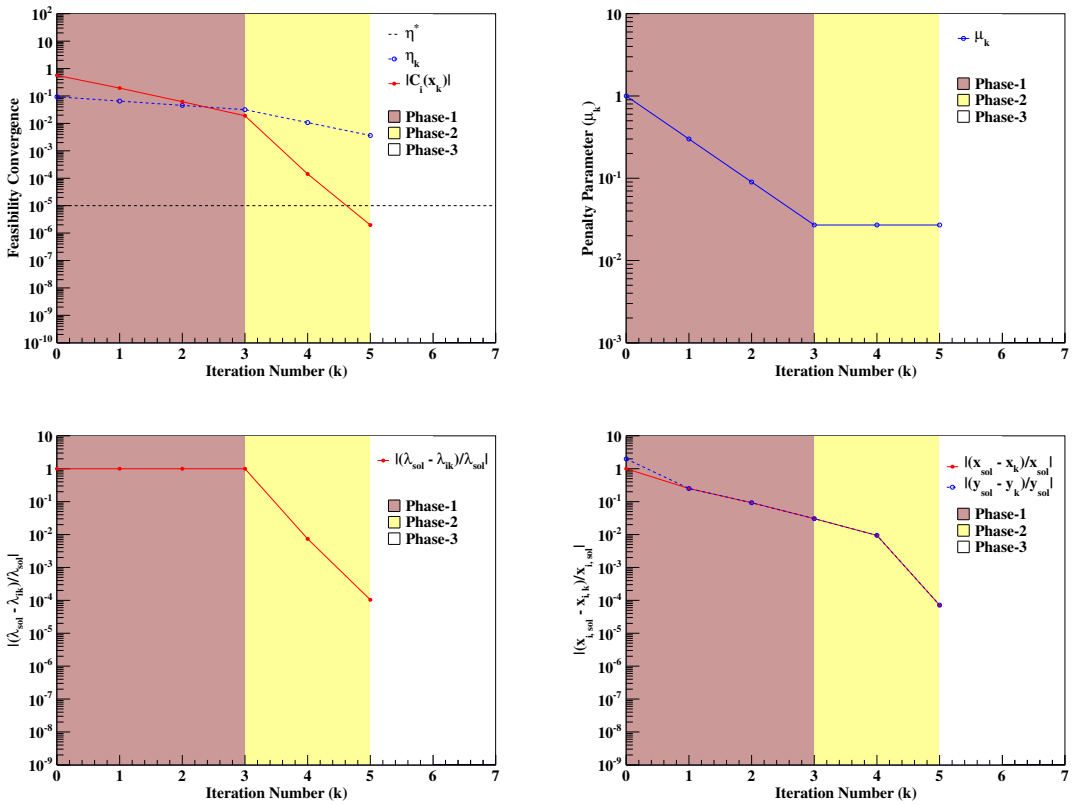

Figure 3. Tracing the convergence state of the solution and the evolution of relevant parameters of the ALM algorithm for the minimization problem depicted in figure 2.

panel of figure 2 shows a contour plot of the augmented Lagrangian,

$$\tilde{\mathcal{L}}(x, y; \mu_5, \lambda_5) = (x + y) - \lambda_5 \left(x^2 + y^2 - 1\right) + \frac{1}{2\mu_5}(x^2 + y^2 - 1)^2 \quad \text{with} \qquad (3.38)$$
$$(\mu_5, \lambda_5) = (0.027, -0.707180),$$

for the final minimization step ($k = 5$) at which the correct solution, (3.37), was obtained. The convexification due to the penalty term

$$\frac{1}{2\mu_5}(x^2 + y^2 - 1)^2$$

ensures that the solution of the constrained optimization problem is (at least) a local minimum. The magenta dots (connected by green lines) in the figure denote the solution, $\vec{x}_k$, of the kth minimization, starting with a randomly chosen initial point, $\vec{x}_0^s = (0, 0.7)$.

In the four panels of figure 3, we trace the evolution of several parameters of the algorithm as well as the properties of the approximate solutions, $\vec{x}_k$. The upper left panel shows the evolution of the intermediate feasibility tolerance, η^k, (blue dashed line) and the real feasibility, $||c_i(x_k)||$, (red solid line) calculated at the end of each iteration, as well as the scale set by the ultimate feasibility tolerance, η^*, given by the horizontal black dashed

JHEP01(2016)026

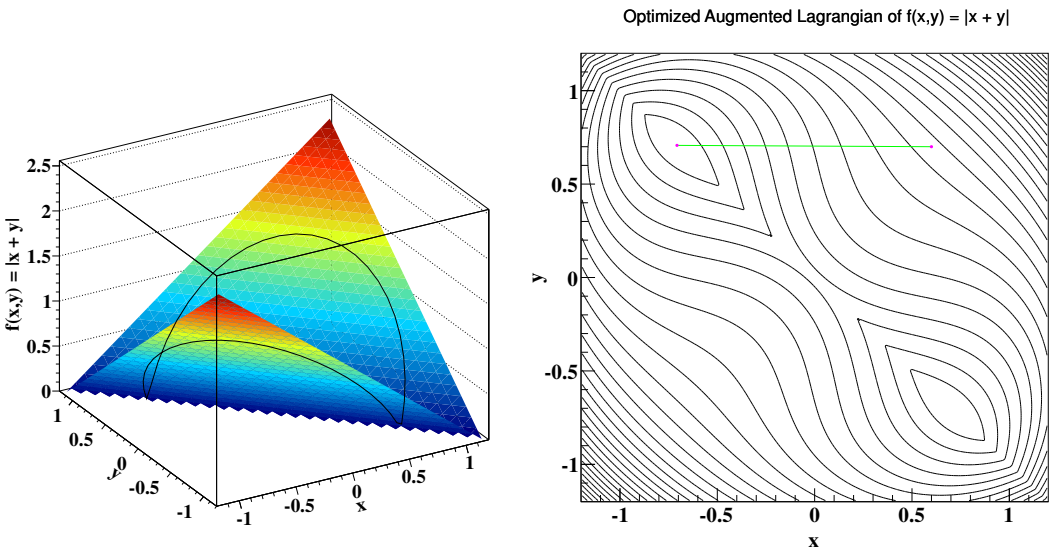

Figure 4. The same as figure 4, but using (3.39) as an objective function instead.

line. We see that in the first two iterations $||c_i(x_k)|| > \eta^k$, and so we are in Phase 1(brown shade). The onset of Phase 2 is marked by the crossing of the red solid and blue dashed lines; in iterations 3 and 4 we are in Phase 2. The terminal Phase 3 is entered when the red solid line dips below the horizontal black dashed line marking the value of η^*, and this finally occurs during the 5th iteration.

The upper right and lower left panels in figure 3 respectively show the evolution of (the initial values of) the penalty parameter, μ_k, and the Lagrange multiplier, λ_k, in each iteration. We see that during Phase 1, μ_k was updated while λ_k was fixed, while in Phase 2, λ_k was updated while μ_k was held fixed. Throughout this process, the solution, $\vec{x}_k$, from each step gradually converges to the analytic solution, (3.32), as shown in the lower right panel in figure 3. Note that from the first iteration on, the solutions are on the diagonal line $x = y$ (see also the right panel in figure 2) — except for their starting values, the red and blue lines in the lower right panel of figure 3 essentially overlap.

3.2.2 Example two

Our second example is very similar to the one considered in the previous subsection, except now we change the objective function to

$$f(x, y) = |x + y|\,, \tag{3.39}$$

and we keep the same constraint as before:

$$x^2 + y^2 - 1 = 0. \tag{3.40}$$

The left panel in figure 4 plots the objective function, $f(x, y)$, as well as the feasible set (the unit circle centered on the origin). Note the "fold" along the line $y = -x$. On this line, the objective function is not a smooth function, hence one of the basic assumptions

JHEP01(2016)026

generally employed in the theory of constrained optimization (see section 2) is not satisfied. Nevertheless, in such cases we typically find that the OPTIMASS algorithm still converges efficiently to the correct solution.

It is clear from figure 4 that the current problem has a two-fold ambiguity, there are two equivalent solutions for the global minimizer:

$$(x^*,\, y^*) = \left(\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}\right) \qquad \text{or} \qquad (x^*,\, y^*) = \left(-\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right). \tag{3.41}$$

Starting from the initial point $(x_0^s, y_0^s) = (0.6, 0.7)$, OPTIMASS converges to

$$(x^*, y^*) = (-0.707132, 0.707132) \tag{3.42}$$

in a single iteration (i.e., without taking $\mu \to 0$). The right panel in figure 2 shows a contour plot of the augmented Lagrangian,

$$\tilde{\mathcal{L}}(x, y; \mu_0, \lambda_0) = |x + y| - \lambda_0 \left(x^2 + y^2 - 1\right) + \frac{1}{2\mu_0}\left(x^2 + y^2 - 1\right)^2, \tag{3.43}$$

for the only step that the algorithm needed, the initial step. The "folded valley" feature of the objective function suggests that one does not need much additional convexification from the penalty term (since we are already in a valley), and the algorithm converges very quickly.

In both of these two toy examples, as well as in numerous physics motivated studies described in the next section, we verified that the numerical solutions obtained with OPTIMASS are stable with respect to small variations of the default initial values of the parameters, and in particular $\vec{x}_0^s$.

4 Calculating M_2 variables with OPTIMASS

In this section, we describe the main intended use of OPTIMASS, the calculation of kinematic variables suitable for analyzing missing energy events at hadron colliders. The calculation involves a minimization of a mass function over a number of invisible particle momenta, subject to certain constraints (e.g., on-shell constraints, or the missing transverse momentum constraint (1.1)). In particular, we will show how the code can be used to calculate the recently proposed M_2 variables with non-linear constraints [26, 76, 77].

We first provide a brief review of the M_2 variables; then define the relevant objective function and identify the sorts of constraints that may be imposed. We then demonstrate the performance of OPTIMASS in the calculation of M_2 in the physically important case of top quark pair production, when both tops decay leptonically.

4.1 Introduction to M_2

The M_2 variable [26, 76, 77] is a $(3+1)$-dimensional analogue of the well-known M_{T2} variable [57]. Both are typically applied to final states that may result from (a) the pair[16]

[16] Hence the subscript "2".

JHEP01(2016)026

production of "mother" particles that (b) subsequently decay to both visible and invisible particles. The best motivated scenarios typically have too many invisible particles in the final state, so that we cannot, in general, reconstruct the masses or momenta of all of the intermediate particles in the event with certainty. Both M_2 and M_{T2} are thus constructed to provide an ansatz for the invisible particle momenta. This ansatz involves the minimization of a suitably defined kinematic mass function of the visible and invisible momenta in the event. M_{T2}, by construction, is restricted to the transverse plane, and does not involve the longitudinal momenta of the invisible particles. On the other hand, M_2 is not limited to the transverse plane, and thus provides an ansatz for the longitudinal invisible momenta as well. This ansatz can be usefully applied to particle mass reconstruction and event topology disambiguation [77, 78]: once we obtain values for the three-momenta of the invisible particles, we can work backwards to reconstruct the masses of the heavier particles produced in the intermediate steps of the relevant decay in terms of the hypothesized masses of the invisible particles in the final state. While the mass of such intermediate resonances is *a priori* unknown, in symmetric event topologies the two decay sides are identical (by definition), and we can impose the condition that the mass of an intermediate resonance of interest is the same in both decay chains. Adding such on-shell constraints further restricts the allowed solution space for the individual invisible momenta, leading in general to a different outcome from the procedure of minimization. Thus the imposition of different on-shell constraints leads to new, physically-motivated kinematic variables.

Let us consider for concreteness a process in which a pair of heavy particles, A, undergo identical two-step, two-body, cascade decays, i.e., each A decays into two (massless) visible particles, a and b, plus a (massive) invisible particle, C, via an on-shell intermediate particle, B, (see figure 5)

$$A \to a\, B \to a\, b\, C. \tag{4.1}$$

For simplicity, we assume that all visible particles are fully distinguishable and that particle a is emitted before particle b, i.e., we do not address the combinatorial issues since they are not relevant for the current discussion of computing the M_2 variables.

Given the event topology of figure 5, one can consider three different subsystem topologies, depending on which of the particles along the red dashed lines are treated as mothers and which are treated as daughters [73, 77]. For example, considering the event as a whole corresponds to subsystem (ab), in which A_i are the mothers, C_i are the daughters, while B_i are intermediate resonances, dubbed "relatives" in [77]. We are interested in placing the maximum possible lower bound on the mass of A, as a function of the hypothesized mass, $\tilde{m}$, of C. The prescription for doing so is well-known (see, e.g., [26]): we minimize of the heavier of the two parent masses, M_{A_1} and M_{A_2}, subject to relevant kinematic constraints. In the simplest case, we only apply the missing transverse momentum constraint, (1.1), and obtain

$$\begin{aligned} M_2(\tilde{m}) &\equiv \min_{\vec{q}_1,\vec{q}_2} \left\{ \max\left[M_{A_1}(\vec{q}_1,\tilde{m}),\ M_{A_2}(\vec{q}_2,\tilde{m}) \right] \right\}, \\ \vec{q}_{1T} + \vec{q}_{2T} &= \vec{\not{p}}_T \end{aligned} \tag{4.2}$$

JHEP01(2016)026

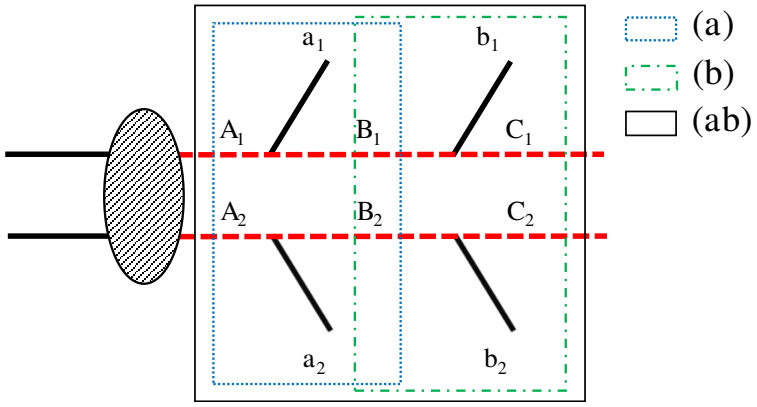

Figure 5. The event topology for the decay process in eq. (4.1), together with the three possible subsystems. The blue dotted, the green dot-dashed, and the black solid lines indicate the subsystems (a), (b), and (ab), respectively.

where $\vec{q}_i$ denotes the three-momentum of the invisible particle, C_i. Note that the missing transverse momentum condition is linear and is easily solved, so that the minimization in (4.2) is unconstrained and can be performed over an unconstrained four-dimensional momentum space, e.g., $\{\vec{q}_{1T}, q_{1z}, q_{2z}\}$.

The situation becomes much more interesting (and challenging) when we consider additional nonlinear constraints. Given the process of figure 5, it is natural to consider additionally constrained versions of eq. (4.2). Having already made the assumption that the two decay chains are identical,[17] we can additionally impose that the particles A_1 and A_2 have the same mass,

$$M_{A_1} = M_{A_2}, \tag{4.3}$$

that the particles B_1 and B_2 have the same mass,

$$M_{B_1} = M_{B_2}, \tag{4.4}$$

or both (4.3) and (4.4). Together with the case where neither (4.3) or (4.4) is required to hold, a total of four variants are therefore possible. Following the same notation as ref. [77], we introduce two more subscripts on the M_2 variable to indicate whether the constraints in eqs. (4.3) and (4.4) were applied during the minimization or not. The first subscript will refer to the parent constraint in eq. (4.3), while the second subscript will refer to the relative constraint in eq. (4.4). If a constraint is imposed, the corresponding index is "C", otherwise it is "X". Therefore, eq. (4.2) can be expressed as M_{2XX} because no extra constraints are imposed:

$$\begin{aligned} M_{2XX}(\tilde{m}) &\equiv \min_{\vec{q}_1,\vec{q}_2} \left\{ \max\left[M_{A_1}(\vec{q}_1, \tilde{m}),\ M_{A_2}(\vec{q}_2, \tilde{m}) \right] \right\}, \\ \vec{q}_{1T} + \vec{q}_{2T} &= \vec{\not{p}}_T. \end{aligned} \tag{4.5}$$

[17] See refs. [60, 61, 78] for relaxing this assumption.

JHEP01(2016)026

The other three variables are formally defined as follows:

$$
\begin{aligned}
M_{2CX}(\tilde{m}) &\equiv \min_{\vec{q}_1,\vec{q}_2} \{\max\left[M_{A_1}(\vec{q}_1,\tilde{m}),\ M_{A_2}(\vec{q}_2,\tilde{m})\right]\}\,, \\
\vec{q}_{1T}+\vec{q}_{2T} &= \vec{\not{p}}_T \\
M_{A_1} &= M_{A_2}
\end{aligned}
\tag{4.6}
$$

$$
\begin{aligned}
M_{2XC}(\tilde{m}) &\equiv \min_{\vec{q}_1,\vec{q}_2} \{\max\left[M_{A_1}(\vec{q}_1,\tilde{m}),\ M_{A_2}(\vec{q}_2,\tilde{m})\right]\}\,, \\
\vec{q}_{1T}+\vec{q}_{2T} &= \vec{\not{p}}_T \\
M^2_{B_1} &= M^2_{B_2}
\end{aligned}
\tag{4.7}
$$

$$
\begin{aligned}
M_{2CC}(\tilde{m}) &\equiv \min_{\vec{q}_1,\vec{q}_2} \{\max\left[M_{A_1}(\vec{q}_1,\tilde{m}),\ M_{A_2}(\vec{q}_2,\tilde{m})\right]\}\,. \\
\vec{q}_{1T}+\vec{q}_{2T} &= \vec{\not{p}}_T \\
M_{A_1} &= M_{A_2} \\
M^2_{B_1} &= M^2_{B_2}
\end{aligned}
\tag{4.8}
$$

Eqs. (4.5)–(4.8) define the four possible M_2 variables for the (ab) subsystem. One can similarly define four M_2 variables for each of the (a) and (b) subsystems, we refer the reader to [77] for the exact definitions.

4.2 Calculating M_2 for dilepton top events

From the definitions (4.6)–(4.8) it is clear that the problem of computing the variables M_{2CX}, M_{2XC}, and M_{2CC} falls into the general category of constrained minimization problems (2.1), (2.2) which OPTIMASS is designed to solve. In the remainder of this section, we shall therefore illustrate the functionality of OPTIMASS with the physics example of figure 5. Specifically, we consider the case of pair-produced top quarks that decay fully leptonically:

$$
pp \to t\bar{t}, \qquad (t \to bW^+ \to bl^+\nu_l), \qquad (\bar{t} \to \bar{b}W^- \to \bar{b}l^-\bar{\nu}_l). \tag{4.9}
$$

Thus, in figure 5, particle A is associated with the top quark, particle B with the W-boson, and particle C with the neutrino. For simplicity, since our major interest is not in the shape of the distributions but in the precision of the minimization procedure, we consider events where the top quarks are produced at threshold and decay according to phase space. We neglect initial and final state radiation, and also do not take into account experimental efficiencies, cuts, combinatorics, and detector resolution. All those effects are important in a real physics analysis, but are irrelevant to the question of evaluating the performance of the OPTIMASS minimization algorithm, which is our goal here.

We use OPTIMASS to compute the values for the M_{2CX}, M_{2XC}, and M_{2CC} variables for all three subsystems. In order to judge the precision of this numerical calculation, ideally we need to identify an alternative method for computing these answers, which would give us reference benchmarks. Fortunately, for the case of the M_{2CX} variable, such a benchmark is provided by the M_{T2} variable itself — we can use the result, proven in ref. [77], that

JHEP01(2016)026

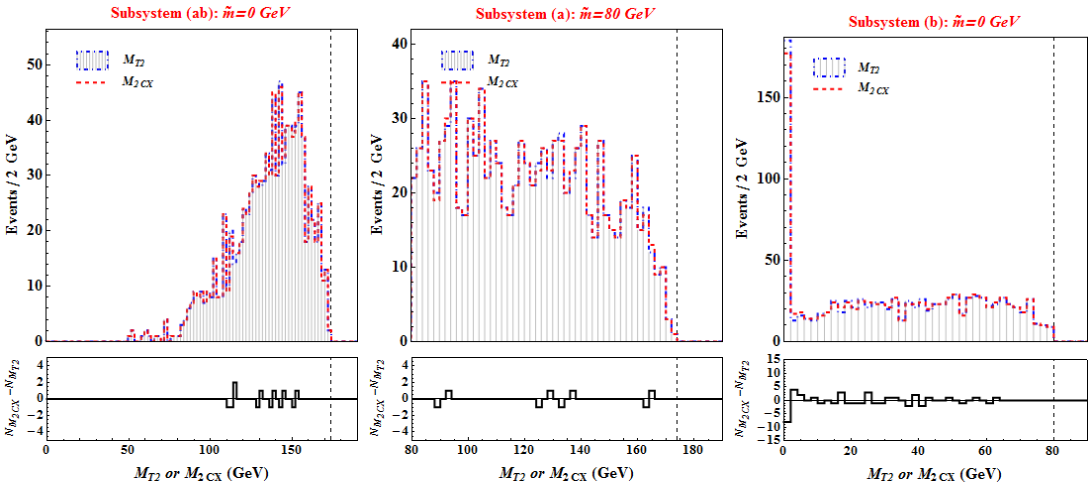

Figure 6. Comparison between M_{T2} (blue shaded histograms) and M_{2CX} (red histograms) for each subsystem. For the (ab) and (a) subsystems, the relevant M_{T2} values are evaluated by the well-known analytic formula, whereas those for the (b) subsystem are computed numerically with the package from ref. [79]. The vertical black dashed lines indicate the expected endpoint of the M_{T2} distribution for the given test mass.

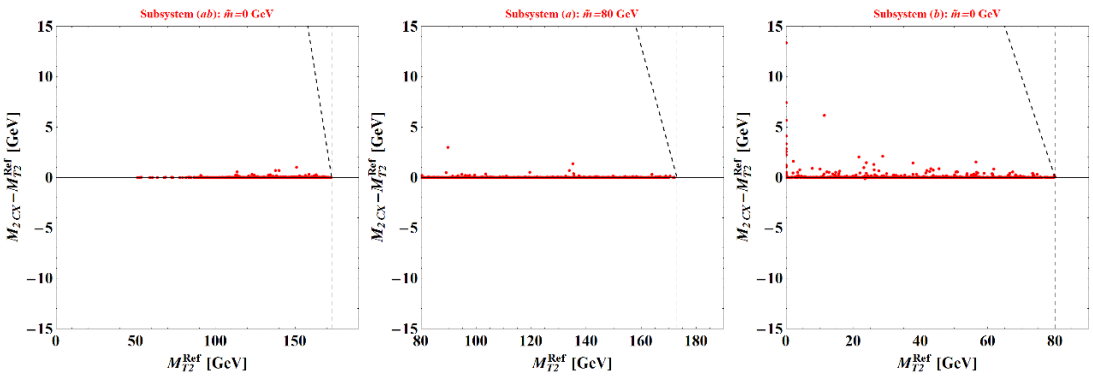

Figure 7. Scatter plots of the difference $M_{2CX} - M_{T2}^{\rm Ref}$ versus the reference value, $M_{T2}^{\rm Ref}$.

M_{2CX} and M_{T2} are identical event by event. This enables us to directly compare the values of M_{T2} and M_{2CX} for each of the three possible subsystems. The Cambridge variable M_{T2} can already be reliably computed with one of several publicly available codes; here we use the package from ref. [79]. Furthermore, for the (ab) and (a) subsystems, analytical formulae for M_{T2} are also available [69, 72], facilitating the comparison.[18]

Figures 6 and 7 show the results from the comparison between the value of M_{2CX} obtained from OPTIMASS and the corresponding reference value, $M_{T2}^{\rm Ref}$, for all three subsystems. Since the exercise is performed with the $t\bar{t}$ decay sample, we take the trial mass to be the true mass of the daughter particle: $\tilde{m} = 0\,\text{GeV}$ for the (ab) and the (b) subsystems

[18] In the case of the (b) subsystem, the b-quarks simulate initial state radiation, in which case no analytical formula for M_{T2} is known, so we need to rely on the computer code from ref. [79].

JHEP01(2016)026

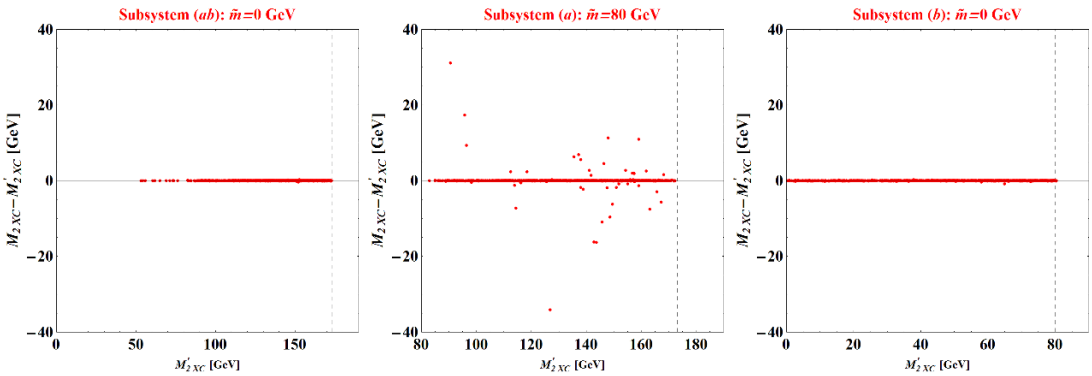

Figure 8. The comparison between the values of M_{2XC} obtained by two different internal codes.

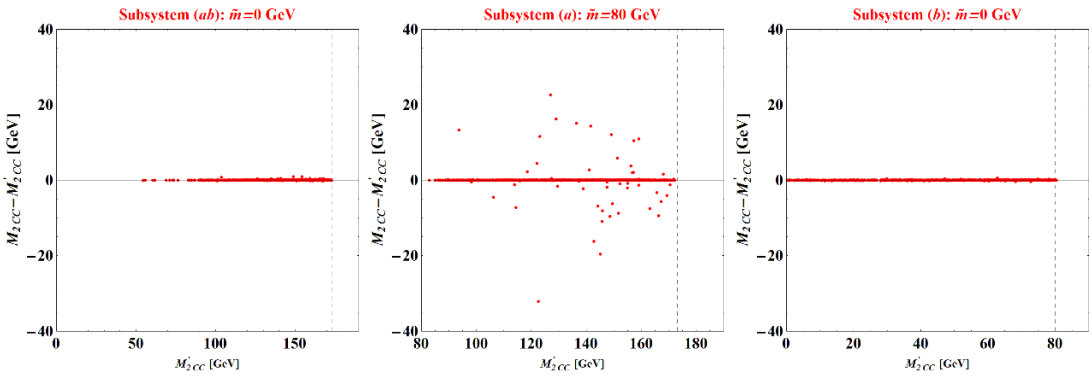

Figure 9. The same as figure 8, but for M_{2CC}.

and $\tilde{m} = 80\,\text{GeV}$ for the (a) subsystem. We choose the M parameter in eq. (3.9) to be 200 GeV for subsystems (ab) and (a), and 100 GeV for subsystem (b). Figure 6 reveals that the distributions of the on-shell constrained M_2 variable, M_{2CX}, (red dashed histograms) are almost identical to the corresponding M_{T2} distributions (blue dot-dashed shaded histograms). Only a handful of events show a difference on the order of $1-2\,\text{GeV}$, as seen in the scatter plots of figure 7. The results shown in figures 6 and 7 were obtained with the default values of the OPTIMASS parameters. Of course, the precision can be further improved by tweaking the relevant tolerance parameters, increasing the maximum number of iterations, or improving on the initial guess of $\vec{x}_0^s$. However, this will come at the cost of increased computation time; we believe that the level of precision seen in these figures should be sufficient for most practical analyses.

Figures 8 and 9 provide a similar validation for the case of the M_{2XC} and M_{2CC} variables. In this case, however, we do not have a readily available benchmark for comparison: first, because analytical formulas for those cases do not exist, and second, because there is no publicly available code which is able to handle M_{2XC} and M_{2CC}. This is why we produced two different versions of our code (created independently by different sets of the current authors), and proceeded to compare their results for M_{2XC} and M_{2CC} in figures 8

JHEP01(2016)026

and 9, respectively. The figures show that the two internal codes agree reasonably well, with notable differences only in about 1% of the events. The events with the largest deviations were scrutinized further, revealing that one of the codes typically found a local minimum, due to a different choice of starting values for $\vec{x}_0^s$. When repeating the minimization with a range of possible choices for $\vec{x}_0^s$, and taking the minimum of the obtained set of values of M_2, the two codes were shown to be in exact agreement.

4.3 Demonstration of OPTIMASS for one event

In conclusion, we supplement the toy examples from section 3.2 with one example of a real $t\bar{t}$ event. The 4-momenta (E, p_x, p_y, p_z) of the four visible particles (in GeV) are

$$\begin{aligned} p_{a1} &= (68.003,\ -8.404,\ 16.069,\ -65.541) \\ p_{b1} &= (56.168,\ -29.282,\ -29.683,\ 37.635) \\ p_{a2} &= (68.003,\ 6.881,\ -56.711,\ -36.890) \\ p_{b2} &= (81.160,\ -27.332,\ 68.553,\ 33.769), \end{aligned} \tag{4.10}$$

thus the missing transverse momentum is

$$\vec{\not{p}}_T = (58.137, 1.772). \tag{4.11}$$

The initial parameters for the algorithm are given by

$$(q^s_{1x,0}, q^s_{1y,0}, q^s_{1z,0}, q^s_{2z,0}) = (-40, +40, -40, +40)\,(\text{GeV}), \qquad (\lambda^0_1, \lambda^0_2) = (0,\ 0).$$

Figure 10 is the analogue of figure 3, showing the convergence to a satisfactory solution for M_{2CC} in subsystem (ab) after the $k = 4$ step, giving

$$\begin{aligned} (q^*_{1x}, q^*_{1y}, q^*_{1z}, q^*_{2z}) &= (38.082,\ 5.612,\ 26.598,\ -8.717)\,(\text{GeV}), \\ (\lambda^{1*}_4, \lambda^{2*}_4) &= (0.000001, -122.329803), \\ \mu_4 &= 0.025. \end{aligned} \tag{4.12}$$

Figure 11 is analogous to figures 2 and 4 for this case. The left panel plots the original objective function, while the right panel shows a contour plot of the augmented Lagrangian function as of the final ($k = 4$) iteration. In the right panel, the set of points which satisfy the constraint eq. (4.3) (eq. (4.4)) is shown in blue (red). As before, the magenta points mark the locations of the minimizer, $\vec{x}_k$, found in the kth iteration.

5 Conclusions

With the restart of the LHC, the quest for new physics has resumed. We believe that kinematic variables like M_2 will play an increasingly important role in searching for SUSY and related models; the gain in sensitivity that these variables provide (see, e.g., [78]) aids both in setting limits on and in discovering BSM physics.

Since the calculation of M_2, like many other kinematic variables, involves a constrained minimization that must be performed numerically, it is important to ensure that this

JHEP01(2016)026

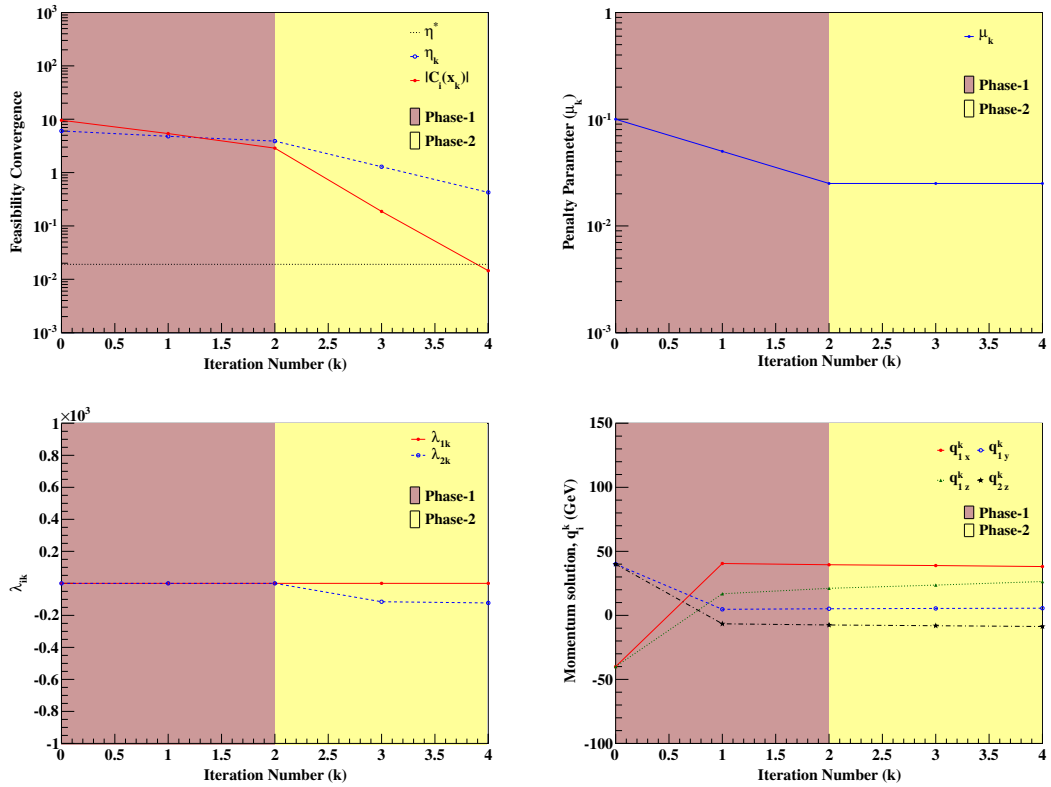

Figure 10. The same as figure 3, but for the M_{2CC} calculation in the (ab) subsystem of the single event considered in section 4.3.

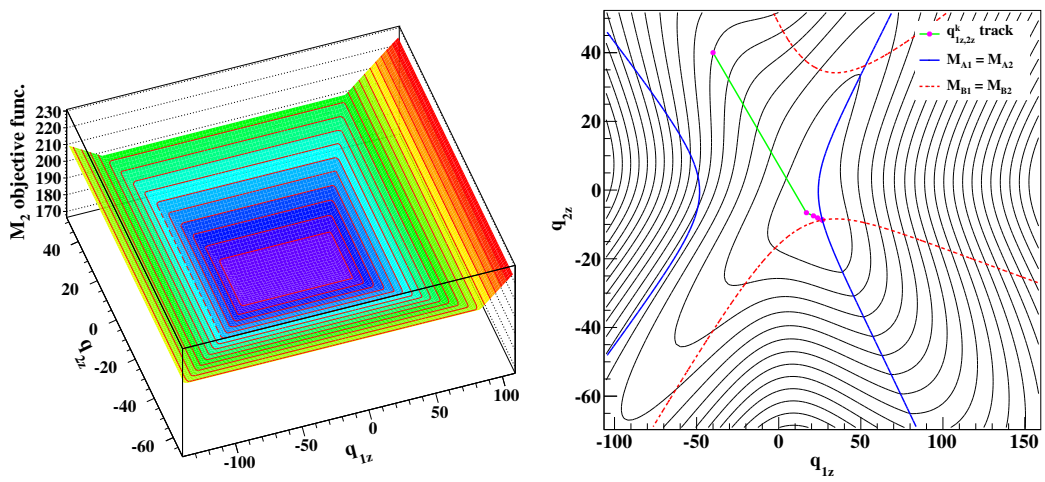

Figure 11. The same as figures 2 and 4, but for the single event considered in section 4.3. Since the objective function has four independent arguments, in order to visualize the evolution of the minimizer, we plot q_{1z} and q_{2z}, having fixed the other two variables, q_{1x} and q_{1y}, to the values which minimize the objective function for the given choice of q_{1z} and q_{2z}.

JHEP01(2016)026

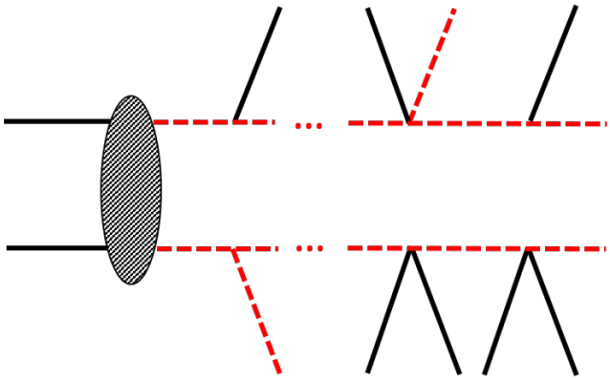

Figure 12. An example of the general event topology which can be handled by OPTIMASS. We allow for two decay chains, involving a sequence of 2-body or 3-body decays. In each decay, the final state particles can be visible, invisible, or both.

calculation is performed in an efficient and reliable way. Thus we have introduced the public package, OPTIMASS, which achieves these important goals. Our algorithm utilizes the ALM and interfaces with the popular unconstrained minimization package, MINUIT.

We have described the relevant issues in what we hope will be sufficient depth to aid the physicist new to the challenges of constrained optimization. The OPTIMASS algorithm has been described in detail and examples of its use have been provided. We compared analytically calculated values of M_{T2} to the M_{2CX} variable obtained using OPTIMASS and found excellent agreement. Other tests of OPTIMASS were performed and the results are encouraging. We stress that while our physics example in section 4 was limited to the dilepton $t\bar{t}$ topology of figure 5, OPTIMASS has been designed to handle arbitrarily general event topologies, as indicated in figure 12:

- *Multiple invisible particles in each decay chain.* In many motivated scenarios, invisible particles may appear not only at the end of the decay chain (as is customary for dark matter particles), but also at intermediate stages. The OPTIMASS code can handle such cases, since the total number of invisible particles is unrestricted. In figure 12, the first decay in the lower chain and the second to last decay in the upper chain provide examples of sources of such intermediate invisible particles.

- *Multi-body decays.* The decay chains can be constructed of two-body decays, three-body decays, etc. Furthermore, a multi-body decay may result in a set of final state particles which can be visible, invisible, or both. As an illustration, in figure 12 we show three two-body decays and three three-body decays. Two of the three-body decays result in visible particles only, while the remaining one produces both a visible and an invisible final state particle.

This more general functionality of OPTIMASS will be explored and demonstrated in a future publication [103].

In conclusion, we look forward to implementing improvements to and extensions of the OPTIMASS framework, and to its use in the search for new physics that lies ahead.

JHEP01(2016)026

A Unconstrained minimization with Minuit

Using the methods of section 2, and in particular the ALM, we can convert a constrained optimization problem into an unconstrained minimization problem; the next step is to actually solve the resulting unconstrained minimization problem. For this task we use Migrad and Simplex, two algorithms which are part of the Minuit library. Thus, after briefly introducing this ubiquitous library, we will discuss these algorithms in sections A.1 and A.2, respectively.

Minuit [84] is a popular function minimization library. It is often used for data analysis, as the minimization of χ^2 functions and likelihoods represents perhaps the main use of minimization in experimental particle physics. Minuit was initially written in Fortran, but has been reimplemented (as Minuit2 [93]) in C++, taking advantage of its object-oriented features; Minuit2 is included in the math library of the omnipresent data analysis package Root [94]. Minuit and Minuit2 (which we will henceforth refer to simply as "Minuit") contain various minimization algorithms, offering the user a choice. Among the several main algorithms (Migrad, Seek, Scan, Simplex), we choose to use Migrad and Simplex which are briefly described in the next two subsections.

A.1 Migrad

Migrad utilizes a variation of the Newton's method called a Variable Metric Method (VMM) [95, 96]. We remind the reader that Newton's method is an iterative method for finding the root of a function $f(x)$ in which

$$x_{n+1} - x_n = -\frac{f(x_n)}{f'(x_n)}. \tag{A.1}$$

Finding a minimum of $f(x)$ rather than a root corresponds to finding a zero of $f'(x)$. In this case the sequence of approximate solutions obtained by Newton's method are described by

$$x_{n+1} - x_n = -\frac{f'(x_n)}{f''(x_n)}. \tag{A.2}$$

The analogous expression in the multidimensional case is

$$\vec{x}_{n+1} - \vec{x}_n = -H^{-1}(\vec{x}_n)\nabla f(\vec{x}_n), \tag{A.3}$$

where $H^{-1}(\vec{x}_n)$ is the inverse of the Hessian matrix. The name "Variable Metric Method" is due to an interesting parallel with General Relativity. Namely, in the limit where the objective function, $f(\vec{x})$, is a quadratic form with minimum at $\vec{x} = \vec{0}$, then, for small $\vec{x}$,

$$f(\vec{x}) - f(\vec{0}) \approx \vec{x}^T \left(\frac{1}{2}H(\vec{x})\right)\vec{x}. \tag{A.4}$$

The expression on the right hand side is a bilinear form with $(1/2)H(\vec{x})$ playing the role of the metric tensor. If one chooses

$$f(\vec{x}) = \sum_{i=1}^{n} x_i^2 \tag{A.5}$$

JHEP01(2016)026

in n dimensions, then $(1/2)H(\vec{x})$ is precisely the n-dimensional Euclidean metric. The quantity on the right hand side of eq. (A.4) is known as the "estimated vertical distance to minimum" (EDM), i.e.,

$$\text{EDM} \equiv \vec{x}^T \left(\frac{1}{2}H(\vec{x})\right)\vec{x}. \tag{A.6}$$

Generally, when using a VMM, the optimality condition will check whether the calculated value for the EDM exceeds a certain tolerance parameter.

Eq. (A.3) describes the essential idea of the VMM. However, the VMM is a "quasi-Newton method" (as opposed to Newton's method itself) because instead of calculating the Hessian (or "metric") exactly, it approximates it iteratively. The main differences among different VMM algorithms lie in the precise form of this iterative approximation procedure. The first VMMs used the so-called DFP updating formula [95–97] (after Davidon, Fletcher, and Powell). Currently, the most common algorithms, including MIGRAD, use the BFGS method [98–101].

A very useful property of VMMs is that subsequent steps are in "conjugate" directions, i.e., in orthogonal directions with respect to the metric provided by the Hessian; additionally, convergence to the minimum is efficient. Thus, the algorithm rarely crosses the folded region of the M_2-objective function where the gradient and Hessian are not defined. This fact motivates the use of the MIGRAD implementation of the VMM for our constrained M_2 calculations.

A.2 Simplex

Unlike VMMs, the downhill simplex (or Nelder-Mead) method [102], does not require the calculation of gradients. Instead, one calculates the values of $f(\vec{x})$ at the $n+1$ vertices of a simplex, a non-degenerate solid in n dimensions with $n+1$ sides and $n+1$ vertices. A new vertex for the simplex is generated in each iteration of the method. If the value of the objective function at the new point is lower than the value at one of the existing vertices, the worst vertex is replaced by the new point. In this way, the volume of the simplex becomes smaller; the algorithm stops when the simplex, now enclosing the minimum, has shrunk to a specified size.

To be more concrete, let us first consider a (large) simplex of $n+1$ points in n dimensions, with vertices

$$p_1, p_2, \ldots, p_n, p_{n+1}. \tag{A.7}$$

These points are ordered so that

$$f_1 \leq f_2 \leq \ldots \leq f_n \leq f_{n+1}, \tag{A.8}$$

where $f_i \equiv f(p_i)$. We define the "center of mass", $\bar{p}$, using all points except p_{n+1} as follows,

$$\bar{p} = \frac{1}{n}\sum_{i=1}^{n} p_i. \tag{A.9}$$

JHEP01(2016)026

In each step, the algorithm tries to replace the worst point, p_{n+1}. First, a new test point, p_r, is obtained by reflection of the worst point about the center of mass,

$$p_r = \bar{p} + \alpha(\bar{p} - p_{n+1}), \tag{A.10}$$

for some typical value of the expansion factor (generally $\alpha = 1$; this is the value we shall use in our use of Simplex). A new point is then determined using the value of $f(p_r)$ as follows:

1. $f_1 \leq f_r \leq f_n$**:** The previous worst point, p_{n+1}, is replaced by p_r, and the points are relabeled in accordance with (A.8).

2. $f_r \leq f_1$**:** The test point, p_r, is the best point, so the current search direction is considered to be effective. We therefore shift the first n points

$$p_1 \to p_2, p_2 \to p_3, \ldots, p_n \to p_{n+1}. \tag{A.11}$$

To determine the new value for p_1, we try one additional point, $p_{s1} = \bar{p} + \beta(p_r - \bar{p})$ (typically $\beta = 2$), and evaluate its functional value, f_{s1}. If $f_{s1} < f_r$, we set $p_1 = p_{s1}$, otherwise $p_1 = p_r$.

3. $f_r > f_n$**:** The simplex may be too big and therefore its size must be reduced. If $f_r > f_{n+1}$, then a new contracted simplex is defined around the best point, p_1, by replacing all points except p_1 by $p_i = p_1 + \delta(p_i - p_1)$ with $0 < \delta < 1$ (typically $\delta = 0.5$). If $f_n < f_r < f_{n+1}$, then p_{n+1} is replaced by p_r. A test of the new inner point, $p_{s2} = \bar{p} - \gamma(\bar{p} - p_{n+1})$ (typically $\gamma = 0.5$), is then performed, and p_{n+1} is replaced by p_{s2} if $f_{s2} < f_{n+1}$.

Since the simplex method is always designed to take as big a step as possible, it is rather *insensitive* to shallow local minima and other small-scale structures in the objective function. Thus, we use the method to identify promising candidates for global minima. Once the location of a possible global minimum has been identified, the Migrad algorithm described above is used to obtain a more precise value of any local minimum in this area, hopefully obtaining an accurate value for the location of the global minimum. The downhill simplex method is implemented in Minuit using the Simplex algorithm.

B Installation and user instructions

The latest version of the Optimass has been developed and designed to achieve the automation of kinematic mass function minimization with constraints for general particle decay system. In particular,

1. It has generality to treat various decay topologies from multiple parent particles where in general the multiplicity can be larger than two.

2. It also has generality to include decay vertices where in general the multiplicity of branch legs can be larger than three.

JHEP01(2016)026

3. It has flexibility to easily define a specific sub-system of intermediate parent particles with effective invisibles, in the full decay system.

4. It also has flexibility to define kinematic constraint functions of user's own interest, in terms of visible and invisible particles' momentum degrees of freedom.

5. All these generality and flexibility can be initiated from user's simple model card file which defines 1) full decay process with user's own particle label scheme, 2) parent node particle in each decay chain, 3) effective invisible nodes in each decay chain, and 4) constraint functions of particle momenta which can be interactively expressed by the user's particle label scheme.

The OPTIMASS is free software written in C++ and PYTHON under the copyleft of the GNU General Public License. The latest version of the OPTIMASS can be downloaded from the following web page:

http://hep-pulgrim.ibs.re.kr/optimass

More detailed OPTIMASS installation guide and the tutorial with examples on how to run the code implementing user's own decay topologies, can be found on the webpage as well.

Acknowledgments

We would like to thank K. Kong for useful discussions. JG, KM, FM, and LP thank their CMS colleagues for useful discussions. WC thanks C. B. Park for useful discussions. DK acknowledges support by LHC-TI postdoctoral fellowship under grant NSF-PHY-0969510. MP is supported by the Korea Ministry of Science, ICT and Future Planning, Gyeongsangbuk-Do and Pohang City for Independent Junior Research Groups at the Asia Pacific Center for Theoretical Physics. MP is also supported by World Premier International Research Center Initiative (WPI Initiative), MEXT, Japan. Work also supported by DOE Grant No. DE-SC0010296 and by IBS under the project code, IBS-R018-D1.

Open Access. This article is distributed under the terms of the Creative Commons Attribution License (CC-BY 4.0), which permits any use, distribution and reproduction in any medium, provided the original author(s) and source are credited.

References

[1] ATLAS collaboration, *Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC*, *Phys. Lett.* **B 716** (2012) 1 [arXiv:1207.7214] [INSPIRE].

[2] CMS collaboration, *Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC*, *Phys. Lett.* **B 716** (2012) 30 [arXiv:1207.7235] [INSPIRE].

[3] S.P. Martin, *A Supersymmetry primer*, hep-ph/9709356 [INSPIRE].

JHEP01(2016)026

[4] M. Drees, R. Godbole and P. Roy, *Theory and phenomenology of sparticles: An account of four-dimensional $N = 1$ supersymmetry in high energy physics*, World Scientific, Hackensack U.S.A. (2004).

[5] H. Baer and X. Tata, *Weak scale supersymmetry: From superfields to scattering events*, Cambridge University Press, Cambridge U.K. (2006).

[6] M. Dine, *Supersymmetry and string theory: Beyond the standard model*, Cambridge University Press, Cambridge U.K. (2007).

[7] T. Appelquist, H.-C. Cheng and B.A. Dobrescu, *Bounds on universal extra dimensions*, *Phys. Rev.* **D 64** (2001) 035002 [hep-ph/0012100] [INSPIRE].

[8] H.-C. Cheng, K.T. Matchev and M. Schmaltz, *Bosonic supersymmetry? Getting fooled at the CERN LHC*, *Phys. Rev.* **D 66** (2002) 056006 [hep-ph/0205314] [INSPIRE].

[9] A.J. Barr and C.G. Lester, *A Review of the Mass Measurement Techniques proposed for the Large Hadron Collider*, *J. Phys.* **G 37** (2010) 123001 [arXiv:1004.2732] [INSPIRE].

[10] L.-T. Wang and I. Yavin, *A Review of Spin Determination at the LHC*, *Int. J. Mod. Phys.* **A 23** (2008) 4647 [arXiv:0802.2726] [INSPIRE].

[11] M. Burns, K. Kong, K.T. Matchev and M. Park, *A General Method for Model-Independent Measurements of Particle Spins, Couplings and Mixing Angles in Cascade Decays with Missing Energy at Hadron Colliders*, *JHEP* **10** (2008) 081 [arXiv:0808.2472] [INSPIRE].

[12] D0 collaboration, V.M. Abazov et al., *A Quasi model independent search for new physics at large transverse momentum*, *Phys. Rev.* **D 64** (2001) 012004 [hep-ex/0011067] [INSPIRE].

[13] CDF collaboration, T. Aaltonen et al., *Measurement of the Top Quark Mass in the Lepton+Jets Channel Using the Lepton Transverse Momentum*, *Phys. Lett.* **B 698** (2011) 371 [arXiv:1101.4926] [INSPIRE].

[14] D.R. Tovey, *Measuring the SUSY mass scale at the LHC*, *Phys. Lett.* **B 498** (2001) 1 [hep-ph/0006276] [INSPIRE].

[15] J. Bramante, J. Kumar and B. Thomas, *Large Jet Multiplicities and New Physics at the LHC*, *Phys. Rev.* **D 86** (2012) 015014 [arXiv:1109.6014] [INSPIRE].

[16] S. El Hedri, A. Hook, M. Jankowiak and J.G. Wacker, *Learning How to Count: A High Multiplicity Search for the LHC*, *JHEP* **08** (2013) 136 [arXiv:1302.1870] [INSPIRE].

[17] A. Hook, E. Izaguirre, M. Lisanti and J.G. Wacker, *High Multiplicity Searches at the LHC Using Jet Masses*, *Phys. Rev.* **D 85** (2012) 055029 [arXiv:1202.0558] [INSPIRE].

[18] D.R. Tovey, *On measuring the masses of pair-produced semi-invisibly decaying particles at hadron colliders*, *JHEP* **04** (2008) 034 [arXiv:0802.2879] [INSPIRE].

[19] G. Polesello and D.R. Tovey, *Supersymmetric particle mass measurement with the boost-corrected contransverse mass*, *JHEP* **03** (2010) 030 [arXiv:0910.0174] [INSPIRE].

[20] K.T. Matchev and M. Park, *A General method for determining the masses of semi-invisibly decaying particles at hadron colliders*, *Phys. Rev. Lett.* **107** (2011) 061801 [arXiv:0910.1584] [INSPIRE].

[21] K. Agashe, R. Franceschini and D. Kim, *Simple "invariance" of two-body decay kinematics*, *Phys. Rev.* **D 88** (2013) 057701 [arXiv:1209.0772] [INSPIRE].

[22] K. Agashe, R. Franceschini, D. Kim and K. Wardlow, *Using Energy Peaks to Count Dark Matter Particles in Decays*, *Phys. Dark Univ.* **2** (2013) 72 [arXiv:1212.5230] [INSPIRE].

JHEP01(2016)026

[23] K. Agashe, R. Franceschini and D. Kim, *Using Energy Peaks to Measure New Particle Masses*, *JHEP* **11** (2014) 059 [arXiv:1309.4776] [INSPIRE].

[24] M.M. Nojiri, D. Toya and T. Kobayashi, *Lepton energy asymmetry and precision SUSY study at hadron colliders*, *Phys. Rev.* **D 62** (2000) 075009 [hep-ph/0001267] [INSPIRE].

[25] H.-C. Cheng and J. Gu, *Measuring Invisible Particle Masses Using a Single Short Decay Chain*, *JHEP* **10** (2011) 094 [arXiv:1109.3471] [INSPIRE].

[26] A.J. Barr et al., *Guide to transverse projections and mass-constraining variables*, *Phys. Rev.* **D 84** (2011) 095031 [arXiv:1105.2977] [INSPIRE].

[27] C.G. Lester, *Mass and Spin Measurement Techniques (for the Large Hadron Collider)*, in *The Dark Secrets of the Terascale: Proceedings, TASI 2011*, 6 June – 11 July 2011, Boulder U.S.A., T.M.P. Tait and K.T. Matchev eds., World Scientific, Singapore (2013).

[28] J. Smith, W.L. van Neerven and J.A.M. Vermaseren, *The Transverse Mass and Width of the W Boson*, *Phys. Rev. Lett.* **50** (1983) 1738 [INSPIRE].

[29] V.D. Barger, A.D. Martin and R.J.N. Phillips, *Perpendicular ν_e Mass From W Decay*, *Z. Phys.* **C 21** (1983) 99 [INSPIRE].

[30] I. Hinchliffe, F.E. Paige, M.D. Shapiro, J. Soderqvist and W. Yao, *Precision SUSY measurements at CERN LHC*, *Phys. Rev.* **D 55** (1997) 5520 [hep-ph/9610544] [INSPIRE].

[31] P. Konar, K. Kong and K.T. Matchev, *$\sqrt{\hat{s}_{min}}$: A Global inclusive variable for determining the mass scale of new physics in events with missing energy at hadron colliders*, *JHEP* **03** (2009) 085 [arXiv:0812.1042] [INSPIRE].

[32] P. Konar, K. Kong, K.T. Matchev and M. Park, *RECO level $\sqrt{s}_{min}$ and subsystem $\sqrt{s}_{min}$: Improved global inclusive variables for measuring the new physics mass scale in $\not{E}_T$ events at hadron colliders*, *JHEP* **06** (2011) 041 [arXiv:1006.0653] [INSPIRE].

[33] T. Robens, *$\sqrt{\hat{s}}_{\rm min}$ resurrected*, *JHEP* **02** (2012) 051 [arXiv:1109.1018] [INSPIRE].

[34] C. Rogan, *Kinematical variables towards new dynamics at the LHC*, arXiv:1006.2727 [INSPIRE].

[35] M.R. Buckley, J.D. Lykken, C. Rogan and M. Spiropulu, *Super-Razor and Searches for Sleptons and Charginos at the LHC*, *Phys. Rev.* **D 89** (2014) 055020 [arXiv:1310.4827] [INSPIRE].

[36] CMS collaboration, *Search for supersymmetry in two-photons+jet events with razor variables in pp collisions at $\sqrt{s} = 8$ TeV*, CMS-PAS-SUS-14-008 (2014).

[37] CMS collaboration, *Search for Supersymmetry Using Razor Variables in Events with b-Tagged Jets in pp Collisions at $\sqrt{s} = 8$ TeV*, *Phys. Rev.* **D 91** (2015) 052018 [arXiv:1502.00300] [INSPIRE].

[38] P.C. Bhat, *Multivariate Analysis Methods in Particle Physics*, *Ann. Rev. Nucl. Part. Sci.* **61** (2011) 281 [INSPIRE].

[39] J. Hubisz, J. Lykken, M. Pierini and M. Spiropulu, *Missing energy look-alikes with 100 pb^{-1} at the LHC*, *Phys. Rev.* **D 78** (2008) 075008 [arXiv:0805.2398] [INSPIRE].

[40] D.S.M. Alves, M.R. Buckley, P.J. Fox, J.D. Lykken and C.-T. Yu, *Stops and $\not{E}_T$: The shape of things to come*, *Phys. Rev.* **D 87** (2013) 035016 [arXiv:1205.5805] [INSPIRE].

[41] K. Kondo, *Dynamical Likelihood Method for Reconstruction of Events With Missing Momentum. 1: Method and Toy Models*, *J. Phys. Soc. Jap.* **57** (1988) 4126 [INSPIRE].

JHEP01(2016)026

[42] K. Kondo, *Dynamical likelihood method for reconstruction of events with missing momentum. 2: Mass spectra for* $2 \to 2$ *processes*, *J. Phys. Soc. Jap.* **60** (1991) 836 [INSPIRE].

[43] J.S. Gainer, J. Lykken, K.T. Matchev, S. Mrenna and M. Park, *The Matrix Element Method: Past, Present and Future*, arXiv:1307.3546 [INSPIRE].

[44] K. Kondo, T. Chikamatsu and S.H. Kim, *Dynamical likelihood method for reconstruction of events with missing momentum. 3: Analysis of a CDF high* p_T *e mu event as* $t\bar{t}$ *production*, *J. Phys. Soc. Jap.* **62** (1993) 1177 [INSPIRE].

[45] R.H. Dalitz and G.R. Goldstein, *The Decay and polarization properties of the top quark*, *Phys. Rev.* **D 45** (1992) 1531 [INSPIRE].

[46] D0 collaboration, B. Abbott et al., *Measurement of the top quark mass in the dilepton channel*, *Phys. Rev.* **D 60** (1999) 052001 [hep-ex/9808029] [INSPIRE].

[47] J.C. Estrada Vigil, *Maximal use of kinematic information for the extraction of the mass of the top quark in single-lepton t anti-t events at D0*, Ph.D. Thesis, Rochester University, New York U.S.A. (2001), FERMILAB-THESIS-2001-07.

[48] M.F. Canelli, *Helicity of the* W *boson in single-lepton* $t\bar{t}$ *events*, Ph.D. Thesis, Rochester University, New York U.S.A. (2003), UMI-31-14921.

[49] D0 collaboration, V.M. Abazov et al., *A precision measurement of the mass of the top quark*, *Nature* **429** (2004) 638 [hep-ex/0406031] [INSPIRE].

[50] CDF collaboration, T. Aaltonen et al., *Top Quark Mass Measurement in the Lepton + Jets Channel Using a Matrix Element Method and in situ Jet Energy Calibration*, *Phys. Rev. Lett.* **105** (2010) 252001 [arXiv:1010.4582] [INSPIRE].

[51] I. Volobouev, *Matrix Element Method in HEP: Transfer Functions, Efficiencies and Likelihood Normalization*, arXiv:1101.2259 [INSPIRE].

[52] A. Demilly, *Mesure de la masse du quark top dans le canal dileptonique électron-muon avec la méthode des éléments de matrice dans l'expérience ATLAS auprés du LHC*, Ph.D. Thesis, Universitè Paris VI, Paris France (2014), CERN-THESIS-2014-140.

[53] CMS collaboration, *Search for a Standard Model Higgs Boson Produced in Association with a Top-Quark Pair and Decaying to Bottom Quarks Using a Matrix Element Method*, *Eur. Phys. J.* **C 75** (2015) 251 [arXiv:1502.02485] [INSPIRE].

[54] J.S. Gainer, J. Lykken, K.T. Matchev, S. Mrenna and M. Park, *Exploring Theory Space with Monte Carlo Reweighting*, *JHEP* **10** (2014) 78 [arXiv:1404.7129] [INSPIRE].

[55] 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*, *JHEP* **07** (2014) 079 [arXiv:1405.0301] [INSPIRE].

[56] D. Schouten, A. DeAbreu and B. Stelzer, *Accelerated Matrix Element Method with Parallel Computing*, *Comput. Phys. Commun.* **192** (2015) 54 [arXiv:1407.7595] [INSPIRE].

[57] C.G. Lester and D.J. Summers, *Measuring masses of semiinvisibly decaying particles pair produced at hadron colliders*, *Phys. Lett.* **B 463** (1999) 99 [hep-ph/9906349] [INSPIRE].

[58] A. Barr, C. Lester and P. Stephens, *m(T2): The Truth behind the glamour*, *J. Phys.* **G 29** (2003) 2343 [hep-ph/0304226] [INSPIRE].

JHEP01(2016)026

[59] P. Konar, K. Kong, K.T. Matchev and M. Park, *Superpartner Mass Measurement Technique using 1D Orthogonal Decompositions of the Cambridge Transverse Mass Variable M_{T2}*, *Phys. Rev. Lett.* **105** (2010) 051802 [arXiv:0910.3679] [INSPIRE].

[60] A.J. Barr, B. Gripaios and C.G. Lester, *Transverse masses and kinematic constraints: from the boundary to the crease*, *JHEP* **11** (2009) 096 [arXiv:0908.3779] [INSPIRE].

[61] P. Konar, K. Kong, K.T. Matchev and M. Park, *Dark Matter Particle Spectroscopy at the LHC: Generalizing M(T2) to Asymmetric Event Topologies*, *JHEP* **04** (2010) 086 [arXiv:0911.4126] [INSPIRE].

[62] G.G. Ross and M. Serna, *Mass determination of new states at hadron colliders*, *Phys. Lett.* **B 665** (2008) 212 [arXiv:0712.0943] [INSPIRE].

[63] A.J. Barr, G.G. Ross and M. Serna, *The Precision Determination of Invisible-Particle Masses at the LHC*, *Phys. Rev.* **D 78** (2008) 056006 [arXiv:0806.3224] [INSPIRE].

[64] W.S. Cho, J.E. Kim and J.-H. Kim, *Amplification of endpoint structure for new particle mass measurement at the LHC*, *Phys. Rev.* **D 81** (2010) 095010 [arXiv:0912.2354] [INSPIRE].

[65] W.S. Cho, W. Klemm and M.M. Nojiri, *Mass measurement in boosted decay systems at hadron colliders*, *Phys. Rev.* **D 84** (2011) 035018 [arXiv:1008.0391] [INSPIRE].

[66] A.J. Barr, B. Gripaios and C.G. Lester, *Re-weighing the evidence for a Higgs boson in dileptonic W-boson decays*, *Phys. Rev. Lett.* **108** (2012) 041803 [*Erratum ibid.* **108** (2012) 109902] [arXiv:1108.3468] [INSPIRE].

[67] Y. Bai, H.-C. Cheng, J. Gallicchio and J. Gu, *Stop the Top Background of the Stop Search*, *JHEP* **07** (2012) 110 [arXiv:1203.4813] [INSPIRE].

[68] A. Papaefstathiou, K. Sakurai and M. Takeuchi, *Higgs boson to di-tau channel in Chargino-Neutralino searches at the LHC*, *JHEP* **08** (2014) 176 [arXiv:1404.1077] [INSPIRE].

[69] W.S. Cho, K. Choi, Y.G. Kim and C.B. Park, *Gluino Stransverse Mass*, *Phys. Rev. Lett.* **100** (2008) 171801 [arXiv:0709.0288] [INSPIRE].

[70] B. Gripaios, *Transverse observables and mass determination at hadron colliders*, *JHEP* **02** (2008) 053 [arXiv:0709.2740] [INSPIRE].

[71] A.J. Barr, B. Gripaios and C.G. Lester, *Weighing Wimps with Kinks at Colliders: Invisible Particle Mass Measurements from Endpoints*, *JHEP* **02** (2008) 014 [arXiv:0711.4008] [INSPIRE].

[72] W.S. Cho, K. Choi, Y.G. Kim and C.B. Park, *Measuring superparticle masses at hadron collider using the transverse mass kink*, *JHEP* **02** (2008) 035 [arXiv:0711.4526] [INSPIRE].

[73] M. Burns, K. Kong, K.T. Matchev and M. Park, *Using Subsystem M_{T2} for Complete Mass Determinations in Decay Chains with Missing Energy at Hadron Colliders*, *JHEP* **03** (2009) 143 [arXiv:0810.5576] [INSPIRE].

[74] C.G. Lester, *The stransverse mass, M_{T2}, in special cases*, *JHEP* **05** (2011) 076 [arXiv:1103.5682] [INSPIRE].

[75] C.H. Lally and C.G. Lester, *Properties of M_{T2} in the massless limit*, arXiv:1211.1542 [INSPIRE].

[76] R. Mahbubani, K.T. Matchev and M. Park, *Re-interpreting the Oxbridge stransverse mass variable M_{T2} in general cases*, *JHEP* **03** (2013) 134 [arXiv:1212.1720] [INSPIRE].

JHEP01(2016)026

[77] W.S. Cho et al., *On-shell constrained M_2 variables with applications to mass measurements and topology disambiguation*, *JHEP* **08** (2014) 070 [arXiv:1401.1449] [INSPIRE].

[78] W.S. Cho et al., *Improving the sensitivity of stop searches with on-shell constrained invariant mass variables*, *JHEP* **05** (2015) 040 [arXiv:1411.0664] [INSPIRE].

[79] H.-C. Cheng and Z. Han, *Minimal Kinematic Constraints and m(T2)*, *JHEP* **12** (2008) 063 [arXiv:0810.5178] [INSPIRE].

[80] J.W. Walker, *A complete solution classification and unified algorithmic treatment for the one- and two-step asymmetric S-transverse mass $\tilde{M}_{\mathrm{T2}}$ event scale statistic*, *JHEP* **08** (2014) 155 [arXiv:1311.6219] [INSPIRE].

[81] C.G. Lester and B. Nachman, *Bisection-based asymmetric M_{T2} computation: a higher precision calculator than existing symmetric methods*, *JHEP* **03** (2015) 100 [arXiv:1411.4312] [INSPIRE].

[82] M.R. Hestenes, *Multiplier and gradient methods*, *J. Optim. Theory Appl.* **4** (1969) 303.

[83] M.J.D. Powell, *A method for nonlinear constraints in minimization problems*, in *Optimization*, R. Fletcher eds., Academic Press, New York U.S.A. (1969), pg. 283.

[84] F. James and M. Roos, *Minuit: A System for Function Minimization and Analysis of the Parameter Errors and Correlations*, *Comput. Phys. Commun.* **10** (1975) 343 [INSPIRE].

[85] J. Nocedal and S.J. Wright, *Numerical optimization*, *Springer series in operations research and financial engineering*, second edition, Springer, New York U.S.A. (2006).

[86] D.P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York U.S.A. (1982).

[87] W.H. Press, S.A. Teukolsky, W.T. Vetterling and B.P. Flannery, *Numerical Recipes in* FORTRAN *77: The Art of Scientific Computing*, Cambridge University Press, Cambridge U.K. (1992).

[88] W.S. Cho, K. Choi, Y.G. Kim and C.B. Park, *M(T2)-assisted on-shell reconstruction of missing momenta and its application to spin measurement at the LHC*, *Phys. Rev.* **D 79** (2009) 031701 [arXiv:0810.4853] [INSPIRE].

[89] C.B. Park, *Reconstructing the heavy resonance at hadron colliders*, *Phys. Rev.* **D 84** (2011) 096001 [arXiv:1106.6087] [INSPIRE].

[90] A.R. Conn, N.I.M. Gould and Ph.L. Toint, *A Globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds*, *SIAM J. Numer. Anal.* **28** (1991) 545.

[91] A.R. Conn, N.I.M. Gould and Ph.L. Toint, *Lecture Notes in Computation Mathematics. Vol. 17: LANCELOT: A Fortran Package for Large-scale Nonlinear Optimization (Release A)*, Springer Verlag, Heidelberg Germany (1992).

[92] LANCELOT optimization software, http://www.numerical.rl.ac.uk/lancelot/blurb.html.

[93] MINUIT2, http://seal.web.cern.ch/seal/MathLibs/Minuit2/html/index.html.

[94] ROOT-'A Data Analysis Framework', http://root.cern.ch/drupal.

[95] W.C. Davidon, *Variable metric method for minimization*, Technical Report ANL-5990 (revised), Argonne National Laboratory, Argonne U.S.A. (1959).

[96] W.C. Davidon, *Variable metric method for minimization*, *SIAM J. Optim.* **1** (1991) 1.

JHEP01(2016)026

[97] R. Fletcher and M. Powell, *A rapidly converging descent method for minimization*, *Comput. J.* **6** (1963) 163.

[98] C.G. Boyden, *The convergence of a class of double-rank minimization algorithms*, *J. Inst. Math. Appl.* **6** (1970) 76.

[99] R. Fletcher, *A New Approach to Variable Metric Algorithms*, *Comput. J.* **13** (1970) 317.

[100] D. Goldfarb, *A Family of Variable Metric Updates Derived by Variational Means*, *Math. Comput.* **24** (1970) 23.

[101] D.F. Shanno, *Conditioning of quasi-Newton methods for function minimization*, *Math. Comput.* **24** (1970) 647.

[102] J.A. Nelder and R. Mead, *A Simplex Method for Function Minimization*, *Comput. J.* **7** (1965) 308 [INSPIRE].

[103] W.S. Cho et al., *Scrutinizing missing energy events at the LHC with OPTIMASS*, in preparation.

JHEP01(2016)026