OPTIMASS

: A Package for the Minimization of Kinematic Mass Functions with Constraints

Cho, Won Sang (IBS-CTPU)

ref1) arXiv:1508.00589
ref2) http://hep-pulgrim.ibs.re.kr/optimass

Sep 07 2015
Kavli-IPMU-Durham-KIAS Workshop

List of Collaborators

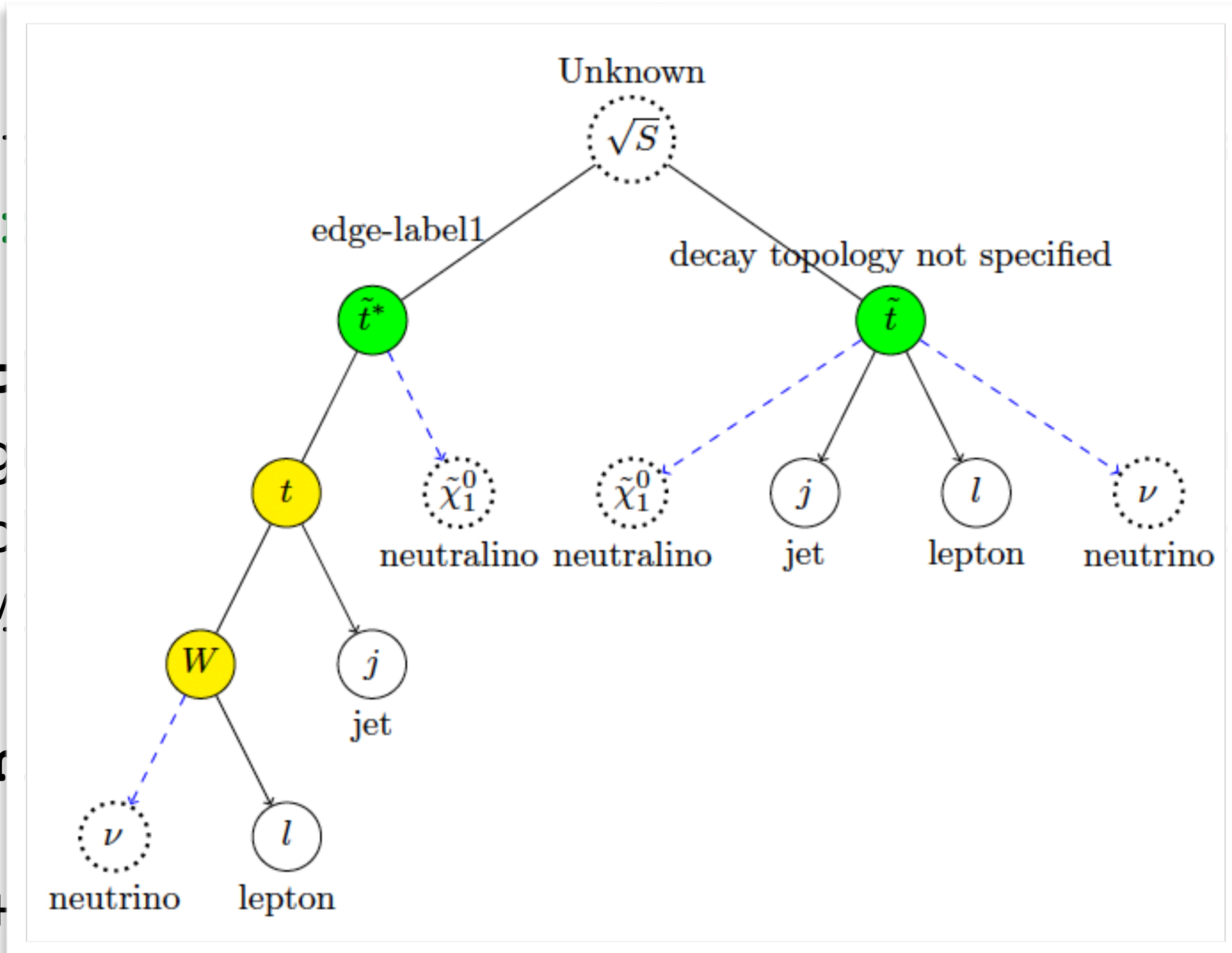
- **Cho, Won Sang (hepkosmos@ibs.re.kr)**
- **Gainer, James S. (gainer@phys.ufl.edu)**
- **Kim, Doojin (immworry@ufl.edu)**
- **Lim, Sung Hak (sunghak.lim@kaist.ac.kr)**
- **Matchev, Konstantin T. (matchev@phys.ufl.edu)**
- **Moortgat, Filip (filip.moortgat@cern.ch)**
- **Pape, Luc (luc.pape@cern.ch)**
- **Park, Myeonghun (parc@apctp.org)**

Event Reconstruction

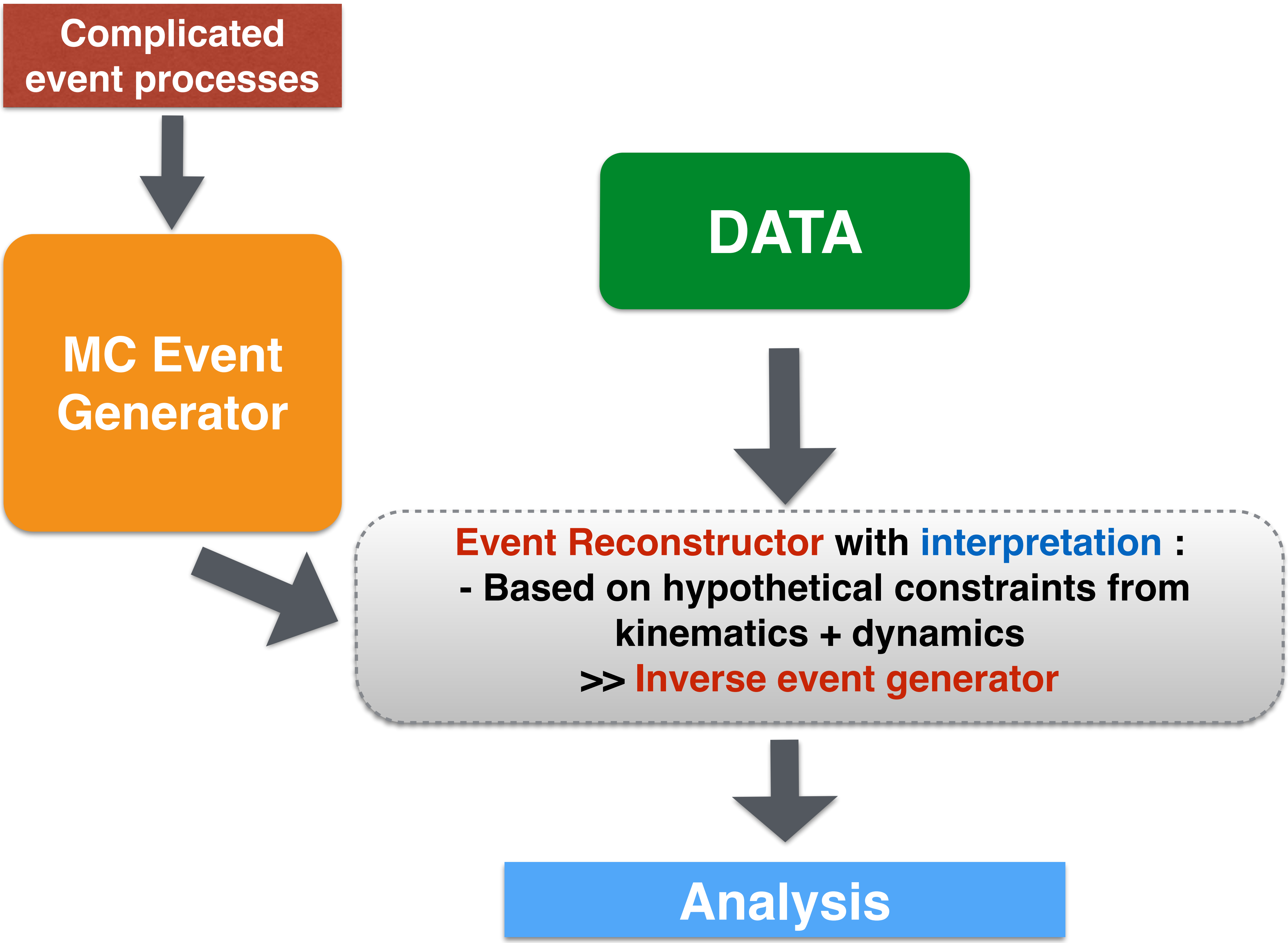
- **In complicated event decay topologies** with multiple invisible particles, **we want to reconstruct :**
  - **the event decay topology**, given a general signature of reconstructed objects (leptons + jets + photons + MET…)
  - **missing momenta**
  - + **invariant masses**

Event Reconstruction

- In … gies wi… , we wa…
  - t … a g… ed c… s + M…
  - m…
  - +…



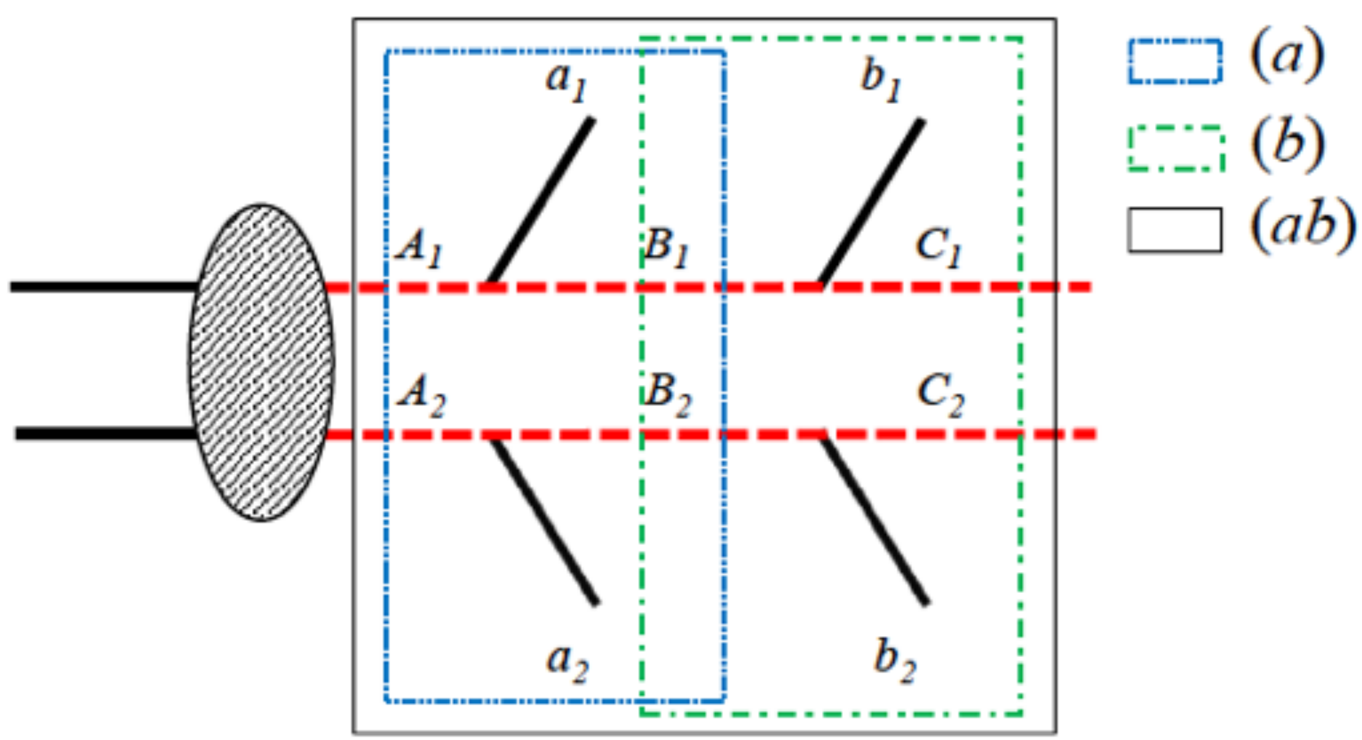
Event Reconstruction in Particle Physics



Mass & Event Reconstruction via Minimisation of Mass Functions

- **Example1) MAOS momentum using MT2 minimisation + on-shell mass relation**
  - Phys.Rev.D79(2009)031701 [0810.4853] : WSC, K.Choi, Y.G.Kim, C.B.Park ; Phys.Rev. D82 (2010) 113017 [1008.2690] : K.Choi, J.S.Lee, C.B.Park ; Phys.Rev. D84 (2011) 096001 [1106.6087] : C.B.Park ; JHEP 1111 (2011) 117 [1109.2201] : K.Choi, D.Guadagnoli, C.B.Park

- **Example2)** with **Constrained-M2 variable** for ttbar-dileptonic decay chain
  - JHEP 1408 (2014)070 [1401.1449] : WSC, J.Gainer, D.Kim, K.Matchev, F.Moortgat, L.Pape, M.Park



- **Power of constrained minimisation (I) : enhanced event saturation to the target mass scale to be measured**

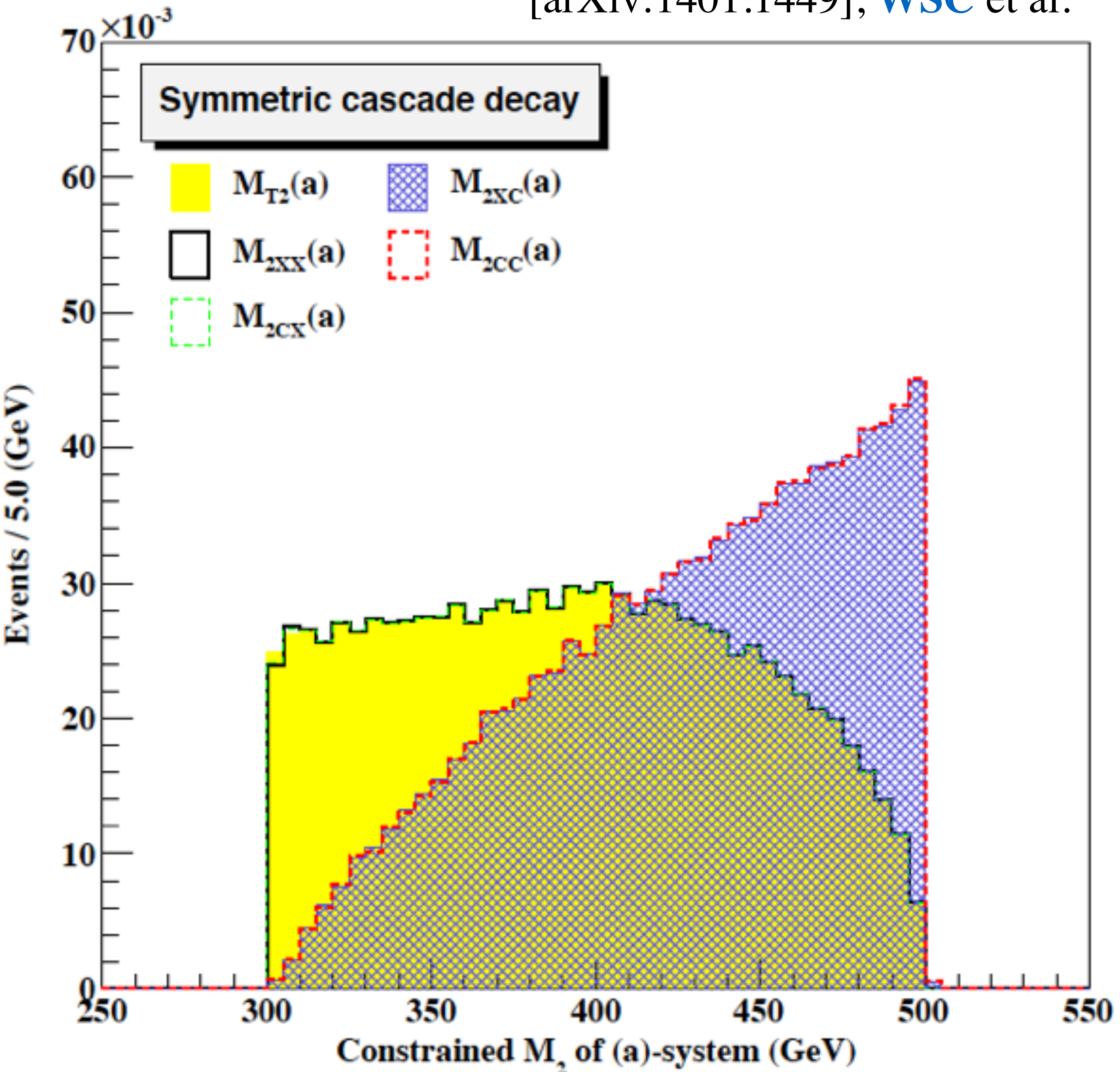
$$M_{2CC} \equiv \min_{\vec{q}_1,\vec{q}_2} \{\max [M_{P_1}(\vec{q}_1, \tilde{m}),\ M_{P_2}(\vec{q}_2, \tilde{m})]\}$$

$$\vec{q}_{1T} + \vec{q}_{2T} = \vec{\not{p}}_T$$

$$M_{P_1} = M_{P_2}$$

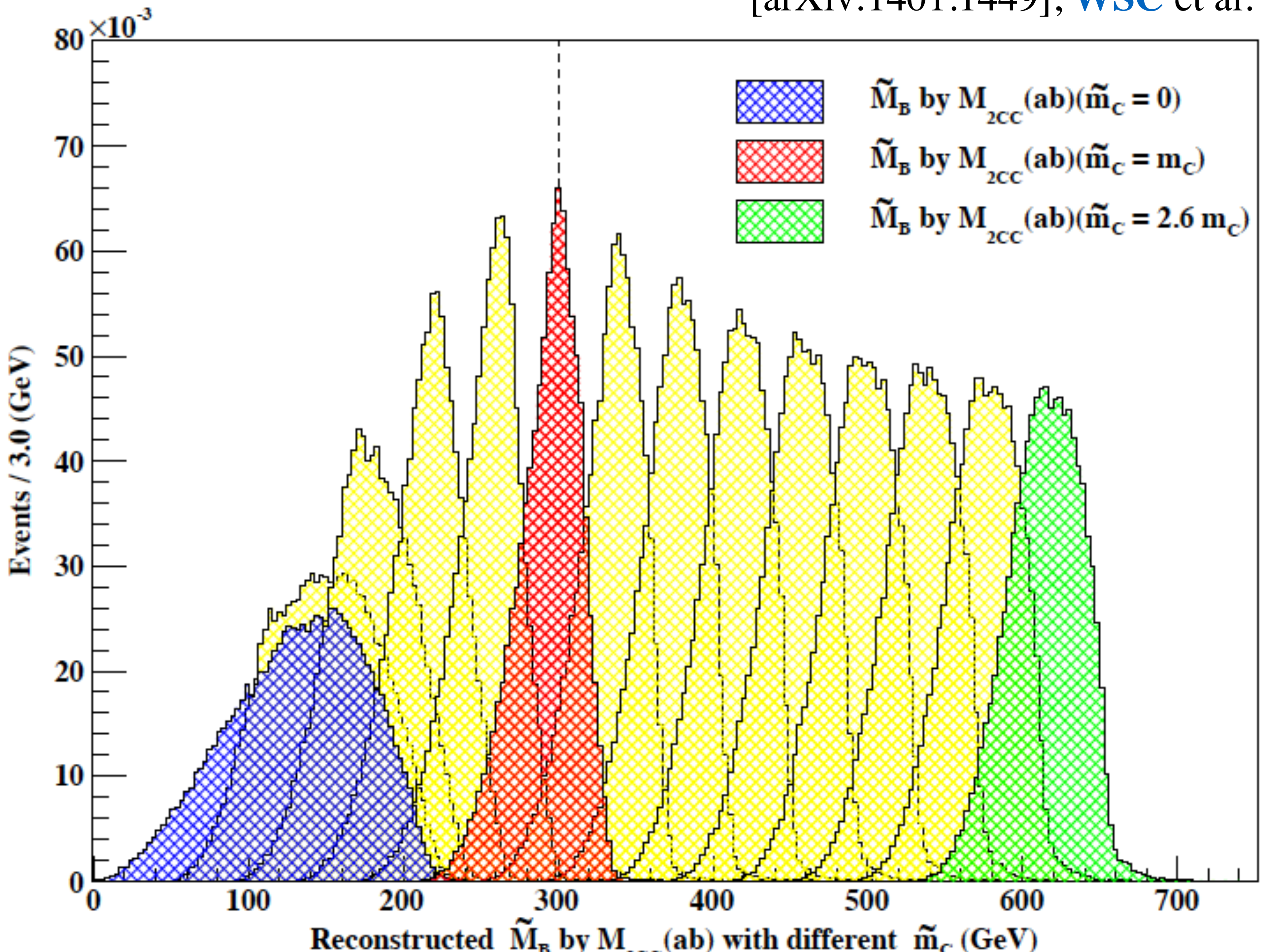
$$M^2_{R_1} = M^2_{R_2}$$

[arXiv:1401.1449], WSC et al.

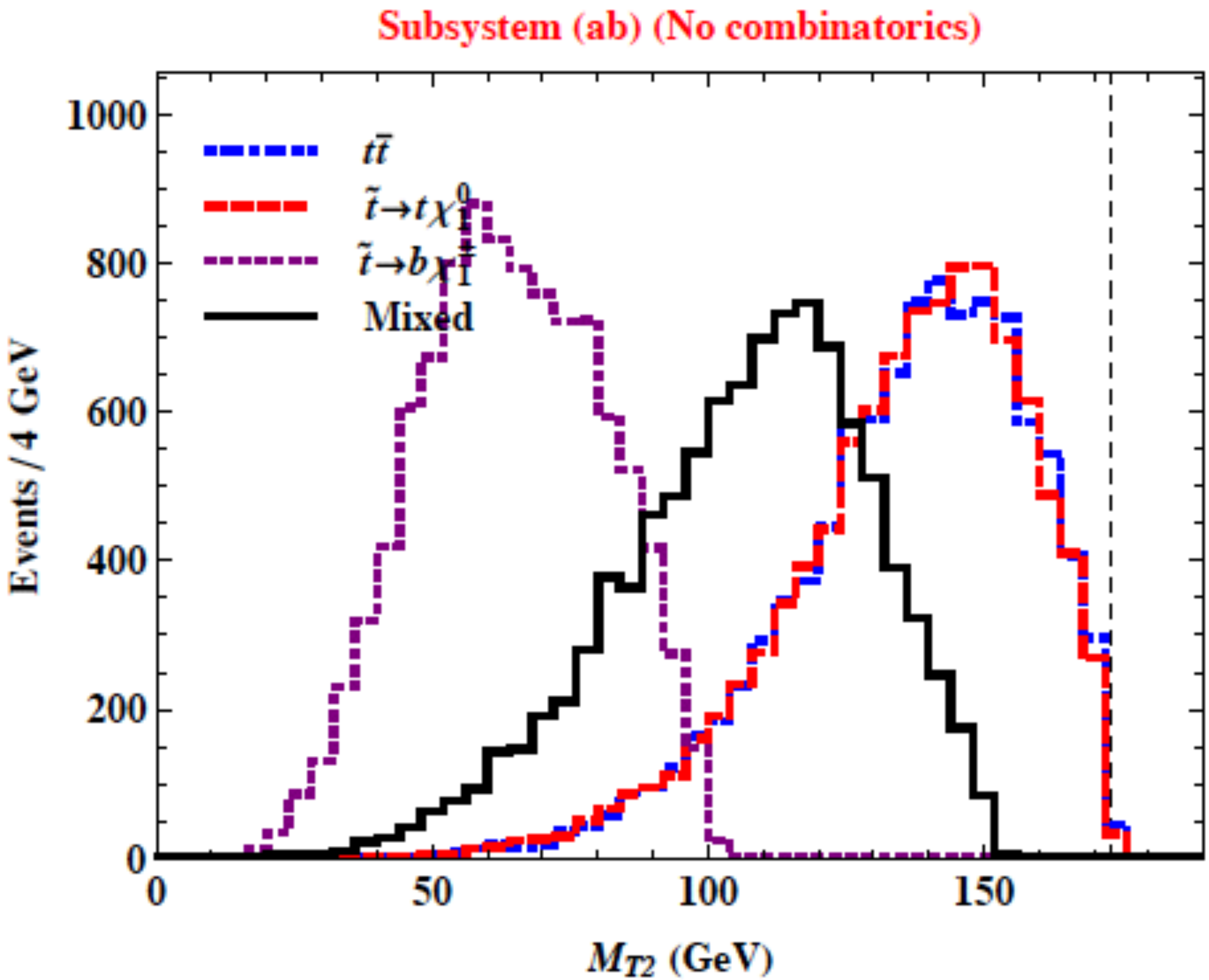


- **Power of constrained minimisation (II) : mass-peak singularity (by true solution) can be restored and utilised for mass measurement, due to the restricted phase space by constraints.**

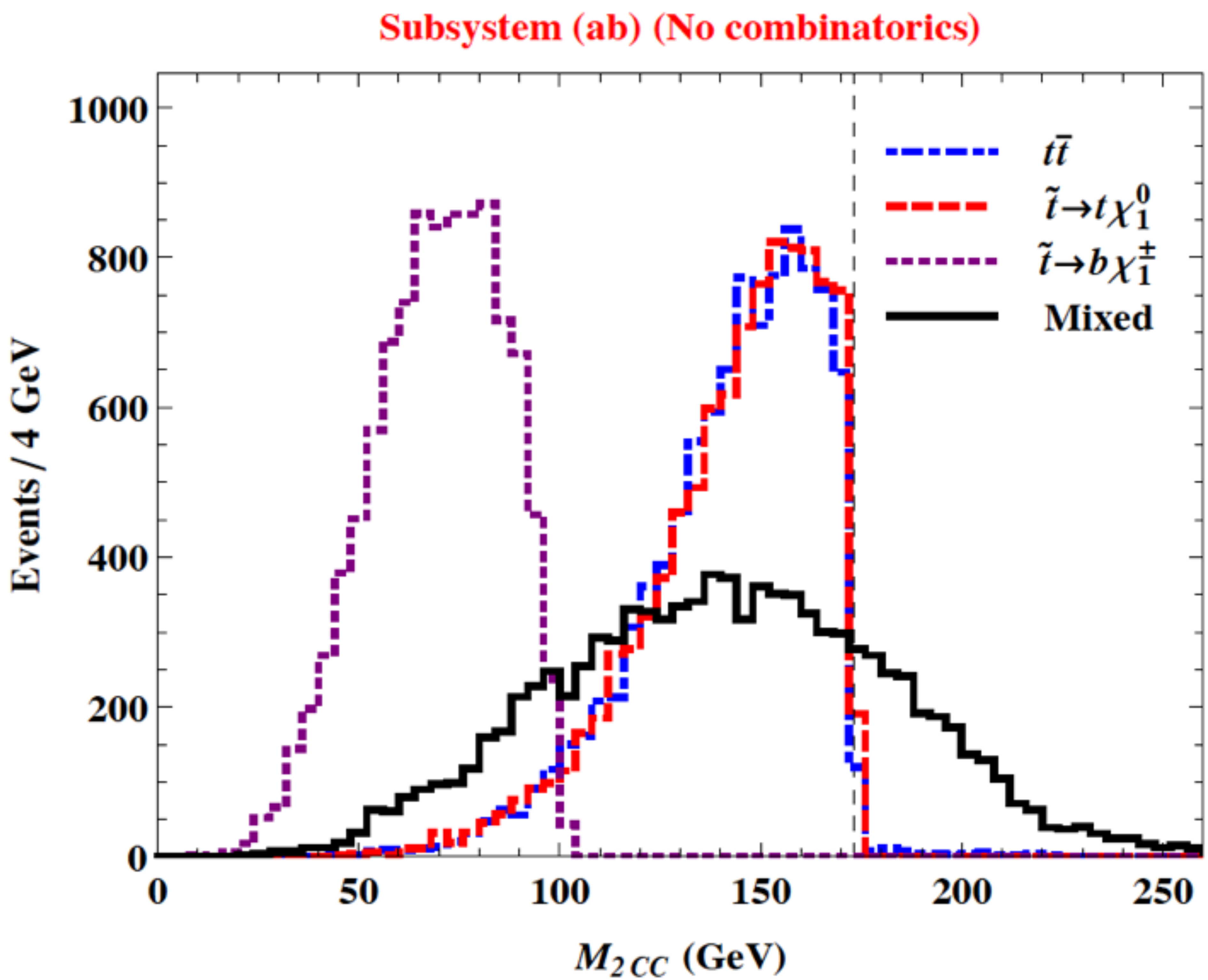
[arXiv:1401.1449], **WSC** et al.



- **Power of constrained minimisation for signal discovery (ex: MT2 vs M2CC)**
  - JHEP 1505 (2015) 040 [1411.0664] D. Kim et al, on ‘Violation of the ttbar endpoint by stop events’



- **Power of constrained minimisation for signal discovery (ex: MT2 vs M2CC)**
  - JHEP 1505 (2015) 040 [1411.0664] D.Kim et al, on ‘Violation of the ttbar endpoint by stop events’



A LANDSCAPE – Minimised mass variables calculated with various kinematic constraints for their own event topologies :

- 'MT' (Transverse mass) for a resonance decay with minimal kinematic constraints; W to l(nu)
  - J. Smith, W.L. van Neerven, J.A.M Vermaseren [Phys. Rev. Lett. 50(1983) 1738]
- 'MT2' (Stransverse mass, Cambridge-MT2) for a pair of resonance decays with minimal kinematic constraints; ~l~l to l(N)l(N)
  - C. G. Lester and D. J. Summers, Phys. Lett. B 463 , 99 (1999) [hep-ph/9906349]
  - A. Barr, C. Lester and P. Stephens, J. Phys. G29 , 2343 (2003) [hep-ph/0304226]
  - H. C. Cheng and Z. Han, JHEP 0812 , 063(2008) arXiv:0810.5178 [hep-ph]

- 'M2C' for a pair of 3 body decays with additional Mjj(max) constraint for ~g~g to jj(N) jj(N);
  - G. G. Ross and M. Serna, Phys.Lett. B 665, 212 (2008) arXiv:0712.0943 [hep-ph]
- 'MT*' for h to WW* with on-shell mW constraint;
  - A. J. Barr, B. Gripaios and C. G. Lester, Phys. Rev. Lett. 108, 041803 (2012) arXiv: 1108.3468 [hep-ph]
- 'MT2W' for ttbar to b(W)+bl(nu) with on-shell mW constraint;
  - Y. Bai, H. C. Cheng, J. Gallicchio and J. Gu, JHEP 1207, 110 (2012) arXiv:1203.4813 [hep-ph]
- 'Mmin' for WZ to l(nu) + tau tau to l(nu)j(nu)j(nu);
  - A. Papaefstathiou, K. Sakurai and M. Takeuchi, JHEP 1408 , 176 (2014) arXiv: 1404.1077[hep-ph]

- ‘**M2**’ as a (1+3)-dim generalisation of ‘MT2’. ‘**M2CC**’ for ttbar dileptonic decay chain :
    - A. J. Barr, T. J. Khoo, P. Konar, K. Kong, C. G. Lester, K. T. Matchev and M. Park,Phys. Rev. D84, 095031 (2011) arXiv:1105.2977 [hep-ph]
    - W. S. Cho, J.Gainer, D.Kim, K.Matchev, F.Moortgat, L.Pape, M.Park, JHEP 1505 , 040 (2015) arXiv: 1401.1449 [hep-ph]
- ‘**M2Cons**’ for antler topology :
    - Partha Konar and Abhaya Kumar Swain, arXiv:1509.00298 [hep-ph]

- **also many other interesting variables for non-reconstructable systems, which taking into account combinatorial uncertainty or with modified objective functions and etc…(MT2min, MCT, Razor, Smin, MTperp, MT2perp, MCT2, … mostly being consistent with general minimal constraints)**

- **Some useful reviews :**

    - A. J. Barr and C. G. Lester : A Review of the Mass Measurement Techniques proposed for the Large Hadron Collider," J. Phys. G7, 123001(2010) arXiv: 1004.2732 [hep-ph]

    - C. G. Lester, "Mass and Spin Measurement Techniques (for the Large Hadron Collider)," lectures given at TASI 2011

It's getting too many…

1) Actually, we need a good dictionary of constrained mass variables.

2) Above all, we need a good framework to calculate such minimised mass variables for

a) multiple-resonance decay systems,

b) targeting general decay topologies with multiple invisible particles,

c) subject to maximal kinematic constraints from a hypothetical topology.

So we develope the OPTIMASS!

Constrained Minimisation

- 1) of **mass functions** (**Sung Hak Lim's talk today!**) of mother particle masses :

$$\tilde{M}(p,q) \quad /. \quad \text{p: visible, q: invisible four momenta}$$

- 2) **over invisible momentum d.o.f** : q

- 3) **subject to constraint functions** : $c_i(p,q)$ involved with on-shell / endpoint relations

$$\bar{M} = \min_{q \in R^n} \tilde{M}(p,q) \quad \text{subject to} \quad c_{i=1..m}(p,q) = 0$$

- **For example) MT2**
  - => $\tilde{M}^2 \equiv \max\left[(p_1+q_1)^2, (p_2+q_2)^2\right]$
  - => subject to **minimal constraints with PT conservation**.

Problem of Constrained Minimisation

- **Analytically,** in principle, we can chase solutions using the method of Lagrange multipliers. However, we easily encounter usual cases where analytical approach is not effective.

$$\nabla L(x, \lambda) = 0$$

- **Numerically,** the solution is hard to be obtained by simple miminisation of Lagrange function in (x, λ) toward a local minimum, because the solution is extremum in (x, λ), not stable in general.

Numerical Algorithm

- **Augmented Lagrangian Method**
  - **Modify the problem**
  - **Constrained Minimisation (in x, lambda)**

TO

  - **A series of Unconstrained Minimisation (in x)**

  - **while the constraint conditions are satisfied by the convexification by penalty-terms**
  - **simultaneously, the Lagrange multipliers get updated and evolved, iteration by iteration !!**

Numerical Algorithm

- Augmented Lagrangian with ..

  1) penalty parameter (mu)

  2) augmented Lagrange parameter (lambda):

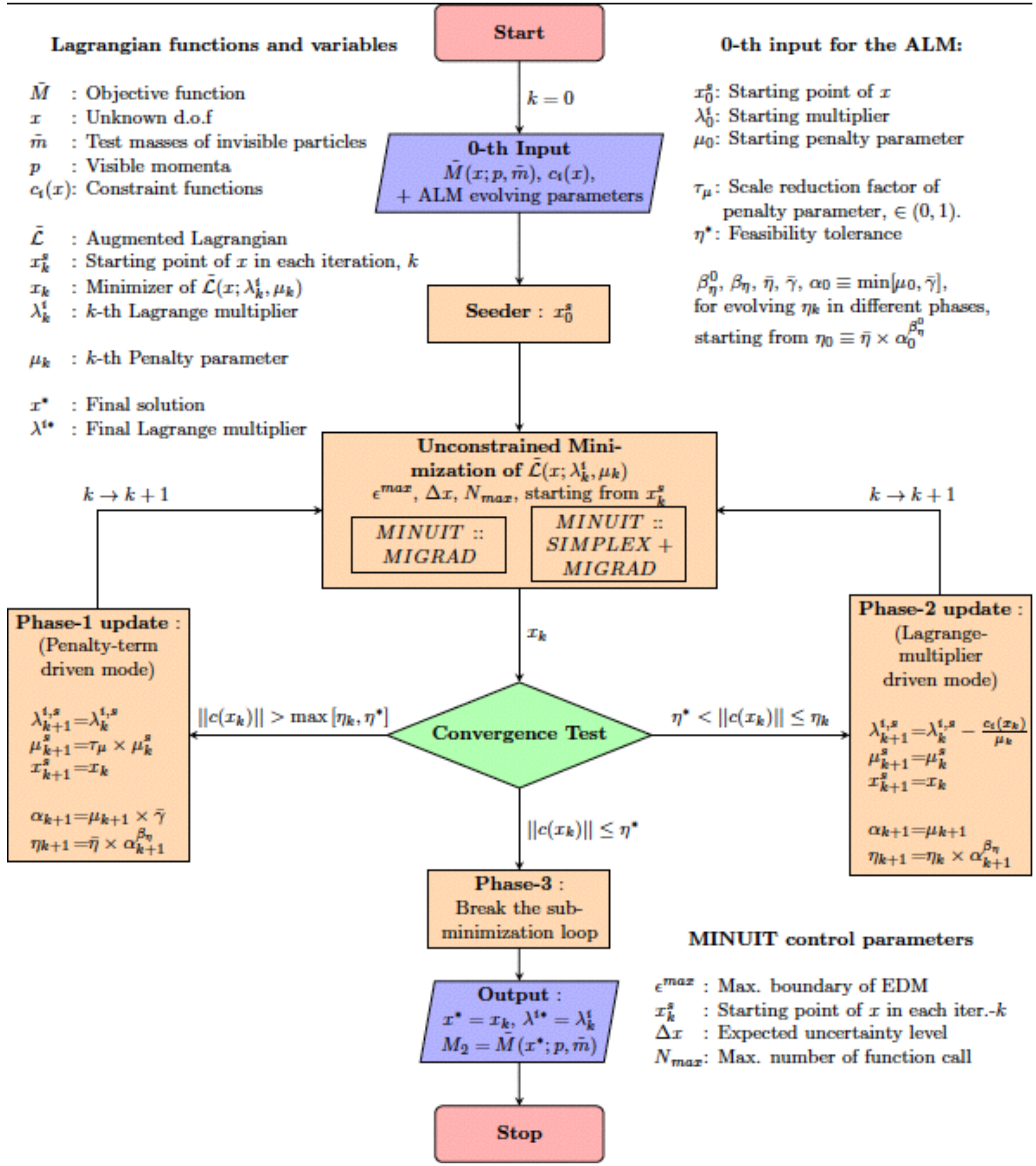
$$\tilde{\mathcal{L}}(\vec{x}\,;\boldsymbol{\lambda},\,\mu) \equiv f(\vec{x}) - \sum_a \lambda_a c_a(\vec{x}) + \frac{1}{2\mu}\sum_a c_a^2(\vec{x})$$

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

Our prescription for ALM – I

- **ALM Loop**
  - In each loop, unconstrained minimisation by **MINUIT**, toward a local minimum in X
  - In each loop, solution's phase and convergence are checked
    - **Optimality** convergence
    - **Feasibility** convergence
  - Evolution in **Phase1**, **driven by penalty term**
  - Evolution in **Phase2**, **driven by augmented Lagrange multiplier term**

Flowchart


Lagrangian functions and variables
$\tilde{M}$: Objective function
x : Unknown d.o.f
$\tilde{m}$: Test masses of invisible particles
p : Visible momenta
$c_i(x)$: Constraint functions
$\tilde{\mathcal{L}}$: Augmented Lagrangian
x_k^s : Starting point of x in each iteration, k
x_k : Minimizer of $\tilde{\mathcal{L}}(x; \lambda_k^i, \mu_k)$
λ_k^i : k-th Lagrange multiplier
μ_k : k-th Penalty parameter
x^* : Final solution
λ^{i*} : Final Lagrange multiplier
Start
$k = 0$
0-th Input
$\tilde{M}(x; p, \tilde{m})$, $c_i(x)$,
+ ALM evolving parameters
0-th input for the ALM:
x_0^s: Starting point of x
λ_0^i: Starting multiplier
μ_0: Starting penalty parameter
τ_μ: Scale reduction factor of penalty parameter, $\in (0, 1)$.
η^*: Feasibility tolerance
β_η^0, β_η, $\bar{\eta}$, $\bar{\gamma}$, $\alpha_0 \equiv \min[\mu_0, \bar{\gamma}]$,
for evolving η_k in different phases,
starting from $\eta_0 \equiv \bar{\eta} \times \alpha_0^{\beta_\eta^0}$
Seeder : x_0^s
Unconstrained Minimization of $\tilde{\mathcal{L}}(x; \lambda_k^i, \mu_k)$
ϵ^{max}, Δx, N_{max}, starting from x_k^s
MINUIT :: MIGRAD
MINUIT :: SIMPLEX + MIGRAD
$k \to k+1$
$k \to k+1$
x_k
Phase-1 update : (Penalty-term driven mode)
$\lambda_{k+1}^{i,s} = \lambda_k^{i,s}$
$\mu_{k+1}^s = \tau_\mu \times \mu_k^s$
$x_{k+1}^s = x_k$
$\alpha_{k+1} = \mu_{k+1} \times \bar{\gamma}$
$\eta_{k+1} = \bar{\eta} \times \alpha_{k+1}^{\beta_\eta}$
$||c(x_k)|| > \max[\eta_k, \eta^*]$
Convergence Test
$\eta^* < ||c(x_k)|| \le \eta_k$
Phase-2 update : (Lagrange-multiplier driven mode)
$\lambda_{k+1}^{i,s} = \lambda_k^{i,s} - \frac{c_i(x_k)}{\mu_k}$
$\mu_{k+1}^s = \mu_k^s$
$x_{k+1}^s = x_k$
$\alpha_{k+1} = \mu_{k+1}$
$\eta_{k+1} = \eta_k \times \alpha_{k+1}^{\beta_\eta}$
$||c(x_k)|| \le \eta^*$
Phase-3 : Break the sub-minimization loop
MINUIT control parameters
ϵ^{max} : Max. boundary of EDM
x_k^s : Starting point of x in each iter.-k
Δx : Expected uncertainty level
N_{max}: Max. number of function call
Output :
$x^* = x_k$, $\lambda^{i*} = \lambda_k^i$
$M_2 = \tilde{M}(x^*; p, \tilde{m})$
Stop

Our prescription for ALM – II

- Utilize the **MINIUT** library for unconstrained minimization at each ALM iteration.
- **MINUIT** (by F. James) : Popular code of **function minimization** and data analysis in HEP Community
  - **MIGRAD** and **SIMPLEX** : Main minimization algorithms of MINUIT
    - **MIGRAD** – 'Variable Metric Method' – Gradient Based 'Quasi-Newton Method'
    - **SIMPLEX** – One of the most popular 'Stepping Method'

Validation

- Simple example 1)

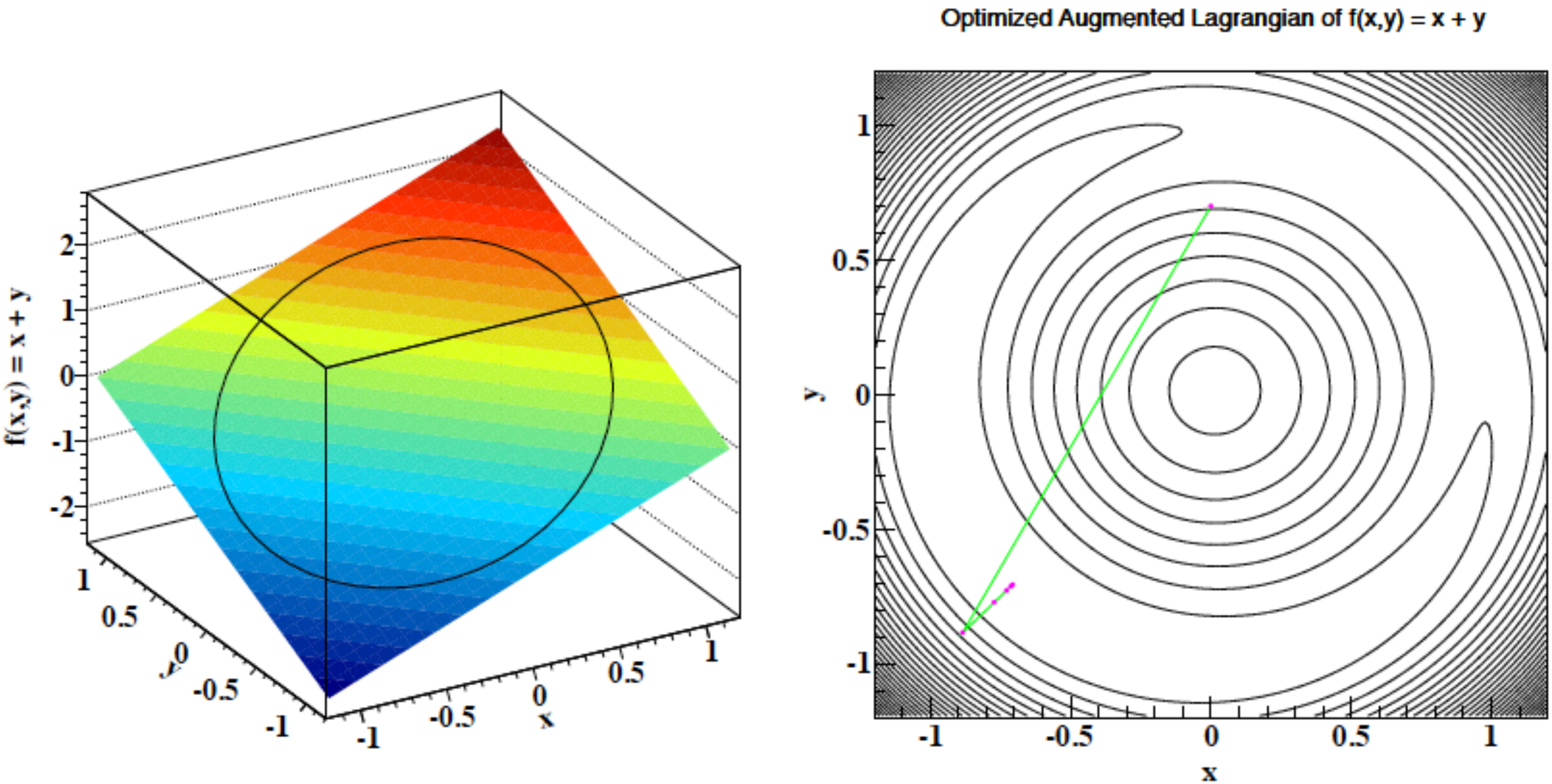


Figure 2. Test of the ALM for the objective function $f(x, y) = x + y$ subject to $x^2 + y^2 = 1$. (a) Plot of the objective function (color coded) and the constraint curve (in black). (b) Contour plot of the augmented Lagragian (3.36) for the last (fifth) iteration. The magenta points denote the minimizers, $\vec{x}_k$, found at each iteration.

Validation

- Example 2) M2CC of ttbar dileptonic decay

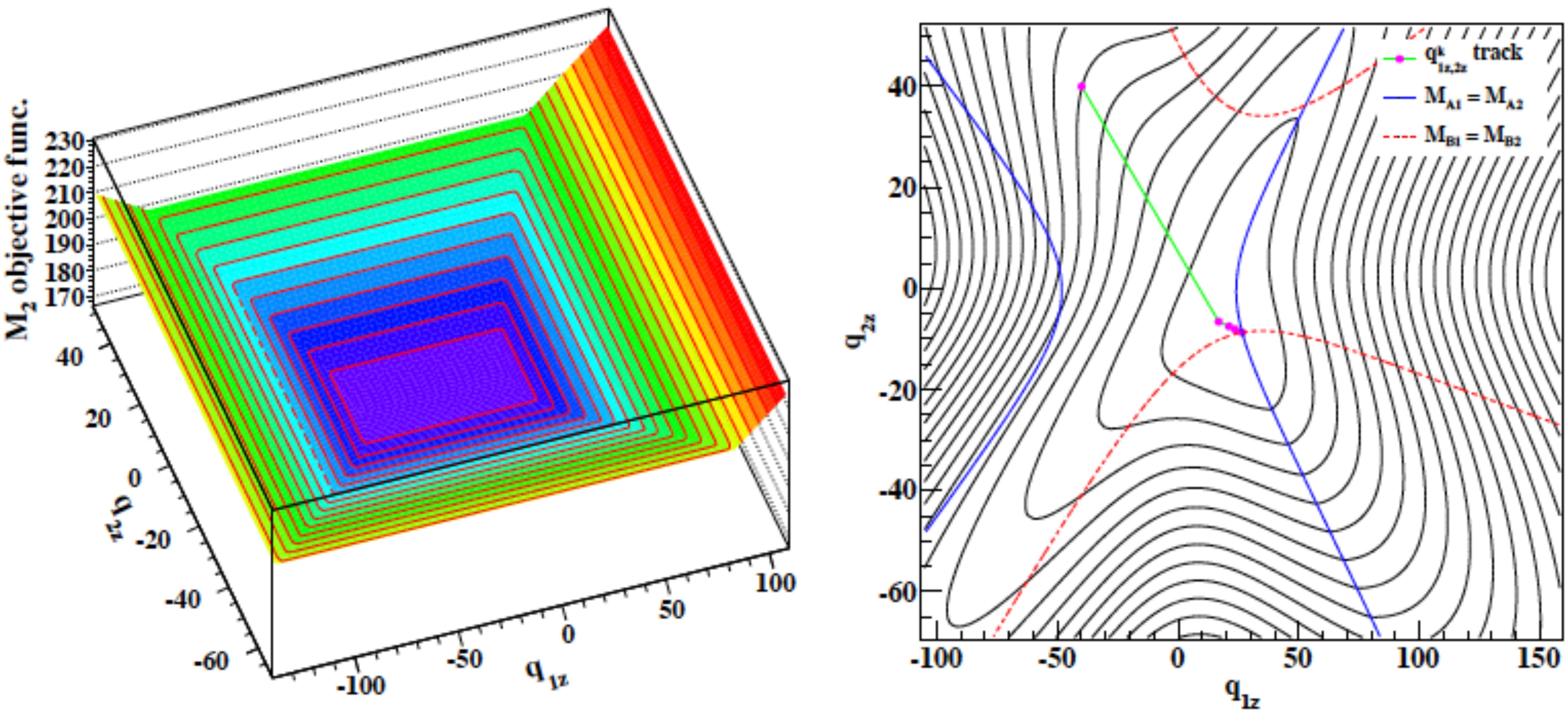


Figure 10. The same as figures 2 and 4, but for the single event considered in section 4.3. Since the objective function has four independent arguments, in order to visualize the evolution of the minimizer, we plot q_{1z} and q_{2z}, having fixed the other two variables, q_{1x} and q_{1y}, to the values which minimize the objective function for the given choice of q_{1z} and q_{2z}.

OPTIMASS-v1 Released!

- Language : C++, Python
- Requirements : gcc(>4.4), Python(>2.6), ROOT with MINUIT2
- Webpage (for download and installation guide):
    - http://hep-pulgrim.ibs.re.kr/optimass

OPTIMASS interface for user's complicated decay topology

- [Full Decay System] Define any number of decay chains, and any type of decay vertices using user's own labelling scheme!

Listing 1: Cards/ttbar-ab.xml

```
# XML
---
<?xml version='1.0' encoding='utf-8'?>
<ProcessCard classname="TTbar_AB" debug="false" version="1.0">
    <!-- ======================= -->
    <!-- Define event decay chain -->
    <!-- ======================= -->
    <DecayChains>
        <DecayChain>
                t1 - b1 w1 , w1 - e1 v1
        </DecayChain>
        <DecayChain>
                t2 - b2 w2 , w2 - e2 v2
        </DecayChain>
    </DecayChains>
```

- **[Subsystem-Mothers]** Define your subsystem's head nodes easily just by **listing the names of (intermediate) mother particles** defined in the full decay system!

```
<!-- ============================================================ -->
<!-- Mother node particle in each decay chain to define objective function -->
<!-- ============================================================ -->
<ParticleMassFunction>
    <ParticleGroup mass_function="M2" group_function="max">
            <Particle label="t1" />
            <Particle label="t2" />
    </ParticleGroup>
</ParticleMassFunction>
```

- **[Subsystem-Effective Invisibles]** Define the **effective invisible nodes by simply tagging** it in the full decay system!

```
<ParticleProperties>
    <Particle name="top" mass="173." />
    <Particle name="bottom" mass="4.18"  />
    <Particle name="wboson" mass="80.419" optimize_target="True" />
    <Particle name="electron" />
    <Particle name="neutrino" invisible="True" />
</ParticleProperties>
```

- **[Kinematic Constraint Functions]** Using the particle names in the full decay chains, their Lorentz 4 momentum d.o.f.(ROOT::TLorentaVector) can freely be used to define constraint functions.

```
<!-- ALM Constraints Configuration -->
<!-- ========================== -->
<Constraints penalty_init="1.">
    <Constraint multiplier_init="0" type="equal">
            w1.M() - w2.M()
    </Constraint>
    <Constraint multiplier_init="0" type="equal">
            t1.M() - t2.M()
    </Constraint>
</Constraints>
```

- **[Combined-Events System Support]** Define multiple PT conservation systems using the full system

```
<!-- =============================== -->
<!-- Subchains for MET conditions -->
<!-- =============================== -->
<ParticleInvisibleSubsystem>
    <Subsystem set_value="manual" >
            <Particle label="t1" />
            <Particle label="t2" />
    </Subsystem>
</ParticleInvisibleSubsystem>
```

OPTIMASS: From Build to Final Run in 6 steps !

- The highest level bullet = top directory of OPTIMASS
- item in BLUE : srcs which need user's input
- item in GREEN : just run the commands!
- item in RED : important directories or executable
- (D,1) alm_base : ALM CORE SRCS for building LIBRARY (just once at the first time)
  - shell> configure; make; make install
    - => Check the ROOT env. with MINUIT2
    - => Install OptiMass library (/lib, /include)

OPTIMASS: From Build to Final Run

- (D) model : User's model repository
  - (D,2) example_models : <user model>.xml : users model files
  - (F,3) model_card.xml : copied from one of '<user model>.xml' files above.
  - (D) dict_src : <user model>.cpp / .h : output dictionary srcs for the model_card.xml
  - (D) main_src : main_<user model>.cpp : output templates for the model_card.xml, for main.cpp
  - (D) model_interpreter : python interpreters/code generators
- (F,4) build_model_dictionary : user's model card reader and related dictionary code generator
  - shell> build_model_dictionary (=> default input (model_card.xml) to output-srcs at dict_src, main_src)

OPTIMASS: From Build to Final Run

- (F,5) main.cpp : customised main event interface from the skeleton main_<user model>.cpp

- (F,6) Makefile : customised Makefile for user's main.cpp, to include additional personal srcs

  - shell> make

  - shell> ./optimass (=> optimass calculation!)

Conclusion :

OPTIMASS as a mass and event reconstructor for hypothetical event topologies.

DATA: [i,j]⟹{??}⟹[visibles]+{invisibles}

↓

OPTIMASS with (general hypothesis –'model_card.xml' for {??})

↓

Physical / Unphysically reconstructed {invisibles} & {node masses}

⟹ **Better discrimination power!**