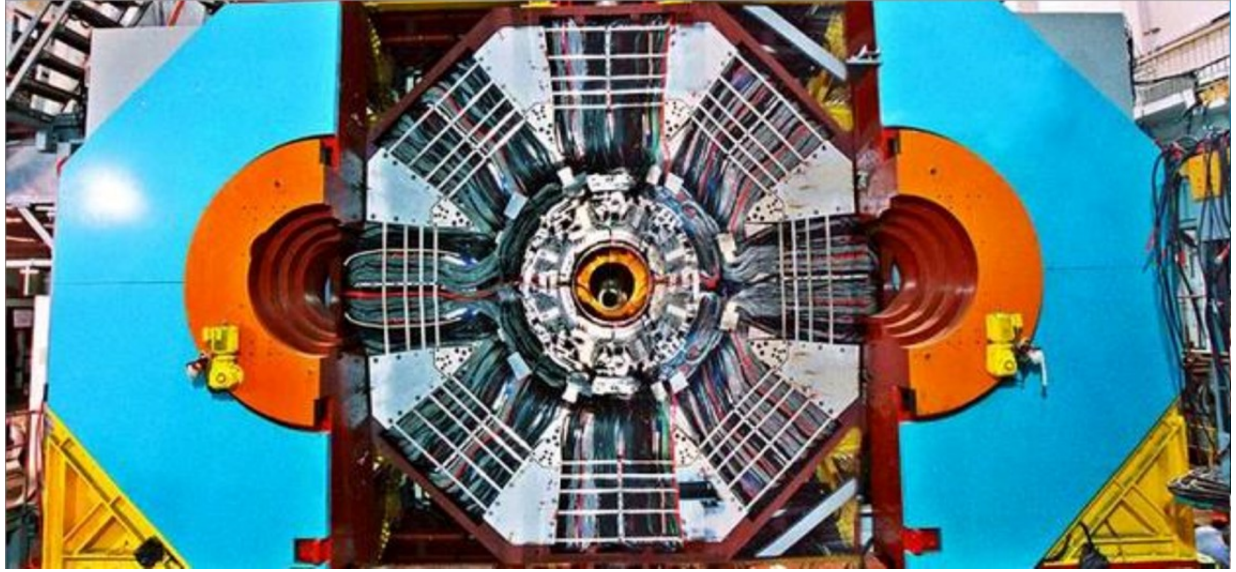


Implementation of amplitude analysis and machine learning at BESIII

Xiao-Rui Lyu（吕晓睿)

University of Chinese Academy of Sciences (UCAS)

(On behalf of the BESIII collaboration)

XVIth Quark Confinement and the Hadron Spectrum Conference

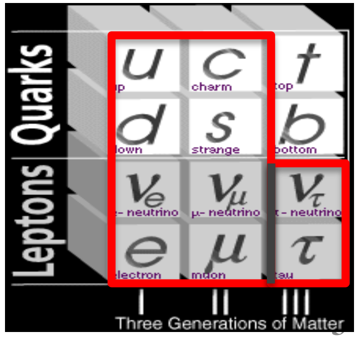
Cairns Convention Centre, Cairns, Queensland, Australia

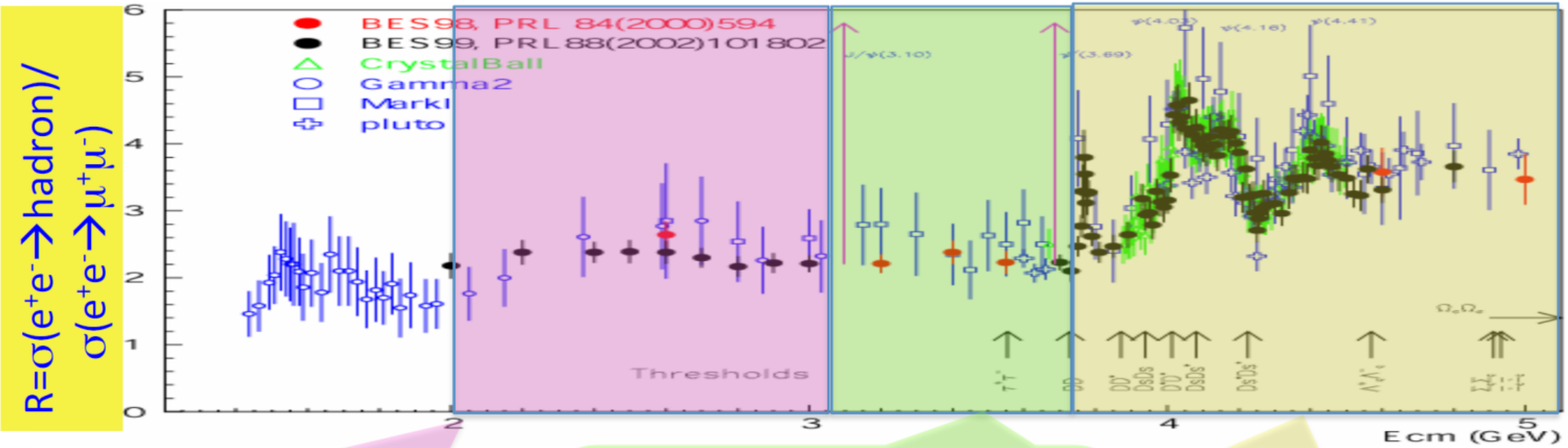
19-24 August 2024 (inclusive)

Outline

- Introduction
- Amplitude analysis tools
- Machine learning
- Summary

Physics at tau-charm Energy Region



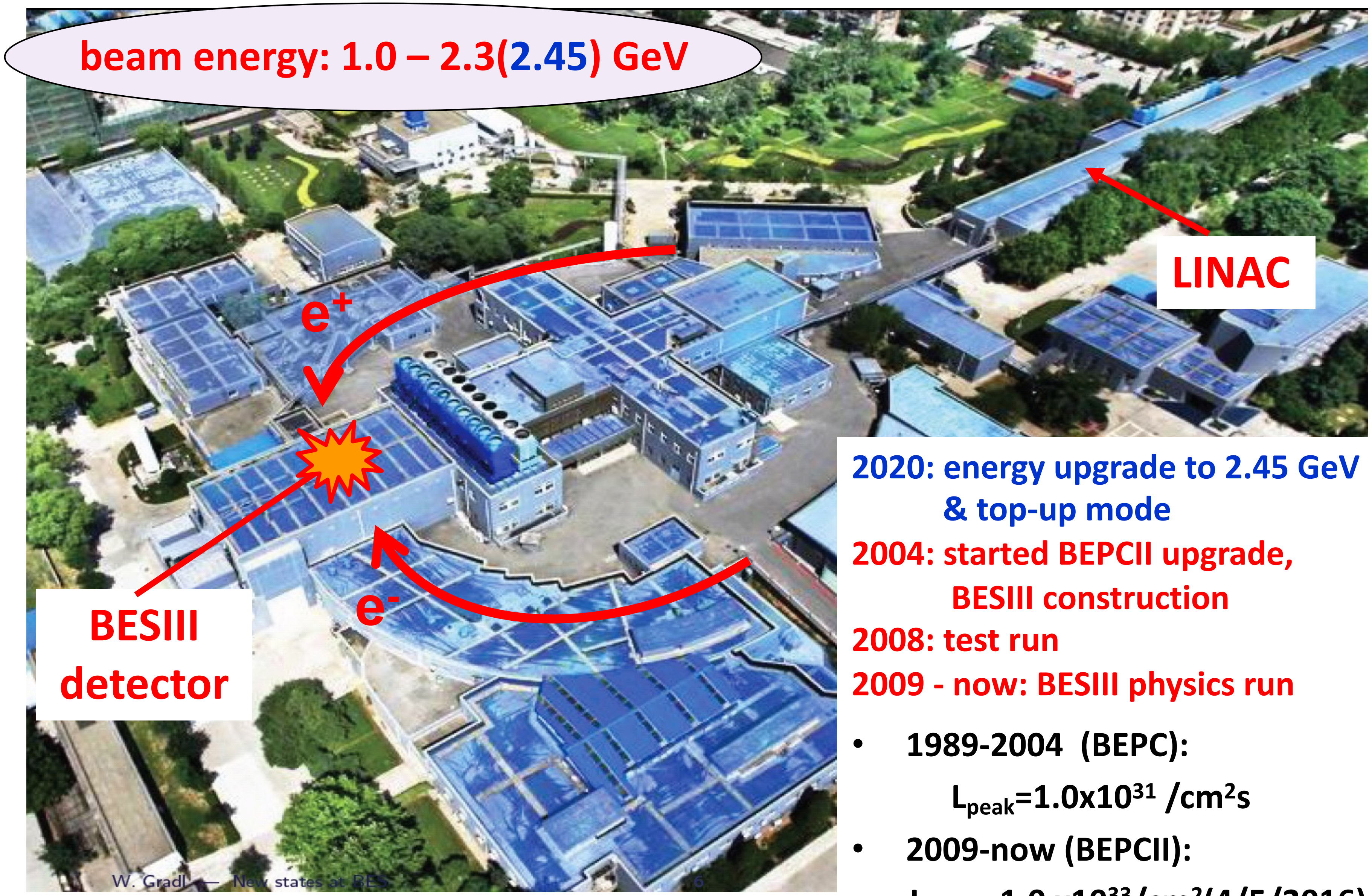


- Hadron form factors
- Y(2175) resonance
- Mutltiquark states with s quark, Zs
- MLLA/LPHD and QCD sum rule predictions

- Light hadron spectroscopy
- Gluonic and exotic states
- Process of LFV and CPV
- Rare and forbidden decays
- Physics with τ lepton

- XYZ particles
- D mesons
- f_D and f_{Ds}
- D_0-$\bar{D}_0$ mixing
- Charm baryons

Beijing Electron Positron Collider (BEPCII)



2020: energy upgrade to 2.45 GeV
& top-up mode
2004: started BEPCII upgrade,
BESIII construction
2008: test run
2009 - now: BESIII physics run

- 1989-2004 (BEPC):
  L_{peak}=1.0x10^{31} /cm^{2}s
- 2009-now (BEPCII):
  L_{peak}= 1.0 x10^{33}/cm^2(4/5/2016)

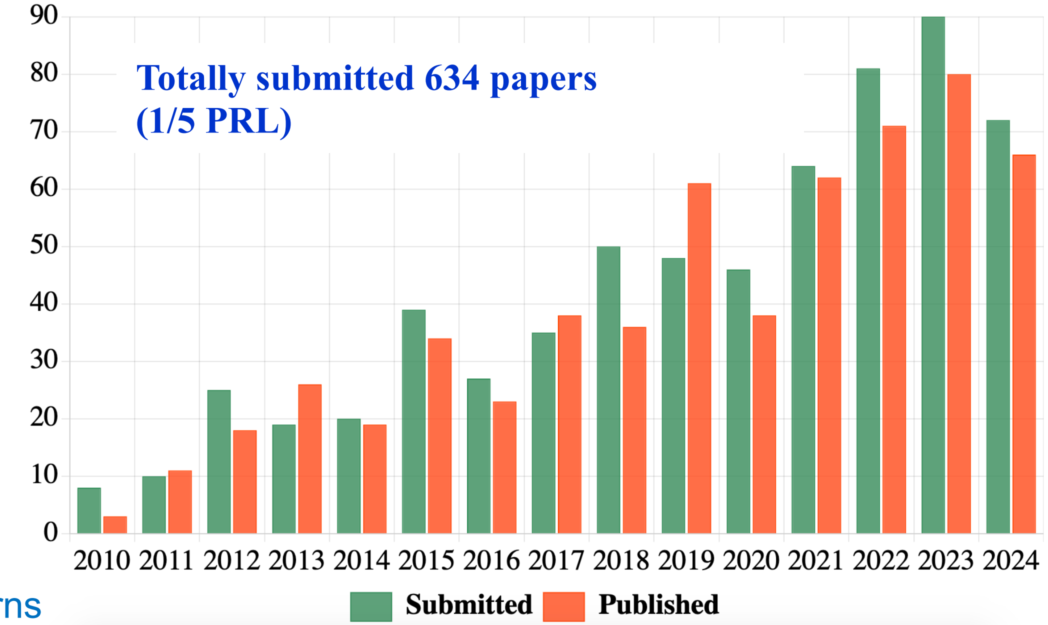
BESIII data sample

2009: 106M $\psi(2S)$
225M J/ψ
2010: 975 pb^{-1} at $\psi(3770)$
2011: 2.9 fb^{-1} *(total)* at $\psi(3770)$
482 pb^{-1} at 4.01 GeV
2012: 0.45B *(total)* $\psi(2S)$
1.3B *(total)* J/ψ
2013: 1092 pb^{-1} at 4.23 GeV
826 pb^{-1} at 4.26 GeV
540 pb^{-1} at 4.36 GeV
10×50 pb^{-1} scan 3.81 — 4.42 GeV
2014: 1029 pb^{-1} at 4.42 GeV
110 pb^{-1} at 4.47 GeV
110 pb^{-1} at 4.53 GeV
48 pb^{-1} at 4.575 GeV
567 pb^{-1} at 4.6 GeV
0.8 fb^{-1} R-scan 3.85 — 4.59 GeV

in total ~55/fb

2015: R-scan 2 — 3 GeV + 2.175 GeV
2016: ~3fb^{-1} at 4.18 GeV (for D_s)
2017: 7×500 pb^{-1} scan 4.19 — 4.27 GeV
2018: more J/ψ *(and tuning new RF cavity)*
2019: 10B *(total)* J/ψ
8×500 pb^{-1} scan 4.13, 4.16, 4.29 — 4.44 GeV
2020: 3.8 fb^{-1} scan **4.61-4.7 GeV**
2021: 2 fb^{-1} scan **4.74-4.95 GeV**; 2.55B $\psi(2S)$
2022: 5 fb^{-1} at $\psi(3770)$
2023: 8.2 fb^{-1} at $\psi(3770)$
2024: ~5 fb^{-1} at $\psi(3770)$; $\psi(3770)$ scan data

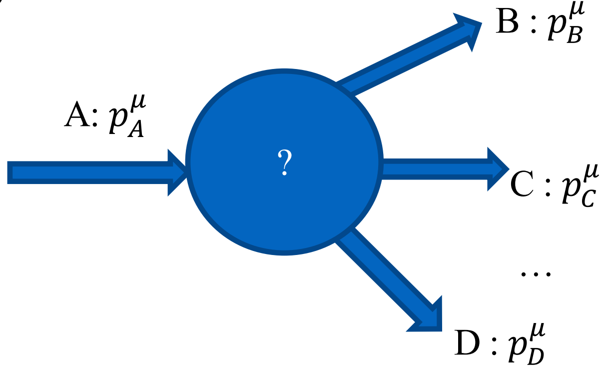
BESIII Publication



Amplitude analysis tools

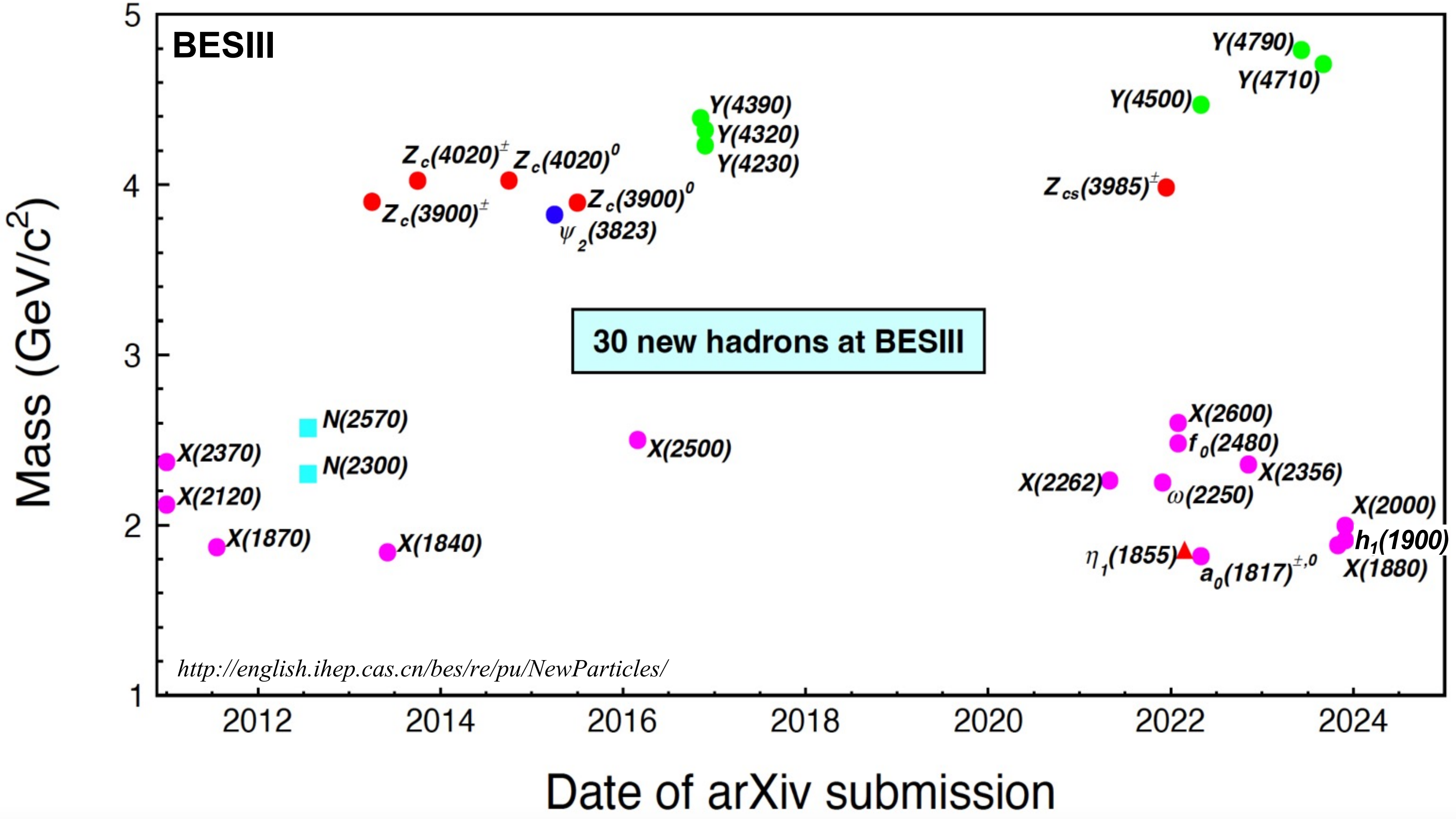
Introduction

- Amplitude analysis / Partial wave analysis (PWA) is a powerful method to study multi-body decay processes, e.g.
  - ✓ to search for (exotic) resonances and measure their properties
  - ✓ to understand CP violation over phase space
- Increasing data statistics and more profound involved physics demand fast PWA fitter and easy coding for different intermediate processes and couple channels
- A general PWA framework using modern acceleration technology (such as GPU, AD, …) is eagerly needed.



$$\sim \boldsymbol{P}(\boldsymbol{p}_B^\mu, \boldsymbol{p}_C^\mu, \boldsymbol{p}_D^\mu; \boldsymbol{\vartheta})$$

Discovered hadrons



Main tools in BESIII

- Closed source / hand coded
  - Tensor formulism: most of charm decays. [$D^+ \to K_S^0\pi^+\pi^0\pi^0$: JHEP09, 077(2023)]
  - Helicity formulism: [$e^+e^- \to \omega\pi^+\pi^-$: JHEP08,159(2023)]
- GPUPWA:
  - First PWA tool based on GPU
  - Used in many PWA of light mesons: [$J/\psi \to \gamma\eta\eta$: PRD87, 092009(2013); $J/\psi \to \gamma\eta\eta'$: PRD106, 072012(2022)]
- FDC-PWA:
  - Feynman Diagram Calculation
  - Used in some baryon final states [$\psi' \to p\bar{p}\eta$:PRD88, 032010(2013); $e^+e^- \to pK^-\bar{\Lambda}$: PRL131, 151901 (2023)]
- TF-PWA:
  - TensorFlow-based, configurable, GPU acceleration, AD
  - as an example: [$\Lambda_c^+ \to \Lambda\pi^+\pi^0$:JHEP12, 033(2022)]
- Other tools:
  - Amptools: [$\chi_{c1} \to \eta\pi^+\pi^-$: PRD95,032002(2017)]
  - PAWIAN: [$e^+e^- \to \phi K^+K^-$: PRD108, 032004 (2023)]
  - ComPWA: [$D^0 \to K_S K^+K^-$: arXiv:2006.02800]

Properties and requirements of PWA tools

- Complex formula
  - Avoid hard coding, automatic formula generation
  - Rule-based amplitude evaluation
  - Constraints in special process
- Multiple dimension.
  - Study relation between many variables, e.g., masses and angles.
  - Proper way to consider resolution
  - Large size MC sample for integration to normalize the PDF.
- Large size of data (e.g.,10B J/ψ decays)
  - Fast calculation to reduce time cost.
  - Distribute the calculation into multi devices.

Configurable

High performance calculation

Configuration

- Why configurable?
  - Global representation for automation and transportability
  - General way to support more decays
- Different level
  - No configuration: hand coding / code templates
  - Decay card like:
    - key-value / command-parameters / structured
    - specify all possible decays (interactions)
    - with addition simplification rules
  - Auto search:
    - provide a large particle database
    - use rules to find all possible intermediate states
    - filter with requirement.

GPUPWA
TensorFlowAnalysis

TF-PWA
PAWIAN
AmpGen

FDC-PWA series
ComPWA series

balance

Automatic,
Simple

Controllable

Symbolic and numerical approaches

- Symbolic approach
  - require a Computer Algebra System (CAS) to simplify formulae
  - write/generate code from CAS outputs
  - procedure: configuration → CAS → formula → generated code → function → amplitude
  - simplifying the formula is difficult and time-consuming
- Numerical approach
  - combine function directly
  - rule based evaluation
  - procedure:
    configuration → function call → amplitude
  - w/o simplified formula, more computation might be required
  - allow caching rule to reduce computation

FDC-PWA series (REDUCE)
ComPWA series (SymPy)

AmpGen
GPUPWA

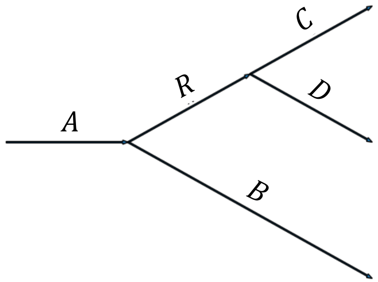
Self hold tensor library

TF-PWA
PAWIAN
TensorFlowAnalysis

TF-PWA: Partial Wave Analysis with TensorFlow

- Fast
  - GPU based
  - Vectorized calculation
  - Automatic differentiation
    Quasi-Newton Method: scipy.optimize
- General
  - Model customization support
- Easy to use
  - Simple configuration file (example provided)
  - Most processing is automatic
  - All necessary functions implemented
  - Rich function support
- Open access
  - https://github.com/jiangyi15/tf-pwa

TF-PWA: amplitude factorization



$$A^{R}_{\lambda_A,\lambda_B',\lambda_C',\lambda_D'}$$

$$\sum_{\lambda} H_{\lambda_R\lambda_B} D^{j_A\star}_{\lambda_A,\lambda_R-\lambda_B}(\varphi_1,\theta_1,0)\, R(M)\, H_{\lambda_C,\lambda_D} D^{j_R\star}_{\lambda_R,\lambda_C-\lambda_D}(\varphi_2,\theta_2,0)$$

$$D^{j_B\star}_{\lambda_B,\lambda_B'}(\alpha_B,\beta_B,\gamma_B) D^{j_C\star}_{\lambda_C,\lambda_C'}(\alpha_C,\beta_C,\gamma_C) D^{j_D\star}_{\lambda_D,\lambda_D'}(\alpha_D,\beta_D,\gamma_D)$$

$$\frac{d\sigma}{d\Phi} \propto \sum_{\lambda_A} \sum_{\lambda_B,\lambda_C,\lambda_D} \left| \sum_{R} A^{R}_{\lambda_A,\lambda_B,\lambda_C,\lambda_D} \right|^2$$

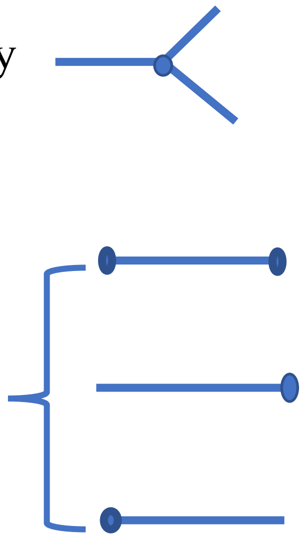
Automatically calculated from decay structure

- automatic factorization of amplitude, as combination of summation and production
- automatic differentiation in likelihood minimization and error propagation
- optional optimization for better performances
  - amplitude caching (eg, resonance lineshape, …)
  - mixed likelihood for simultaneous fit

Feynman rules

Decay

Particle



user defined

$$A^{0\to 1+2} = H_{\lambda_1,\lambda_2} D^{j_0\star}_{\lambda_0,\lambda_1-\lambda_2}(\varphi,\theta,0)$$

Wigner-D matrix

$$R(M) = \frac{1}{m_0^2 - M^2 - i m_0 \Gamma}, \cdots$$

1 or $\rho = 1 + \vec{p}\cdot\vec{\sigma}$

$$D^{j_1\star}_{\lambda_1,\lambda_1'}(\alpha,\beta,\gamma)$$

alignment

probability: $|\mathcal{A}|^2$

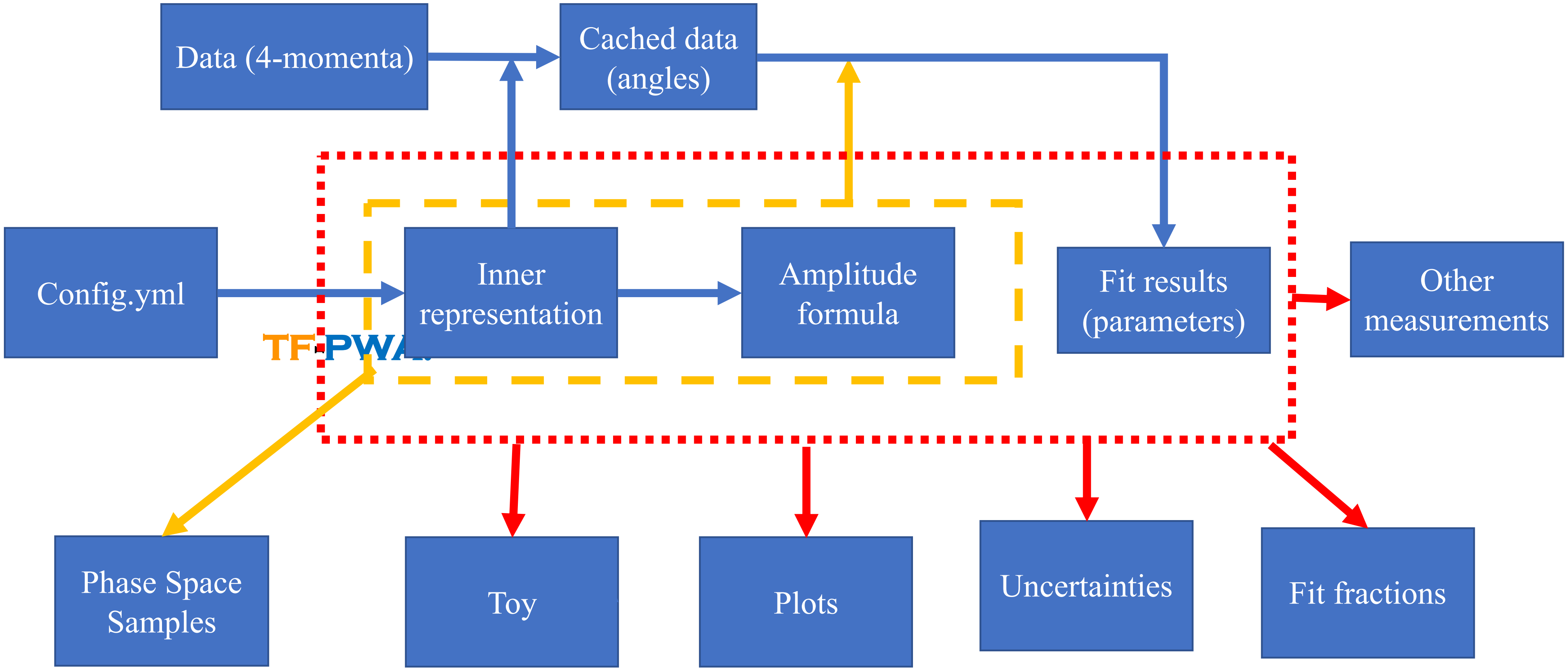
Decay Group: $\mathcal{A} = \tilde{A}_1 + \tilde{A}_2 + \cdots$

Decay Chain: $\tilde{A} = A_1 R A_2 \cdots$

Decay: Wigner D-matrix, $A = HD^{*J}(\phi,\theta,0)$

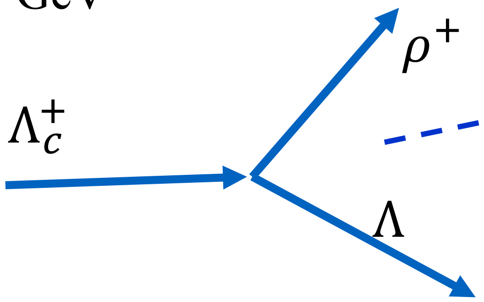
Particle: Breit-Wigner: $R(m)$, user defined

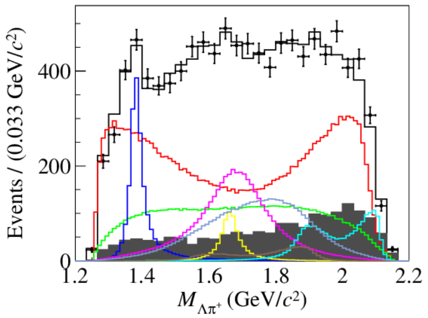
TF-PWA architecture

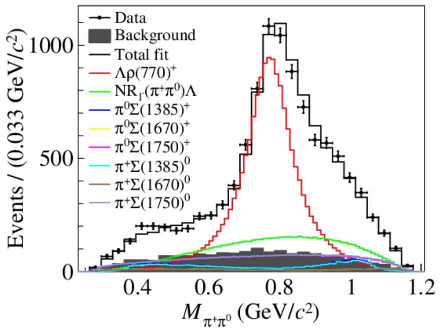


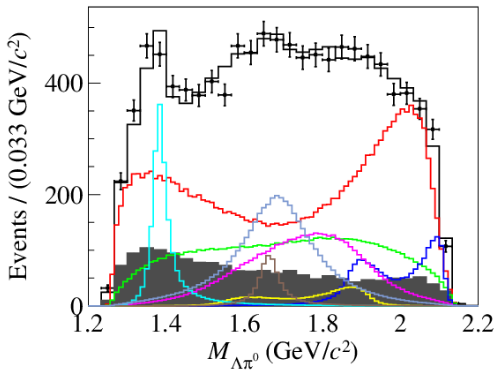
Example fit of $\Lambda_c^+ \to \Lambda\pi^+\pi^0$ JHEP12, 033(2022)

- Simultaneous fit to 7 energy points from 4.6 to 4.7 GeV
  - ✓ in total around 10k events and 854k MC
  - ✓ 38 free parameters
  - ✓ dominated by $\Lambda_c^+ \to \Lambda\rho$: 57.2 ± 4.2%
  - ✓ clear peak for $\Lambda_c^+ \to \pi\Sigma(1385)$









Listing 1: Full view of *config.yml* for $\Lambda_c^+ \to \Lambda\pi^+\pi^0$.

```
decay:
    L_c:
        - [rho, L, p_break: True]
        - [S_p, pi0, p_break: True]
        - [S_0, pip, p_break: True]
    rho: [pip, pi0]
    S_0: [L, pi0]
    S_p: [L, pip]
particle:
    $top: L_c               # initial particle
    $finals: [L, pip, pi0] # final particles
    # rules for replacement
    rho: [ rho(770), NR ]
    S_0: [ Sigma1385z, Sigma1670z, Sigma1750z ]
    S_p: [ Sigma1385p, Sigma1670p, Sigma1750p ]
    # particle proporities
    rho(770):
        J: 1
        P: -1
        mass: 0.77511
        width: 0.1491
        model: GS_rho
    # or import from file
    $include: Resonances.yml
data:
    # particle order in data files
    dat_order: [L, pip, pi0]
    # path of data files
    data: ["data.dat"]
    phsp: ["mc.dat"]
# additional configuration, plot
plot:
    mass:
        S_0: {display: "$M(\\Lambda \\pi^{0})$"}
        S_p: {display: "$M(\\Lambda \\pi^{+})$"}
        rho: {display: "$M(\\pi^{+} \\pi^{0})$"}
```

all decay chains added automatically

settings of properties of particles and resonances

datasets

plotting

```
with config.params_trans() as pt:
    # g1 is fixed to 1
    g2_r = pt["L_c->Sigma1385p.pi0_g_ls_1r"]
    g2_phi = pt["L_c->Sigma1385p.pi0_g_ls_1i"]
    alpha = 2*g2_r*tf.cos(g2_phi) / (1+g2_r*g2_r)
print(alpha, pt.get_error(alpha))
```

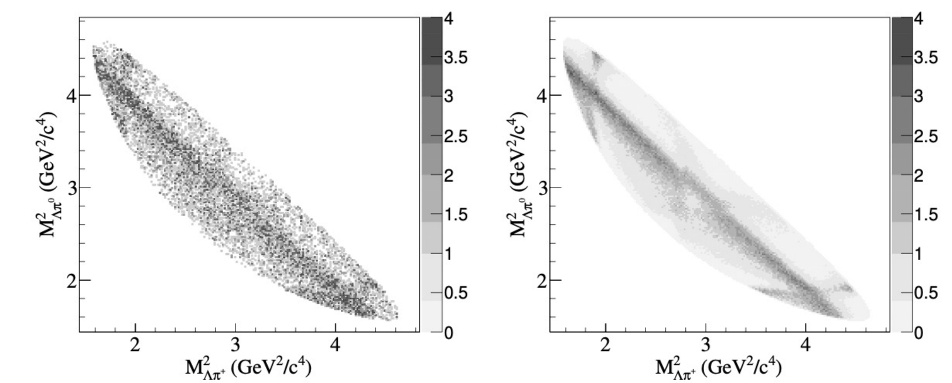
$\frac{1}{2}^+(\Lambda_c^+) \to \frac{3}{2}^+(\Sigma(1385)^+) + 0^-(\pi^0)$

Amplitude	Magnitude	Phase ϕ (rad)
$g_{1,\frac{3}{2}}^{\Sigma(1385)^+}$	1.0 (fixed)	0.0 (fixed)
$g_{2,\frac{3}{2}}^{\Sigma(1385)^+}$	1.29 ± 0.25	2.82 ± 0.18

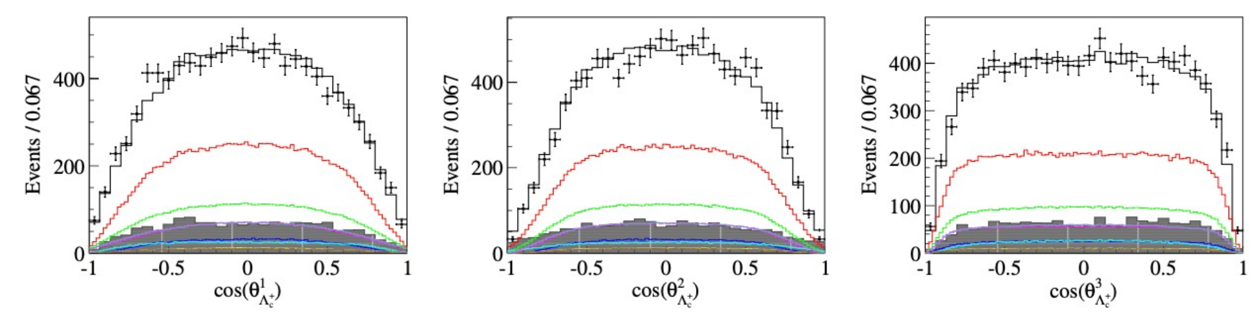
$$\alpha_{\Sigma(1385)\pi} = \frac{|H_{0,\frac{1}{2}}^{\Sigma(1385)}|^2 - |H_{0,\frac{-1}{2}}^{\Sigma(1385)}|^2}{|H_{0,\frac{1}{2}}^{\Sigma(1385)}|^2 + |H_{0,\frac{-1}{2}}^{\Sigma(1385)}|^2} = \frac{2\Re\left(g_{1,\frac{3}{2}}^{\Sigma(1385)} \cdot \bar{g}_{2,\frac{3}{2}}^{\Sigma(1385)}\right)}{|g_{1,\frac{3}{2}}^{\Sigma(1385)}|^2 + |g_{2,\frac{3}{2}}^{\Sigma(1385)}|^2}$$

$$-0.789 \pm 0.098 \pm 0.056$$

Various outputs of the fitting results



Process	Magnitude	Phase ϕ (rad)	FF (%)	Significance
$\Lambda\rho(770)^+$	1.0 (fixed)	0.0 (fixed)	57.2 ± 4.2	$36.9\,\sigma$
$\Sigma(1385)^+\pi^0$	0.43 ± 0.06	-0.23 ± 0.18	7.18 ± 0.60	$14.8\,\sigma$
$\Sigma(1385)^0\pi^+$	0.37 ± 0.07	2.84 ± 0.23	7.92 ± 0.72	$16.0\,\sigma$
$\Sigma(1670)^+\pi^0$	0.31 ± 0.08	-0.77 ± 0.23	2.90 ± 0.63	$5.1\,\sigma$
$\Sigma(1670)^0\pi^+$	0.41 ± 0.07	2.77 ± 0.20	2.65 ± 0.58	$5.2\,\sigma$
$\Sigma(1750)^+\pi^0$	1.75 ± 0.21	-1.73 ± 0.11	16.6 ± 2.2	$10.1\,\sigma$
$\Sigma(1750)^0\pi^+$	1.83 ± 0.21	1.34 ± 0.11	17.5 ± 2.3	$10.2\,\sigma$
$\Lambda + NR_{1^-}$	4.05 ± 0.47	2.16 ± 0.13	29.7 ± 4.5	$10.5\,\sigma$



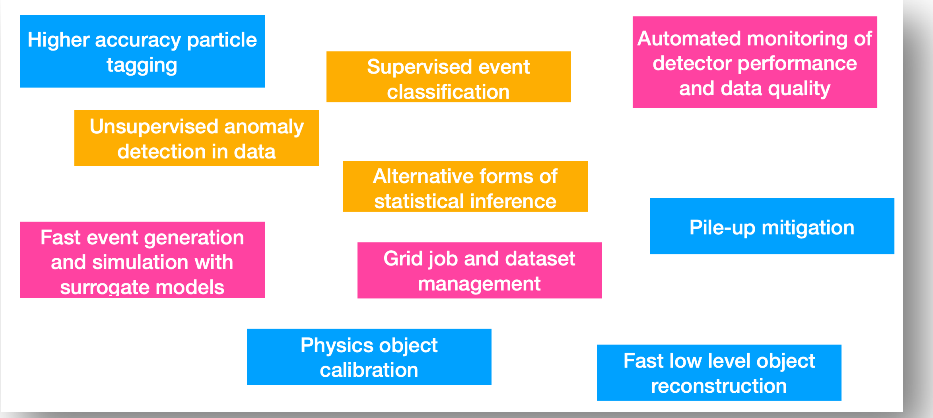
	Result
$\frac{\mathcal{B}(\Lambda_c^+\to\Lambda\rho(770)^+)}{\mathcal{B}(\Lambda_c^+\to\Lambda\pi^+\pi^0)}$	$(57.2 \pm 4.2 \pm 4.9)\%$
$\frac{\mathcal{B}(\Lambda_c^+\to\Sigma(1385)^+\pi^0)\cdot\mathcal{B}(\Sigma(1385)^+\to\Lambda\pi^+)}{\mathcal{B}(\Lambda_c^+\to\Lambda\pi^+\pi^0)}$	$(7.18 \pm 0.60 \pm 0.64)\%$
$\frac{\mathcal{B}(\Lambda_c^+\to\Sigma(1385)^0\pi^+)\cdot\mathcal{B}(\Sigma(1385)^0\to\Lambda\pi^0)}{\mathcal{B}(\Lambda_c^+\to\Lambda\pi^+\pi^0)}$	$(7.92 \pm 0.72 \pm 0.80)\%$
$\mathcal{B}(\Lambda_c^+ \to \Lambda\rho(770)^+)$	$(4.06 \pm 0.30 \pm 0.35 \pm 0.23) \times 10^{-2}$
$\mathcal{B}(\Lambda_c^+ \to \Sigma(1385)^+\pi^0)$	$(5.86 \pm 0.49 \pm 0.52 \pm 0.35) \times 10^{-3}$
$\mathcal{B}(\Lambda_c^+ \to \Sigma(1385)^0\pi^+)$	$(6.47 \pm 0.59 \pm 0.66 \pm 0.38) \times 10^{-3}$
$\alpha_{\Lambda\rho(770)^+}$	$-0.763 \pm 0.053 \pm 0.045$
$\alpha_{\Sigma(1385)^+\pi^0}$	$-0.917 \pm 0.069 \pm 0.056$
$\alpha_{\Sigma(1385)^0\pi^+}$	$-0.789 \pm 0.098 \pm 0.056$

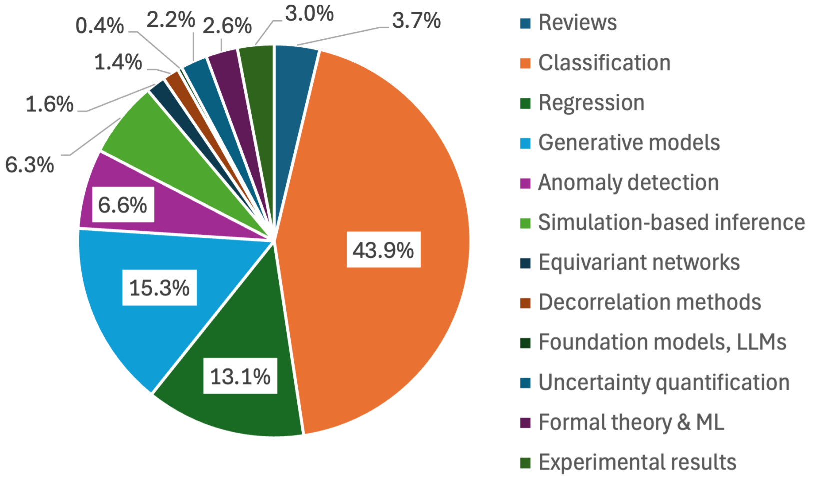
I.F.	$\Lambda + NR_{1^-}$	$\Sigma(1385)^0\pi^+$	$\Sigma(1385)^+\pi^0$	$\Sigma(1670)^0\pi^+$	$\Sigma(1670)^+\pi^0$	$\Sigma(1750)^0\pi^+$	$\Sigma(1750)^+\pi^0$
$\Sigma(1385)^0\pi^+$	-0.50 ± 0.38						
$\Sigma(1385)^+\pi^0$	-0.76 ± 0.36	-0.05 ± 0.04					
$\Sigma(1670)^0\pi^+$	-0.36 ± 0.17	-0.00 ± 0.00	-0.66 ± 0.09				
$\Sigma(1670)^+\pi^0$	-0.34 ± 0.15	-0.58 ± 0.12	0.00 ± 0.00	0.04 ± 0.02			
$\Sigma(1750)^0\pi^+$	-8.1 ± 3.1	-0.03 ± 0.00	0.43 ± 0.07	-0.01 ± 0.00	0.08 ± 0.05		
$\Sigma(1750)^+\pi^0$	-7.2 ± 3.1	0.35 ± 0.08	-0.02 ± 0.00	0.23 ± 0.05	-0.00 ± 0.00	-6.23 ± 0.92	
$\Lambda\rho(770)^+$	-2.7 ± 4.4	-5.94 ± 0.56	-6.01 ± 0.46	0.72 ± 0.29	1.29 ± 0.26	-2.1 ± 1.3	-3.1 ± 1.3

Machine learning

Active directions in HEP Machine Learning

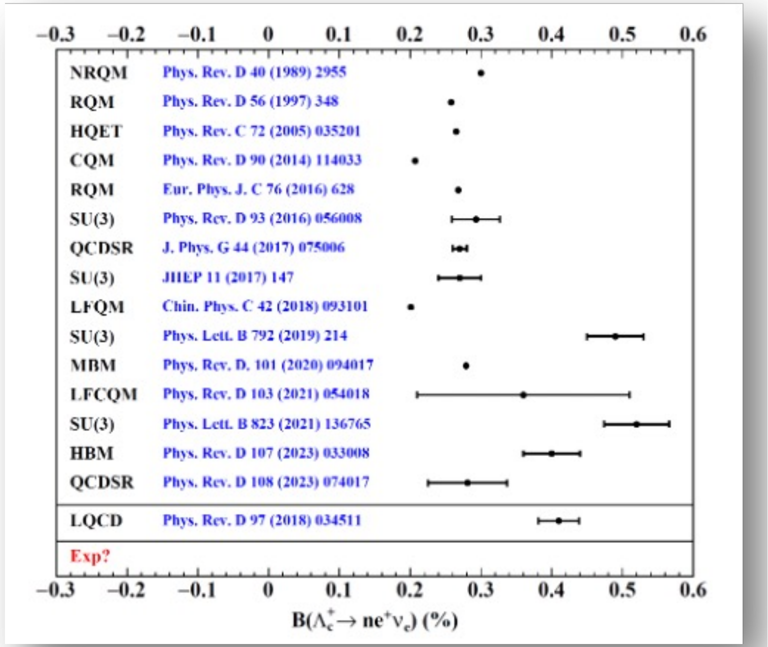
- **Machine Learning (ML) is an increasingly important in many aspects of HEP studies**
- **Ideal platform of BESIII in ML studies:**
  - ✓ large labeled background-free training samples: e.g., 10B J/ψ events
  - ✓ high-quality fully simulated MC samples
  - ✓ rich topology: low-level detector response ➔ particle 4-momentum ➔ full decay tree
  - ✓ energy-momentum conservation in event: hidden symmetry can inspire new ML structures

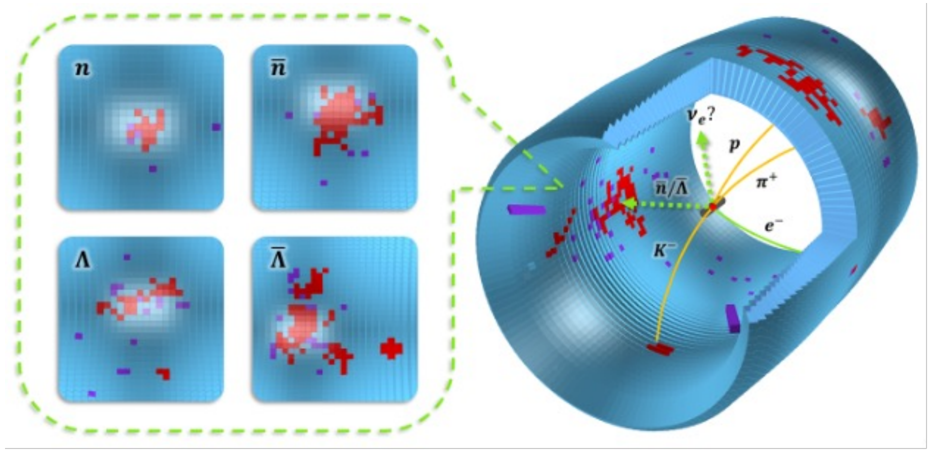


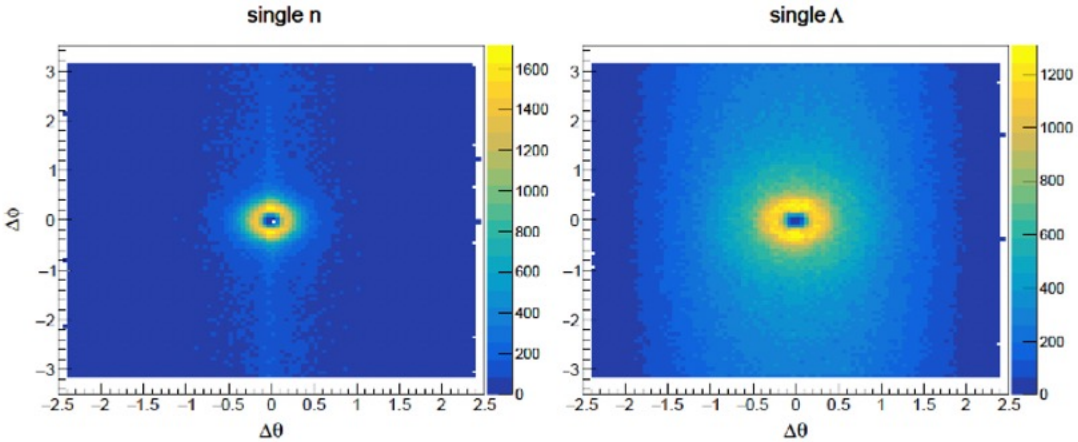


Example: hunting for $\Lambda_c^+ \to n e^+ \nu$

- **Important process of semi-leptonic Λ_c^+ decay to probe strong dynamics in charmed baryon**
- **Challenges:**
  - ✓ neutrino is missing in detection
  - ✓ dominant backgrounds from $\Lambda_c^+ \to \Lambda(\to n\pi^0)e^+\nu$, with ~10x yields than that of the pursuing signals
  - ✓ elusive neutron detection due to neutral charge and contaminations from the photon showers (& noises) in electro-magnetic calorimeter (EMC)
- **Need advanced ML tool to identify neutron showers in EMC**

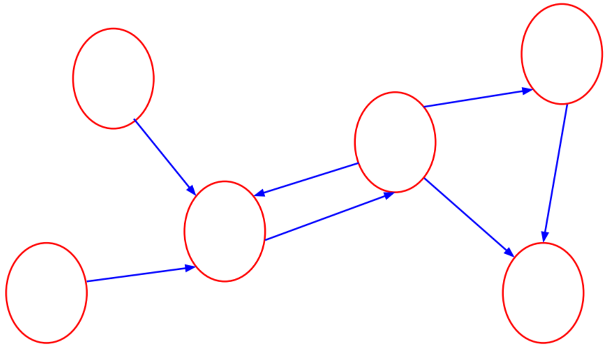


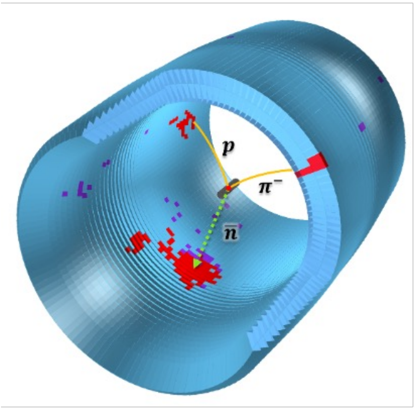




Why Graph Neural Networks (GNN)

- Many neural network architectures are specialized for sequential and image-like data such as RNNs, transformers and CNNs.
- GNN can model more arbitrary relations among data objects by treating them as edges between nodes in a graph.





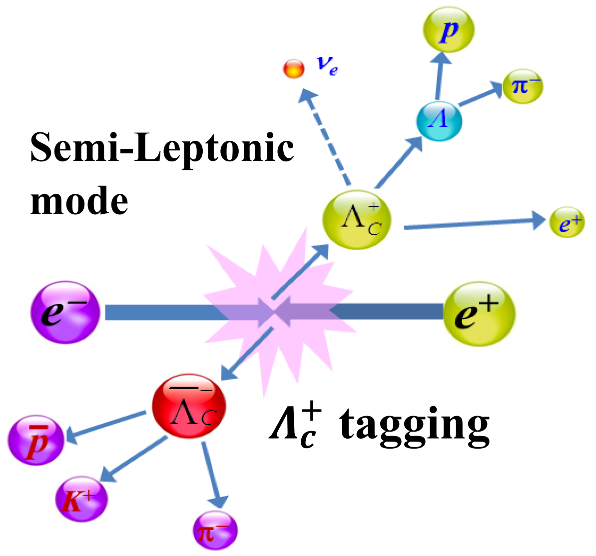
- Sharing of parameters across node and edge updates in the graph.
- Permutation invariance

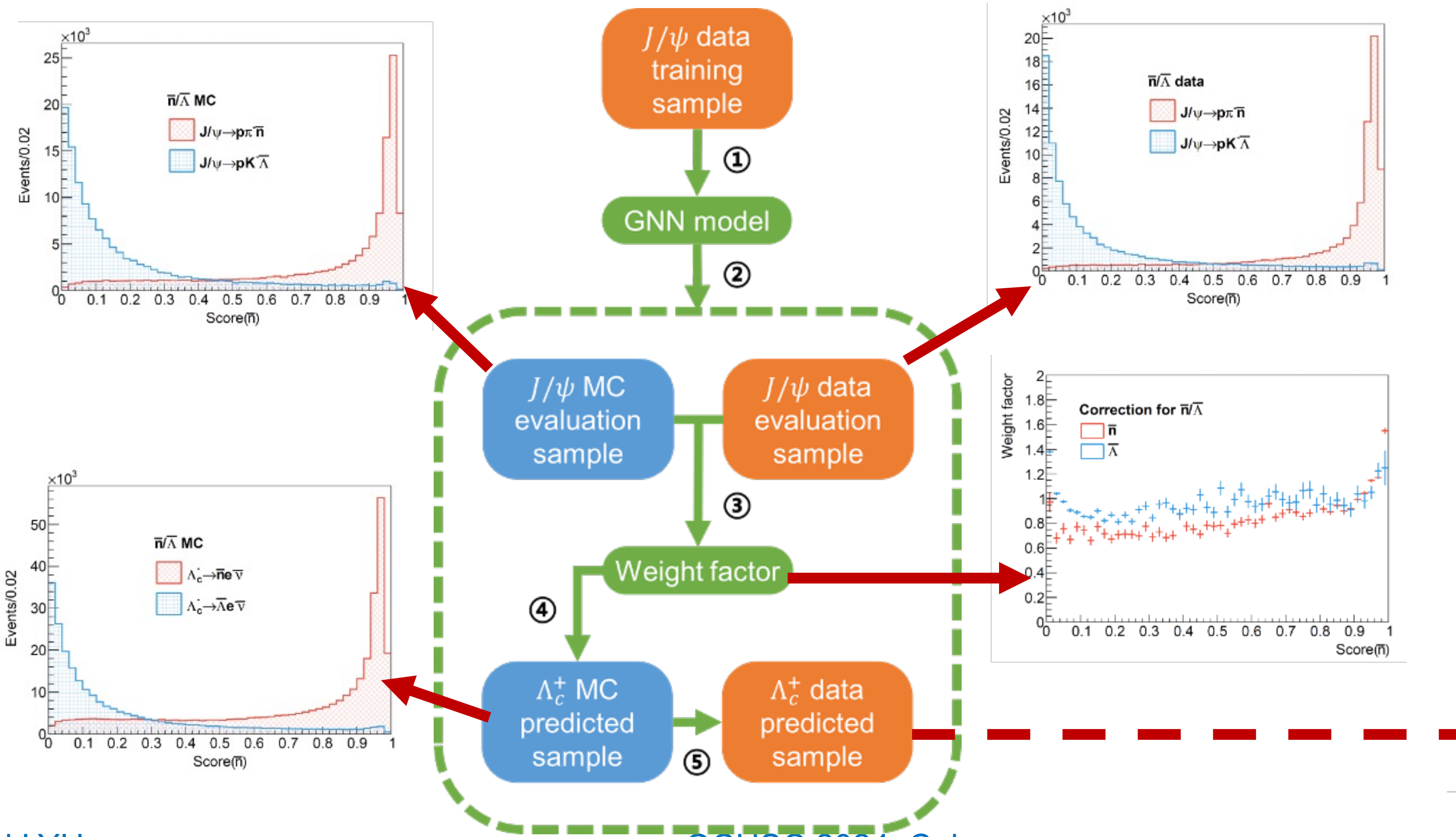
- Nearly unlimited labeled samples
- Structured data
- Clear training objectives

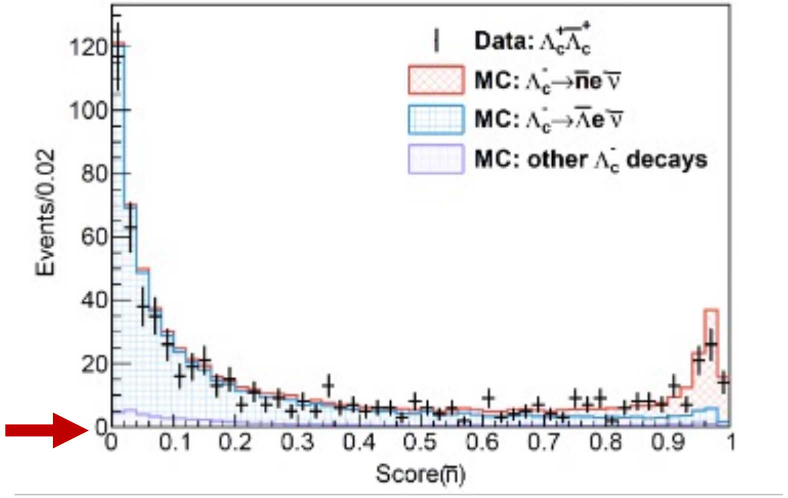
This fits well to the final state particles in physics collisions, where we deal with various objects like tracks/showers and their kinematic relations.

Analysis strategy

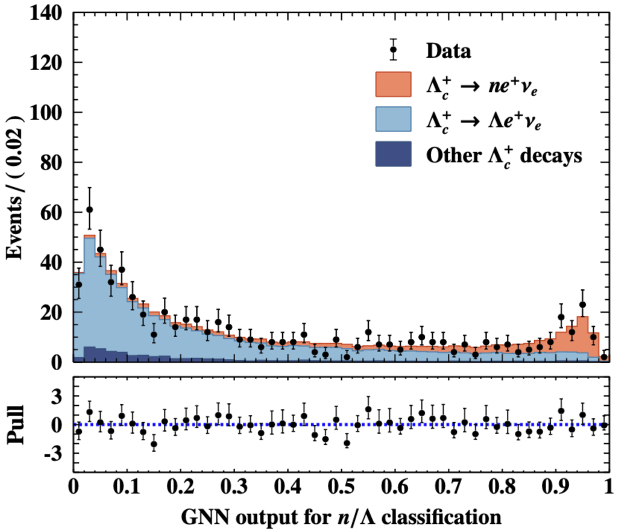
- Threshold Λ_c^+ production: clean environment and Λ_c^+ tagging
- Train GNN with **ParticleNet** using control data from $J/\psi \to \bar{p}n\pi^+, \bar{p}\Lambda K^+$ and c.c. modes based on 10B J/ψ decays

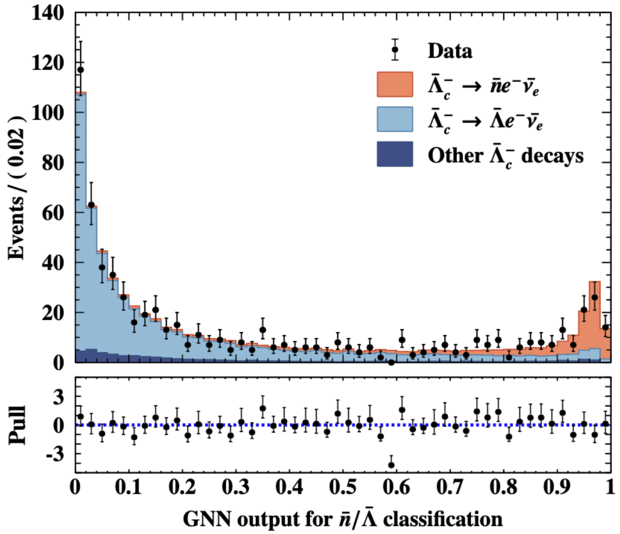


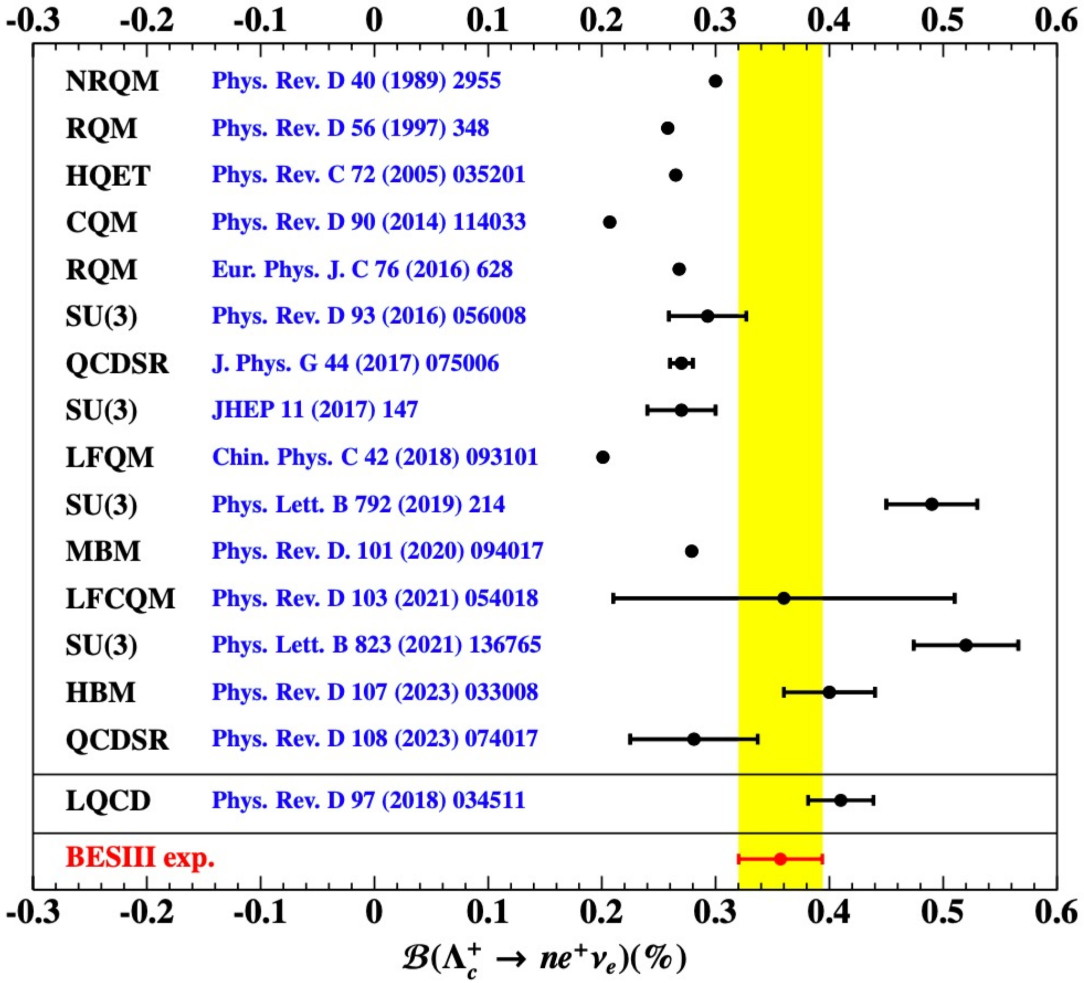




Observation of $\Lambda_c^+ \to n e^+ \nu$





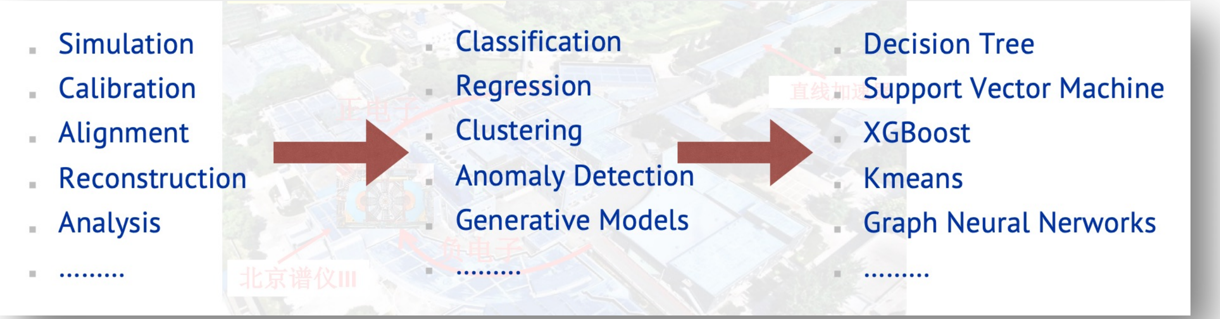


$$\mathcal{B}(\Lambda_c^+ \to n e^+ \nu_e) = (0.357 \pm 0.034_{\rm stat.} \pm 0.014_{\rm syst.})\%,$$ **(>10 σ)**

good control of systematics on GNN training

- **Model settings**: network weight initialization, batch processing sequence and dropout layer are randomly varied
- **Domain shift**: validation of independent control sample via $J/\psi \to \Sigma^+(n\pi^+)\bar{\Sigma}^-(\bar{p}\pi^0)$ and $J/\psi \to \Xi^-(\Lambda\pi^-)\bar{\Xi}^+(\bar{\Lambda}\pi^+)$

Incomplete list of ML efforts at BESIII

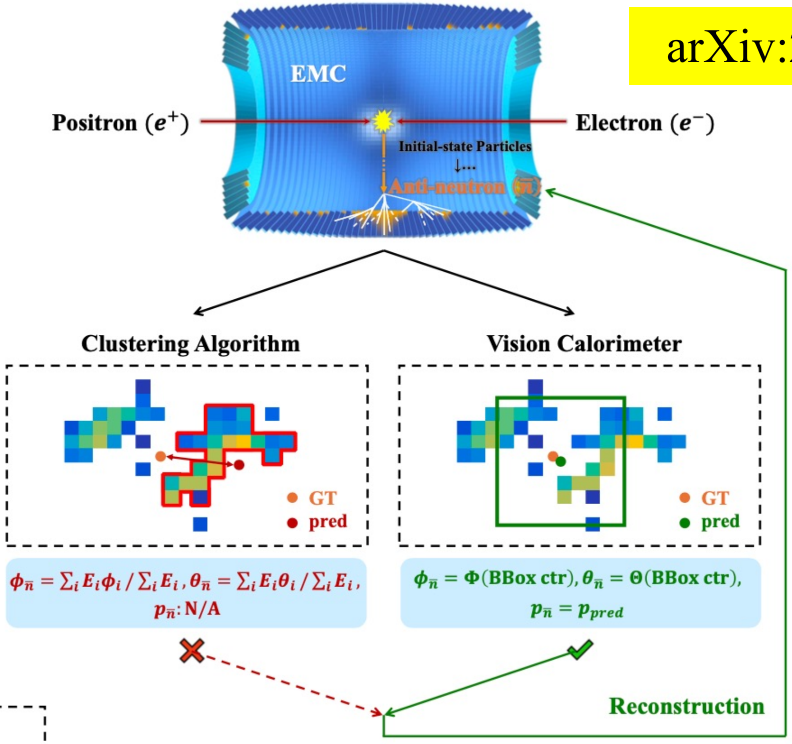


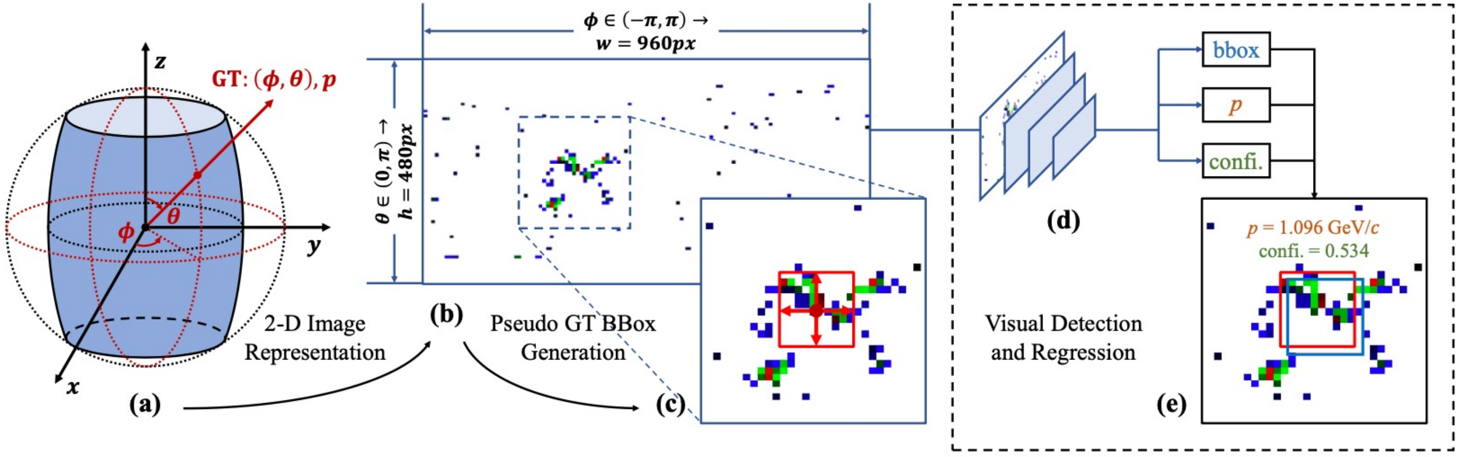
- AI assistant Dr. Sai for BESIII based on LLM Xiwu: *https://drsai.ihep.ac.cn*
- XGBoost regressor for cluster reconstruction in CGEM-IT
- MDC hit noise filtering via GNN
- MDC track clustering based on DBSCAN and RANCAS
- Simultaneous track finding and track fitting with DNN
- XGBoost classifier for particle identifications
- XGBoost for multi-dimensional kinematic reweighting
- Transformer in charm tagging
- Position reconstruction and energy regression for neutron/K_L via advanced visual model
- …

Anti-neutron reconstruction via Vision Calorimeter

arXiv:2408.10599

- We introduce Vision Calorimeter (ViC), the first baseline of deep-learning-based reconstruction method to establish the mapping between EMC readouts and the physical properties of incident particles.
- Two tasks: incident position prediction and incident momentum regression

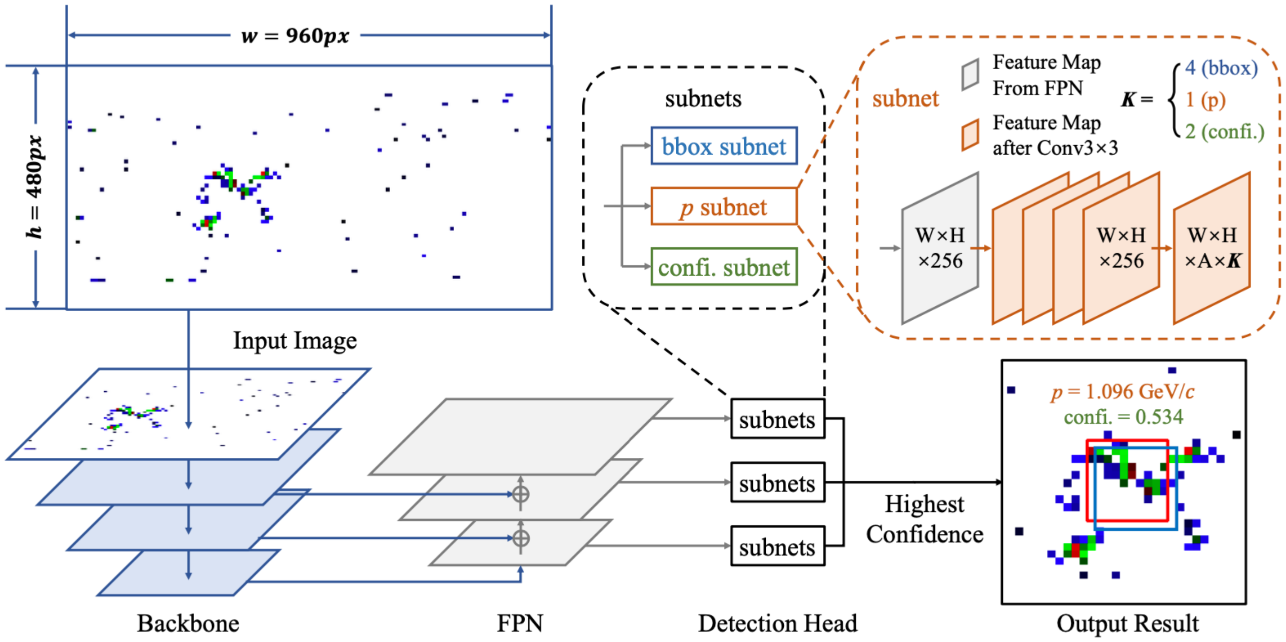


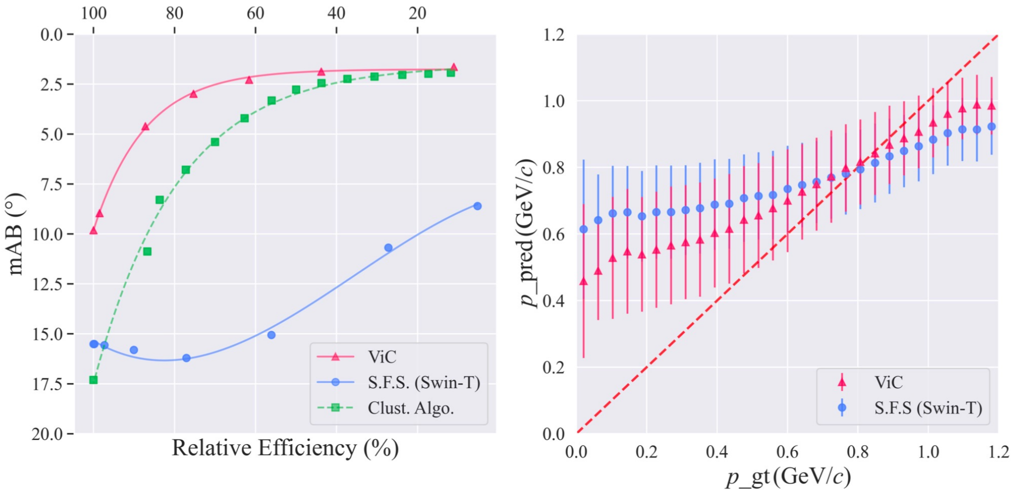


- ✓ EMC hit maps from anti-neutron control sample in data
- ✓ convert Positions and Energies of EMC deposited hits to Pixel Intensities
- ✓ convert each annotated incident position to a pseudo GT BBox
- ✓ predict physical properties of incident particles with a unified deep learning network

Implementation and Performance

- Backbone: Swin Transformer to generate the feature representation
- FPN: manages scale variations
- Detection head: RetinaNet to regress the position and momentum, and estimate confidence levels
- Code is available at https://github.com/yuhongtian17/ViC
- pretrained the backbone network on the vision dataset (i.e., ImageNet)
- baseline experiment on 12 epochs using 4× RTX 4090 GPUs





- significant error reduction in positioning
- first realization of momentum prediction

Summary

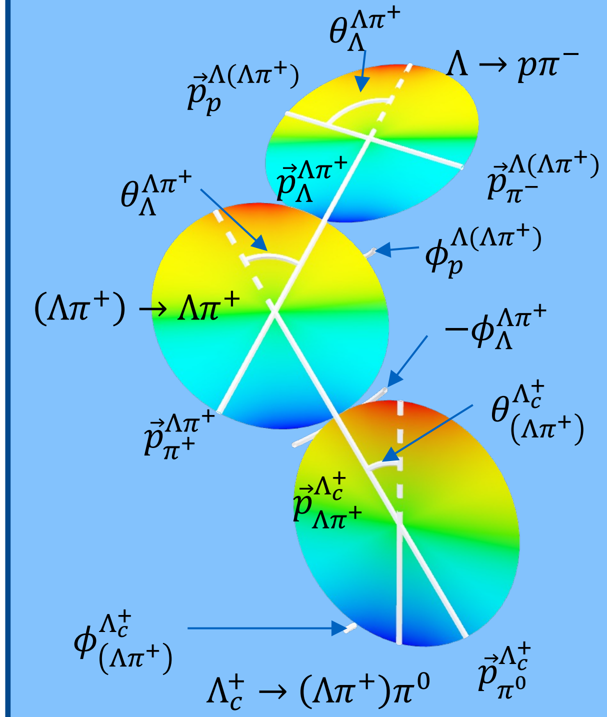
- Many amplitude analysis tools developed at BESIII, which produce many physics results
- An example of general propose tool TF-PWA
  - user friendly with simple configuration and automatic amplitude construction
  - GPU optimization, Automatic differentiation, Rich function support
- Many practices on machine learning at BESIII
- An example of GNN implementation in observation of $\Lambda_c^+ \to n e^+ \nu$
  - EMC shower discrimination between neutron and $\Lambda \to n\pi^0$
  - good understanding of systematics on GNN models taking advantage of clean control samples in e^+e^- experiments
- A baseline model ViC for anti-neutron reconstruction based on deep learning
  - showing a significant error reduction in incident position prediction compared to the conventional method
  - pioneering the implementation of incident momentum regression

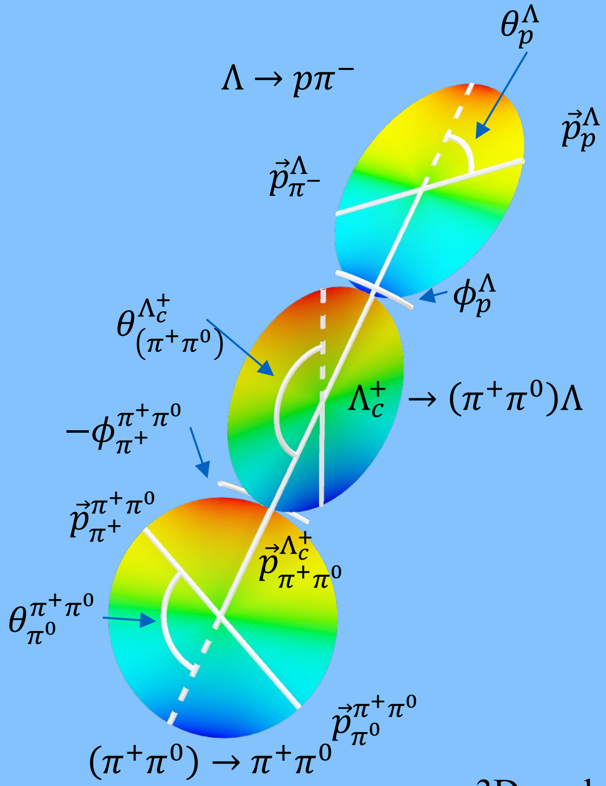
Thanks for your attentions!

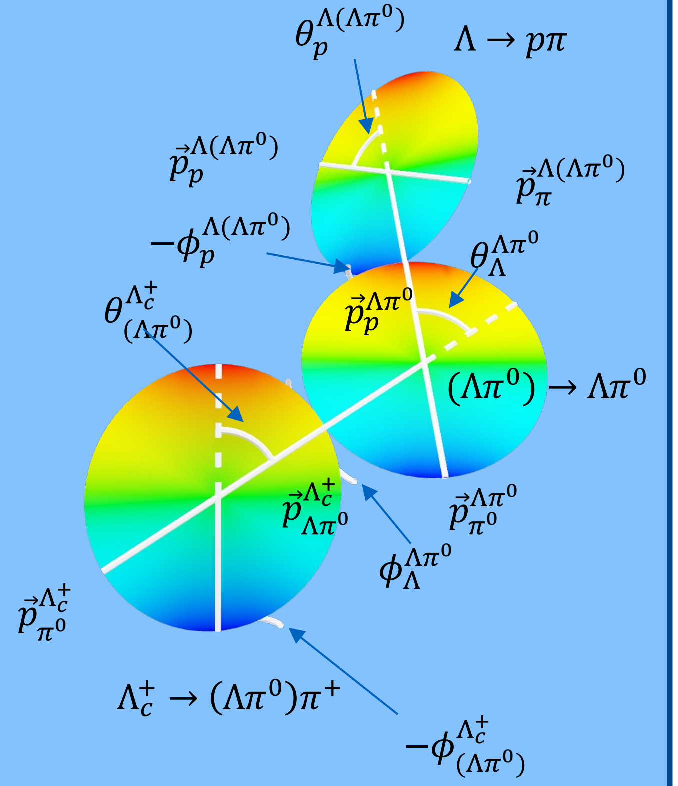
Backup

Automatic Angle Plot

$\vec{p}_B^{A}$ means momentum of B in the rest frame of A
ϕ means the rotation is anticlockwise, while $-\phi$ for clockwise
The sign Is dependent on data







Auto calculated by TF-PWA,
Only required the 4-monmenta

TF-PWA also provide reverse process:
Mass + helicity angle -> 4-monmenta

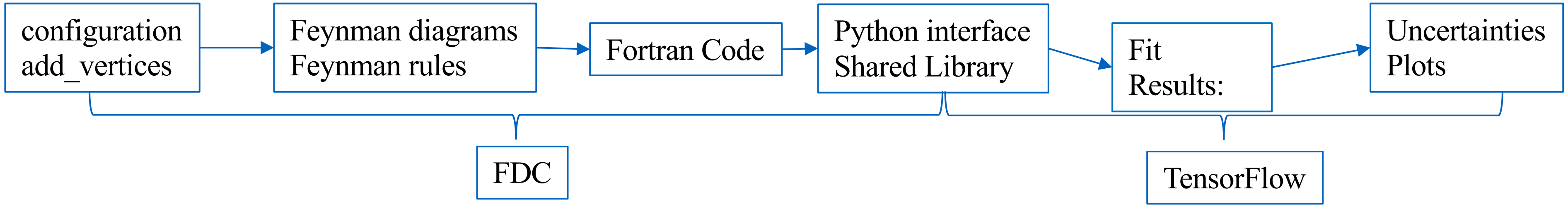
3D model generated by a script using TF-PWA.

FDC-PWA

www1.ihep.ac.cn/wjx/pwa/index.html

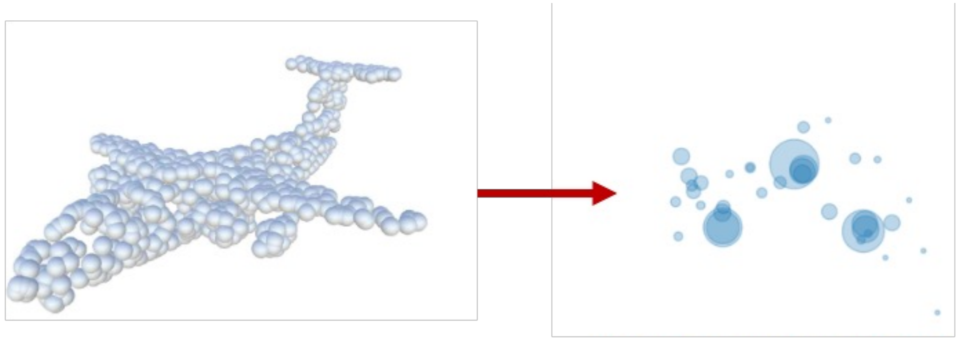
- FDC: Feynman Diagram Calculation
  - Construct the Lagrangian and deduce Feynman rules automatically
  - Generation of all Feynman diagrams and amplitudes for a given process.
- FDC-PWA:
  - Construct effective strong interaction model
  - Generate Fortran code to calculate Partial waves amplitudes
  - Fit for coupling parameters with TensorFlow on GPU (include AD)

Many steps, but automatic



Data representations

- As **Particle Cloud (Point Cloud)**
  - Unordered, permutations-invariant set of particles
  - Each particle carries spatial coordinates + additional features.
    - charge, momentum…. track & shower parameters, etc.
  - Symmetry-preserving, high expressiveness, low computational cost.



Point cloud of an aircraft generated by 3D scanning

Point cloud of a HEP event