ID: 90
yanqiyu17@mails.ucas.ac.cn

# Professor Based ReWeight for GENIE Generator

Qiyu Yan[1,2], Júlia Tena-Vidal[3], Marco Roda[4], Costas Andreopoulos[4] and Xianguo Lu[2] on behalf of GENIE Collaboration

[1]University of the Chinese Academy of Sciences, [2]University of Warwick, [3]Tel Aviv University, [4]University of Liverpool

## 1. Why We Need ReWeight?

- A Prediction is made of multiple models convoluted together:

Mass ordering, mixing angle, CPV.

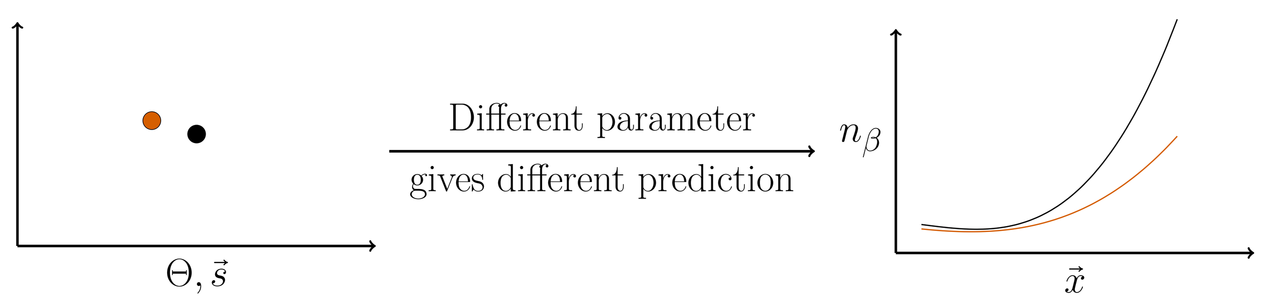
| Neutrino Flux Model | Oscillation Parameters | Interaction Model | Detector Model |
|---|---|---|---|
| $\Phi_\alpha(\vec{p_\nu};\vec{s_\Phi})$ | $P_{\alpha\beta}(\vec{p_\nu};\vec{\Theta})$ | $\sigma_\beta(E_\nu,\vec{x_0};\vec{s_\sigma})$ | $R(\vec{x_0},\vec{x};\vec{s_R})$ |

Interaction Generator Internal Models

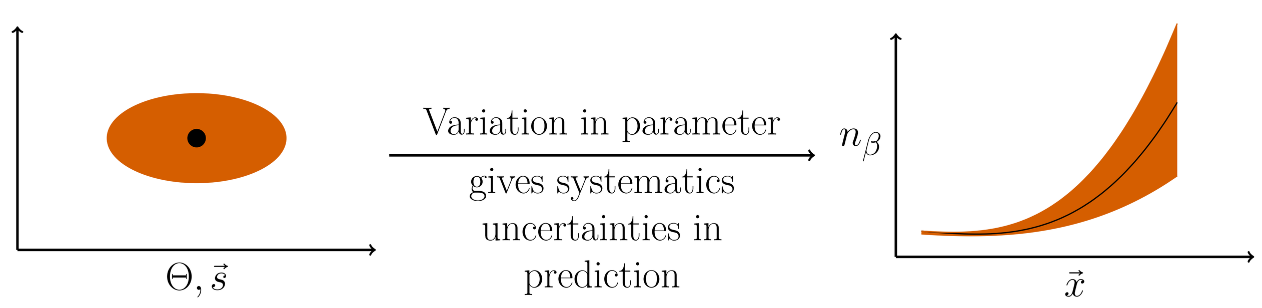
$$\sigma_\beta(E_\nu,\vec{x_0};\vec{s_\sigma}) = \sum\!\!\!\!\!\!\int_{c_i,\vec{y}} \mathcal{M}_1(E_\nu,\vec{x_0},c_i,\vec{y};\vec{s_\sigma})\cdots\mathcal{M}_n(E_\nu,\vec{x_0},c_i,\vec{y};\vec{s_\sigma})$$

$$n_\beta(\vec{x};\Theta,\vec{s}) = \sum\!\!\!\!\!\!\int_{\alpha,\vec{p_\nu},\vec{x_0}} \Phi_\alpha(\vec{p_\nu};\vec{s_\Phi})\,P_{\alpha\beta}(\vec{p_\nu};\vec{\Theta})\,\sigma_\beta(E_\nu,\vec{x_0};\vec{s_\sigma})\,R(\vec{x_0},\vec{x};\vec{s_R})$$

- It is near impossible to work analytically with convolution chain: **Monte Carlo** method is used to evaluate the prediction.
- Model prediction is parameter dependent




- Parameters comes with uncertainties, the uncertainties will be propagated to prediction




- **Being able to propagate uncertainties from parameters to prediction is crucial to understanding the systematics of the prediction.**
  CPU time and storage is expensive, we can't do full Monte Carlo for each parameter set: **ReWeight** is an approach to get model prediction for different parameter set using existing Monte Carlo samples.

## 2. The Mathematics Behind ReWeight

Mathematics tells us, Monte Carlo samples in "permitted phase space" $\lambda = (\vec{p_\nu},\vec{x_0}) \in \Omega$ to calculate convolution:

$$n_\beta(\vec{x};\Theta,\vec{s}) = \sum_\alpha \int \mathrm{d}\vec{p_\nu} \int \mathrm{d}\vec{x_0}\,\Phi_\alpha(\vec{p_\nu};\vec{s_\Phi})\,P_{\alpha\beta}(\vec{p_\nu};\vec{\Theta})\,\underbrace{\sigma_\beta(E_\nu,\vec{x_0};\vec{s_\sigma})}_{\int_{\vec{y}}\mathcal{M}_1(E_\nu,\vec{x_0},\vec{y})\cdots\mathcal{M}_n(E_\nu,\vec{x_0},\vec{y})}\,R(\vec{x_0},\vec{x};\vec{s_R})$$
$$\approx \frac{V}{N}\sum_{i=1}^{N} f(\lambda_i)$$

where $V = \int \mathrm{d}\vec{p_\nu} \int \mathrm{d}\vec{x_0}$ is the volume of the phase space, $N$ is the number of samples. $f$ is defined as

$$f = \frac{\Phi_\alpha(\vec{p_\nu};\vec{s_\Phi})\,P_{\alpha\beta}(\vec{p_\nu};\vec{\Theta})\,\sigma_\beta(E_\nu,\vec{x_0};\vec{s_\sigma})\,R(\vec{x_0},\vec{x};\vec{s_R})}{p(\lambda)}$$

$p(\lambda)$ is the probability density function we used to sample $\lambda$.

Usually, we start with "equal weight" sample (which is the usual case for GENIE [1] and NuWro):

$$p(\lambda) = k\cdot\Phi_\alpha(\vec{p_\nu};\vec{s_\Phi})\,P_{\alpha\beta}(\vec{p_\nu};\vec{\Theta})\,\sigma_\beta(E_\nu,\vec{x_0};\vec{s_\sigma})\,R(\vec{x_0},\vec{x};\vec{s_R})$$
$$f = k^{-1}$$

If samples are already generated for given parameter set $s$, then the prediction for an altered parameter set $s'$ can be obtained by reweighting the samples:

$$n_\beta(\vec{x};\Theta,\vec{s}') \approx \frac{V}{N}\sum_{i=1}^{N}\frac{\Phi_\alpha(\vec{p_\nu};\vec{s_\Phi}')\,P_{\alpha\beta}(\vec{p_\nu};\vec{\Theta}')\,\sigma_\beta(E_\nu,\vec{x_0};\vec{s_\sigma}')\,R(\vec{x_0},\vec{x};\vec{s_R}')}{k\cdot\Phi_\alpha(\vec{p_\nu};\vec{s_\Phi})\,P_{\alpha\beta}(\vec{p_\nu};\vec{\Theta})\,\sigma_\beta(E_\nu,\vec{x_0};\vec{s_\sigma})\,R(\vec{x_0},\vec{x};\vec{s_R})}$$
$$= \frac{V}{N}\sum_{i=1}^{N} f(\lambda_i;s)\,w(\lambda,s,s')$$

with additional factor $w(\lambda,s,s') = \frac{\Phi_\alpha(\vec{p_\nu};\vec{s_\Phi}')P_{\alpha\beta}(\vec{p_\nu};\vec{\Theta}')\sigma_\beta(E_\nu,\vec{x_0};\vec{s_\sigma}')R(\vec{x_0},\vec{x};\vec{s_R}')}{\Phi_\alpha(\vec{p_\nu};\vec{s_\Phi})P_{\alpha\beta}(\vec{p_\nu};\vec{\Theta})\sigma_\beta(E_\nu,\vec{x_0};\vec{s_\sigma})R(\vec{x_0},\vec{x};\vec{s_R})}$, that accounts for the sample $\{\lambda_i\}$ being sampled from $s$ and being reweighted to $s'$.

- But for interaction model, we need to rely on the internal model of the interaction generator, which is another convolution chain ($\mathcal{M}_i$)
  When only variation of $\vec{s_\sigma}$ is considered, the weight ideally should be

$$w(E_\nu,\vec{x_0};s,s') = \frac{\sigma_\beta(E_\nu,\vec{x_0};\vec{s_\sigma}')}{\sigma_\beta(E_\nu,\vec{x_0};\vec{s_\sigma})}$$

But the practical way is (since we can't get the cross section analytically):

$$w(E_\nu,\vec{x_0},\vec{y};s,s') = \frac{\mathcal{M}_1(E_\nu,\vec{x_0},\vec{y};\vec{s_\sigma}')\cdots\mathcal{M}_n(E_\nu,\vec{x_0},\vec{y};\vec{s_\sigma}')}{\mathcal{M}_1(E_\nu,\vec{x_0},\vec{y};\vec{s_\sigma})\cdots\mathcal{M}_n(E_\nu,\vec{x_0},\vec{y};\vec{s_\sigma})}$$

## 3. Limitations of Current Generator ReWeight

- **Weight calculation is highly dependent on internal models**:
  - Manpower needed to implement reweight code per each model;
  - Continuous maintaining effort needed to keep the reweight code up-to-date with the models.
- **"Internal phase space" variables $\vec{y}$ are needed to calculate the per-model weight**:
  - Need to save those information on event generation: hard or nearly-impossible for models like cascade and hadronization.
- **Can not reweight missing phase space into existence**:
  - Missing visible phase space ($\vec{x_0}$): hard cut on final state kinematics, this is intrinsically un-weightable.
  - **Missing internal phase space** ($\vec{y}$): this is a limitation of the traditional reweight.

## 4. Professor Based ReWeight

But we don't essentially need to get the weight with the internal models: Monte Carlo itself can give us information about $\sigma_\beta(E_\nu,\vec{x_0};\vec{s_\sigma}) = \frac{\mathrm{d}^n\sigma_\beta(E_\nu|\vec{s_\sigma})}{\mathrm{d}\vec{x}^n}$.

Considering:
- Auto-correlation of $\vec{x}$ (e.g. conservation law);
- Some varied parameter only affect a subset of models (e.g. changes in QEL $M_A$ only affect the $\frac{\mathrm{d}\sigma}{\mathrm{d}Q^2}$);

we could replace the differential cross section from the "final state" $\vec{x}$ to the "observable" $\vec{K}$:

$$\frac{\mathrm{d}^n\sigma_\beta(E_\nu\mid\vec{s_\sigma})}{\mathrm{d}\vec{x}^n} = \frac{\mathrm{d}^N\sigma_\beta(E_\nu\mid\vec{s_\sigma})}{\mathrm{d}\vec{K}^N}f(\vec{K};\vec{x})$$

where $f(\vec{K};\vec{x})$ contains the Jacobian of the transformation and the contribution from integrated out unrelated components of $\vec{x}$, which would be cancelled when we calculate the ratio to get the weight.

**With those in mind, we can propose a new reweight method based on parameterized differential cross section in terms of observable $\vec{K}$:**
- Scan different $\{s_{\sigma,i}\} \in S$, generate Monte Carlo for each $s_{\sigma,i}$ scanned;
- Process the generated sample, represent the estimation of $\frac{\mathrm{d}^N\sigma_\beta(E_\nu|\vec{s_\sigma})}{\mathrm{d}\vec{K}^N}$ by a $N+1$ dimensional histogram binned by $\vec{K}$ and $E_\nu$ for each sample;
- For each given $(\vec{K},E_\nu)$ bin $j$, consider the value of $\frac{\mathrm{d}^N\sigma_\beta(E_\nu|\vec{s_\sigma})}{\mathrm{d}\vec{K}^N}$ on different $s_{\sigma,i}$ as a function of $s_\sigma$, use polynomial interpolation to get polynomial function $P_j(s_\sigma)$;

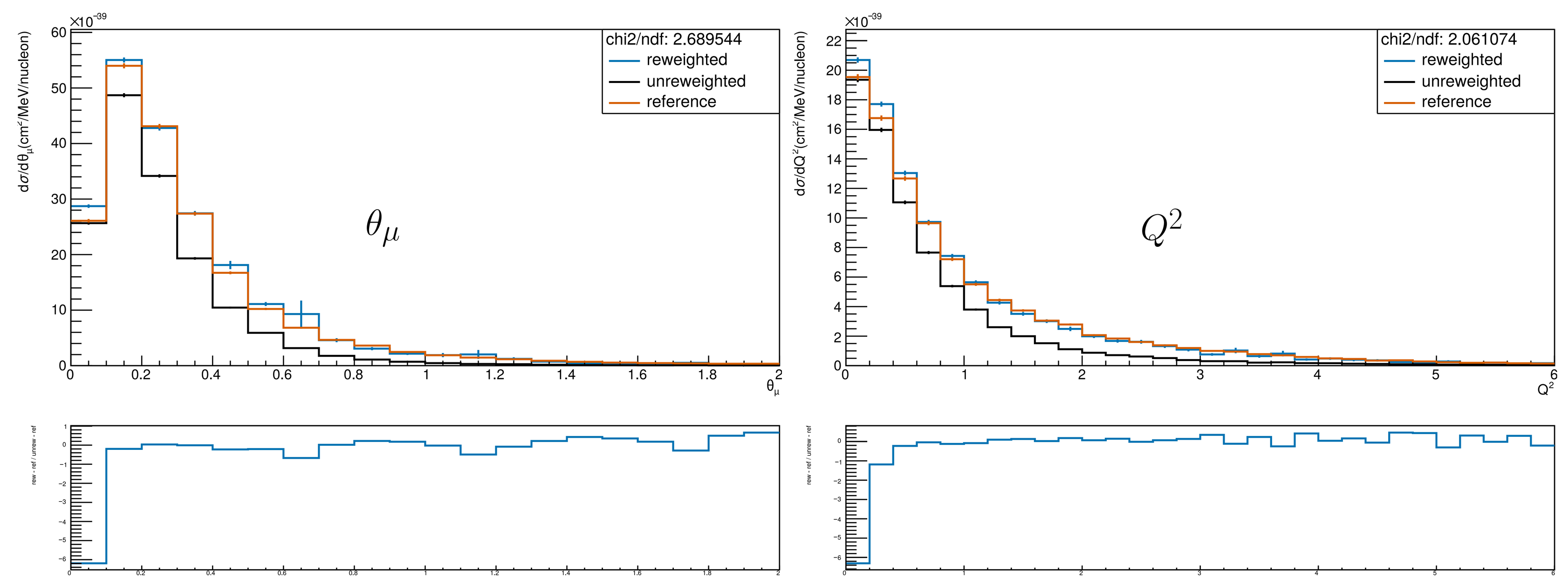
**Professor2 [3] is a tool to provide essential support to scan the parameter space and do the interpolation on the large database generated from the Monte Carlo.** Many tools in GENIE tunning [4] are reused to parameterize the differential cross section.
All $P_j(s_\sigma)$ together, make up a full parameterized differential cross section. When the ReWeight code handles an event:
- For each event, calculate the $\vec{K}$ and $E_\nu$, locate the corresponding bin $j$;
- Use the polynomial function $P_j(s_\sigma)$ to get the differential cross section for the event;
- Weight is calculated as the ratio of the differential cross section

$$w = \frac{P_j(s'_\sigma)}{P_j(s_\sigma)}$$

**In this way we can remove dependence on internal model details and the internal phase information of the ReWeight process.**

## 5. Simple Proof of Concept for New ReWeight

Starting with GENIE Tune G18_10a_02_11b, observable $\vec{K}$ is chosen as $(\vec{p_\mu},W,E_\nu)$.
ReWeight from the default ($M_A^{\mathrm{QE}} = 0.994989$, $M_A^{\mathrm{RES}} = 1.088962$) to $M_A^{\mathrm{QE}} = 0.77$, $M_A^{\mathrm{RES}} = 1.64$:




**The reweight can get the distribution of $\theta_\mu$ and $Q^2$ correct.**

## 6. Summary

- New Professor2 based reweight would extend the capability of current reweight tool.
  - **Less maintenance effort needed**
  - **More systematics will become reweightable**
- Currently under active development: planed to be **next major upgrade** for GENIE ReWeight.
- Cooperate with tunning efforts [2, 4], this will enable full propagation of errors from **experimental data to model** and **model to prediction**

[1] C. Andreopoulos et al. The GENIE Neutrino Monte Carlo Generator. *Nucl. Instrum. Meth.*, A614:87–104, 2010.
[2] Weijun Li, Marco Roda, Julia Tena-Vidal, Costas Andreopoulos, Xianguo Lu, Adi Ashkenazi, Joshua Barrow, Steven Dytman, Hugh Gallagher, Alfonso Andres Garcia Soto, et al. First combined tuning on transverse kinematic imbalance data with and without pion production constraints. *arXiv preprint arXiv:2404.08510*, 2024.
[3] VA Muradyan, NZ Akopov, HM Ghumaryan, and GA Karyan. "professor2" package for tuning the parameters of fragmentation process in $e^+e^-$ annihilation. *Journal of Contemporary Physics (Armenian Academy of Sciences)*, 56(2):85–90, 2021.
[4] Júlia Tena-Vidal et al. Neutrino-nucleon cross-section model tuning in GENIE v3. *Phys. Rev. D*, 104(7):072009, 2021.