Correction methods for detector effects on cumulants

Toshihiro Nonaka
University of Tsukuba
EMMI workshop @CCNU

筑波大学
University of Tsukuba

筑波大学
宇宙史研究センター
Tomonaga Center for the History of the Universe

EMMI

Outline

- ✓ **Efficiency correction**
- ✓ **Unfolding**
  - **Methodology**
  - **Results and Discussions**

More efficient formulas for efficiency correction and the importance of precise correction.

T. Nonaka, M. Kitazawa and S. Esumi
PRC.95.064912

Efficiency correction

✓ **Efficiency correction on cumulants has been established.**

- A. Bzdak and V. Koch : PRC.86.044904, PRC.91.027901, X. Luo : PRC.91.034907
- M. Kitazawa : PRC.86.024904, M. Kitazawa and M. Asakawa : PRC.86.024904

✓ **They are derived via simple relationship between true and measured factorial moments based on the binomial model.**

N : true particles, M : measured particles
ε : efficiency, F : true factorial moment
f : measured factorial moment, a(b): index for (anti)particle

✦ Factorial moment

$$f_{ab} = \sum_{M=a}^{\infty} \sum_{\bar{M}=b}^{\infty} p(M,\bar{M}) \frac{M!}{(M-a)!} \frac{\bar{M}!}{(\bar{M}-b)!},$$

$$F_{ab} = \sum_{N=a}^{\infty} \sum_{\bar{N}=b}^{\infty} P(N,\bar{N}) \frac{N!}{(N-a)!} \frac{\bar{N}!}{(\bar{N}-b)!}.$$

✦ Binomial distribution

$$B(M,N;\varepsilon) = \binom{N}{M} \varepsilon^{M} (1-\varepsilon)^{N-M}$$

$$f_{ab} = \varepsilon^{a} \bar{\varepsilon}^{b} F_{ab}$$

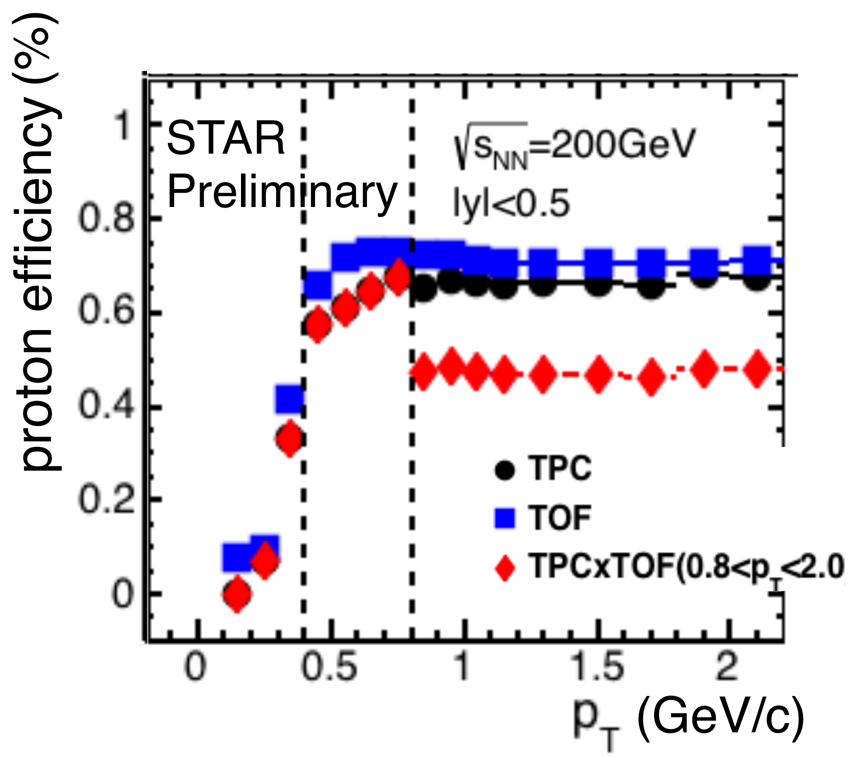
measured cumulant → measured factorial moment

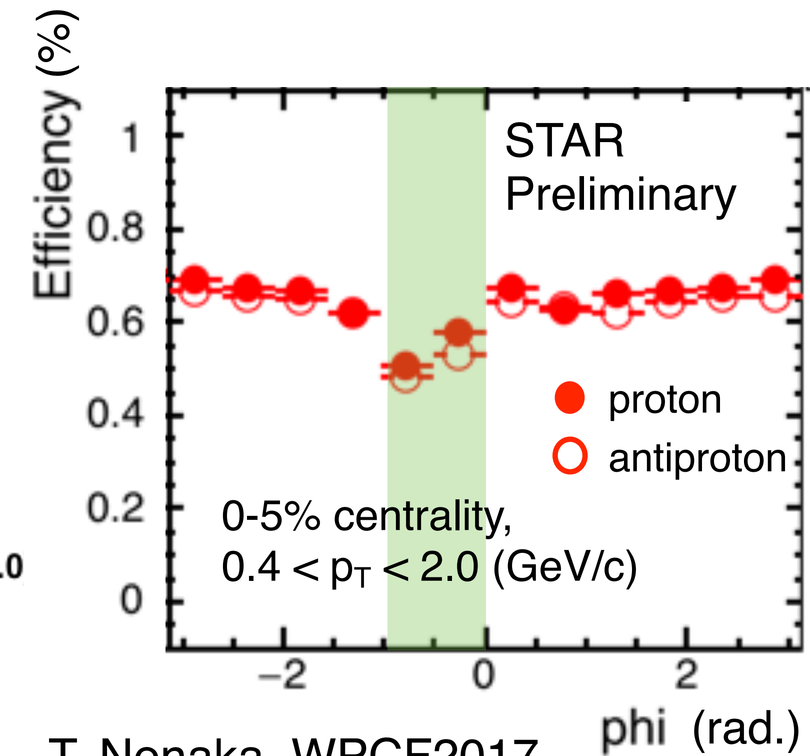
measured factorial moment → (just divide by efficiency) → corrected factorial moment

corrected factorial moment → corrected cumulants

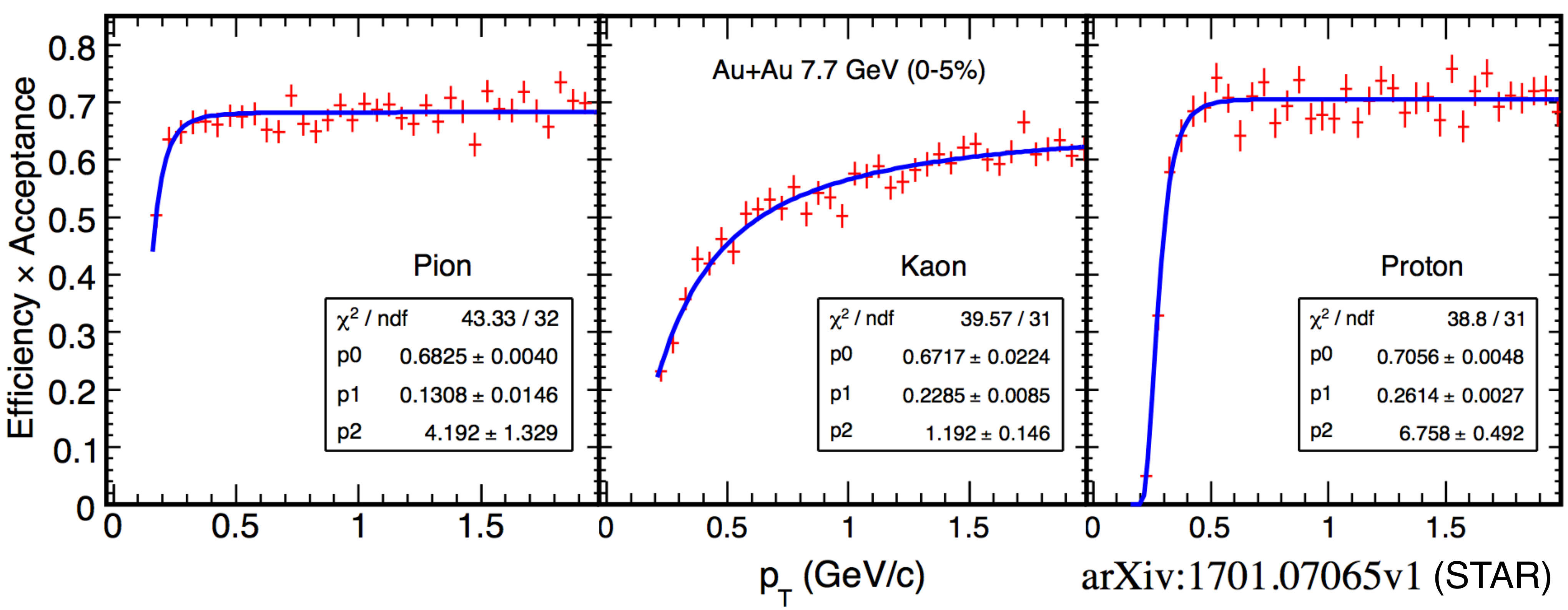
Efficiency bin

- ✓ **Experimentally,**
  - **efficiency will depend on p_T ,rapidity and azimuthal angle.**
  - **different particle species have different efficiencies.**
- ✓ **Such efficiency differences need to be implemented in the efficiency correction.**





T. Nonaka, WPCF2017



arXiv:1701.07065v1 (STAR)

In case of many efficiency bins…

1 eff. bin

$$
\begin{gathered}
\kappa_4(\Delta N) = (((f_{10}/\varepsilon_1) + 7(f_{20}/\varepsilon_1^2) + 6(f_{30}/\varepsilon_1^3) + (f_{40}/\varepsilon_1^4) - 4(f_{10}/\varepsilon_1)^2 - \\
12(f_{20}/\varepsilon_1^2)(f_{10}/\varepsilon_1) - 4(f_{30}/\varepsilon_1^3)(f_{10}/\varepsilon_1) + 6(f_{10}/\varepsilon_1)^3 + 6(f_{20}/\varepsilon_1^2)(f_{10}/\varepsilon_1)^2 - 3(f_{10}/\varepsilon_1)^4) - \\
4((f_{11}/\varepsilon_1/\varepsilon_2) - (f_{10}/\varepsilon_1)(f_{01}/\varepsilon_2) + 3(f_{21}/\varepsilon_1^2/\varepsilon_2) - 3(f_{20}/\varepsilon_1^2)(f_{01}/\varepsilon_2) + (f_{31}/\varepsilon_1^3/\varepsilon_2) - \\
(f_{30}/\varepsilon_1^3)(f_{01}/\varepsilon_2) - 3(f_{11}/\varepsilon_1/\varepsilon_2)(f_{10}/\varepsilon_1) + 3(f_{10}/\varepsilon_1)^2(f_{01}/\varepsilon_2) - 3(f_{21}/\varepsilon_1^2/\varepsilon_2)(f_{10}/\varepsilon_1) + \\
3(f_{20}/\varepsilon_1^2)(f_{10}/\varepsilon_1)(f_{01}/\varepsilon_2) + 3(f_{11}/\varepsilon_1/\varepsilon_2)(f_{10}/\varepsilon_1)^2 - 3(f_{10}/\varepsilon_1)^3(f_{01}/\varepsilon_2)) + 6((f_{11}/\varepsilon_1/\varepsilon_2) + \\
(f_{12}/\varepsilon_1/\varepsilon_2^2) - 2(f_{11}/\varepsilon_1/\varepsilon_2)(f_{01}/\varepsilon_2) + (f_{10}/\varepsilon_1)(f_{01}/\varepsilon_2)^2 + (f_{21}/\varepsilon_1^2/\varepsilon_2) + (f_{22}/\varepsilon_1^2/\varepsilon_2^2) - \\
2(f_{21}/\varepsilon_1^2/\varepsilon_2)(f_{01}/\varepsilon_2) + (f_{20}/\varepsilon_1^2)(f_{01}/\varepsilon_2)^2 - 2(f_{11}/\varepsilon_1/\varepsilon_2)(f_{10}/\varepsilon_1) - 2(f_{12}/\varepsilon_1/\varepsilon_2^2)(f_{10}/\varepsilon_1) + \\
4(f_{11}/\varepsilon_1/\varepsilon_2)(f_{10}/\varepsilon_1)(f_{01}/\varepsilon_2) - 3(f_{10}/\varepsilon_1)^2(f_{01}/\varepsilon_2)^2 + (f_{10}/\varepsilon_1)^2(f_{01}/\varepsilon_2) + (f_{02}/\varepsilon_2^2)(f_{10}/\varepsilon_1)^2) - \\
4((f_{11}/\varepsilon_1/\varepsilon_2) + 3(f_{12}/\varepsilon_1/\varepsilon_2^2) + (f_{13}/\varepsilon_1/\varepsilon_2^3) - 3(f_{11}/\varepsilon_1/\varepsilon_2)(f_{01}/\varepsilon_2) - 3(f_{12}/\varepsilon_1/\varepsilon_2^2)(f_{01}/\varepsilon_2) + \\
3(f_{11}/\varepsilon_1/\varepsilon_2)(f_{01}/\varepsilon_2)^2 - 3(f_{10}/\varepsilon_1)(f_{01}/\varepsilon_2)^3 - (f_{10}/\varepsilon_1)(f_{01}/\varepsilon_2) - 3(f_{02}/\varepsilon_2^2)(f_{10}/\varepsilon_1) - \\
(f_{03}/\varepsilon_2^3)(f_{10}/\varepsilon_1) + 3(f_{10}/\varepsilon_1)(f_{01}/\varepsilon_2)^2 + 3(f_{02}/\varepsilon_2^2)(f_{10}/\varepsilon_1)(f_{01}/\varepsilon_2)) + ((f_{01}/\varepsilon_2) + \\
7(f_{02}/\varepsilon_2^2) + 6(f_{03}/\varepsilon_2^3) + (f_{04}/\varepsilon_2^4) - 4(f_{01}/\varepsilon_2)^2 - 12(f_{02}/\varepsilon_2^2)(f_{01}/\varepsilon_2) - 4(f_{03}/\varepsilon_2^3)(f_{01}/\varepsilon_2) + \\
6(f_{01}/\varepsilon_2)^3 + 6(f_{02}/\varepsilon_2^2)(f_{01}/\varepsilon_2)^2 - 3(f_{01}/\varepsilon_2)^4)) - 3(((f_{10}/\varepsilon_1) + (f_{20}/\varepsilon_1^2) - (f_{10}/\varepsilon_1)^2) - \\
2((f_{11}/\varepsilon_1/\varepsilon_2) - (f_{10}/\varepsilon_1)(f_{01}/\varepsilon_2)) + ((f_{01}/\varepsilon_2) + (f_{02}/\varepsilon_2^2) - (f_{01}/\varepsilon_2)^2))^2
\end{gathered}
$$

- ✓ **Number of terms drastically increases!**
- ✓ **Although it can be automated, calculations don't finish…**

Number of factorial moments

m : order of cumulant
M : # of efficiency bins

$$
N_m^{\mathrm{fm}} = \sum_{r=1}^{m} {}_{r+M-1}C_r = {}_{m+M}C_m - 1
$$

$$
\sim M^m \text{ for large } M
$$

2 eff. bins : 412 terms

$$
\begin{gathered}
\kappa_4(\Delta N) = (((f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2) + 7(f_{2000}/\varepsilon_1^2 + f_{1100}/\varepsilon_1/\varepsilon_2 + f_{1100}/\varepsilon_1/\varepsilon_2 + f_{0200}/\varepsilon_2^2) + 6(f_{3000}/\varepsilon_1^3 + f_{2100}/\varepsilon_1^2/\varepsilon_2 + \\
f_{2100}/\varepsilon_1^2/\varepsilon_2 + f_{1200}/\varepsilon_1/\varepsilon_2^2 + f_{2100}/\varepsilon_1^2/\varepsilon_2 + f_{1200}/\varepsilon_1/\varepsilon_2^2 + f_{1200}/\varepsilon_1/\varepsilon_2^2 + f_{0300}/\varepsilon_2^3) + (f_{4000}/\varepsilon_1^4 + f_{3100}/\varepsilon_1^3/\varepsilon_2 + f_{3100}/\varepsilon_1^3/\varepsilon_2 + \\
f_{2200}/\varepsilon_1^2/\varepsilon_2^2 + f_{3100}/\varepsilon_1^3/\varepsilon_2 + f_{2200}/\varepsilon_1^2/\varepsilon_2^2 + f_{2200}/\varepsilon_1^2/\varepsilon_2^2 + f_{1300}/\varepsilon_1/\varepsilon_2^3 + f_{3100}/\varepsilon_1^3/\varepsilon_2 + f_{2200}/\varepsilon_1^2/\varepsilon_2^2 + f_{2200}/\varepsilon_1^2/\varepsilon_2^2 + f_{1300}/\varepsilon_1/\varepsilon_2^3 + \\
f_{2200}/\varepsilon_1^2/\varepsilon_2^2 + f_{1300}/\varepsilon_1/\varepsilon_2^3 + f_{1300}/\varepsilon_1/\varepsilon_2^3 + f_{0400}/\varepsilon_2^4) - 4(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)^2 - 12(f_{2000}/\varepsilon_1^2 + f_{1100}/\varepsilon_1/\varepsilon_2 + f_{1100}/\varepsilon_1/\varepsilon_2 + \\
f_{0200}/\varepsilon_2^2)(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2) - 4(f_{3000}/\varepsilon_1^3 + f_{2100}/\varepsilon_1^2/\varepsilon_2 + f_{2100}/\varepsilon_1^2/\varepsilon_2 + f_{1200}/\varepsilon_1/\varepsilon_2^2 + f_{2100}/\varepsilon_1^2/\varepsilon_2 + f_{1200}/\varepsilon_1/\varepsilon_2^2 + \\
f_{1200}/\varepsilon_1/\varepsilon_2^2 + f_{0300}/\varepsilon_2^3)(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2) + 6(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)^3 + 6(f_{2000}/\varepsilon_1^2 + f_{1100}/\varepsilon_1/\varepsilon_2 + f_{1100}/\varepsilon_1/\varepsilon_2 + f_{0200}/\varepsilon_2^2)(f_{1000}/\varepsilon_1 + \\
f_{0100}/\varepsilon_2)^2 - 3(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)^4) - 4((f_{1010}/\varepsilon_1/\varepsilon_3 + f_{1001}/\varepsilon_1/\varepsilon_4 + f_{0110}/\varepsilon_2/\varepsilon_3 + f_{0101}/\varepsilon_2/\varepsilon_4) - (f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)(f_{0010}/\varepsilon_3 + \\
f_{0001}/\varepsilon_4) + 3(f_{2010}/\varepsilon_1^2/\varepsilon_3 + f_{2001}/\varepsilon_1^2/\varepsilon_4 + f_{1110}/\varepsilon_1/\varepsilon_2/\varepsilon_3 + f_{1101}/\varepsilon_1/\varepsilon_2/\varepsilon_4 + f_{1110}/\varepsilon_1/\varepsilon_2/\varepsilon_3 + f_{1101}/\varepsilon_1/\varepsilon_2/\varepsilon_4 + f_{0210}/\varepsilon_2^2/\varepsilon_3 + \\
f_{0201}/\varepsilon_2^2/\varepsilon_4) - 3(f_{2000}/\varepsilon_1^2 + f_{1100}/\varepsilon_1/\varepsilon_2 + f_{1100}/\varepsilon_1/\varepsilon_2 + f_{0200}/\varepsilon_2^2)(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4) + (f_{3010}/\varepsilon_1^3/\varepsilon_3 + f_{3001}/\varepsilon_1^3/\varepsilon_4 + \\
f_{2110}/\varepsilon_1^2/\varepsilon_2/\varepsilon_3 + f_{2101}/\varepsilon_1^2/\varepsilon_2/\varepsilon_4 + f_{2110}/\varepsilon_1^2/\varepsilon_2/\varepsilon_3 + f_{2101}/\varepsilon_1^2/\varepsilon_2/\varepsilon_4 + f_{1210}/\varepsilon_1/\varepsilon_2^2/\varepsilon_3 + f_{1201}/\varepsilon_1/\varepsilon_2^2/\varepsilon_4 + f_{2110}/\varepsilon_1^2/\varepsilon_2/\varepsilon_3 + \\
f_{2101}/\varepsilon_1^2/\varepsilon_2/\varepsilon_4 + f_{1210}/\varepsilon_1/\varepsilon_2^2/\varepsilon_3 + f_{1201}/\varepsilon_1/\varepsilon_2^2/\varepsilon_4 + f_{1210}/\varepsilon_1/\varepsilon_2^2/\varepsilon_3 + f_{1201}/\varepsilon_1/\varepsilon_2^2/\varepsilon_4 + f_{0310}/\varepsilon_2^3/\varepsilon_3 + f_{0301}/\varepsilon_2^3/\varepsilon_4) - \\
(f_{3000}/\varepsilon_1^3 + f_{2100}/\varepsilon_1^2/\varepsilon_2 + f_{2100}/\varepsilon_1^2/\varepsilon_2 + f_{1200}/\varepsilon_1/\varepsilon_2^2 + f_{2100}/\varepsilon_1^2/\varepsilon_2 + f_{1200}/\varepsilon_1/\varepsilon_2^2 + f_{1200}/\varepsilon_1/\varepsilon_2^2 + f_{0300}/\varepsilon_2^3)(f_{0010}/\varepsilon_3 + \\
f_{0001}/\varepsilon_4) - 3(f_{1010}/\varepsilon_1/\varepsilon_3 + f_{1001}/\varepsilon_1/\varepsilon_4 + f_{0110}/\varepsilon_2/\varepsilon_3 + f_{0101}/\varepsilon_2/\varepsilon_4)(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2) + 3(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)^2(f_{0010}/\varepsilon_3 + \\
f_{0001}/\varepsilon_4) - 3(f_{2010}/\varepsilon_1^2/\varepsilon_3 + f_{2001}/\varepsilon_1^2/\varepsilon_4 + f_{1110}/\varepsilon_1/\varepsilon_2/\varepsilon_3 + f_{1101}/\varepsilon_1/\varepsilon_2/\varepsilon_4 + f_{1110}/\varepsilon_1/\varepsilon_2/\varepsilon_3 + f_{1101}/\varepsilon_1/\varepsilon_2/\varepsilon_4 + f_{0210}/\varepsilon_2^2/\varepsilon_3 + \\
f_{0201}/\varepsilon_2^2/\varepsilon_4)(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2) + 3(f_{2000}/\varepsilon_1^2 + f_{1100}/\varepsilon_1/\varepsilon_2 + f_{1100}/\varepsilon_1/\varepsilon_2 + f_{0200}/\varepsilon_2^2)(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)(f_{0010}/\varepsilon_3 + \\
f_{0001}/\varepsilon_4) + 3(f_{1010}/\varepsilon_1/\varepsilon_3 + f_{1001}/\varepsilon_1/\varepsilon_4 + f_{0110}/\varepsilon_2/\varepsilon_3 + f_{0101}/\varepsilon_2/\varepsilon_4)(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)^2 - 3(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)^3(f_{0010}/\varepsilon_3 + \\
f_{0001}/\varepsilon_4)) + 6((f_{1010}/\varepsilon_1/\varepsilon_3 + f_{1001}/\varepsilon_1/\varepsilon_4 + f_{0110}/\varepsilon_2/\varepsilon_3 + f_{0101}/\varepsilon_2/\varepsilon_4) + (f_{1020}/\varepsilon_1/\varepsilon_3^2 + f_{1011}/\varepsilon_1/\varepsilon_3/\varepsilon_4 + f_{1011}/\varepsilon_1/\varepsilon_3/\varepsilon_4 + \\
f_{1002}/\varepsilon_1/\varepsilon_4^2 + f_{0120}/\varepsilon_2/\varepsilon_3^2 + f_{0111}/\varepsilon_2/\varepsilon_3/\varepsilon_4 + f_{0111}/\varepsilon_2/\varepsilon_3/\varepsilon_4 + f_{0102}/\varepsilon_2/\varepsilon_4^2) - 2(f_{1010}/\varepsilon_1/\varepsilon_3 + f_{1001}/\varepsilon_1/\varepsilon_4 + f_{0110}/\varepsilon_2/\varepsilon_3 + \\
f_{0101}/\varepsilon_2/\varepsilon_4)(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4) + (f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4)^2 + (f_{2010}/\varepsilon_1^2/\varepsilon_3 + f_{2001}/\varepsilon_1^2/\varepsilon_4 + f_{1110}/\varepsilon_1/\varepsilon_2/\varepsilon_3 + \\
f_{1101}/\varepsilon_1/\varepsilon_2/\varepsilon_4 + f_{1110}/\varepsilon_1/\varepsilon_2/\varepsilon_3 + f_{1101}/\varepsilon_1/\varepsilon_2/\varepsilon_4 + f_{0210}/\varepsilon_2^2/\varepsilon_3 + f_{0201}/\varepsilon_2^2/\varepsilon_4) + (f_{2020}/\varepsilon_1^2/\varepsilon_3^2 + f_{2011}/\varepsilon_1^2/\varepsilon_3/\varepsilon_4 + f_{2011}/\varepsilon_1^2/\varepsilon_3/\varepsilon_4 + \\
f_{2002}/\varepsilon_1^2/\varepsilon_4^2 + f_{1120}/\varepsilon_1/\varepsilon_2/\varepsilon_3^2 + f_{1111}/\varepsilon_1/\varepsilon_2/\varepsilon_3/\varepsilon_4 + f_{1111}/\varepsilon_1/\varepsilon_2/\varepsilon_3/\varepsilon_4 + f_{1102}/\varepsilon_1/\varepsilon_2/\varepsilon_4^2 + f_{1120}/\varepsilon_1/\varepsilon_2/\varepsilon_3^2 + f_{1111}/\varepsilon_1/\varepsilon_2/\varepsilon_3/\varepsilon_4 + \\
f_{1111}/\varepsilon_1/\varepsilon_2/\varepsilon_3/\varepsilon_4 + f_{1102}/\varepsilon_1/\varepsilon_2/\varepsilon_4^2 + f_{0220}/\varepsilon_2^2/\varepsilon_3^2 + f_{0211}/\varepsilon_2^2/\varepsilon_3/\varepsilon_4 + f_{0211}/\varepsilon_2^2/\varepsilon_3/\varepsilon_4 + f_{0202}/\varepsilon_2^2/\varepsilon_4^2) - 2(f_{2010}/\varepsilon_1^2/\varepsilon_3 + \\
f_{2001}/\varepsilon_1^2/\varepsilon_4 + f_{1110}/\varepsilon_1/\varepsilon_2/\varepsilon_3 + f_{1101}/\varepsilon_1/\varepsilon_2/\varepsilon_4 + f_{1110}/\varepsilon_1/\varepsilon_2/\varepsilon_3 + f_{1101}/\varepsilon_1/\varepsilon_2/\varepsilon_4 + f_{0210}/\varepsilon_2^2/\varepsilon_3 + f_{0201}/\varepsilon_2^2/\varepsilon_4)(f_{0010}/\varepsilon_3 + \\
f_{0001}/\varepsilon_4) + (f_{2000}/\varepsilon_1^2 + f_{1100}/\varepsilon_1/\varepsilon_2 + f_{1100}/\varepsilon_1/\varepsilon_2 + f_{0200}/\varepsilon_2^2)(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4)^2 - 2(f_{1010}/\varepsilon_1/\varepsilon_3 + f_{1001}/\varepsilon_1/\varepsilon_4 + \\
f_{0110}/\varepsilon_2/\varepsilon_3 + f_{0101}/\varepsilon_2/\varepsilon_4)(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2) - 2(f_{1020}/\varepsilon_1/\varepsilon_3^2 + f_{1011}/\varepsilon_1/\varepsilon_3/\varepsilon_4 + f_{1011}/\varepsilon_1/\varepsilon_3/\varepsilon_4 + f_{1002}/\varepsilon_1/\varepsilon_4^2 + f_{0120}/\varepsilon_2/\varepsilon_3^2 + \\
f_{0111}/\varepsilon_2/\varepsilon_3/\varepsilon_4 + f_{0111}/\varepsilon_2/\varepsilon_3/\varepsilon_4 + f_{0102}/\varepsilon_2/\varepsilon_4^2)(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2) + 4(f_{1010}/\varepsilon_1/\varepsilon_3 + f_{1001}/\varepsilon_1/\varepsilon_4 + f_{0110}/\varepsilon_2/\varepsilon_3 + f_{0101}/\varepsilon_2/\varepsilon_4)(f_{1000}/\varepsilon_1 + \\
f_{0100}/\varepsilon_2)(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4) - 3(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)^2(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4)^2 + (f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)^2(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4) + \\
(f_{0020}/\varepsilon_3^2 + f_{0011}/\varepsilon_3/\varepsilon_4 + f_{0011}/\varepsilon_3/\varepsilon_4 + f_{0002}/\varepsilon_4^2)(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)^2) - 4((f_{1010}/\varepsilon_1/\varepsilon_3 + f_{1001}/\varepsilon_1/\varepsilon_4 + f_{0110}/\varepsilon_2/\varepsilon_3 + \\
f_{0101}/\varepsilon_2/\varepsilon_4) + 3(f_{1020}/\varepsilon_1/\varepsilon_3^2 + f_{1011}/\varepsilon_1/\varepsilon_3/\varepsilon_4 + f_{1011}/\varepsilon_1/\varepsilon_3/\varepsilon_4 + f_{1002}/\varepsilon_1/\varepsilon_4^2 + f_{0120}/\varepsilon_2/\varepsilon_3^2 + f_{0111}/\varepsilon_2/\varepsilon_3/\varepsilon_4 + f_{0111}/\varepsilon_2/\varepsilon_3/\varepsilon_4 + \\
f_{0102}/\varepsilon_2/\varepsilon_4^2) + (f_{1030}/\varepsilon_1/\varepsilon_3^3 + f_{1021}/\varepsilon_1/\varepsilon_3^2/\varepsilon_4 + f_{1021}/\varepsilon_1/\varepsilon_3^2/\varepsilon_4 + f_{1012}/\varepsilon_1/\varepsilon_3/\varepsilon_4^2 + f_{1021}/\varepsilon_1/\varepsilon_3^2/\varepsilon_4 + f_{1012}/\varepsilon_1/\varepsilon_3/\varepsilon_4^2 + \\
f_{1012}/\varepsilon_1/\varepsilon_3/\varepsilon_4^2 + f_{1003}/\varepsilon_1/\varepsilon_4^3 + f_{0130}/\varepsilon_2/\varepsilon_3^3 + f_{0121}/\varepsilon_2/\varepsilon_3^2/\varepsilon_4 + f_{0121}/\varepsilon_2/\varepsilon_3^2/\varepsilon_4 + f_{0112}/\varepsilon_2/\varepsilon_3/\varepsilon_4^2 + f_{0121}/\varepsilon_2/\varepsilon_3^2/\varepsilon_4 + f_{0112}/\varepsilon_2/\varepsilon_3/\varepsilon_4^2 + \\
f_{0112}/\varepsilon_2/\varepsilon_3/\varepsilon_4^2 + f_{0103}/\varepsilon_2/\varepsilon_4^3) - 3(f_{1010}/\varepsilon_1/\varepsilon_3 + f_{1001}/\varepsilon_1/\varepsilon_4 + f_{0110}/\varepsilon_2/\varepsilon_3 + f_{0101}/\varepsilon_2/\varepsilon_4)(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4) - 3(f_{1020}/\varepsilon_1/\varepsilon_3^2 + \\
f_{1011}/\varepsilon_1/\varepsilon_3/\varepsilon_4 + f_{1011}/\varepsilon_1/\varepsilon_3/\varepsilon_4 + f_{1002}/\varepsilon_1/\varepsilon_4^2 + f_{0120}/\varepsilon_2/\varepsilon_3^2 + f_{0111}/\varepsilon_2/\varepsilon_3/\varepsilon_4 + f_{0111}/\varepsilon_2/\varepsilon_3/\varepsilon_4 + f_{0102}/\varepsilon_2/\varepsilon_4^2)(f_{0010}/\varepsilon_3 + \\
f_{0001}/\varepsilon_4) + 3(f_{1010}/\varepsilon_1/\varepsilon_3 + f_{1001}/\varepsilon_1/\varepsilon_4 + f_{0110}/\varepsilon_2/\varepsilon_3 + f_{0101}/\varepsilon_2/\varepsilon_4)(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4)^2 - 3(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)(f_{0010}/\varepsilon_3 + \\
f_{0001}/\varepsilon_4)^3 - (f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4) - 3(f_{0020}/\varepsilon_3^2 + f_{0011}/\varepsilon_3/\varepsilon_4 + f_{0011}/\varepsilon_3/\varepsilon_4 + f_{0002}/\varepsilon_4^2)(f_{1000}/\varepsilon_1 + \\
f_{0100}/\varepsilon_2) - (f_{0030}/\varepsilon_3^3 + f_{0021}/\varepsilon_3^2/\varepsilon_4 + f_{0021}/\varepsilon_3^2/\varepsilon_4 + f_{0012}/\varepsilon_3/\varepsilon_4^2 + f_{0021}/\varepsilon_3^2/\varepsilon_4 + f_{0012}/\varepsilon_3/\varepsilon_4^2 + f_{0012}/\varepsilon_3/\varepsilon_4^2 + f_{0003}/\varepsilon_4^3)(f_{1000}/\varepsilon_1 + \\
f_{0100}/\varepsilon_2) + 3(f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4)^2 + 3(f_{0020}/\varepsilon_3^2 + f_{0011}/\varepsilon_3/\varepsilon_4 + f_{0011}/\varepsilon_3/\varepsilon_4 + f_{0002}/\varepsilon_4^2)(f_{1000}/\varepsilon_1 + \\
f_{0100}/\varepsilon_2)(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4)) + ((f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4) + 7(f_{0020}/\varepsilon_3^2 + f_{0011}/\varepsilon_3/\varepsilon_4 + f_{0011}/\varepsilon_3/\varepsilon_4 + f_{0002}/\varepsilon_4^2) + 6(f_{0030}/\varepsilon_3^3 + \\
f_{0021}/\varepsilon_3^2/\varepsilon_4 + f_{0021}/\varepsilon_3^2/\varepsilon_4 + f_{0012}/\varepsilon_3/\varepsilon_4^2 + f_{0021}/\varepsilon_3^2/\varepsilon_4 + f_{0012}/\varepsilon_3/\varepsilon_4^2 + f_{0012}/\varepsilon_3/\varepsilon_4^2 + f_{0003}/\varepsilon_4^3) + (f_{0040}/\varepsilon_3^4 + f_{0031}/\varepsilon_3^3/\varepsilon_4 + \\
f_{0031}/\varepsilon_3^3/\varepsilon_4 + f_{0022}/\varepsilon_3^2/\varepsilon_4^2 + f_{0031}/\varepsilon_3^3/\varepsilon_4 + f_{0022}/\varepsilon_3^2/\varepsilon_4^2 + f_{0022}/\varepsilon_3^2/\varepsilon_4^2 + f_{0013}/\varepsilon_3/\varepsilon_4^3 + f_{0031}/\varepsilon_3^3/\varepsilon_4 + f_{0022}/\varepsilon_3^2/\varepsilon_4^2 + f_{0022}/\varepsilon_3^2/\varepsilon_4^2 + \\
f_{0013}/\varepsilon_3/\varepsilon_4^3 + f_{0022}/\varepsilon_3^2/\varepsilon_4^2 + f_{0013}/\varepsilon_3/\varepsilon_4^3 + f_{0013}/\varepsilon_3/\varepsilon_4^3 + f_{0004}/\varepsilon_4^4) - 4(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4)^2 - 12(f_{0020}/\varepsilon_3^2 + f_{0011}/\varepsilon_3/\varepsilon_4 + \\
f_{0011}/\varepsilon_3/\varepsilon_4 + f_{0002}/\varepsilon_4^2)(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4) - 4(f_{0030}/\varepsilon_3^3 + f_{0021}/\varepsilon_3^2/\varepsilon_4 + f_{0021}/\varepsilon_3^2/\varepsilon_4 + f_{0012}/\varepsilon_3/\varepsilon_4^2 + f_{0021}/\varepsilon_3^2/\varepsilon_4 + \\
f_{0012}/\varepsilon_3/\varepsilon_4^2 + f_{0012}/\varepsilon_3/\varepsilon_4^2 + f_{0003}/\varepsilon_4^3)(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4) + 6(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4)^3 + 6(f_{0020}/\varepsilon_3^2 + f_{0011}/\varepsilon_3/\varepsilon_4 + f_{0011}/\varepsilon_3/\varepsilon_4 + \\
f_{0002}/\varepsilon_4^2)(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4)^2 - 3(f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4)^4)) - 3(((f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2) + (f_{2000}/\varepsilon_1^2 + f_{1100}/\varepsilon_1/\varepsilon_2 + f_{1100}/\varepsilon_1/\varepsilon_2 + \\
f_{0200}/\varepsilon_2^2) - (f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)^2) - 2((f_{1010}/\varepsilon_1/\varepsilon_3 + f_{1001}/\varepsilon_1/\varepsilon_4 + f_{0110}/\varepsilon_2/\varepsilon_3 + f_{0101}/\varepsilon_2/\varepsilon_4) - (f_{1000}/\varepsilon_1 + f_{0100}/\varepsilon_2)(f_{0010}/\varepsilon_3 + \\
f_{0001}/\varepsilon_4)) + ((f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4) + (f_{0020}/\varepsilon_3^2 + f_{0011}/\varepsilon_3/\varepsilon_4 + f_{0011}/\varepsilon_3/\varepsilon_4 + f_{0002}/\varepsilon_4^2) - (f_{0010}/\varepsilon_3 + f_{0001}/\varepsilon_4)^2))^2
\end{gathered}
$$

P. Tribedy

3 eff. bins : 1188 terms

More efficient formulas

- ✓ Derivation using factorial cumulants.
- ✓ For more details, see PRC.95.064912.

measured cumulant → measured **factorial cumulant**

↓ just divide by efficiency

corrected cumulants ← corrected **factorial cumulant**

$$q_{(r,s)} = q_{(a^r/p^s)} = \sum_{i=1}^{M} (a_i^r/p_i^s) n_i$$

M : # of efficiency bins
n : # of particles
p : efficiency
a : electric charge

$$\langle Q\rangle_c = \langle q_{(1,1)}\rangle_c, \tag{62}$$

$$\langle Q^2\rangle_c = \langle q_{(1,1)}^2\rangle_c + \langle q_{(2,1)}\rangle_c - \langle q_{(2,2)}\rangle_c, \tag{63}$$

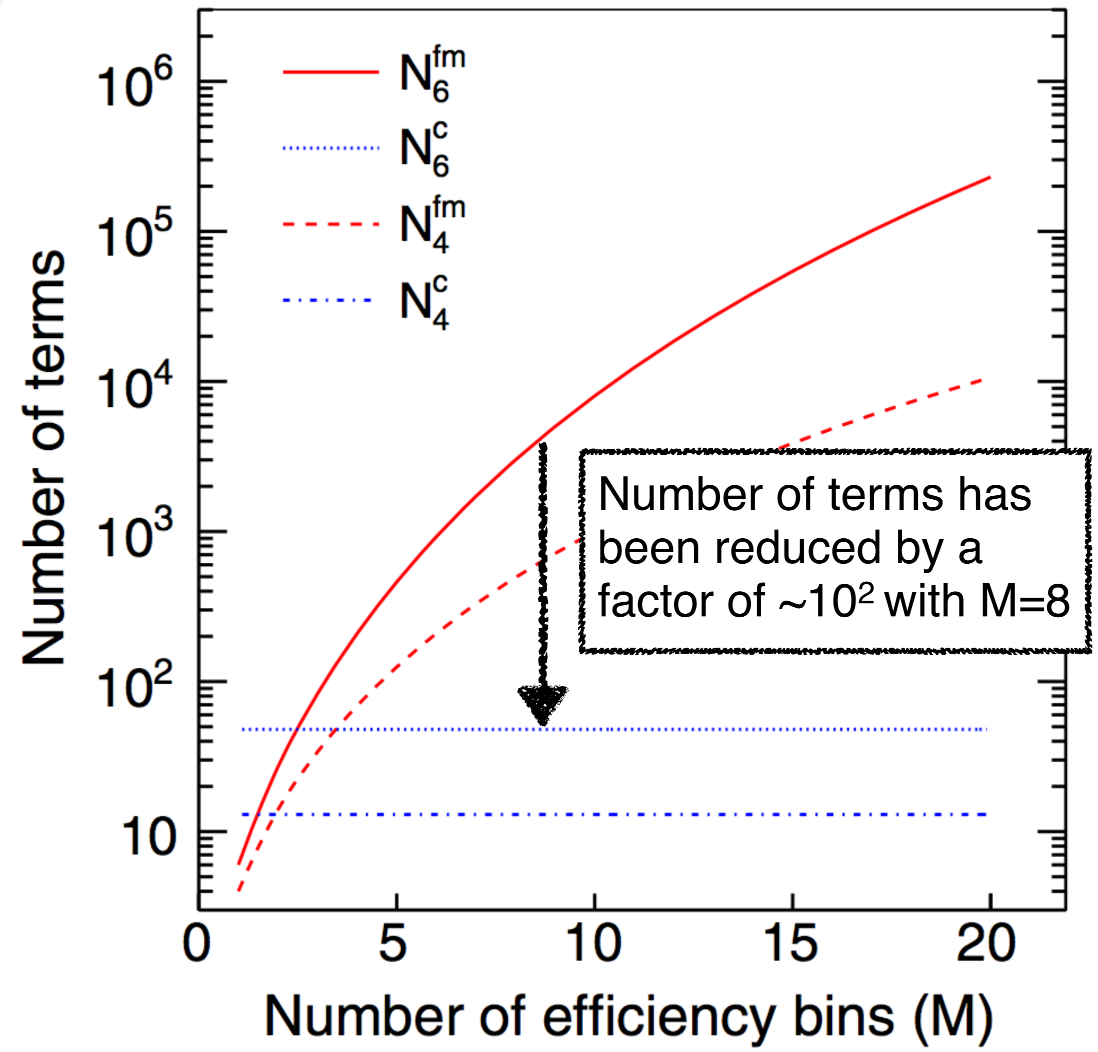
$$\langle Q^3\rangle_c = \langle q_{(1,1)}^3\rangle_c + 3\langle q_{(1,1)}q_{(2,1)}\rangle_c - 3\langle q_{(1,1)}q_{(2,2)}\rangle_c + \langle q_{(3,1)}\rangle_c - 3\langle q_{(3,2)}\rangle_c + 2\langle q_{(3,3)}\rangle_c, \tag{64}$$

$$\begin{aligned}\langle Q^4\rangle_c = {} & \langle q_{(1,1)}^4\rangle_c + 6\langle q_{(1,1)}^2 q_{(2,1)}\rangle_c - 6\langle q_{(1,1)}^2 q_{(2,2)}\rangle_c + 4\langle q_{(1,1)}q_{(3,1)}\rangle_c + 3\langle q_{(2,1)}^2\rangle_c + 3\langle q_{(2,2)}^2\rangle_c - 12\langle q_{(1,1)}q_{(3,2)}\rangle_c \\ & + 8\langle q_{(1,1)}q_{(3,3)}\rangle_c - 6\langle q_{(2,1)}q_{(2,2)}\rangle_c + \langle q_{(4,1)}\rangle_c - 7\langle q_{(4,2)}\rangle_c + 12\langle q_{(4,3)}\rangle_c - 6\langle q_{(4,4)}\rangle_c,\end{aligned} \tag{65}$$

$$\begin{aligned}\langle Q^5\rangle_c = {} & \langle q_{(1,1)}^5\rangle_c + 10\langle q_{(1,1)}^3 q_{(2,1)}\rangle_c - 10\langle q_{(1,1)}^3 q_{(2,2)}\rangle_c + 10\langle q_{(1,1)}^2 q_{(3,1)}\rangle_c - 30\langle q_{(1,1)}^2 q_{(3,2)}\rangle_c + 20\langle q_{(1,1)}^2 q_{(3,3)}\rangle_c + 15\langle q_{(2,2)}^2 q_{(1,1)}\rangle_c \\ & + 15\langle q_{(2,1)}^2 q_{(1,1)}\rangle_c - 30\langle q_{(1,1)}q_{(2,1)}q_{(2,2)}\rangle_c + 5\langle q_{(1,1)}q_{(4,1)}\rangle_c - 35\langle q_{(1,1)}q_{(4,2)}\rangle_c + 60\langle q_{(1,1)}q_{(4,3)}\rangle_c - 30\langle q_{(1,1)}q_{(4,4)}\rangle_c \\ & + 10\langle q_{(2,1)}q_{(3,1)}\rangle_c - 30\langle q_{(2,1)}q_{(3,2)}\rangle_c + 20\langle q_{(2,1)}q_{(3,3)}\rangle_c - 10\langle q_{(2,2)}q_{(3,1)}\rangle_c + 30\langle q_{(2,2)}q_{(3,2)}\rangle_c - 20\langle q_{(2,2)}q_{(3,3)}\rangle_c + \langle q_{(5,1)}\rangle_c \\ & - 15\langle q_{(5,2)}\rangle_c + 50\langle q_{(5,3)}\rangle_c - 60\langle q_{(5,4)}\rangle_c + 24\langle q_{(5,5)}\rangle_c,\end{aligned} \tag{66}$$

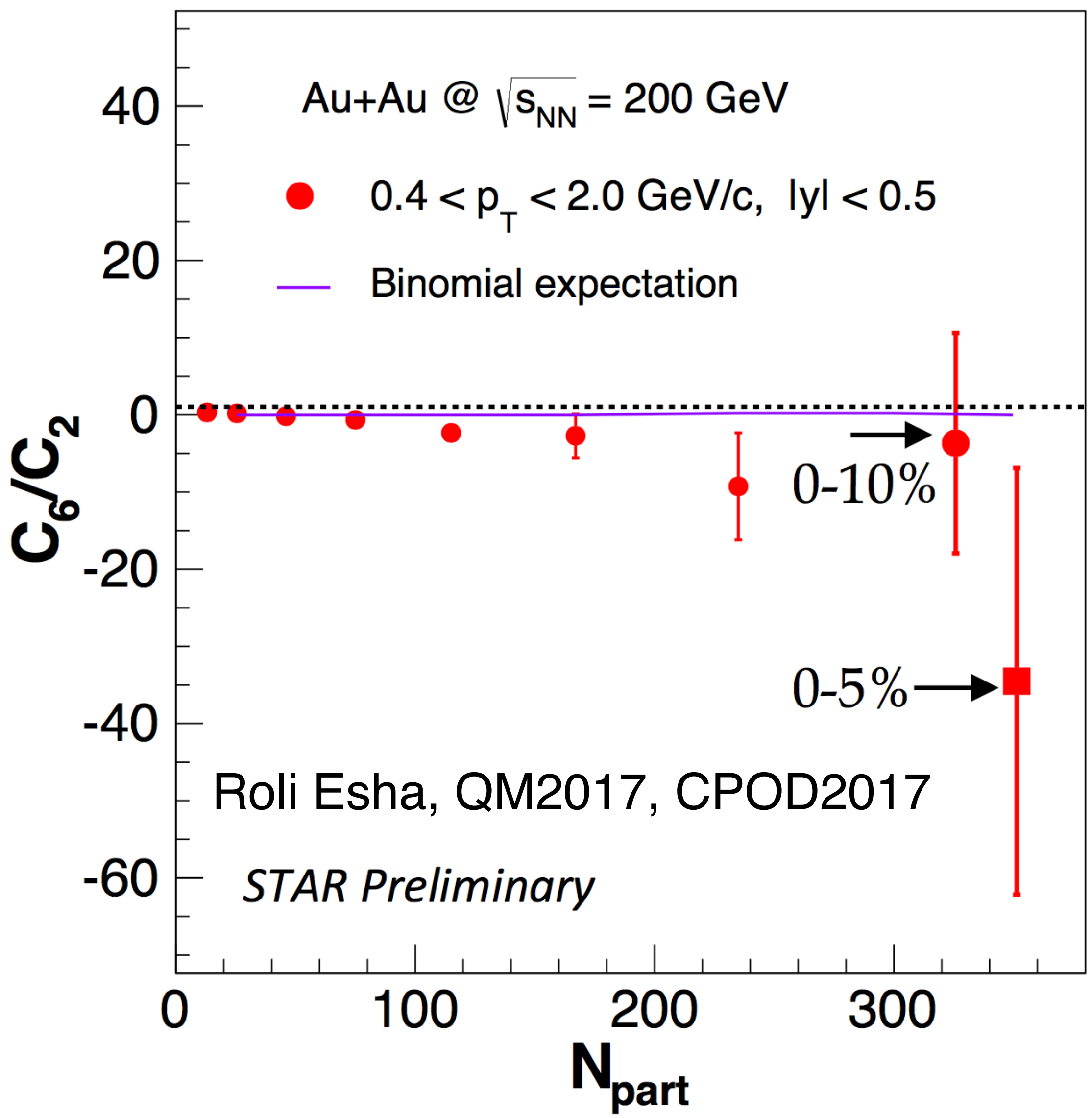
$$\begin{aligned}\langle Q^6\rangle_c = {} & \langle q_{(1,1)}^6\rangle_c + 15\langle q_{(1,1)}^4 q_{(2,1)}\rangle_c - 15\langle q_{(1,1)}^4 q_{(2,2)}\rangle_c + 20\langle q_{(1,1)}^3 q_{(3,1)}\rangle_c - 60\langle q_{(1,1)}^3 q_{(3,2)}\rangle_c + 40\langle q_{(1,1)}^3 q_{(3,3)}\rangle_c - 90\langle q_{(1,1)}^2 q_{(2,2)}q_{(2,1)}\rangle_c \\ & + 45\langle q_{(1,1)}^2 q_{(2,1)}^2\rangle_c + 45\langle q_{(1,1)}^2 q_{(2,2)}^2\rangle_c + 15\langle q_{(2,1)}^3\rangle_c - 15\langle q_{(2,2)}^3\rangle_c + 15\langle q_{(1,1)}^2 q_{(4,1)}\rangle_c - 105\langle q_{(1,1)}^2 q_{(4,2)}\rangle_c + 180\langle q_{(1,1)}^2 q_{(4,3)}\rangle_c \\ & - 90\langle q_{(1,1)}^2 q_{(4,4)}\rangle_c - 45\langle q_{(2,1)}^2 q_{(2,2)}\rangle_c + 45\langle q_{(2,2)}^2 q_{(2,1)}\rangle_c + 60\langle q_{(1,1)}q_{(2,1)}q_{(3,1)}\rangle_c - 180\langle q_{(1,1)}q_{(2,1)}q_{(3,2)}\rangle_c \\ & + 120\langle q_{(1,1)}q_{(2,1)}q_{(3,3)}\rangle_c - 60\langle q_{(1,1)}q_{(2,2)}q_{(3,1)}\rangle_c + 180\langle q_{(1,1)}q_{(2,2)}q_{(3,2)}\rangle_c - 120\langle q_{(1,1)}q_{(2,2)}q_{(3,3)}\rangle_c + 6\langle q_{(1,1)}q_{(5,1)}\rangle_c \\ & - 90\langle q_{(1,1)}q_{(5,2)}\rangle_c + 300\langle q_{(1,1)}q_{(5,3)}\rangle_c - 360\langle q_{(1,1)}q_{(5,4)}\rangle_c + 144\langle q_{(1,1)}q_{(5,5)}\rangle_c + 15\langle q_{(2,1)}q_{(4,1)}\rangle_c - 105\langle q_{(2,1)}q_{(4,2)}\rangle_c \\ & + 180\langle q_{(2,1)}q_{(4,3)}\rangle_c - 90\langle q_{(2,1)}q_{(4,4)}\rangle_c - 15\langle q_{(2,2)}q_{(4,1)}\rangle_c + 105\langle q_{(2,2)}q_{(4,2)}\rangle_c - 180\langle q_{(2,2)}q_{(4,3)}\rangle_c + 90\langle q_{(2,2)}q_{(4,4)}\rangle_c \\ & + 10\langle q_{(3,1)}^2\rangle_c - 60\langle q_{(3,1)}q_{(3,2)}\rangle_c + 40\langle q_{(3,1)}q_{(3,3)}\rangle_c + 90\langle q_{(3,2)}^2\rangle_c - 120\langle q_{(3,2)}q_{(3,3)}\rangle_c + 40\langle q_{(3,3)}^2\rangle_c + \langle q_{(6,1)}\rangle_c - 31\langle q_{(6,2)}\rangle_c \\ & + 180\langle q_{(6,3)}\rangle_c - 390\langle q_{(6,4)}\rangle_c + 360\langle q_{(6,5)}\rangle_c - 120\langle q_{(6,6)}\rangle_c,\end{aligned} \tag{67}$$

- ✓ Number of terms does not depend on efficiency bins.
- ✓ Calculation cost has been drastically suppressed.



net-proton C_6 in STAR

$$
\begin{aligned}
\langle Q^6\rangle_c = {} & \langle q_{(1,1)}^6\rangle_c + 15\langle q_{(1,1)}^4 q_{(2,1)}\rangle_c - 15\langle q_{(1,1)}^4 q_{(2,2)}\rangle_c + 20\langle q_{(1,1)}^3 q_{(3,1)}\rangle_c - 60\langle q_{(1,1)}^3 q_{(3,2)}\rangle_c + 40\langle q_{(1,1)}^3 q_{(3,3)}\rangle_c - 90\langle q_{(1,1)}^2 q_{(2,2)} q_{(2,1)}\rangle_c \\
& + 45\langle q_{(1,1)}^2 q_{(2,1)}^2\rangle_c + 45\langle q_{(1,1)}^2 q_{(2,2)}^2\rangle_c + 15\langle q_{(2,1)}^3\rangle_c - 15\langle q_{(2,2)}^3\rangle_c + 15\langle q_{(1,1)}^2 q_{(4,1)}\rangle_c - 105\langle q_{(1,1)}^2 q_{(4,2)}\rangle_c + 180\langle q_{(1,1)}^2 q_{(4,3)}\rangle_c \\
& - 90\langle q_{(1,1)}^2 q_{(4,4)}\rangle_c - 45\langle q_{(2,1)}^2 q_{(2,2)}\rangle_c + 45\langle q_{(2,2)}^2 q_{(2,1)}\rangle_c + 60\langle q_{(1,1)} q_{(2,1)} q_{(3,1)}\rangle_c - 180\langle q_{(1,1)} q_{(2,1)} q_{(3,2)}\rangle_c \\
& + 120\langle q_{(1,1)} q_{(2,1)} q_{(3,3)}\rangle_c - 60\langle q_{(1,1)} q_{(2,2)} q_{(3,1)}\rangle_c + 180\langle q_{(1,1)} q_{(2,2)} q_{(3,2)}\rangle_c - 120\langle q_{(1,1)} q_{(2,2)} q_{(3,3)}\rangle_c + 6\langle q_{(1,1)} q_{(5,1)}\rangle_c \\
& - 90\langle q_{(1,1)} q_{(5,2)}\rangle_c + 300\langle q_{(1,1)} q_{(5,3)}\rangle_c - 360\langle q_{(1,1)} q_{(5,4)}\rangle_c + 144\langle q_{(1,1)} q_{(5,5)}\rangle_c + 15\langle q_{(2,1)} q_{(4,1)}\rangle_c - 105\langle q_{(2,1)} q_{(4,2)}\rangle_c \\
& + 180\langle q_{(2,1)} q_{(4,3)}\rangle_c - 90\langle q_{(2,1)} q_{(4,4)}\rangle_c - 15\langle q_{(2,2)} q_{(4,1)}\rangle_c + 105\langle q_{(2,2)} q_{(4,2)}\rangle_c - 180\langle q_{(2,2)} q_{(4,3)}\rangle_c + 90\langle q_{(2,2)} q_{(4,4)}\rangle_c \\
& + 10\langle q_{(3,1)}^2\rangle_c - 60\langle q_{(3,1)} q_{(3,2)}\rangle_c + 40\langle q_{(3,1)} q_{(3,3)}\rangle_c + 90\langle q_{(3,2)}^2\rangle_c - 120\langle q_{(3,2)} q_{(3,3)}\rangle_c + 40\langle q_{(3,3)}^2\rangle_c + \langle q_{(6,1)}\rangle_c - 31\langle q_{(6,2)}\rangle_c \\
& + 180\langle q_{(6,3)}\rangle_c - 390\langle q_{(6,4)}\rangle_c + 360\langle q_{(6,5)}\rangle_c - 120\langle q_{(6,6)}\rangle_c, \qquad (67)
\end{aligned}
$$



✓ **It was difficult to calculate C_6 and its errors by conventional formulas using factorial moments within the realistic time scale, which is the reason why I started to work with M. Kitazawa for efficient formulas.**

M	Factorial moment	New method
4	64.7 s	30.8 s
8	17.3×10^2 s	31.3 s
12	14.1×10^3 s	32.3 s
200		62.7 s

implementations. The codes are executed on the same CPU (3 GHz Intel Core i7) for 1×10^5 events and $M = 4, 8, 12$, and 200. One finds that the CPU time with the conventional method

In STAR preliminary results for C_6/C_2, M=8 is implemented, and $2\text{x}10^8$ events were analyzed for 0-10% centrality. If we use the conventional formulas, then

$17.3\text{x}10^2$ (s) x 2000 ~ 944 hours ~ 40 days

would be necessary to calculate the data points. We need to repeat this with ~300 times to estimate the statistical errors!

Analytical calculation

- ✓ **Assume two distributions which have the same cumulants (C_m+C_m=2C_m) with different efficiencies.**
- ✓ **Apply correction using the averaged efficiency and see the deviation.**

$$\bar{\varepsilon} = (\varepsilon_{\rm A} + \varepsilon_{\rm B})/2 \qquad \Delta\varepsilon = \varepsilon_{\rm A} - \varepsilon_{\rm B}$$

$$\Delta K_m = K_m - K_m^{\rm (ave)} = 2C_m - K_m^{\rm (ave)}$$

- ✓ **The 1st order cumulant can be recovered by averaged efficiency.**

$$K_1^{\rm ave} = \langle N_A \rangle + \langle N_B \rangle = \frac{\langle n_A \rangle}{\bar{\varepsilon}} + \frac{\langle n_B \rangle}{\bar{\varepsilon}}$$
$$= \frac{\varepsilon_A C_1}{\bar{\varepsilon}} + \frac{\varepsilon_B C_1}{\bar{\varepsilon}} = 2C_1$$

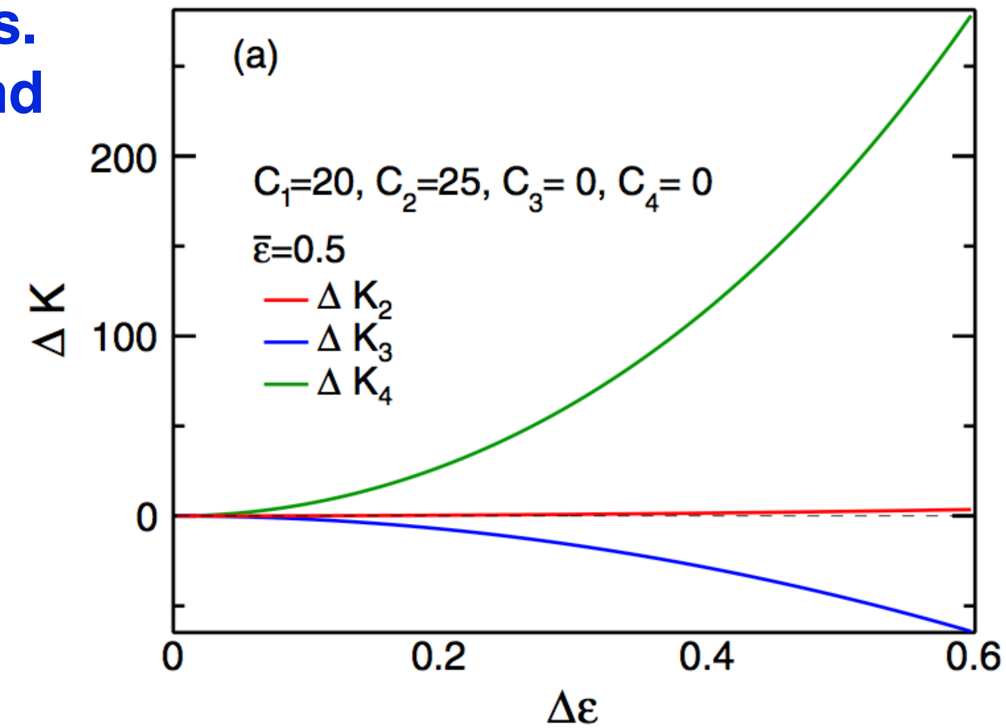
- ✓ **Higher the order of cumulant is, larger deviation appears.**
- ✓ **Interestingly, deviation becomes zero if both distributions are Poisson (C_m=C_1).**

$$\Delta K_2 = \frac{1}{2}\left(\frac{\Delta\varepsilon}{\bar{\varepsilon}}\right)^2 (C_2 - C_1),$$

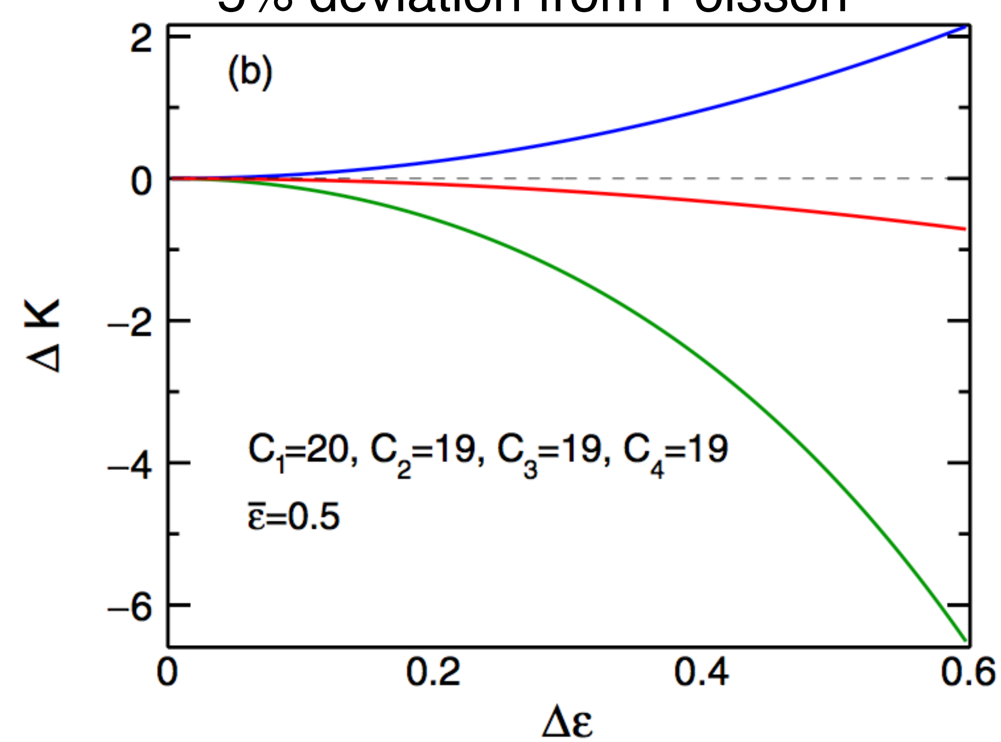
$$\Delta K_3 = \frac{3}{2}\left(\frac{\Delta\varepsilon}{\bar{\varepsilon}}\right)^2 (C_3 - 2C_2 + C_1),$$

$$\Delta K_4 = \frac{1}{2}\left(\frac{\Delta\varepsilon}{\bar{\varepsilon}}\right)^2 (6C_4 - 18C_3 + 19C_2 - 7C_1) + \frac{1}{8}\left(\frac{\Delta\varepsilon}{\bar{\varepsilon}}\right)^4 (C_4 - 6C_3 + 11C_2 - 6C_1),$$

- Gauss



- 5% deviation from Poisson

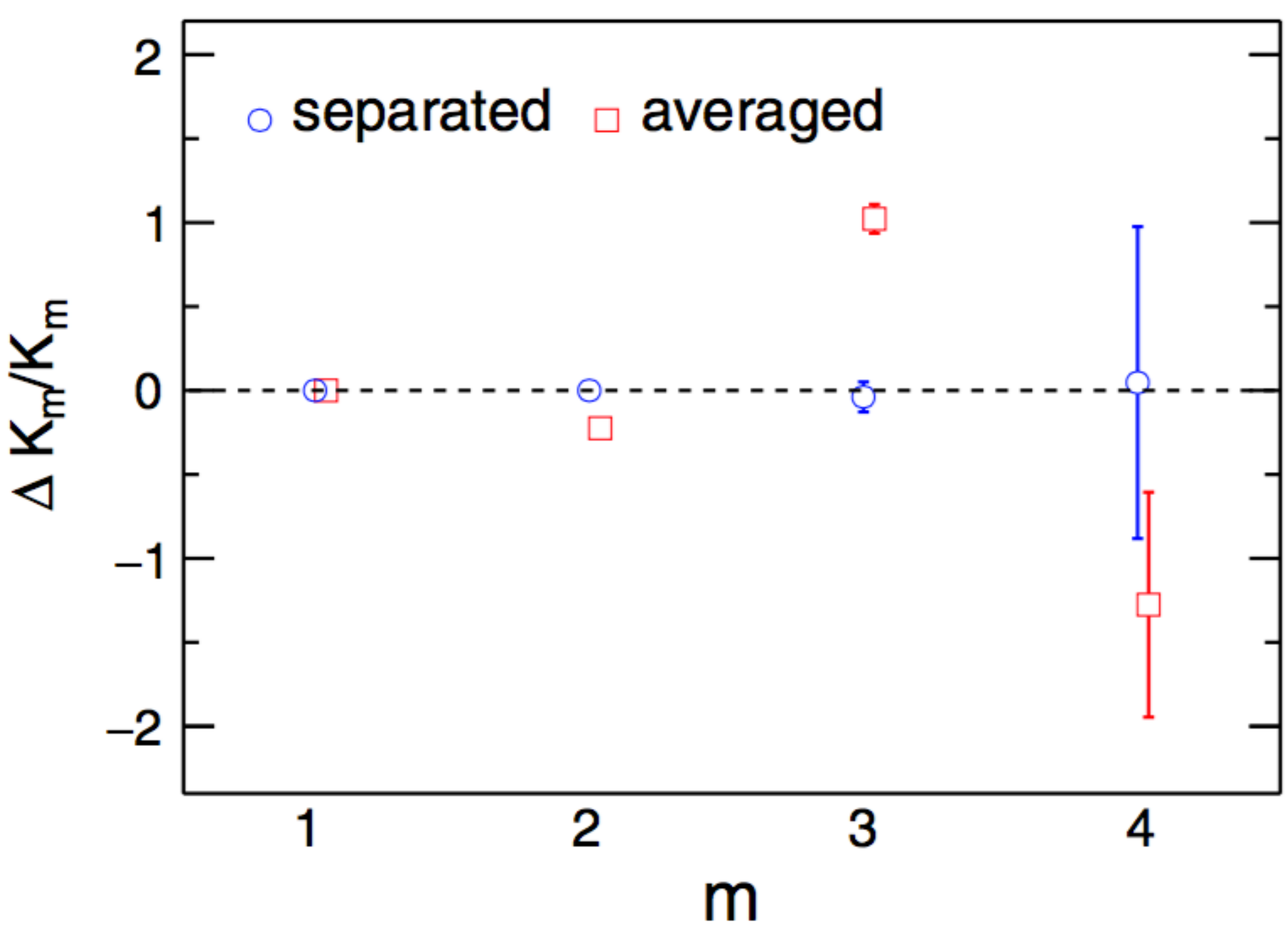


Toy model for net-charge fluctuation

- ✓ Now we can check the validity of using averaged efficiency with toy model numerically.
- ✓ In net-charge publication from STAR, weighted averaged efficiency between π/K/p is used for efficiency correction.
- ✓ Assume pions follow Gauss and the others are Poisson.

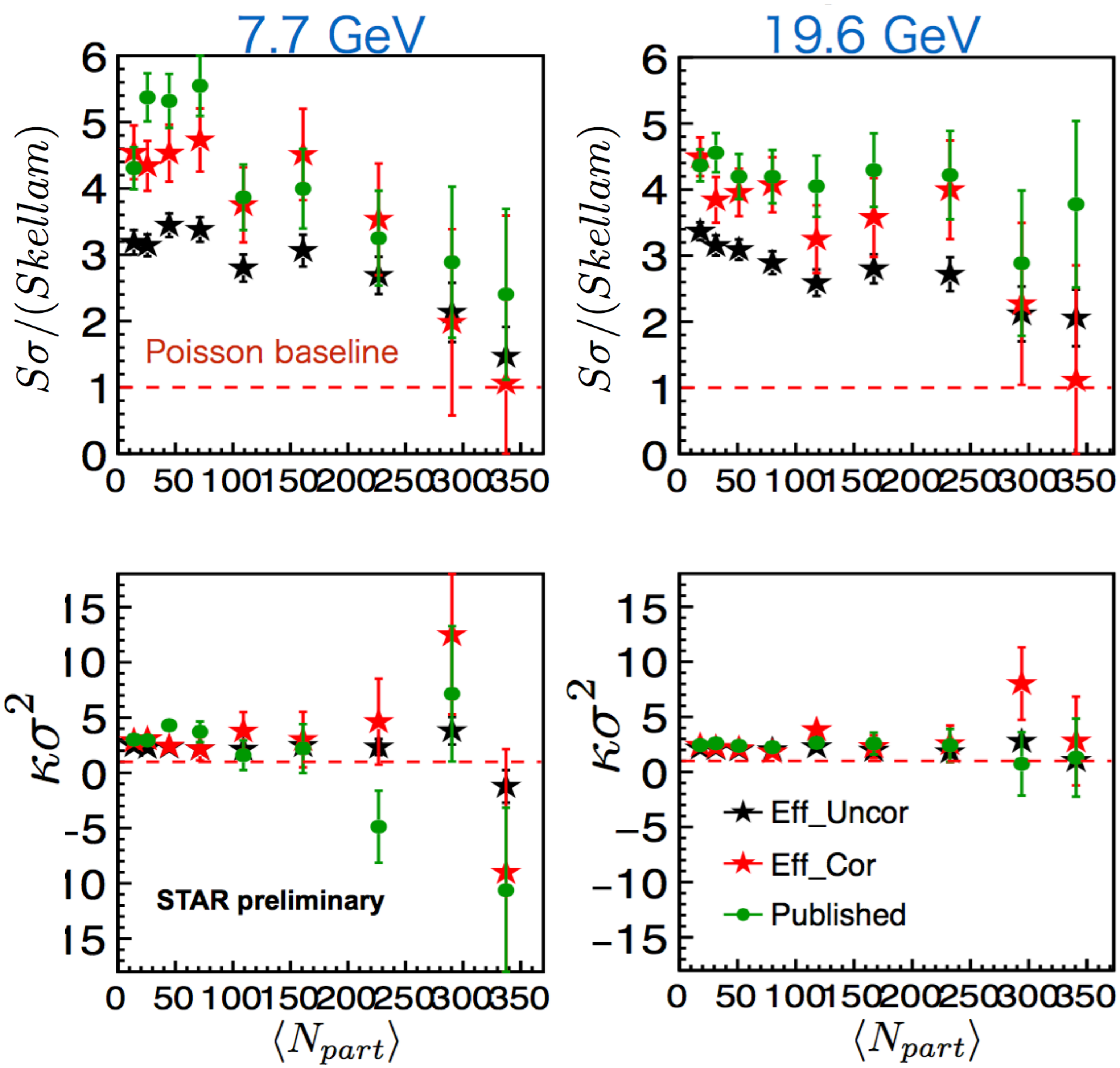
$$\varepsilon^{\pm}_{\pi Kp} = \frac{\sum_i \varepsilon_i^{\pm} N_i^{\pm}}{\sum_i N_i^{\pm}}$$

Particles	$P(N)$	Charge	Mean	Sigma	Efficiency
π^+	Gauss	$+1$	30	8	0.3
K^+	Poisson	$+1$	10		0.6
p	Poisson	$+1$	8		0.9
π^-	Gauss	-1	25	7	0.25
K^-	Poisson	-1	4		0.55
$\bar{p}$	Poisson	-1	3		0.85



- ✓ Finite deviations are observed for m>=2.
- ✓ If π/K/p are all Poisson, deviation becomes zero.

Recent STAR preliminary results

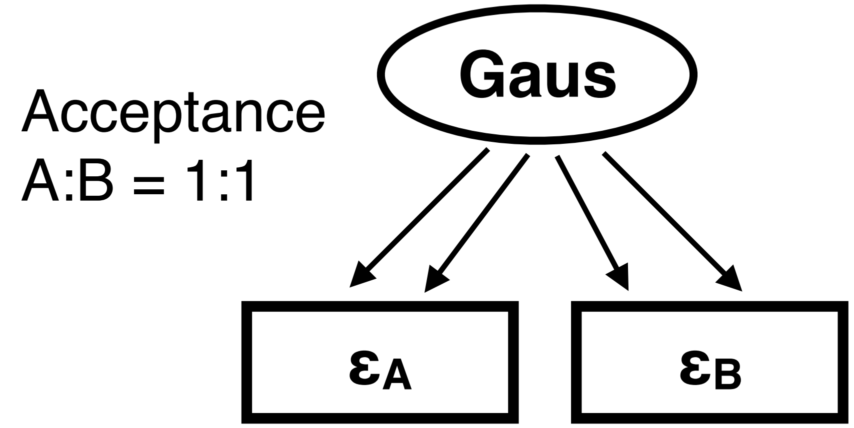


- ✓ **Particle species dependent efficiency is implemented in efficiency correction for net-charge fluctuation at the STAR experiment.**
- ✓ **Results are consistent with the publication within errors.**
- ✓ **Separated efficiency correction will be important in BESII!!**

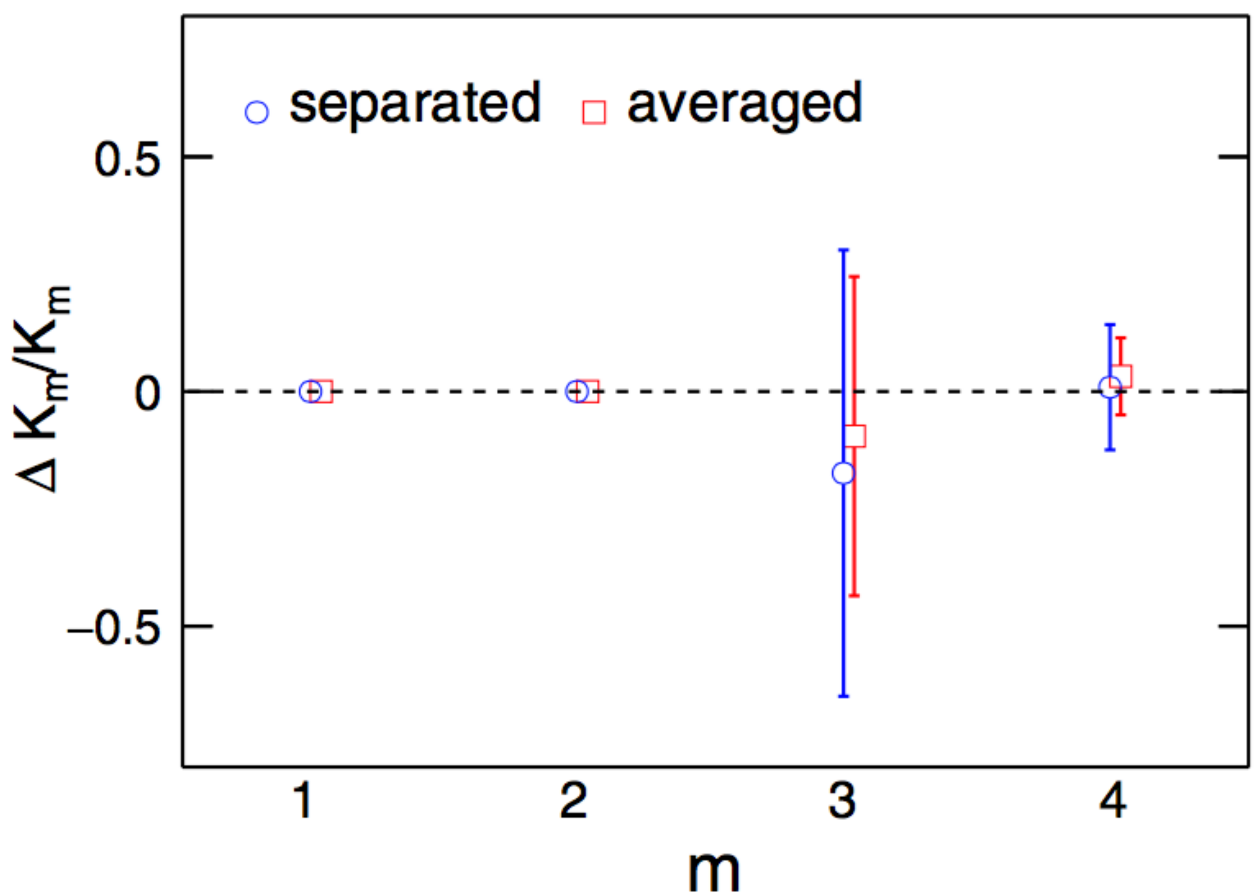
✦ **T. Sugiura, JPS fall meeting 2017**

Toy model for acceptance dependent efficiency

- ✓ Generate a Gaus distribution.
- ✓ Particles randomly incident on two detectors A and B, which have different efficiencies ε_A and ε_B.
- ✓ Apply correction using averaged efficiency for A and B.



$P(N)$	Charge	Mean	Sigma	Efficiency
Gauss	$+1$	20	$\sqrt{32}$	$\varepsilon_A^+ = 0.9,\ \varepsilon_B^+ = 0.3$
Gauss	-1	8	$\sqrt{8}$	$\varepsilon_A^- = 0.4,\ \varepsilon_B^- = 0.8$



- ✓ No deviation.
- ✓ When we focus on one particle, it is measured with the averaged efficiency randomly and independently, which corresponds to the case of single efficiency bin.

→Underlying physics is identical for A and B.

Conclusion

✓ New efficiency correction formulas have been developed which can drastically reduce the calculation cost compared to conventional ones in case of many efficiency bins.

✓ Efficiency correction has to be performed with appropriate efficiency bins. Otherwise, results can artificially deviate.

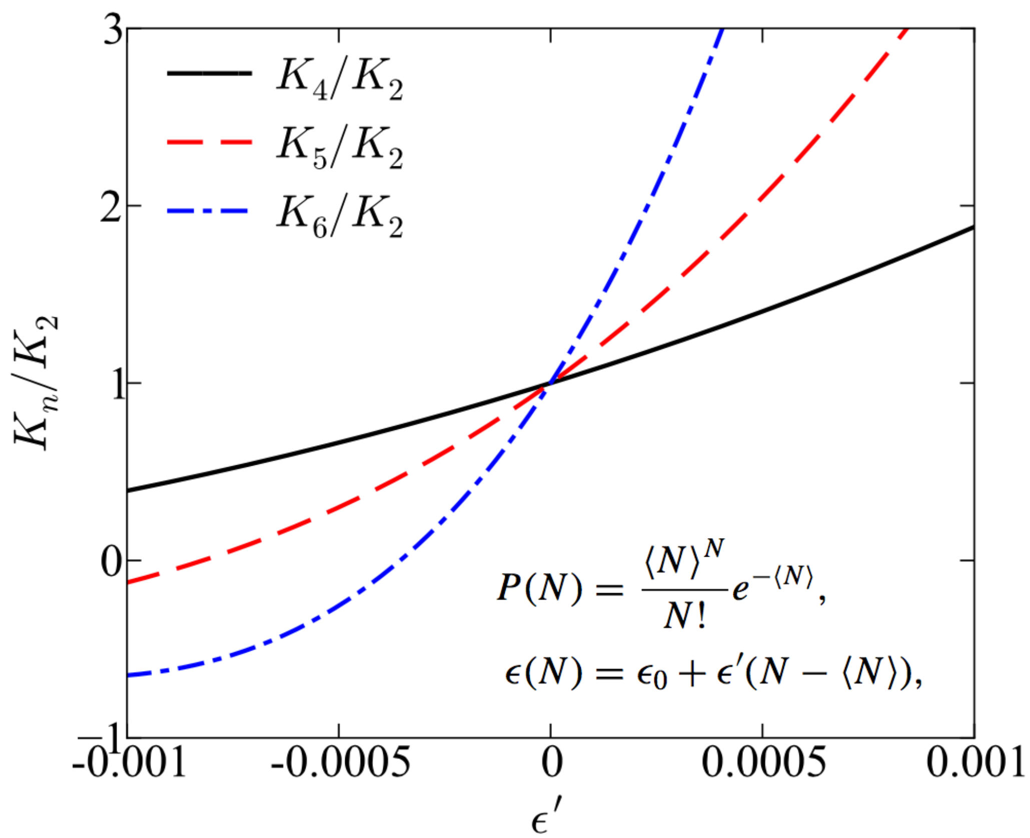
New unfolding method to reconstruct any unknown distribution around the QCD critical point and its application to STAR data with binomial model

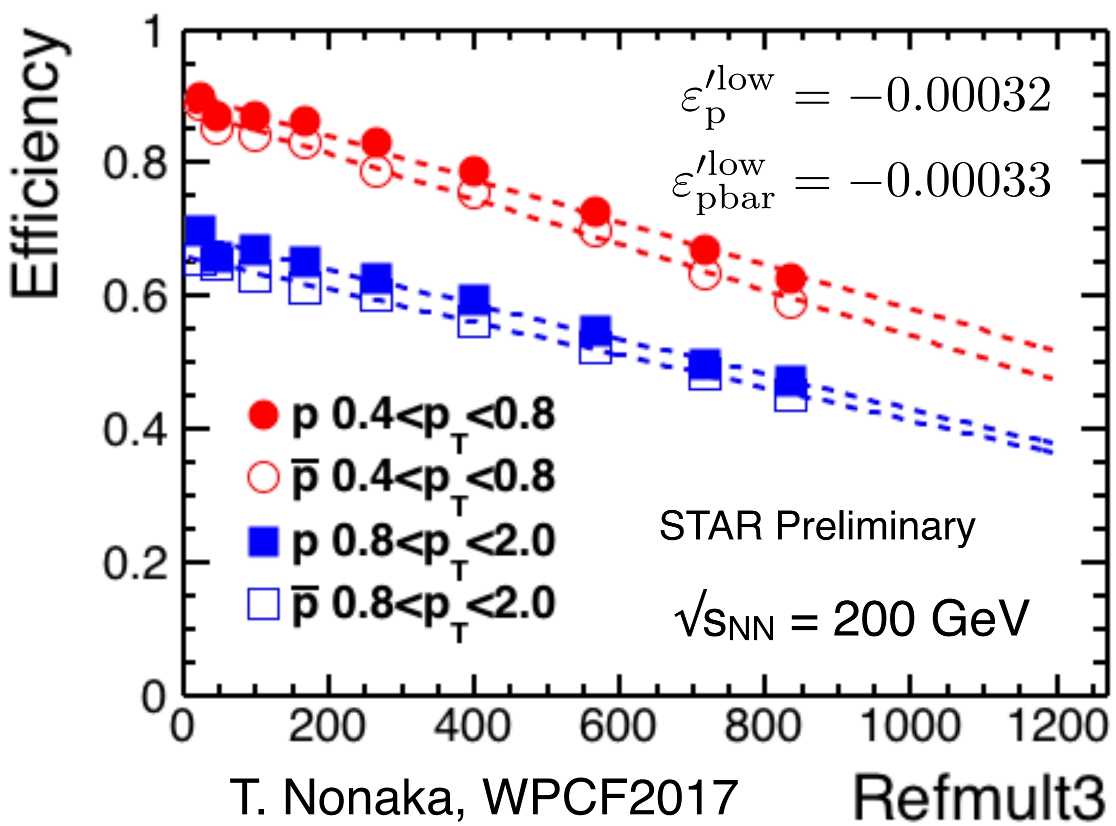
S. Esumi, X. Luo, B. Mohanty, T. Nonaka, T. Sugiura, N. Xu

See also ShinIchi's talk on Tuesday

Known issues on eff. correction

- ✓ **Efficiency correction on cumulants has been established based on the binomial model.**
  - M. Kitazawa : PRC.86.024904, M. Kitazawa and M. Asakawa : PRC.86.024904
  - A. Bzdak and V. Koch : PRC.86.044904, PRC.91.027901, X. Luo : PRC.91.034907
  - T. Nonaka et al : PRC.94.034909
  - T. Nonaka, M. Kitazawa, S. Esumi : PRC.95.064912
- ✓ **Efficiency correction does not work in some cases, e.g. multiplicity dependent efficiency, non-binomial efficiency.**
  - A. Bzdak et al : PRC.94.064907





T. Nonaka, WPCF2017

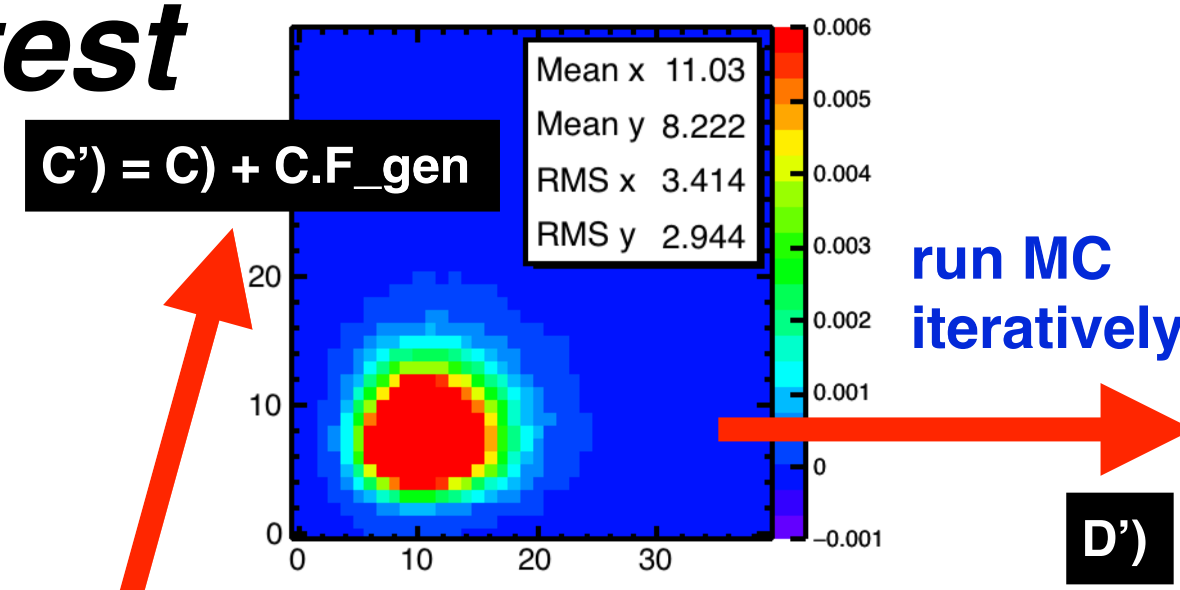
Known issues

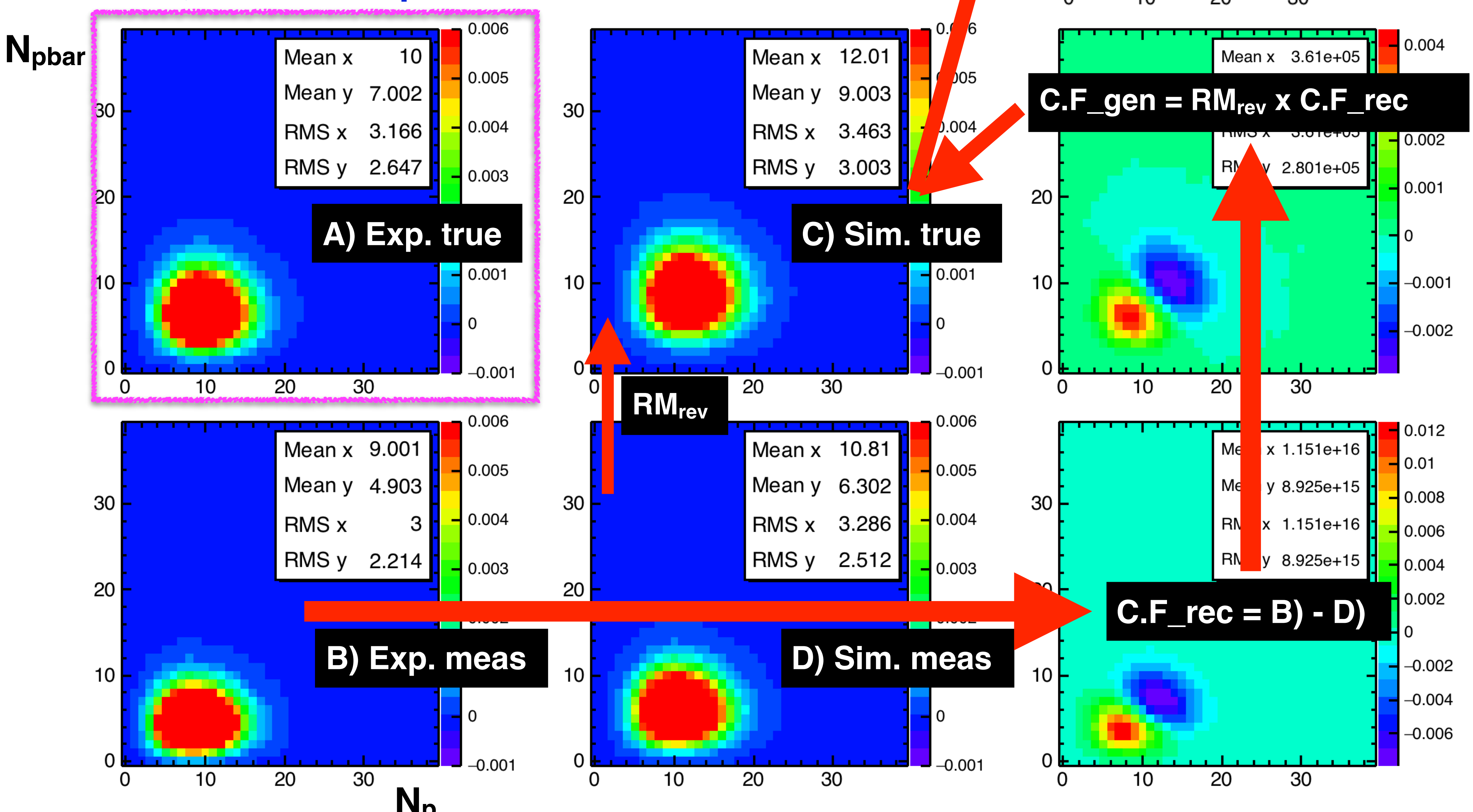
- ✓ **Efficiency correction on cumulants has been established based on the binomial model.**
  - M. Kitazawa : PRC.86.024904, M. Kitazawa and M. Asakawa : PRC.86.024904
  - A. Bzdak and V. Koch : PRC.86.044904, PRC.91.027901, X. Luo : PRC.91.034907
  - T. Nonaka et al : PRC.94.034909
  - T. Nonaka, M. Kitazawa, S. Esumi : PRC.95.064912
- ✓ **Efficiency correction does not work in some cases, e.g. multiplicity dependent efficiency, non-binomial efficiency.**
  - A. Bzdak et al : PRC.94.064907

- ✓ **Unfolding would be necessary to correct these effect.**
  - **We suggest new unfolding method to reconstruct any unknown distribution around the QCD critical point.**

Methodology : Poisson test

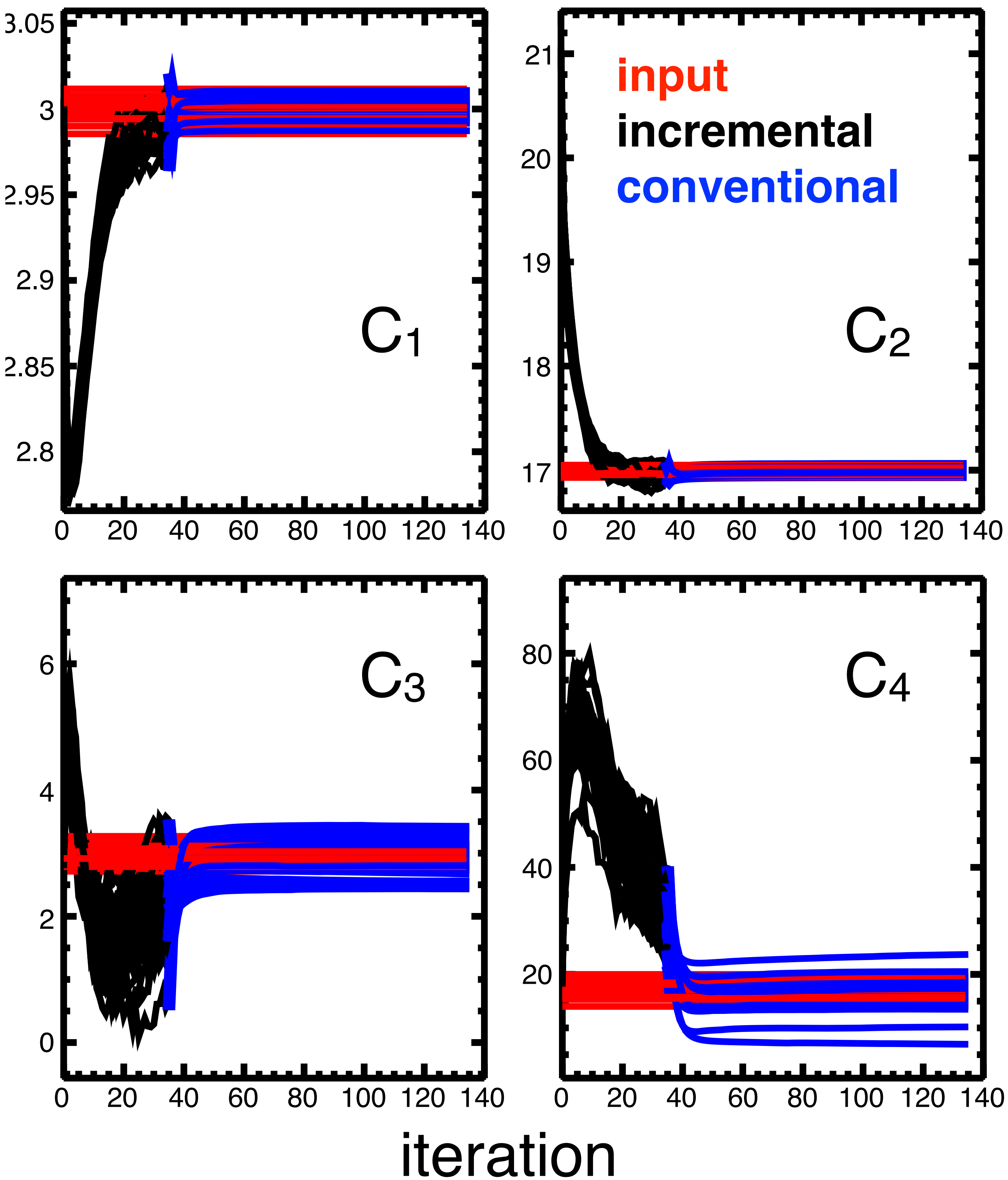
- ✓ Two Poisson distributions which have different mean value are generated and randomly sampled with efficiency.
- ✓ Difference between exp.meas and sim.meas is applied to sim.true to get the corrected distribution, which is repeated with iterative MC.





MC filter : binomial efficiency $\varepsilon_p = 0.9$, $\varepsilon_{pbar} = 0.7$

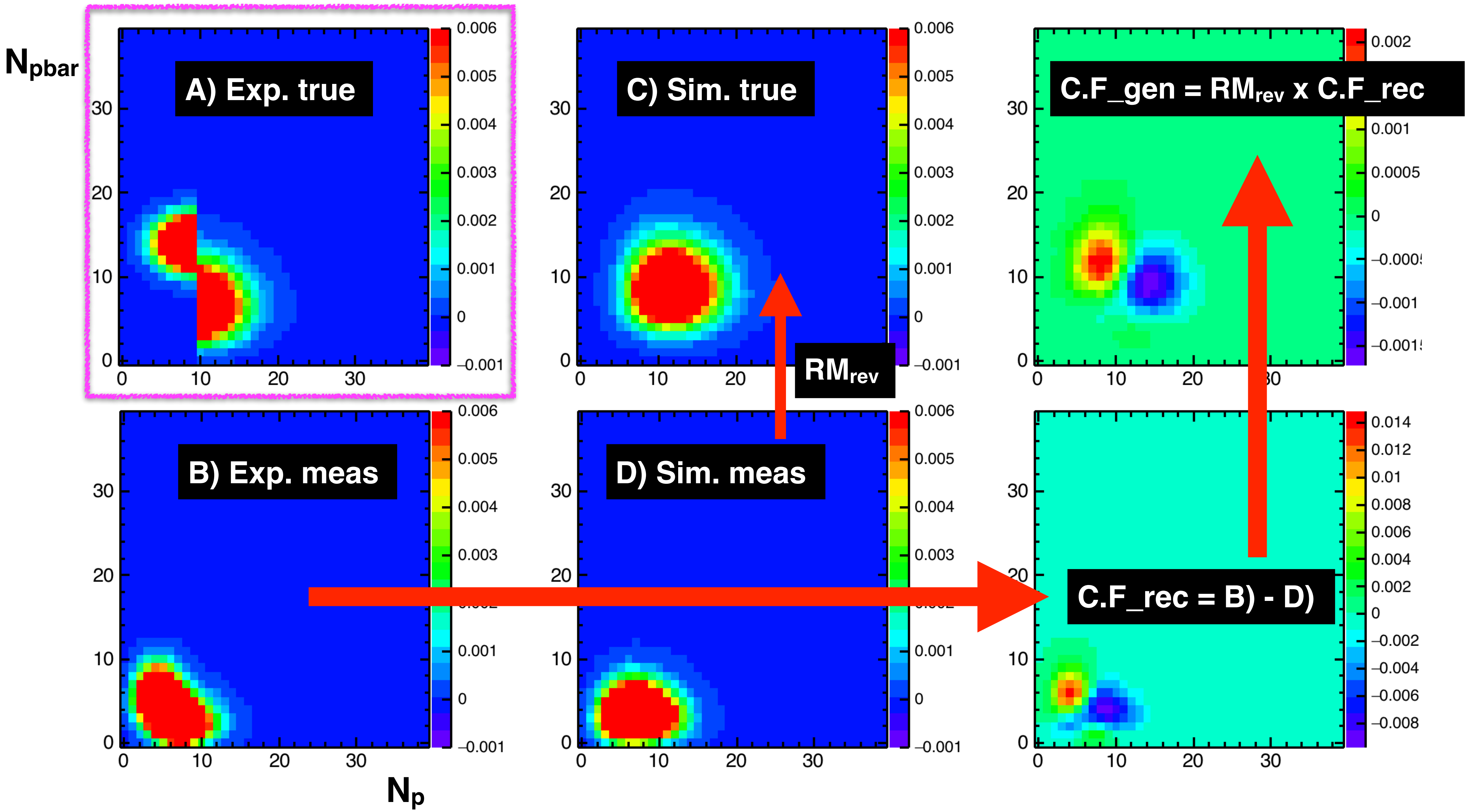
Poisson test : Cumulants



- ✓ We can stop iterations once cumulants don't change with iterations.
- ✓ Incremental unfolding is effective way to recover bins that don't exist in simulation, but seems difficult to get higher order cumulants converged.
- ✓ Conventional unfolding (not updating the response matrix) is also implemented to get cumulants converged.

Critical shape test

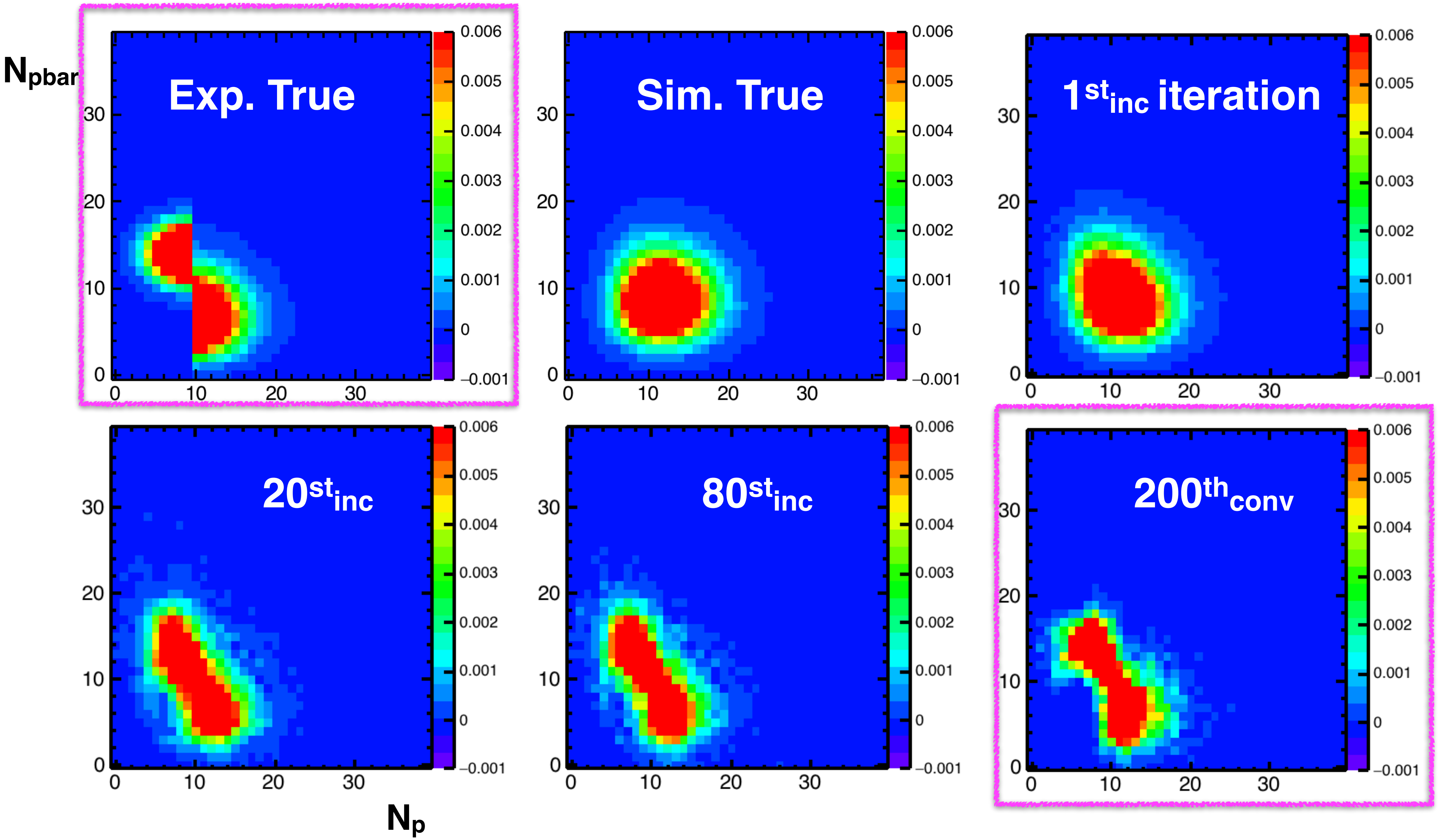
✓ Can we extract any unknown distribution??



MC filter : binomial efficiency $\varepsilon_p = 0.6$, $\varepsilon_{pbar} = 0.4$

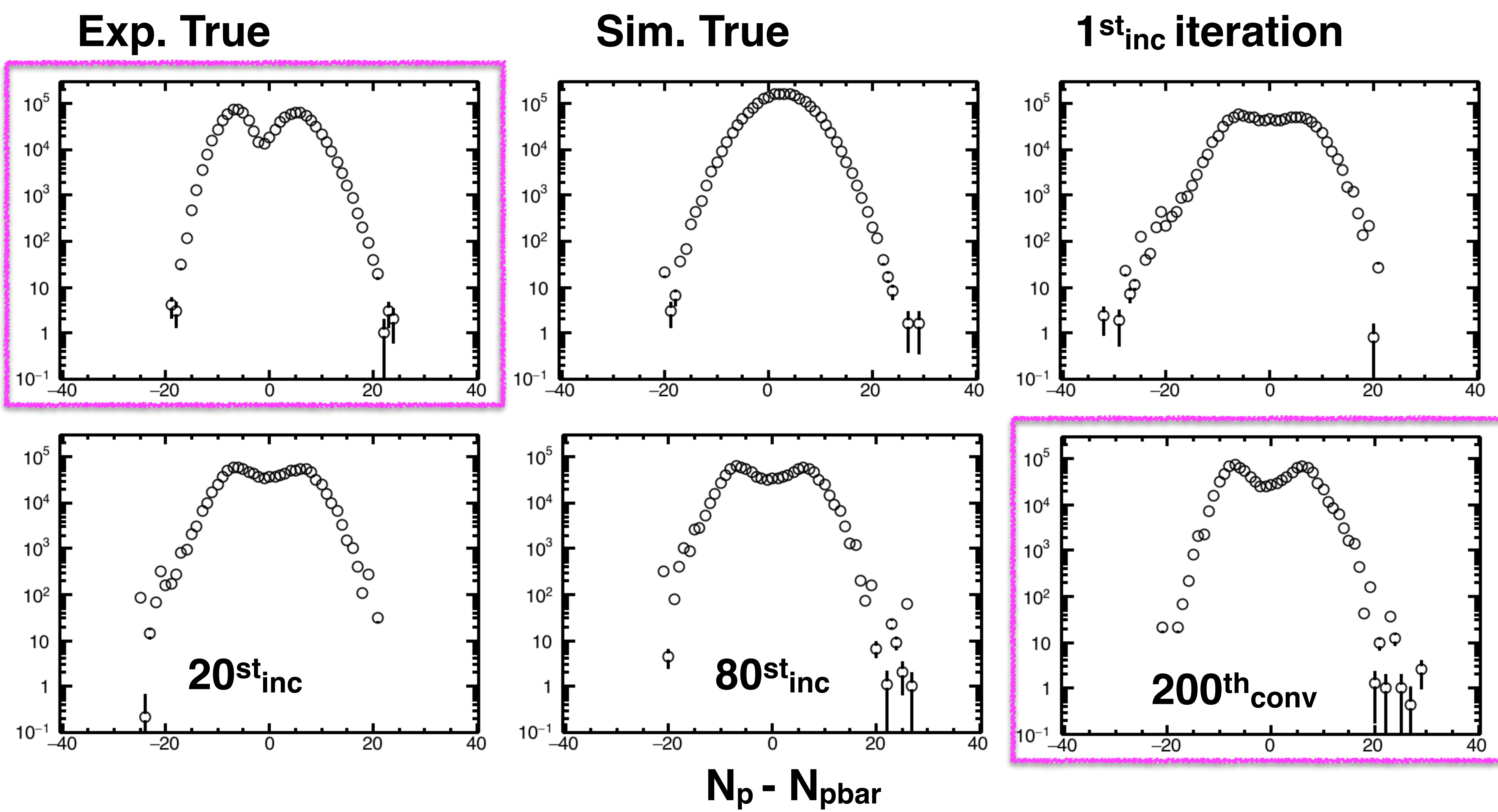
Critical shape test

✓ **Yes! Critical shape has been recovered.**

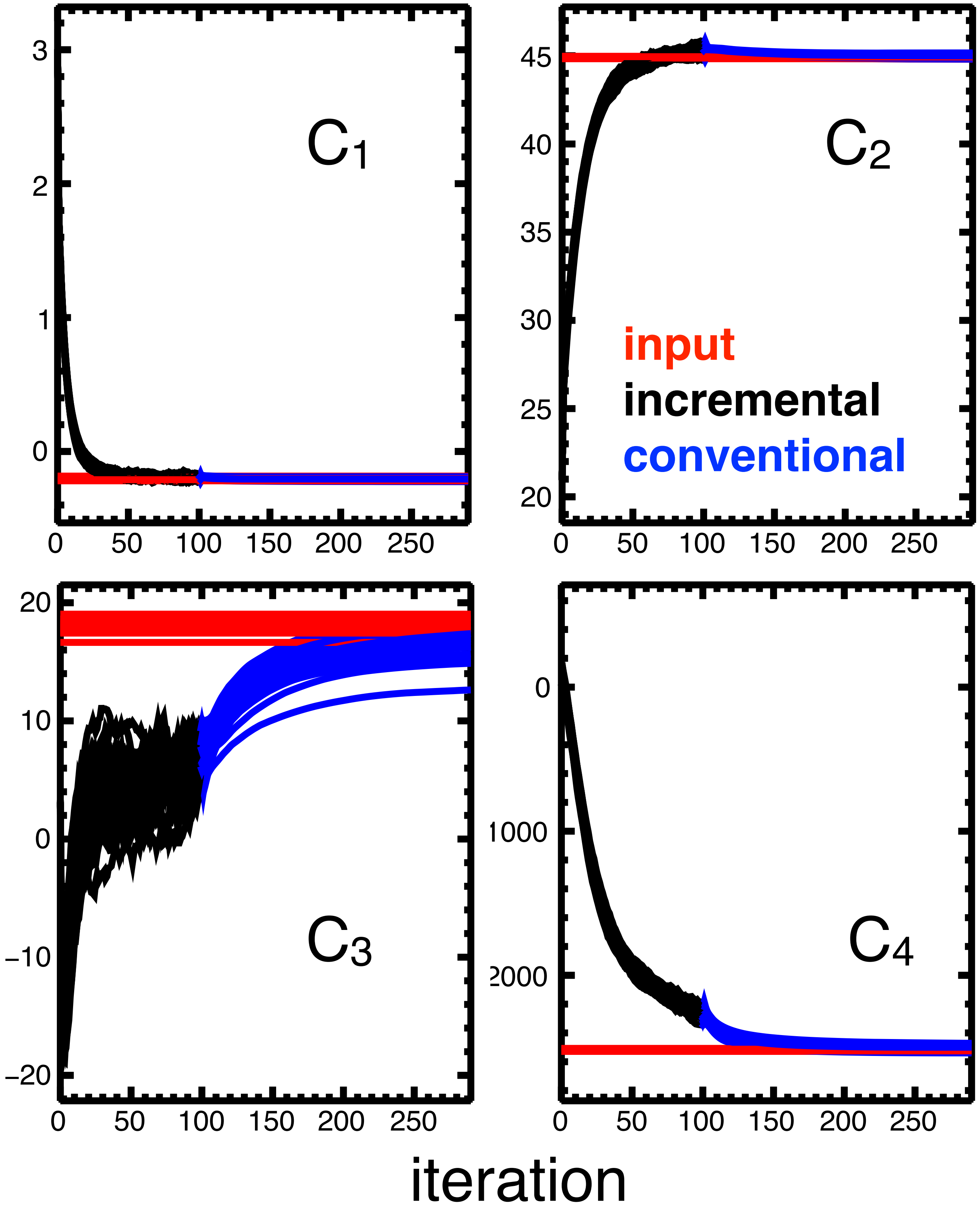


Critical shape test : Net-distribution

✓ **Two-peak structure in net-distribution has been recovered.**



Critical shape test : Cumulants

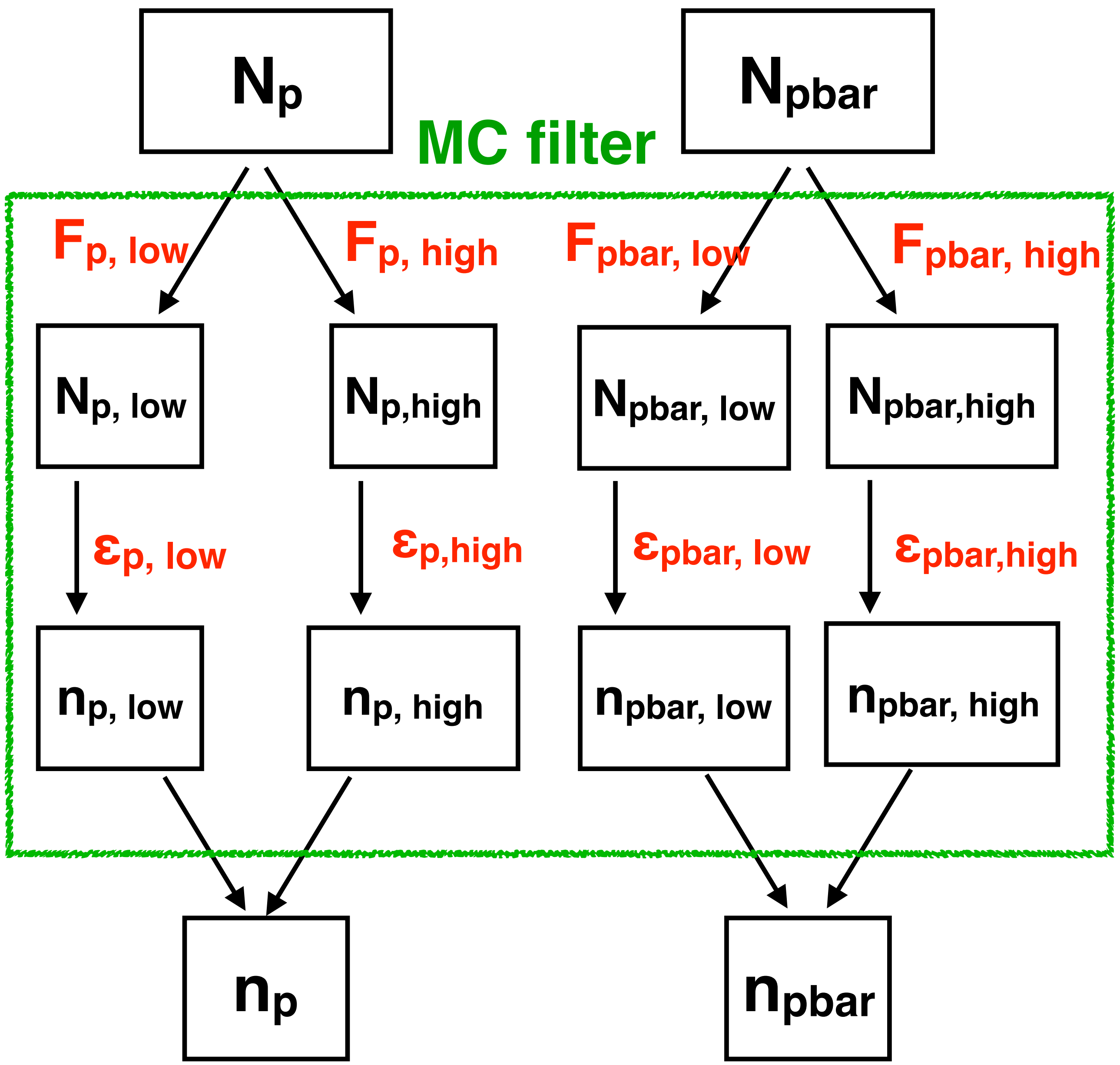


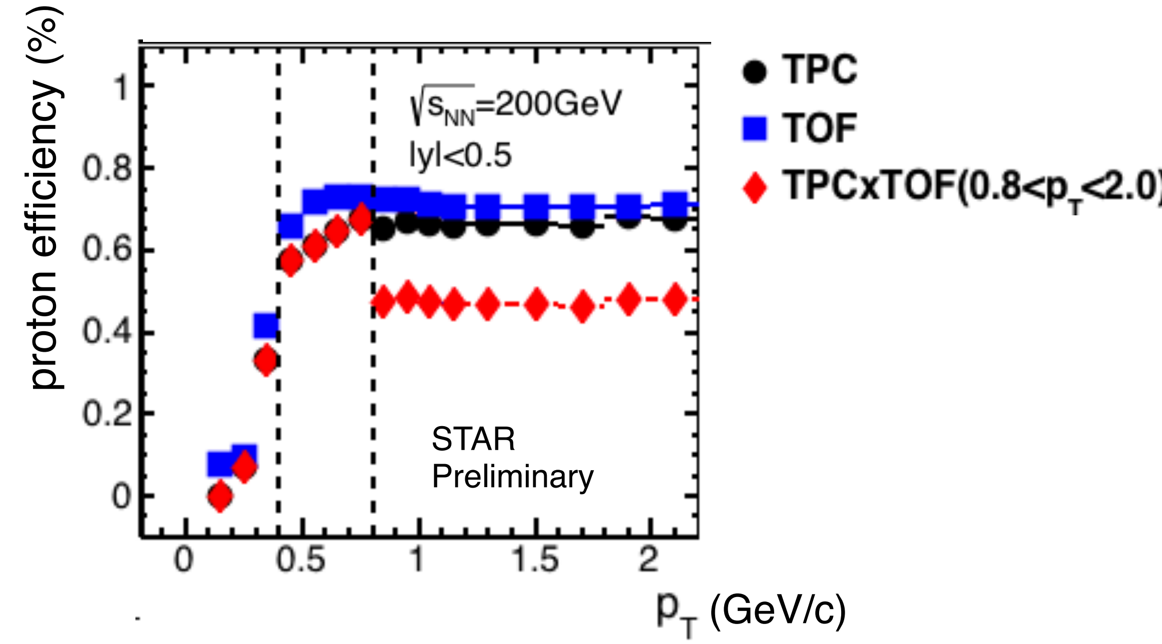
- ✓ **Cumulants have converged to input values.**
- ✓ **New unfolding method is established.**

Before we start non-binomial study, it is important to apply unfolding to experimental data assuming the binomial model and see whether the results are consistent with efficiency correction or not.

p_T dependent efficiency

F : fraction of protons at low(high) p_T region with respect to $0.4<p_T<2.0$ (GeV/c), which is calculated with efficiency corrected C_1 for (anti)proton.

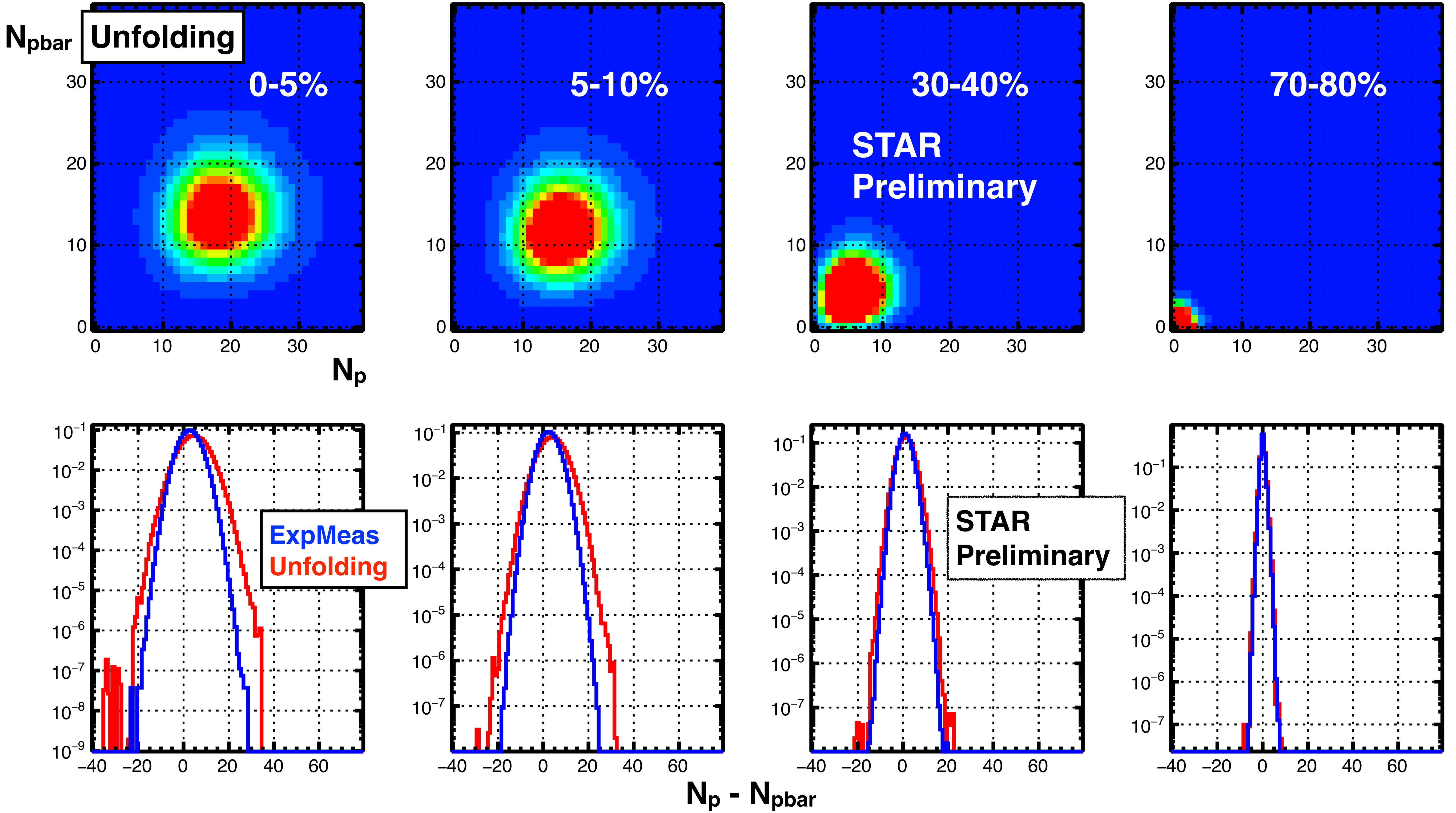




- ✦ **p_T dependent efficiency correction has been implemented in net-proton analysis.**
- ✦ **For unfolding, p_T dependent efficiency can be included inside the MC filter.**

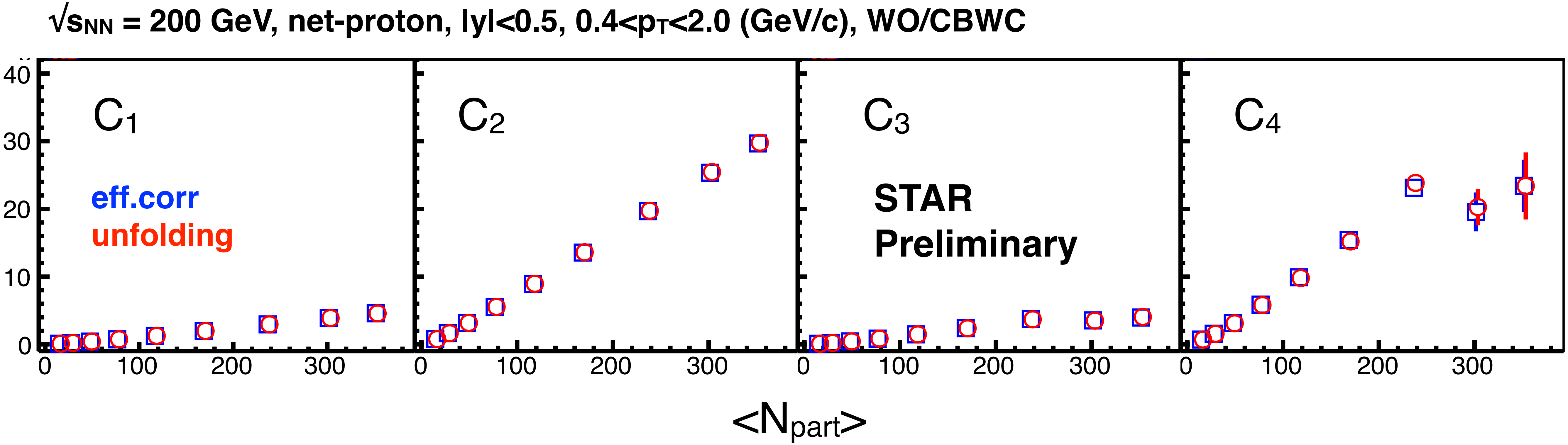
Unfolded distributions at $\sqrt{s_{NN}}$ = 200 GeV

$\sqrt{s_{NN}}$ = 200 GeV, net-proton, |y| < 0.5, 0.4 < p_T < 2.0 (GeV/c), without CBWC, binomial model



Results

✓ **Unfolding with binomial model gives consistent results with efficiency correction, which indicates that the unfolding approach works well.**

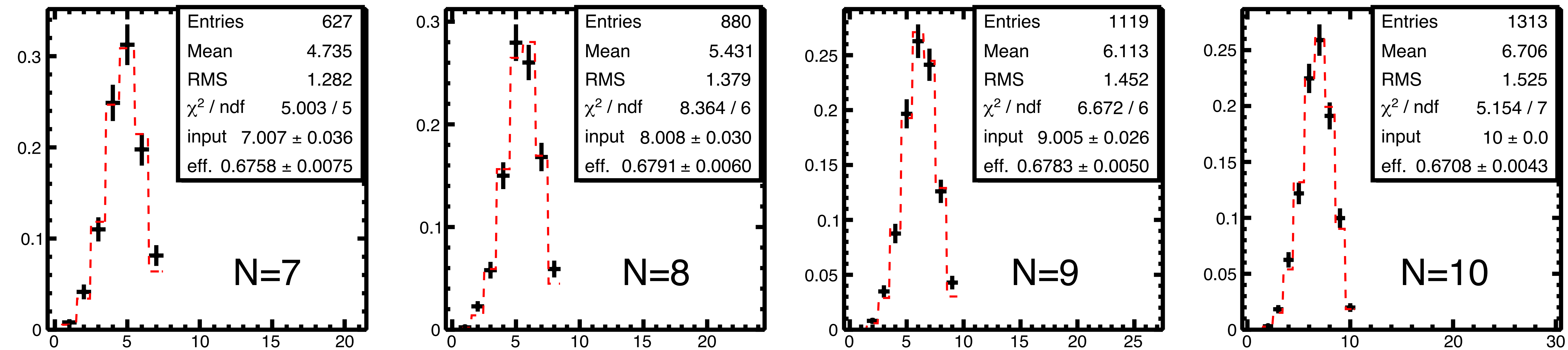


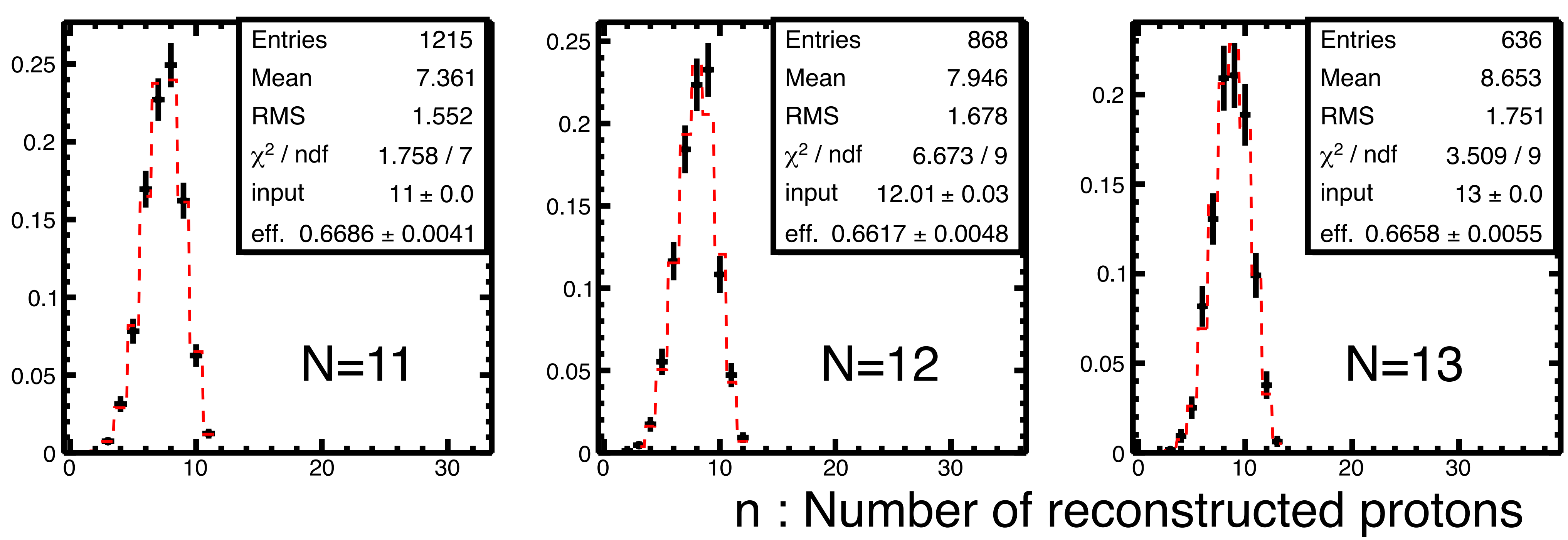
Fit to the embedding samples

- ✓ **Fitting to the embedding samples with the binomial distribution has been performed, which looks binomial within errors.**
- ✓ **Non-binomial distribution can be also used for fitting.**

$$B(n,N;\epsilon) = \frac{N!}{n!(N-n)!}\epsilon^n(1-\epsilon)^{N-n}$$

N : # of input particles
ε : efficiency





embedding samples for proton,
$\sqrt{s_{NN}}$ = 200 GeV,
0-5% centrality,
1.0 < p_T < 2.0 (GeV/c)

STAR Preliminary

Summary

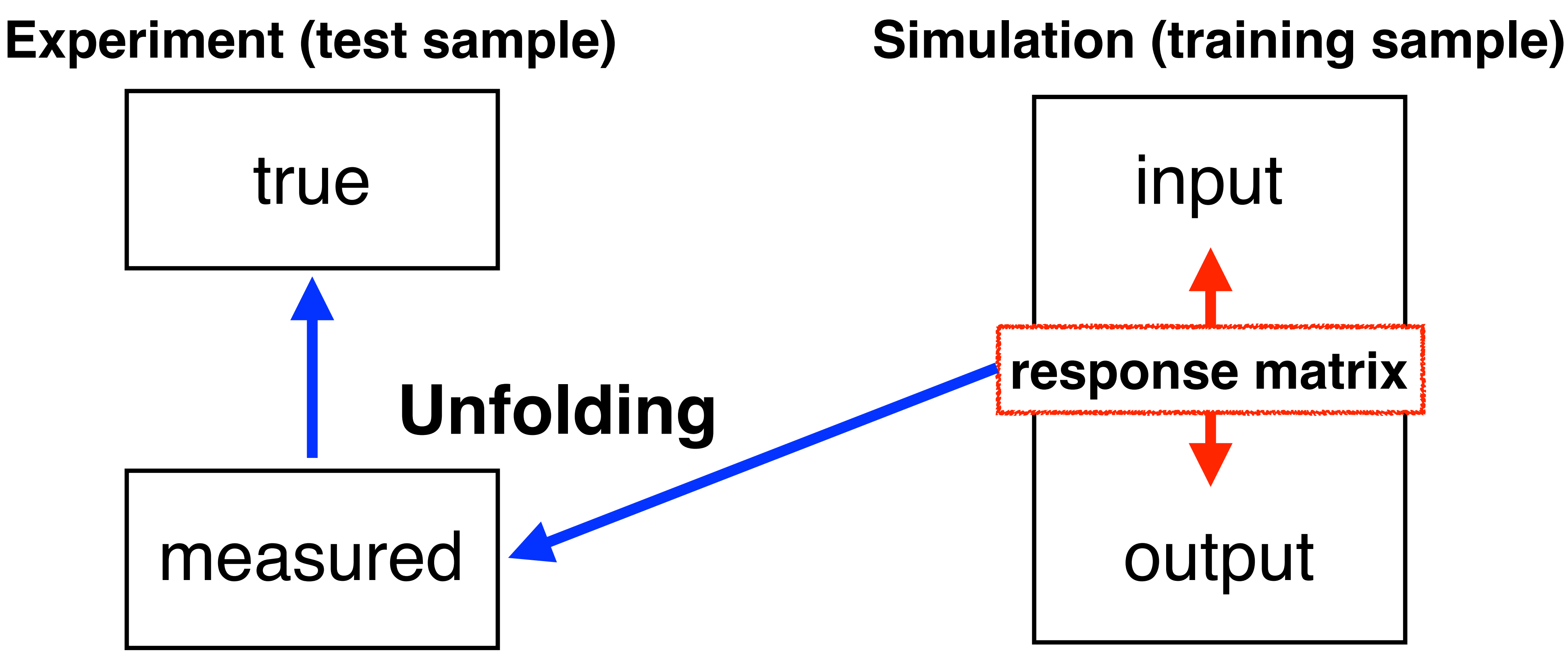
- ✓ Appropriate efficiency bins have to be implemented in efficiency correction, which can be achieved with reasonable CPU time by our new formulas.
- ✓ New unfolding method has been established, which can reproduce the experimental results using efficiency correction.
- ✓ Will test the non-binomial efficiency.

Thank you

Back up

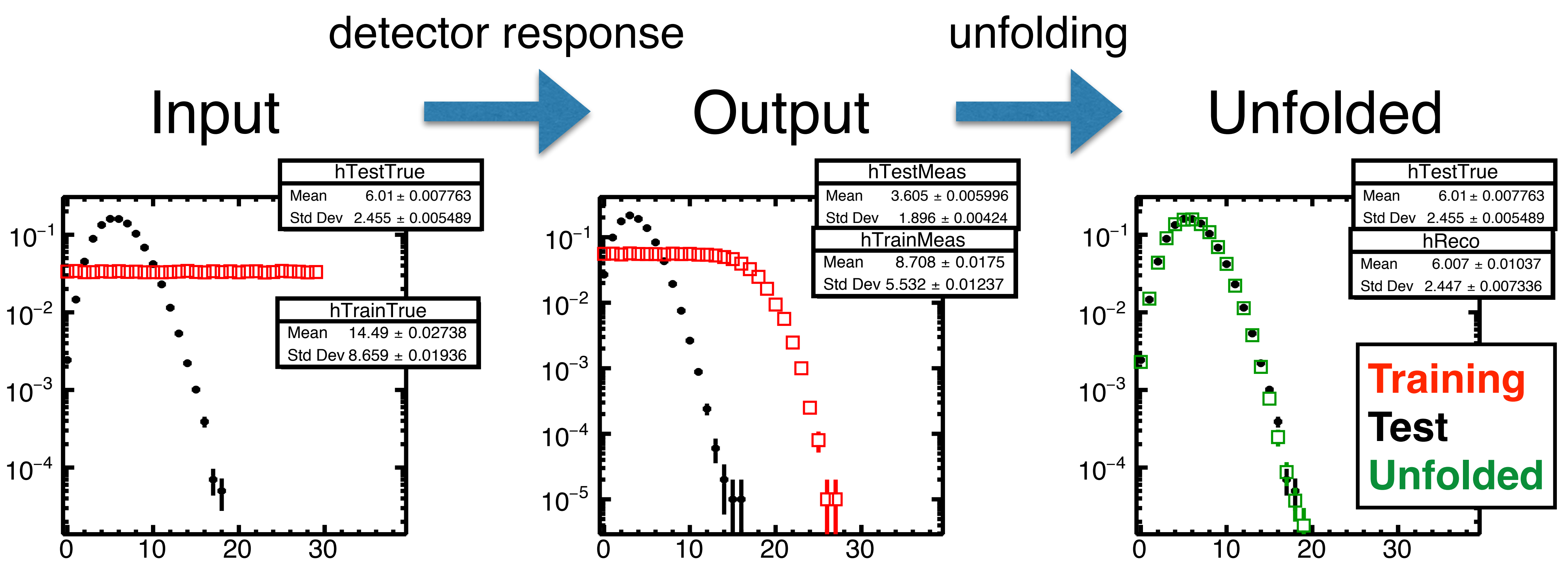
Unfolding : RooUnfold

- “RooUnfold” package in ROOT is commonly used to unfold pT, jet spectrum, etc.
- Can be applied for unknown distribution near the critical point?



Unfolding : RooUnfold

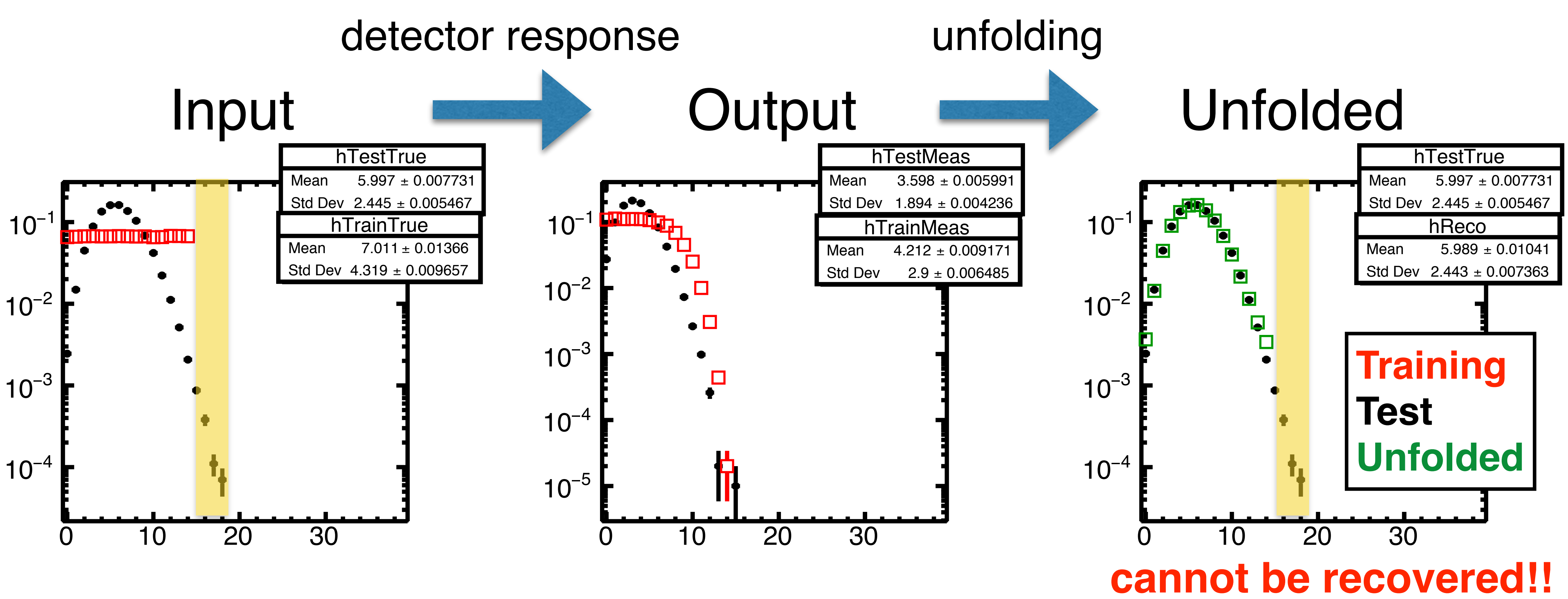
- "RooUnfold" package in ROOT is commonly used to unfold pT, jet spectrum, etc.
- Can be applied for unknown distribution near the critical point?



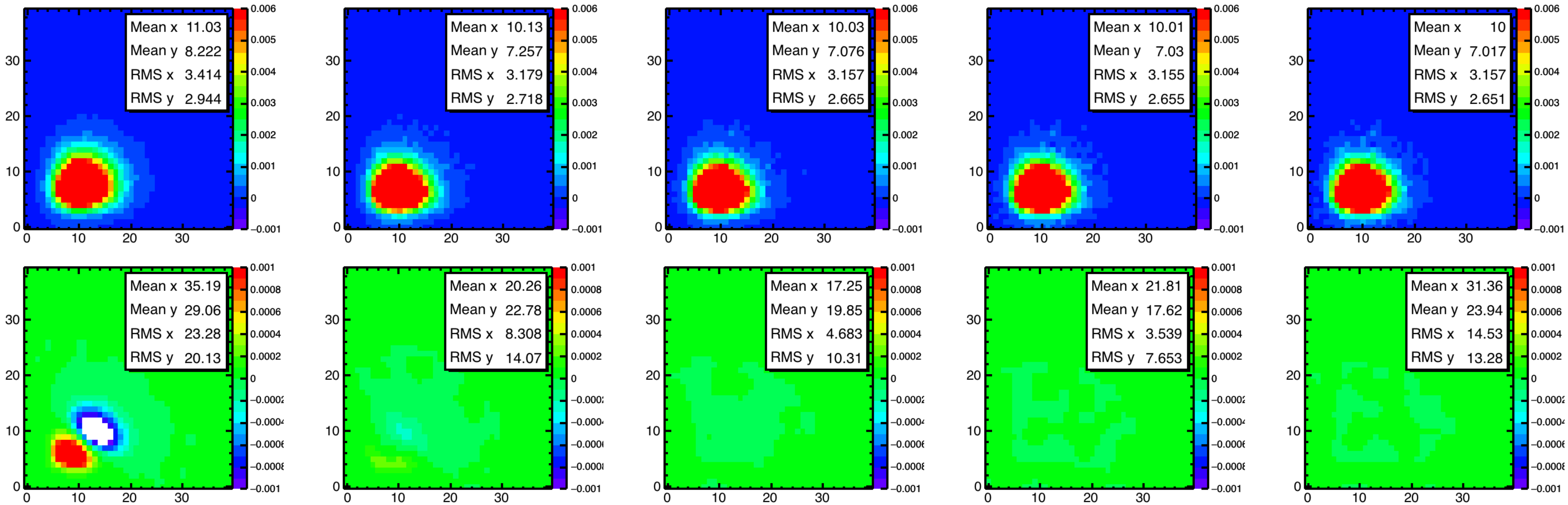
Training sample covers wider bins than test sample

Issue on RooUnfold

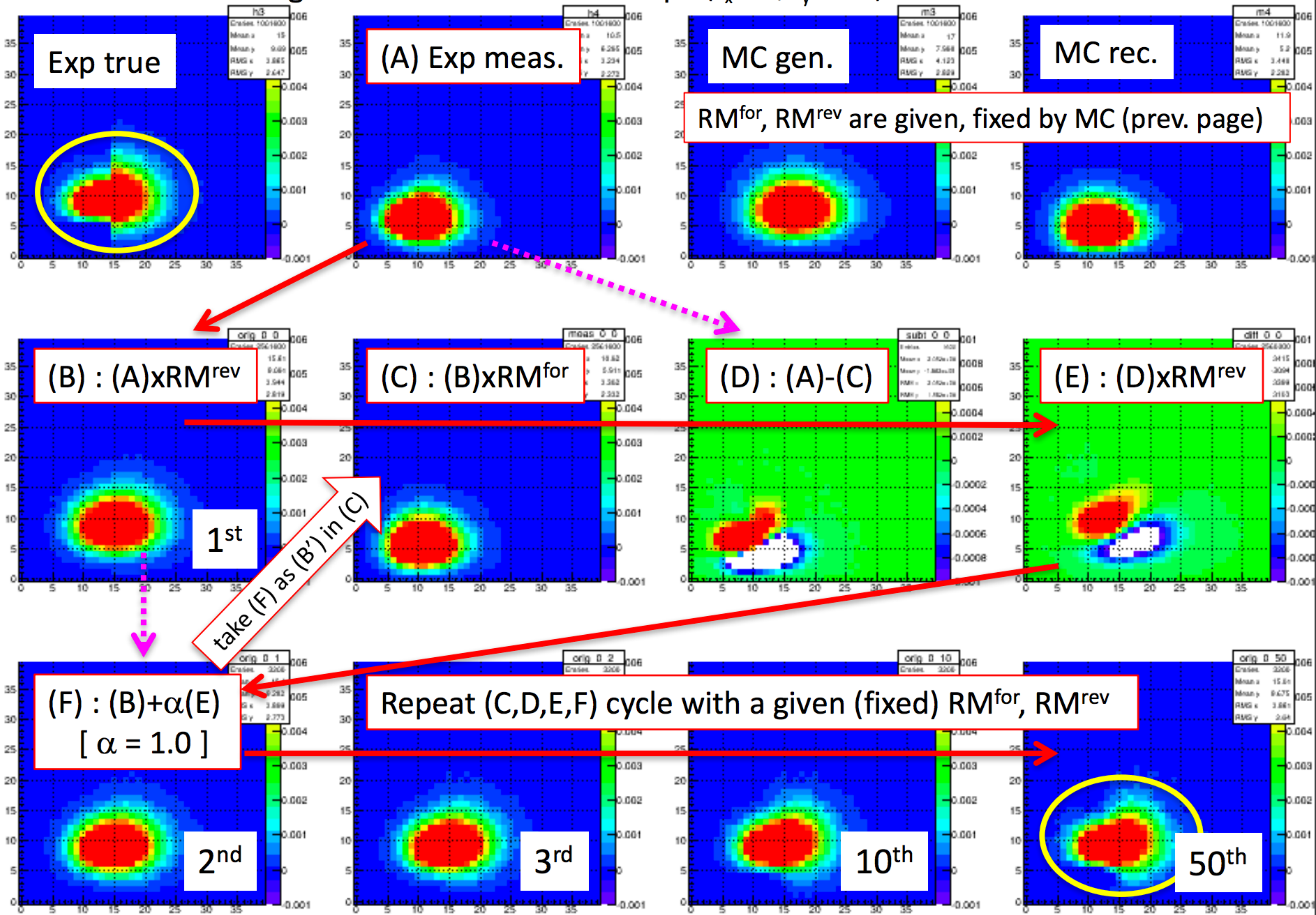
- **Empty bins in training samples cannot be recovered.**
- **We cannot use the conventional unfolding around the critical point.**



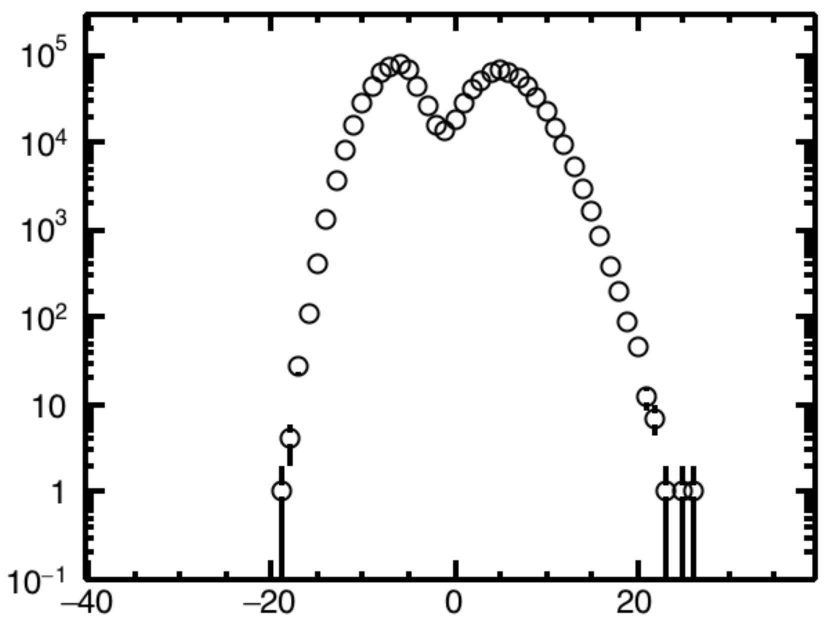
Training sample does not cover all bins in test sample


Mean x 11.03
Mean y 8.222
RMS x 3.414
RMS y 2.944
Mean x 10.13
Mean y 7.257
RMS x 3.179
RMS y 2.718
Mean x 10.03
Mean y 7.076
RMS x 3.157
RMS y 2.665
Mean x 10.01
Mean y 7.03
RMS x 3.155
RMS y 2.655
Mean x 10
Mean y 7.017
RMS x 3.157
RMS y 2.651
Mean x 35.19
Mean y 29.06
RMS x 23.28
RMS y 20.13
Mean x 20.26
Mean y 22.78
RMS x 8.308
RMS y 14.07
Mean x 17.25
Mean y 19.85
RMS x 4.683
RMS y 10.31
Mean x 21.81
Mean y 17.62
RMS x 3.539
RMS y 7.653
Mean x 31.36
Mean y 23.94
RMS x 14.53
RMS y 13.28

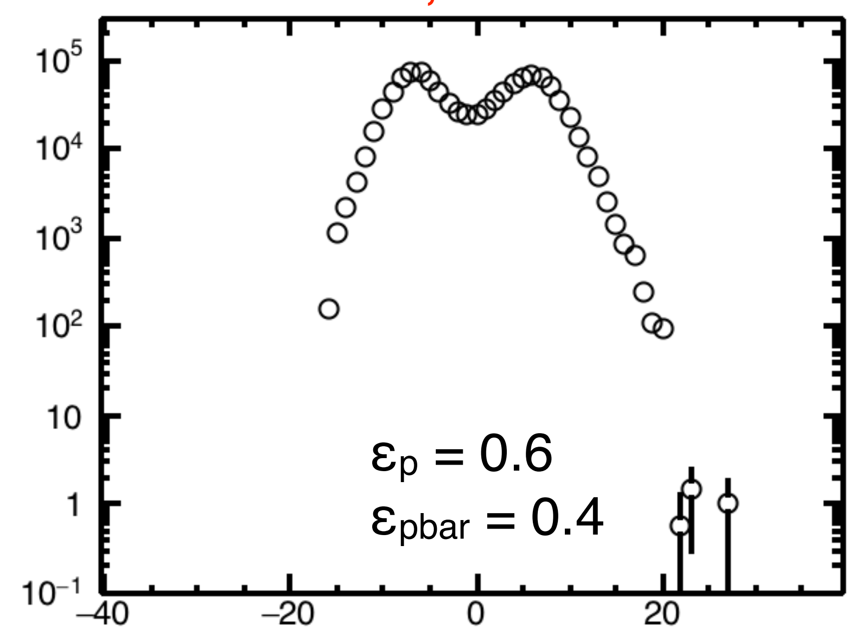
Conventional unfolding method with a critical shape (ε_x=0.7, ε_y=0.65)



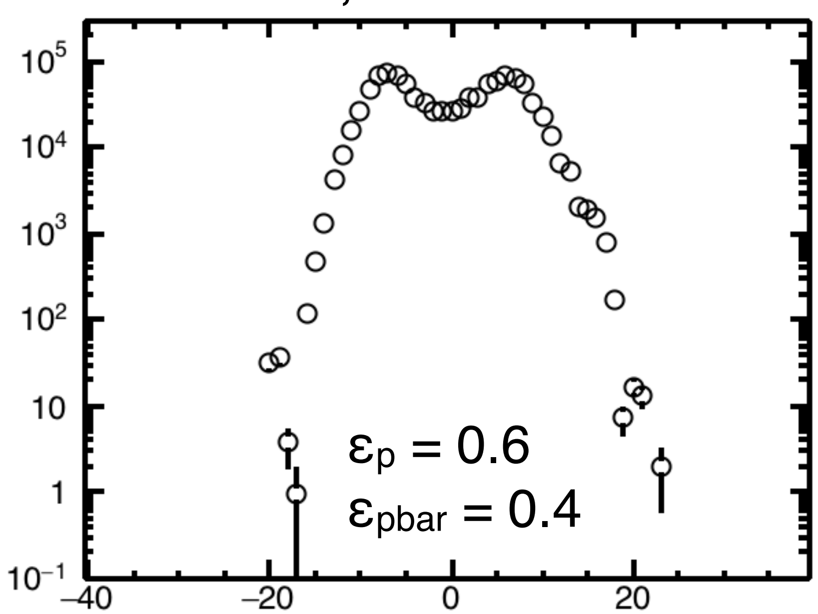
Exp. True 1M events



10M events, 35 inc. + 100 conv.



1M events, 100 inc. + 100 conv.



1M events, 35 inc. + 100 conv.