

RECONSTRUCTING GENERALIZED PARTON DISTRIBUTIONS WITH NEURAL FIELDS: WHAT THE DATA LEAVE FREE

Assemblée Générale 2026 du GDR QCD
October 7-9, Nantes

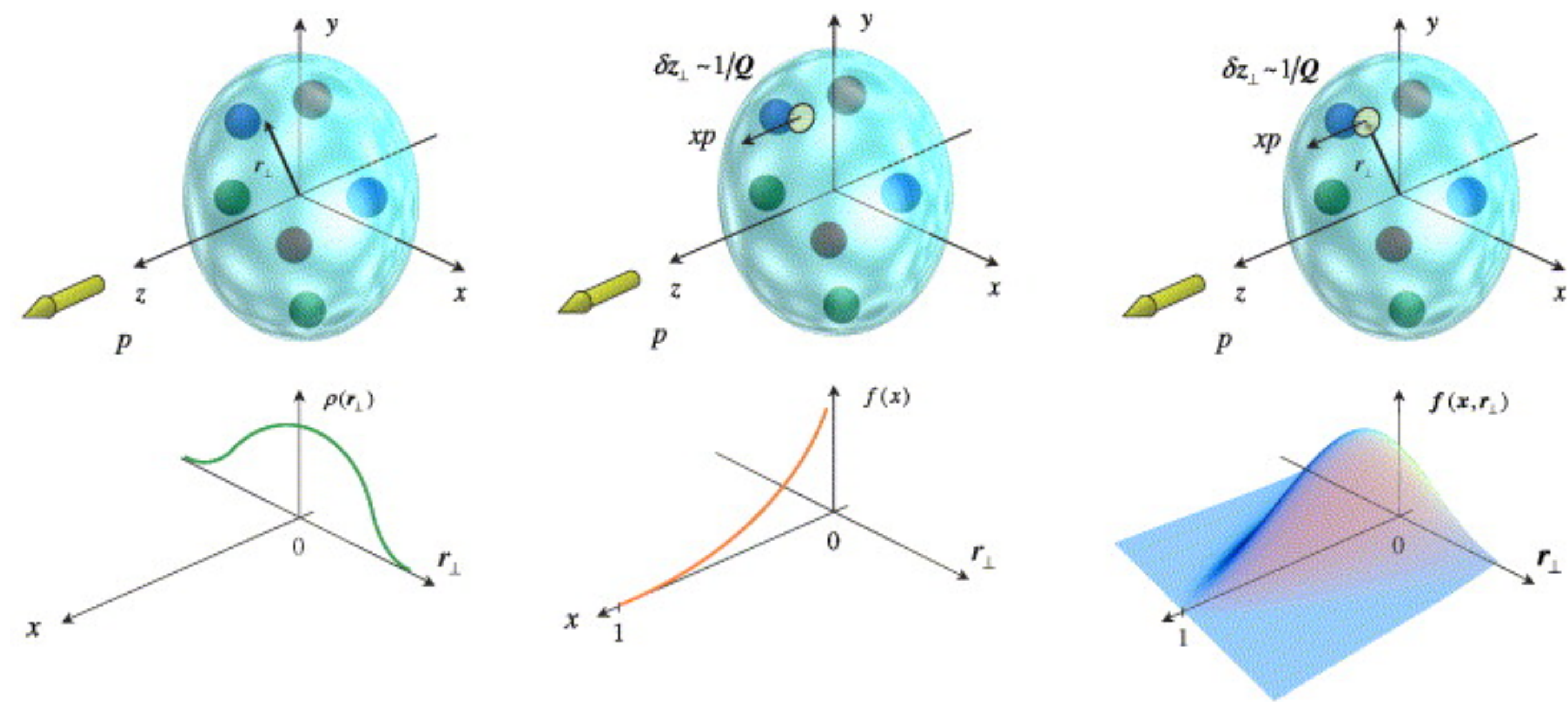
MARIJA ČUIĆ
IN COLLABORATION WITH CÉDRIC MEZRAG AND JÉRÔME BOBIN



HADRON FEMTOGRAPHY

We want to uncover the inner structure of nucleons:

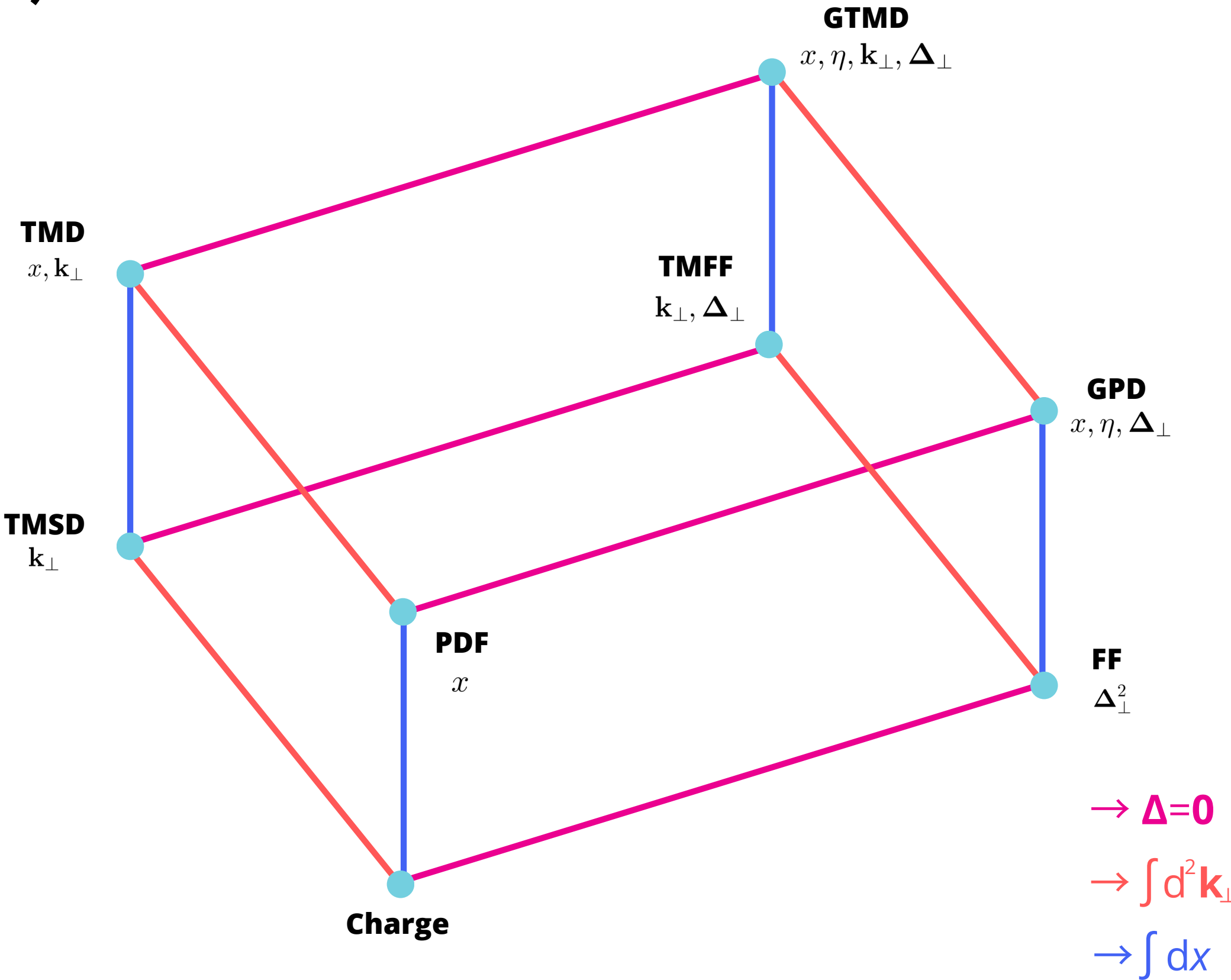
- 3D position and momenta of partons (quarks and gluons)
- spin and angular momentum distribution
- mechanical properties such as pressure



Form factors,
transverse position of
partons

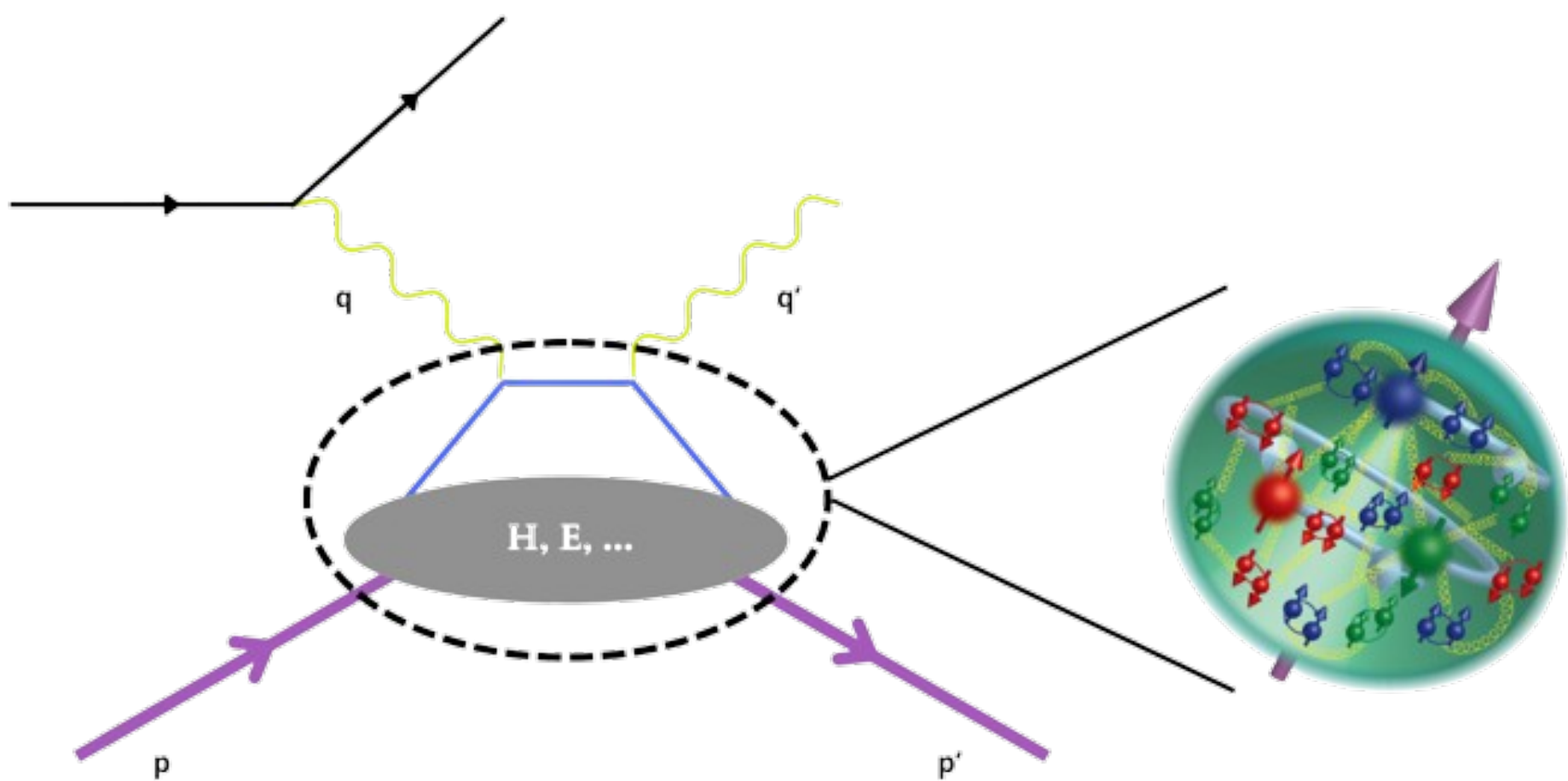
PDFs, longitudinal
momentum of
partons

GPDs, transverse position
and longitudinal momentum
of partons

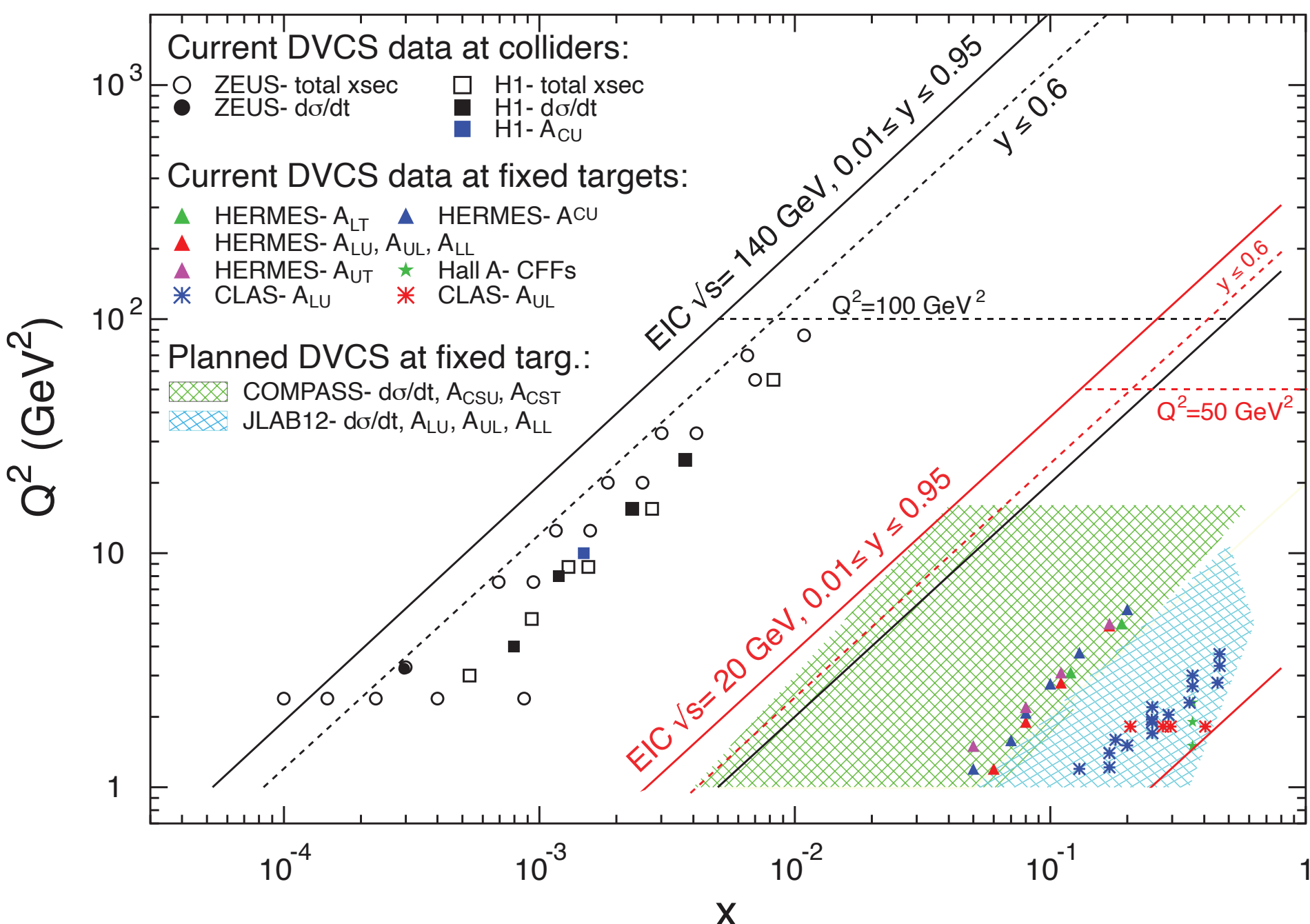


INVERSE PROBLEM

Deeply virtual Compton scattering: $e^-(k_1) + N(P_1) \rightarrow e^-(k_2) + N(P_2) + \gamma(q_2)$



Limited experimental coverage:



$$\frac{d^5\sigma_{DVCS}}{dx_B dy dt d\phi d\varphi} \propto 4(1-x_B) \left(|\mathcal{H}|^2 + |\widetilde{\mathcal{H}}|^2 \right) + \dots$$

$$\mathcal{H}^A(\xi, \Delta^2, Q^2) = \underbrace{\int_{-1}^1 \frac{dx}{2\xi} {}^A T\left(x, \xi \mid \alpha_s(\mu_R), \left\{ \frac{Q^2}{\mu^2} \right\}\right)}_{\text{hard scale}} \underbrace{H^A(x, \xi, \Delta^2, \mu_F^2)}_{\text{soft scale}}$$

Inverse problem: observables $\longrightarrow$ GPDs inside convolution
 $\longrightarrow$ ill-posed problem!

Additionally, experiment dominantly probes the DGLAP region, specifically at LO

$$\Im \mathcal{H}(\xi, t, Q^2) = \pi \sum_{q=u,d,\dots} e_q^2 \left[H^q(x = \xi, \xi, t, \mu_F^2) - H^q(x = -\xi, \xi, t, \mu_F^2) \right].$$

Any extraction beyond that is unreliable. How do we access the ERBL region?

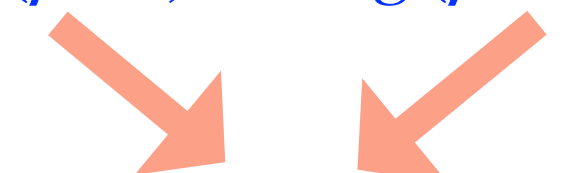
DOUBLE DISTRIBUTION REPRESENTATION OF GPDS

GPD definition:

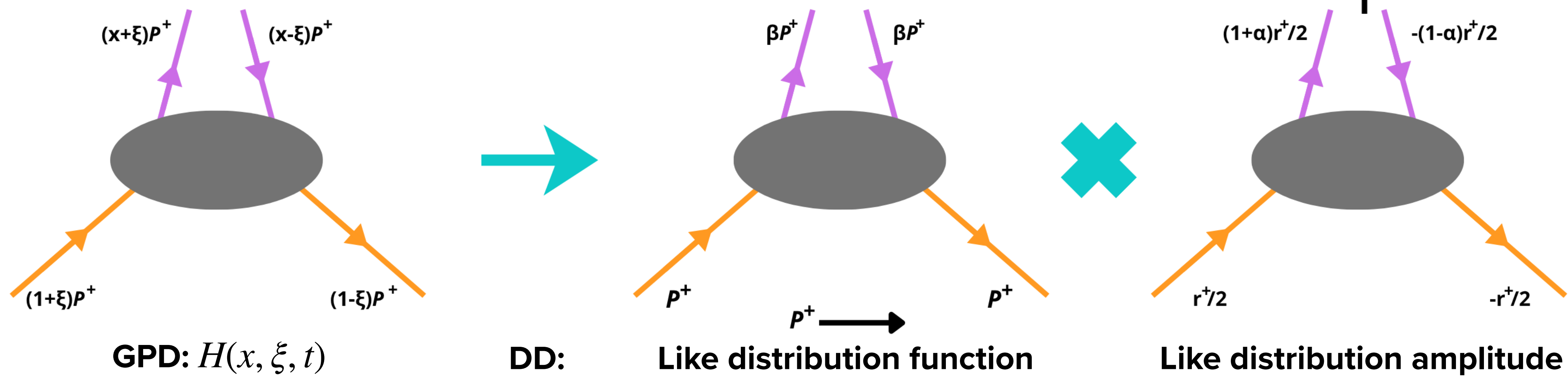
$$F_{\Lambda_2, \Lambda_1}^q = \frac{1}{2} \int \frac{dz^-}{2\pi} e^{ixP^+z^-} \left\langle P_2, \Lambda_2 \left| \bar{\psi}^q \left(-\frac{z^-}{2} \right) \gamma^+ \psi^q \left(\frac{z^-}{2} \right) \right| P_1, \Lambda_1 \right\rangle \Bigg|_{\substack{z_\perp^+ = 0 \\ z_\perp = 0}} = \frac{1}{2P^+} \left[H^q(x, \xi, t) \bar{u}(P_2, \Lambda_2) \gamma^+ u(P_1, \Lambda_1) + E^q(x, \xi, t) \bar{u}(P_2, \Lambda_2) \frac{i\sigma^{+\mu} \Delta_\mu}{2M_H} u(P_1, \Lambda_1) \right]$$

Alternative parametrization of the matrix element:

$$\left\langle P + \frac{\Delta}{2} \left| \bar{\psi}^q \left(-\frac{z^-}{2} \right) \gamma^+ \psi^q \left(\frac{z^-}{2} \right) \right| P - \frac{\Delta}{2} \right\rangle = \iint_{\Omega} d\beta \, d\alpha e^{-i\beta P^+ z^- + i\alpha \frac{1}{2} \Delta^+ z^-} (2P^+ f(\beta, \alpha) - \Delta^+ g(\beta, \alpha))$$


double distributions


Radon transform: $H(x, \xi) = \iint_{\Omega} d\beta \, d\alpha \delta(x - \beta - \xi\alpha) (f(\beta, \alpha) + \xi g(\beta, \alpha))$



RADON TRANSFORM

Support:



Symmetries: $H(x, -\xi, t) = H(x, \xi, t) \Rightarrow f(\beta, -\alpha) = f(\beta, \alpha), \quad g(\beta, -\alpha) = -g(\beta, \alpha)$

We obtain a 2D function by performing integrals over the lines $x - \beta - \xi\alpha = 0$. The $\beta < 0$ describes antiquark distributions, which we do not consider at this point.

Goal: extract GPDs in the DGLAP region, obtain the corresponding double distribution, extend it to the ERBL region, invert the Radon transform and obtain the GPD over the whole domain. Analytically hard to perform, we rely on machine learning. [\[arXiv:2401.12013\]](#)

NEURAL FIELD FOR THE DD

A **NEURAL FIELD** IS A NEURAL NETWORK THAT REPRESENTS A CONTINUOUS PHYSICAL FIELD AS A FUNCTION OF COORDINATES. RATHER THAN STORING DATA ON A DISCRETE GRID, THE NETWORK WEIGHTS ENCODE THE FIELD IMPLICITLY — IT CAN BE EVALUATED AT ANY POINT IN THE DOMAIN. HERE, THE DOUBLE DISTRIBUTION $F(\beta, \alpha)$ IS REPRESENTED AS A NEURAL FIELD: THE NETWORK TAKES (β, α) COORDINATES AS INPUT AND RETURNS THE FIELD VALUE THERE.

Goloskokov-Kroll DD:

$$F(\beta, \alpha) = 2 N_u \beta^{-\alpha_0} (1 - \beta)^{b_u} \frac{\Gamma(2b + 2)}{2^{2b+1} \Gamma(b + 1)^2} \cdot \frac{[(1 - \beta)^2 - \alpha^2]^b}{(1 - \beta)^{2b+1}} \rightarrow F(\beta, \alpha) = \underbrace{\exp\left(\mathbf{Sym}\left(\mathbf{MLP}(\log_{\text{norm}} \beta, \alpha)\right)\right)}_{\text{singularity and positivity}} \cdot \underbrace{\tanh\left(\kappa \cdot [(1 - \beta)^2 - \alpha^2]_+\right)}_{\text{boundary vanishing}} \overbrace{\mathbf{1}[\beta > 0]}^{\text{valence support}}$$

FOURIER FEATURES REPLACE THE RAW INPUT COORDINATES WITH A HIGH-DIMENSIONAL ENCODING $[\sin(2\pi Bx), \cos(2\pi Bx)]$, WHERE B IS A RANDOM MATRIX WITH ENTRIES DRAWN FROM $\mathcal{N}(0, \sigma^2)$. THIS OVERCOMES THE SPECTRAL BIAS OF STANDARD MLPS — THE TENDENCY TO LEARN LOW-FREQUENCY FUNCTIONS FIRST — ALLOWING THE NETWORK TO REPRESENT SHARP, PEAKED FUNCTIONS LIKE THE GK DOUBLE DISTRIBUTION. HERE WE USE 128 RANDOM FFS.

The architecture is physics informed through the loss function:

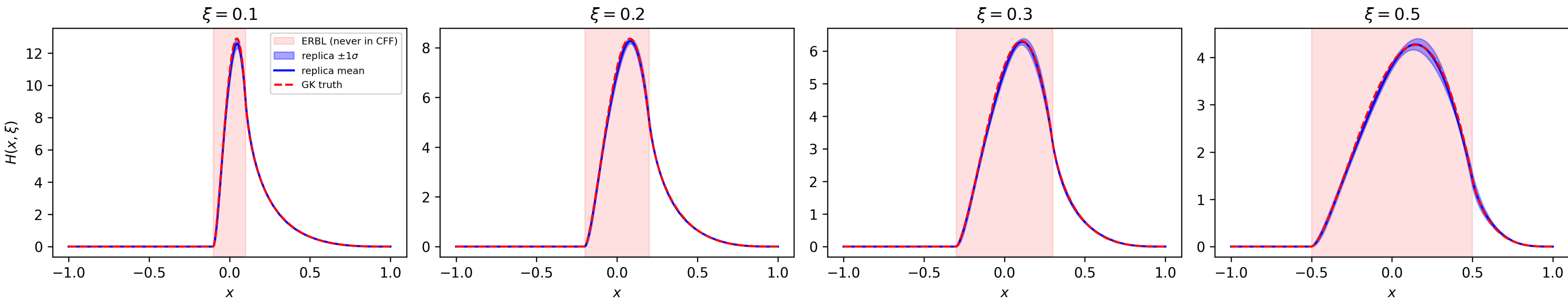
$$\mathcal{L} = \mathcal{L}_{\text{CFF}} + \lambda_{\text{PDF}} \mathcal{L}_{\text{PDF}} + \lambda_{\text{FF}} \mathcal{L}_{\text{FF}}$$

$$\mathcal{L}_{\text{CFF}} = \frac{1}{N_d} \sum_i \left(\frac{\Im \mathcal{H}(x_i, \xi_i) - \Im \mathcal{H}_i^{\text{data}}}{\max(|\Im \mathcal{H}_i^{\text{data}}|, 0.01)} \right)^2, \quad \mathcal{L}_{\text{PDF}} = \frac{1}{N_p} \sum_j \left(\frac{H(x_j, \xi \rightarrow 0) - q(x_j)}{\max(|q(x_j)|, 0.01)} \right)^2, \quad \mathcal{L}_{\text{FF}} = \text{Var}_{\xi}[F_1(\xi)] + 0.1 \left(\langle F_1(\xi) \rangle_{\xi} - F_1^{\text{target}} \right)^2$$

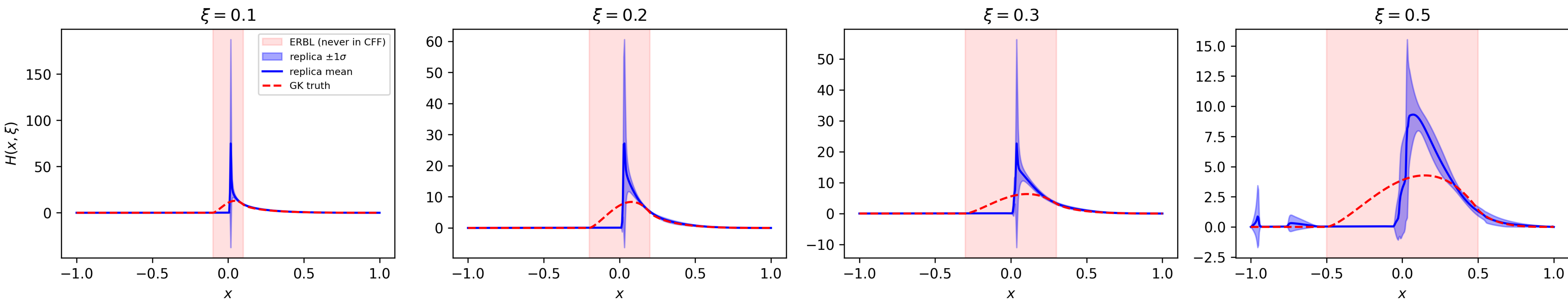
COMPARISON OF TRAINING WITH AND WITHOUT DDS

80 CFF training data points with 20% uncertainty, smoothness penalty on DD (3rd derivative)

GPD with replica uncertainty (10 replicas, 20% CFF error, experimental coverage, arch=dd)

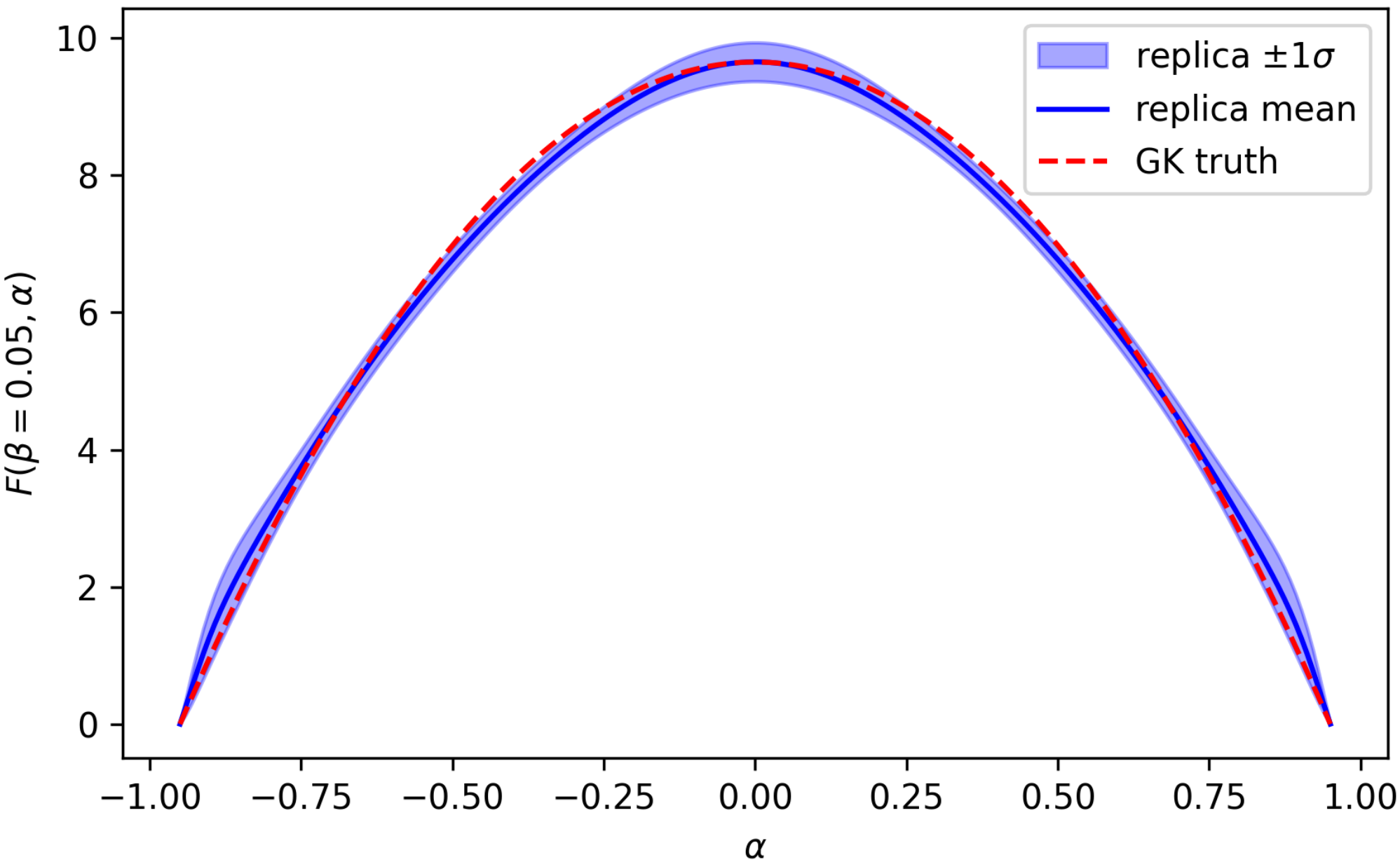


GPD with replica uncertainty (10 replicas, 20% CFF error, experimental coverage, arch=direct)

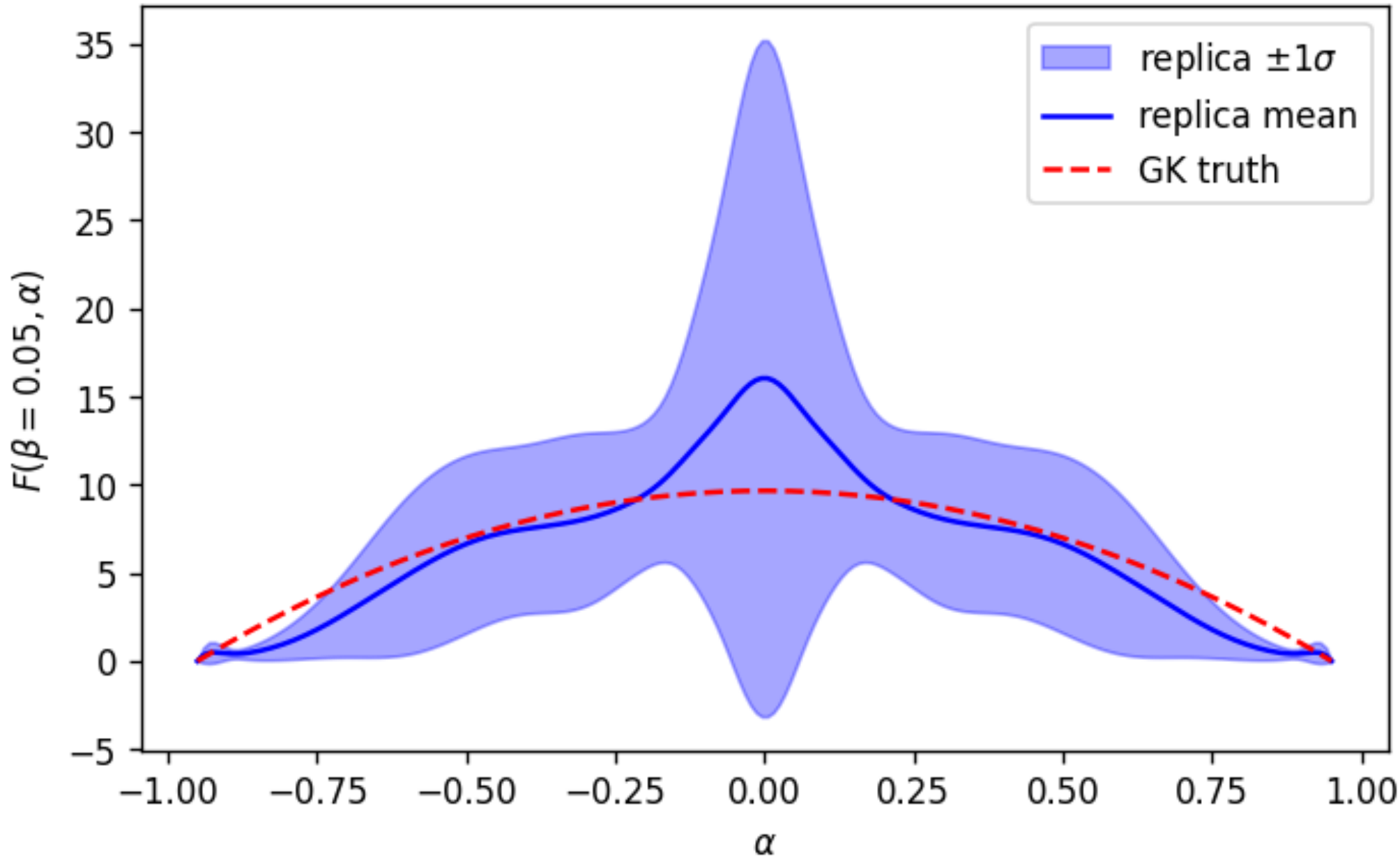


DD WITH AND WITHOUT SMOOTHNESS

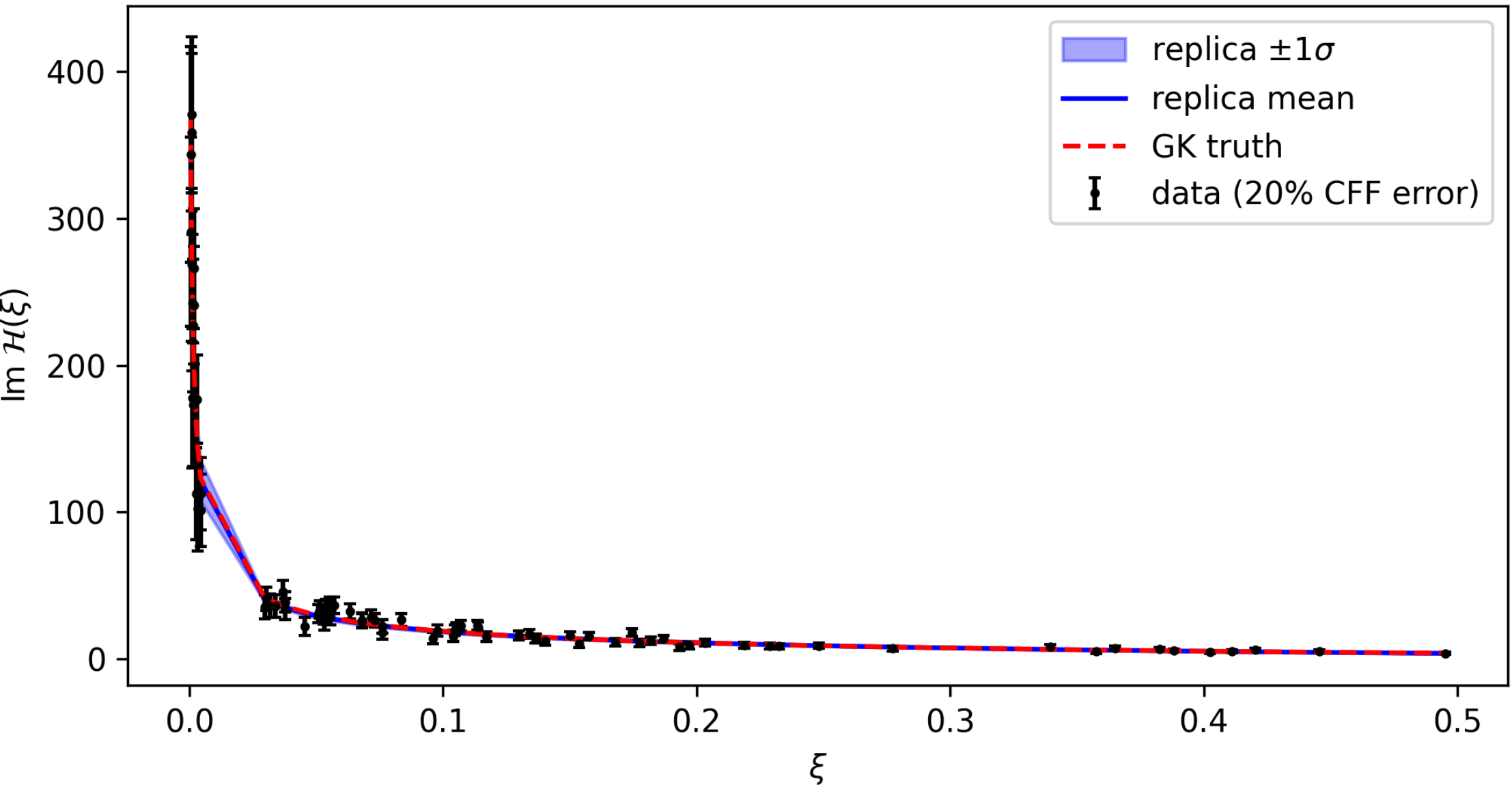
DD alpha-slice with replica band



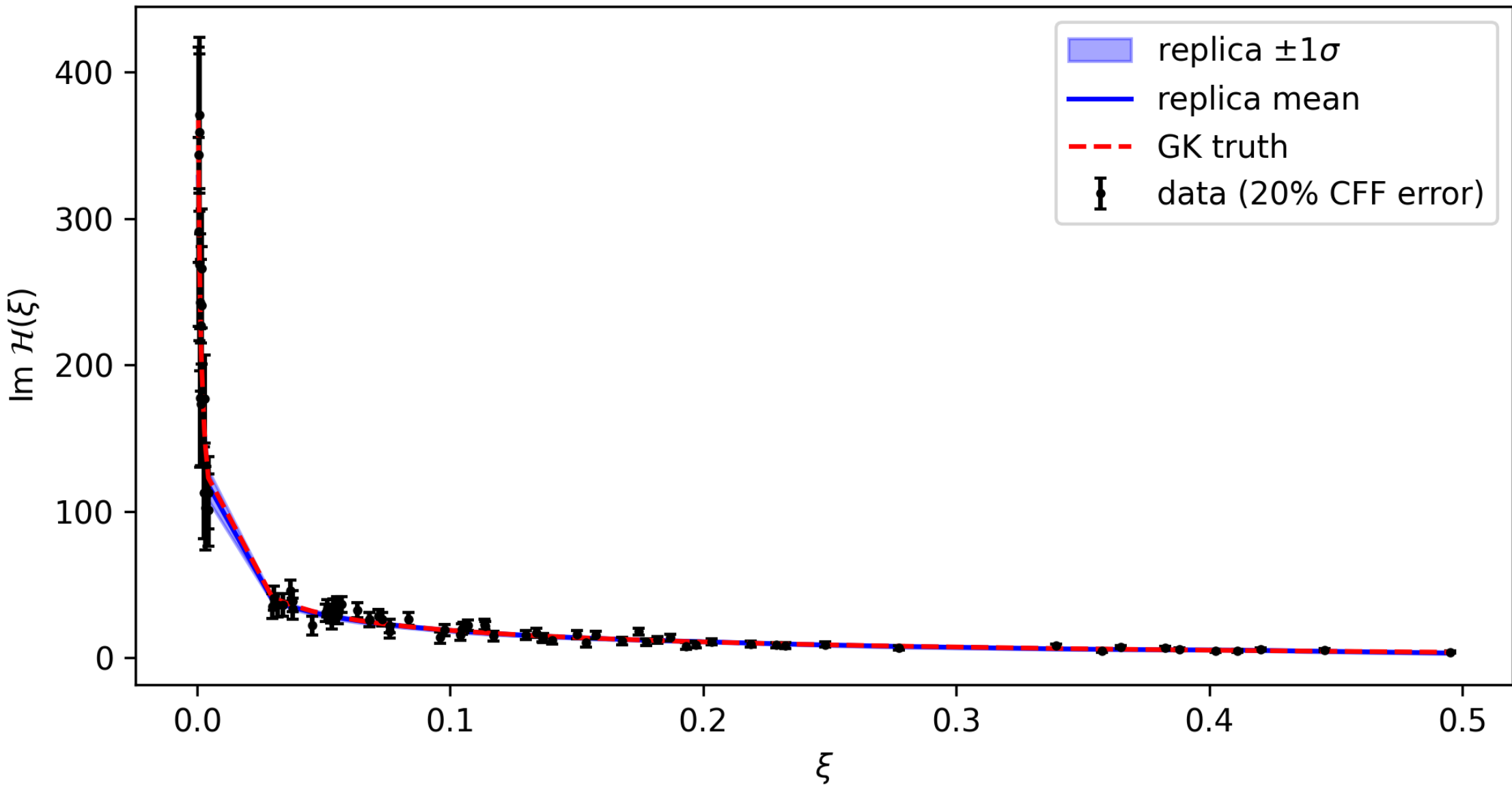
DD alpha-slice with replica band



CFF with replica band (10 replicas, 20% CFF error, arch=dd)



CFF with replica band (10 replicas, 20% CFF error, arch=direct)



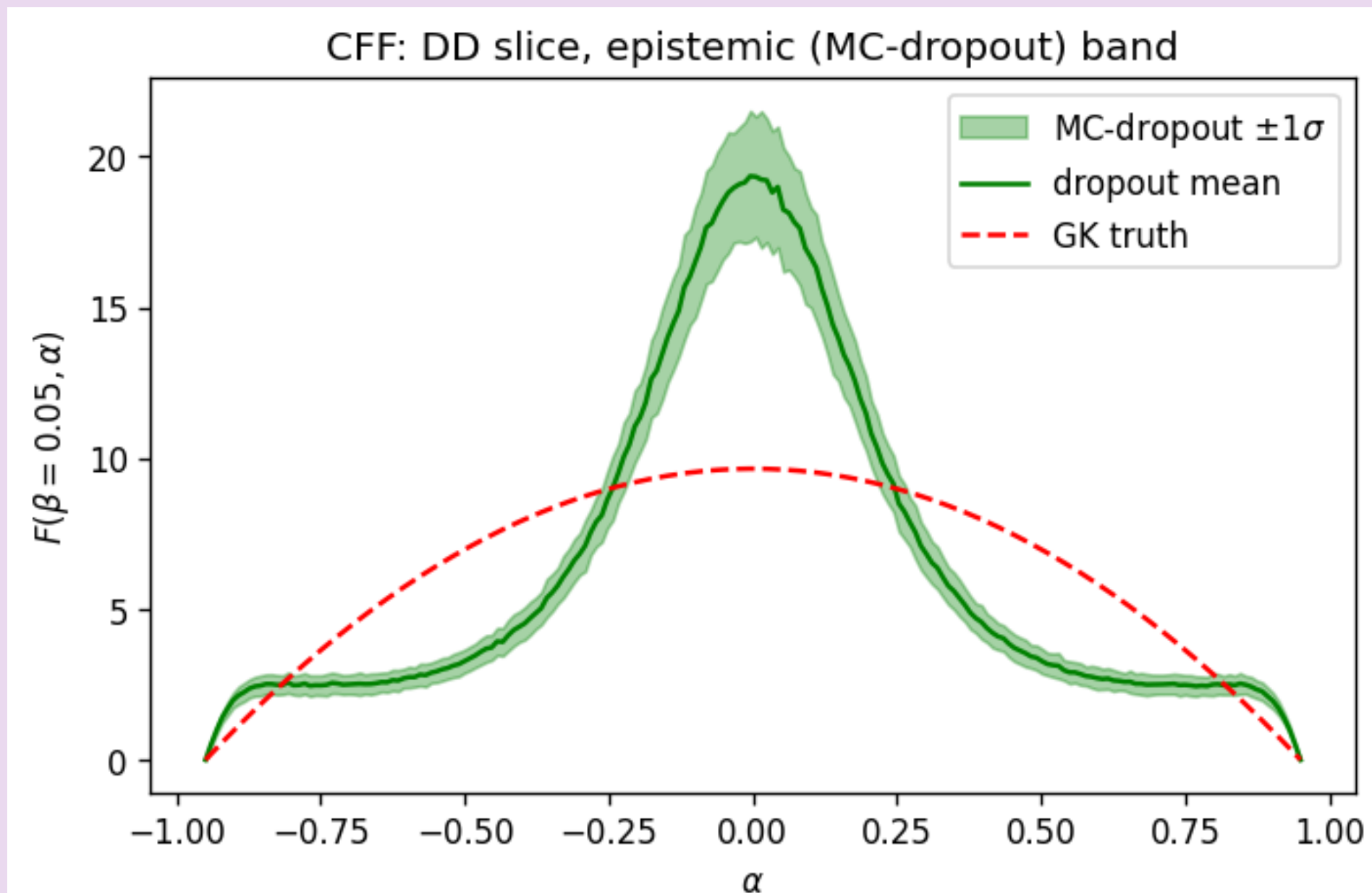
SHADOW MODES

Result of the inverse problem: $F_S^A(x, \xi, t) = F_E^A(x, \xi, t) - F_T^A(x, \xi, t)$. Must satisfy all properties of the GPD and vanish in physical observables:

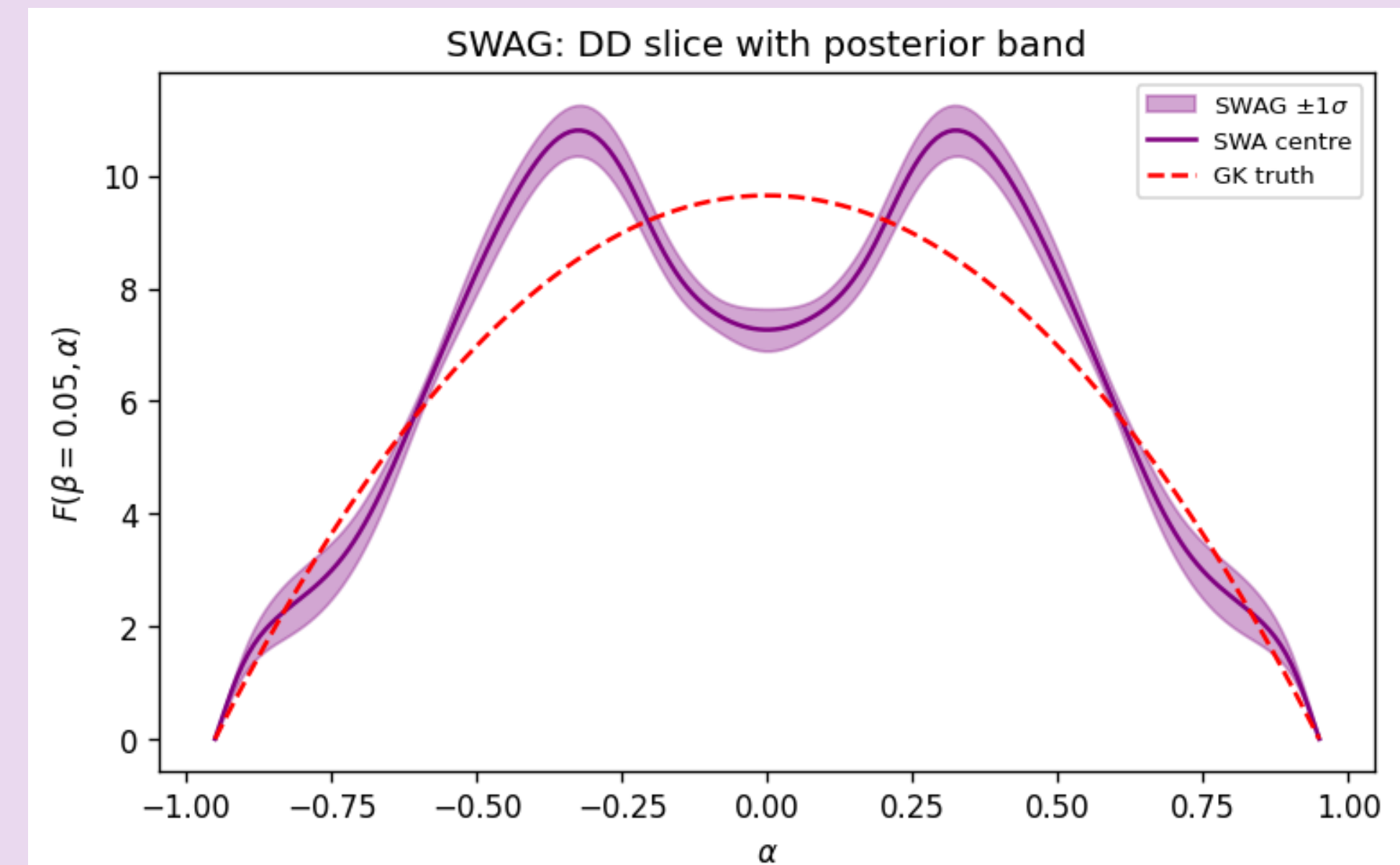
- Polynomiality
- Zero contribution to CFF: $\sum_A {}^A T \left(x, \xi \mid \alpha_s(\mu_R), \left\{ \frac{Q^2}{\mu^2} \right\} \right) \otimes H^A(x, \xi, \Delta^2, \mu_F^2)$
- Zero contribution to PDF: $H_S^A(x, 0, 0) = 0$

UNDERSTANDING THE NULL-SPACE

MC-Dropout: Adds isotropic random noise to network weights during inference; spread across many forward passes = uncertainty band.

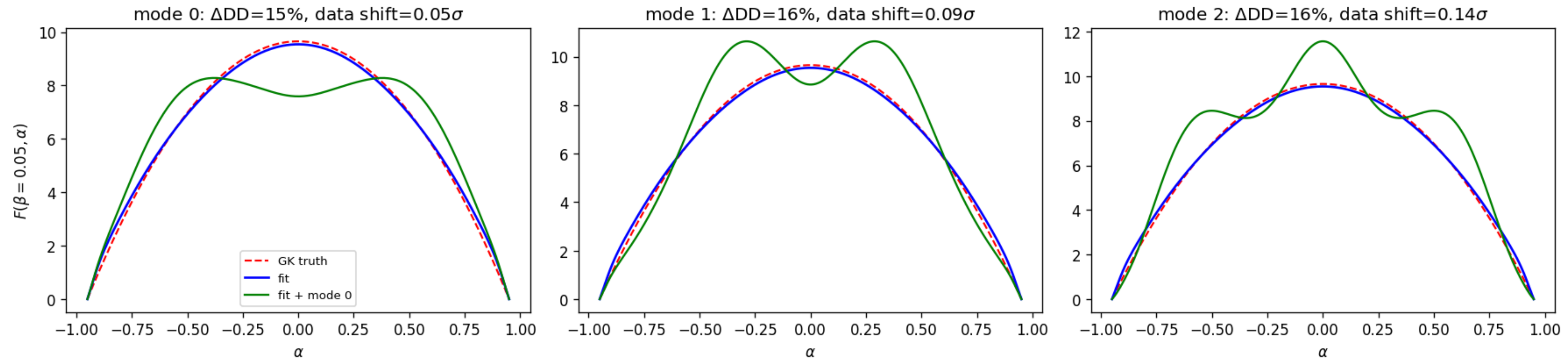


Stochastic Weight Averaging: Continues training at constant learning rate after convergence; fits a Gaussian to the trajectory of visited weights and samples from it. Measures scatter of SGD iterates around one minimum - a property of the *optimizer*, not the *data*.



JACOBIAN OF THE NETWORK

The Jacobian $J_{\text{data}} = \partial \text{Im}H / \partial \theta$ measures how each CFF prediction responds to each network parameter. Its SVD splits parameter space into directions the data constrain and directions they cannot see. Projecting the DD Jacobian $J_{\text{DD}} = \partial F / \partial \theta$ onto that null space isolates the deformations that change the DD while leaving every CFF prediction invariant - the shadow modes.



Each mode is walked outward to $\Delta\chi^2 = 1$, and sampling within that budget gives the uncertainty band.

