24th July 2019

MIAPP programme on
Dynamics of Large-scale Structure Formation

GridSPT: grid-based calculation for perturbation theory of large-scale structure

Atsushi Taruya
(Yukawa Institute for Theoretical Physics)

With

Takahiro Nishimichi (YITP), Donghui Jeong (Penn State)

Plan of talk

Grid-based algorithm for standard perturbation theory of large-scale structure: demonstration & application

Introduction

GridSPT

Demonstration & application

Summary

Based on

AT, Nishimichi & Jeong, PRD98, 103532 ('18) ++

Perturbation theory (PT) calculation of Large-scale structure

Single-stream approximation of cosmological Vlasov-Poisson system

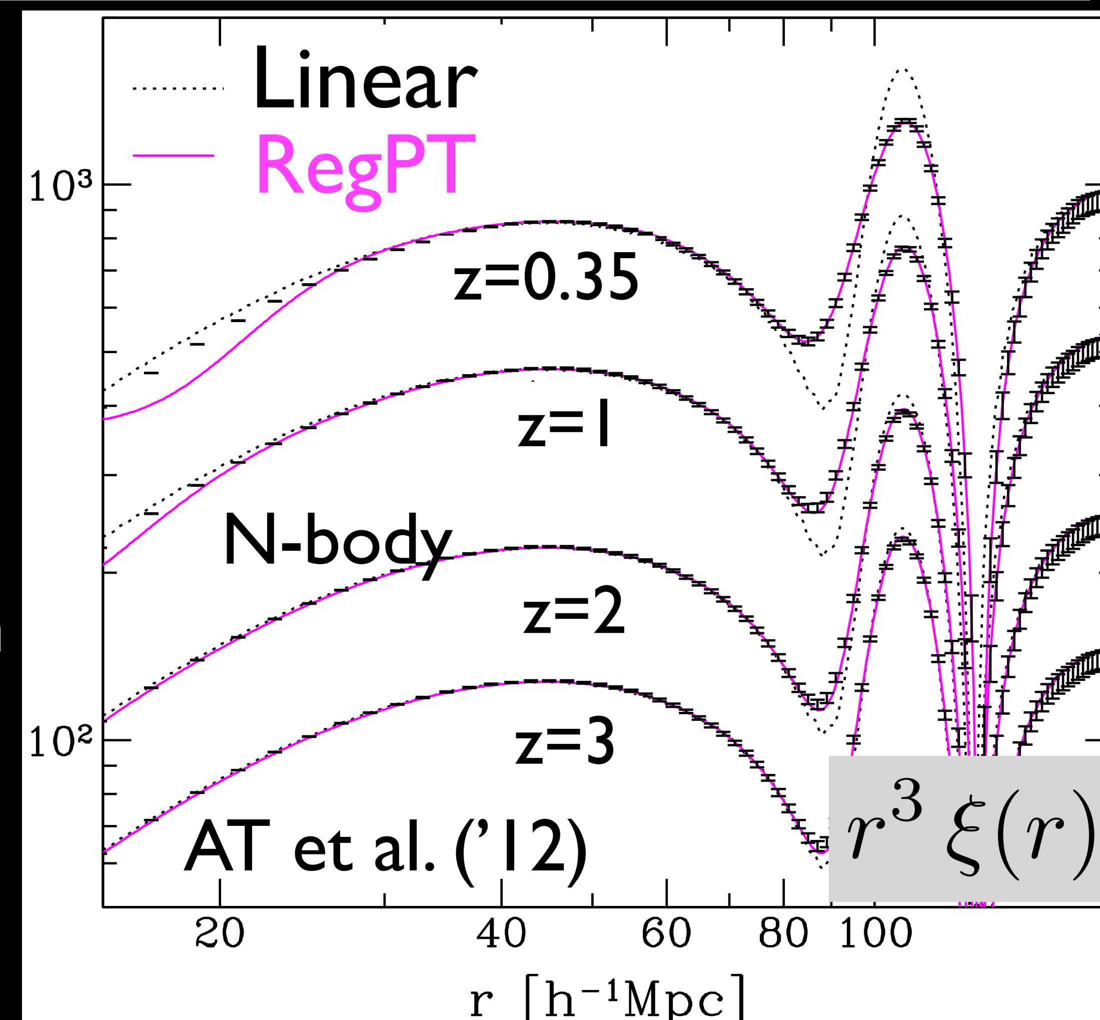
$$\frac{\partial \delta}{\partial t} + \frac{1}{a}\nabla\left[(1+\delta)\boldsymbol{v}\right] = 0,$$

$$\frac{\partial \boldsymbol{v}}{\partial t} + \frac{1}{a}(\boldsymbol{v}\cdot\nabla)\,\boldsymbol{v} = -\frac{1}{a}\frac{\partial \Phi}{\partial \boldsymbol{x}},$$

$$\frac{1}{a^2}\nabla^2\Phi = 4\pi\, G\, \rho_{\rm m}\, \delta$$

Resummation technique

EFT treatment & bias expansion



Small expansion parameter: $|\delta| \ll 1$

$$\delta = \delta^{(1)} + \delta^{(2)} + \delta^{(3)} + \cdots$$

Statical calculation is easy to do analytically

Many realizations are not necessary *(c.f. N-body simulation)*

Grid-based PT calculation ?

Taking observational systematics (survey mask & geometry) into account,

- Mock galaxy catalog construction
- Covariance estimation
- …

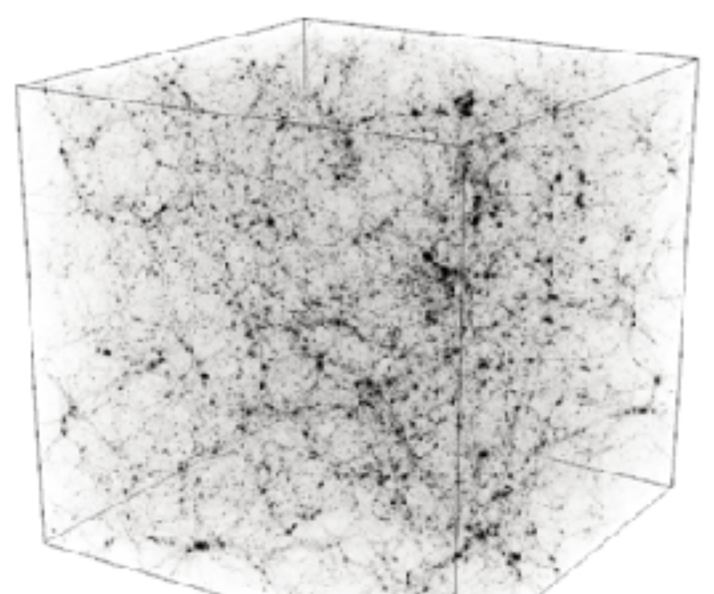
Analytical PT calculation ceases to be tractable

Still, PT approach is useful in constructing mocks:

Peak patch, PThalo, Pinocchio, HALLOGEN, Patchy, EZmock, …

(Bond & Mayers '96; Scoccimarro & Sheth '02; Monaco et al. '02; Avila et al. '15; Kitaura et al. '14,'15; Chuang et al. '15)

… particle-based methods using Lagrangian PT

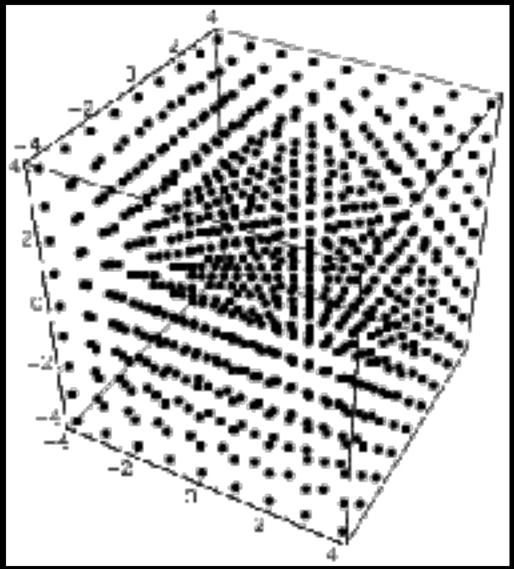


http://www.projet-horizon.fr/

Q Is there grid-based method using standard PT (SPT) approach ? If we can exploit such a method, is it useful ?

Generating standard PT on grids

How can we generate higher-order density fields on grids ?



$$\delta(\boldsymbol{x}) = \delta_1(\boldsymbol{x}) + \delta_2(\boldsymbol{x}) + \delta_3(\boldsymbol{x}) + \cdots$$

In conventional approach,

We first go to Fourier space, and use the recursion formula for SPT kernels, F_n and G_n (e.g., Goroff et al. '86)

$$\delta_n(\boldsymbol{k}) = \int \frac{d^3\boldsymbol{k}_1 \dots d^3\boldsymbol{k}_n}{(2\pi)^{3(n-1)}} \delta_{\rm D}(\boldsymbol{k} - \boldsymbol{k}_{1,\dots,n}) \, F_n(\boldsymbol{k}_1, \dots, \boldsymbol{k}_n) \delta_0(\boldsymbol{k}_1) \cdots \delta_0(\boldsymbol{k}_n)$$

Linear density field

Going to higher-order, this is computationally costly

Grid-based calculation up to 3rd order (Roth & Porciani '11, Tassev'14)

Real-space recursion formula

We can exploit a real-space counterpart of SPT recursion formula

Real-space recursion formula (AT, Nishimichi & Jeong '18, Tobias' talk)

$$\delta(\boldsymbol{x}) = \sum_n (D_+)^n\, \delta_n(\boldsymbol{x}), \quad \theta(\boldsymbol{x}) \equiv -\frac{\nabla\cdot\boldsymbol{v}}{a\,H\,f} = \sum_n (D_+)^n\, \theta_n(\boldsymbol{x})$$

D_+ : Linear growth factor

$$f(a) = \frac{d\ln D_+(a)}{d\ln a}$$

Linear order (n=1)

$$\delta_1(\boldsymbol{x}) = \theta_1(\boldsymbol{x}) = \delta_0(\boldsymbol{x})$$ random field

Higher order (n≥2)

$$\boldsymbol{u} \equiv \nabla(\nabla^{-2}\theta)$$

$$\begin{pmatrix} \delta_n(\boldsymbol{x}) \\ \theta_n(\boldsymbol{x}) \end{pmatrix} = \frac{2}{(2n+3)(n-1)} \begin{pmatrix} n+\frac{1}{2} & 1 \\ \frac{3}{2} & n \end{pmatrix} \sum_{m=1}^{n-1} \begin{pmatrix} (\nabla\delta_m)\cdot\boldsymbol{u}_{n-m} + \delta_m\,\theta_{n-m} \\ [\partial_j(\boldsymbol{u}_m)_k][\partial_k(\boldsymbol{u}_{n-m})_j] + \boldsymbol{u}_m\cdot(\nabla\theta_{n-m}) \end{pmatrix}$$

➡ Making use of FFT, RHS can be evaluated quickly

C++ code: *GridSPT* *(will be made public)*

GridSPT: projected density field

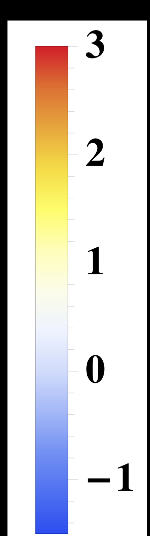
AT, Nishimichi & Jeong ('18)

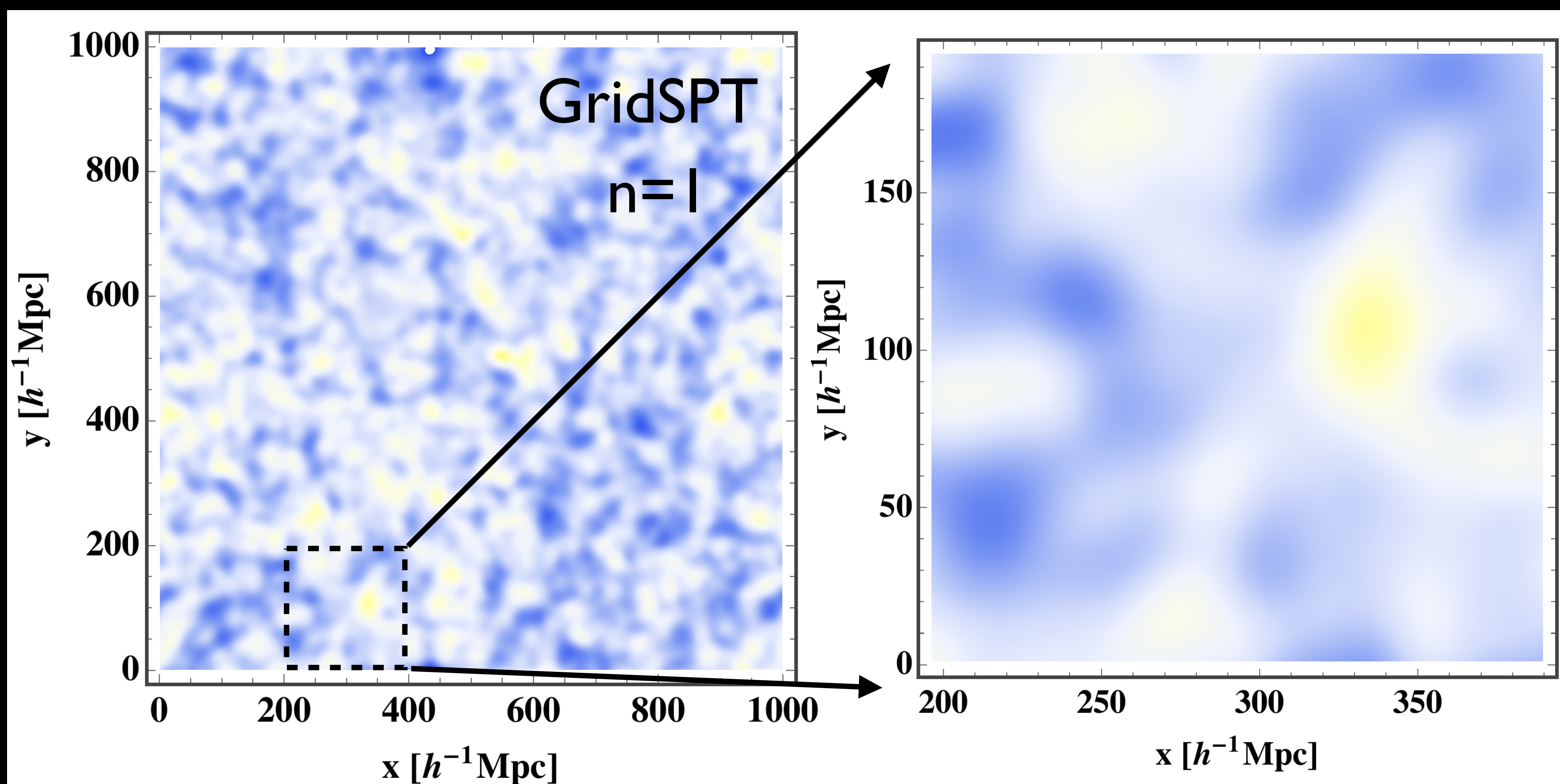
Gaussian filter of R=10 h^{-1}Mpc (depth: h^{-1}Mpc)

GridSPT density field $\delta_1(\boldsymbol{x})$

$N_{grid}=512^3$
$L_{box}=1{,}000\ h^{-1}$Mpc

z=0





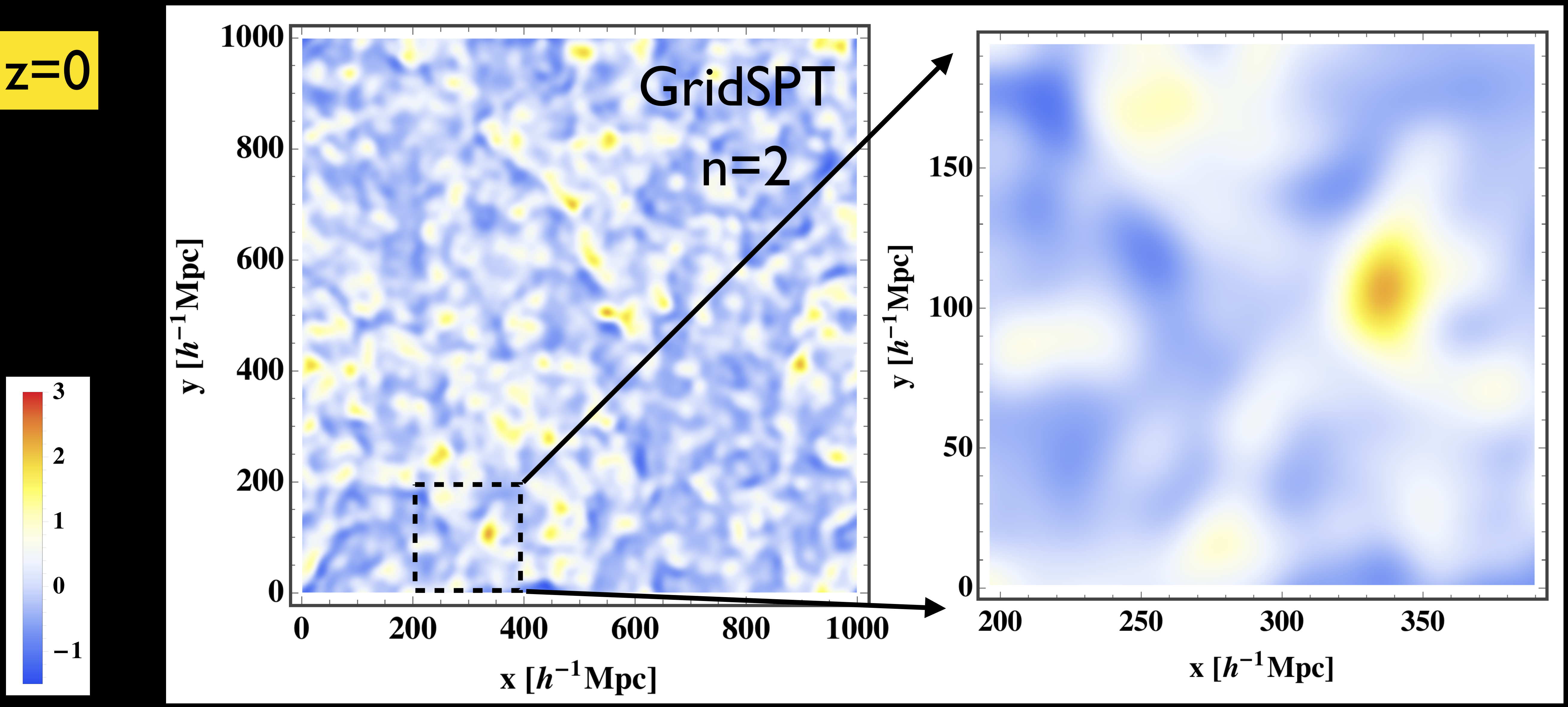
GridSPT: projected density field

AT, Nishimichi & Jeong ('18)

Gaussian filter of R=10 h^{-1}Mpc (depth: h^{-1}Mpc)

GridSPT density field $\delta_1(\boldsymbol{x}) + \delta_2(\boldsymbol{x})$

z=0



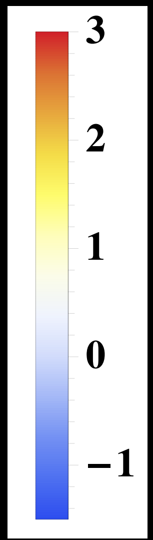
GridSPT: projected density field

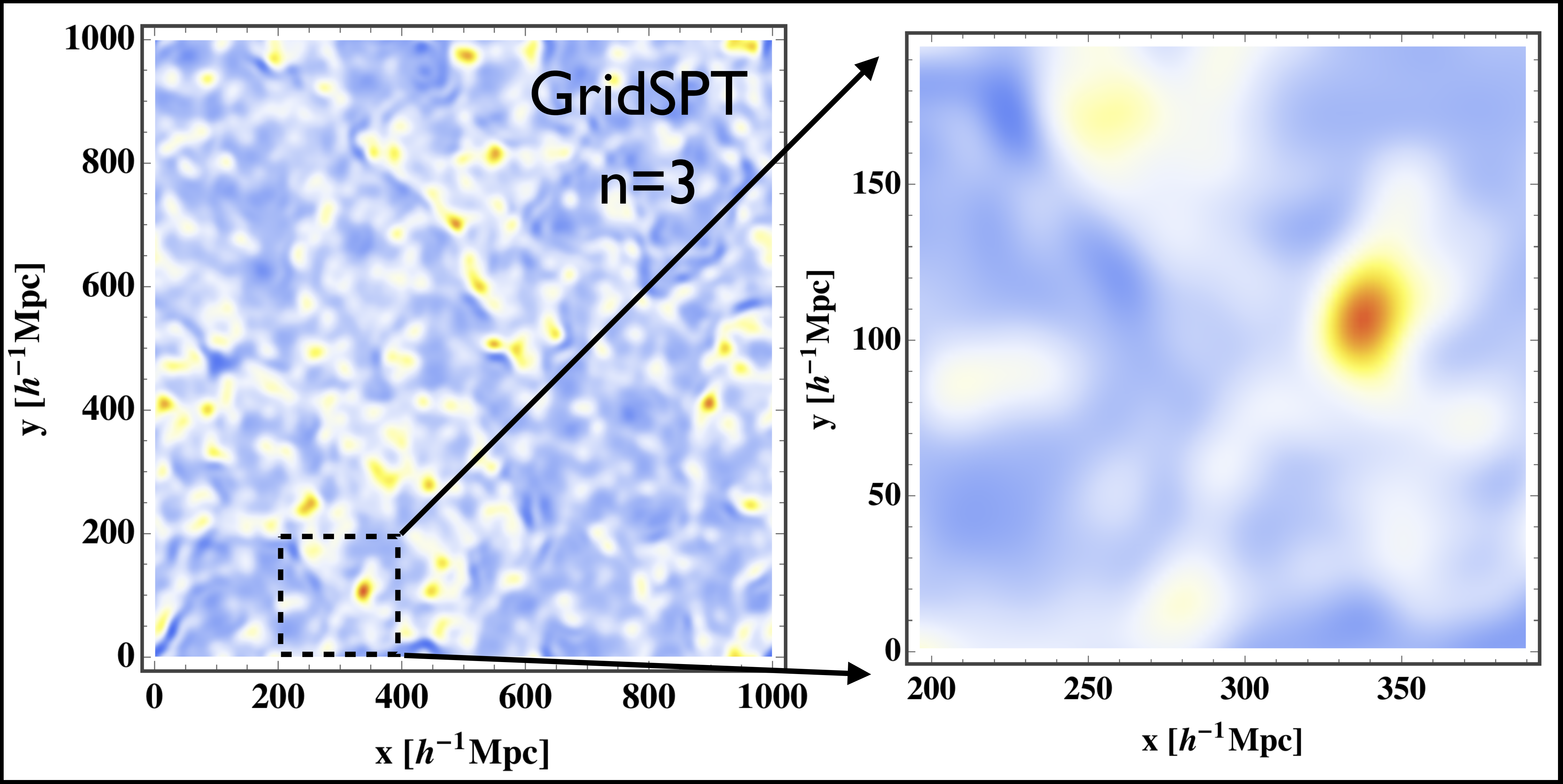
AT, Nishimichi & Jeong ('18)

Gaussian filter of R=10 h^{-1}Mpc (depth: h^{-1}Mpc)

GridSPT density field $\delta_1(\boldsymbol{x}) + \delta_2(\boldsymbol{x}) + \delta_3(\boldsymbol{x})$

z=0





GridSPT: projected density field

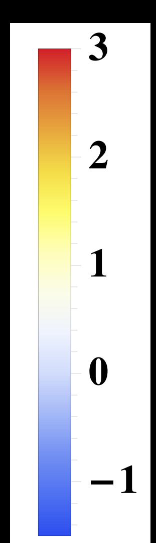
AT, Nishimichi & Jeong ('18)

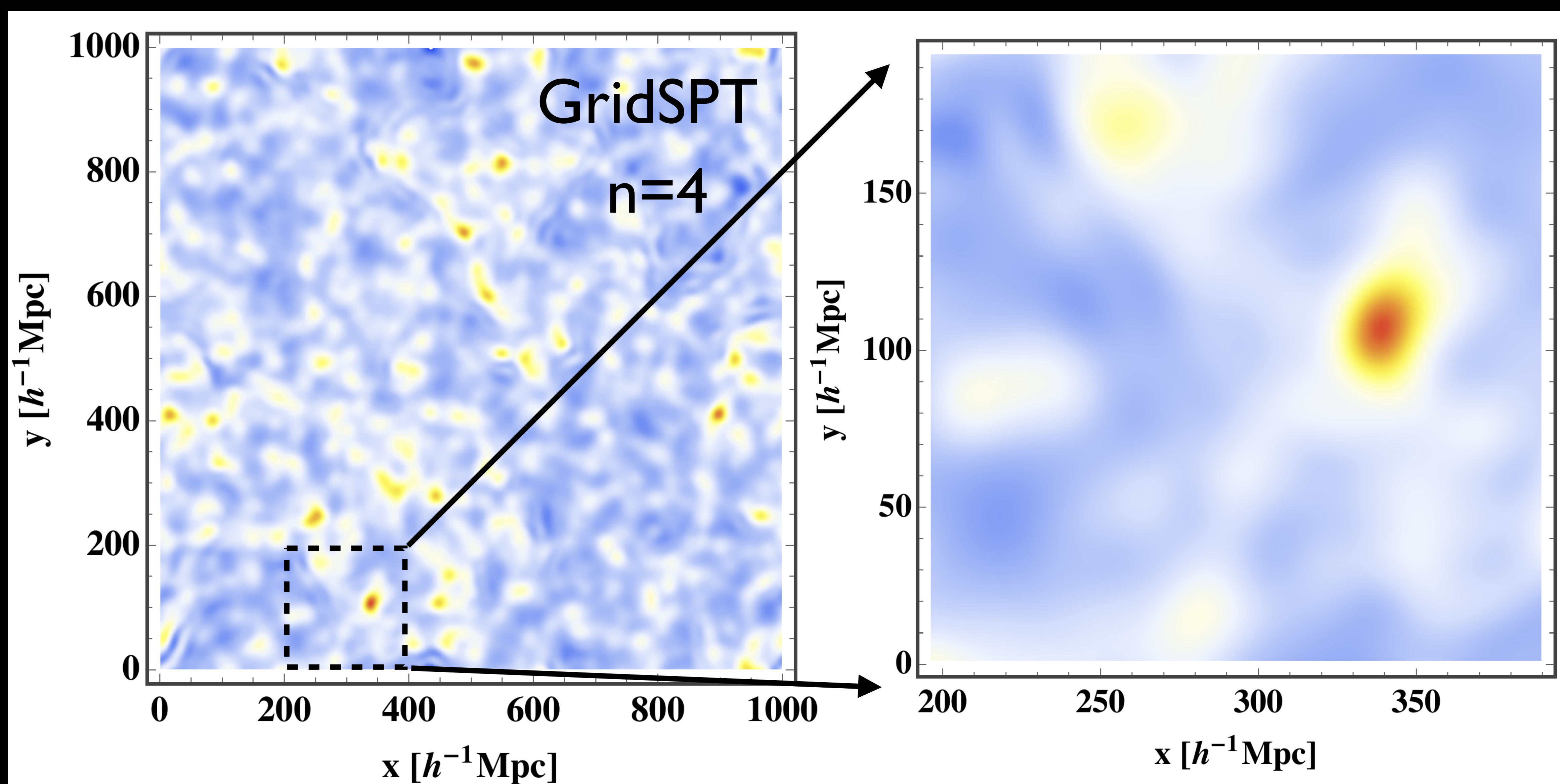
Gaussian filter of R=10 h-1Mpc (depth: h-1Mpc)

GridSPT density field

$$\delta_1(\boldsymbol{x}) + \delta_2(\boldsymbol{x}) + \delta_3(\boldsymbol{x}) + \delta_4(\boldsymbol{x})$$

z=0





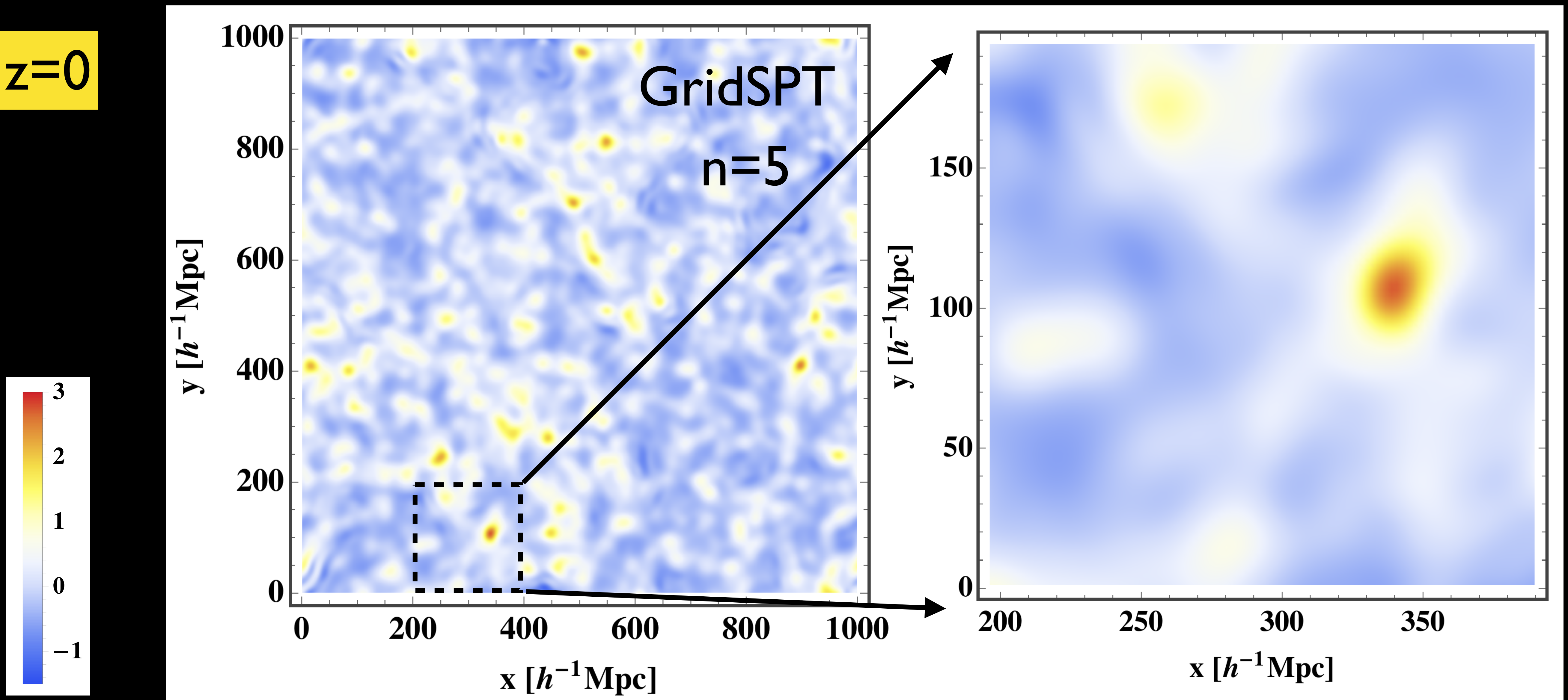
GridSPT: projected density field

AT, Nishimichi & Jeong ('18)

Gaussian filter of R=10 h^{-1}Mpc (depth: h^{-1}Mpc)

GridSPT density field $\delta_1(\boldsymbol{x}) + \delta_2(\boldsymbol{x}) + \delta_3(\boldsymbol{x}) + \delta_4(\boldsymbol{x}) + \delta_5(\boldsymbol{x})$

z=0



N-body: projected density field

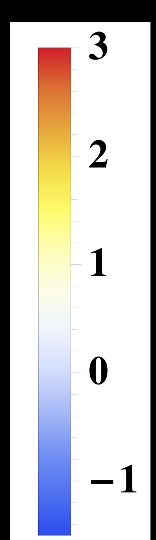
AT, Nishimichi & Jeong ('18)

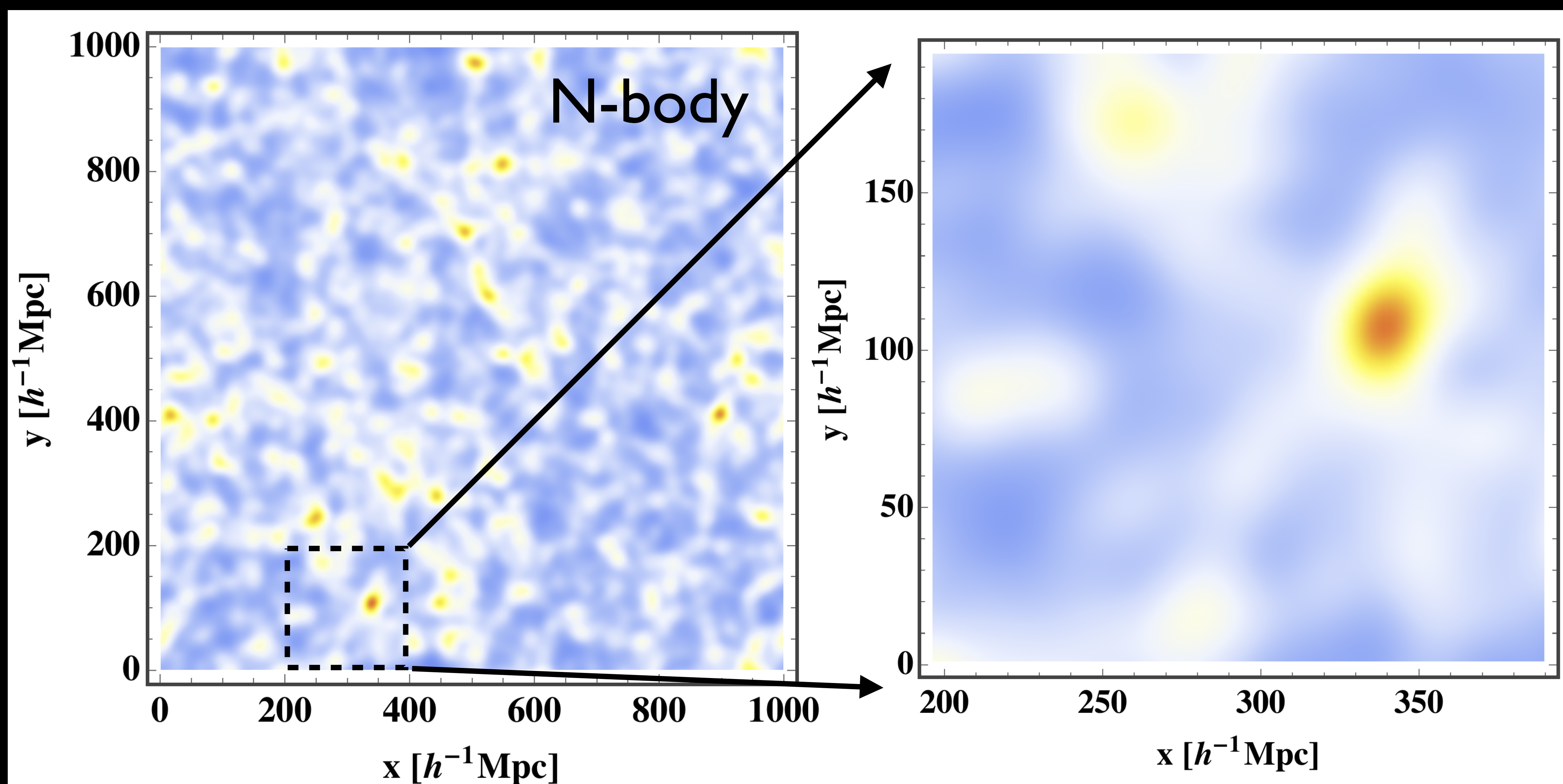
Gaussian filter of R=10 h^{-1}Mpc (depth: h^{-1}Mpc)

$N_{particle}$=1,024^3
L_{box}=1,000 h^{-1}Mpc

N-body density field $\delta_{\mathrm{N-body}}(\boldsymbol{x})$

z=0

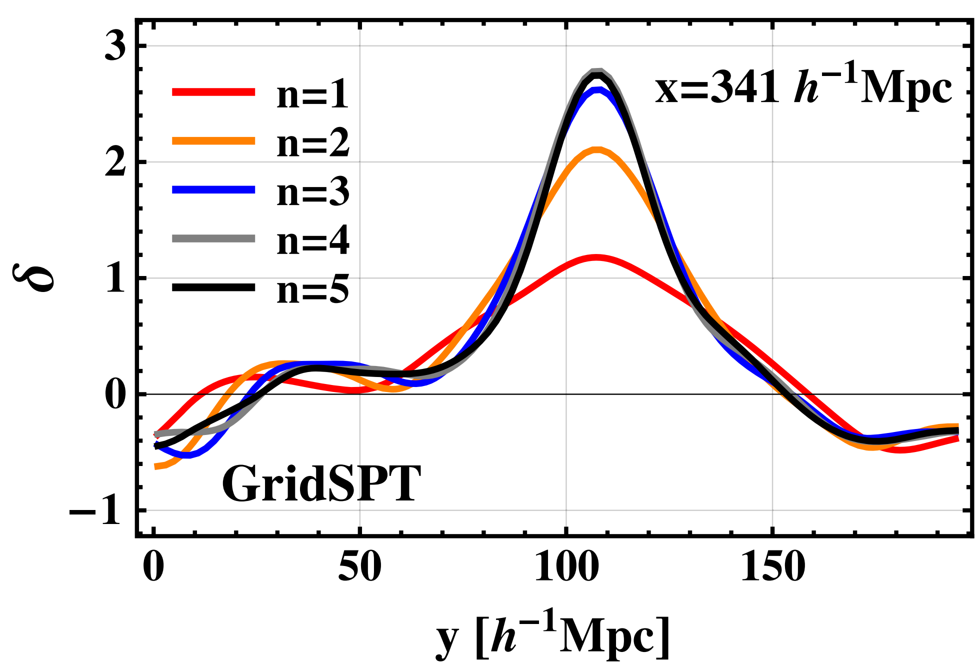


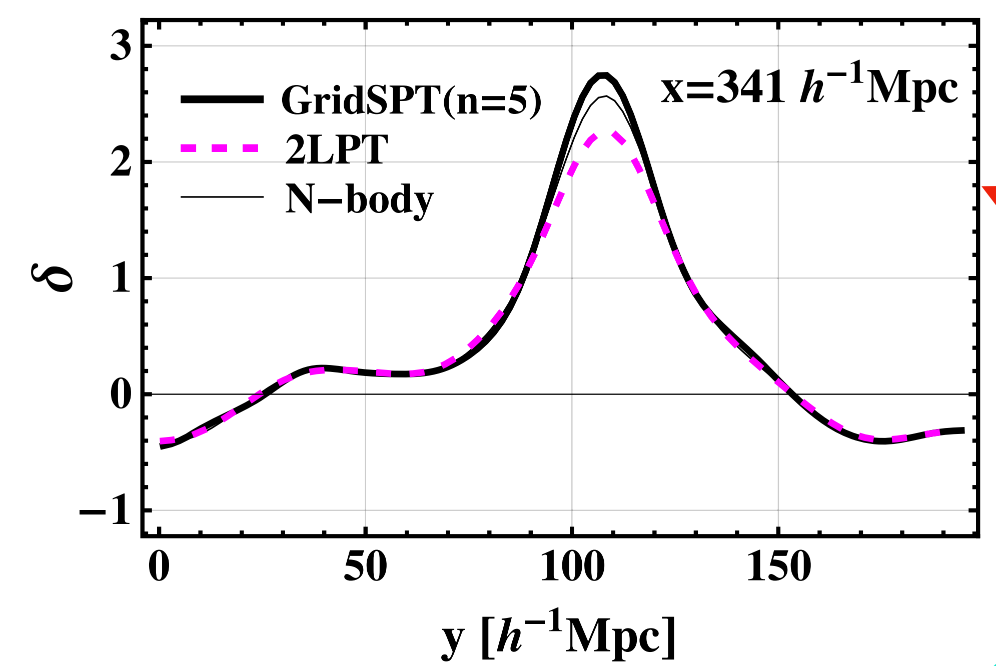


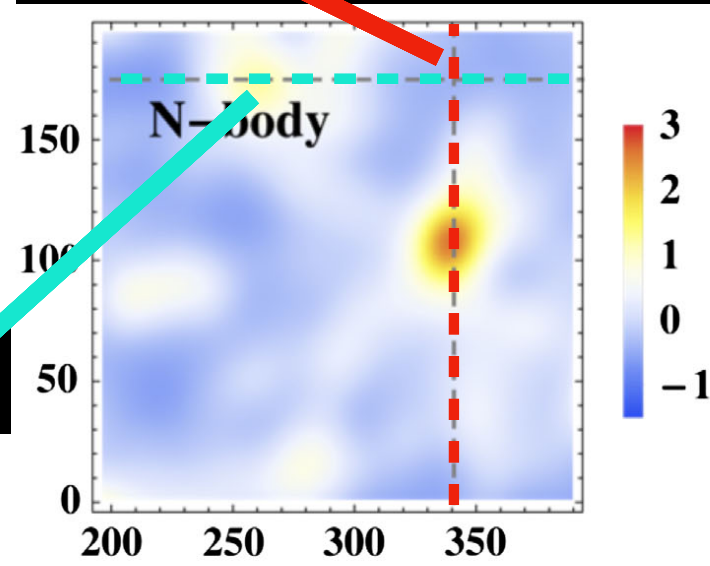
GridSPT vs N-body: 1D slice

AT, Nishimichi & Jeong ('18)

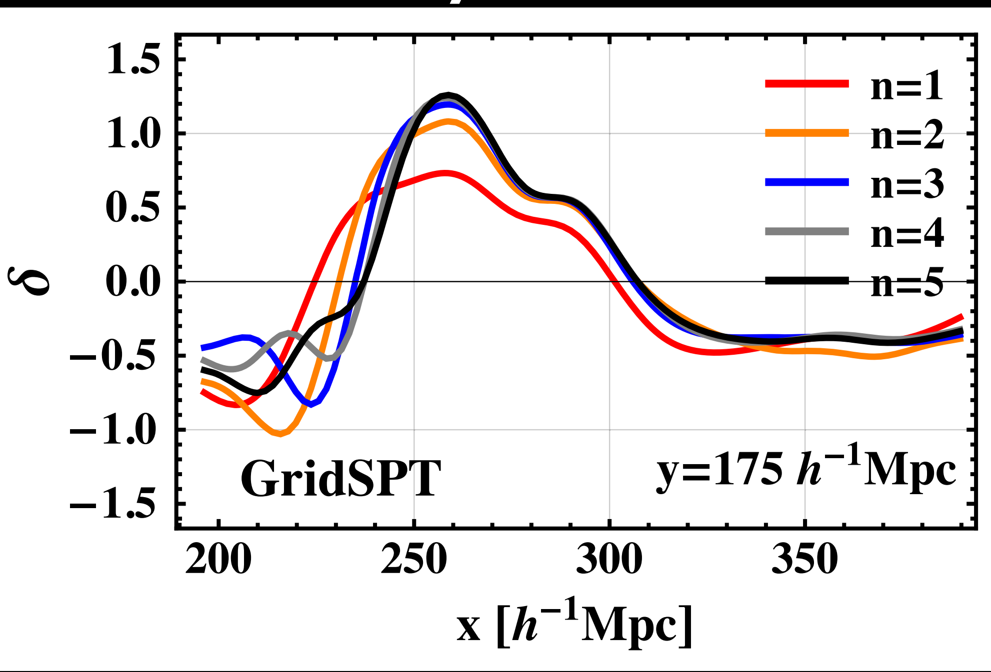
High density

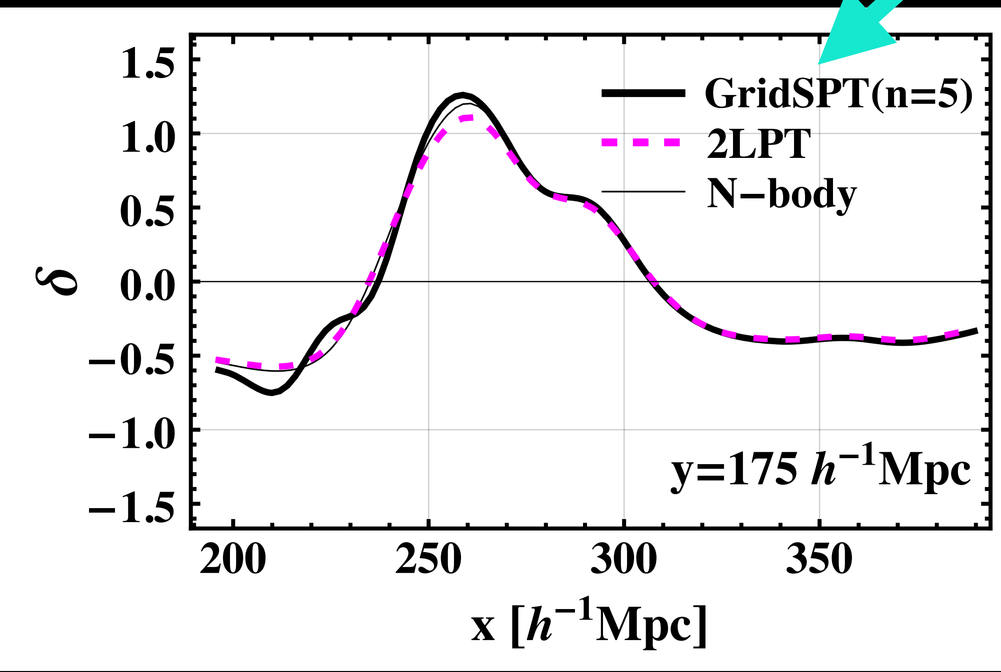




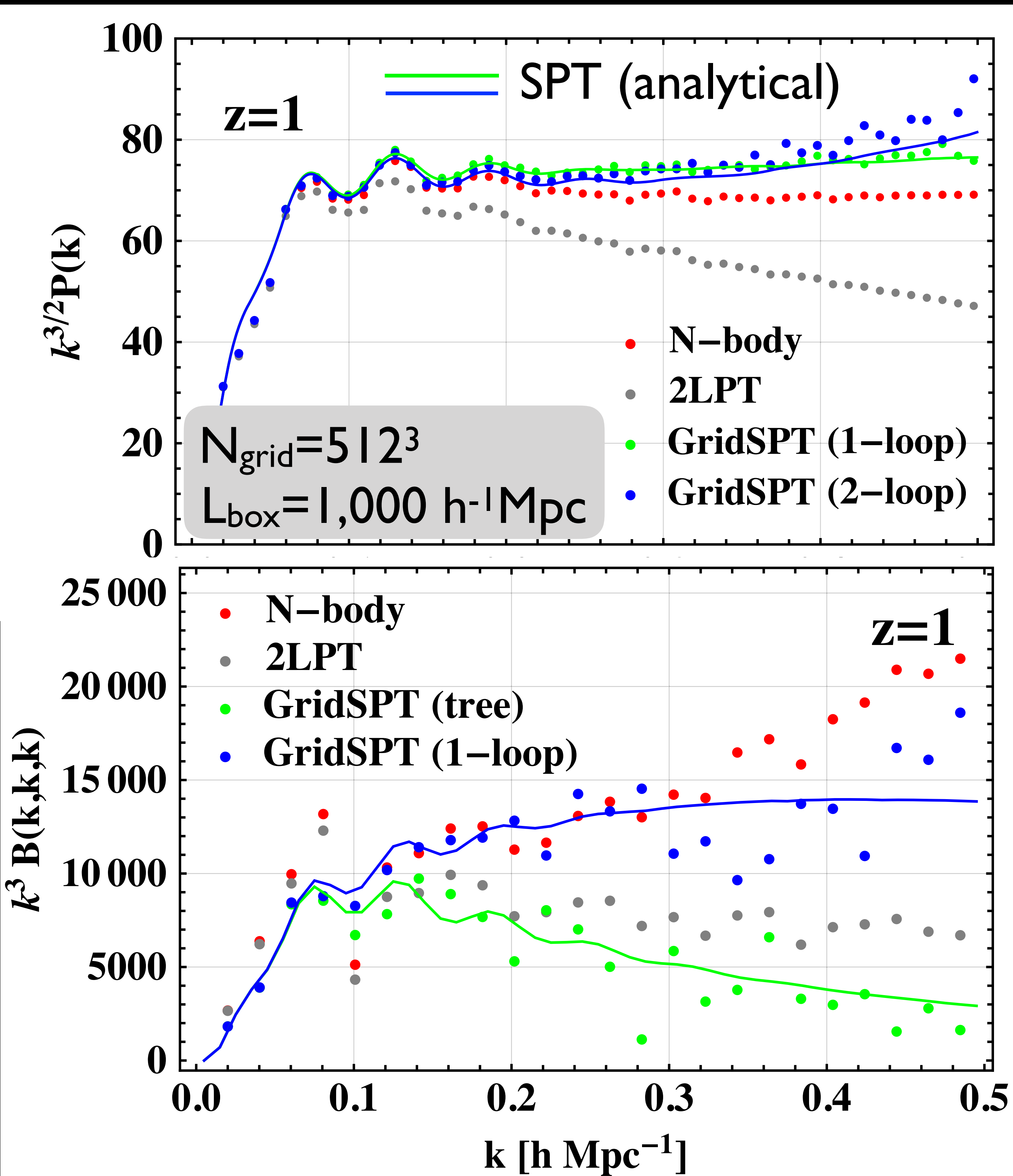


Low density





GridSPT: Statistical calculations



Can be made similarly to analytical PT, but at field level

Power spectrum

$$P_{1-\mathrm{loop}} = P_{11} + (2\,P_{13} + P_{22})$$

$$P_{2-\mathrm{loop}} = P_{11} + (2P_{13} + P_{22}) + (2\,P_{15} + 2\,P_{24} + P_{33})$$

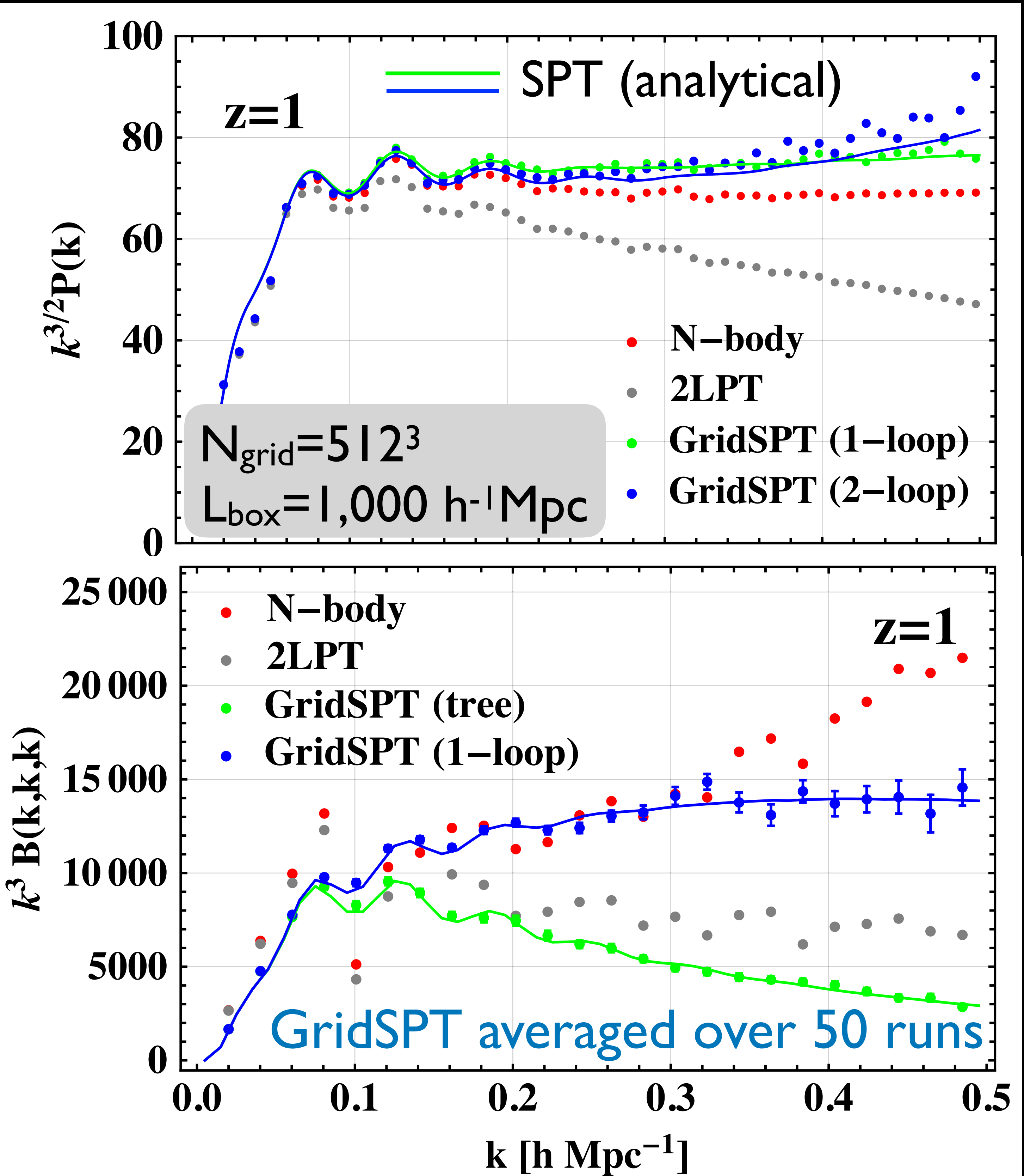
Bispectrum (equilateral)

$$B_{\mathrm{tree}} = 3\,B_{112}$$

$$B_{1-\mathrm{loop}} = 3\,B_{112} + (3\,B_{114} + 6\,B_{123} + B_{222})$$

AT, Nishimichi & Jeong ('18)

GridSPT: Statistical calculations



Can be made similarly to analytical PT, but at field level

Power spectrum

$$P_{1-\mathrm{loop}} = P_{11} + (2\,P_{13} + P_{22})$$

$$P_{2-\mathrm{loop}} = P_{11} + (2P_{13} + P_{22}) + (2\,P_{15} + 2\,P_{24} + P_{33})$$

Bispectrum (equilateral)

$$B_{\mathrm{tree}} = 3\,B_{112}$$

$$B_{1-\mathrm{loop}} = 3\,B_{112} + (3\,B_{114} + 6\,B_{123} + B_{222})$$

AT, Nishimichi & Jeong ('18)

Applications

GridSPT allows to

- generate higher-order random fields on grids very quickly
- provide a basis to compute statistical quantities

  (using the same grid-based measurement code as in N-body data)

A face-to-face comparison with N-body simulations

⟶ Measurement/calibration of EFT parameters (Tobias' talk)

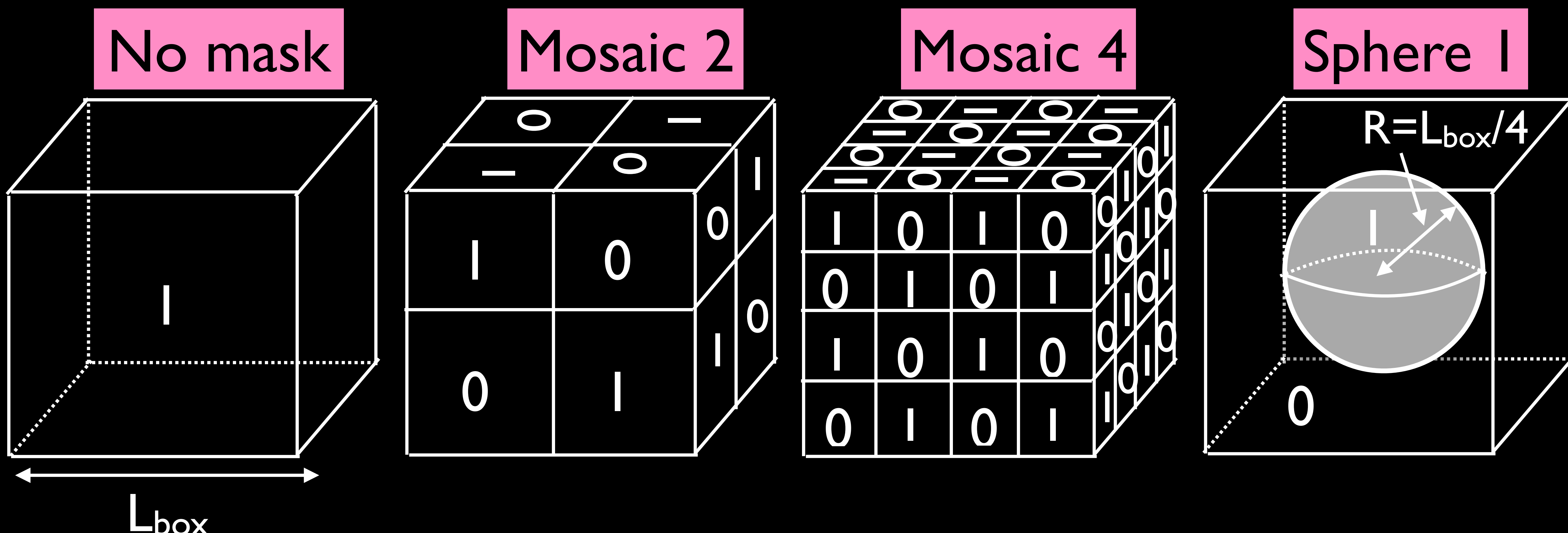
A quick generation of a large number of realizations, also accounting for observational systematics (e.g., survey mask)

➡ Covariance estimation

Covariance estimation

~ Proof-of-concept study taking account of survey masks ~

- L_{box} =512 h^{-1}Mpc, $N_{grid}=256^3$
- Artificial survey window function : $W(x) = \begin{cases} 1 \\ 0 \end{cases}$



GridSPT covariance is compared with N-body results with 10^4 runs

Non-Gaussian covariance

Power spectrum covariance

$$\mathrm{cov}(k_i, k_j) = \frac{2}{N_i}\{P(k_i)\}^2\,\delta^{\mathrm{K}}_{ij} + \frac{1}{V}\overline{T}_{ij}$$

Trispectrum

Can be decomposed into covariances of SPT power spectra:

$(\hat{P}_{11}, \hat{P}_{12}, \hat{P}_{13}, \hat{P}_{22}, \cdots)$

$$\frac{1}{V}\overline{T}^{\mathrm{tree}}_{ij} = \mathrm{cov}[\hat{P}_{11}(k_i), \hat{P}_{22}(k_j)] + 2\,\mathrm{cov}[\hat{P}_{12}(k_i), \hat{P}_{12}(k_j)] + 2\,\mathrm{cov}[\hat{P}_{11}(k_i), \hat{P}_{13}(k_j)] + (i \longleftrightarrow j)$$

$$\frac{1}{V}\overline{T}^{\mathrm{1-loop}}_{ij} = 2\,\mathrm{cov}[\hat{P}_{11}(k_i), \hat{P}_{15}(k_j)] + \cdots$$

The formulas holds irrespective of survey geometry & survey mask

Covariance estimation in GridSPT

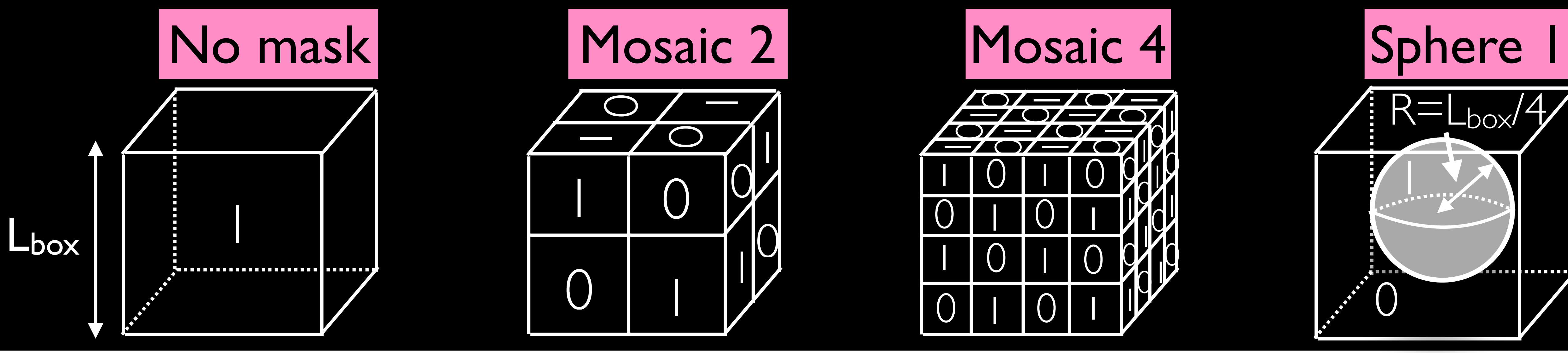
1. Measure SPT power spectra $\hat{P}_{11}, \hat{P}_{12}, \hat{P}_{13}, \hat{P}_{22}, \cdots$ in each realization
2. Evaluate $\mathrm{cov}\,[\hat{P}_{ab}(k_i), \hat{P}_{cd}(k_j)]$ from many realization data
3. Sum up these contributions in the right way

Covariance estimation

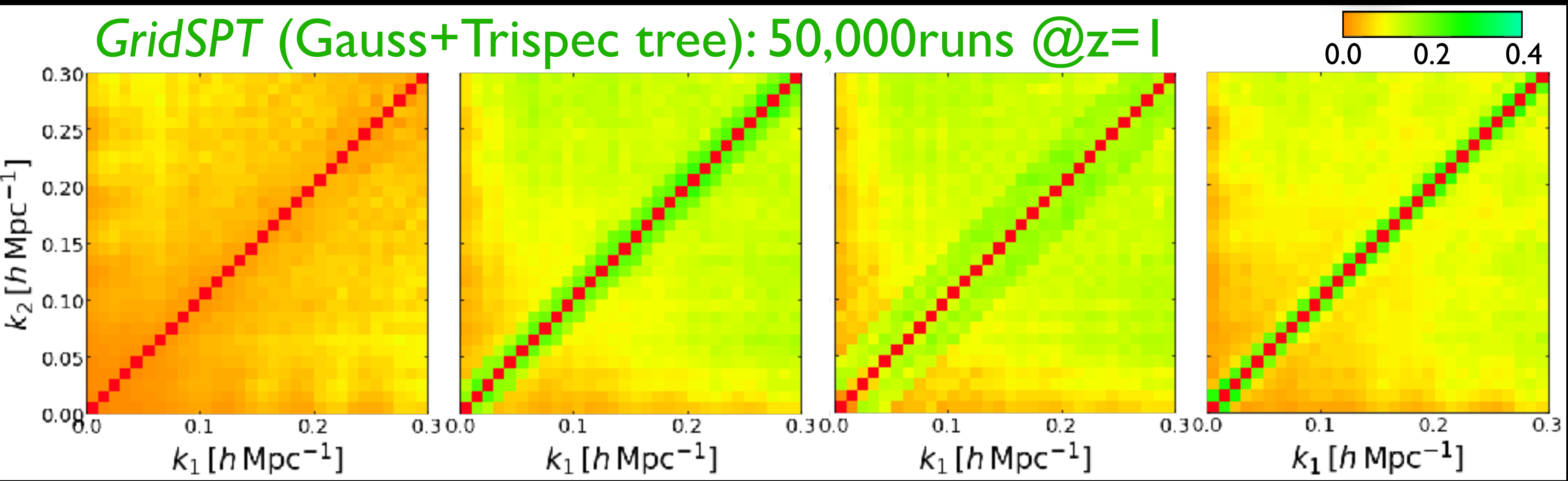
(AT, Nishimichi & Jeong in prep.)

$$r(k_1, k_2) = \frac{\mathrm{cov}(k_1, k_2)}{\sqrt{\mathrm{cov_{sim}}(k_1, k_1)\mathrm{cov_{sim}}(k_2, k_2)}}$$

Local mean subtracted



GridSPT (Gauss+Trispec tree): 50,000runs @z=1

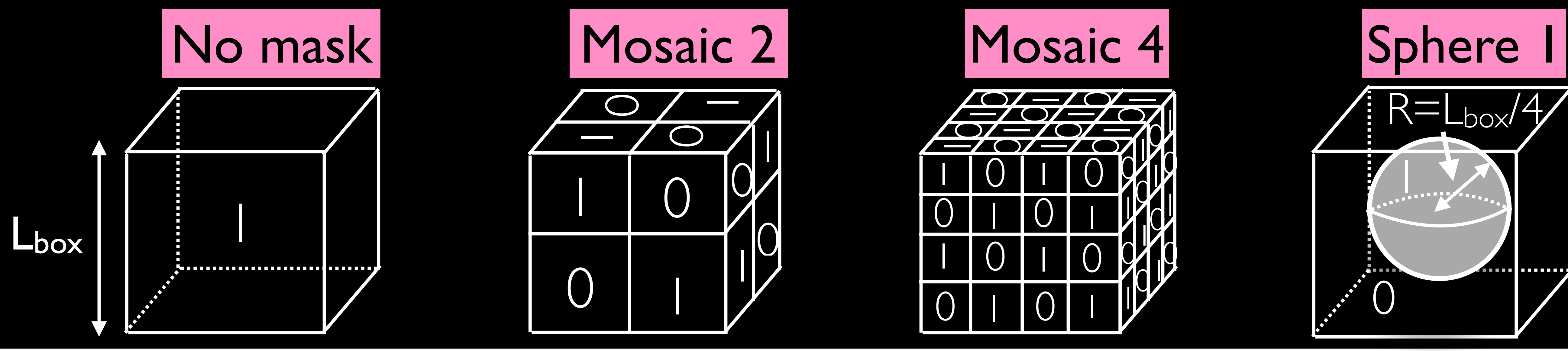


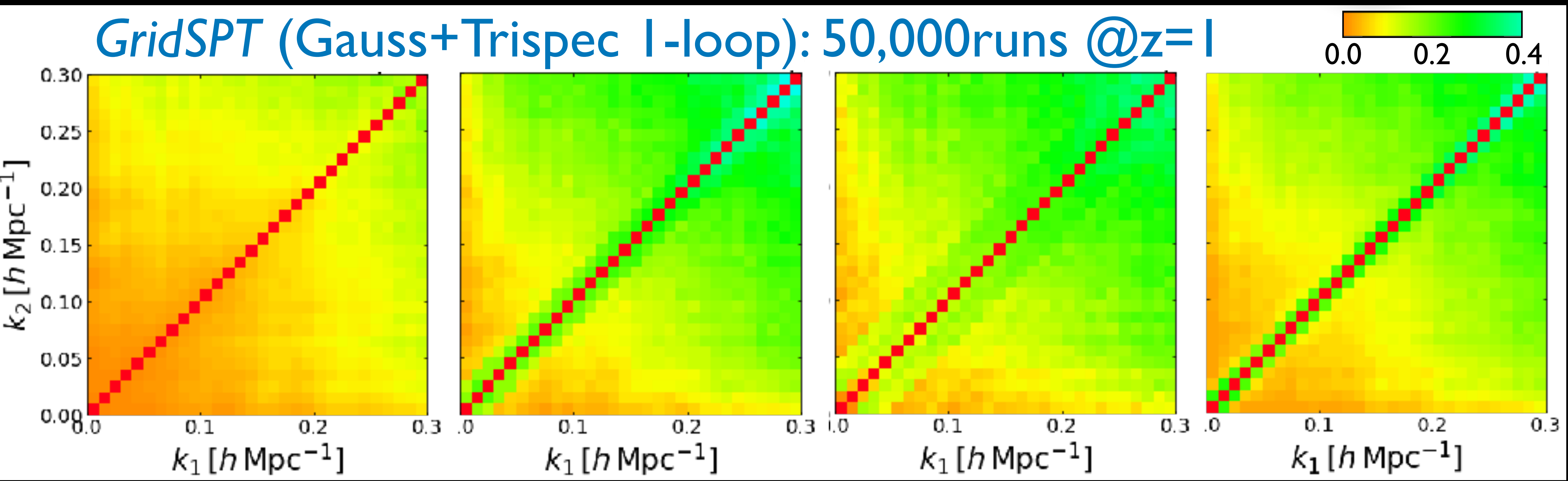
Covariance estimation

(AT, Nishimichi & Jeong in prep.)

$$r(k_1, k_2) = \frac{\mathrm{cov}(k_1, k_2)}{\sqrt{\mathrm{cov_{sim}}(k_1, k_1)\mathrm{cov_{sim}}(k_2, k_2)}}$$

Local mean subtracted



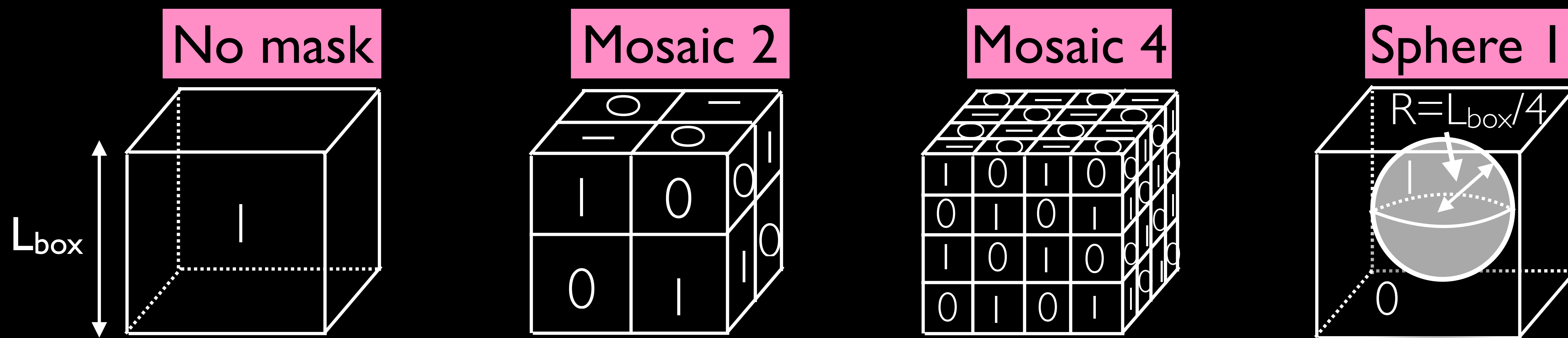


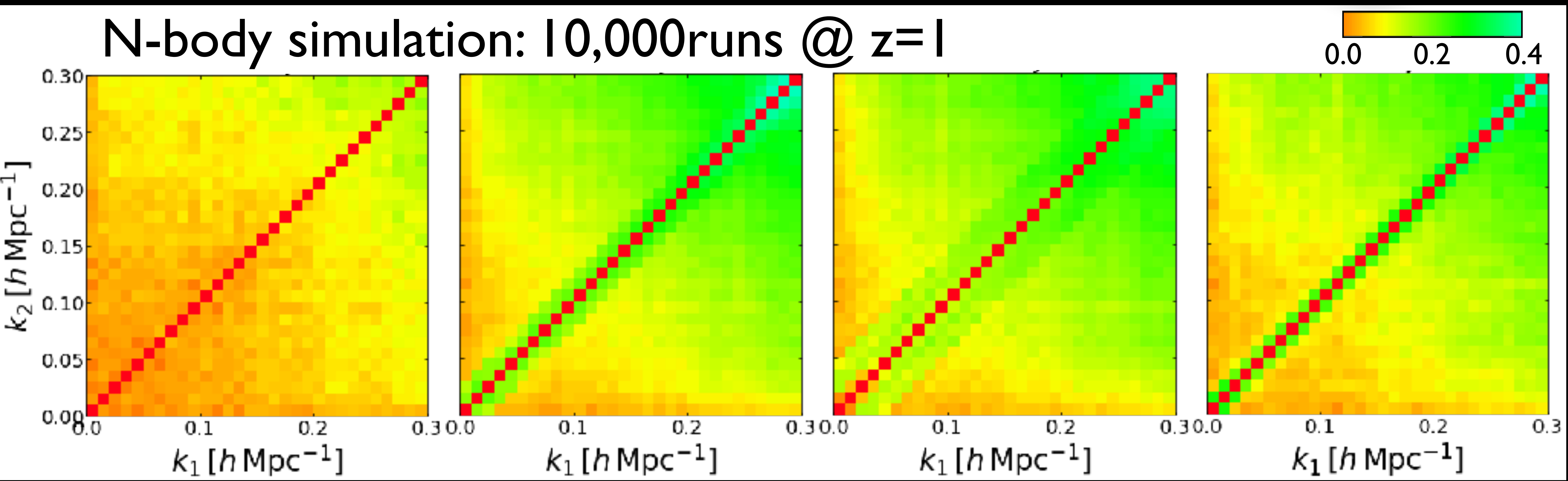
Covariance estimation

(AT, Nishimichi & Jeong in prep.)

$$r(k_1, k_2) = \frac{\mathrm{cov}(k_1, k_2)}{\sqrt{\mathrm{cov}_{\mathrm{sim}}(k_1, k_1)\mathrm{cov}_{\mathrm{sim}}(k_2, k_2)}}$$

Local mean subtracted



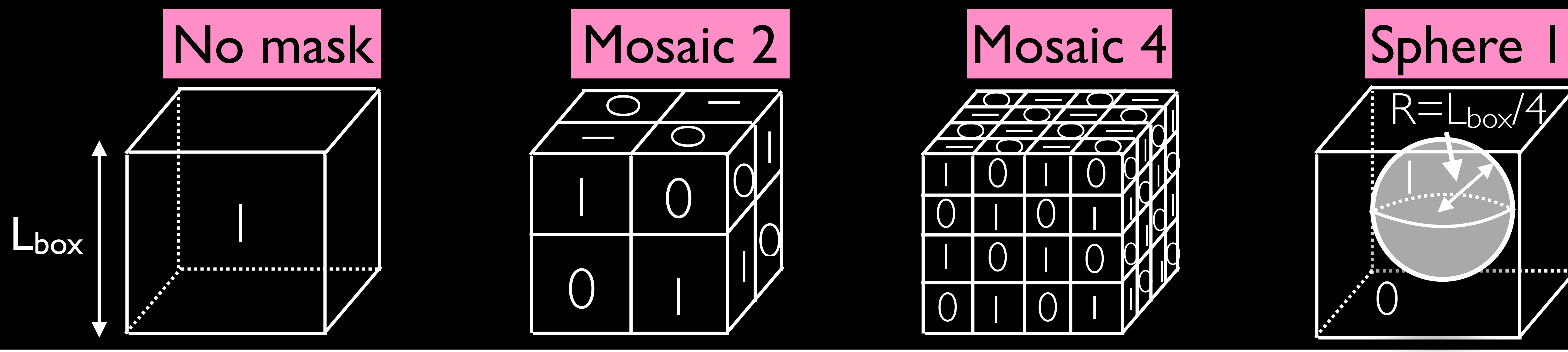


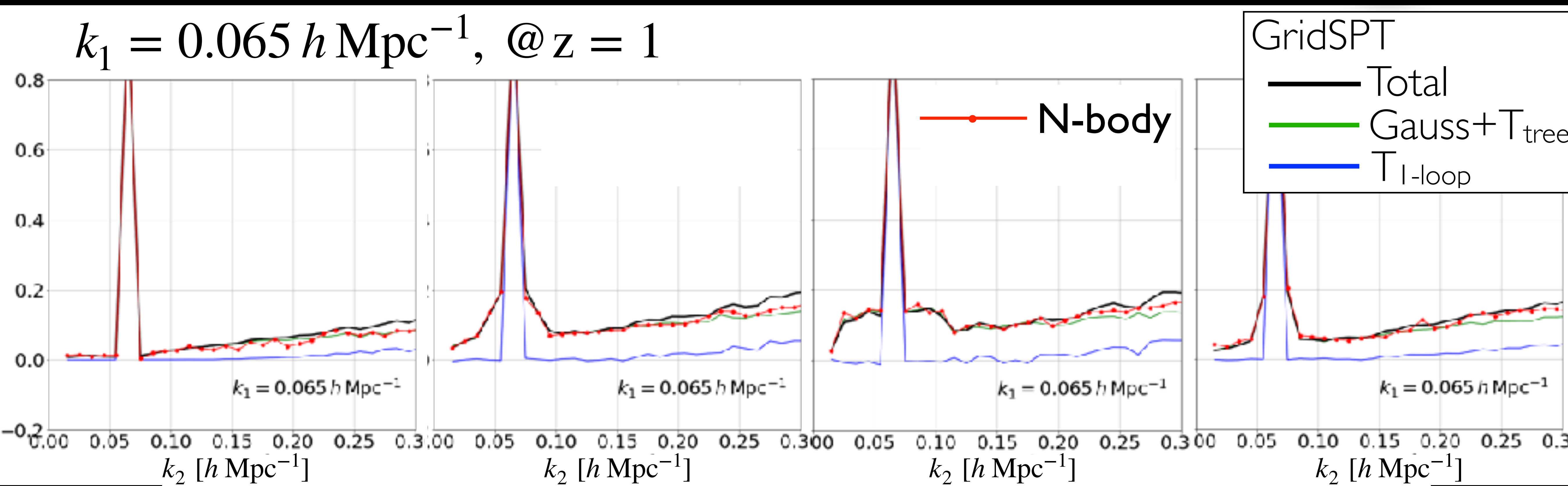
Covariance estimation

(AT, Nishimichi & Jeong in prep.)

$$r(k_1, k_2) = \frac{\mathrm{cov}(k_1, k_2)}{\sqrt{\mathrm{cov}_{\mathrm{sim}}(k_1, k_1)\mathrm{cov}_{\mathrm{sim}}(k_2, k_2)}}$$

Local mean subtracted



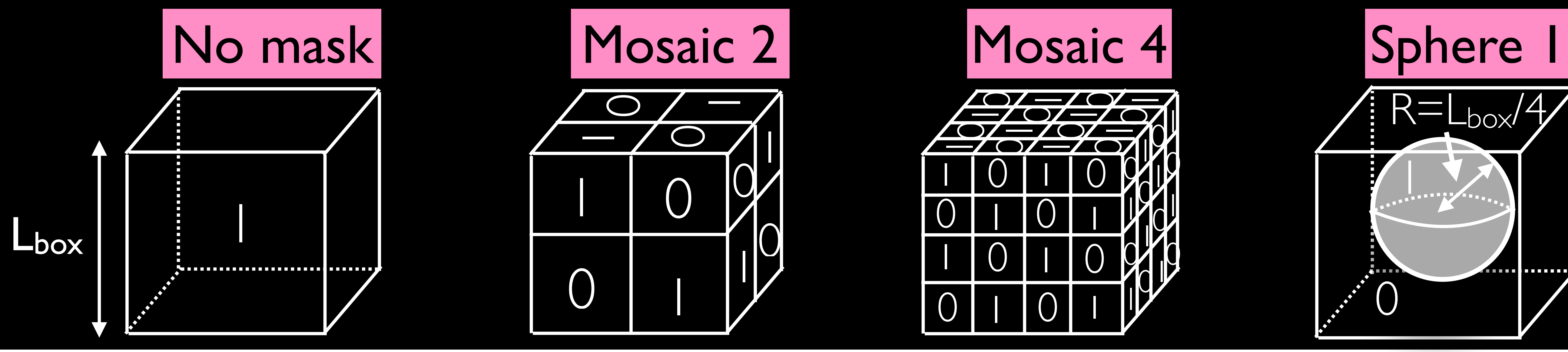


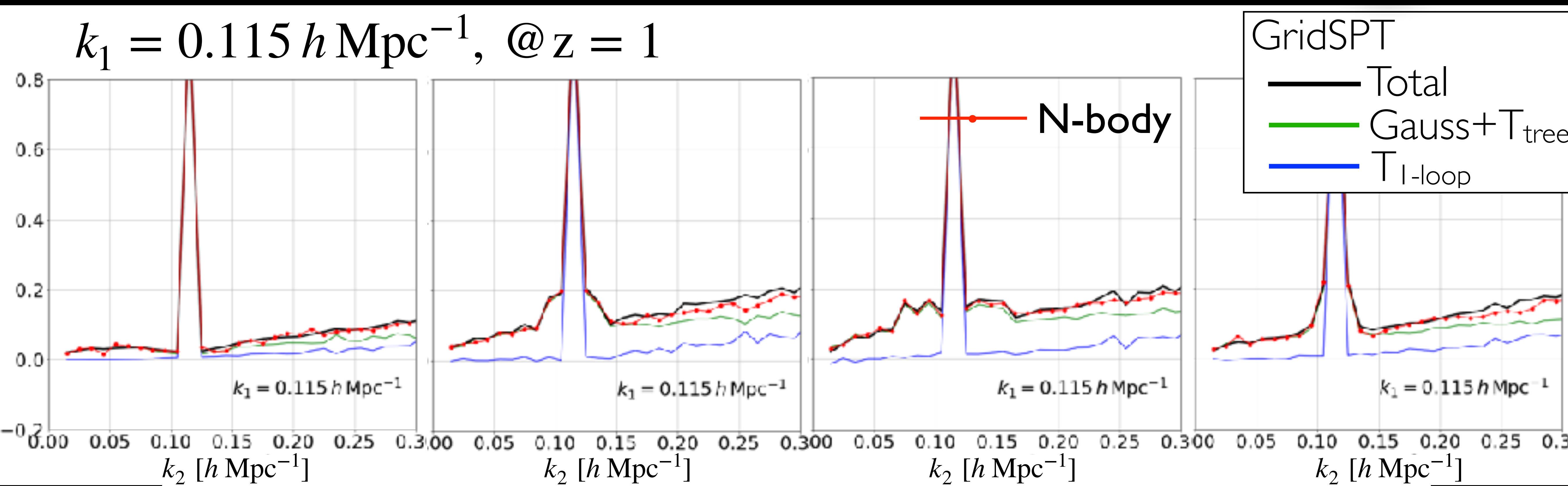
Covariance estimation

(AT, Nishimichi & Jeong in prep.)

$$r(k_1, k_2) = \frac{\mathrm{cov}(k_1, k_2)}{\sqrt{\mathrm{cov}_{\mathrm{sim}}(k_1, k_1)\mathrm{cov}_{\mathrm{sim}}(k_2, k_2)}}$$

Local mean subtracted



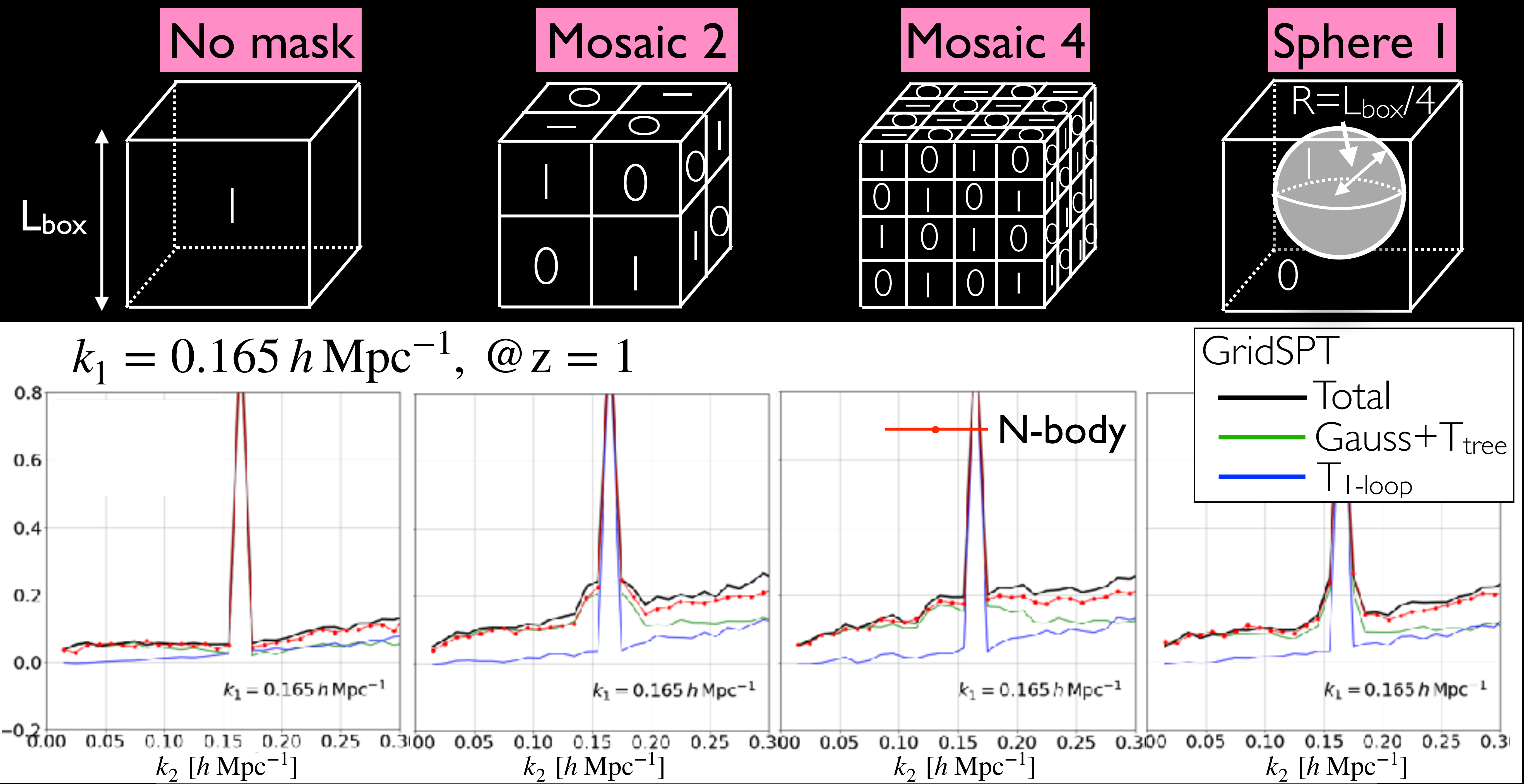


Covariance estimation

(AT, Nishimichi & Jeong in prep.)

$$r(k_1, k_2) = \frac{\mathrm{cov}(k_1, k_2)}{\sqrt{\mathrm{cov_{sim}}(k_1, k_1)\mathrm{cov_{sim}}(k_2, k_2)}}$$

Local mean subtracted



Summary

GridSPT: FFT-based code to generate density field on grids in standard perturbation theory: demonstration & application

Generating density fields in SPT calculations to *5th* order,

- Demonstration: morphological & statistical properties compared with N-body simulation & 2LPT
- Application: covariance calculations with survey mask *(trispectrum at 1-loop order)*

Other possible applications

- Mock catalogs
- Reconstructing initial density field

Incorporating field-level EFT & bias expansion is easy & straightforward