Carpooling to solve the cosmological simulation bottleneck

CARPool: fast, accurate computation of large-scale structure statistics by pairing costly and cheap cosmological simulations (arXiv:*2009.08970*)

N.Chartier, B.Wandelt, Y.Akrami and F.Villaescusa-Navarro

About observations and observables

Data sets of next generation galaxy surveys: unmatched statistical power to constrain initial perturbations, cosmic structure growth and expansion history

Euclid Space Telescope, DESI, Rubin Observatory LSST, Subaru HSC & PFS, SKA, WFIRST, SPHEREx...

Motivation

We need theoretical predictions of large-scale structure (LSS) statistics.

Possible routes

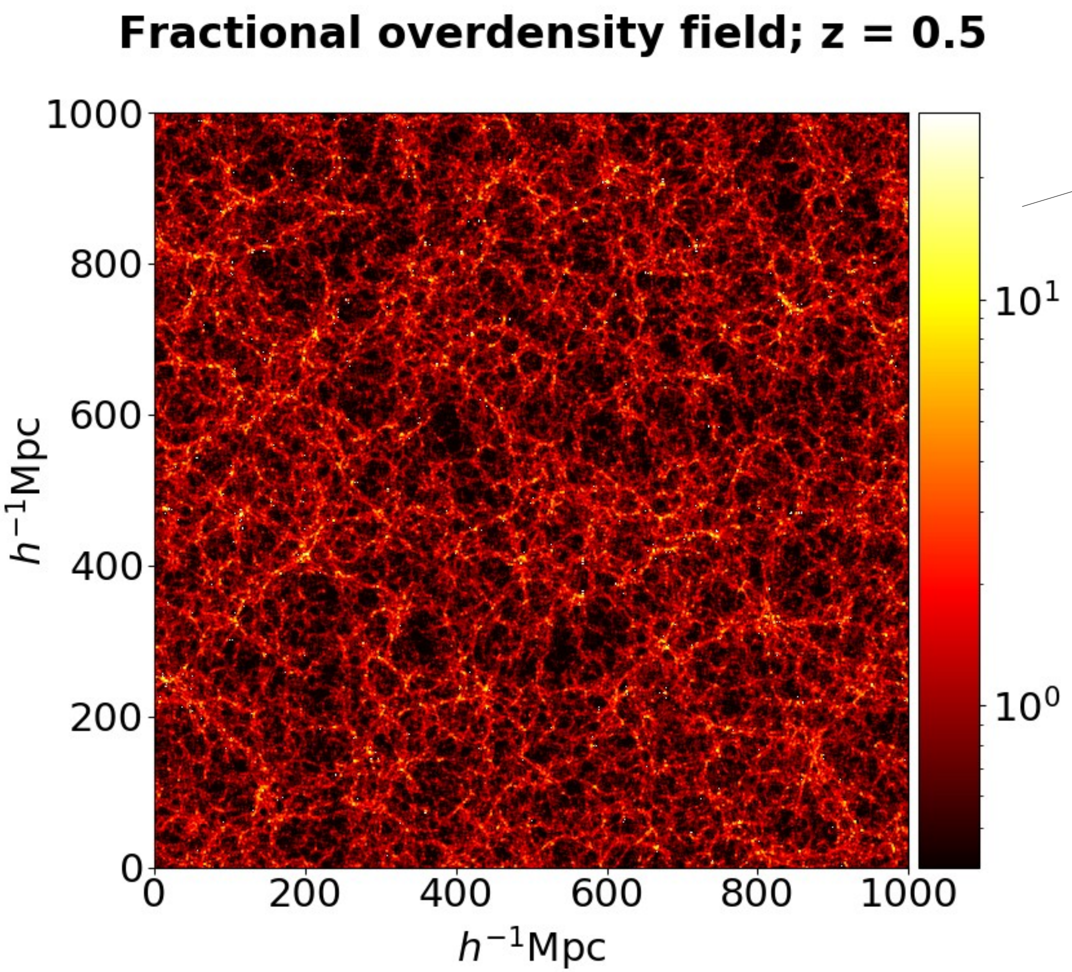
- **Costly N-body codes**, unmatched for the non-linear regime of structure growth (*GADGET, GreeM*, *HACC*, PKDGRAV3…)

- Analytical computations with ***LPT***, ***SFT***, ***EFT***…

- Approximate solvers (*surrogates*): Particle-Mesh codes (**PM**), emulators, Neural Networks...

Accuracy is traded for computational speed (especially in the non-linear regime), statistical unbiasedness not guaranteed...

Statistics of observables from N-body simulations



$$\rho(\boldsymbol{x})/\bar{\rho}$$

- Matter Power Spectrum
- Matter Bispectrum
- Halo Mass Function...

- Observables (bins) are collected into a vector $\boldsymbol{y}$ of size p
- Estimator $\hat{\boldsymbol{\mu}}$ of $\mathbb{E}\left[\boldsymbol{y}\right]$?

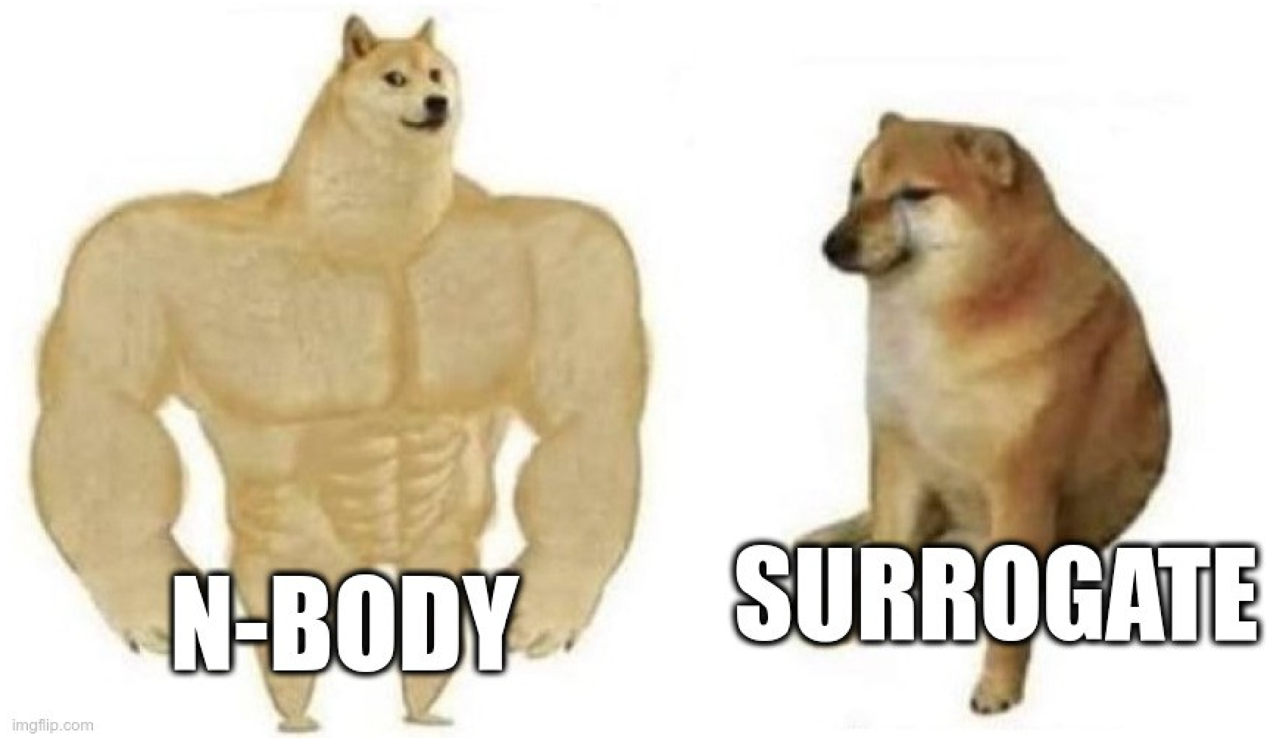
Random events and estimation

- $\boldsymbol{y_1}, \ldots, \boldsymbol{y_N}$ are N independent random realizations sampled on seeds $\mathrm{r}_1, \ldots, r_N$
- Estimation of the mean $\mathbb{E}[\boldsymbol{y}] = \boldsymbol{\mu} \in \mathbb{R}^p$ with $\bar{\boldsymbol{y}} = \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{y_n}$
- $\boxed{\text{Standard deviation of each element } \bar{y_i} \text{ decreases as } \mathcal{O}(N^{-\frac{1}{2}})}$

$$\longrightarrow \boldsymbol{y}_n = \begin{pmatrix} y_{1n} \\ y_{2n} \\ \vdots \\ y_{pn} \end{pmatrix}$$

Can we get the best of both worlds?

An unbiased and faster estimator



“CARPool” Method:

Variance reduction with
“N-body + surrogate” pairs

Numerical analysis
What we are going to use

ΛCDM cosmology, Redshift z=0.5

$\boldsymbol{y}$ $\longrightarrow$ N-body simulations – GADGET – are from the *Quijote Simulations* (Villaescusa-Navarro et al., 2019)

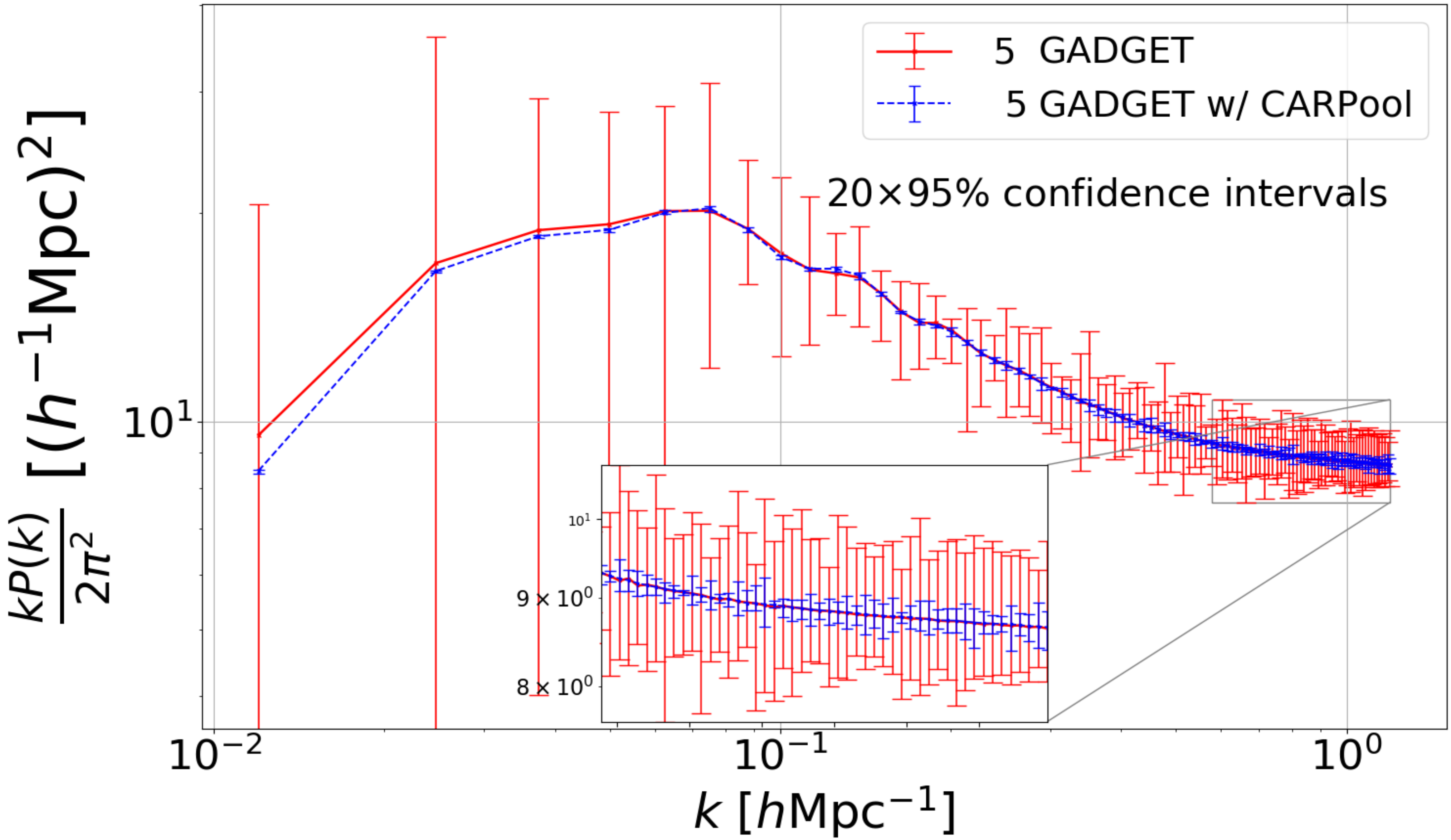
$\boldsymbol{c}$ $\longrightarrow$ The cheap *surrogate* is L-PICOLA (Howlett, 2015 b), an *MPI* implementation of COLA (Tassev, 2013)

An example

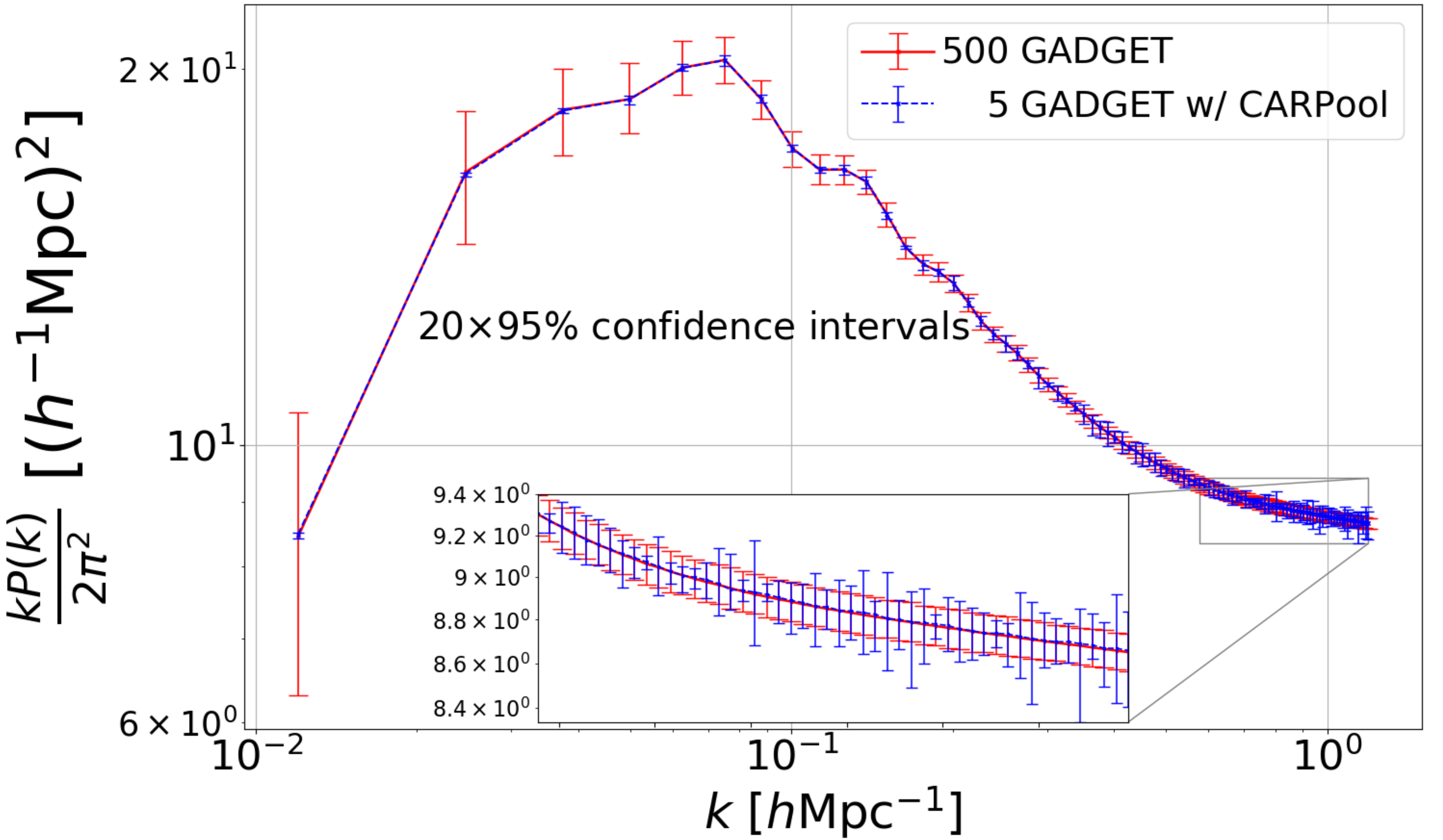
Matter power spectrum
95 linearly spaced bins with :

$k_{max} = 1.184\ h\mathrm{Mpc}^{-1}$ $\Delta k = 3.147\text{e-}2\ h\mathrm{Mpc}^{-1}$

CARPool estimate Vs. N-body only



CARPool estimate Vs. N-body only



What is the trick?

Control Variates principle

- **Observables from simulations:**

N-body code	"cheap" surrogate
$\boldsymbol{y} = \begin{pmatrix} y_1 & y_2 & \dots & y_p \end{pmatrix}^{\boldsymbol{T}}$	$\boldsymbol{c} = \begin{pmatrix} c_1 & c_2 & \dots & c_q \end{pmatrix}^{\boldsymbol{T}}$
$\mathbb{E}\left[\boldsymbol{y}\right] = \boldsymbol{\mu}$	$\mathbb{E}\left[\boldsymbol{c}\right] = \boldsymbol{\mu_c}$
Unknown truth	*(Un)known "wrong" truth*

- **Intuition with two random scalars:**

$$\sigma^2_{y+c} = \sigma^2_y + \sigma^2_c + 2\mathrm{cov}(y, c)$$

Control Variates for simulations

- **Scalar case (“bin per bin”):**

$$
\begin{cases}
\mathrm{x}(\beta) = y - \beta\,(c - \mu_c)\,,\beta \in \mathbb{R} \\[2ex]
\mathbb{E}\left[x(\beta)\right] = \mathbb{E}\left[y\right] = \mu \\[2ex]
\beta^{\star} = argmin_{\beta \in \mathbb{R}}\left[\sigma^2_{x(\beta)}\right] = cov(y,c)/\sigma^2_c \\[2ex]
\dfrac{\sigma^2_{x(\beta^{\star})}}{\sigma^2_y} = 1 - \rho^2_{yc}
\end{cases}
$$

Error bar (1D)

Optimality

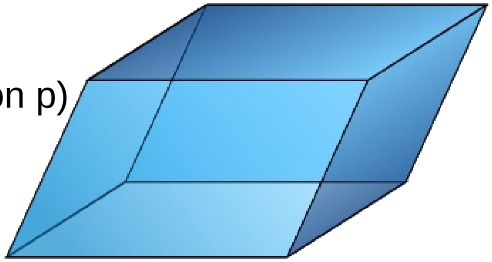
WE DON’T CARE ABOUT THE BIAS OF THE CHEAP ESTIMATOR

Control Variates for simulations

- **Multivariate case:** $\boldsymbol{x}(\boldsymbol{\beta}) = \boldsymbol{y}$ - $\boldsymbol{\beta}\left(\boldsymbol{c} - \boldsymbol{\mu_c}\right), \boldsymbol{\beta} \in \mathbb{R}^{p \times q}$

Proof in *Rubinstein & Marcus (1985)*

$$\boldsymbol{\beta}^\star = \boldsymbol{\Sigma_{yc}}\boldsymbol{\Sigma_{cc}^{-1}}$$



$$\frac{det\left(\boldsymbol{\Sigma_{x(\beta)x(\beta)}}\right)}{det\left(\boldsymbol{\Sigma_{yy}}\right)} = \prod_{n=1}^{s=rank(\boldsymbol{\Sigma_{yc}})} \left(1 - \lambda_n^2\right)$$

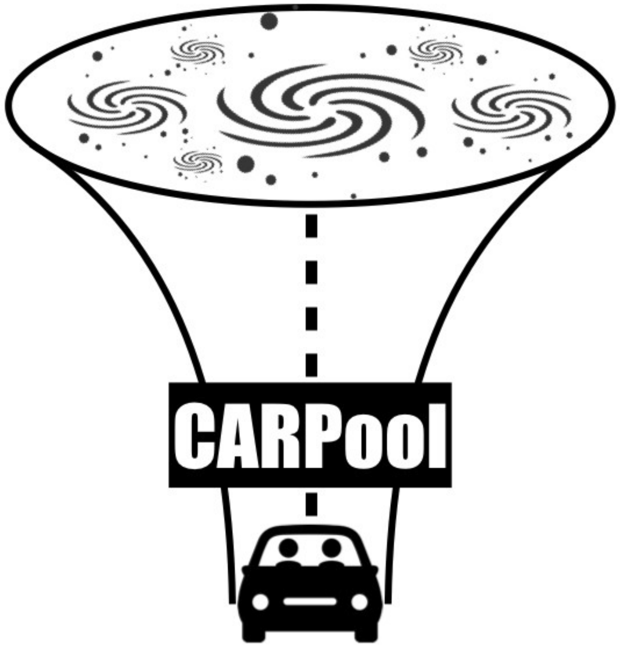
Squared canonical cross-correlations

1) The estimate is unbiased by construction

2) The control matrix/coefficient gives optimal variance reduction

3) The more correlated the full simulation and the surrogate statistics, the better

CARPool



- **In practice:**

$\beta^\star$ must be estimated with data

$\boldsymbol{\mu_c}$ is unknown

Convergence **A**cceleration by **R**egression and **Pool**ing

1 Estimate $\bar{\boldsymbol{\mu}}_{\boldsymbol{c}}$ from M fast surrogates

2 With N "simulation + surrogate" pairs, compute

$$\bar{\boldsymbol{x}}(\hat{\boldsymbol{\beta}}) = \bar{\boldsymbol{y}} - \hat{\boldsymbol{\beta}}\,(\bar{\boldsymbol{c}} - \overline{\boldsymbol{\mu}}_{\boldsymbol{c}})$$

- **N-body sims only**

$$\bar{\mathbf{y}} = \frac{1}{L}\sum_{n=1}^{L}\mathbf{y_n}$$

Vs.

- **Multivariate CARPool**

$$\boldsymbol{\beta}^{\star} = \boldsymbol{\Sigma}_{\boldsymbol{yc}}\boldsymbol{\Sigma}_{\boldsymbol{cc}}^{\dagger}$$

Moore-Penrose pseudo-inverse

Vs.

- **Univariate CARPool**

$$\beta^{diag} = \begin{pmatrix} \frac{cov(y_1,c_1)}{\sigma_{c_1}^2} & & 0 \\ & \frac{cov(y_2,c_2)}{\sigma_{c_2}^2} & \\ 0 & & \ddots & \\ & & & \frac{cov(y_p,c_p)}{\sigma_{c_p}^2} \end{pmatrix}$$

$$\frac{1}{N}\sum_{n=1}^{N}\boldsymbol{x_n}(\hat{\boldsymbol{\beta}}) = \bar{\boldsymbol{x}}(\hat{\boldsymbol{\beta}})$$

$$\mathrm{N} \ll L$$

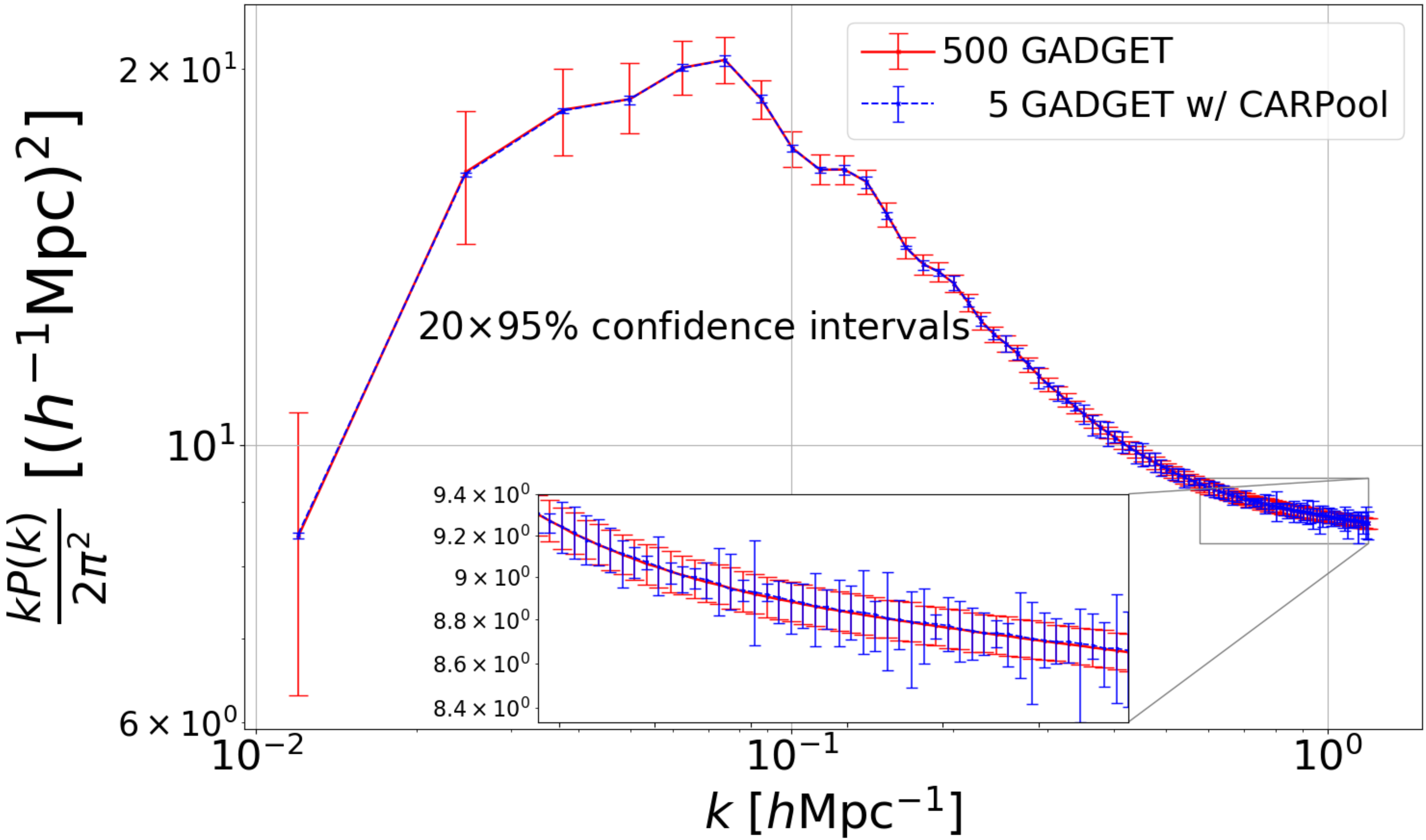
(hopefully)

Back to the first example

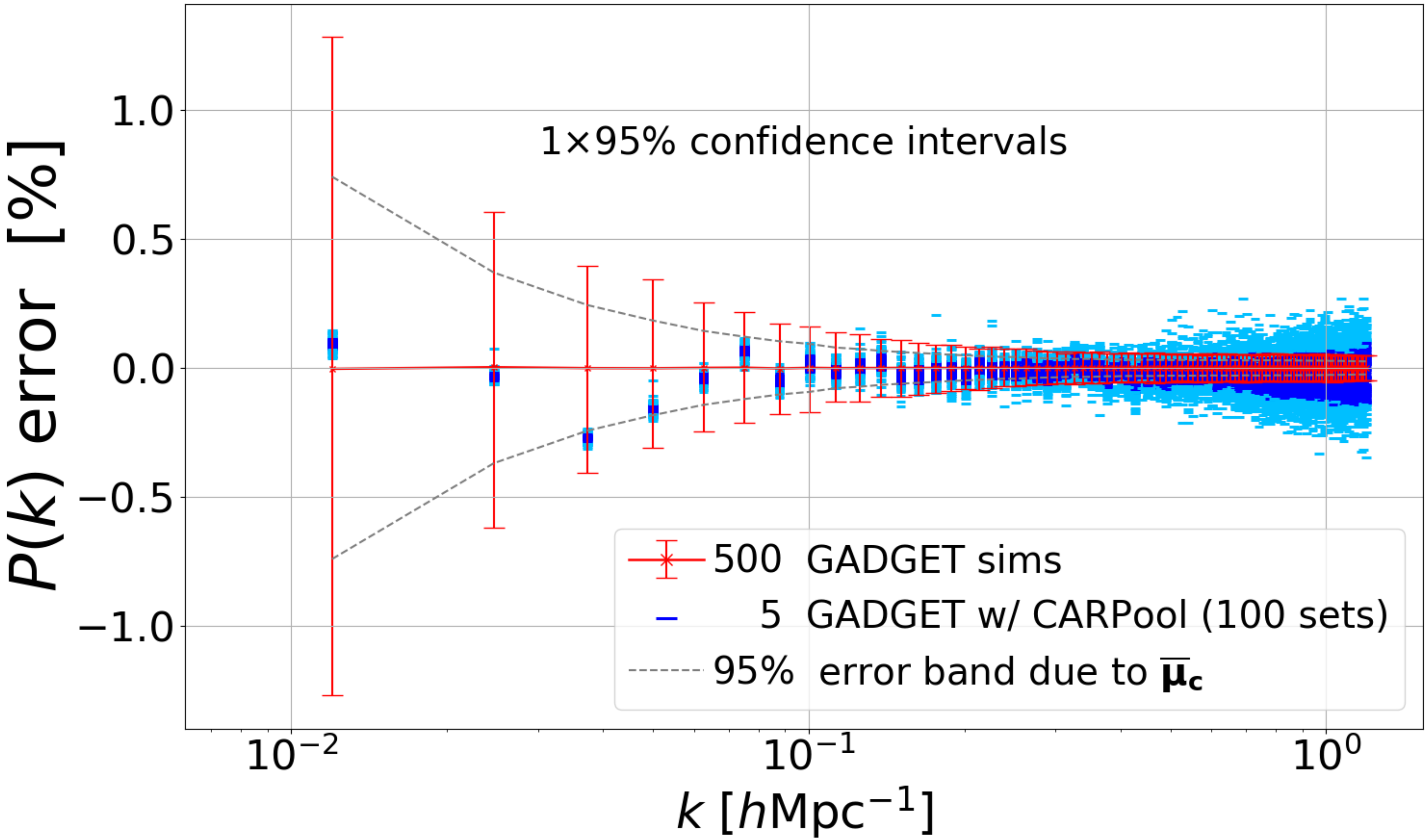
Matter power spectrum
95 linearly spaced bins with :

$$k_{max} = 1.184\ h\mathrm{Mpc}^{-1} \qquad \Delta k = 3.147\text{e-}2\ h\mathrm{Mpc}^{-1}$$

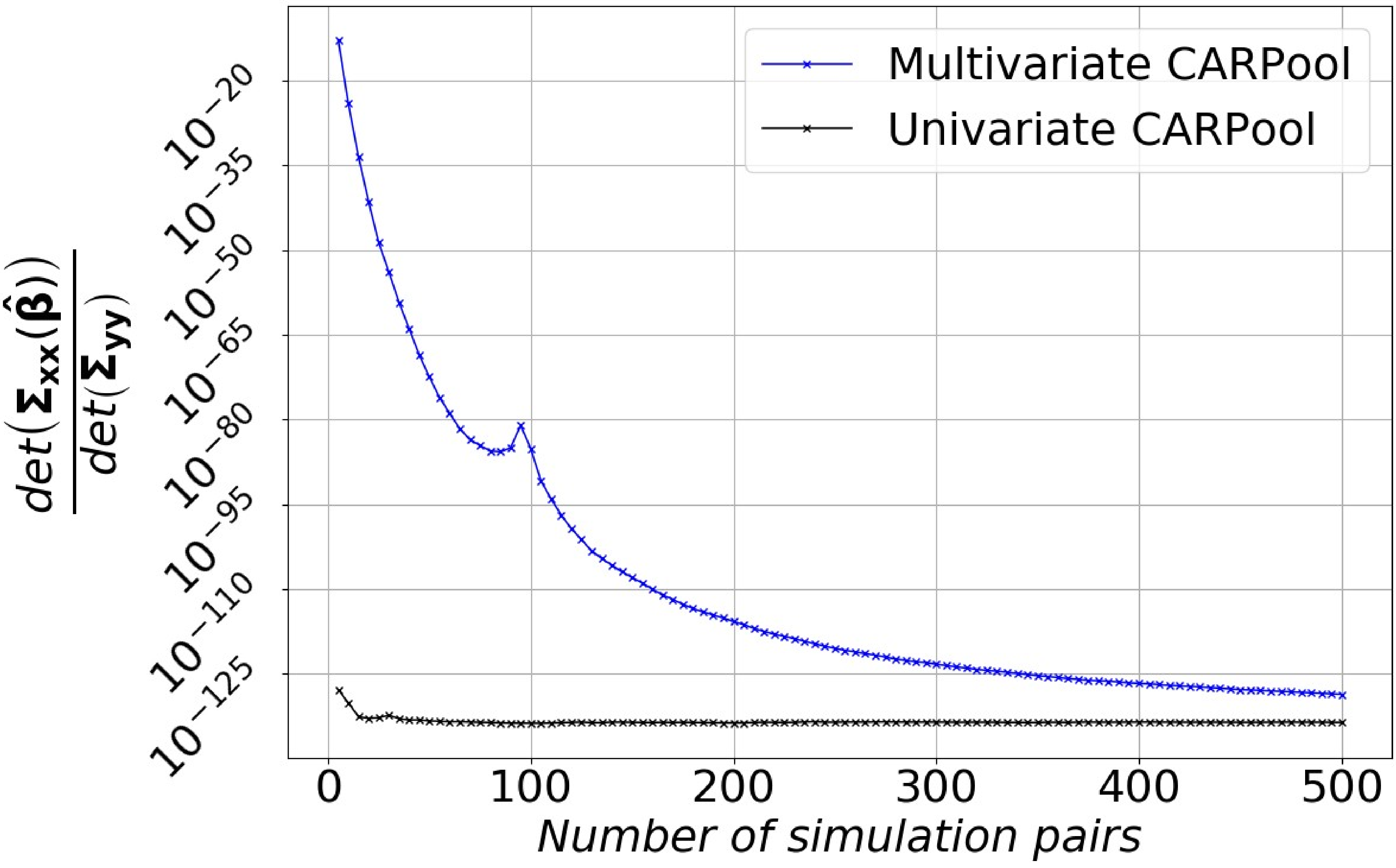
CARPool estimate Vs. N-body only



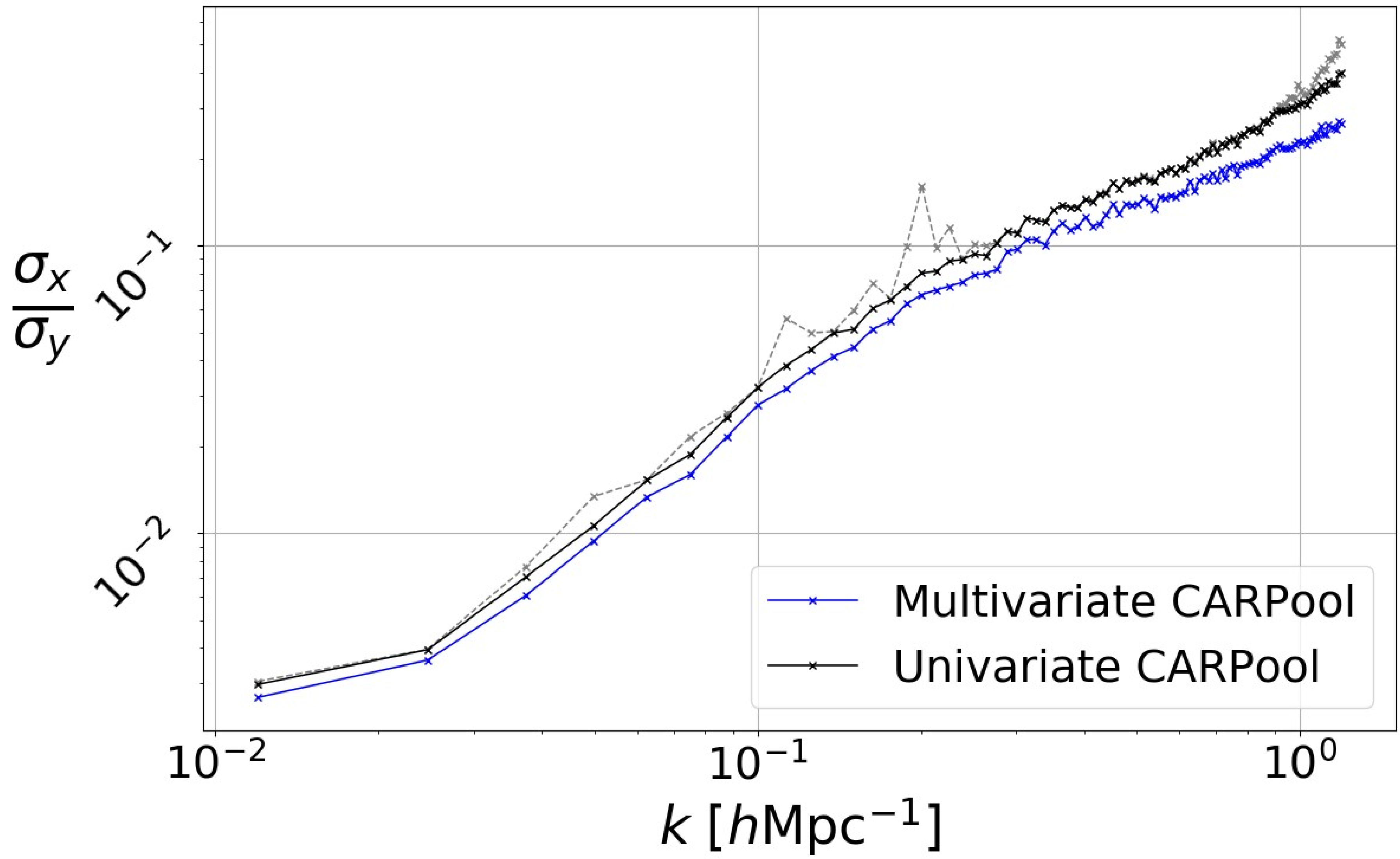
Confidence in CARPool estimate (Pk)



Generalized variance reduction (Pk)



Standard deviation reduction (Pk)

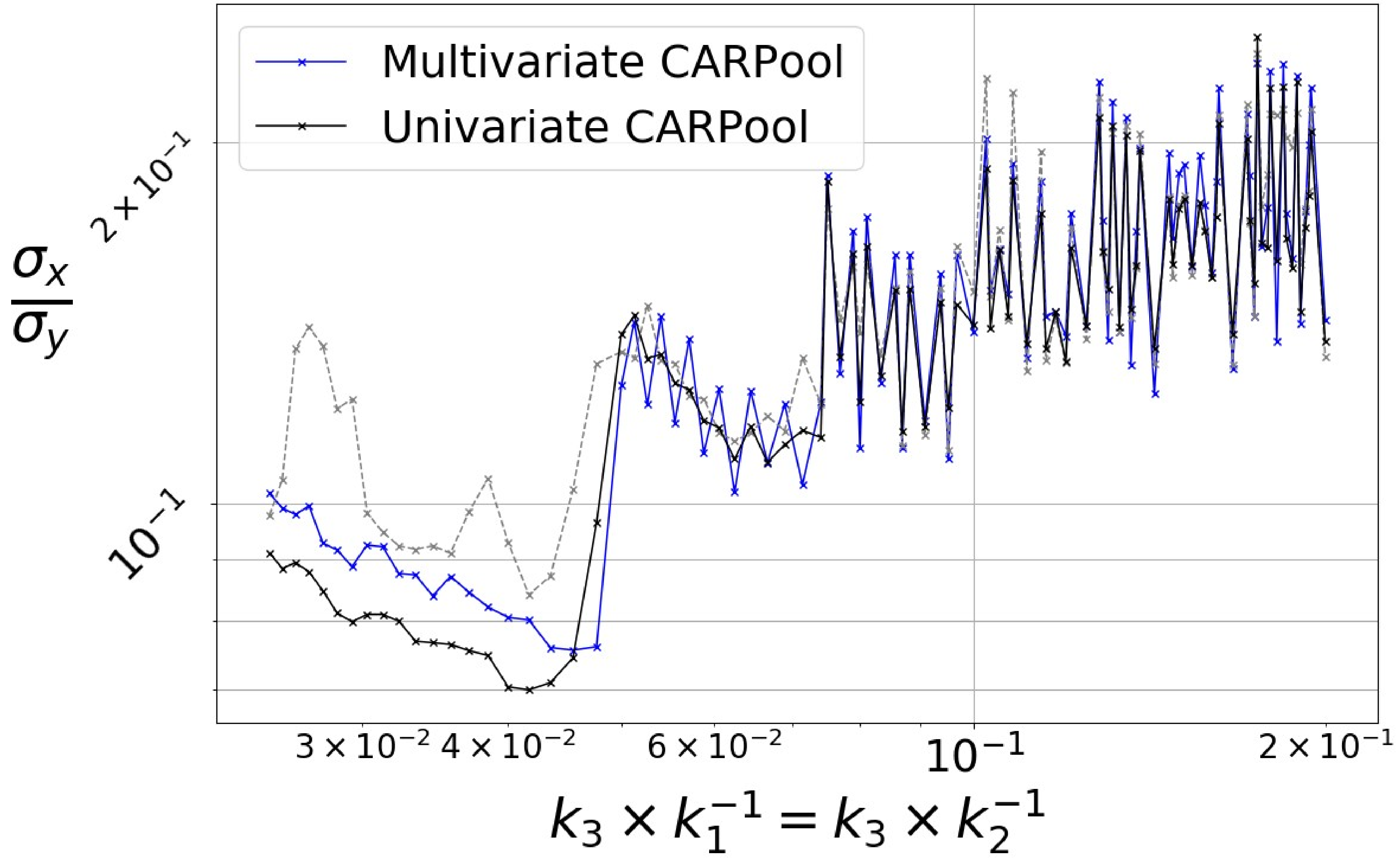


Matter Bispectrum
73 squeezed triangle configurations

$k_1 = k_2$

$$\text{Function of } \frac{k_3}{k_1}$$

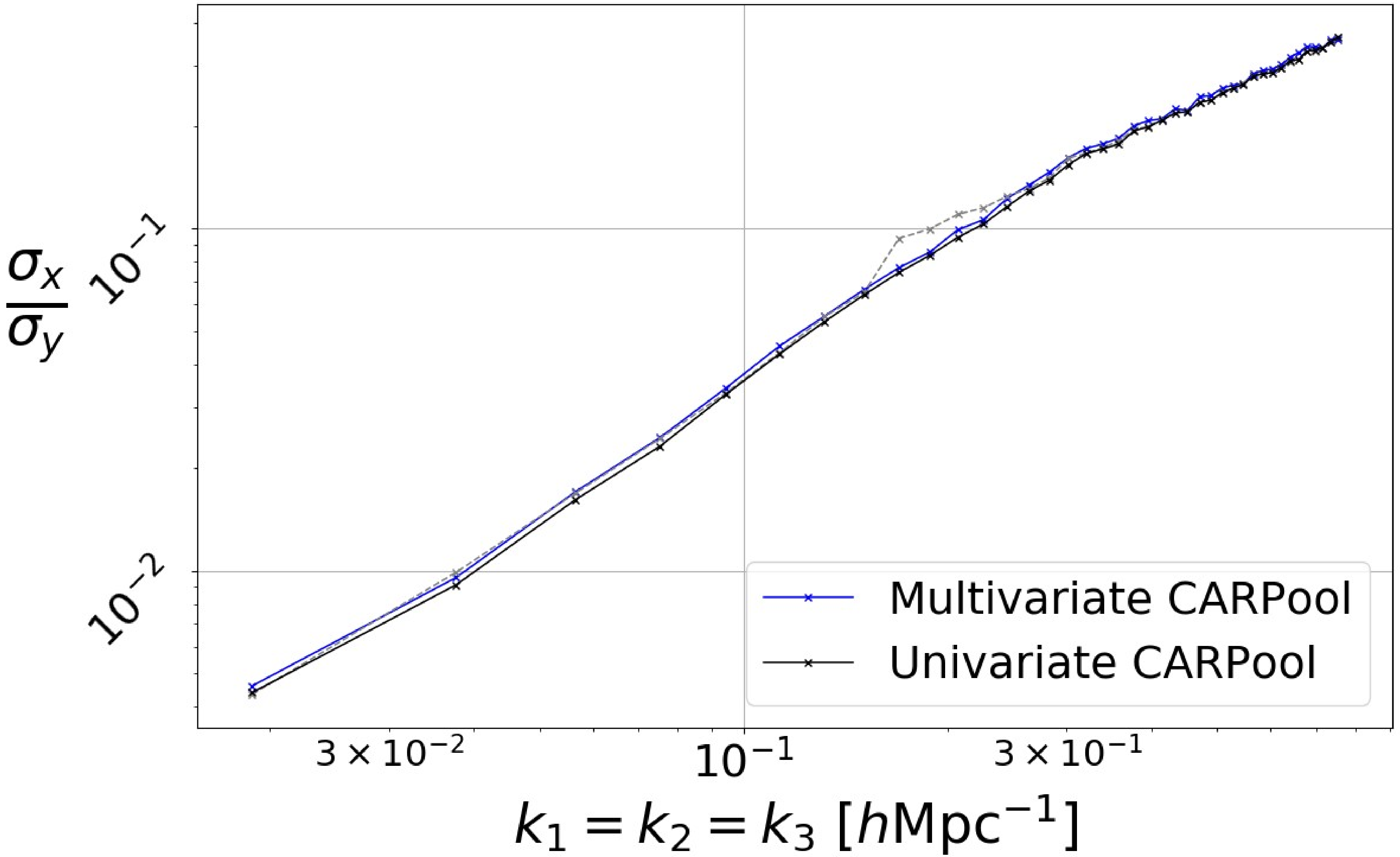
Standard deviation reduction (Bk)



Matter reduced Bispectrum
40 equilateral triangle configurations

$$k_1 = k_2 = k_3$$

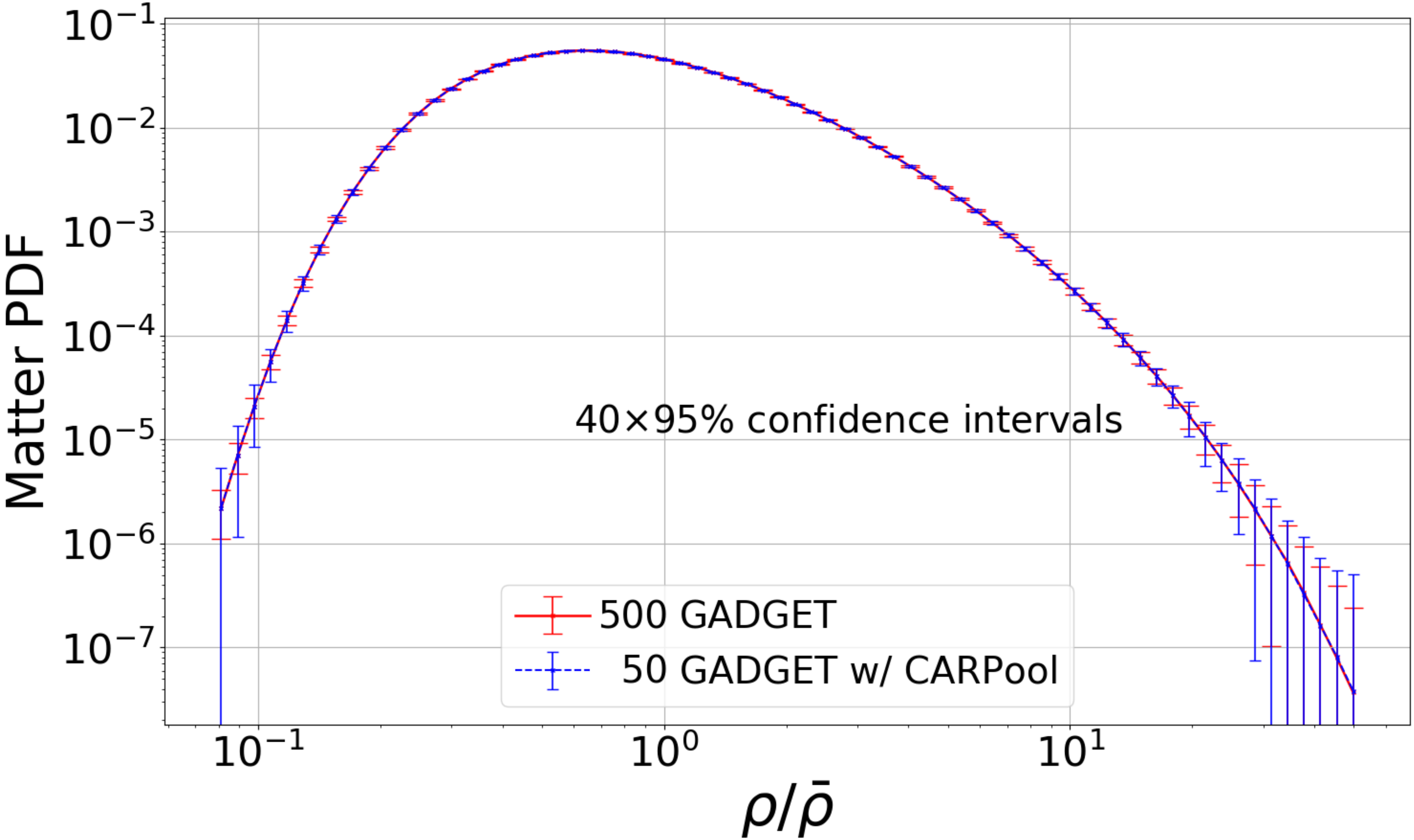
Confidence in CARPool estimate (Qk)

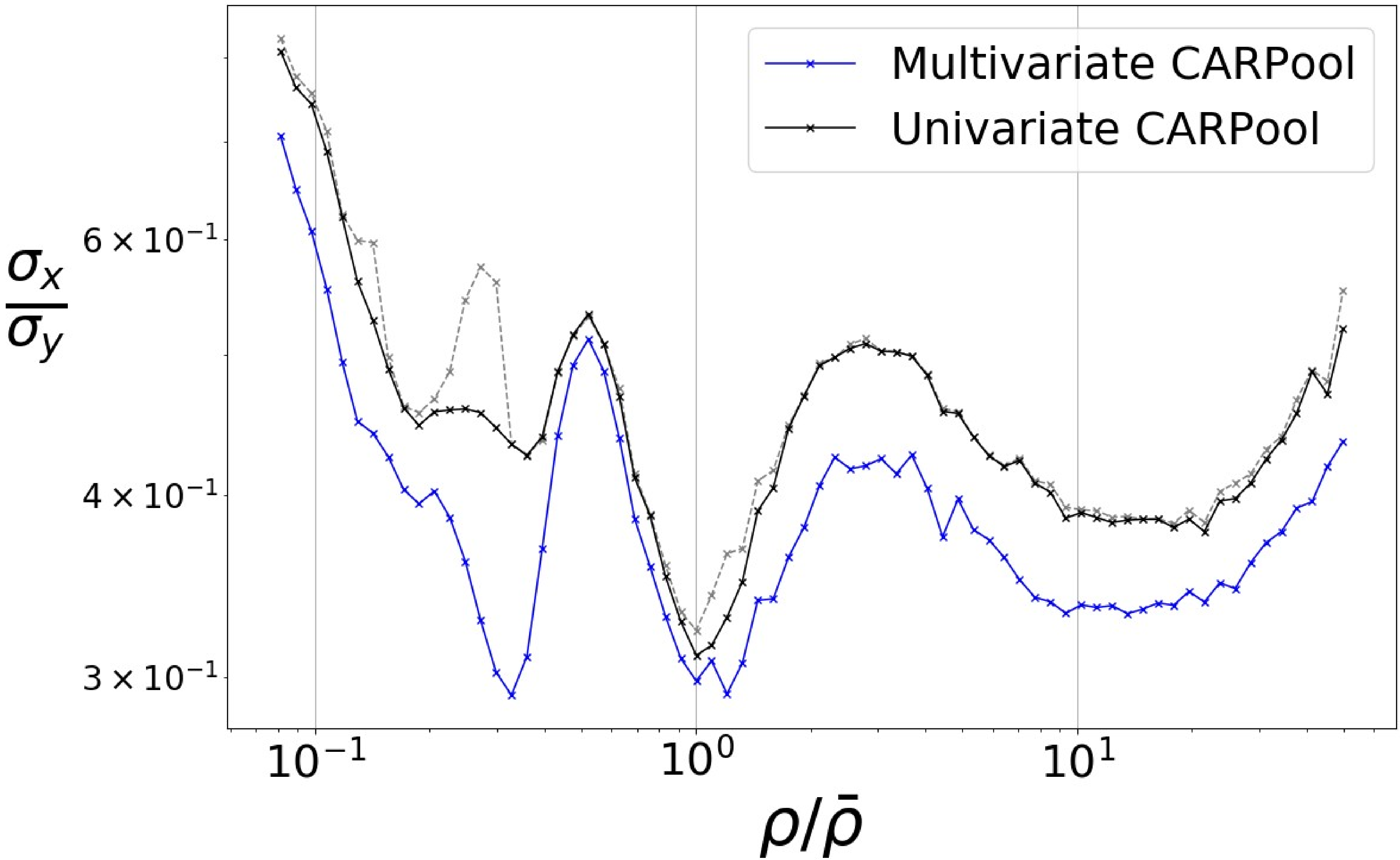


Matter PDF

70 bins $\rho/\bar{\rho} \in [0.08, 50]$

Univariate CARPool for PDF





Multivariate CARPool
Univariate CARPool
6×10^{-1}
4×10^{-1}
3×10^{-1}
$\frac{\sigma_x}{\sigma_y}$
10^{-1}
10^{0}
10^{1}
$\rho/\bar{\rho}$

Conclusion and discussion

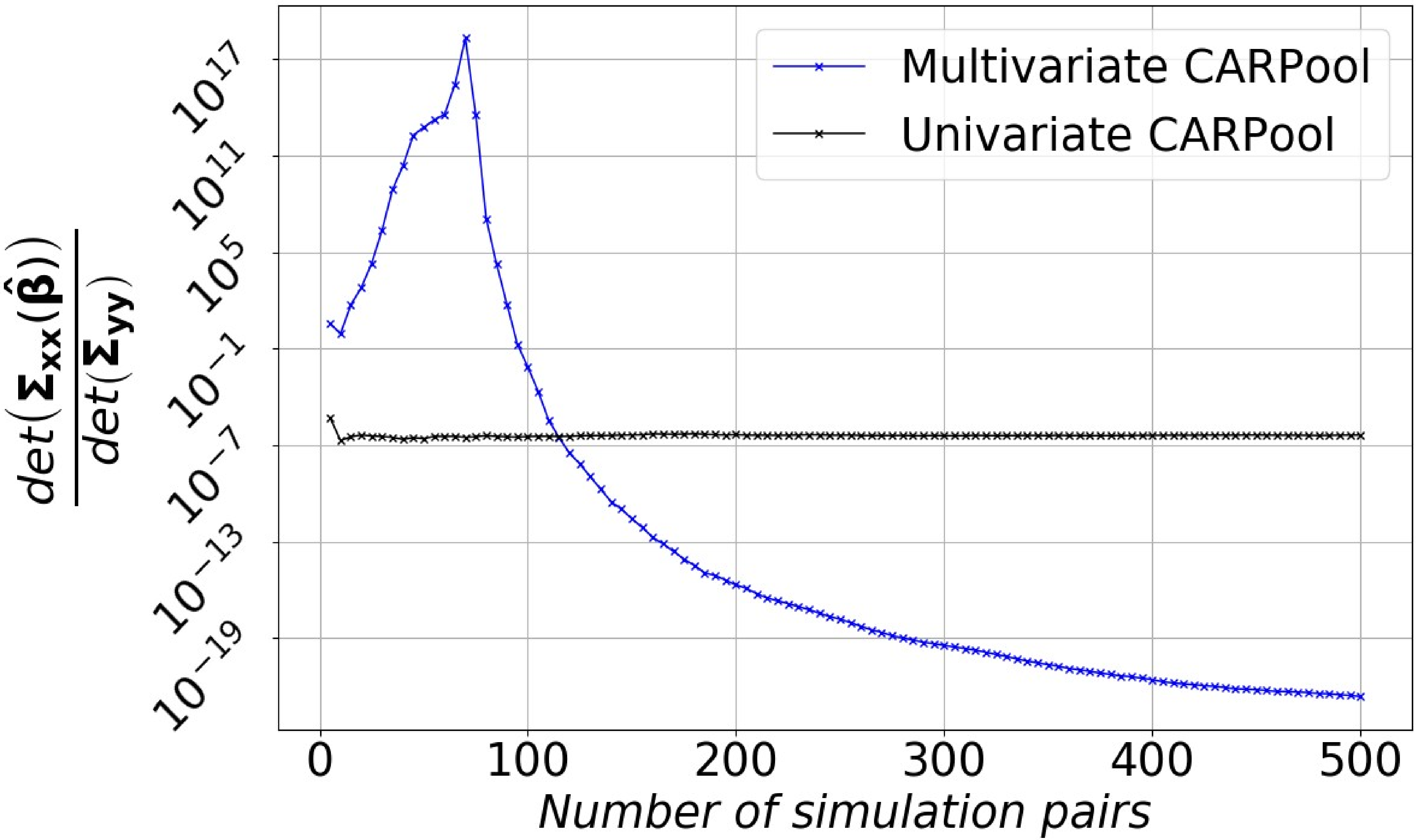
- CARPool reduces variance by factors 10 to 100, even in the non-linear regime.
- With only **5 GADGET-III simulations, CARPool** is able to compute Fourier-space two-point and three-point functions of the matter distribution at a precision comparable to **500 GADGET-III simulations**.
- We have variance reduction even for the matter PDF.
  The remapping technique proposed by *Leclercq et al. (2013)*, that increases the correlation between LPT-evolved density fields and simulations, can improve the chosen surrogate.
- CARPool can be implemented with various “N-body + surrogate” pairs. All you need is:

  1) An inexpensive surrogate and statistics computation.
  2) Strong correlation with the costly simulations.

Thank you for your attention!

(backup slides)

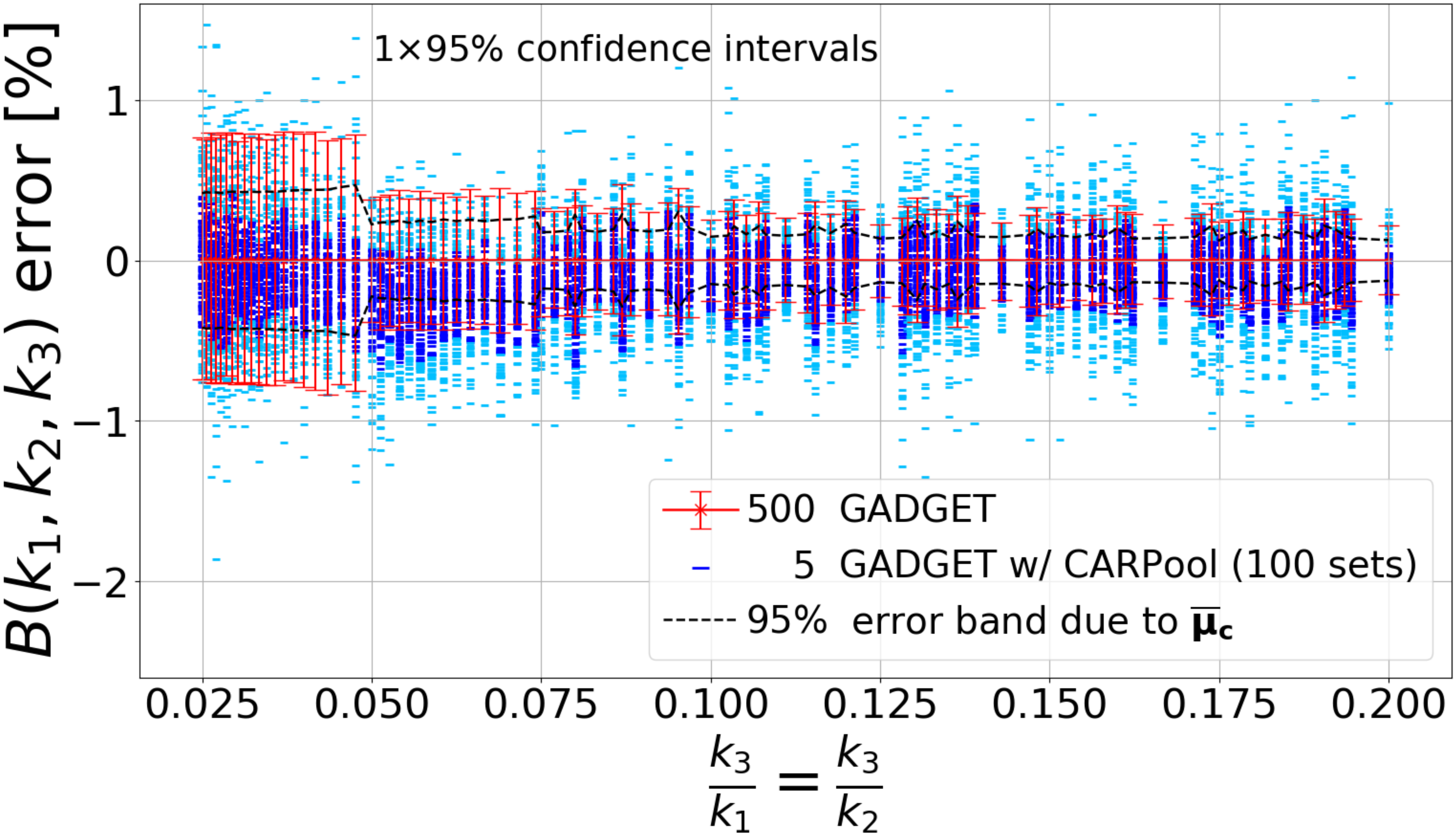
Generalized variance reduction PDF



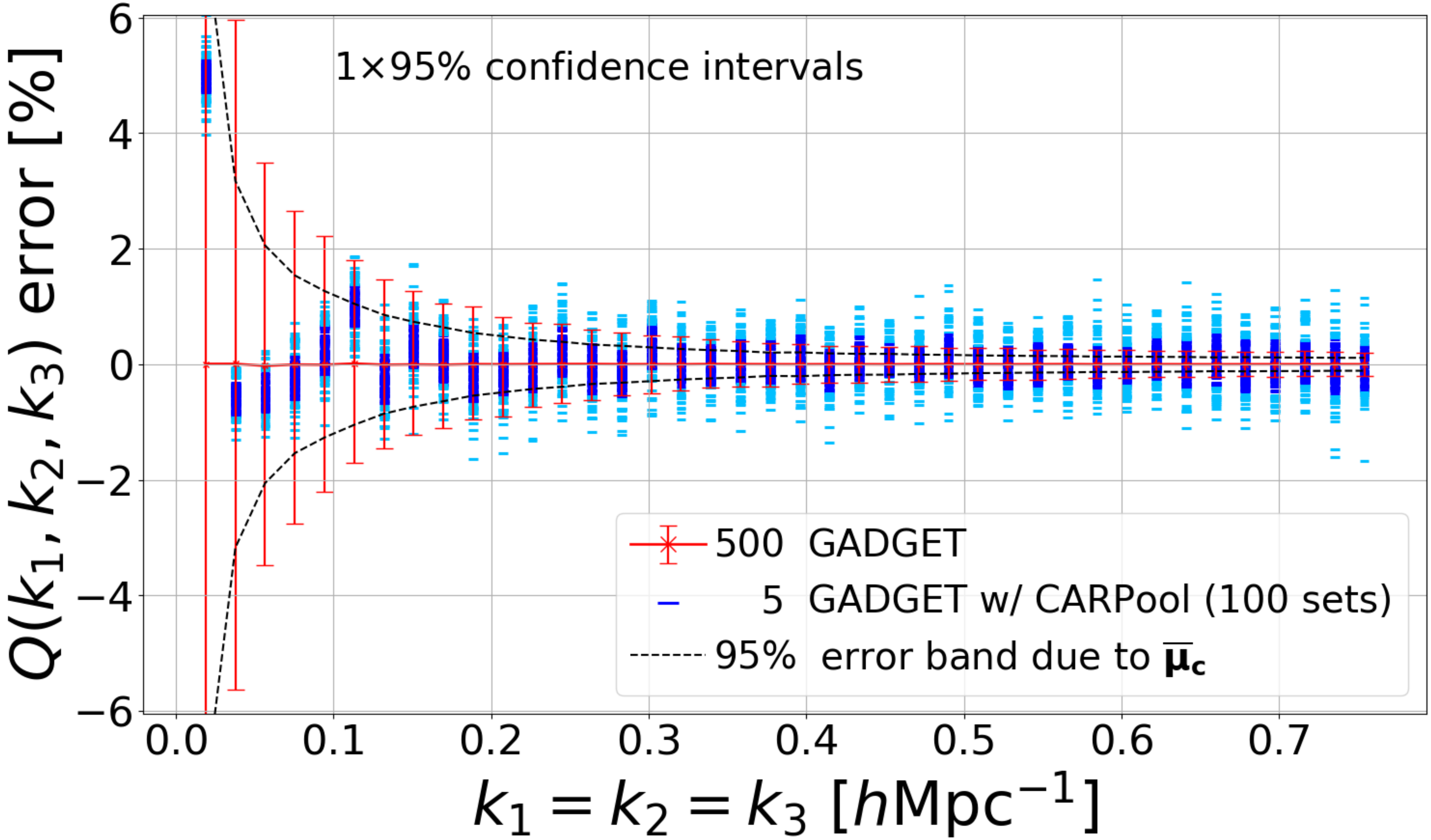
The smoothing trick

$$\beta^{diag} = \begin{pmatrix} \frac{cov(y_1, c_1)}{\sigma^2_{c_1}} & & & \\ & \frac{cov(y_2, c_2)}{\sigma^2_{c_2}} & & 0 \\ 0 & & \ddots & \\ & & & \frac{cov(y_p, c_p)}{\sigma^2_{c_p}} \end{pmatrix}$$

$$\circledast \boldsymbol{w}_{flat}$$



1×95% confidence intervals
$B(k_1, k_2, k_3)$ error [%]
1
0
−1
−2
500 GADGET
5 GADGET w/ CARPool (100 sets)
95% error band due to $\overline{\boldsymbol{\mu}}_{\mathbf{c}}$
0.025
0.050
0.075
0.100
0.125
0.150
0.175
0.200
$\frac{k_3}{k_1} = \frac{k_3}{k_2}$



1×95% confidence intervals
500 GADGET
5 GADGET w/ CARPool (100 sets)
95% error band due to $\overline{\boldsymbol{\mu}}_{\mathbf{c}}$
$Q(k_1, k_2, k_3)$ error [%]
$k_1 = k_2 = k_3$ [hMpc^{-1}]
6
4
2
0
−2
−4
−6
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7