Conformal Bootstrap and Vibe Coding

Yuan Xin
SIMIS, Fudan University

Foreword

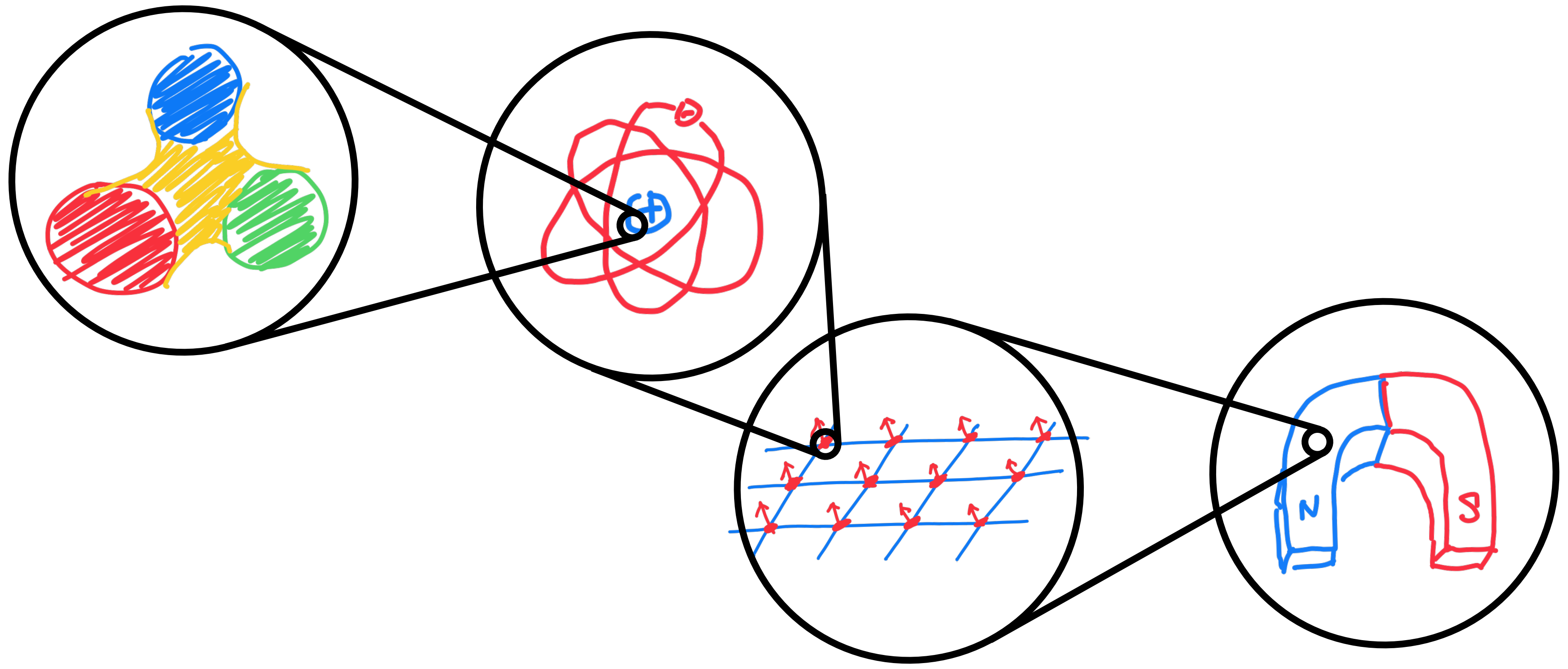
- AI is more powerful than ever.
- Numerical conformal bootstrap is one of the sharpest tools for CFT, but each incremental progress involves a lot of work which can hopefully be made easier with AI.
- It's an ongoing study - I'll show how it's motivated, designed, and where we are.

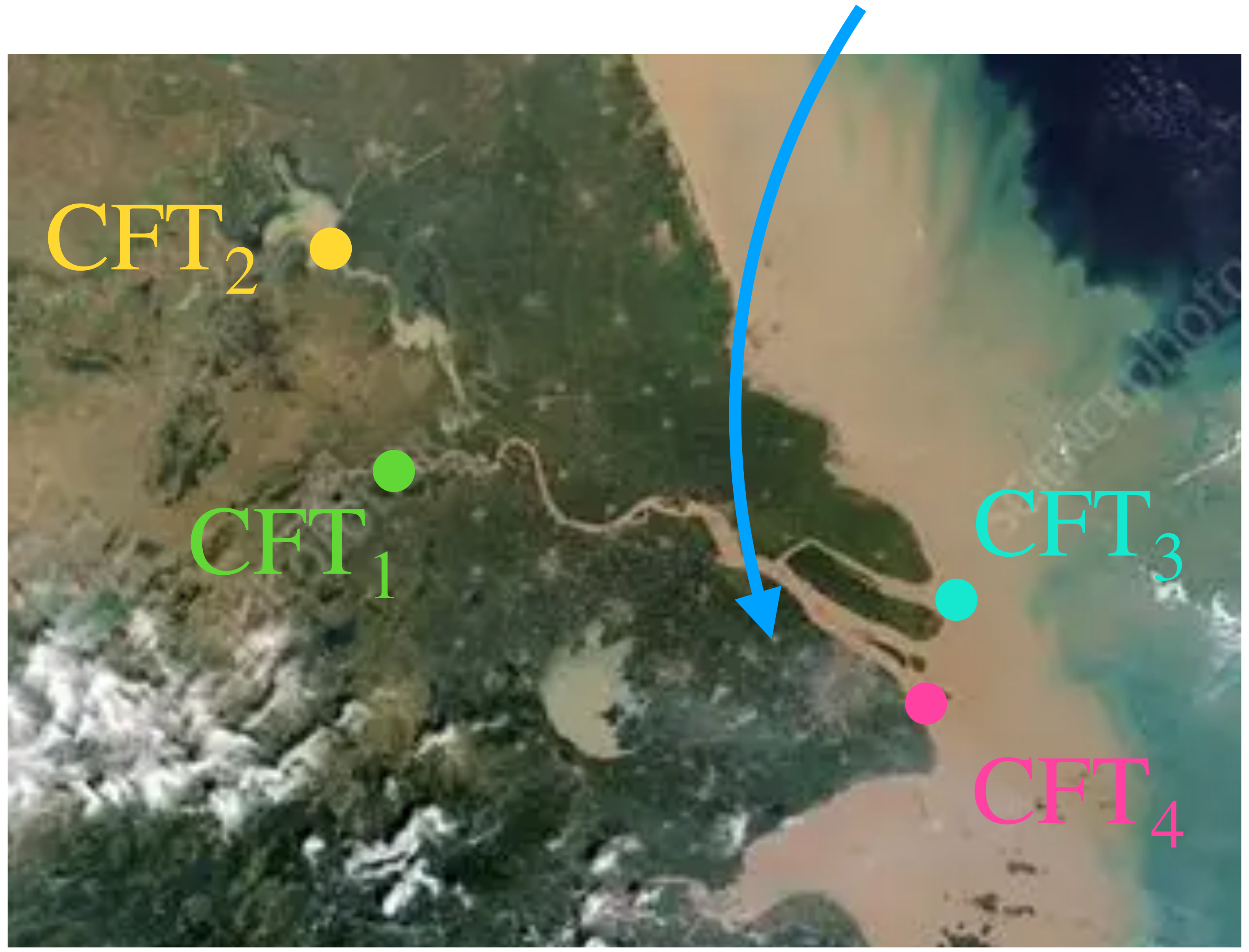
Physical world as RG flows between CFTs

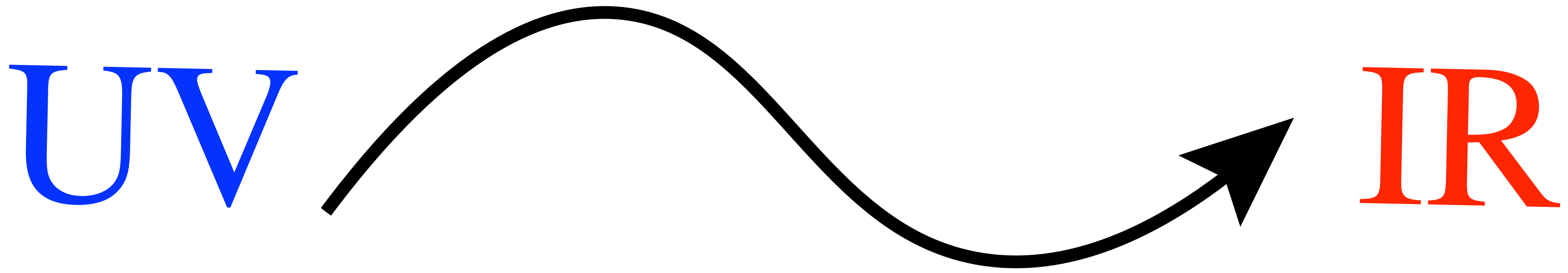
- QFTs are RG flows between CFTs.

$$S = S_{\mathrm{CFT}} + g \int d^d x \mathcal{O}_{\mathrm{rel}}(x)$$

- This is a non-perturbative definition of QFTs that hopefully helps us understand the material world.







What is our program?

- Step 1 - Solve the CFT.

- Step 2 - Solve the QFT.

- Go to bed.

What is our program?

- Step 1 - Solve the CFT.

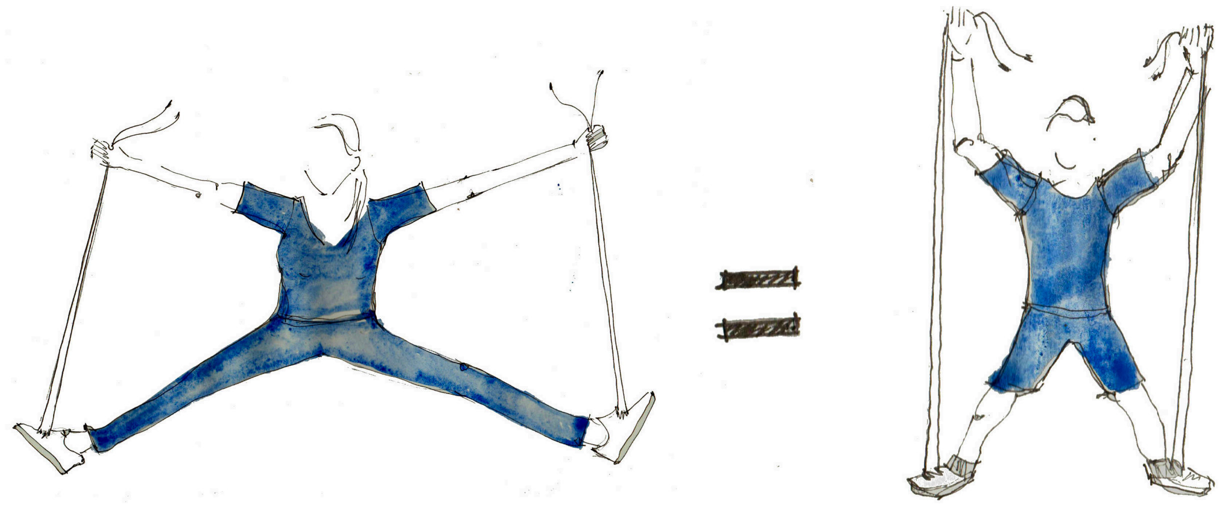
- Step 2 - Solve the QFT.

- ~~Go to bed.~~

What is our program?

- Step 1 - Solve the CFT.
- Step 2 - Solve the QFT.

- Conformal bootstrap, simulation...



$$\langle \mathcal{O}_1\mathcal{O}_2\mathcal{O}_3\mathcal{O}_4\rangle = \langle \mathcal{O}_1\mathcal{O}_2\mathcal{O}_3\mathcal{O}_4\rangle$$

$$\mathcal{O}_1\mathcal{O}_2 = \sum_i \lambda_{12i} B_i(x_{12},\partial)\mathcal{O}_i$$

Crossing: $$\langle \mathcal{O}_1\mathcal{O}_2\mathcal{O}_3\mathcal{O}_4\rangle = \sum_i \lambda_{12i}\lambda_{34i} G^{1234}_{\Delta_i,\ell_i}\left(\frac{x_{12}^2x_{34}^2}{x_{13}^2x_{24}^2}, \frac{x_{14}^2x_{23}^2}{x_{13}^2x_{24}^2}\right)$$

Positivity: $$\lambda^2 \geq 0$$ Constraints on Δ_i, λ_{12i}

What is our program?

- Step 1 - Solve the CFT.

Obtained arbitrarily many $\{\, \Delta_i, \lambda_{12i} \,\}$

- Step 2 - Solve the QFT.

$$S = S_{\text{CFT}} + g \int d^d x \mathcal{O}_{\text{rel}}(x)$$

$$H = H_{\text{CFT}} + gV$$

$$\langle \mathcal{O}_i | H | \mathcal{O}_j \rangle = \langle \mathcal{O}_i | H_{\text{CFT}} | \mathcal{O}_j \rangle + g \langle \mathcal{O}_i | V | \mathcal{O}_j \rangle$$

$\delta_{ij} \Delta_i$

$\lambda_{ij,\mathcal{O}_{\text{rel}}}$

Radial Quantization

$$H|n\rangle = E_n |n\rangle$$

2d: Yurov, Zamolodchikov \`90, \`91
+ many more
d>2: Hogevorst, Rychkov, van Rees \`14
Elias-Miro, Hardy, \`20
Others..

“Truncated Conformal Space Approach”

What is our program?

- Step 1 - Solve the CFT.

Obtained arbitrarily many $\{\, \Delta_i, \lambda_{12i} \,\}$

- Step 2 - Solve the QFT.

$$S = S_{\rm CFT} + g\int d^dx\,\mathcal{O}_{\rm rel}(x)$$

$$H = H_{\rm CFT} + gV$$

$$\langle \mathcal{O}_i | H | \mathcal{O}_j \rangle = \langle \mathcal{O}_i | H_{\rm CFT} | \mathcal{O}_j \rangle + g \langle \mathcal{O}_i | V | \mathcal{O}_j \rangle$$

Beyond the scope of this project.

But we'll discuss what it requires

from the bootstrap.

Radial Quantization

$$H|n\rangle = E_n|n\rangle$$

2d: Yurov, Zamolodchikov `90, `91
+ many more
d>2: Hogevorst, Rychkov, van Rees `14
Elias-Miro, Hardy, `20
Others..

"Truncated Conformal Space Approach"

An example fully worked out - 2D Ising Field

- CFT side:

Belavin, Polyakov, Zamolodchikov `84
Or any standard CFT textbook

∞-dimensional Virasoro Algebra L_m

$$[L_n, L_m] = (n-m)L_{n+m} + \frac{c}{12}(n^3 - n)\delta_{n+m,0}$$

Positivity of gram matrix

$$\langle h|L_{\{M\}}L_{\{-N\}}|h\rangle \succeq 0$$

Has exact solution for $c = 1/2$

$$\Delta = 0, \frac{1}{8}, 1$$ + descendants

- QFT side:

Yurov, Zamolodchikov `90, `91

$$H_{\mathrm{IFT}} = \frac{1}{2\pi}\left(H_{\mathrm{CFT}} + \int dx\ m\epsilon(x) + g\sigma(x)\right)$$

$$\mathrm{H} = \begin{array}{cc} \begin{pmatrix} \frac{1}{8}+\frac{m}{2} & m & m & g & \frac{g}{2} & \frac{g}{8} & \frac{g}{64} \\ m & \frac{17}{8}+\frac{m}{2} & 0 & \frac{g}{8} & \frac{9g}{16} & \frac{25g}{64} & \frac{225g}{512} \\ m & 0 & \frac{33}{8}+\frac{m}{2} & \frac{9g}{128} & \frac{25g}{256} & \frac{441g}{1024} & \frac{1225g}{8192} \\ g & \frac{g}{8} & \frac{9g}{128} & 0 & m & m & 0 \\ \frac{g}{2} & \frac{9g}{16} & \frac{25g}{256} & m & 1 & 0 & m \\ \frac{g}{8} & \frac{25g}{64} & \frac{441g}{1024} & m & 0 & 3 & m \\ \frac{g}{64} & \frac{225g}{512} & \frac{1225g}{8192} & 0 & m & m & 4 \end{pmatrix} & \begin{array}{l} |\sigma\rangle \\ |\partial^2\sigma\rangle \\ |\partial^4\sigma\rangle \\ |1\rangle \\ |\epsilon\rangle \\ |\partial^2\epsilon\rangle \\ |T\bar{T}\rangle \end{array} \end{array}$$

An example fully worked out - 2D Ising Field

- CFT side:

Belavin, Polyakov, Zamolodchikov `84
Or any standard CFT textbook

∞-dimensional Virasoro Algebra L_m

$$[L_n, L_m] = (n-m)L_{n+m} + \frac{c}{12}(n^3 - n)\delta_{n+m,0}$$

Positivity of gram matrix

$$\langle h|L_{\{M\}}L_{\{-N\}}|h\rangle \succeq 0$$

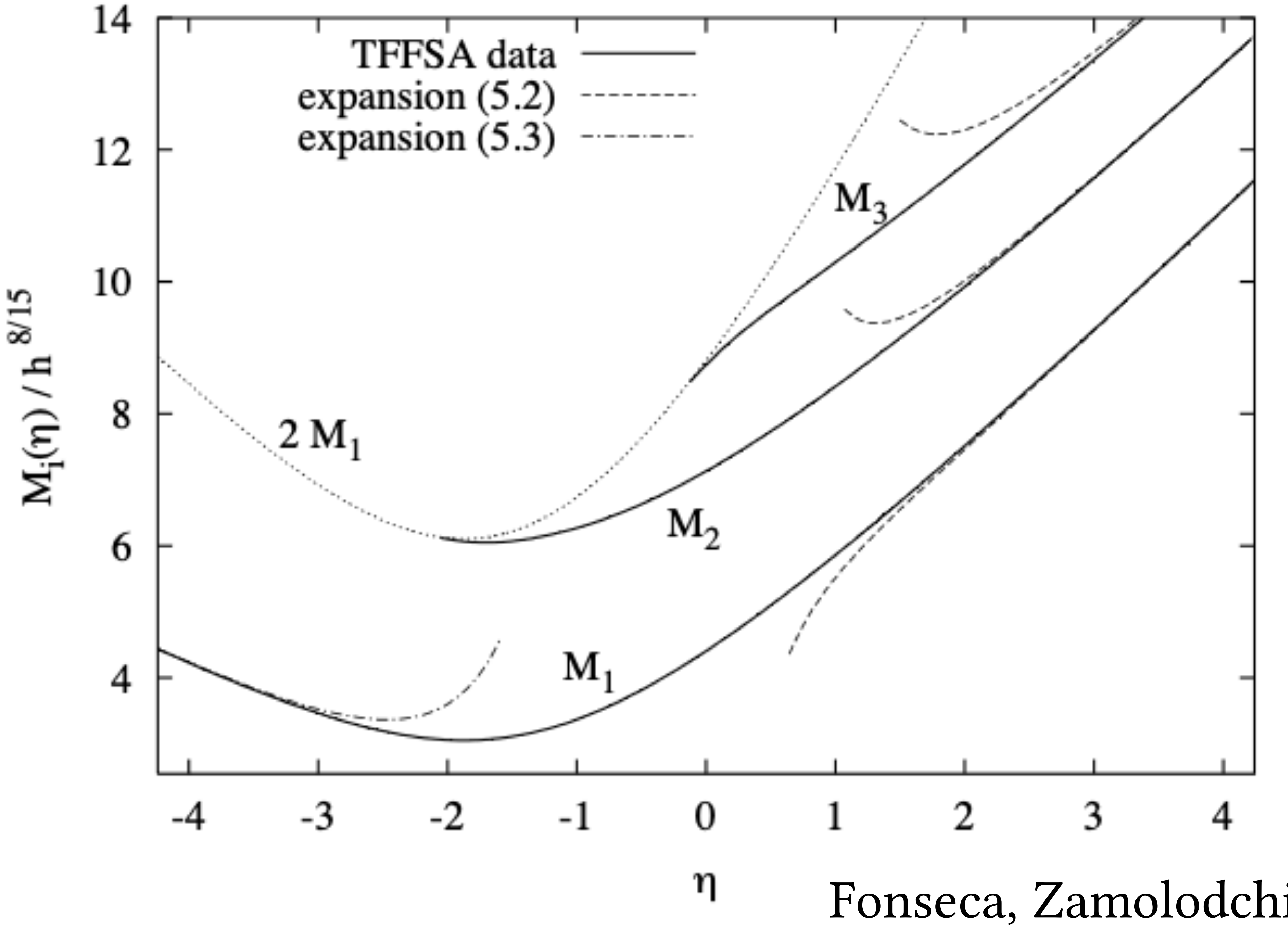
Has exact solution for $c = 1/2$

$$\Delta = 0, \frac{1}{8}, 1$$ + descendants

- QFT side:

Yurov, Zamolodchikov `90, `91

$$H_{\text{IFT}} = \frac{1}{2\pi}\left(H_{\text{CFT}} + \int dx \; m\epsilon(x) + g\sigma(x)\right) \qquad \eta = \frac{m}{g^{8/15}}$$



Fonseca, Zamolodchikov `01

Challenges

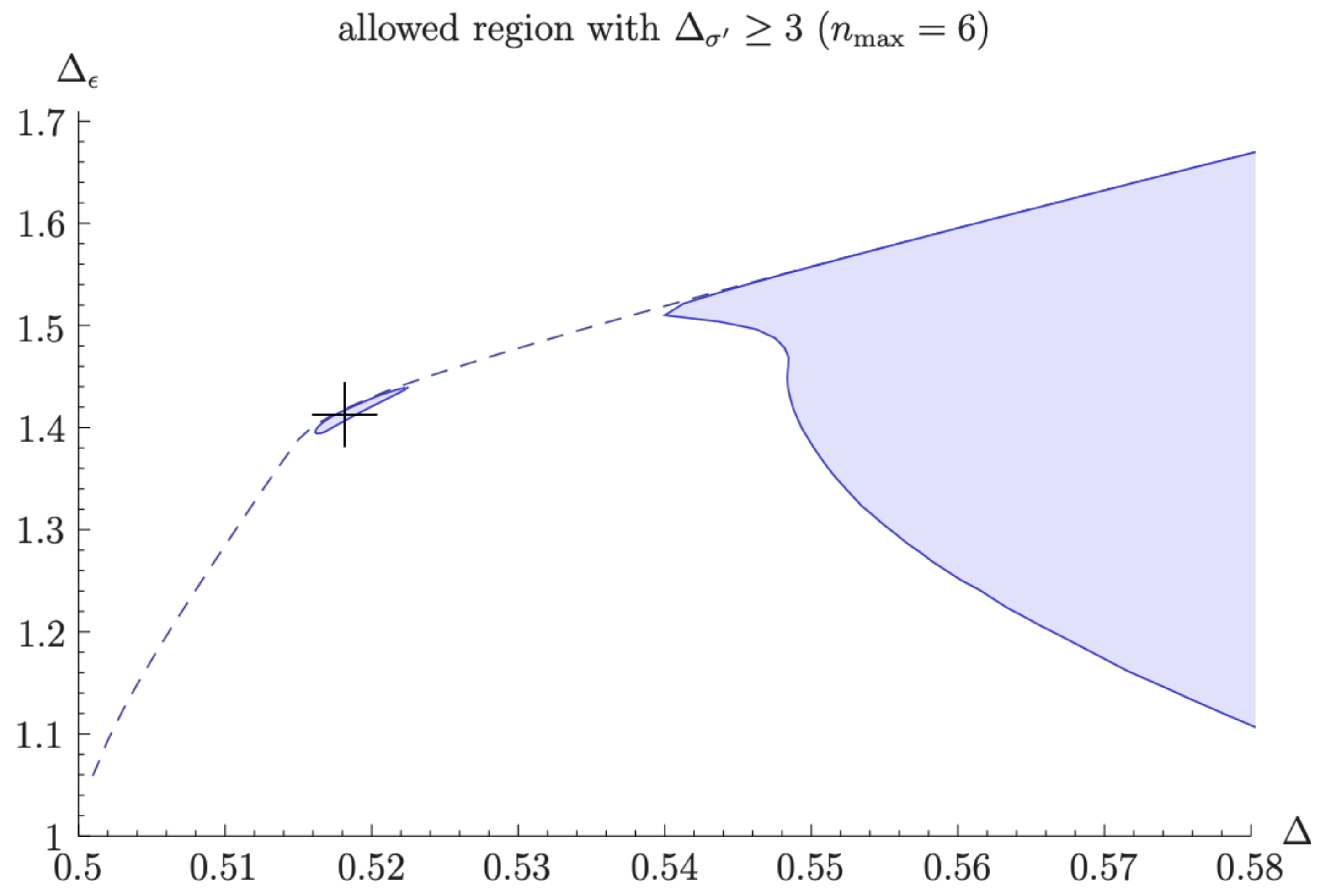
- Many theories
- A lot of CFT data

Challenges

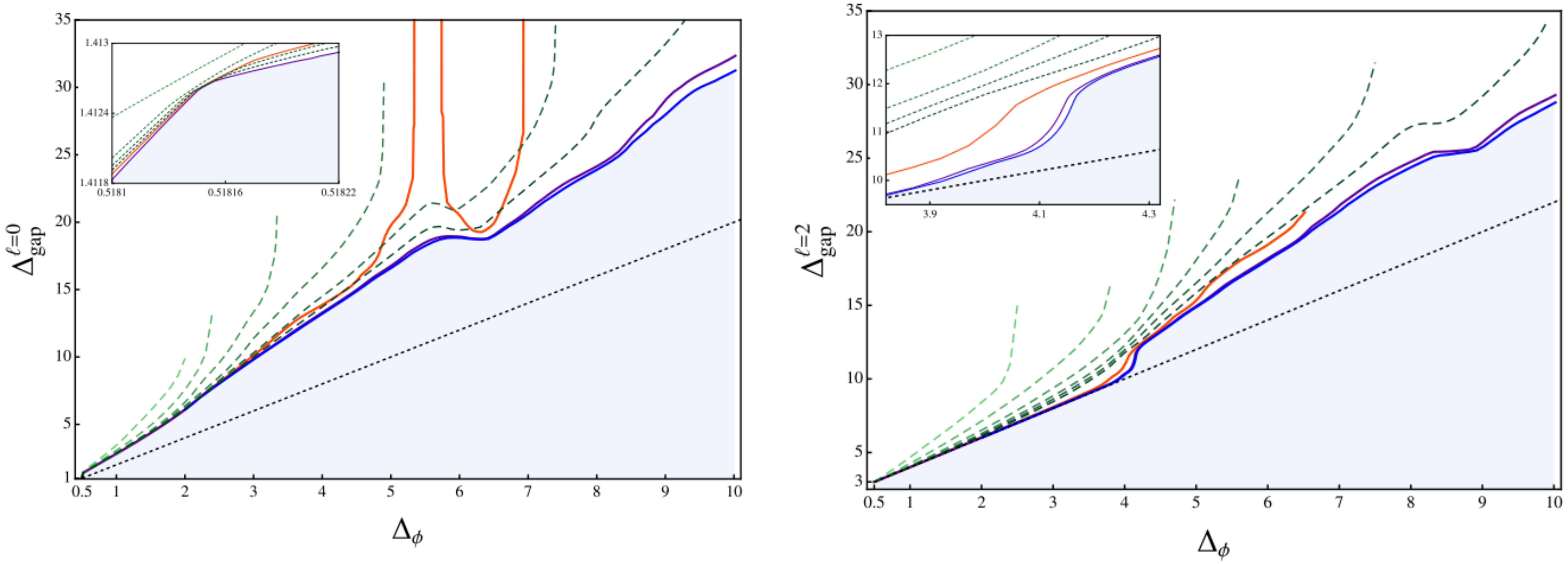
- Many theories - we'd like to isolate more theories

3D theories with $\mathbb{Z}_2$ symmetry: seem nothing is happening for larger Δ_σ

State-of-art technology shows there are much more interesting structures



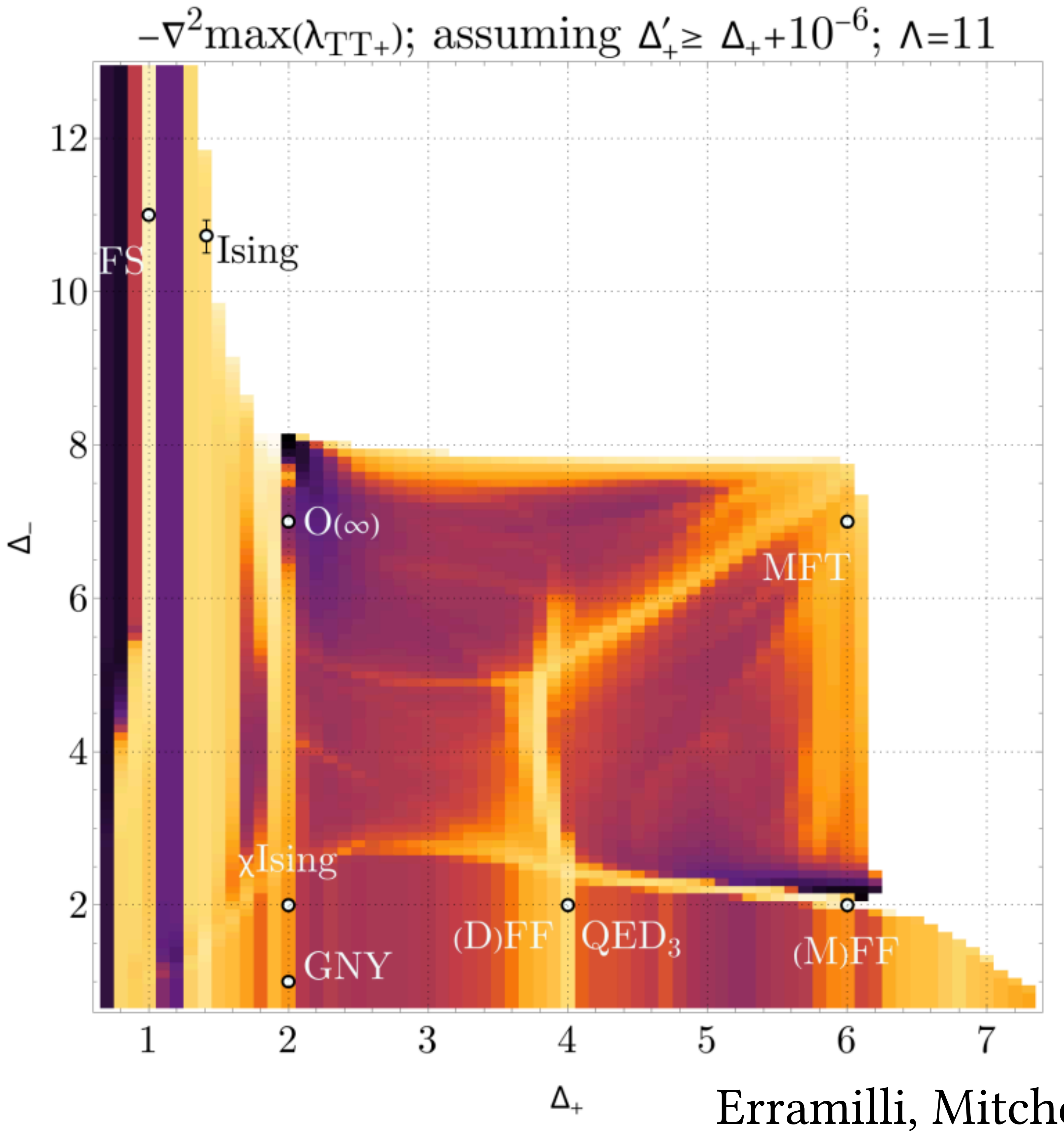
Kos, Poland, Simmons-Duffin `14



Ghosh, Zheng `26

Challenges

- Many theories - we'd like to isolate more theories



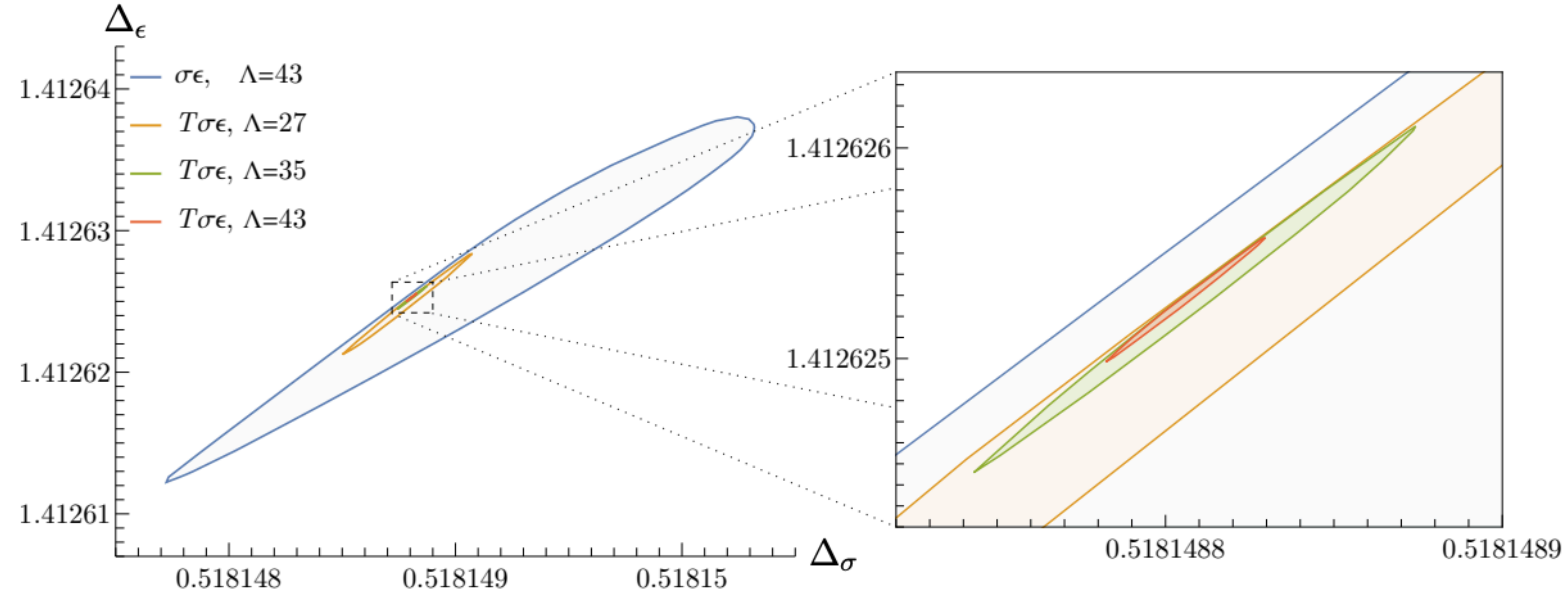
Erramilli, Mitchel `26

A "universal" map of the entire 3D CFT landscape from stress tensor bootstrap $\langle TTTT \rangle$.

OPE coefficient upper bound reveal many interesting structures,

Challenges

- A lot of CFT data



State-of-art bootstrap bound of 3D Ising model is a size-10^{-10} island.

In terms of CFT data, these are the only parameters that are bounded:

What about e.g. $\lambda_{\epsilon'\epsilon\epsilon'}$?

	$T\sigma\epsilon$ (this work)	prev. bootstrap	monte carlo	fuzzy sphere
Δ_σ	0.518148806**(24)**	0.5181489(10)* [11]	0.518142(20) [20]	0.524 [21], 0.519 [22]
Δ_ϵ	1.41262528**(29)**	1.412625(10)* [11]	1.41265(13) [20]	1.414 [21], 1.403 [22]
$\lambda_{\sigma\sigma\epsilon}$	1.05185373(11)	1.0518537(41) [11]	1.051(1) [23]	1.0539(18) [24]
$\lambda_{\epsilon\epsilon\epsilon}$	1.53244304(58)	1.532435(19) [11]	1.533(5) [23]	1.5441(23) [24]
c_T/c_B	0.946538675(42)	0.946534(11) [25]	0.952(29) [26]	0.955(21) [24]
$\lambda_{TT\epsilon}$	0.95331513(42)	0.958(7) [27]	—	0.9162(73) [24]
n_B	0.933444559(75)	0.933(4)** [19]	—	—
n_F	0.013094116(33)	0.014(4)** [19]	—	—

Chang, Dommes, Erramilli, Homrich, Kravchuk, Liu, Mitchell, Poland, Simmons-Duffin, `24

Challenges

$$H = H_{\mathrm{CFT}} + gV$$

$$\langle \mathcal{O}_i | H | \mathcal{O}_j \rangle = \langle \mathcal{O}_i | H_{\mathrm{CFT}} | \mathcal{O}_j \rangle + g \langle \mathcal{O}_i | V | \mathcal{O}_j \rangle$$

- A lot of CFT data:
  - We see that the data required from the QFT side is $\lambda_{ij,\mathcal{O}_{\mathrm{rel}}}$,
  - meaning in general both i and j can go to higher dimension.
  - Standard σ, ϵ mixing allows only 1 operator in OPE coefficient to get heavy.
  - We will need to add ϵ' into the mixed bootstrap study.
  - What would be the best strategy for this? Is there any measurement of how far we can go? It will require a lot of experiments.

From simulation, but not bootstrap

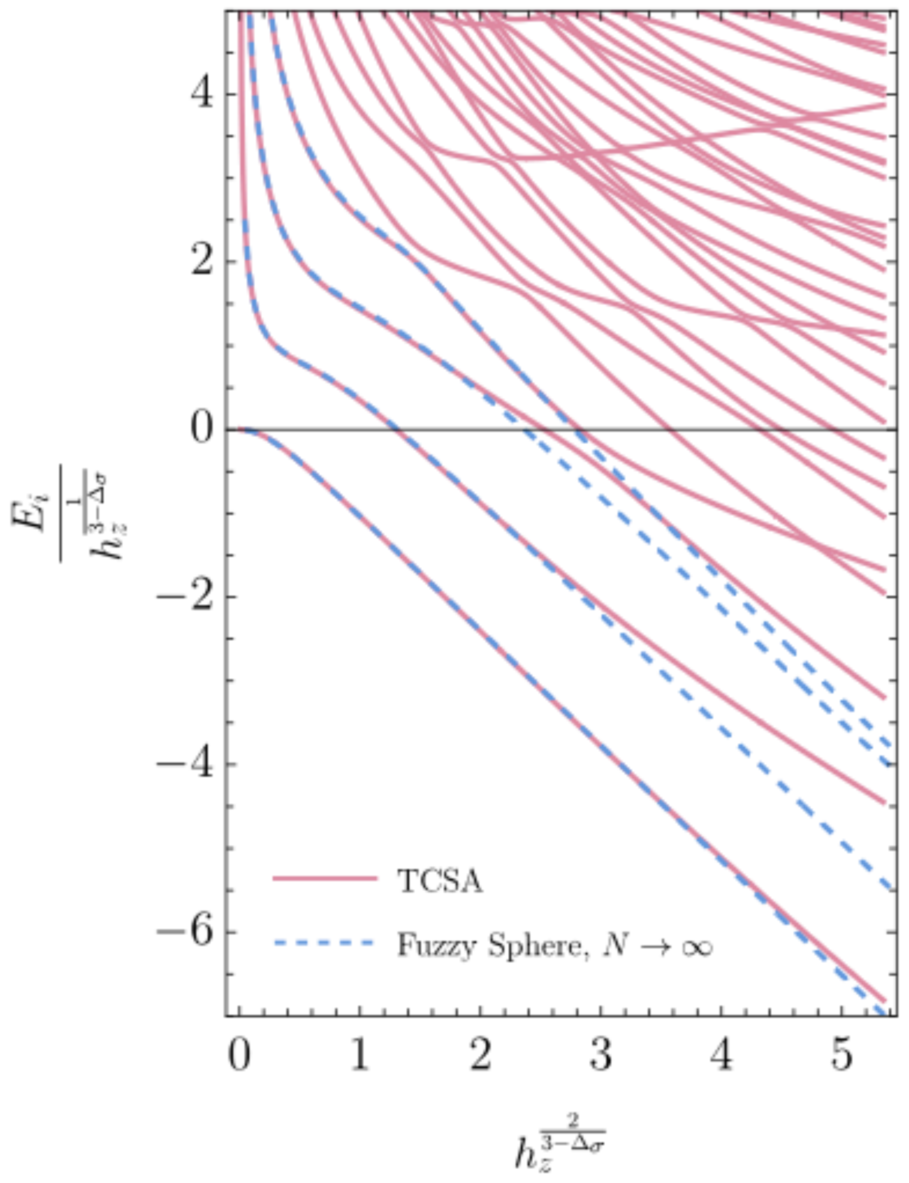
3D Ising CFT
Synthesized Fuzzy Sphere Data

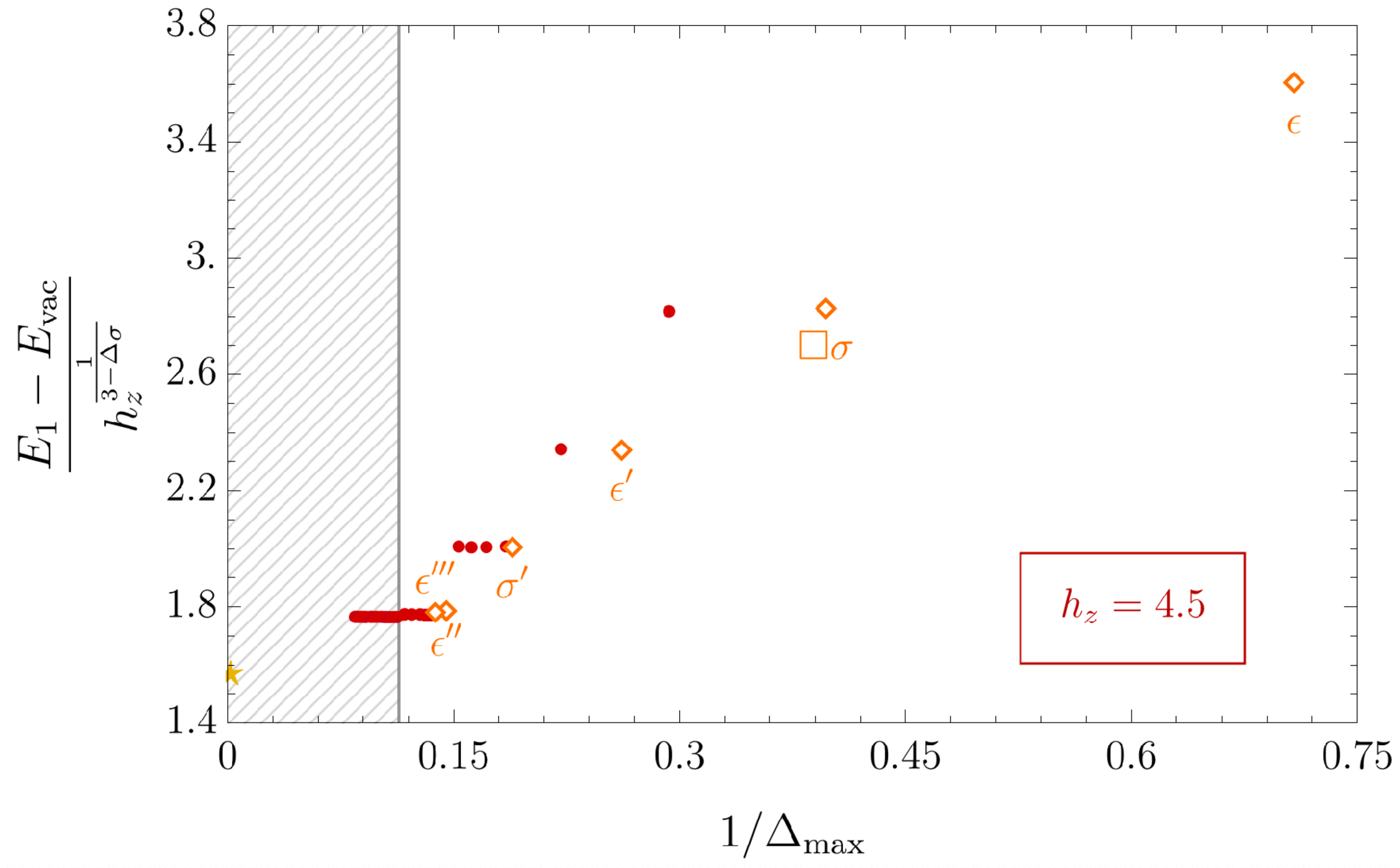
$\boldsymbol{\sigma}$	σ	σ'	$\sigma_{\mu_1\mu_2}$	$\sigma'_{\mu_1\mu_2}$
1	1	0	0	0
ϵ	1.05185	−0.057235	0.778319	0.0348
ϵ'	−0.053012	1.270	0.397	−0.823
ϵ''	−0.00073	1.350	−0.118	0.411
ϵ'''	−0.0002	0.180	0.434	0.027
$T'_{\mu_1\mu_2}$	0.02114	0.392	{−0.114,0.455,1.070}	{−0.229,0.344,1.300}
$T''_{\mu_1\mu_2}$	0.00096	−0.012	{0.174,1.640,0.517}	{0.015,−0.404,−0.223}
$C_{\mu_1\mu_2\mu_3\mu_4}$	0.276304	0.029	{−0.023,0.209,−0.224}	{−0.017,0.060,0.025}
$C'_{\mu_1\mu_2\mu_3\mu_4}$	0.00784	0.082	{0.058,−0.147,−0.809}	{−0.044,0.155,0.102}
$C''_{\mu_1\mu_2\mu_3\mu_4}$	0.009508	0.197	{−0.092,0.210,0.873}	{−0.231,0.284,−0.221}

Fitzpatrick, Katz, Fardelli, `26

3D Ising Field Theory

$$H_{\rm IFT} = H_{\rm CFT} + \int_{\rm sphere} d^2x\ h\sigma(x)$$



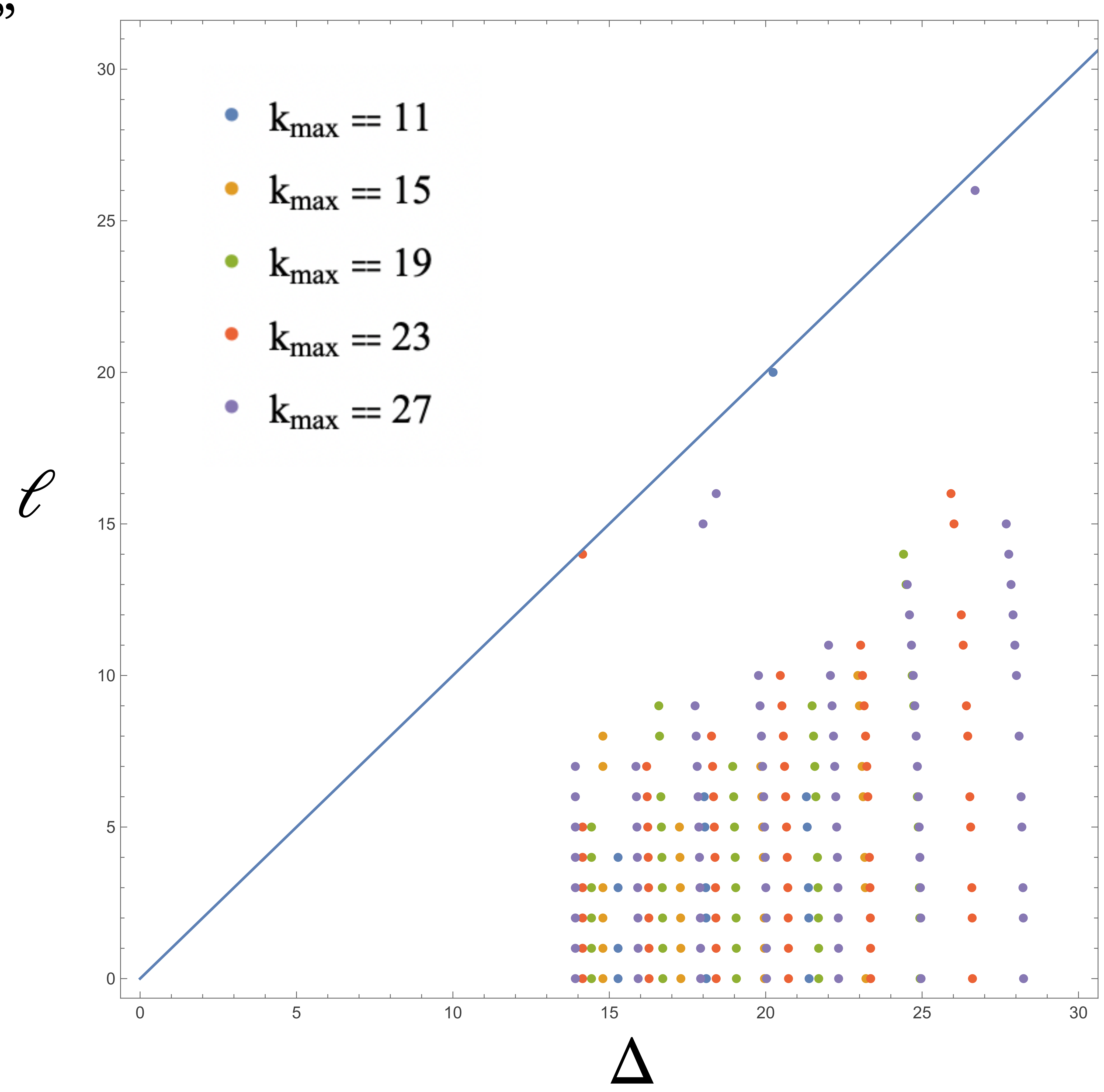


Fitzpatrick, Katz, Fardelli, YX, to appear

An "AlphaFold" for numerical bootstrap?

- Possible - through "extremal functional method"
- Additional challenge from the precise data.
- The numerical bootstrap solutions need to be almost exactly correct in order to provide a useful certified bound.
- Many spurious structures show up making it impossible to fit.

Erramilli `26



Toward an AI-assisted bootstrap scheme

- It has been hard to come up with a good fit of bootstrap solutions.
- LLM gives a new hope: instead of fitting data, we just directly let AI learn how to use the numerical bootstrap tools.
- There are established programs that cover every phase of conformal bootstrap study, but it is a challenge for everyone who learn them from scratch.

Toward an AI-assisted bootstrap scheme

- Take minimal steps.
- Goal: compute the unknown OPE coefficient $\lambda_{\epsilon'\epsilon'\epsilon}$ and $\lambda_{\epsilon'\epsilon'\epsilon'}$ with low-order bootstrap setups.
- A full-blown setup will include all possible combination of the 3 external operators σ, ϵ and ϵ'.
- A simplified setup is between σ and ϵ'.
- Before trying anything new, we will see if we can reproduce the old σ and ϵ result, and use that to verify the conformal block normalization.

	$T\sigma\epsilon$ (this work)
Δ_σ	0.518148806(**24**)
Δ_ϵ	1.41262528(**29**)
$\lambda_{\sigma\sigma\epsilon}$	1.05185373(11)
$\lambda_{\epsilon\epsilon\epsilon}$	1.53244304(58)
c_T/c_B	0.946538675(42)
$\lambda_{TT\epsilon}$	0.95331513(42)
n_B	0.933444559(75)
n_F	0.013094116(33)

Chang, et al, `24

	Rigorous bounds	
Δ_σ	0.518157(**35**)	$\Lambda = 19$
Δ_ϵ	1.41265(**36**)	$\Lambda = 19$
$\lambda_{\sigma\sigma\epsilon}$	1.05185(**12**)	$\Lambda = 19$
$\lambda_{\epsilon\epsilon\epsilon}$	1.53240(**58**)	$\Lambda = 19$
$\Delta_{\epsilon'}$	3.82951(**61**)	$\Lambda = 31$
$\lambda_{\sigma\sigma\epsilon'}$	0.05304(**16**)	$\Lambda = 19$
$\lambda_{\epsilon\epsilon\epsilon'}$	1.5362(**12**)	$\Lambda = 19$
$\Delta_{\sigma'}$	5.262(**89**)	$\Lambda = 19$
$\lambda_{\sigma\epsilon\sigma'}$	0.0565(**15**)	$\Lambda = 19$
$\frac{c_T}{c_T^{\text{free}}}$	0.946543(**42**)	$\Lambda = 19$
$\Delta_{T'}$	5.499(**17**)	$\Lambda = 19$
$\lambda_{\sigma\sigma T'}$	0.02107(**20**)	$\Lambda = 19$
$\lambda_{\epsilon\epsilon T'}$	1.355(**30**)	$\Lambda = 19$

Reehorst, `21

On the learning curve of bootstrap

$$\langle \mathcal{O}_1 \mathcal{O}_2 \mathcal{O}_3 \mathcal{O}_4 \rangle = \langle \mathcal{O}_3 \mathcal{O}_2 \mathcal{O}_1 \mathcal{O}_4 \rangle$$

$$\langle \mathcal{O}_1 \mathcal{O}_2 \mathcal{O}_3 \mathcal{O}_4 \rangle = \sum_i \lambda_{12i} \lambda_{34i} G^{1234}_{\Delta_i, \ell_i} \left(\frac{x_{12}^2 x_{34}^2}{x_{13}^2 x_{24}^2}, \frac{x_{14}^2 x_{23}^2}{x_{13}^2 x_{24}^2} \right)$$

$$\sum_{\mathcal{O}+} \left(\lambda_{\sigma\sigma\mathcal{O}} \quad \lambda_{\epsilon\epsilon\mathcal{O}}\right) \vec{V}_{+,\Delta,\ell} \begin{pmatrix} \lambda_{\sigma\sigma\mathcal{O}} \\ \lambda_{\epsilon\epsilon\mathcal{O}} \end{pmatrix} + \sum_{\mathcal{O}-} \lambda^2_{\sigma\epsilon\mathcal{O}} \vec{V}_{-,\Delta,\ell} = 0,$$

Polynomial matrix vectors

`scalar_blocks` for computing the conformal blocks and expand them in Taylor series around the crossing self-dual point. `blocks_3d` if the external operators are spinning.

Number of poles in rational approximation
Order of approximation
Derivatives in Taylor expansion
Numerical precision

`Autoboot` for generating the minimal set of linearly independent crossing equations and separate the OPE according to the representation theory

`Haskell or Simpleboot` for wrapping the conformal block Taylor expansion information with the prefactors in the crossing equations which respects the representation theory.

`sdpb` for actually computing the semidefinite programming.

The biggest overhead is gone

- Days or weeks -> 2.5 hours

```
# Goal: learn autoboot

## Context
The goal is to learn how to use autoboot to generate input for crossing
equations and integrate in haskell bootstrap routines. Begin with
understanding the input and output of autoboot. Then see if you can connect
autoboot output to haskell setup and produce bootstrap bounds for small
derivative number (say Lambda=11). Start with single correlator bootstrap and
```

⋮

```
## Resource
Autoboot repository and paper
https://github.com/selpoG/autoboot/
https://arxiv.org/abs/1903.10522
Tutorial for bootstrap tools including simpleboot and haskell setups
https://gitlab.com/AikeLiu/Bootstrap-Mini-Course
Ning Su's simpleboot. Autoboot is integrated.
https://gitlab.com/bootstrapcollaboration/simpleboot
A more recent haskell setup
https://gitlab.com/davidsd/scalars-3d
O(N) bootstrap
https://arxiv.org/abs/1603.04436
O(2) precision bootstrap with 3 operator mixing
https://arxiv.org/abs/1912.03324
```

```
**🤖 claude · `05-19 06:42:14`**

Plan approved. Executing step 00 (copy PLAN.md, scaffold directories, write
task stubs), then launching step 01.

**🤖 claude · `05-19 09:17:00`**

**Pipeline complete.** Summary of what landed:

## Working end-to-end pipeline (autoboot → Haskell SDPB → bounds)

| Step | Result |
|---|---|
| 01 env-setup | ✓ both apptainers profiled, autoboot cloned |
| 02 autoboot-io | ✓ O(3) σσσσ crossing equations in `.m` + `.json`;
coefficients exactly match arXiv:1603.04436 |
| 02v verify | **PASS** |
| 03 scalars-3d | ✓ cloned + 25+ symbol api-map; **built end-to-end via 10
successive fixes** for missing foreign libs (libmpfr-dev, libgmp-dev, libblas/
lapack-dev, zlib, libqhull, libgfortran.so.5) — all extracted as local .debs,
zero host writes |
```

The detailed setup vibed

- The benefit of setting up AI assisted bootstrap program is that you can use natural language to dictate the detailed spectrum assumptions:

- Isolation of operators:
  - From
$$\sum_{\mathcal{O}^+} \left(\lambda_{\sigma\sigma\mathcal{O}} \quad \lambda_{\epsilon\epsilon\mathcal{O}}\right) \vec{V}_{+,\Delta,\ell} \begin{pmatrix} \lambda_{\sigma\sigma\mathcal{O}} \\ \lambda_{\epsilon\epsilon\mathcal{O}} \end{pmatrix} + \sum_{\mathcal{O}^-} \lambda^2_{\sigma\epsilon\mathcal{O}} \vec{V}_{-,\Delta,\ell} \;=\; 0,$$
  - To
$$(1 \ 1)\, \vec{V}_{\text{vac}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} + (\lambda_{\sigma\sigma\epsilon} \ \lambda_{\epsilon\epsilon\epsilon})\, \vec{V}_{\text{ext}} \begin{pmatrix} \lambda_{\sigma\sigma\epsilon} \\ \lambda_{\epsilon\epsilon\epsilon} \end{pmatrix} + (\lambda_{\sigma\sigma T} \ \lambda_{\epsilon\epsilon T})\, \vec{V}_T \begin{pmatrix} \lambda_{\sigma\sigma T} \\ \lambda_{\epsilon\epsilon T} \end{pmatrix}$$
$$+ \sum_{\mathcal{O}^+} (\lambda_{\sigma\sigma\mathcal{O}} \ \lambda_{\epsilon\epsilon\mathcal{O}})\, \vec{V}_{+,\Delta,\ell} \begin{pmatrix} \lambda_{\sigma\sigma\mathcal{O}} \\ \lambda_{\epsilon\epsilon\mathcal{O}} \end{pmatrix} + \sum_{\mathcal{O}^-} \lambda^2_{\sigma\epsilon\mathcal{O}} \vec{V}_{-,\Delta,\ell} = 0$$

- Make the setup stronger by putting in $\lambda_{\sigma\sigma\epsilon}$, $\lambda_{\epsilon\epsilon\epsilon}$, c_T at the accurate value.

Matching conditions - remarkable success

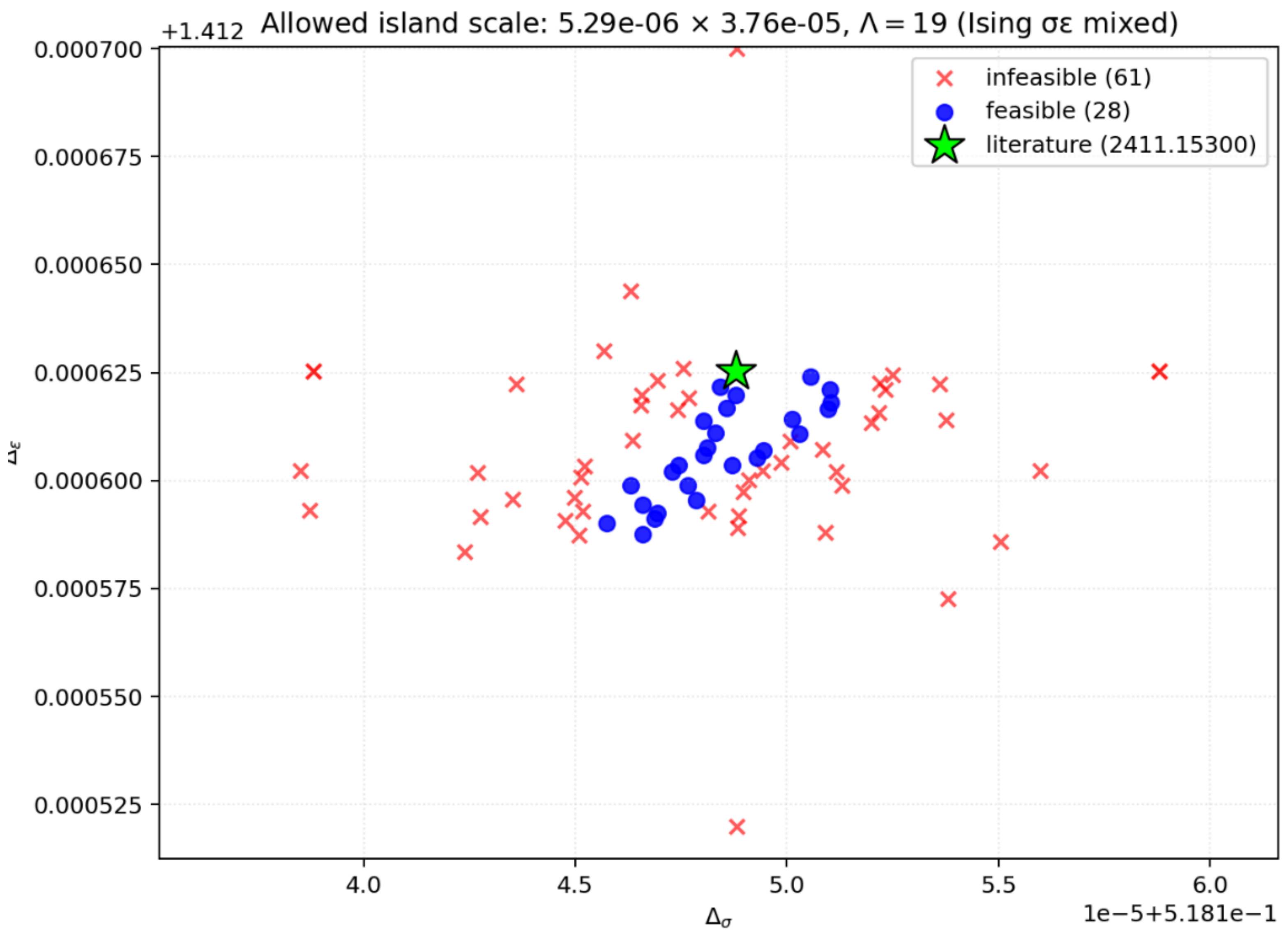
- When merging different numerical packages, one of the frequently seen issue is the mismatch of normalization.

- The conformal block may contain a (Δ, ℓ) - dependent prefactor.

$$g^{\Delta_{12},\Delta_{34}}_{\Delta,\ell}(r,\eta) \underset{r\to 0}{\sim} \mathcal{N}_{d,\ell}(4r)^{\Delta}\mathrm{Geg}_{\ell}(\eta) ,$$

$\mathcal{N}_{d,\ell}$	Reference
$\frac{\ell!}{(-2)^\ell (d/2-1)_\ell}$	Dolan and Osborn (2001b, 2004), Rattazzi *et al.* (2008), Penedones *et al.* (2016), **this review**
$\frac{\ell!}{(d-2)_\ell}$	Dolan and Osborn (2011), Hogervorst and Rychkov (2013), El-Showk *et al.* (2012, 2014b), Costa *et al.* (2016b), JuliBoots (Paulos, 2014b), cboot (Ohtsuki, 2016)
$\frac{(-1)^\ell \ell!}{4^\Delta (d/2-1)_\ell}$	Kos *et al.* (2014a, 2015b, 2016), Li *et al.* (2017b) PyCFTBoot (Behan, 2017a)
$\frac{\ell!}{(d/2-1)_\ell}$	Poland *et al.* (2012), Poland and Stergiou (2015)
$\frac{\ell!}{4^\Delta (d-2)_\ell}$	Kos *et al.* (2014b) Mathematica notebook (Simmons-Duffin, 2015b)
$\frac{(-1)^\ell \ell!}{(d/2-1)_\ell}$	Simmons-Duffin (2017c)

Poland, Rychkov, Vichi, `18



- The prefactor rescales OPE coefficients. So an isolated bootstrap setup with some OPE coefficients assumed is a strong check.

Now we can vibe the genuinely new setup

- Replace the ϵ with ϵ' operator:

This is new, we also isolate $\epsilon'\epsilon' \sim \epsilon$

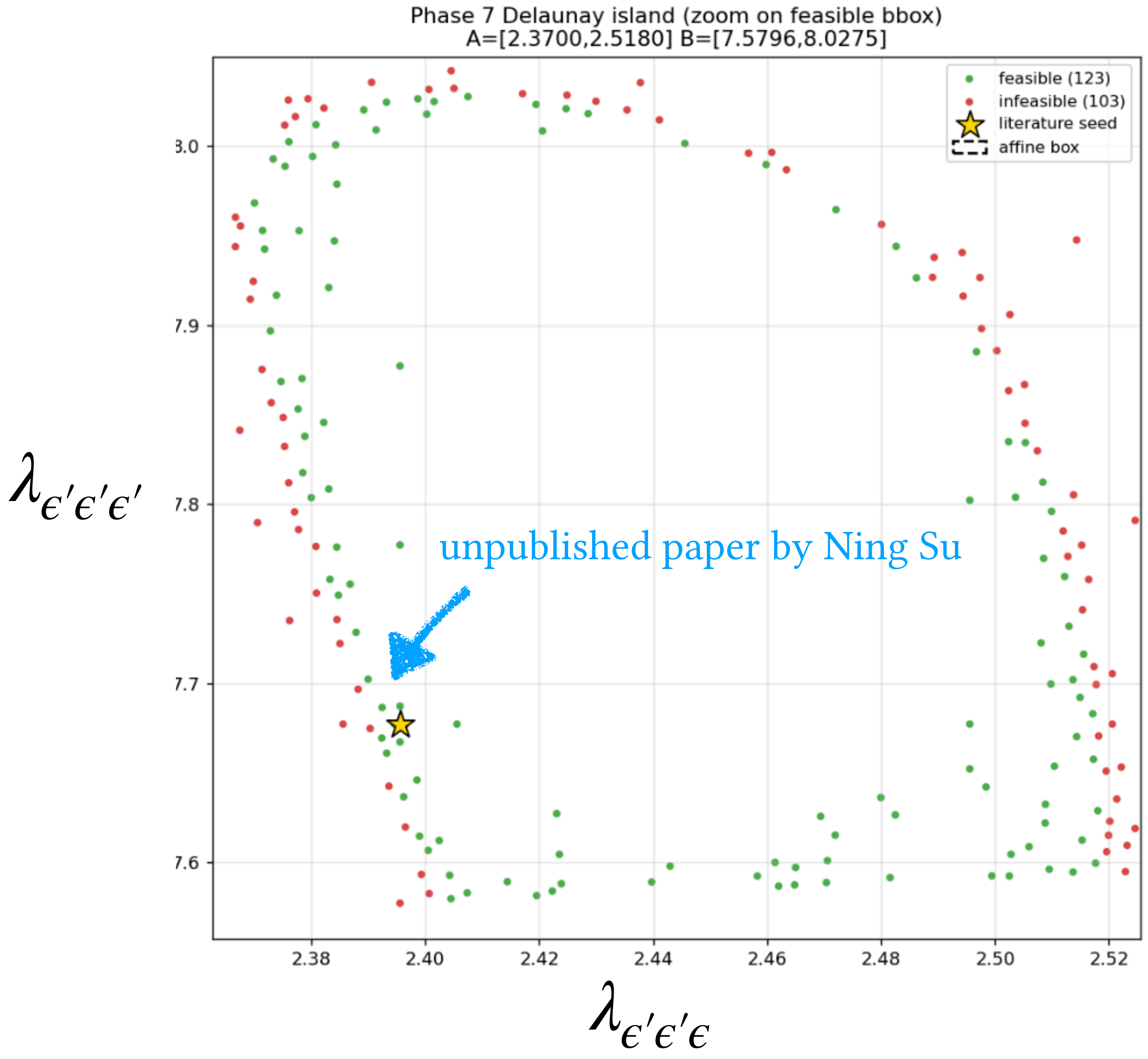
Now the external guy is ϵ'

$$
\begin{aligned}
(1\ 1)\,\vec{V}_{\text{vac}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} + (\lambda_{\sigma\sigma\epsilon'}\ \lambda_{\epsilon'\epsilon'\epsilon'})\,\vec{V}_{\text{ext}} \begin{pmatrix} \lambda_{\sigma\sigma\epsilon'} \\ \lambda_{\epsilon'\epsilon'\epsilon'} \end{pmatrix} + (\lambda_{\sigma\sigma\epsilon}\ \lambda_{\epsilon'\epsilon'\epsilon})\,\vec{V}_{\epsilon} \begin{pmatrix} \lambda_{\sigma\sigma\epsilon} \\ \lambda_{\epsilon'\epsilon'\epsilon} \end{pmatrix} \\
+ (\lambda_{\sigma\sigma T}\ \lambda_{\epsilon'\epsilon' T})\,\vec{V}_{T} \begin{pmatrix} \lambda_{\sigma\sigma T} \\ \lambda_{\epsilon'\epsilon' T} \end{pmatrix} + \sum_{\mathcal{O}+} (\lambda_{\sigma\sigma\mathcal{O}}\ \lambda_{\epsilon'\epsilon'\mathcal{O}})\,\vec{V}_{+,\Delta,\ell} \begin{pmatrix} \lambda_{\sigma\sigma\mathcal{O}} \\ \lambda_{\epsilon'\epsilon'\mathcal{O}} \end{pmatrix} + \sum_{\mathcal{O}-} \lambda^2_{\sigma\epsilon'\mathcal{O}} \vec{V}_{-,\Delta,\ell} = 0
\end{aligned}
$$

- All dimensions Δ_σ , Δ_ϵ , $\Delta_{\epsilon'}$ are known. Central charge c_T known. Two OPE coefficients $\lambda_{\sigma\sigma\epsilon}$, $\lambda_{\sigma\sigma\epsilon'}$ known. Two OPE coefficients $\lambda_{\epsilon'\epsilon'\epsilon}$, $\lambda_{\epsilon'\epsilon'\epsilon'}$ unknown.

Island - wait…

- Use this setup, the agent concluded with an "island":



Claude:
Found an island with $\lambda_{\epsilon'\epsilon'\epsilon'} \in [7.580,8.027]$

YX:
No you didn't. Search for a new lower bound for $\lambda_{\epsilon'\epsilon'\epsilon'}$

Claude:
Added 20 new grids in domain [7.50,7.59]

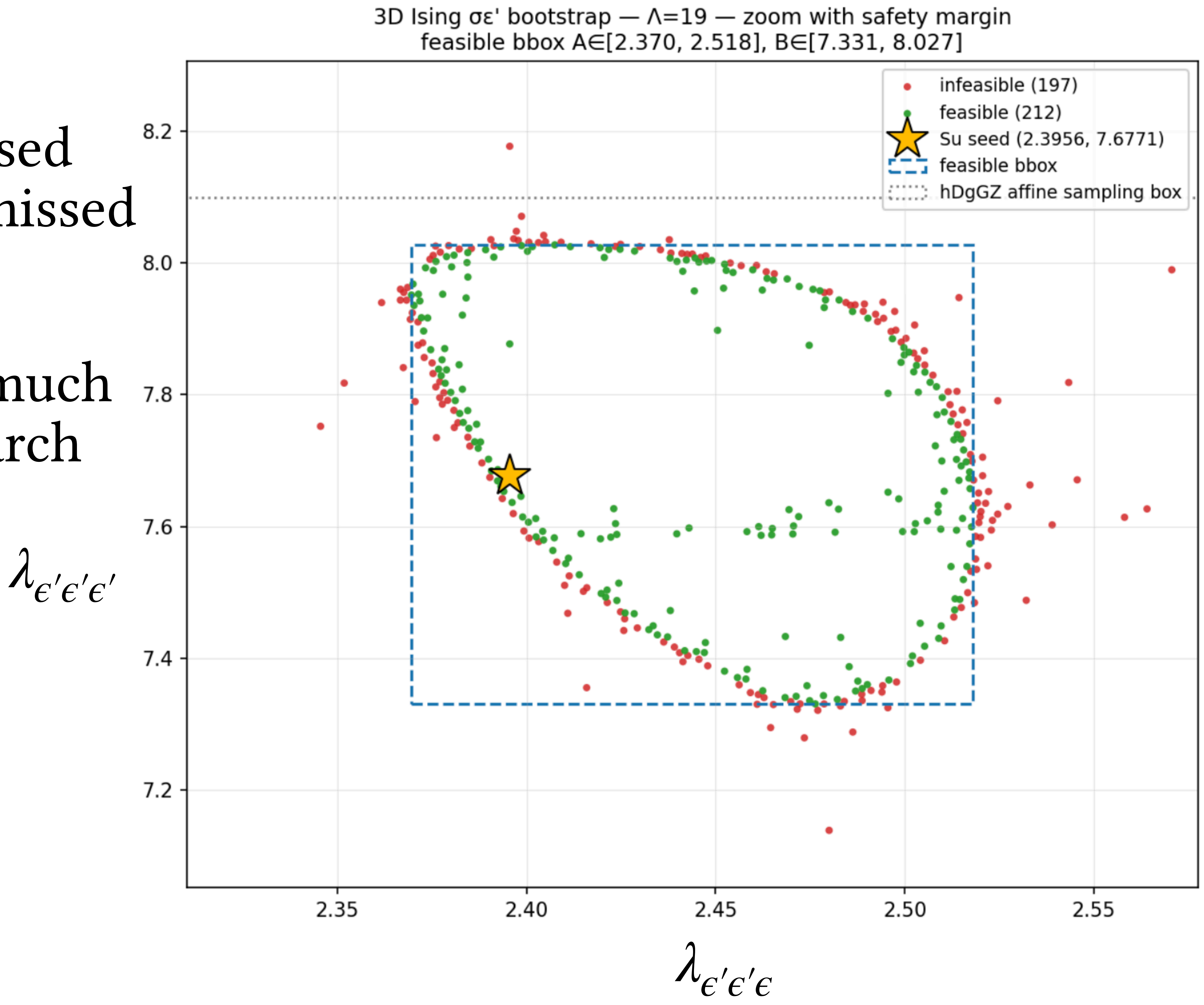
YX:
Can you just look at your plot?

The final island

- Claude then printed the plot pdf and used image process tools and found that it missed the lower bound.

- The fix was to add new lower bounds much further than the plot boundary and search again.

- Then the island looks good.



It’s always good to be suspicious

- We then challenged Claude how the OPE coefficients was put in to the bootstrap equation, and it turns out the setup was again unexpected.

- It’s worth slowing down a little to explain this difference.

$$
\begin{aligned}
&(1\ 1)\,\vec{V}_{\text{vac}}\begin{pmatrix}1\\1\end{pmatrix}+(\lambda_{\sigma\sigma\epsilon'}\ \lambda_{\epsilon'\epsilon'\epsilon'})\,\vec{V}_{\text{ext}}\begin{pmatrix}\lambda_{\sigma\sigma\epsilon'}\\ \lambda_{\epsilon'\epsilon'\epsilon'}\end{pmatrix}+(\lambda_{\sigma\sigma\epsilon}\ \lambda_{\epsilon'\epsilon'\epsilon})\,\vec{V}_{\epsilon}\begin{pmatrix}\lambda_{\sigma\sigma\epsilon}\\ \lambda_{\epsilon'\epsilon'\epsilon}\end{pmatrix}\\
&\qquad+(\lambda_{\sigma\sigma T}\ \lambda_{\epsilon'\epsilon' T})\,\vec{V}_{T}\begin{pmatrix}\lambda_{\sigma\sigma T}\\ \lambda_{\epsilon'\epsilon' T}\end{pmatrix}+\sum_{\mathcal{O}^+}(\lambda_{\sigma\sigma\mathcal{O}}\ \lambda_{\epsilon'\epsilon'\mathcal{O}})\,\vec{V}_{+,\Delta,\ell}\begin{pmatrix}\lambda_{\sigma\sigma\mathcal{O}}\\ \lambda_{\epsilon'\epsilon'\mathcal{O}}\end{pmatrix}+\sum_{\mathcal{O}^-}\lambda^2_{\sigma\epsilon'\mathcal{O}}\vec{V}_{-,\Delta,\ell}=0
\end{aligned}
$$

- For each point $(\lambda_{\epsilon'\epsilon'\epsilon}\,, \lambda_{\epsilon'\epsilon'\epsilon'})$ in the plot, all isolated OPE coefficients are numerical numbers, so the V_{vac}, V_{T}, V_{ext}, V_{ϵ} can all be merged to the same numerical vector.

It's always good to be suspicious

$$(1\ 1)\,\vec{V}_{\text{vac}}\begin{pmatrix}1\\1\end{pmatrix}+(\lambda_{\sigma\sigma\epsilon'}\ \lambda_{\epsilon'\epsilon'\epsilon'})\,\vec{V}_{\text{ext}}\begin{pmatrix}\lambda_{\sigma\sigma\epsilon'}\\ \lambda_{\epsilon'\epsilon'\epsilon'}\end{pmatrix}+(\lambda_{\sigma\sigma\epsilon}\ \lambda_{\epsilon'\epsilon'\epsilon})\,\vec{V}_{\epsilon}\begin{pmatrix}\lambda_{\sigma\sigma\epsilon}\\ \lambda_{\epsilon'\epsilon'\epsilon}\end{pmatrix}$$
$$+(\lambda_{\sigma\sigma T}\ \lambda_{\epsilon'\epsilon'T})\,\vec{V}_{T}\begin{pmatrix}\lambda_{\sigma\sigma T}\\ \lambda_{\epsilon'\epsilon'T}\end{pmatrix}+\sum_{\mathcal{O}^+}(\lambda_{\sigma\sigma\mathcal{O}}\ \lambda_{\epsilon'\epsilon'\mathcal{O}})\,\vec{V}_{+,\Delta,\ell}\begin{pmatrix}\lambda_{\sigma\sigma\mathcal{O}}\\ \lambda_{\epsilon'\epsilon'\mathcal{O}}\end{pmatrix}+\sum_{\mathcal{O}^-}\lambda^2_{\sigma\epsilon'\mathcal{O}}\vec{V}_{-,\Delta,\ell}=0$$

- The semidefinite programming problem is thus

$$\vec{V}_{\text{merge}}=(1\ 1)\,\vec{V}_{\text{vac}}\begin{pmatrix}1\\1\end{pmatrix}+(\lambda_{\sigma\sigma\epsilon'}\ \lambda_{\epsilon'\epsilon'\epsilon'})\,\vec{V}_{\text{ext}}\begin{pmatrix}\lambda_{\sigma\sigma\epsilon'}\\ \lambda_{\epsilon'\epsilon'\epsilon'}\end{pmatrix}+(\lambda_{\sigma\sigma\epsilon}\ \lambda_{\epsilon'\epsilon'\epsilon})\,\vec{V}_{\epsilon}\begin{pmatrix}\lambda_{\sigma\sigma\epsilon}\\ \lambda_{\epsilon'\epsilon'\epsilon}\end{pmatrix}$$
$$+(\lambda_{\sigma\sigma T}\ \lambda_{\epsilon'\epsilon'T})\,\vec{V}_{T}\begin{pmatrix}\lambda_{\sigma\sigma T}\\ \lambda_{\epsilon'\epsilon'T}\end{pmatrix}$$

find $\vec{\alpha}$ such that $\quad \alpha\cdot V_{\text{merge}}=1 \quad$ then the spectrum is ruled out

$$\left.\begin{array}{l}\alpha\cdot V_{+}\succeq 0\\ \alpha\cdot V_{-}\geq 0\end{array}\right.\quad \forall\Delta,\ \ell$$

It's always good to be suspicious

- The semidefinite programming problem is thus

$$\vec{V}_{\text{merge}} = (1\ 1)\, \vec{V}_{\text{vac}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} + (\lambda_{\sigma\sigma\epsilon'}\ \lambda_{\epsilon'\epsilon'\epsilon'})\, \vec{V}_{\text{ext}} \begin{pmatrix} \lambda_{\sigma\sigma\epsilon'} \\ \lambda_{\epsilon'\epsilon'\epsilon'} \end{pmatrix} + (\lambda_{\sigma\sigma\epsilon}\ \lambda_{\epsilon'\epsilon'\epsilon})\, \vec{V}_{\epsilon} \begin{pmatrix} \lambda_{\sigma\sigma\epsilon} \\ \lambda_{\epsilon'\epsilon'\epsilon} \end{pmatrix}$$
$$+ (\lambda_{\sigma\sigma T}\ \lambda_{\epsilon'\epsilon' T})\, \vec{V}_{T} \begin{pmatrix} \lambda_{\sigma\sigma T} \\ \lambda_{\epsilon'\epsilon' T} \end{pmatrix}$$

find $\vec{\alpha}$ such that $\begin{array}{l} \alpha \cdot V_{\text{merge}} = 1 \\ \alpha \cdot V_{+} \succeq 0 \\ \alpha \cdot V_{-} \geq 0 \end{array}\ \forall \Delta,\ \ell$ then the spectrum is ruled out

- Instead, Claude created new **positivity channels** even though all λ's are fixed

$$\alpha \cdot (\lambda_{\sigma\sigma\epsilon'}\ \lambda_{\epsilon'\epsilon'\epsilon'})\, \vec{V}_{\text{ext}} \begin{pmatrix} \lambda_{\sigma\sigma\epsilon'} \\ \lambda_{\epsilon'\epsilon'\epsilon'} \end{pmatrix} \geq 0 \qquad \alpha \cdot (\lambda_{\sigma\sigma\epsilon}\ \lambda_{\epsilon'\epsilon'\epsilon})\, \vec{V}_{\epsilon} \begin{pmatrix} \lambda_{\sigma\sigma\epsilon} \\ \lambda_{\epsilon'\epsilon'\epsilon} \end{pmatrix} \geq 0 \qquad \alpha \cdot (\lambda_{\sigma\sigma T}\ \lambda_{\epsilon'\epsilon' T})\, \vec{V}_{T} \begin{pmatrix} \lambda_{\sigma\sigma T} \\ \lambda_{\epsilon'\epsilon' T} \end{pmatrix} \geq 0$$

- This setup would allow all OPE coefficients to rescale, thus the bound will be **weaker**.

Recap

- We initiated a study of higher irrelevant operator bootstrap in 3D Ising CFT with ai-assistance.
- The setup was non-trivial to begin with. Without human prior knowledge we do not expect the agent can come up with this operator isolation scheme.
- Under the instruction, the agent assistance drastically reduced the setup time for the numerics, and made it convenient to modify the setup using natural language.
- However, human expert needs to constantly verify and challenge the result from the agent to keep it on track.

Would agents internalize our intuition?

- We believe it will.
- In the short term, the experience is easy to save and distribute. Skill can be saved on demand, and sessions can be forked.
- In the long term, we have seen that useful experience or instructions that used to require skill files are internalized into the base model in later versions, and the new model becomes smarter.
- This suggests a way that the physics community can contribute to the development of AI.

Some future goals

- A systematic analysis on the convergence of mixed bootstrap involving lightest and heavier operators. WIP with Yuhang Wu and Yunxiao Ye.
- A functional numerical bootstrap analysis. Numerical bootstrap can be made much more efficient using the analytic functional basis, but the setup still need work and AI-assistance would speed it up significantly.
- Numerical bootstrap involving spinning operators - complexity used to be formidable to learn from scratch, but made feasible with the help of AI.

Thank you