Progress on generalized approximate contraction

Johnnie Gray

Tensor Subgroup

2021-01-21

High Level Overview:

- Want abstract approach where geometry is just another input:
  - Take **TN geometry**
  - Find **contraction tree**
  - Find **ordering** for tree
  - Choose **gauging** method
  - Choose **compression** method

mix'n'match

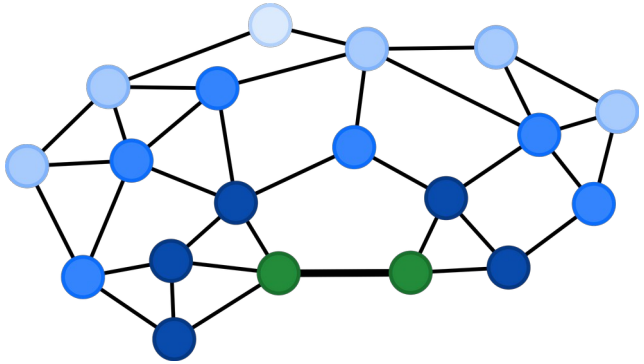
Automatic approximate contraction

Why?

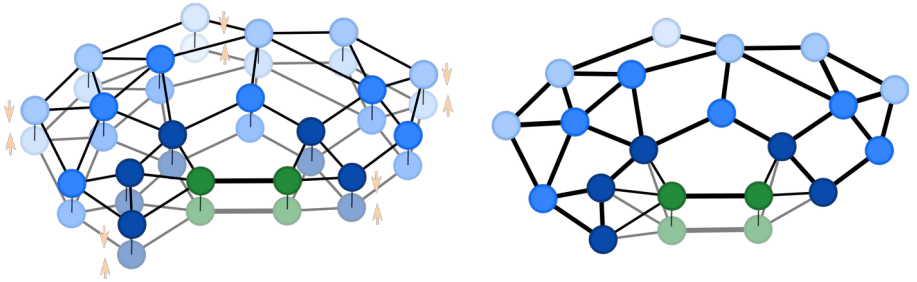
- Unified approach to 2D, 3D, …
- Contract things with unusual geometries
  - Quantum circuits
  - Weighted model counting instances
  - Weird lattices
  - Partially contracted regular TNs
- Transplant algorithms easily
- Improve existing approaches?

TN Geometry

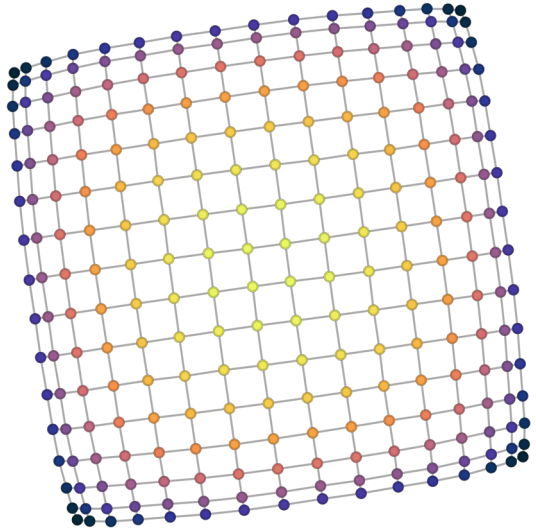
Scalar TN:

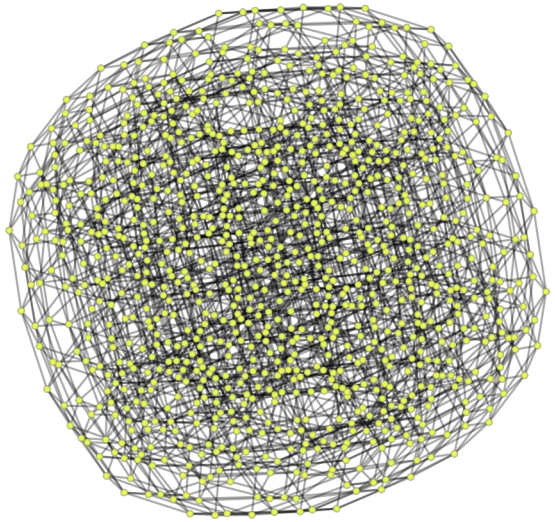


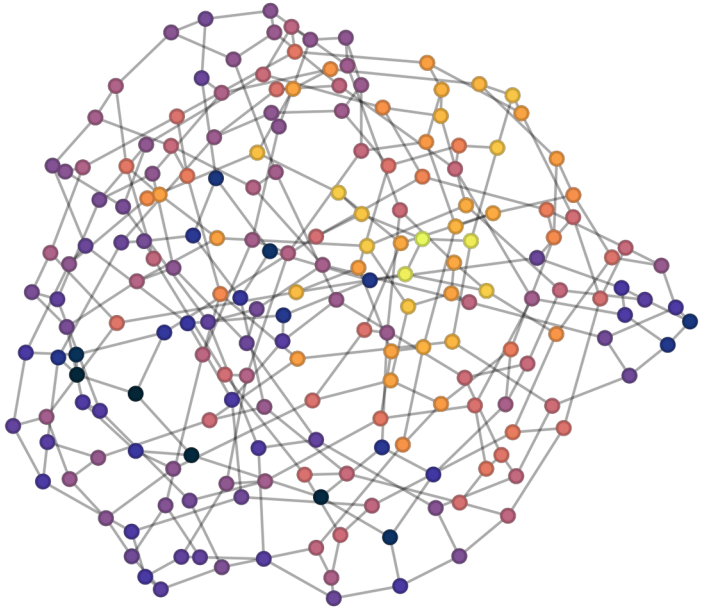
‘vector’ TN:



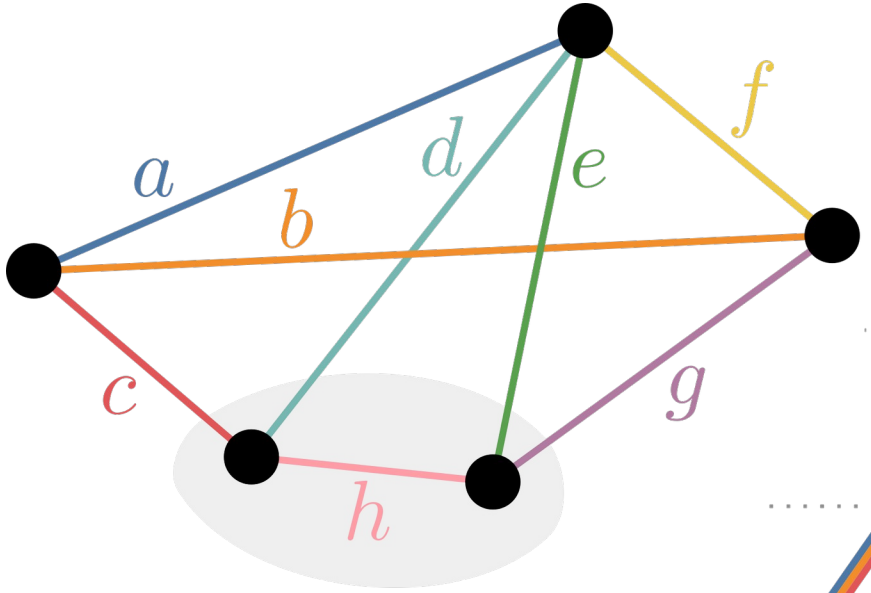
Centrality – useful heuristic

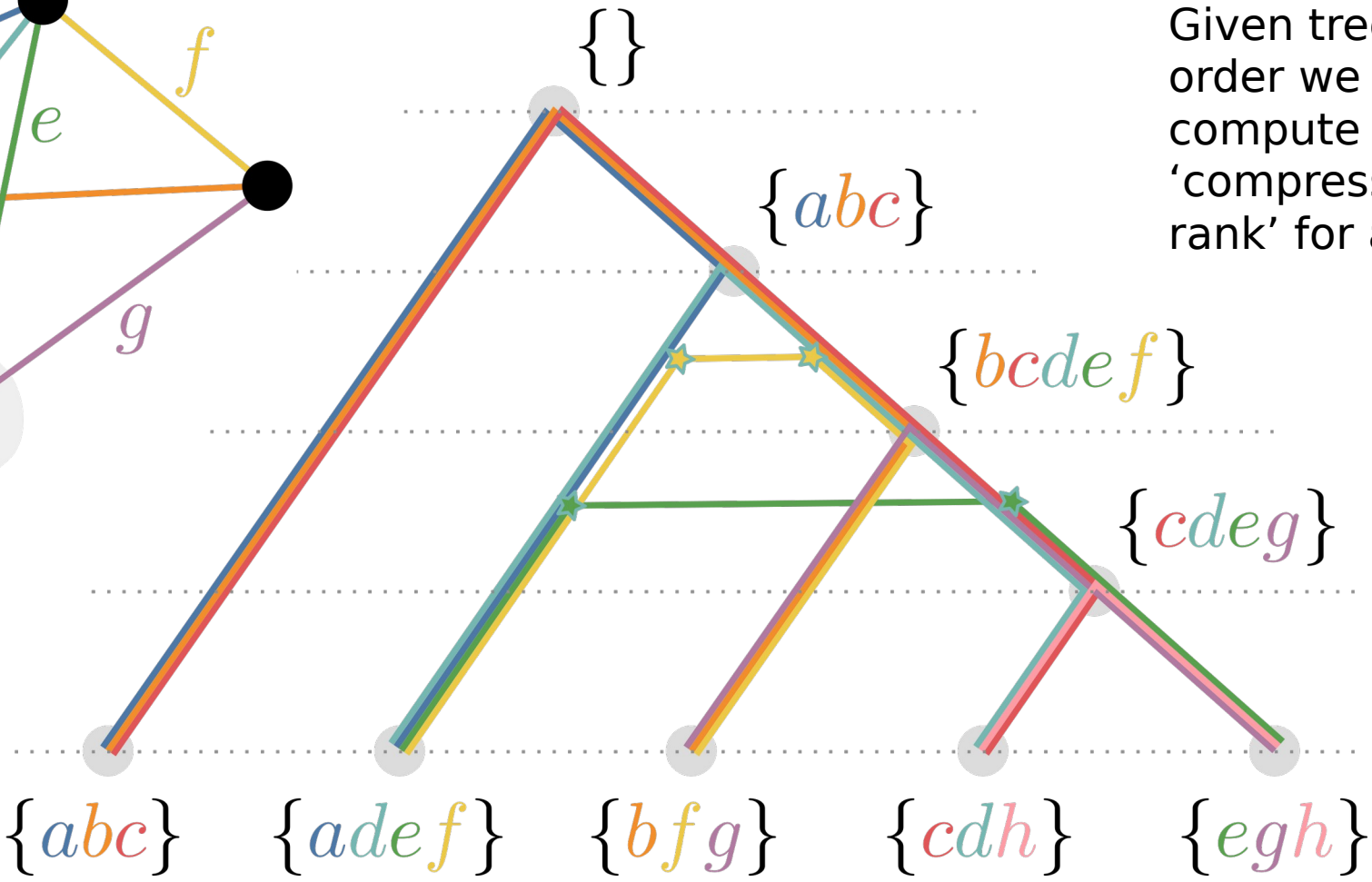






Tree Ordering & ‘Compressed rank’





Given tree **and** order we can compute ‘compressed rank’ for a tree

Once you can compress indices away – operations and costs of nodes in tree are no longer commute – order matters

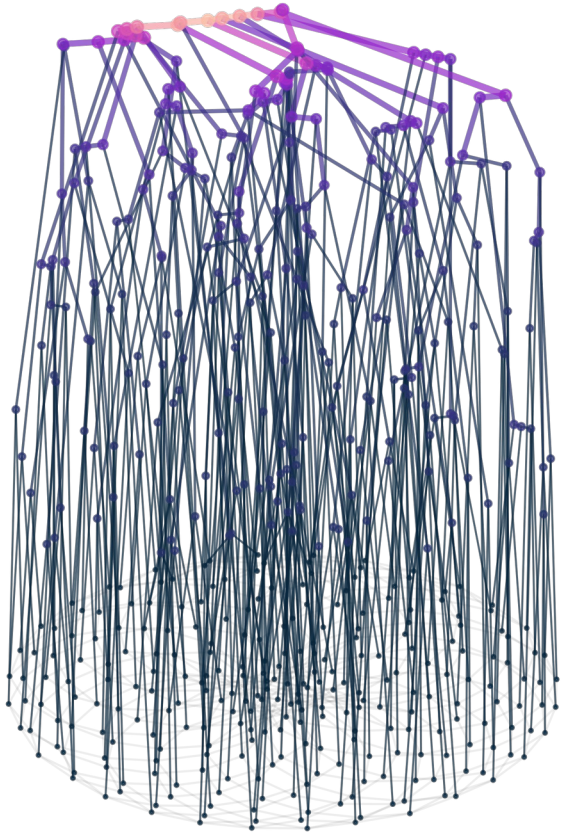
(precludes dynamic programming

TN Contraction Tree – Compressed Greedy

- Generic approach, assess all possible contractions, and greedily select based on

    1. Compressed output size
    2. Centrality
    3. Subgraph size
    4. ...

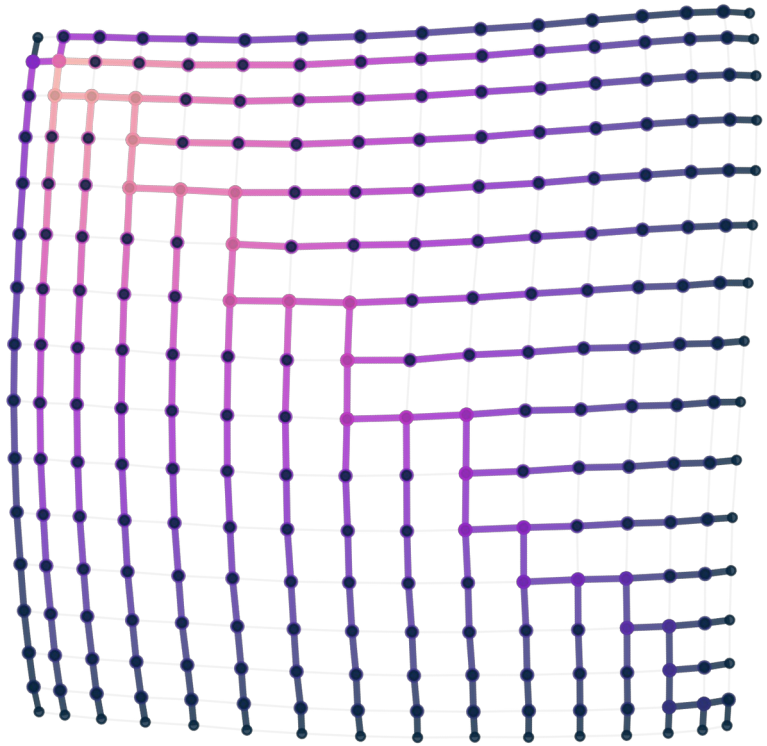
- Take hyper-optimization approach – tune heuristic function that assesses contraction with respect to total compressed rank

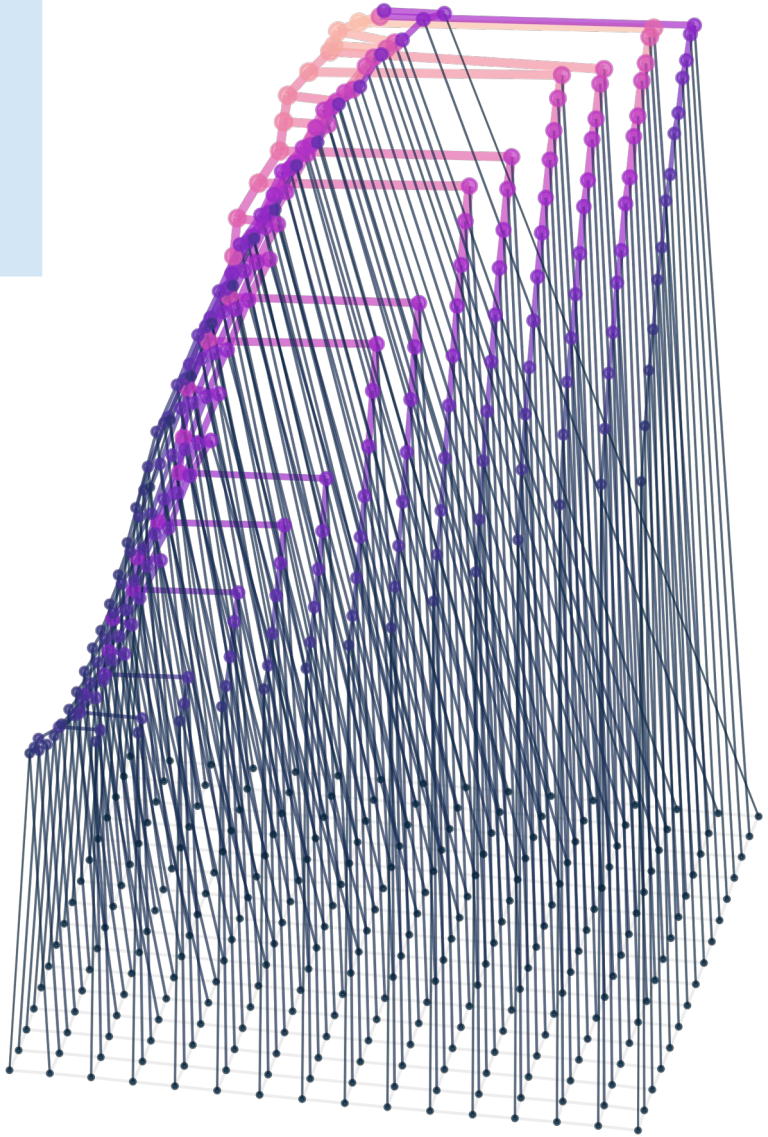


TN Contraction Tree – ‘Boundary Style’

Generate tree from span of original graph:

- Feed a few heuristics like centrality into span generator, then check quality and repeat
- Restricted set of contraction trees, works well for regular lattices

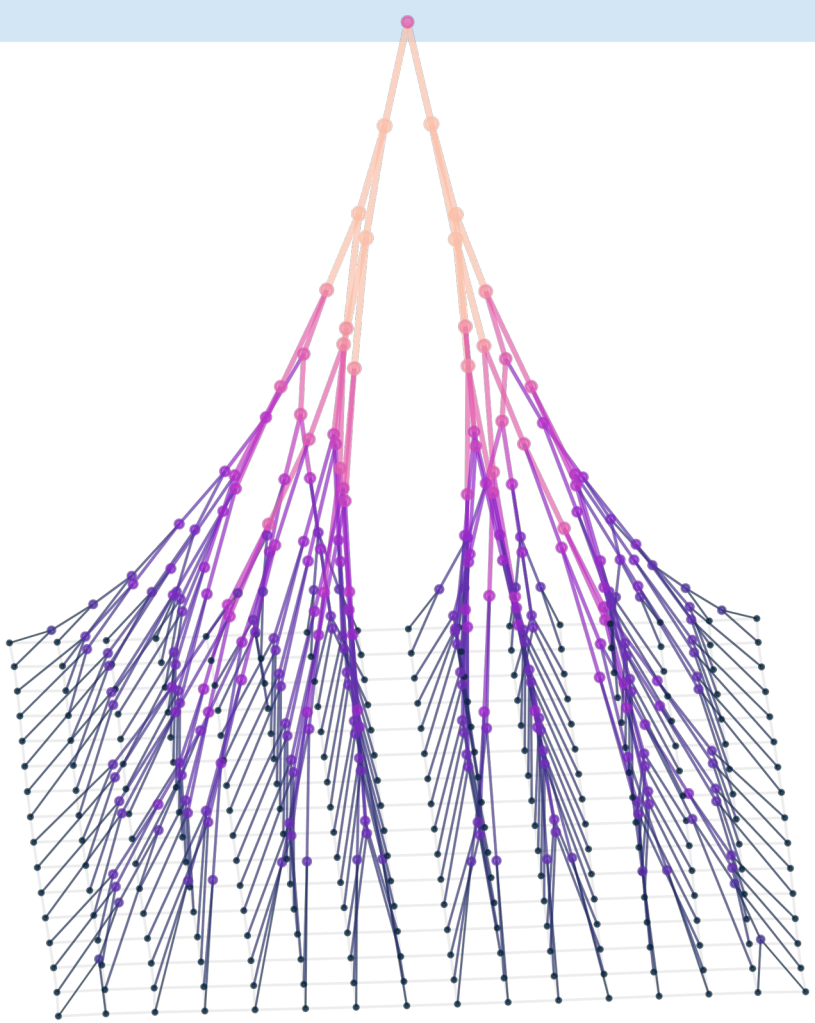


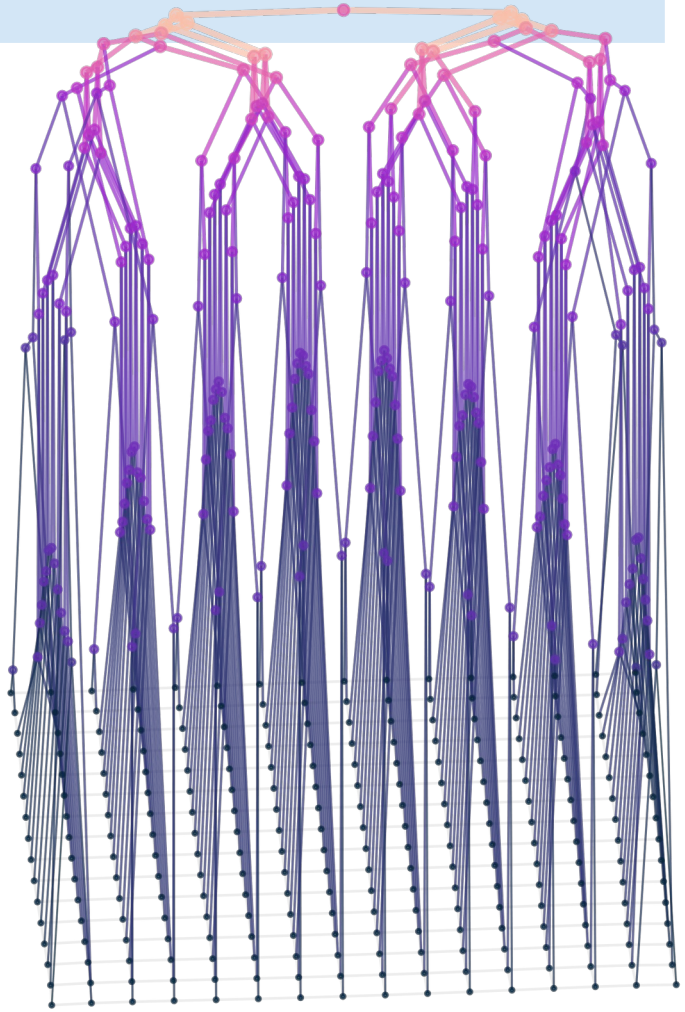


TN Contraction Tree – ‘RG-style’

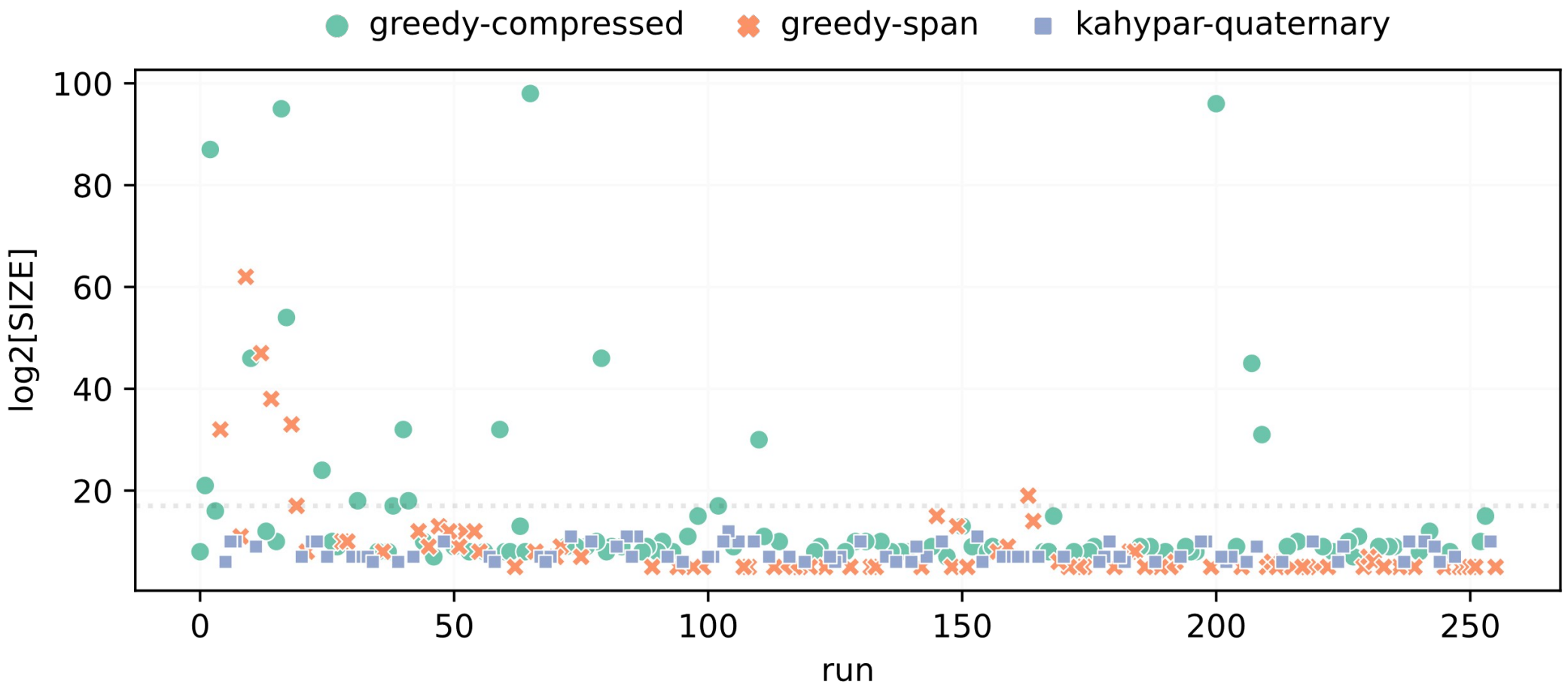
- Generate tree by grouping together small communities of tensors, compressing between these essentially coarse grains the lattice

- Seems good for cases where no obvious boundary or regularity

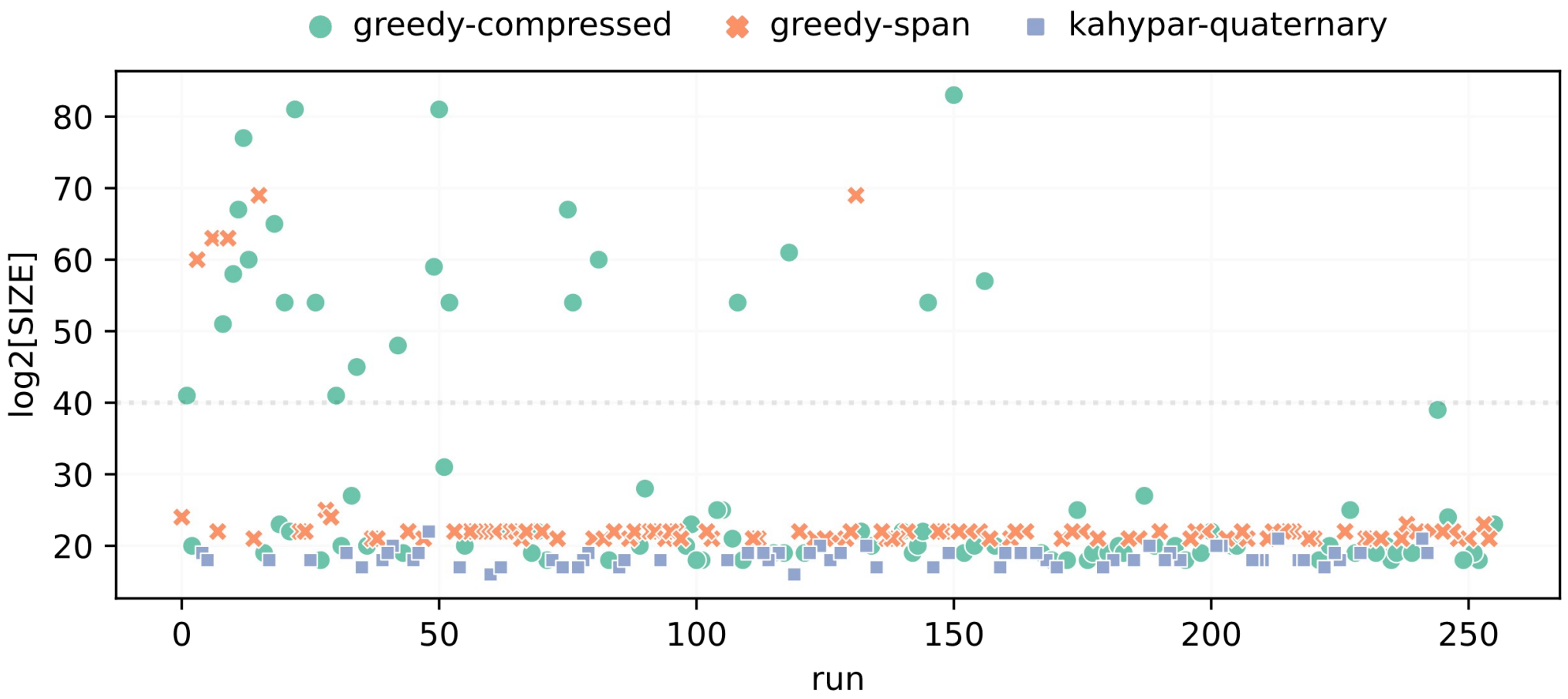




Compressed rank search results - lattice



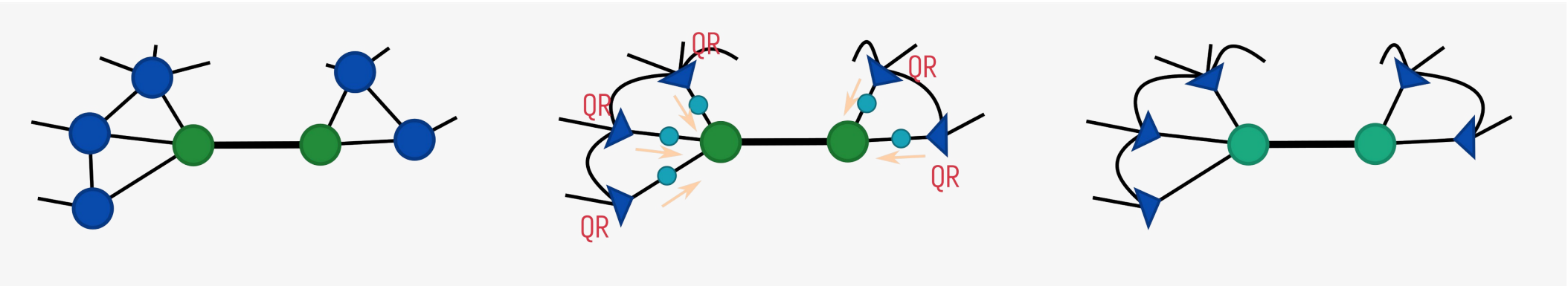
Compressed rank search results - random



Gauging methods – local ‘canonization’

- Basic and cheap strategy that would exactly orthogonalize any tree tensor network is the distance is set to .

local canonization "perfect for trees"

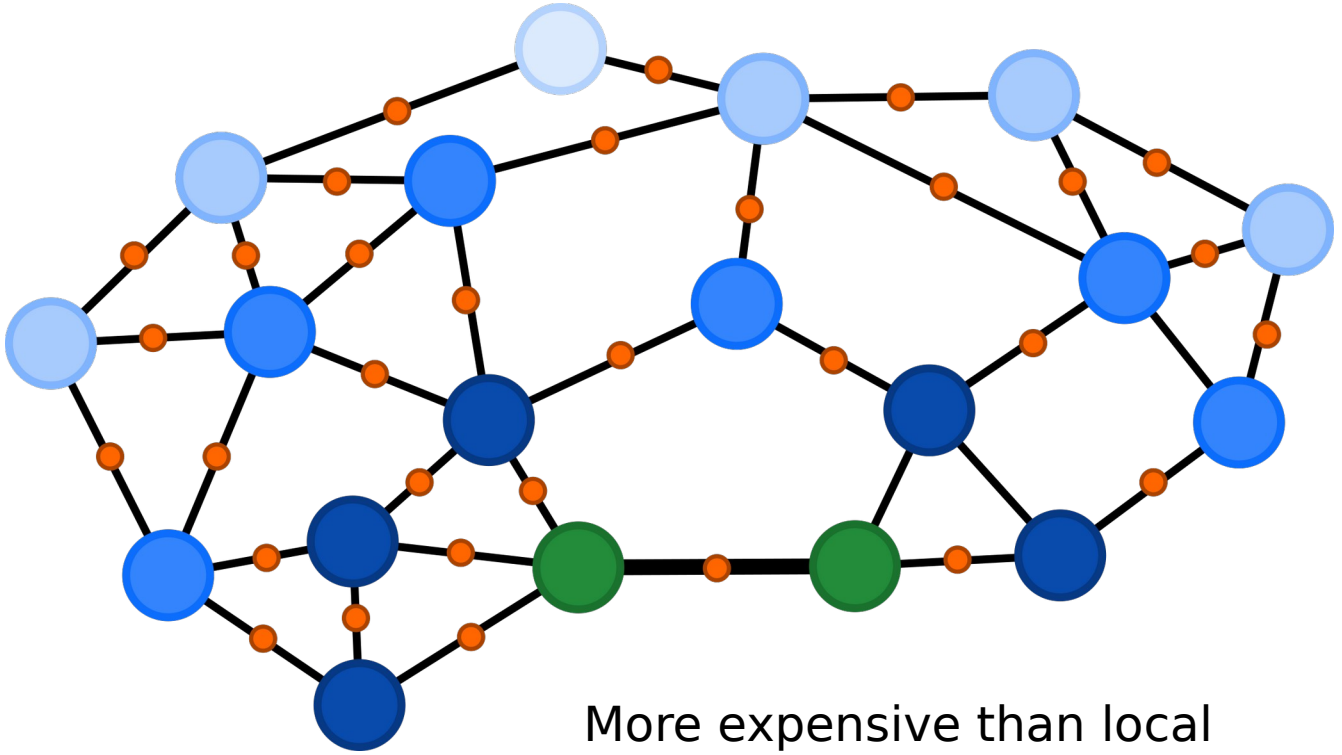


Gauging methods – ‘simple’ gauging

1. Iteratively perform reduced SVD on all bonds and accumulate singular values

2. Absorb (maybe not?)

3. Can re-do for just local portions on TN e.g. after contraction or



More expensive than local canonization but overhead still constant

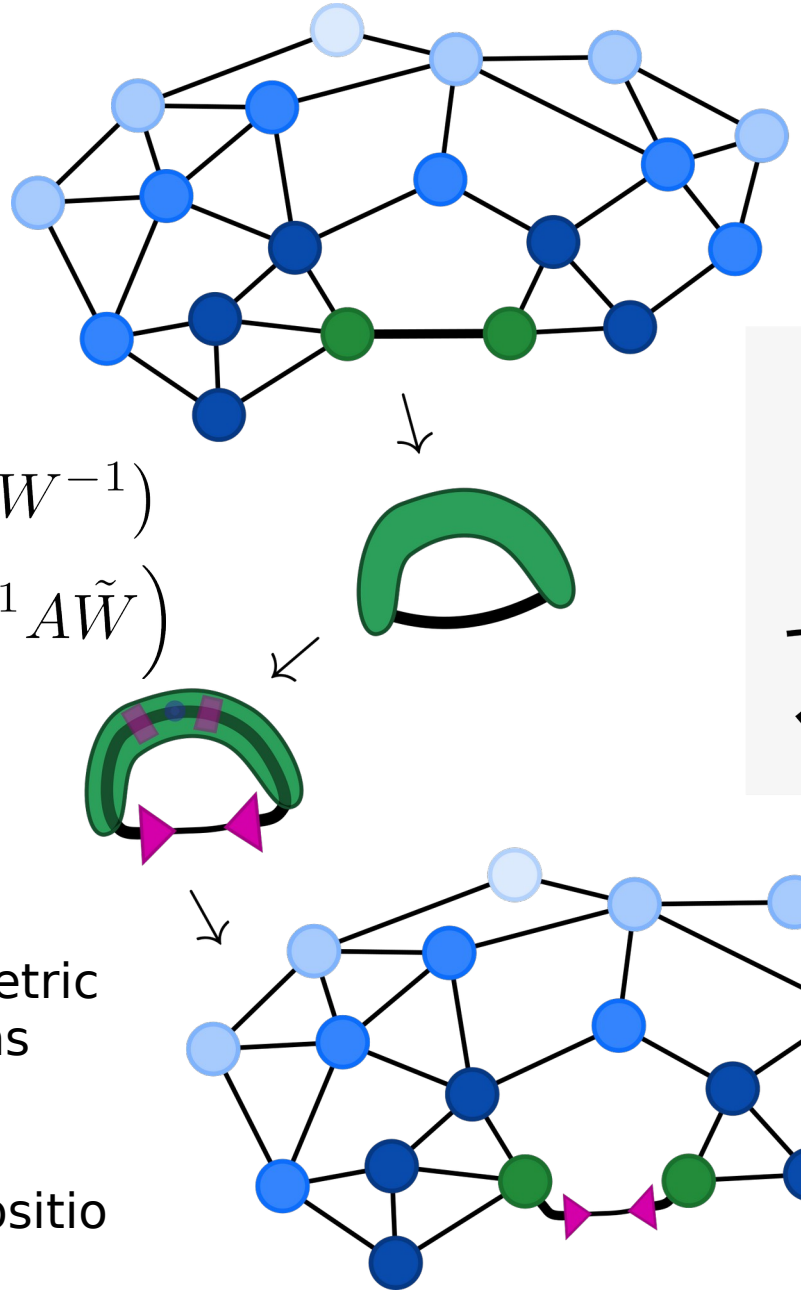
Gauging methods – ‘full-bond’ gauging

What if we can (approximately) contract the full environment of a bond?

Want to insert a pair of isometries that *would* act as gauges if full rank, that preserve the trace value

Trace value is not guaranteed to be positive

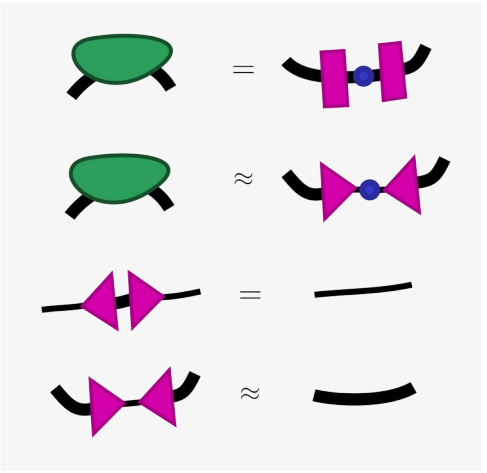
$$z = \mathrm{Tr}\,(A)$$
$$= \mathrm{Tr}\left(AWW^{-1}\right)$$
$$\approx \mathrm{Tr}\left(\tilde{W}^{-1}A\tilde{W}\right)$$



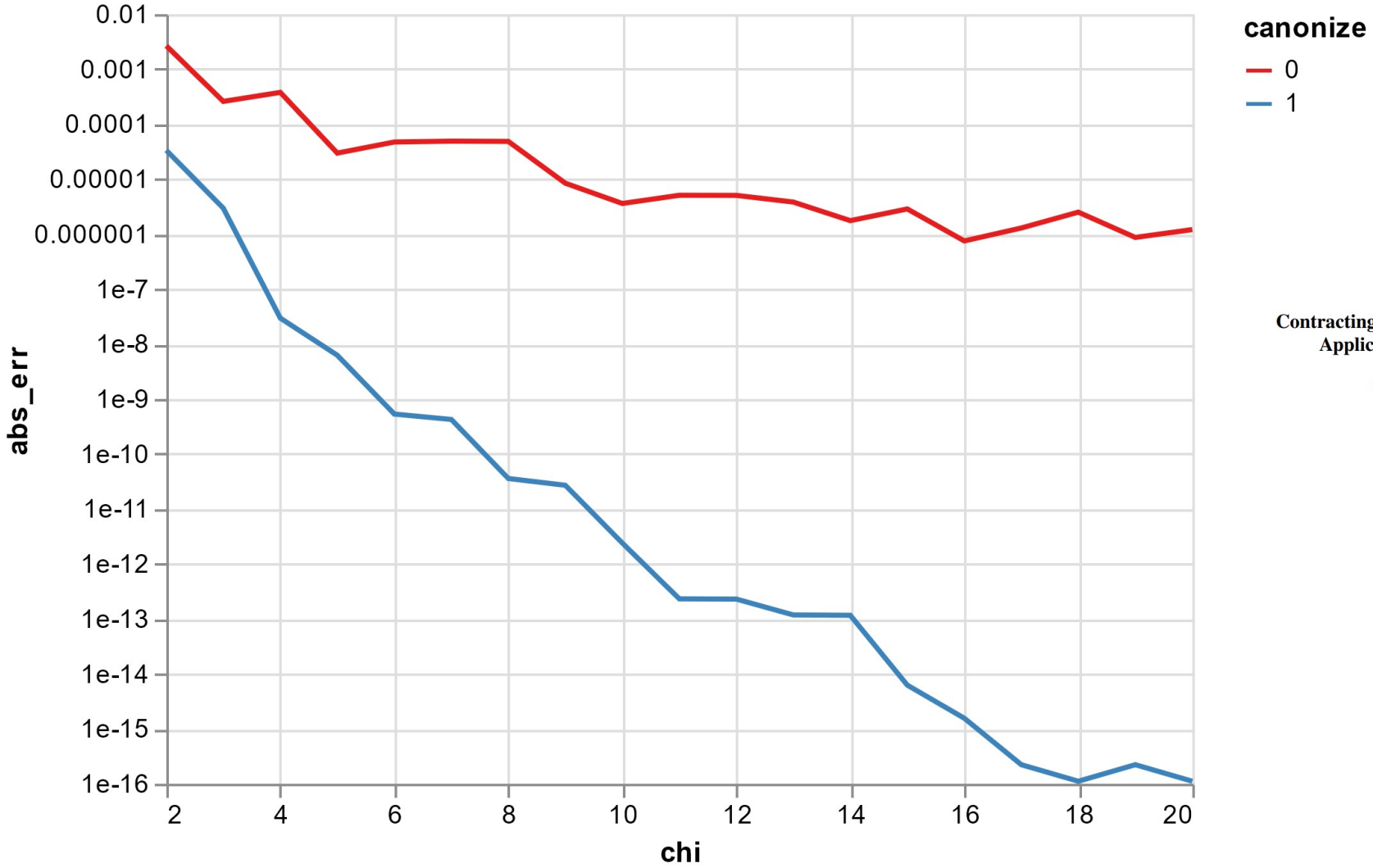
Various symmetric decompositions possible, non-Hermitian eigendecompositio n basic choice

Need inverse not dagger

$$A = W\lambda W^{-1}$$



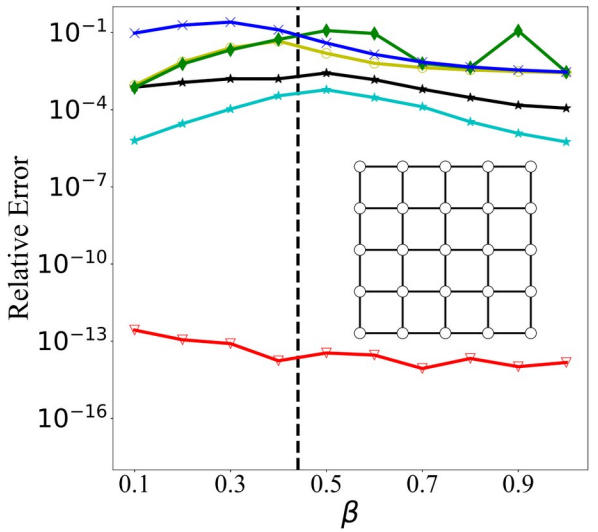
Results – 16x16 Ising lattice β=0.44 - [easy]



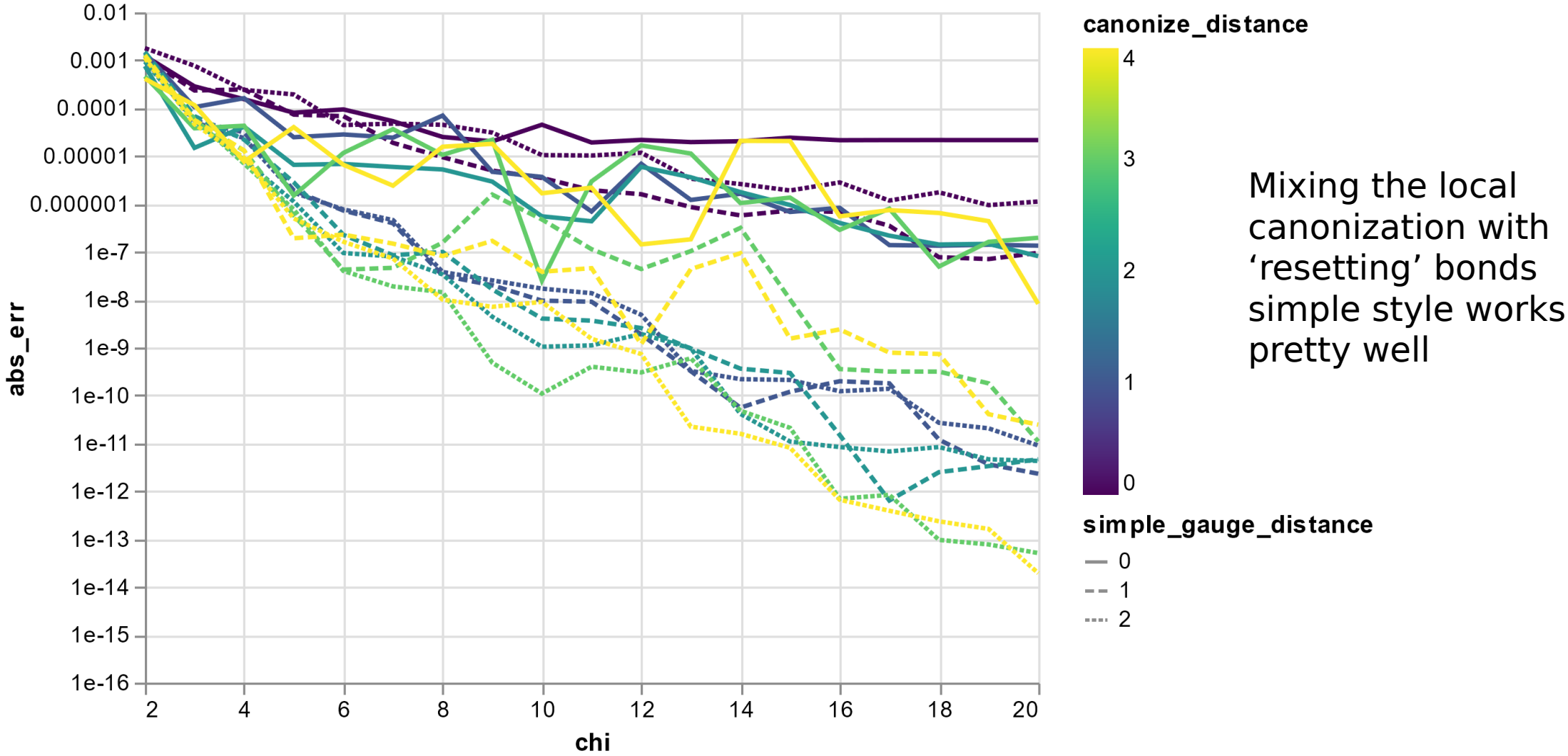
Normal MPS boundary contraction already overpowered

Contracting Arbitrary Tensor Networks: General Approximate Algorithm and Applications in Graphical Models and Quantum Circuit Simulations

Feng Pan,[1,2,*] Pengfei Zhou,[1,2,*] Sujie Li,[1,2,*] and Pan Zhang[1,3,4,†]

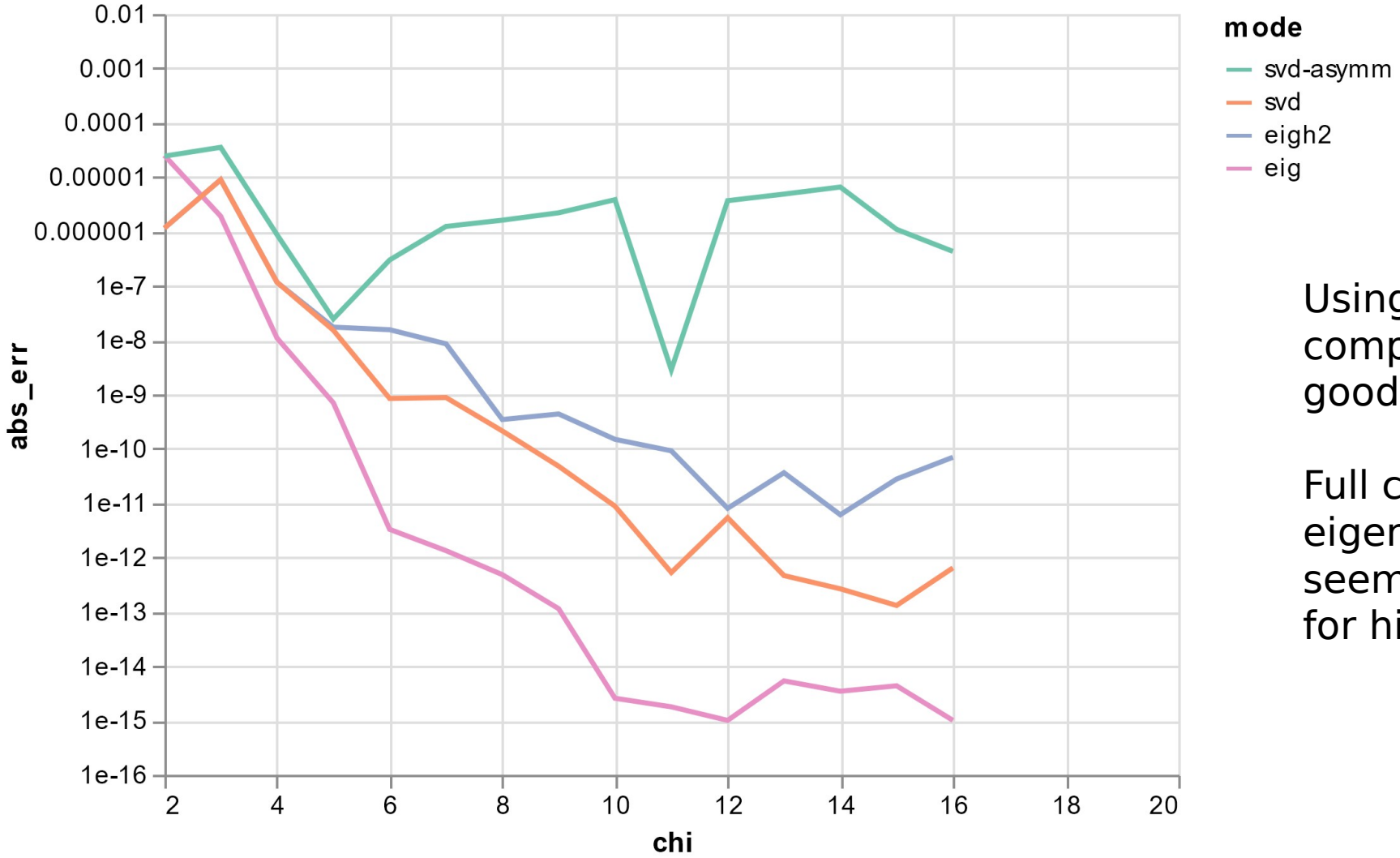


Results – 16x16 Ising lattice β=0.44 - [easy]



Mixing the local canonization with ‘resetting’ bonds simple style works pretty well

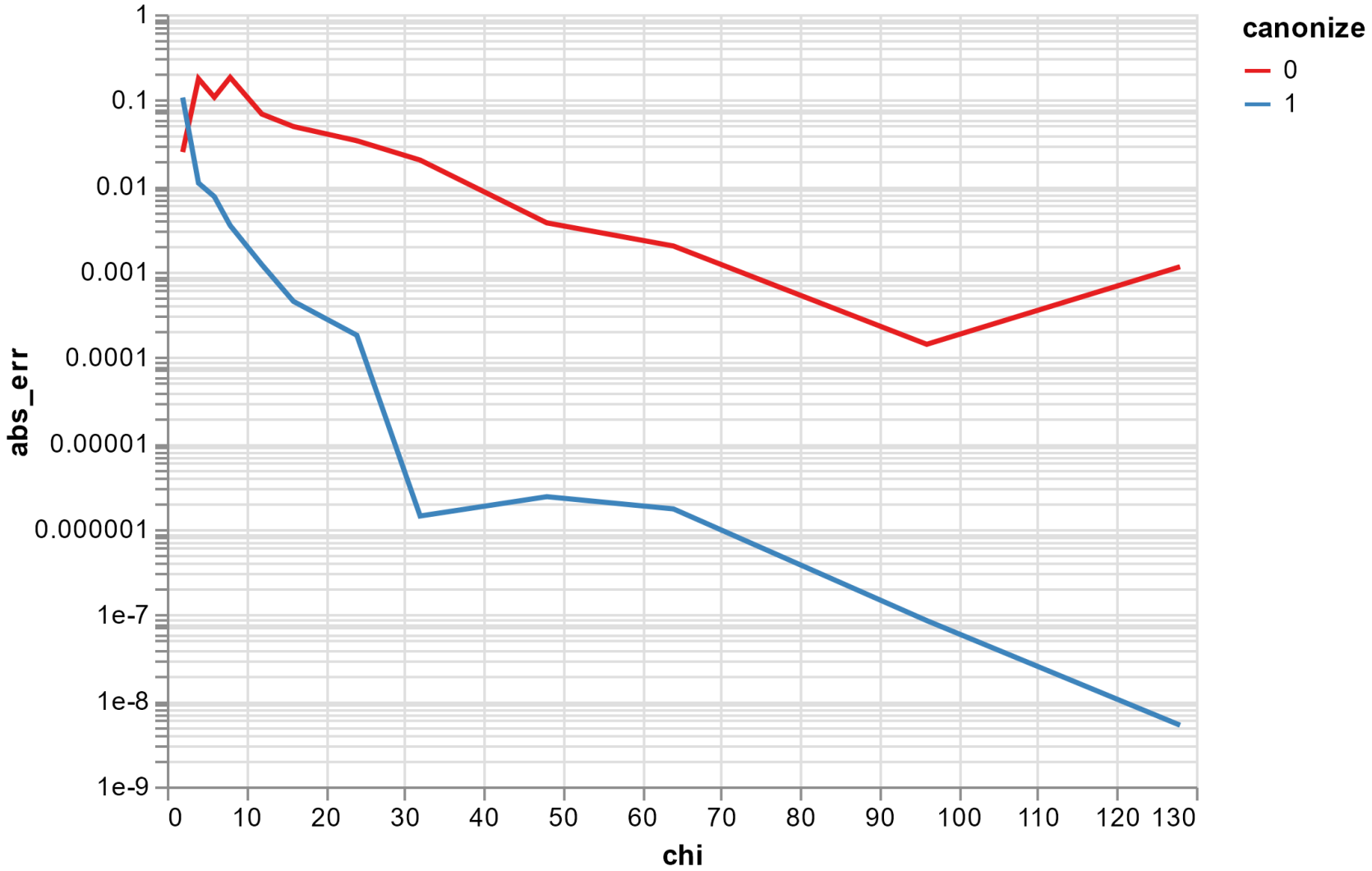
Results – 16x16 Ising lattice β=0.44 - [easy]



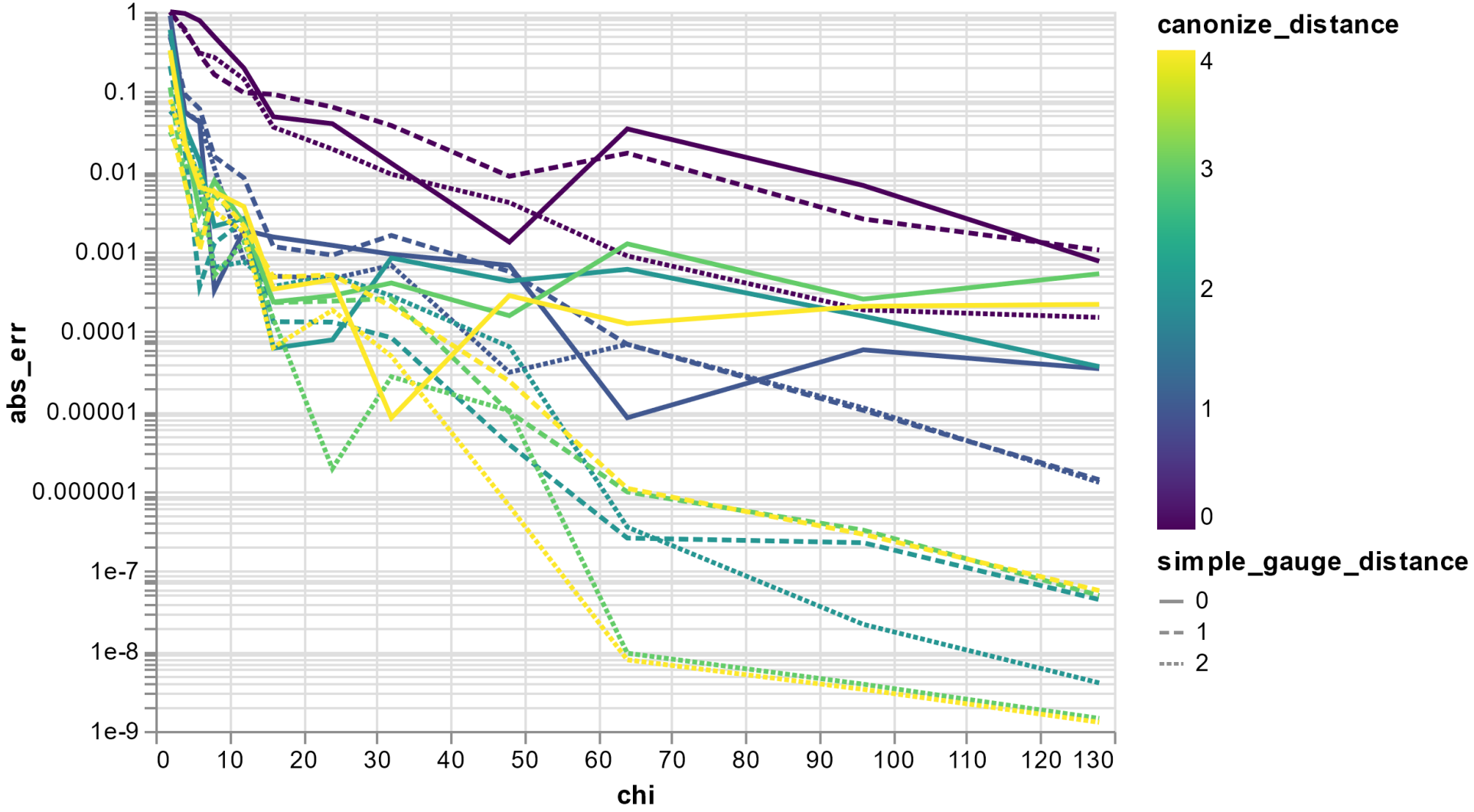
Using full-bond compression very very good

Full complex eigendecomposition seems to be required for highest accuracy

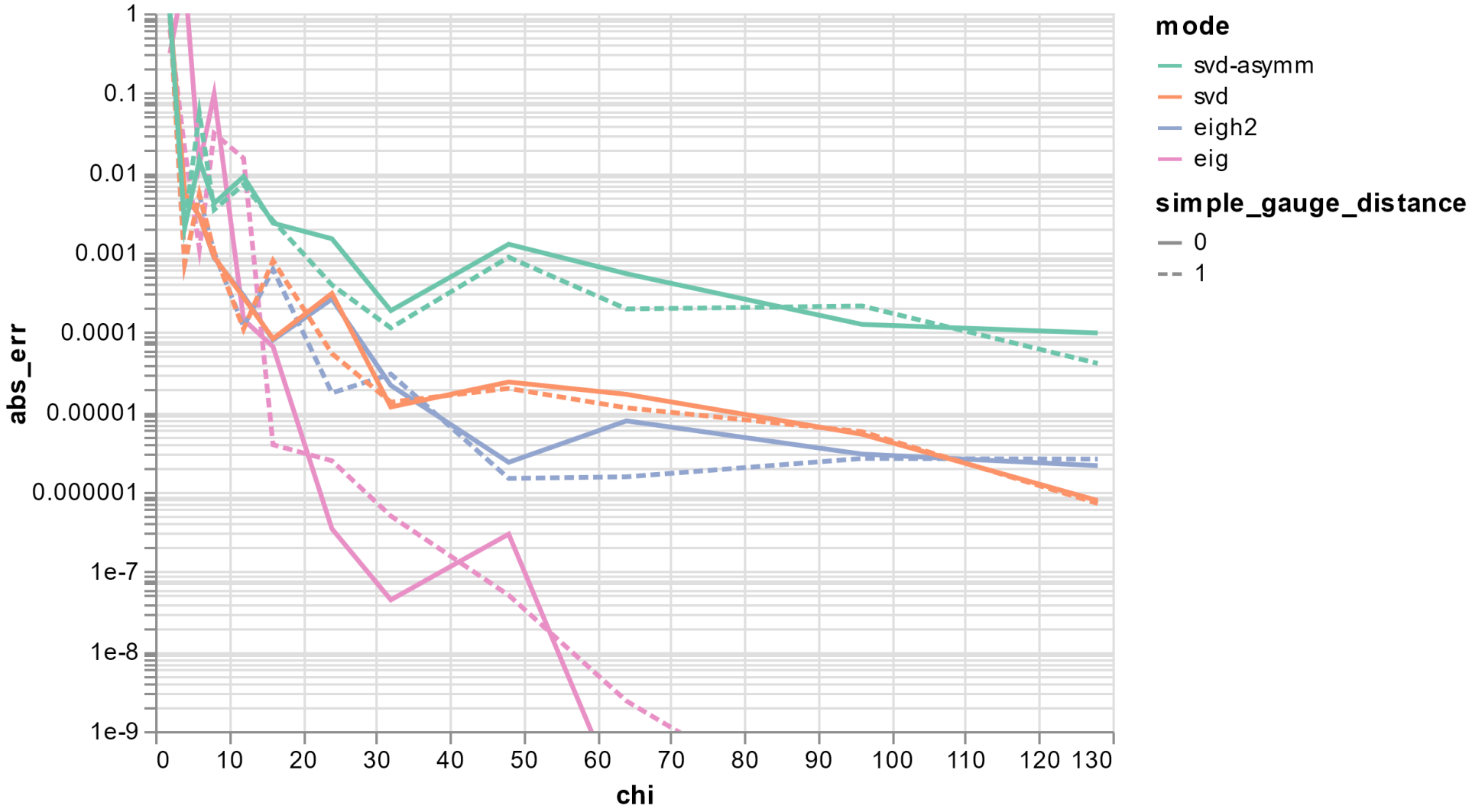
Results – 10x10 D=2 PEPS sandwich- [medium]



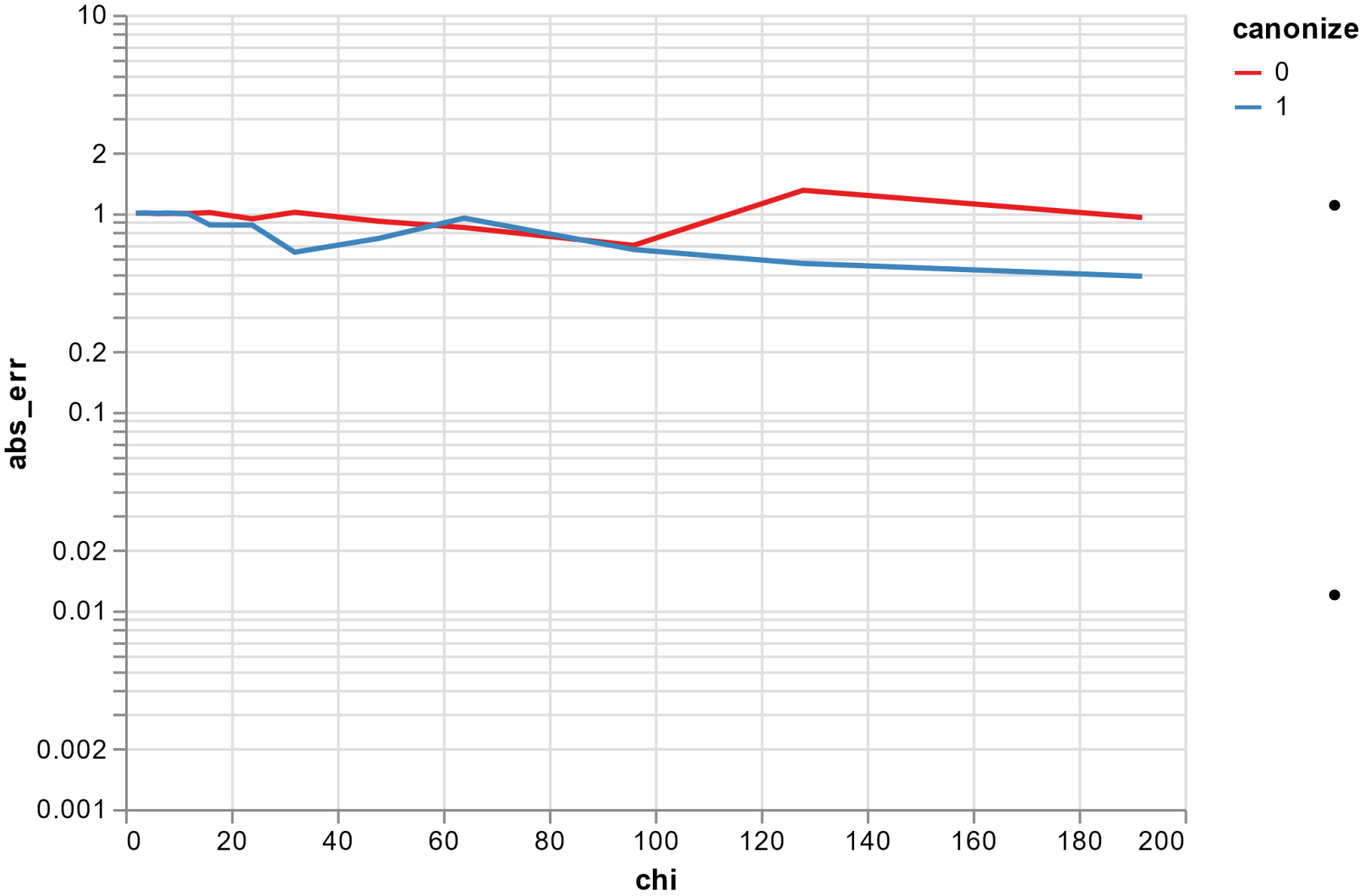
Results – 10x10 D=2 PEPS sandwich- [medium]



Results – 10x10 D=2 PEPS sandwich- [medium]

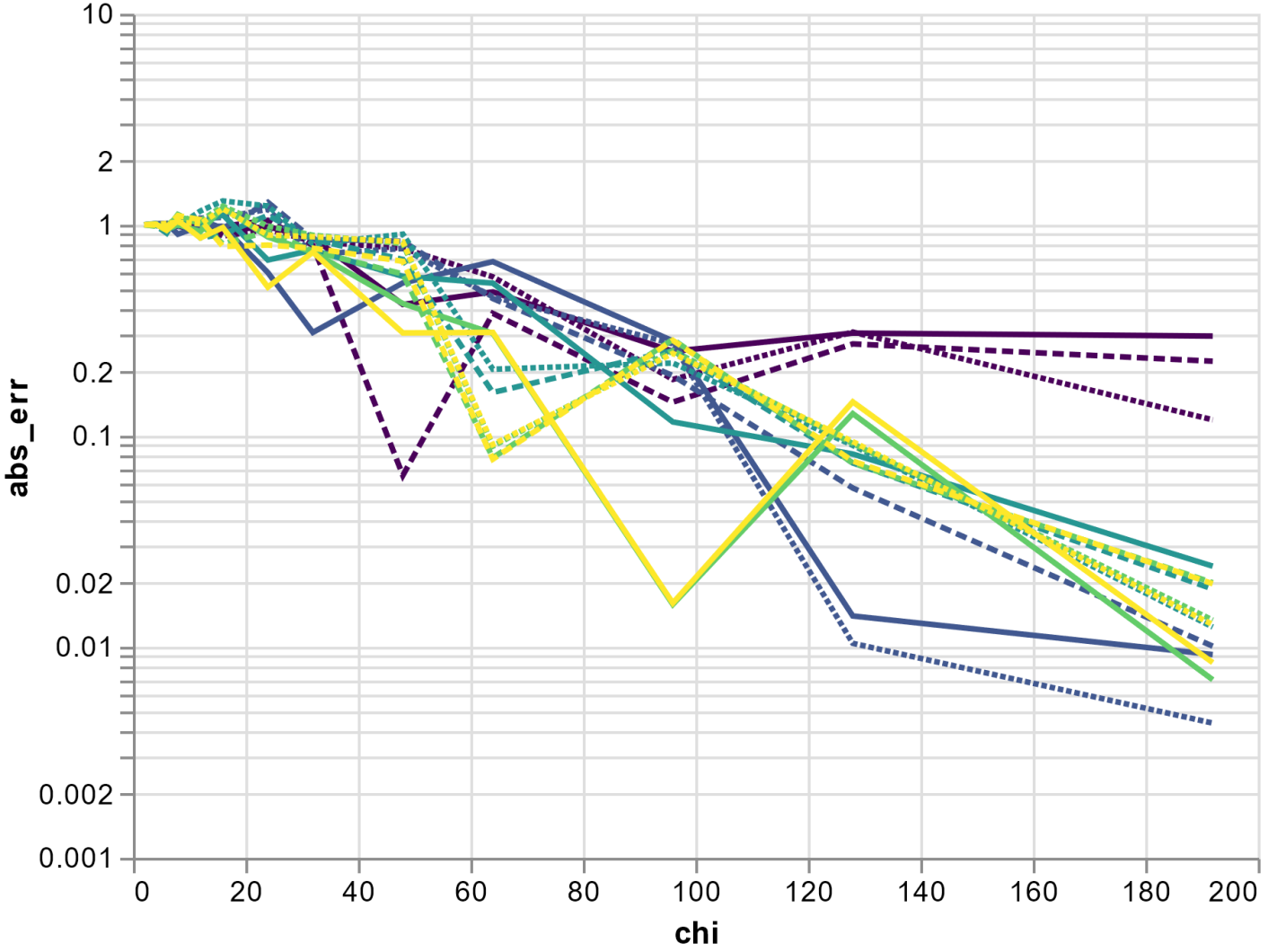


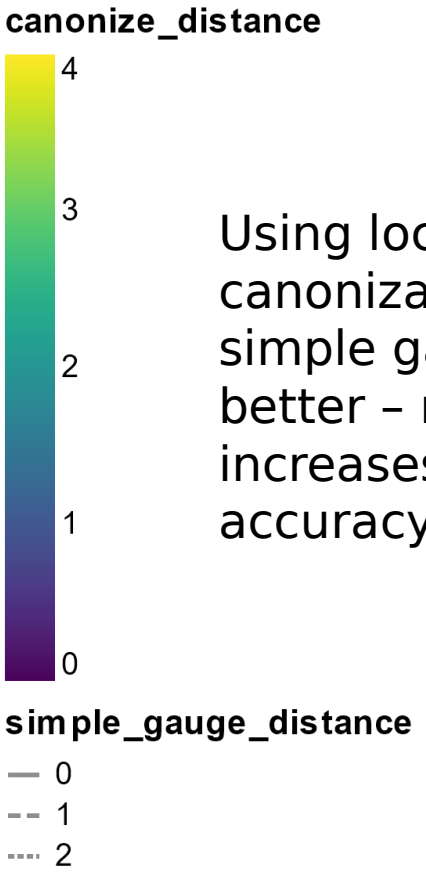
Results – 10x10 D=4 rand TN - [hard]



- Completely random 2D TN (i.e. not formed from random PEPS sandwich), much much harder
- Boundary contract not meaningful even with

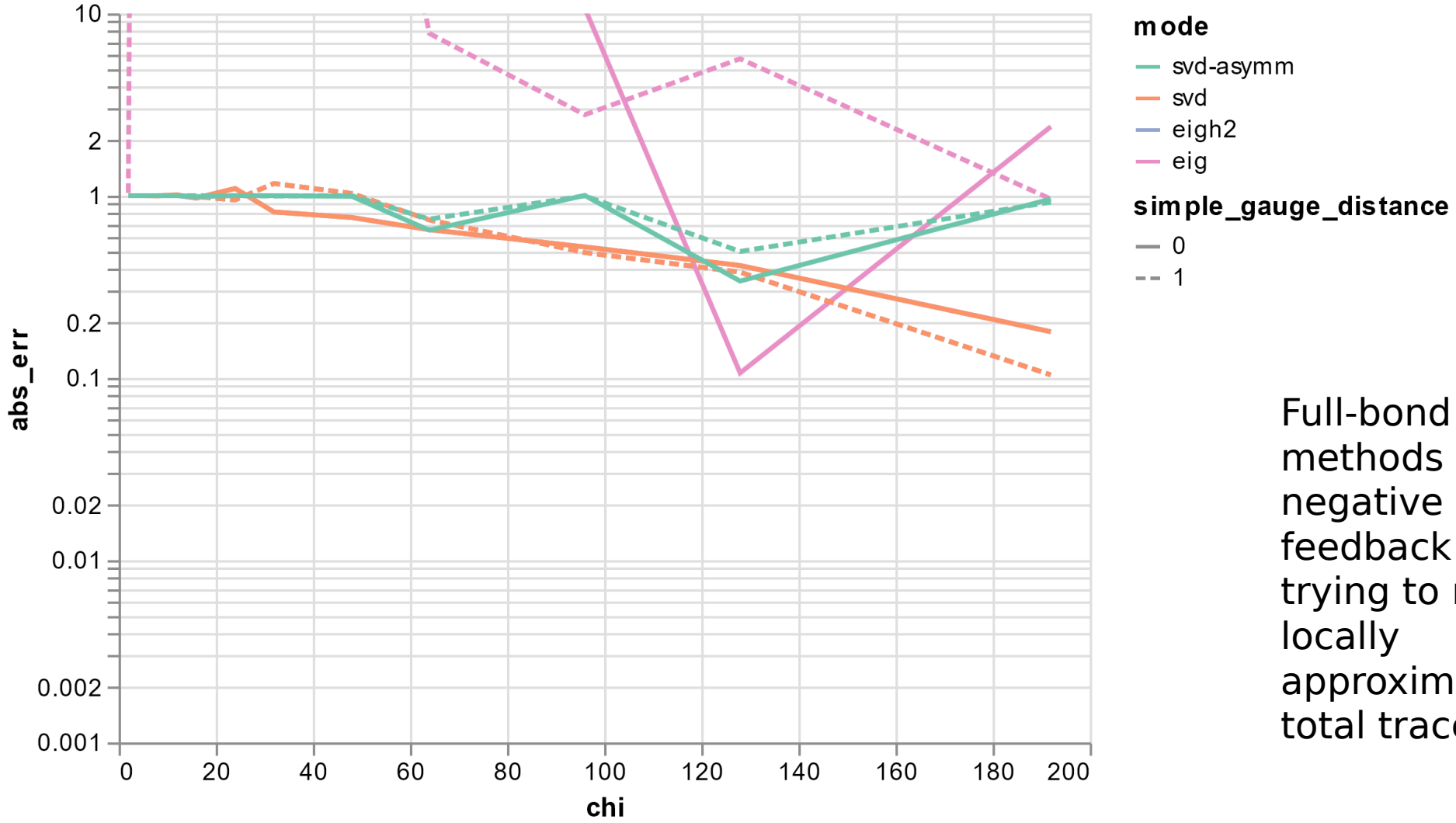
Results – 10x10 D=4 rand TN - [hard]





Using local canonization and simple gauging a bit better – reliable increases for accuracy with

Results – 10x10 D=4 rand TN - [hard]



Full-bond methods fail – negative feedback from trying to match locally approximated total trace value?