Coarse graining: towards a cylindrically consistent dynamics

Bianca Dittrich
(Perimeter Institute and Albert Einstein Institute)

[BD, 1205.6127, to appear in New J. Phys.]
[BD, Martin-Benito, v. Massenbach, w.i.p.]

Perimeter Institute for Theoretical Physics

Max Planck Research Group Canonical and Covariant Dynamics of Quantum Gravity

Max Planck Institute for Gravitational Physics (Albert Einstein Institute)

International Loop Quantum Gravity Seminar, Nov 2012

Overview

A. Motivation and summary

B. Coarse graining and continuum limit for classical system: the 2D scalar field

C. Formalization: cylindrical consistency

D. Coarse graining for quantum systems: tensor network algorithms

E. Applications to spinfoams / spinnets

thanks to collaborations and discussions with:
Benjamin Bahr, Valentin Bonzom, Frank Eckert, Frank Hellmann, Wojchiech Kaminski, Etera Livine, Mercedes Martin-Benito, Felix v. Massenbach, Erik Schnetter, Sebastian Steinhaus, Guifre Vidal, ...

Why coarse graining spin foams?

- Extract effective dynamics of the regime with many building blocks (‘large scale’ regime)

- Do the models lead to a phase describing 4D smooth manifolds on macroscopic scales?
  [Spin foams are generalized lattice gauge theories. Standard non-Abelian lattice gauge theory in 4D is believed to be confining, which correspond to a phase where degenerate geometries dominate.]

- Metric degrees of freedoms at all scales?

- Restoration of diff or triangulation/lattice independence? [Bahr, BD et al 09-11, Rovelli ’11]

- Large scale limit not equal `large j limit/few building blocks’ for spin foams?
  [Hellmann, Kaminski 12, Perini 12]

- applications to cosmology (effective dynamics for homogeneous modes), ...

Coarse graining state sums: splitting the sum

effective amplitude
includes sum over
finer field variables

amplitude function

$$Z = \sum_{\psi} a(\psi) = \sum_{\Psi} \sum_{\psi:\, B(\psi)=\Psi} a(\psi) = \sum_{\Psi} a'(\Psi)$$

coarse field variables

blocking of finer field variables

•How to block finer variables into coarser ones?

•What is the [finite dimensional] space of models, renormalization flow takes place in?

•How to truncate the flow back to this space?

•How to deal with non-local couplings?

•How to coarse grain the boundary?
Should we require triangulation independence for the boundary?

Questions for coarse graining

•How to block finer variables into coarser ones?

•What is the [finite dimensional] space of models, renormalization flow takes place in?

•How to truncate the flow back to this space?

•How to deal with non-local couplings?

•How to coarse grain the boundary?
Should we require triangulation independence for the boundary?

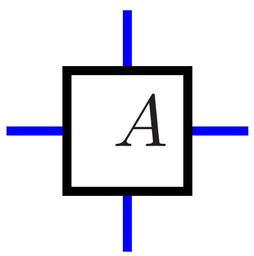
➡ tensor network renormalization provides answers
•THIS TALK:
-procedure for classical systems: blocking and truncation chosen by hand [BD 12]
-procedure for quantum systems: blocking and truncation chosen dynamically
[methods developed in condensed matter/ q-information: Levin-Nave, Wen-Gu, Vidal, Verstraete, ...'00's+]

Summary of the method

State sums with (generalized) boundaries

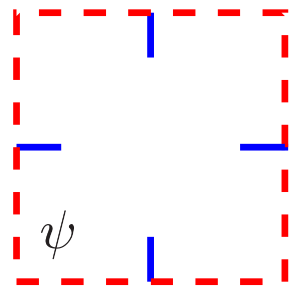
State sum models associate amplitudes to space time regions with boundary (data)

[Oeckl 03]

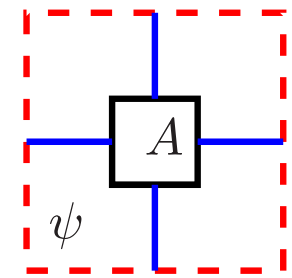


$$A(x_1, x_2, x_3, x_4) = \sum_{x_{\text{bulk}}} a(x_1, x_2, x_3, x_4, x_{\text{bulk}})$$

where x are boundary data



$\psi(x_1, x_2, x_3, x_4)$
is a boundary wave function

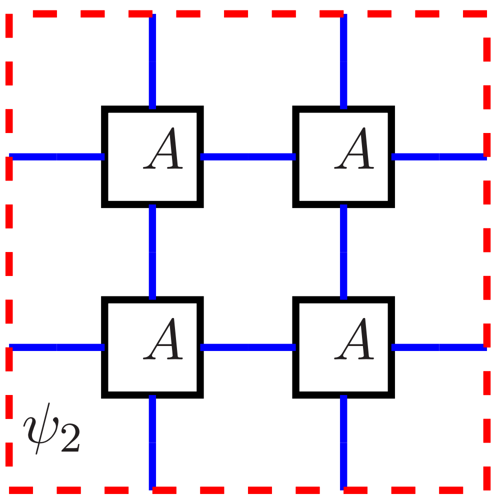


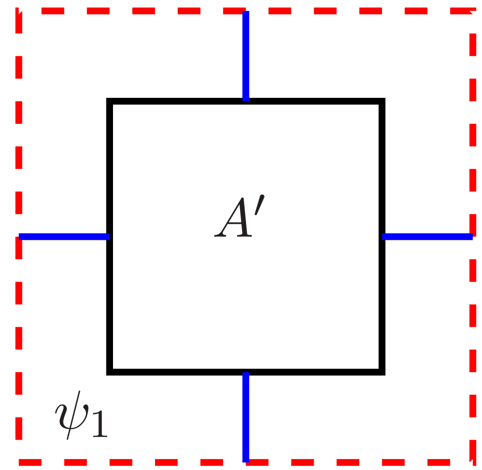
A is an (anti-)linear functional on bdry Hilbert space $\mathcal{H}_1$,

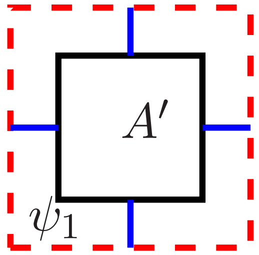
$$A(\psi) = \sum_{x_i} A(x_i)\bar{\psi}(x_i)$$

defines (transition) amplitudes

Coarse graining space time regions





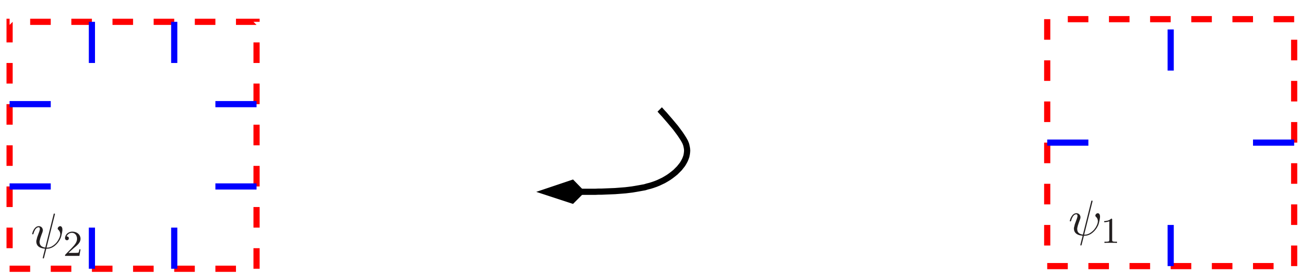


Amplitude for a ‘larger’ region glued from amplitudes of smaller regions, acts on ‘refined’ bdry Hilbert space $\mathcal{H}_2$

We want to define an effective amplitude acting on coarser boundary Hilbert space $\mathcal{H}_1$

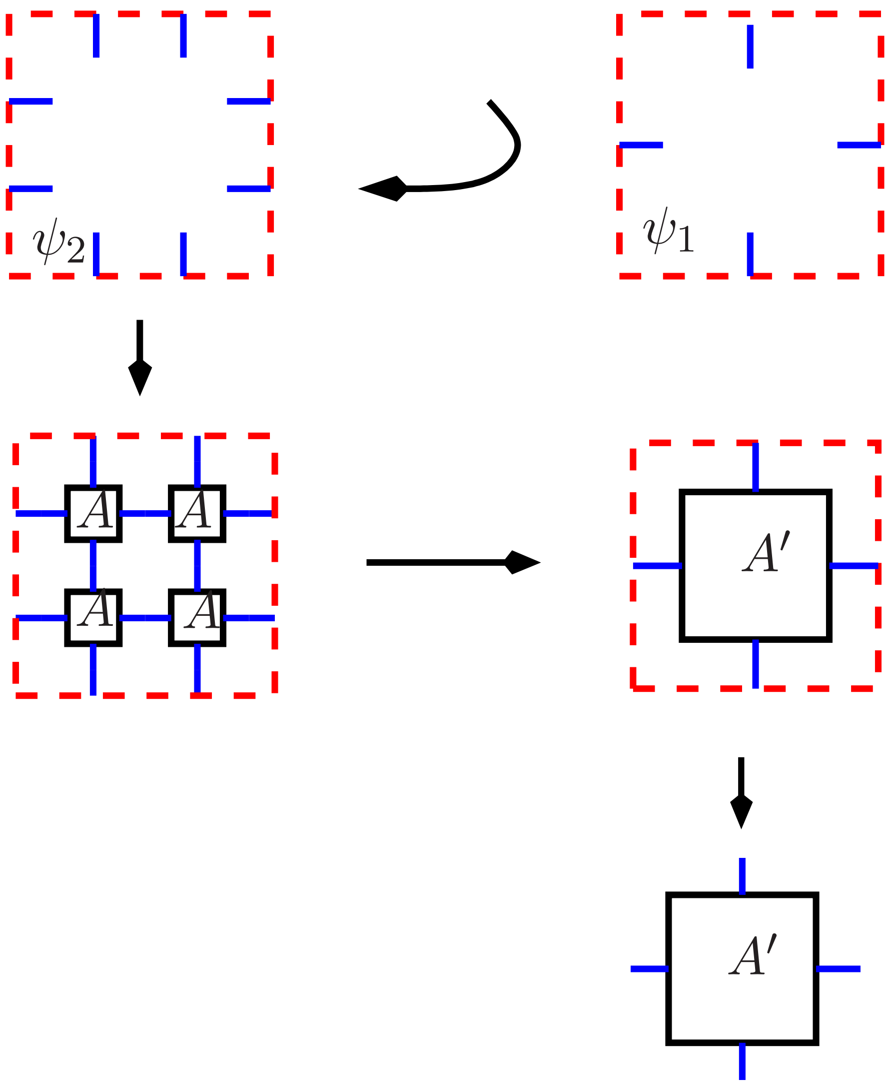
Take (rescaled) effective amplitude as new amplitude for original region

(no rescaling necessary for gravity or reparametrization invariant systems)



Need to relate coarser and finer bdry Hilbert spaces by embedding maps

Embedding boundaries



Via the embedding map we can find the effective amplitude functional A' on $\mathcal{H}_1$.

Take A' as new amplitude functional. Iterate and find fixed point.

Classical procedure: 2D scalar field

[BD New J. Phys. 12]

Classical procedure: 2D scalar field

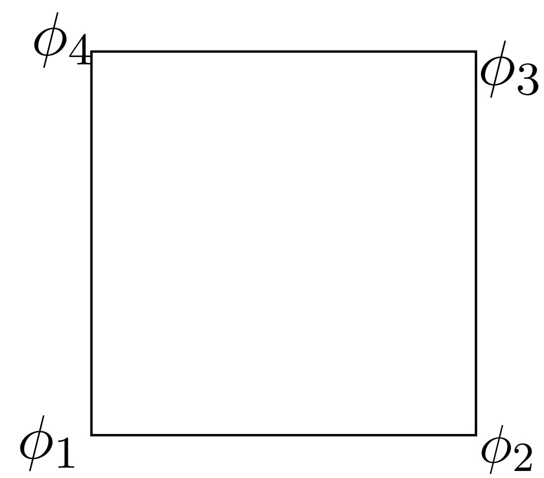
quantum | **classical**

Amplitude functional $\longrightarrow$ Hamilton's (principal) function

$||$

action evaluated on solution
depending on boundary data

Discrete action (for field theories):
first guess of Hamilton's principal function
for basic building blocks
depending on discretization of boundary data

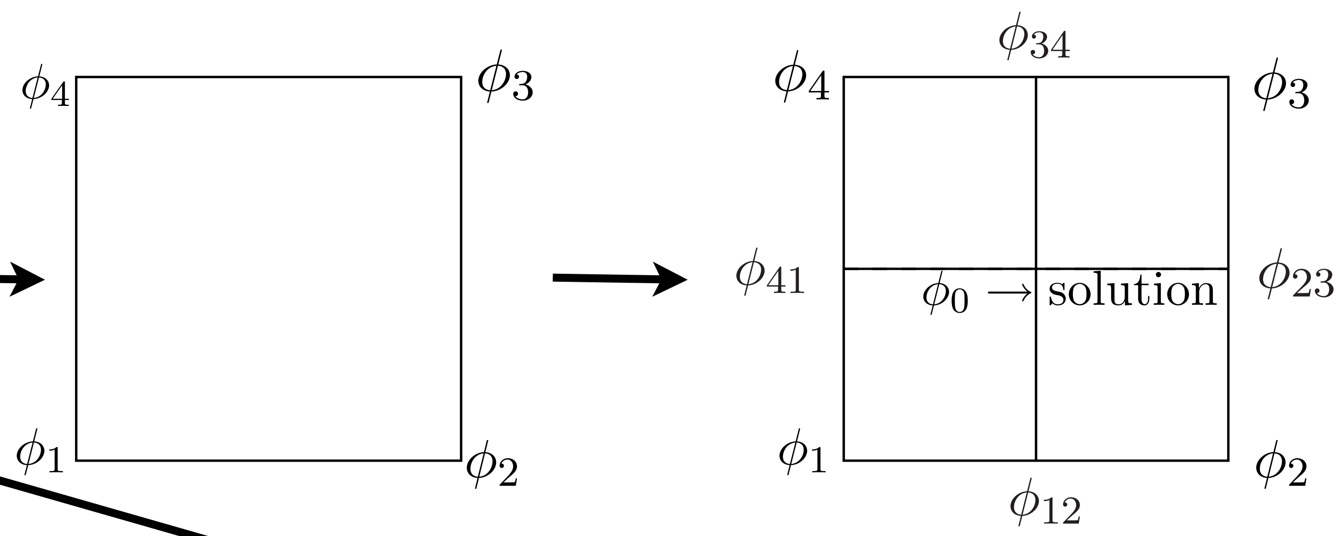


-basic building block: square
-scalar field ϕ associated to vertices
-action for massless free field

$$S_4 = (\phi_1 + \phi_2)^2 + (\phi_2 + \phi_3)^2 + (\phi_3 + \phi_4)^2 + (\phi_4 + \phi_1)^2$$

Iteration procedure

$S_4(\phi_1, \ldots, \phi_4)$ →



→ $S_8(\phi_1, \ldots, \phi_4, \phi_{12}, \ldots, \phi_{41})$

piecewise linear embedding

$\iota : \mathcal{C}_4 \to \mathcal{C}_8$ defined by

$\phi_{12} = \frac{1}{2}(\phi_1 + \phi_2), \ldots$

$S_4'(\phi_1, \ldots, \phi_4) =$

$S_8(\phi_1, \ldots, \frac{1}{2}(\phi_1 + \phi_2), \ldots)$

iterate

flow in parameter α:

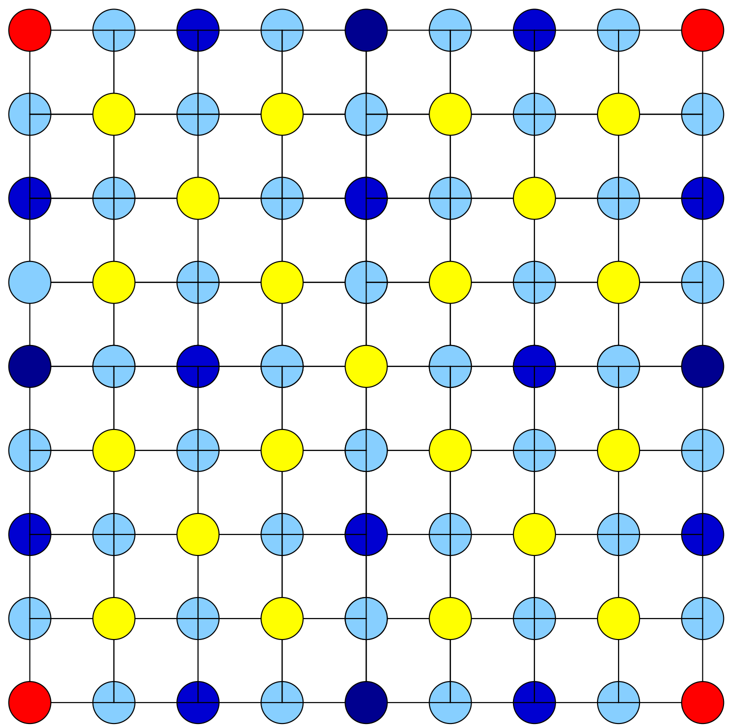
$$S_4' = (\phi_1 + \phi_2)^2 + (\phi_2 + \phi_3)^2 + (\phi_3 + \phi_4)^2 + (\phi_4 + \phi_1)^2 - \alpha(\phi_1 - \phi_2 + \phi_3 - \phi_4)^2$$

diagonal couplings

fixed point: $\alpha^* = \frac{2}{3}$

Understanding the approximation

After N iteration find an approximation to Hamilton's function for square with 2^N basic squares and 'edge wise' linear boundary fields.



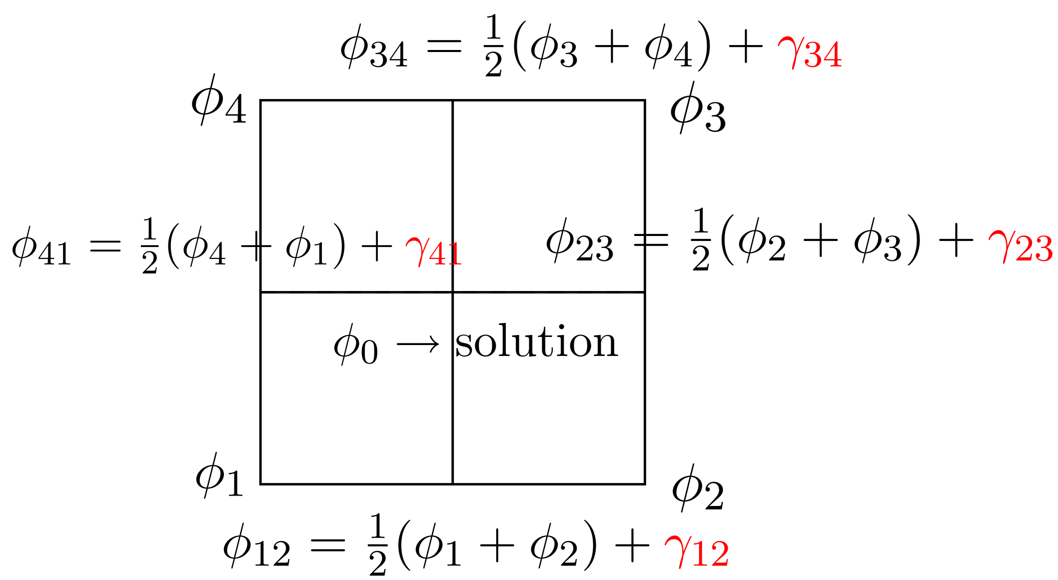
- free boundary fields
- eom's have been solved for these fields
- approximated fields in first iteration
- approximated fields in second iteration
- approximated fields in second iteration

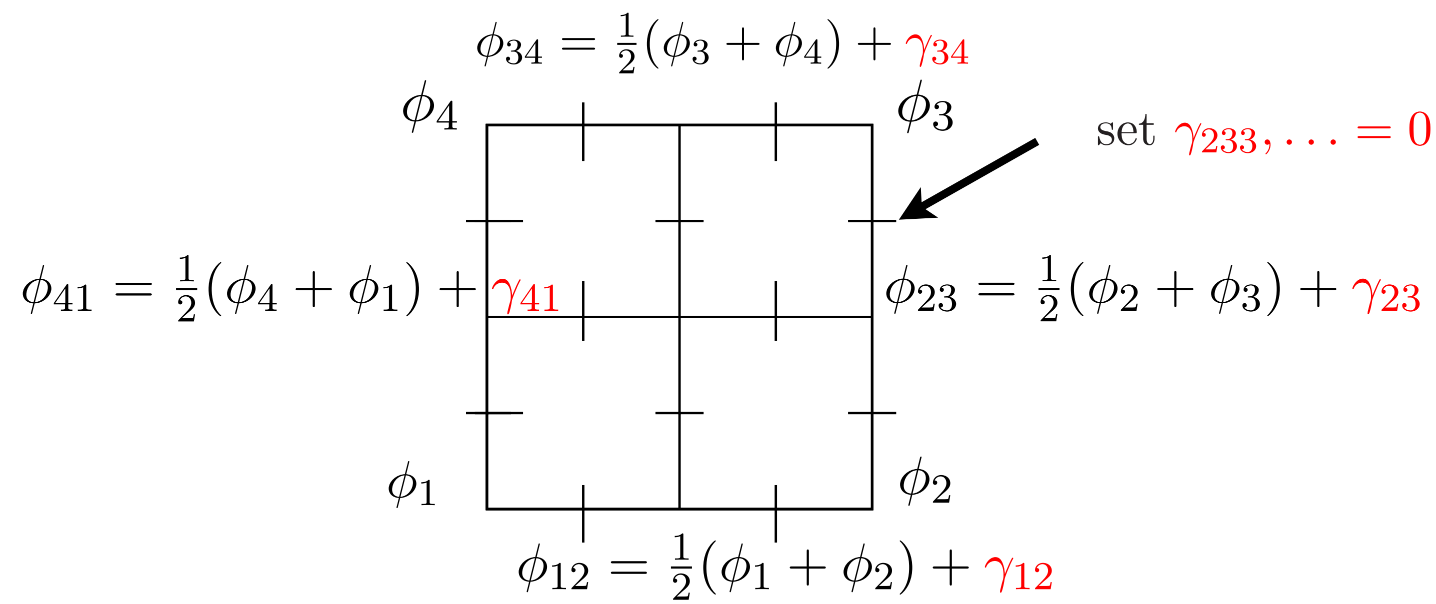
Fixed point: approximation to continuum Hamilton's function evaluated on 'edge wise' linear boundary data.

For free massless scalar field actually exact!

The same procedure for squares with refined boundary data will in general give a correction to this approximation.

For more refined boundary data





Start with Hamilton's function for refined boundary data.

Glue 4 such squares.

Find new Hamilton's function, evaluate on same type of boundary data

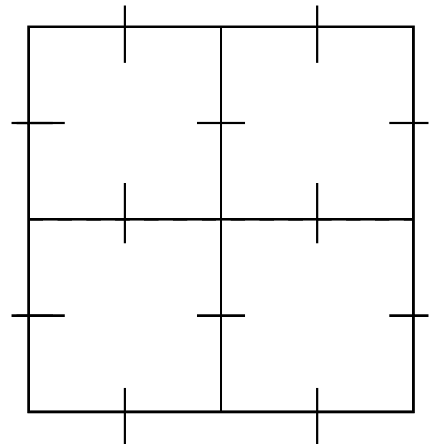
Fixed point action:

$$
\begin{aligned}
S_8^* \quad = \quad & S_4^*(\phi_i) + \\
& \phi_1\gamma_{12} + \ldots - \phi_1\gamma_{23} + \ldots + \\
& 2.28\gamma_{12}^2 + \ldots - 1.20\gamma_{12}\gamma_{23} + \ldots - 0.34\gamma_{12}\gamma_{34} + \ldots
\end{aligned}
$$

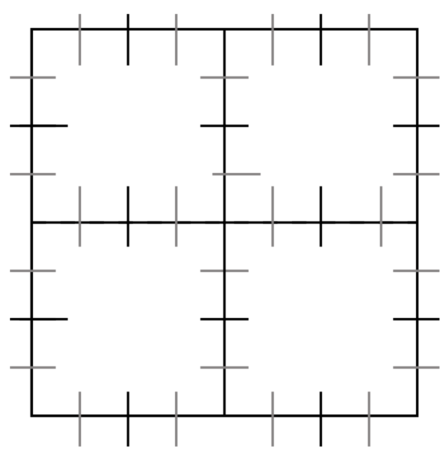
Special for massless case: this part does not change.

We can find Hamilton's function for more and more refined boundary data.

Even more refined boundary data



$$\begin{aligned} S_8^* &= S_4^*(\phi_i) + \\ &\phi_1\gamma_{12} + \ldots - \phi_1\gamma_{23} + \ldots + \\ &2.28\gamma_{12}^2 + \ldots - 1.20\gamma_{12}\gamma_{23} + \ldots - 0.34\gamma_{12}\gamma_{34} + \ldots \end{aligned}$$

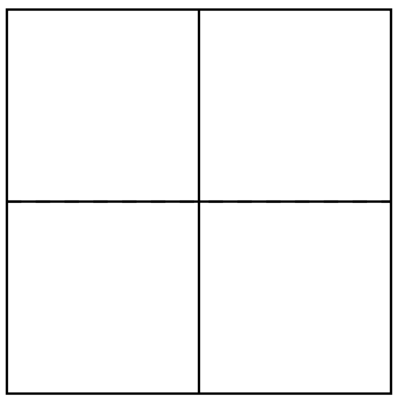


$$\begin{aligned} S_{16}^* &= S_4^*(\phi_i) + \\ &\phi_1\gamma_{12} + \ldots - \phi_1\gamma_{23} + \ldots + \\ &2.20\gamma_{12}^2 + \ldots - 1.18\gamma_{12}\gamma_{23} + \ldots 0.36\gamma_{12}\gamma_{34} + \ldots + \\ &\phi\cdot\kappa \;+\; \gamma\cdot\kappa \;+\; \kappa\cdot\kappa \quad -\text{terms} \end{aligned}$$

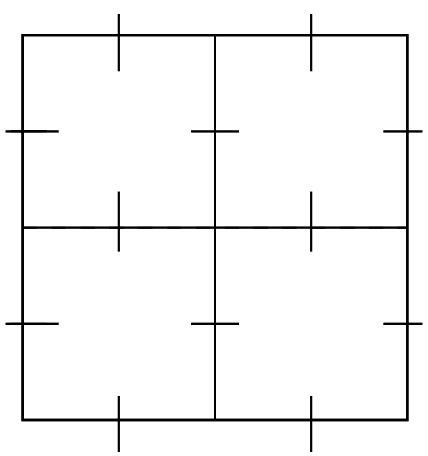
Comparing the fixed points found for different truncations allows to judge the convergence to the continuum result.
To compare the fixed points we need to use the embedding maps.

Nonlinear potential

$$S = \ldots \lambda a^2 (\phi_1^4 + \phi_2^4 + \phi_3^4 + \phi_4^4)$$



include gamma fields



Only flow in second order in lambda.

- in 4 to 1 square coarse graining scheme:

$$S_4^* = \ldots - 0.0039 \lambda^2 a^4 \phi_1^6 + \ldots$$

include gamma fields

$$S_8^* = \ldots - 0.0050 \lambda^2 a^4 \phi_1^6 + \ldots$$

- in 16 to 1 square coarse graining scheme:

$$S_4^* = \ldots - 0.0045 \lambda^2 a^4 \phi_1^6 + \ldots$$

include gamma fields

$$S_8^* = \ldots - 0.0051 \lambda^2 a^4 \phi_1^6 + \ldots$$

```
1.3333333333315365` f1^2 - 0.6666666666684854` f1 f2 +
1.3333333333314925` f2^2 - 1.3333333333342485` f1 f3 - 0.6666666666685183` f2 f3
1.3333333333314699` f3^2 - 0.666666666668474` f1 f4 - 1.3333333333342474` f2 f4 -
0.6666666666685073` f3 f4 + 1.3333333333315154` f4^2 + 0.9999999999981569` f1 g1
0.9999999999981566` f2 g12 - 1.0000000000006235` f3 g12 - 1.000000000000623` f4
2.275571229481218` g12^2 + 0.9999999999981743` f1 g14 - 1.000000000000623` f2 g1
1.0000000000006235` f3 g14 + 0.9999999999981741` f4 g14 - 1.2003244371886126` g
2.275571229481229` g14^2 - 1.0000000000006235` f1 g23 + 0.9999999999981071` f2 g
0.9999999999981073` f3 g23 - 1.000000000000623` f4 g23 - 1.2003244371886133` g1
0.342740787679371 44` g14 g23 + 2.275571229481184` g23^2 - 1.0000000000006237` f1
1.000000000000623` f2 g34 + 0.9999999999981246` f3 g34 + 0.9999999999981241` f4
0.342740787679371 33` g12 g34 - 1.2003244371886135` g14 g34 -
1.2003244371886128` g23 g34 + 2.2755712294811956` g34^2 + 0.16000050862673076` A
0.15999999999980186` A^2 f1^3 f2 λ + 0.16000025431313264` A^2 f1^2 f2^2 λ +
0.15999999999980186` A^2 f1 f2^3 λ + 0.16000050862673076` A^2 f2^4 λ +
0.03999987284364215` A^2 f1^3 f3 λ + 0.07999987284355044` A^2 f1^2 f2 f3 λ +
0.11999961853091433` A^2 f1 f2^2 f3 λ + 0.1599999999998019` A^2 f2^3 f3 λ +
0.026666666666676175` A^2 f1^2 f3^2 λ + 0.079999872843550 45` A^2 f1 f2 f3^2 λ +
0.1600002543131327` A^2 f2^2 f3^2 λ + 0.03999987284364215` A^2 f1 f3^3 λ +
0.15999999999980194` A^2 f2 f3^3 λ + 0.16000050862673076` A^2 f3^4 λ +
0.1599999999998019` A^2 f1^3 f4 λ + 0.11999961853091431` A^2 f1^2 f2 f4 λ +
0.07999987284355041` A^2 f1 f2^2 f4 λ + 0.03999987284364212` A^2 f2^3 f4 λ +
0.07999987284355048` A^2 f1^2 f3 f4 λ + 0.10666666666670467` A^2 f1 f2 f3 f4 λ +
0.07999987284355041` A^2 f2^2 f3 f4 λ + 0.07999987284355042` A^2 f1 f3^2 f4 λ +
0.11999961853091436` A^2 f2 f3^2 f4 λ + 0.15999999999980186` A^2 f3^3 f4 λ +
0.16000025431313267` A^2 f1^2 f4^2 λ + 0.07999987284355042` A^2 f1 f2 f4^2 λ +
0.026666666666667615` A^2 f2^2 f4^2 λ + 0.11999961853091436` A^2 f1 f3 f4^2 λ +
0.07999987284355041` A^2 f2 f3 f4^2 λ + 0.16000025431313267` A^2 f3^2 f4^2 λ +
0.1599999999998019` A^2 f1 f4^3 λ + 0.03999987284364212` A^2 f2 f4^3 λ +
0.15999999999980188` A^2 f3 f4^3 λ + 0.16000050862673076` A^2 f4^4 λ +
0.23404611294318675` A^2 f1^3 g12 λ + 0.38019491748597123` A^2 f1^2 f2 g12 λ +
0.3801949174859712` A^2 f1 f2^2 g12 λ + 0.23404611294318672` A^2 f2^3 g12 λ +
0.07060152907909614` A^2 f1^2 f3 g12 λ + 0.14120305815818512` A^2 f1 f2 f3 g12 λ +
0.13371304570645157` A^2 f2^2 f3 g12 λ + 0.03576923209637391` A^2 f1 f3^2 g12 λ +
0.06878200606574425` A^2 f2 f3^2 g12 λ + 0.02756813187199903` A^2 f3^3 g12 λ +
0.13371304570645154` A^2 f1^2 f4 g12 λ + 0.14120305815818 5` A^2 f1 f2 f4 g12 λ +
0.0706015290790961` A^2 f2^2 f4 g12 λ + 0.07153846419274867` A^2 f1 f3 f4 g12 λ +
0.0715384641927487` A^2 f2 f3 f4 g12 λ + 0.04273643894668 6967` A^2 f3^2 f4 g12 λ +
0.06878200606574421` A^2 f1 f4^2 g12 λ + 0.03576923209637387` A^2 f2 f4^2 g12 λ +
0.04273643894668695` A^2 f3 f4^2 g12 λ + 0.02756813187199901` A^2 f4^3 g12 λ +
0.2689800157390 8337` A^2 f1^2 g12^2 λ + 0.43249761240805384` A^2 f1 f2 g12^2 λ +
0.2689800157390 8326` A^2 f2^2 g12^2 λ + 0.06064067843449934` A^2 f1 f3 g12^2 λ +
0.0770262228913406` A^2 f2 f3 g12^2 λ + 0.014802224472345759` A^2 f3^2 g12^2 λ +
0.07702622289134058` A^2 f1 f4 g12^2 λ + 0.060640678434499315` A^2 f2 f4 g12^2 λ +
0.022935581617311747` A^2 f3 f4 g12^2 λ + 0.014802224472345747` A^2 f4^2 g12^2 λ +
0.19877467155568482` A^2 f1 g12^3 λ + 0.1987746715556848` A^2 f2 g12^3 λ +
0.021982795409024473` A^2 f3 g12^3 λ + 0.021982795409024466` A^2 f4 g12^3 λ +
0.0672544547227 4261` A^2 g12^4 λ + 0.2340461129431868` A^2 f1^3 g14 λ +
0.13371304570645154` A^2 f1^2 f2 g14 λ + 0.06878200606574422` A^2 f1 f2^2 g14 λ +
0.027568131871999014` A^2 f2^3 g14 λ + 0.07060152907909616` A^2 f1^2 f3 g14 λ +
0.0715384641927487` A^2 f1 f2 f3 g14 λ + 0.04273643894668697` A^2 f2^2 f3 g14 λ +
0.03576923209637391` A^2 f1 f3^2 g14 λ + 0.042736438946686994` A^2 f2 f3^2 g14 λ +
0.02756813187199903` A^2 f3^3 g14 λ + 0.38019491748597134` A^2 f1^2 f4 g14 λ +
0.14120305815818504` A^2 f1 f2 f4 g14 λ + 0.03576923209637388` A^2 f2^2 f4 g14 λ +
0.14120305815818504` A^2 f1 f3 f4 g14 λ + 0.07153846419274867` A^2 f2 f3 f4 g14 λ +
```

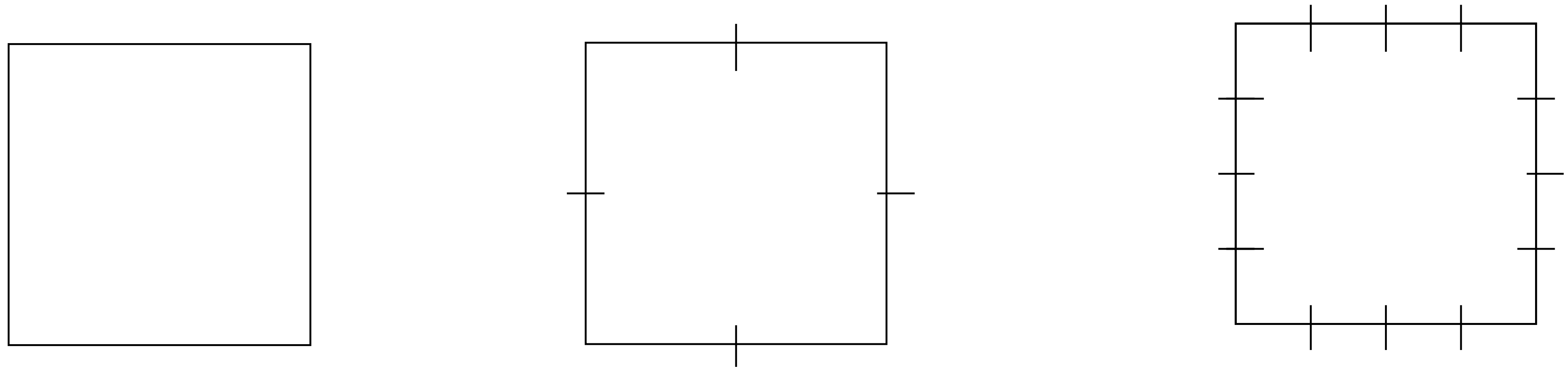
Hamilton's function: page 1 of 20 ...

Applications

[more discussion in BD New J. Phys. 12]

- the set-up can be understood as introducing complex building blocks with local couplings between building blocks instead of simple building blocks with complicated non-local couplings
- this can be used to define an (renormalization) improved discretization/numerical scheme
- leads to higher order difference equations, but in a controlled way
- can be used to find perfect discretizations and to define continuum limit

Formalize: cylindrical consistency



Embedding of configuration spaces into each other: $\iota_{bb'} : \mathcal{C}_b \to \mathcal{C}_{b'}$

example: $\iota_{bb'}(\phi_1, \phi_2, \phi_3, \ldots) = (\phi_1, \frac{1}{2}(\phi_1 + \phi_2), \phi_2, \frac{1}{2}(\phi_2 + \phi_3), \phi_3, \ldots)$

consistency: $\iota_{bb''} = \iota_{b'b''} \circ \iota_{bb'}$

continuum configuration space / inductive limit:

$$\mathcal{C}_{\text{ind}} = \cup_b \mathcal{C}_b / \sim$$

$c_b \sim c_{b'}$ if $\iota_{bb''}(c_b) = \iota_{b'b''}(c_{b'})$

Formalization: cylindrical consistency

We are asking for a cylindrically consistent Hamilton's function:

$$\{S_b\}_{b\in\mathcal{B}} \text{ is cylindrically consistent}$$

$$S_b = \iota^\star_{bb'} S_{b'} \quad \text{i.e.} \quad S_b(c) = S_{b'}(\iota_{bb'}(c)) \quad \forall c \in \mathcal{C}_b$$

We approximate these as fixed point actions involving two boundaries:

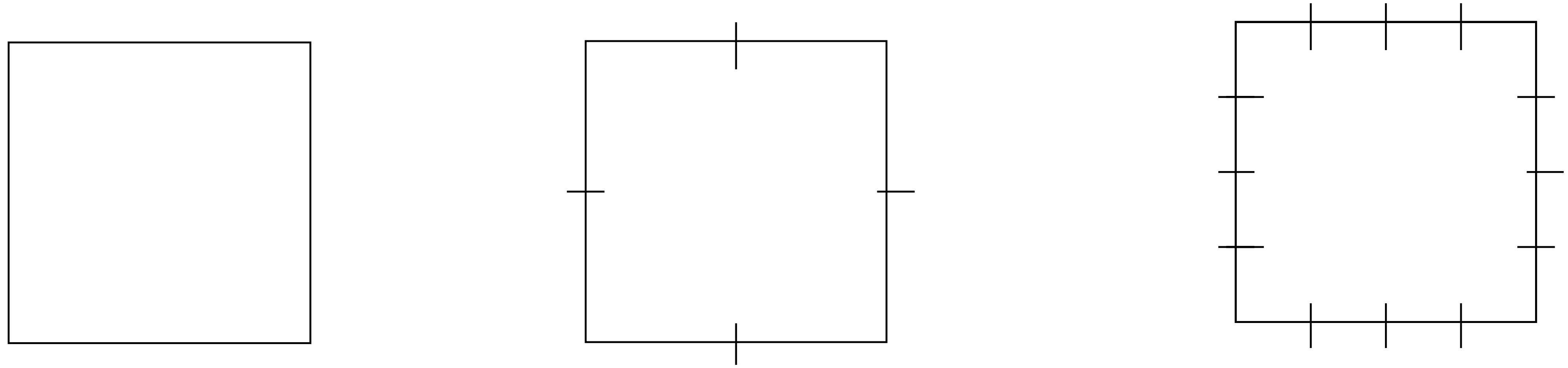
$$\iota^\star_{bb'} S^b_{b'} = S^*_b$$

Hamilton's function for (finer) b' computed from (fixed point) action for (coarser) b

If S^*_b does not depend on choice of (finer) boundary b' is coincides with continuum result.

Cylindrically consistent dynamics: cylindrically consistent Hamilton's function.
Gives continuum result for discrete bdry data (which represent continuum bdry data).

Towards quantum theory



•Configuration spaces are replaced by Hilbert spaces: $\iota_{bb'}: \quad \mathcal{H}_b \to \mathcal{H}_{b'}$

•Hamilton's function is replaced by amplitude map acting on boundary Hilbert space: $A_b : \mathcal{H}_b \mapsto \mathbb{C}$

associates an amplitude (physical vacuum wave function) to region with boundary b

•Cylindrical consistency: $(\iota_{bb'})^\star A_{b'}(\psi_b) = A_{b'}(\iota_{bb'}(\psi_b))$

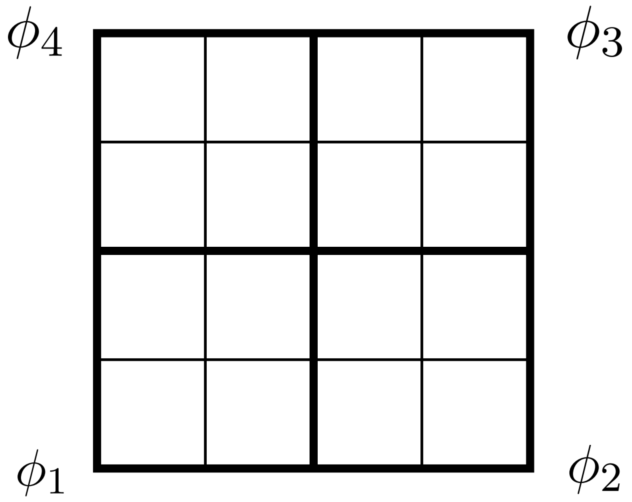
i.e. result does not depend on which boundary b we perform computation.

Cylindrically consistent dynamics: cylindrically consistent amplitude map.
Gives continuum result for discrete bdry data (which represent continuum bdry data).

Choice of embeddings

- determines quality of approximation

- should be adjusted to the dynamics of the system:
  Ideal case: embedding reproduces behaviour of solution (along inner boundaries)



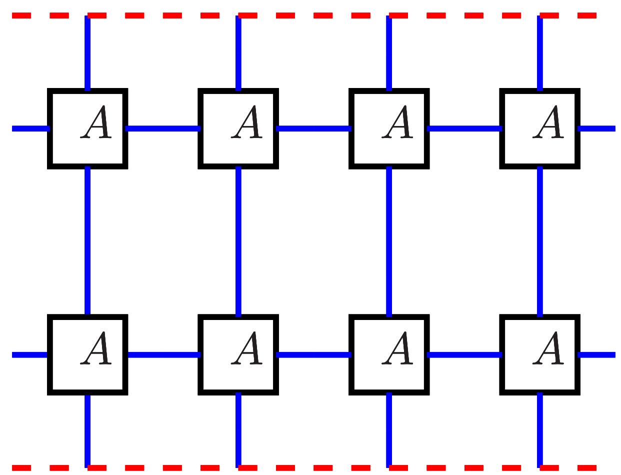
Solution with edgewise linear bdry data leads also to edgewise linear data for smaller squares.

Choice of embeddings becomes even more crucial in quantum theory.
Implemented into algorithms based on tensor networks.

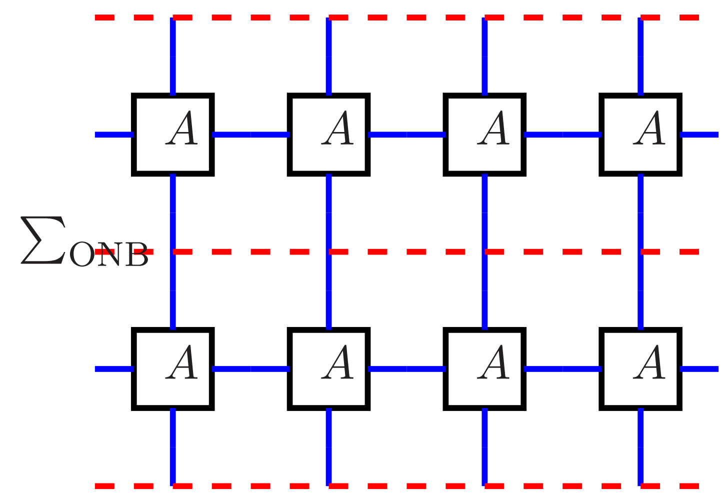
Quantum theory: tensor network algorithms

[Levin & Nave, Gu & Wen, Vidal ...'00's+]
[BD, Eckert, Martin-Benito, New. J. Phys. '11]
[BD, Martin-Benito, v. Massenbach w.i.p.]

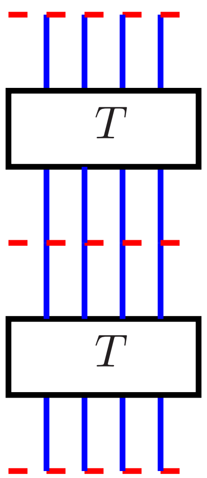
Motivation: transfer operator technique



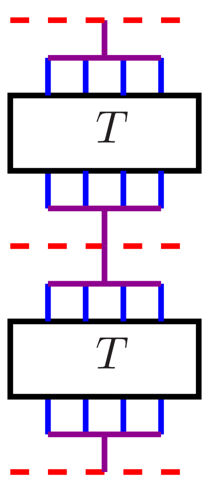
Transition amplitude between two states $\langle\psi_1|\mathcal{A}|\psi_2\rangle$



insert id $= \sum_{\text{ONB}} |\psi\rangle\langle\psi|$



$\mathcal{A} = T^N$

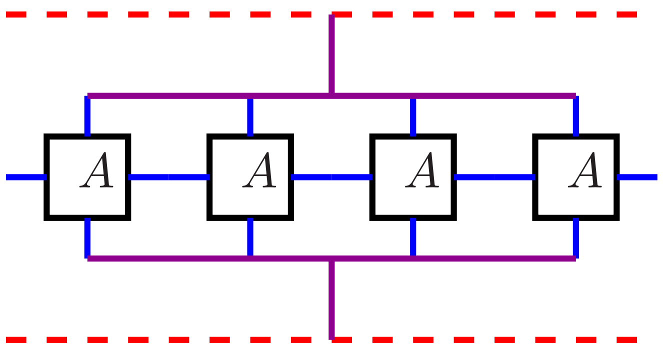


Truncate by restricting $\sum_{\text{ONB}}$ to the eigenvectors of T with the χ largest (in mod) eigenvalues.

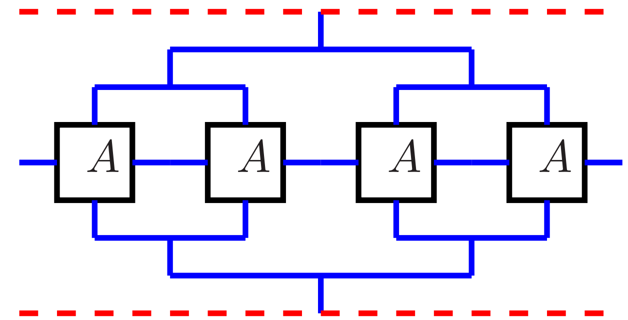
Expect good approximation if ψ_1, ψ_2 are in span of these eigenvectors.

But: explicit diagonalization of T difficult.

Dynamically determined embedding maps

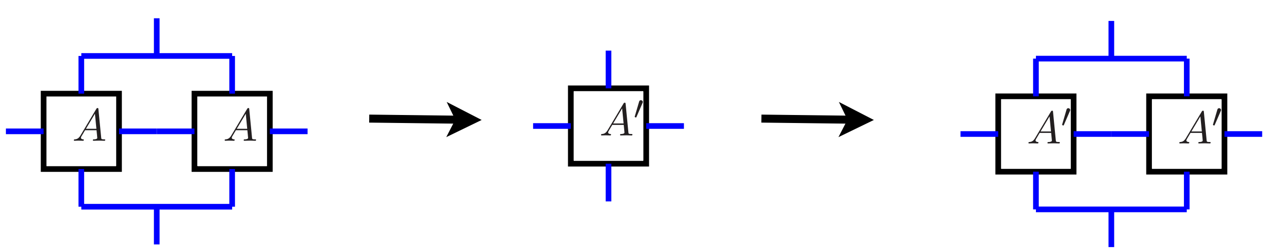


Truncate by restricting $\sum_{\mathrm{ONB}}$ to the eigenvectors of T with the χ largest (in mod) eigenvalues.

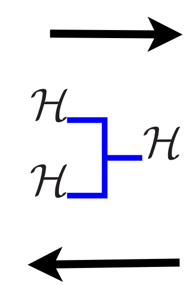


Localize truncations, diagonalize only subparts of transfer operator

iteration procedure



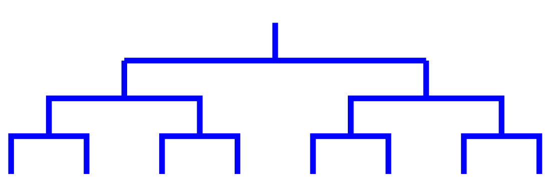
blocking



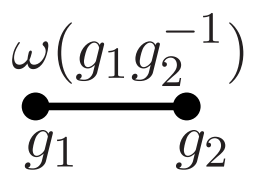
Determined by (generalized) EV-decomposition.

embedding

embedding map after 3 iterations

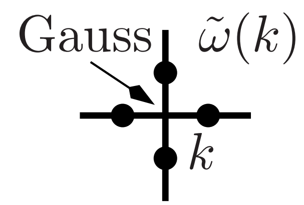


Example: Ising model

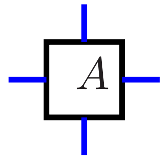


group elements ± 1
at vertices,
edge weights ω

$\longrightarrow$ Fourier trafo



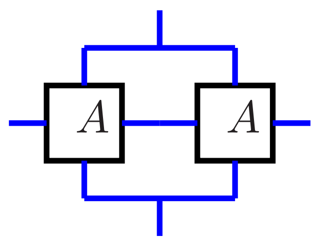
rep labels $k = 0, 1$ at edges
edge weights $\tilde{\omega}(k)$
Gauss constraints at vertices



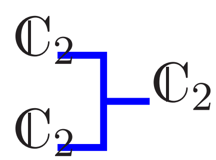
$$A(k_1, \ldots, k_4) = \sqrt{\tilde{\omega}_1} \cdots \sqrt{\tilde{\omega}_4}\, \delta(k_1 + k_2 - k_3 - k_4)$$

Example: Ising model

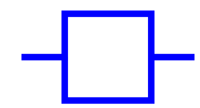
determine embedding maps



embedding maps



condition on embedding maps



$= \mathrm{id}$

Embedding maps parametrized by:

$\cos(\alpha)$ $\sin(\alpha)$ $1/\sqrt{2}$ $1/\sqrt{2}$

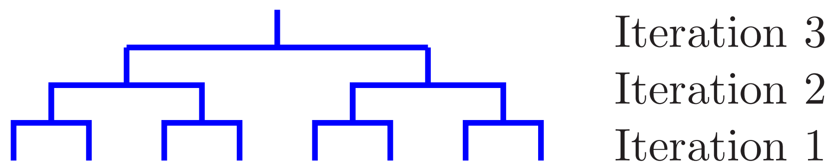
high temperature: $\cos\alpha = 1,\ \alpha = 0$
(symmetric phase)

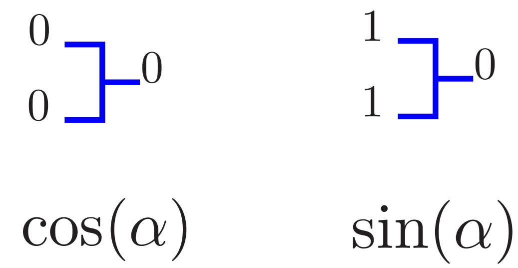
$\tilde{\omega}(1) = 0$ $\alpha = 0$

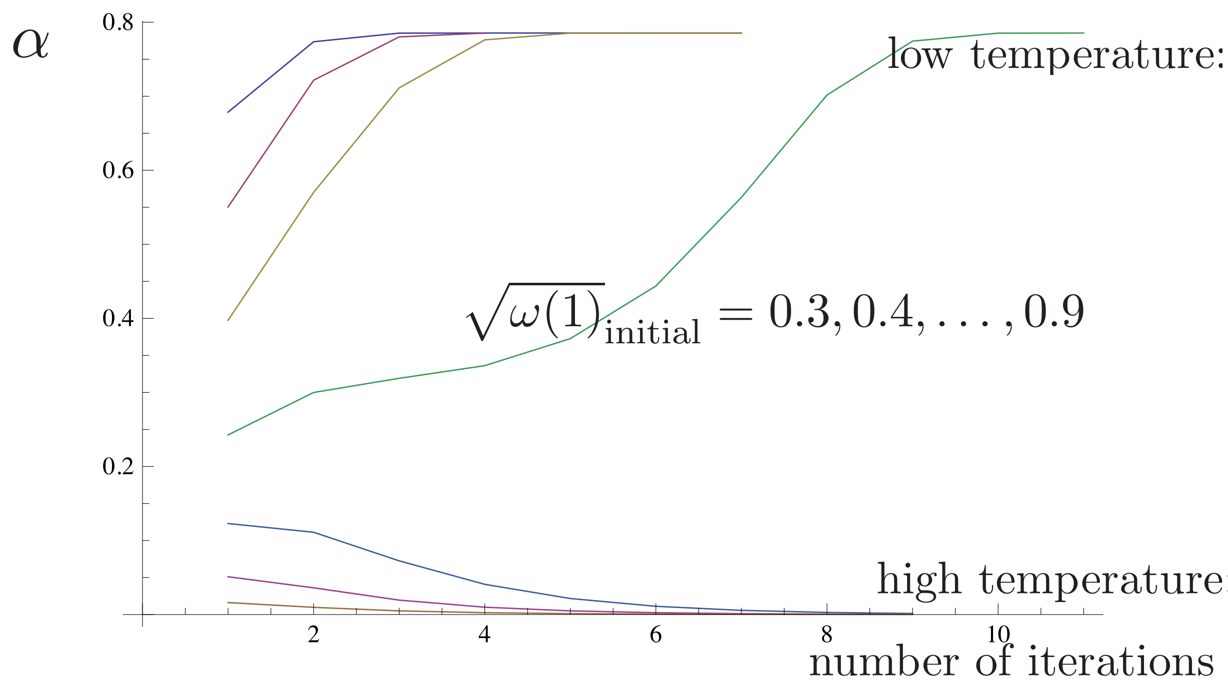
$\tilde{\omega}(1) = 1$ $\alpha = \frac{\pi}{4}$

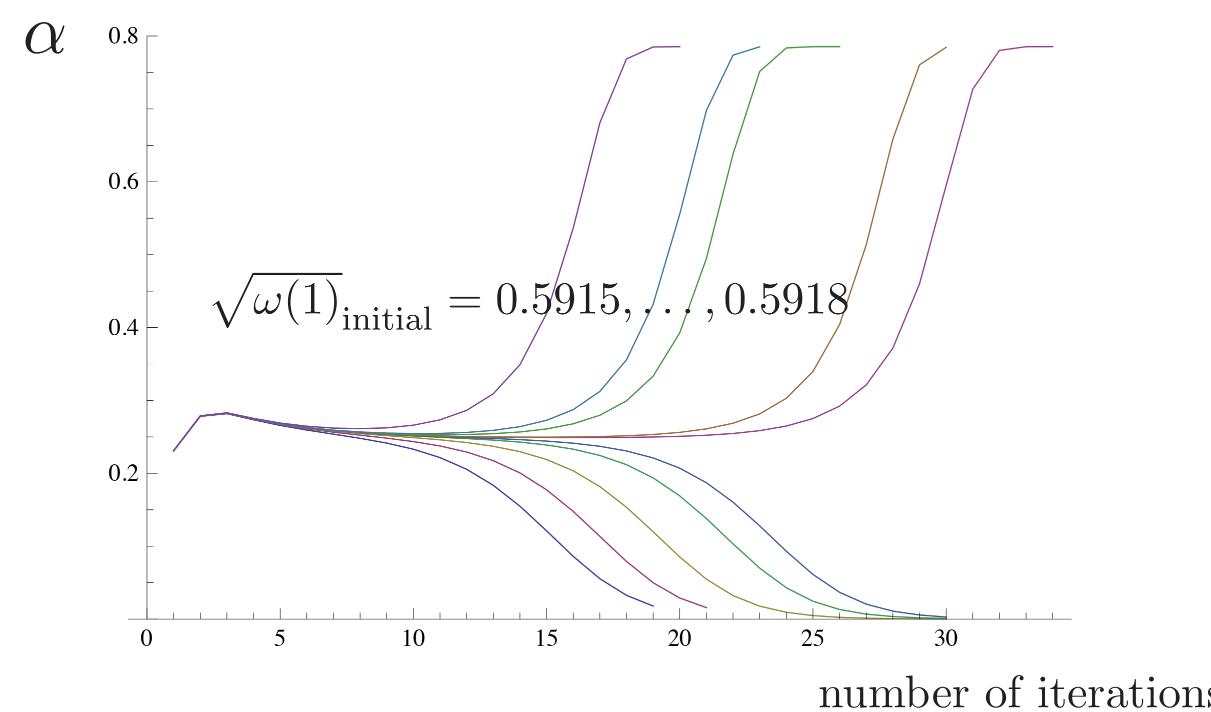
low temperature: $\cos\alpha = \sin\alpha = \frac{1}{\sqrt{2}},\ \alpha = \frac{\pi}{4}$
(symmetry broken phase)

Example: Ising model





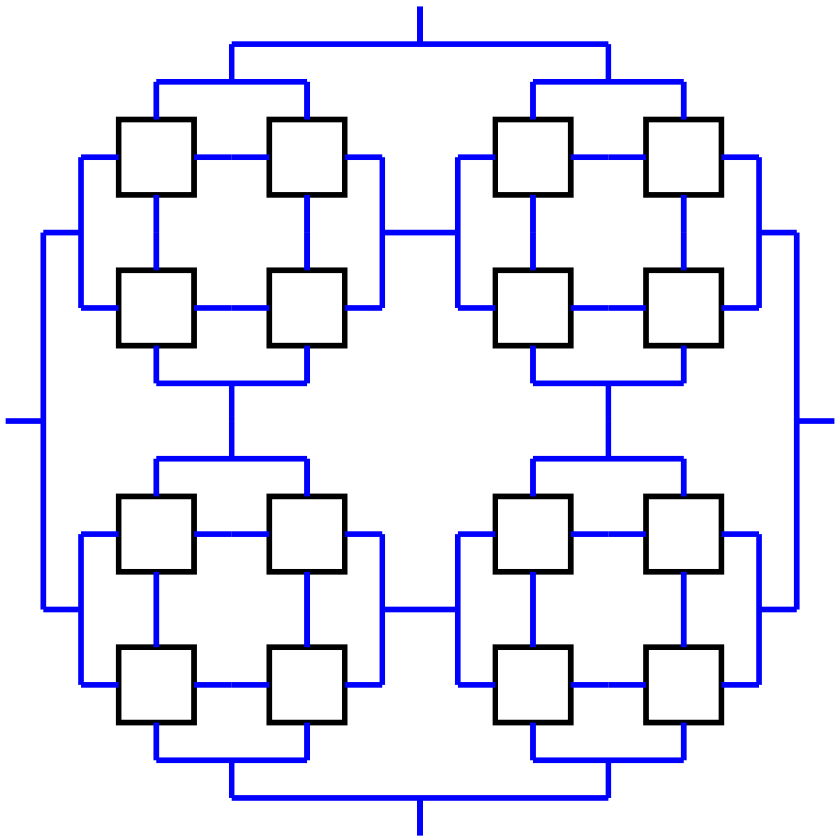




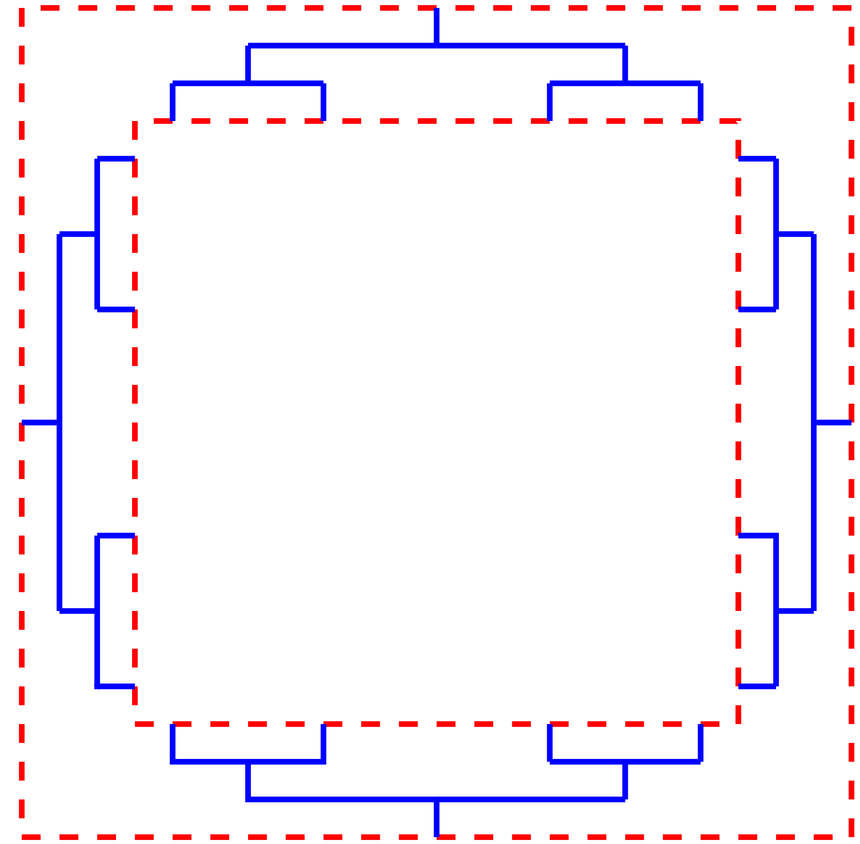
Plateau (scale free dynamics) of almost constant embedding maps around phase transition

Embeddings determined by the dynamics of the system. Represent the physical vacuum for finer degrees of freedom.

The procedure for 2D state sum

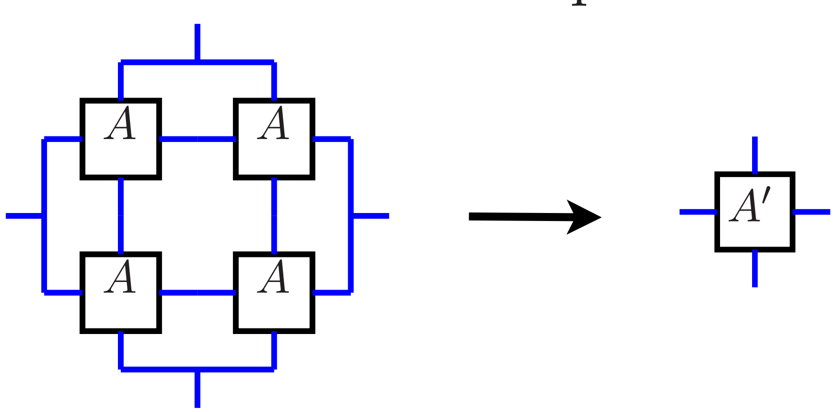


approximation



embedding maps
needed to compare results
for different bond dimensions

iteration step



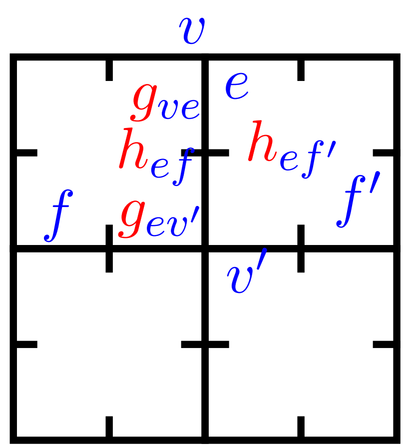
convergence defines continuum limit

Application to spin foams / spin nets

Holonomy formulation

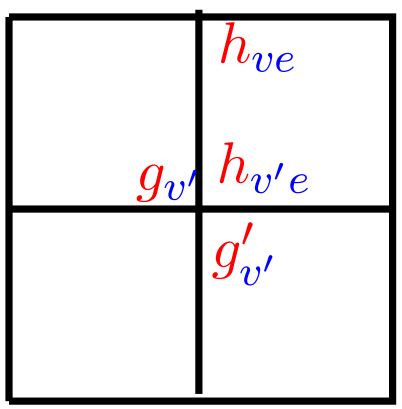
[Bahr, BD, Hellmann, Kaminski 1208.3388]

spin foams



generalizes correspondence between
4D lattice gauge theories and
2D Ising like models

spin nets [BD, Eckert, Martin-Benito '11]



associate

- to every edge e two group elements $g_{ve}, g_{ev'}$
- to every edge–face pair ef a group element h_{ef}

$$Z = \int \prod_{(ef)} dh_{ef} \prod_{(ev)} dg_{ev} \prod_{(ef)} E(h_{ef}) \prod_{f} \delta(g_{ve} h_{ef} g_{ev'} \cdots)$$

simplicity constraints

This parametrization covers BC, EPRL and FK models.

associate

- to every vertex v two group elements g_v, g'_v
- to every vertex–edge pair ve a group element h_{ve}

$$Z = \int \prod_{(ve)} dh_{ve} \prod_{v} dg_v dg'_v \prod_{(ve)} E(h_{ve}) \prod_{e} \delta(g_v h_{ve} g'_v g_{v'} h_{v'e} g'_{v'})$$

simplicity constraints

Integrating out g_v, g'_v gives a tensor network model.

Application to spin nets

- spin nets allow interesting models in 2D (whereas spin foams need higher dimensions)
- experience from lattice gauge theory:
  statistical properties between corresponding foams and nets might be similar
- we can probe the behaviour of simplicity constraints under coarse graining
- in particular by studying embedding maps for transfer operator
- transfer operator incorporates simplicity constraints

$$T = K \cdot W \cdot K$$

[BD, Hellmann, Kaminski 1209.4539]

project onto subspace
determined by simplicity constrains

- as blocking is determined by the dynamics:

➡Is this blocking geometrically meaningful?

➡Are the simplicity constraints relaxed under coarse graining?

Application to spin nets

- [BD, Eckert, Martin-Benito '11] study of Abelian spin net models without non-trivial simplicity constraints
- [BD, Martin-Benito, v. Massenbach wip] study of models based on permutation group S3 with simplicity constraints: simulations are running!
- near future prospect: numerical study of SU2 quantum group models
- use the interplay between embedding maps and truncation for analytical investigations

Stay tuned!

Answers

- How to block finer variables into coarser ones?
- What is the [finite dimensional] space of models, renormalization flow takes place in?
- How to truncate the flow back to this space?
- How to deal with non-local couplings?
- How to coarse grain the boundary?
Should we require triangulation independence for the boundary?

Thanks!