The University of
Nottingham

UNITED KINGDOM · CHINA · MALAYSIA

Weinfurtner, Silke and De las Cuevas, Gemma and Martin-Delgado, Miguel Angel and Briegel, Hans J. (2014) Reducing space-time to binary information. Journal of Physics A: Mathematical and Theoretical, 47 (9). 095301/1-095301/14. ISSN 1751-8121

Access from the University of Nottingham repository:
http://eprints.nottingham.ac.uk/40862/1/2014-01-22-QIG.pdf

Reducing Spacetime to Binary Information

Silke Weinfurtner[1], Gemma De las Cuevas[2,3,4], Miguel Angel Martin-Delgado[5] and Hans J. Briegel[3,4]

[1]*SISSA, Via Bonomea 265, 34136, Trieste, Italy* and *INFN, Sezione di Trieste*
[2]*Max-Planck-Institut für Quantenoptik, Hans-Kopfermann-Str. 1, D-85748, Garching, Germany*
[3]*Institut für Theoretische Physik, Universität Innsbruck, Technikerstraße 25, A-6020 Innsbruck, Austria*
[4]*Institut für Quantenoptik und Quanteninformation der Österreichischen*
Akademie der Wissenschaften, A-6020 Innsbruck, Austria
[5]*Departamento de Física Teórica I, Universidad Complutense, 28040 Madrid, Spain*

We present a new description of discrete space-time in 1+1 dimensions in terms of a set of elementary geometrical units that represent its independent classical degrees of freedom. This is achieved by means of a binary encoding that is ergodic in the class of space-time manifolds respecting coordinate invariance of general relativity. Space-time fluctuations can be represented in a classical lattice gas model whose Boltzmann weights are constructed with the discretized form of the Einstein-Hilbert action. Within this framework, it is possible to compute basic quantities such as the Ricci curvature tensor and the Einstein equations, and to evaluate the path integral of discrete gravity. The description as a lattice gas model also provides a novel way of quantization and, at the same time, to quantum simulation of fluctuating space-time.

A central task in defining a discrete version of space-time is to identify the relevant degrees of freedom. Additionally, any theory of discrete spacetime must take into account coordinate invariance, which is a fundamental property of General Relativity (GR). This renders the identification of fundamental degrees of freedom nontrivial. Solving this problem corresponds to fixing the gauge of the coordinate invariance symmetry of GR at the discrete level.

Here we address this question and present a binary description of space-time in 1+1 dimensions. To do so, we borrow the discretization method of causal dynamical triangulation (CDT) [1]. Following this route leads to a formulation of spacetimes in terms of a statistical mechanical model, which we identify with a lattice gas model [2]. Fluctuations of spacetime are thereby represented by different states of the lattice gas, and the Boltzmann factor for this statistical model is given by the discretized action of general relativity [3].

The central idea of this work is to replace the dynamical geometry of space-time with a fixed physical lattice with binary degrees of freedom at its vertices. This construction allows us to digitalize the geometrical information content of space-times. More precisely, we show how to construct a foliated triangulation T from a bit array λ by using 'forks', and conversely how to construct a bit array λ from a foliated triangulation T. For the latter construction, one first maps T to its dual T^*, T^* to an integer string S, and S to λ, as indicated in the following commuting diagram (the details of which will be explained below).

$$
\begin{array}{ccc}
\text{Triangulation } T & \xleftarrow{\quad\text{forks}\quad} & \text{Bit array } \lambda \\
\downarrow & & \uparrow \\
\text{Dual triangulation } T^* & \xrightarrow{\qquad\qquad} & \text{Integer string } S
\end{array}
\tag{1}
$$

In addition, we will translate the Pachner moves to operations on the integer string encoding. (The Pachner moves are transformations in the set of triangulations which are ergodic, that is, any two triangulations are related by a finite sequence of Pachner moves [5]. See also [6, 7] for related work.)

Finally, we will use our binary encoding to formulate meaningful quantities of discrete gravity in 1+1 dimensions in the natural language of information processing.

In order to establish these results, let us first recall some basic properties of CDT [1]. In CDT the continuous manifold of space-time is approximated by a piecewise linear manifold [8], where the edge lengths of the simplices are assumed to be constant (space-like edges have length l_{edge}^2 and time-like edges $-\alpha l_{\text{edge}}^2$, we shall henceforth assume $\alpha = -1$, which is one of the points in the 'Euclidean sector' that gives rise to an interesting new phase [4]). Additionally, only manifolds that obey a global proper discrete time are considered, i.e. manifolds with a discrete global time foliation. On the simplicial manifold one can define curvature, an action, and other quantities [8].

We shall here focus on the case of 1+1 dimensions, in which the configuration space $\mathcal{T}_t$ is formed by all foliated triangulations T with with t discrete proper-time steps (see Fig. 1). Additionally, we restrict ourselves to simplicial manifolds with a fixed topology, such as $S^1 \times S^1$ (periodic boundary conditions (pbc) in time and space), or $[0, 1] \times S^1$ (open boundary conditions in time and pbc in space). We consider only connected simplicial manifolds; this implies, in particular, that there is at least one simplex per spatial slice. The topology of the simplicial manifold is characterized by the Euler characteristic $\chi := N_0 - N_1 + N_2$, where N_0, N_1, N_2 is the number of vertices, edges and faces of T, respectively. Note that in 2 dimensions χ is related to the curvature via the Gauss–Bonnet theorem.

In this setting, the Pachner moves consist of two op-

erations (called Rule 1 and Rule 2; see Fig. 1). The curvature R_i is evaluated at every vertex i:

$$R_i = \pi \frac{6 - c_i}{c_i} \qquad (2)$$

where c_i is the coordination number (i.e. number of adjacent vertices) of i. The discretized Einstein–Hilbert action takes the form

$$S(T) = \gamma \chi - \kappa N_2, \qquad (3)$$

where γ and κ are coupling constants related to Newton's constant G_N and the effective cosmological constant Λ in the continuum, respectively [1]. The quantization is carried out by means of a path integral formulation of the action (3), which (for a specific topology such as $[0,1] \times S^1$) takes the form

$$Z = \sum_{T \in \mathcal{T}_t([0,1] \times S^1)} \frac{1}{C(T)} e^{-S(T)}. \qquad (4)$$

Here $C(T)$ is the order of the automorphism group of T, and is the remnant of coordinate invariance of GR in the discrete theory.

Considerable progress has been made over the last few years —in terms of analytic in $1 + 1$ [9] and numerical studies in $2 + 1$ [10] and $3 + 1$ [4] dimensions— to investigate the continuum limit of CDT. Interesting connections with other continuum quantum gravity approaches have also been established [11–13]. A connection between foliated triangulations in terms of random walks was established in [14]. Finally, let us mention that other discrete models of quantum gravity have been proposed, e.g. [15, 16]. Although we have motivated above that we may interpret our discretizations of space-time as real at a fundamental level, we also mention that some investigations have found that discrete models of spacetime tend to suffer from non-adiabaticities of the evolution of fields that live on the spacetime, which can lead to more particle production than what seems to be allowed by cosmological observational constraints [17].

I. FROM BIT ARRAYS TO TRIANGULATIONS.

We now show how a bit array λ encodes a foliated triangulation T, as indicated in Diagram (1). The key observation is that every foliated triangulation T is built entirely (except for the boundaries) out of certain building blocks that we call 'forks'. The idea is the following: while, obviously, the building block of a general triangulation is the triangle, the basic unit of a foliated triangulation is a pair of triangles that share a space-like edge. We identify the fork with this unit; more precisely, each fork consists of 1 'center', 3 legs and 2 faces (see little diagram on bottom of Fig. 2(b)). Thus, we can describe a foliated triangulation by 'comparing' it to a reference lattice and specifying what forks are present and what
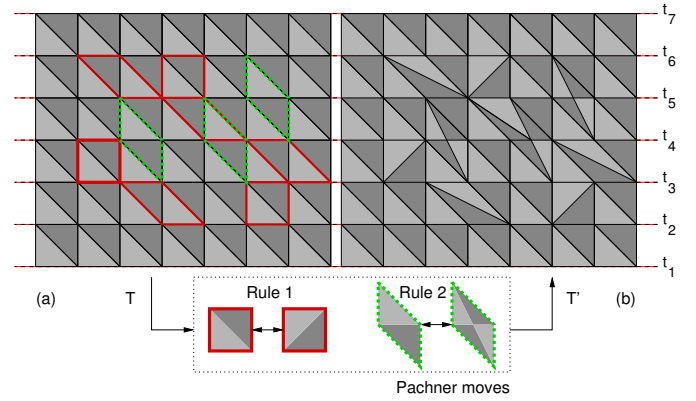

FIG. 1: CDT considers only triangulations that obey a global time foliation. (a) A foliated triangulation corresponding to a flat geometry, since the coordination number of every vertex is 6, excluding the boundaries (see Eq. (2)). Any other foliated triangulation, such as (b), can be obtained by applying a sequence of the Pachner moves Rule 1 and Rule 2.

forks are absent. This renders a description which is in spirit similar to that of a lattice gas model, where the description of a fluid, with molecules absent or present, is mapped to the description of a magnet with two-level spins on a fixed lattice [2].

Let us be more precise. Consider a 2D square lattice where a binary variable λ_{nm} is associated with every vertex (n, m) of the lattice, with $1 \leq n \leq N$ and $1 \leq m \leq M$ (see Fig. 2(a)). This forms a bit array λ. To transform this bit array to a foliated triangulation, we put a fork on every site (n, m) it $\lambda_{nm} = 1$, and no fork if $\lambda_{nm} = 0$. This collection of forks defines a graph $T = (V, E)$ in a natural way: each center of a fork defines a vertex $v \in V$, and its legs become edges $e \in E$ of T. The space-like edge and the time-like edge pointing upwards connect to the first vertex to the left, and the time-like edge pointing downwards connects to the first vertex directly below or to the left (see Fig. 2). This can be formally described as follows. Let p be a rank 3 tensor with components

$$p_{n,m,m'} := \lambda_{nm'} \times [(m - m' \pmod M) + 1] . \qquad (5)$$

Then the vertex at site (n, m) is adjacent to the vertices at sites $(n+1, a)$, (n, b), $(n-1, c)$ where a, b, c are defined by the inequalities

$$\begin{aligned} 0 < p_{n+1,m-1,a} < p_{n+1,m-1,m'} & \quad \forall m' \in [M], \; m' \neq a \\ 0 < p_{n,m-1,b} < p_{n,m-1,m'} & \quad \forall m' \in [M], \; m' \neq b \\ 0 < p_{n-1,m,c} < p_{n-1,m,m'} & \quad \forall m' \in [M], \; m' \neq c, \end{aligned} \qquad (6)$$

where $[M] = \{1, 2, \ldots, M\}$.

II. FROM TRIANGULATIONS TO BIT ARRAYS.

We now show how to map a foliated triangulation to a bit array. The recipe consists of six steps. First, construct the dual graph of T, denoted T^*, whose basic
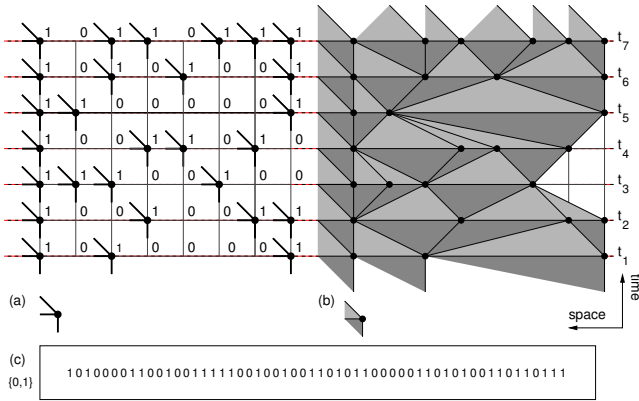
FIG. 2: *From bit arrays to triangulations.* The bit array (a) represents the foliated triangulation (b). Each bit stands for the presence / absence of a fork, which consists of one vertex, one space-like edge and two time-like edges. While the bits are fixed on a two-dimensional grid, their configuration determine the geometry, similar to a lattice gas model. (For illustration issues we have chosen non-periodic boundary conditions in both time and space.) To store the triangulation (b) on a double precision floating-point format (64-bit) machine a single data unit is required (c).

building block is the d-fork (short for dual fork). This consists of one time-like edge and two space-like edges "pointing to the right" (see Fig. 4(a,b)). Second, note that every d-fork crosses one space-like edge of T. We use this fact to attach the label $S_i \in \{1, 2, \ldots, N\}$ to a fork i if it crosses a space-like edge at time t_{S_i} (see Fig. 4(c), where the labels are also represented as colors). Third, record the labels of the d-forks that appear in T^* from left to right and write them down in an integer string $S := (S_1, S_2, \ldots, S_F)$ (cf. Diagram (1)). This string representation has certain symmetries. Two integer strings correspond to the same dual triangulation if they are related by a sequence of the following operations:

(i) commutations of contiguous integers if the integers differ by at least two, i.e. $(\ldots, S_i, S_{i+1}, \ldots) \sim (\ldots, S_{i+1}, S_i, \ldots)$ if $|S_i - S_{i+1}| \geq 2$;

(ii) cyclic permutations, i.e. $(S_1, S_2, \ldots, S_F) \sim (S_F, S_1, S_2, \ldots, S_{F-1})$;

(iii) inversion operation, i.e. $(S_1, S_2, \ldots, S_F) \sim (S_F, S_{F-1}, \ldots, S_1)$.

Notice that (ii) and (iii) are only applicable for periodic boundary conditions on the spatial slices, while (i) is a degeneracy introduced by the integer string encoding.

Fourth, define a string $S^{\text{flat}} := (S^{\text{N}}, S^{\text{N}}, \ldots, S^{\text{N}})$, consisting of complete integer-sequences, $S^{\text{N}} := (1, 2, \ldots, N)$. Apply operation (i) to arrange any S in successive integer-sequences, by allowing for incomplete sequences, e.g. $(1, 3, 4, 6)$ (see Fig. 4(d)). Fifth, define the extended string S^{E} by adding zeros to S wherever there is a missing element, e.g. $(1, 0, 3, 0, 0, 6)$ (see Fig. 4(e)).

Generally,

$$S^{\text{E}} := (\underbrace{0 \ldots 0}_{\Gamma_1}, S_1, \underbrace{0 \ldots 0}_{\Gamma_2}, S_2, \ldots, S_F, \underbrace{0 \ldots 0}_{\Gamma_{F+1}}) \quad (7)$$

where $A := (N, S_1, S_2, \ldots, S_F, 1)$, and Γ is a vector with components $\Gamma_i := A_{i+1} - A_i + N - 1 \pmod{N}$, for $i \in \{1, 2, \ldots, F+1\}$.

Sixth, map S^{E} to λ by arranging the entries of S in a two dimensional array of size $N \times M$ (going from bottom to top and left to right), and then replacing each positive entry by a 1. Formally,

$$\lambda_{ij} = \Theta(S^{\text{E}}_{jN-i+1}) \quad \text{for} \quad \begin{cases} 1 \leq i \leq N \\ 1 \leq j \leq \frac{F^{\text{E}}}{N} \end{cases}, \quad (8)$$

where $\Theta(x) = 0$ if $x = 0$ and $\Theta(x) = 1$ for $x > 0$ (see Fig. 4(f)).

A. Lattice size in binary encoding

The minimum lattice size necessary to encode all triangulations T up to F forks distributed on N spatial slices is given by $N \times \tilde{M}$, where $\tilde{M} = F - N + 1$. To show this, let us temporarily put aside the requirement that there is at least one fork on every spatial slice, equivalent to the requirement that S contains every element in $\{1, 2, \ldots, N\}$ at least once. It is easy to see that $S = \{X, X, \ldots, X\}$, where all elements $X \in \{1, 2, \ldots, N\}$ are the same, maximizes $\Gamma = (N - X, N - 1, N - 1, \ldots, N - 1, X - 1)$, where Γ is a vector of length $F + 1$, such that $F^{\text{E}} = F + (N - 1)F = FN$ and $\tilde{M} = F$. Consequently, by reintroducing the constraint to have at least one fork per spatial slice, the string S that will result in the largest $\tilde{M}$, has as many elements of the same kind, that are $F - N + 1$ d-forks on a single spatial slice X. It is best to study this case using the dual brick picture, where it is straightforward to see that d-forks, $X \pm n$, for $n \geq 1$, can be moved freely in both directions by using operations (i)-(iii) (see Fig. 3). Therefore any of the $N - 1$ d-forks can be absorbed in a sequence that contains already one X, and thus the number of sequences necessary to encode *all* possible triangulations with F forks distributed on N spatial slices is $F - N + 1$. According to the mapping, see Eq. (7) in the main text, every sequence in S^{E} corresponds to one row in the corresponding λ.

Note that by construction the integer encoding represents a coordinate-free encoding of triangulations. Consequently, operations (i)-(iii) introduce an equivalence class on the space of all integer strings, and there exists a straightforward algorithm to single out its representatives. This can be illustrated with the help of a simple example, evaluating all unique histories for triangulations with 3 spatial slices from 3 to 5 forks.

Starting from all possible strings, one can apply operations (i)-(iii) to single out one representative $\tilde{S}$ of every equivalence class: $(1, 2, 3)$, $(1, 2, 3, 3)$, $(1, 2, 2, 3)$, $(1, 2, 3, 2)$, $(1, 1, 2, 3)$, $(1, 2, 3, 3, 3)$,
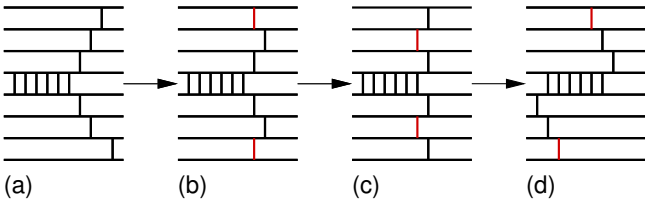
FIG. 3: Four different dual triangulating are represented (a-d). It is shown, that by successive application of operations (i)-(iii) of the main article acting on (a) one obtains (d). The the d-forks colored in red are moved with respect to the previous dual triangulations.

$(1, 2, 2, 3, 3)$, $(1, 2, 3, 2, 3)$, $(1, 2, 3, 3, 2)$, $(1, 2, 2, 2, 3)$, $(1, 2, 2, 3, 2)$, $(1, 2, 3, 2, 3)$, $(1, 1, 2, 3, 3)$, $(1, 1, 2, 2, 3)$, $(1, 1, 2, 3, 2)$, $(1, 2, 1, 2, 3)$, and $(1, 1, 1, 2, 3)$. See appendix A for a more thorough discussion.

III. APPLICATIONS OF BINARY AND INTEGER ENCODING.

We now rephrase various quantities of CDT in binary / integer language. Last but not least, we compare the degeneracies in the configuration space in the binary with the integer encoding.

A. Binary equivalent of Ricci scalar

Consider first the Ricci scalar (Eq. (2)). The coordination number of a vertex at site (n, m) is 3 (because each fork has 3 edges) plus additional Δ_{nm} edges which depend on the surrounding bit array; explicitly:

$$\Delta_{nm} = \sum_{j=1}^{F(d-m)} [\lambda_{n+1,F(m+j-1)} + \lambda_{n+1,F(m+j)} + \lambda_{n+1,F(m+j)}] \quad (9)$$

where $F(m') := m' \pmod{M}$, and d is the index of the first non-zero entry to the left of m given by $0 < p(n, m+1, d) < p(n, m+1, m') \; \forall m' \neq d$ (see Eqs. (5), (6)). The Ricci scalar R_{nm} is then given by

$$R_{nm} = \pi \lambda_{nm} \frac{3 - \Delta_{nm}}{3 + \Delta_{nm}}. \quad (10)$$

The action (Eq. (3)) is also easily computed in the binary encoding. To compute the Euler characteristic, note that every fork is associated to one vertex, 3 edges and 2 faces, hence it does not contribute to the value of χ. Thus only forks relative to the boundary (that is, those which are placed at the boundary, either in the space or time dimension) contribute to χ. For instance, for a torus (topology $S^1 \times S^1$), $\chi = 0$. The volume is then given by twice the number of forks, $N_2 = 2F$. This, together with $C(T)$, which can be evaluated exactly for

1+1 dimensional triangulations [18], suffices to evaluate the discretized path integral (Eq. (4)).

Finally, the Pachner moves can also be expressed in the integer encoding. This allows us to show that the Pachner moves in 1+1 dimensions are ergodic, i.e they generate all simplicial triangulations.

B. Integer equivalent of Pachner moves

The Pachner moves are certain operations that when applied on a triangulation T generate other triangulations T'. They are claimed to be ergodic, i.e. for any pair of triangulations, there always exists a sequence of Pachner moves that one can apply on one triangulation that yield the other. For foliated triangulations in 1+1 dimensions there are only two such moves (called Rule 1 and Rule 2, see Fig. 1 of the main text). Here we define the Pachner moves in the integer string description, that is, transformations on the integer strings which correspond to applying the Pachner move on the corresponding triangulation, as the following diagram illustrates.

$$
\begin{array}{ccccc}
T & \longrightarrow & T^* & \longrightarrow & S \\
\downarrow{\text{Pachner}} & & & & \downarrow{\text{"integer Pachner"}} \\
T' & \longrightarrow & (T')^* & \longrightarrow & S'
\end{array}
$$
$$(11)$$

The action of Rule 1 (R1) is easily understood in the dual triangulation, where it simply exchanges the order of two neighboring consecutive d-forks. In the integer description it corresponds to the following operation

$$(\ldots, S_i, S_{i+1}, \ldots) \xrightarrow{\text{R1}} (\ldots, S_{i+1}, S_i, \ldots). \quad (12)$$

where $|S_i - S_{i+1}| = 1$. Rule 2 (R2) has an equally simple equivalent,

$$
\begin{aligned}
(\; \ldots, (n-1), n, (n+1) \ldots) &\xrightarrow{\text{R2}} \\
(\; \ldots, (n-1), n, n, (n+1), \ldots),
\end{aligned}
\quad (13)
$$

duplicating a d-fork and placing it right next to it. It is thus clear that R1 and R2 connect different equivalence classes in the string encoding. This demonstrates the ergodicity of the Pachner moves in 1+1 dimensions: starting from the minimal string $S = (1, 2, 3, \ldots, N-2, N-1, N)$, successive application of R1 and R2 generate all possible strings, and hence all possible triangulations. Fig. 5 shows the action of the "integer Pachner moves" on triangulations mentioned in the main text.

C. Degeneracies of the various encodings

The binary and the integer encoding differ in the size of their configuration spaces. To show this, let us for a moment relax the constraint of only considering connected triangulations to get a rough estimate of the size
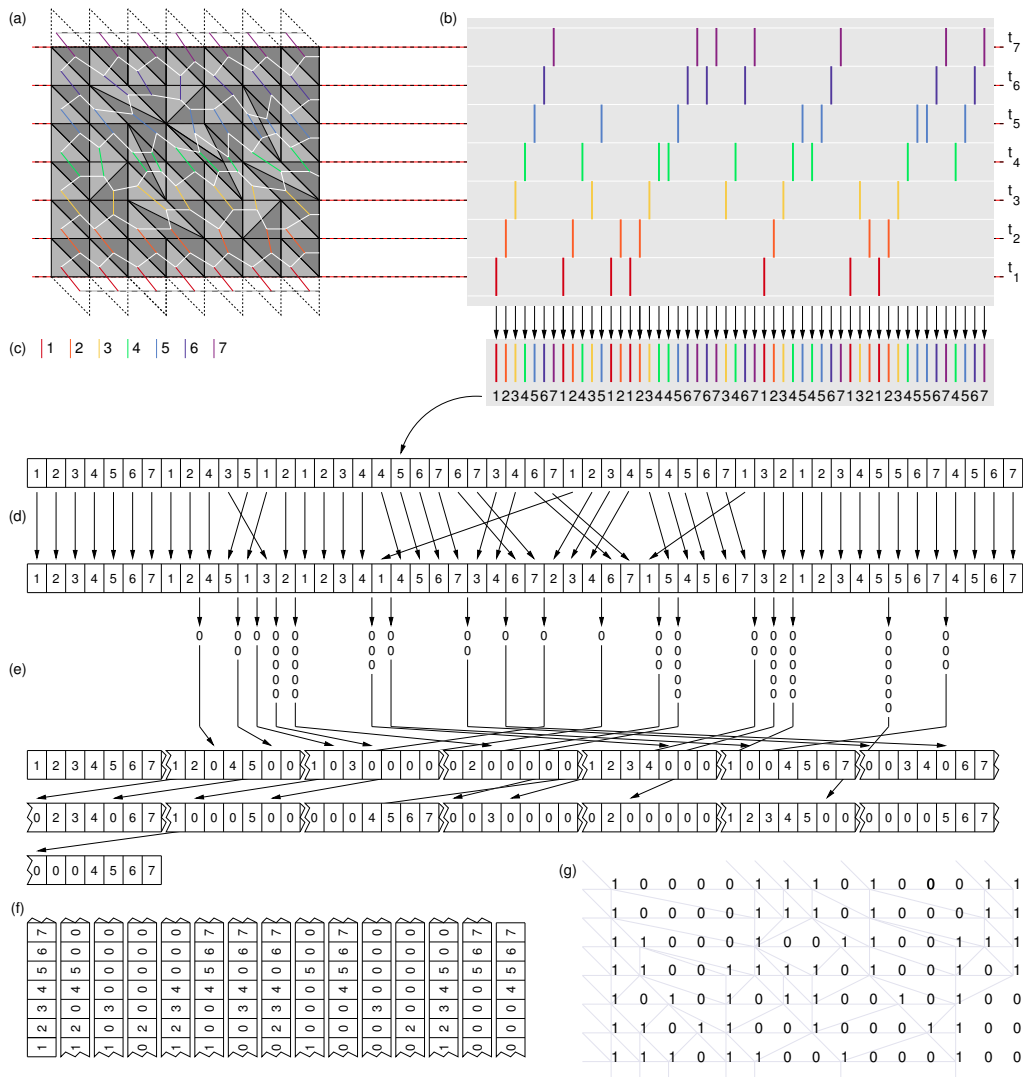
FIG. 4: *From triangulations to bit arrays.* The triangulation T of panel (a) (which is the same as in Fig. 1(b)) is mapped to the bit array of panel (g). (a) We first construct the dual of T (a), and redraw it to give it the appearance of a brick wall (b), where every vertical (colored) line represents a dual fork (d-fork). The dual graph is mapped to a string S (c). Using operations (i)-(iii) we reorder S in successive incomplete integer-sequences (d), and extend the string by adding zeros where the integer-sequence is incomplete (e). Finally, we map this extended string to the bit array with Eq. (8) (f,g). (Notice that we consider here open boundary conditions on the spatial slices.)

of the configuration space. For $F \gg N$ we can simplify the minimum lattice size required to encode all triangulations to $N \times F$. Thus the number of triangulations in the binary encoding is growing $\propto (2^N)^F$, while the size of the integer encoding grows $\propto N^F$. Since both encodings are ergodic (in the sense of containing all triangulations with F forks distributed over N spatial slices at least once) the binary encoding is less efficient than the integer encoding. The integer encoding has the additional advantage that the degeneracies of the configuration space are related to algebraic symmetry operations acting on the strings (see above).

This can be demonstrated with the help of the following example. Consider the integer string $S = (1, 2, 3, \ 1, 2, 3, \ 1, 2, 3)$, representing a triangu-

lation with coordination number 6 everywhere except at the boundaries. Even for this trivial example the mapping to a binary encoding is not unique, if the lattice size is larger then 3×3. In Fig. 6 we have depicted 6 seemingly different encodings of S. Fig. 6(a-d) are simply different binary encodings of S, while Fig. 6(e-f) are mappings of S to λ after having applied various symmetry operations on S.

The differences in the configuration space, if not taken into account when taking the continuum limit, may lead to a different structure of the phases. We are currently investigating this issue further.
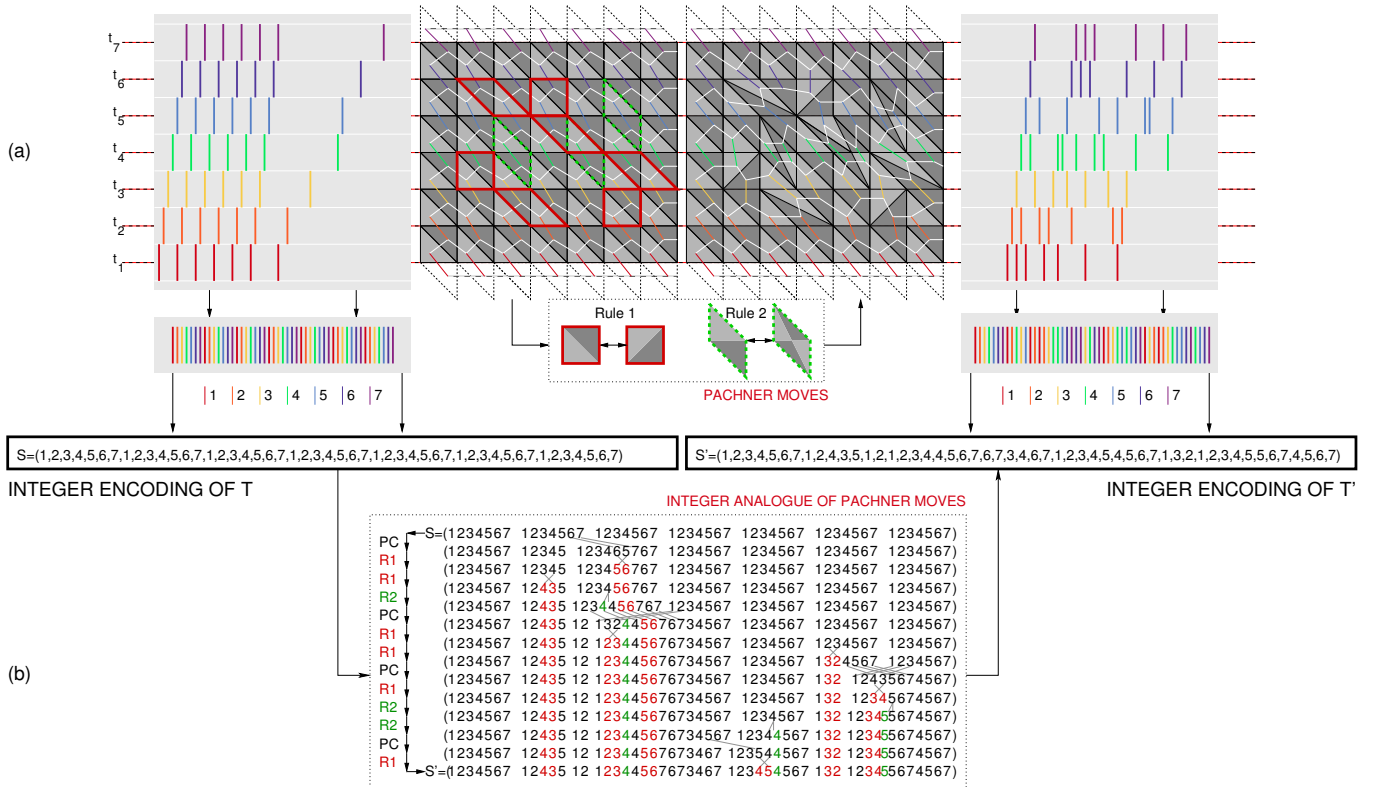
FIG. 5: We show explicitly how to apply successively R1 and R2 on T to obtain T' in both, the simplicial (a) and the integer string (b) encoding. In order to apply R1 it is sometimes necessary to use the pairwise commutation relations (PC), i.e. $(\ldots, S_i, S_{i+1}, \ldots) \sim (\ldots, S_{i+1}, S_i, \ldots)$ if $|S_i - S_{i+1}| \geq 2$. Notice that the integer strings S and S' correspond to the triangulations T and T' of Fig. 1 in the main text, respectively.
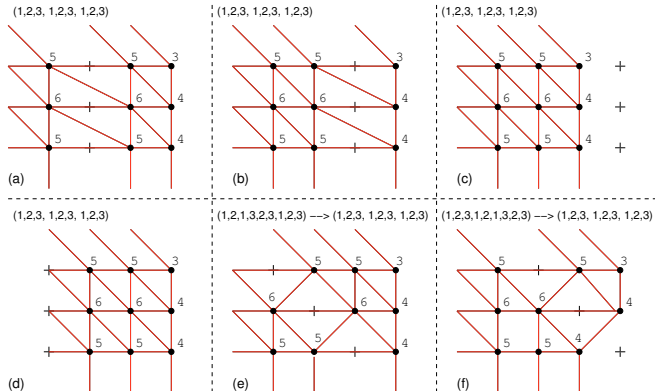

FIG. 6: Six different binary encodings of $\bar{S} = (1, 2, 3,\ 1, 2, 3,\ 1, 2, 3)$ in a 3×4 lattice are represented. The numbers at the vertices are the coordination numbers.

IV. CONCLUSION AND OUTLOOK

We have presented a binary encoding of discrete space-time in 1+1 dimensions. To this end, we have borrowed the discretization method of CDT and shown how to compute various quantities of interest of this theory in the binary description. Our results enable us to express

a classical theory of discrete space-time as a lattice gas model. This approach has several potential applications. To begin with, it provides a natural framework for quantization. Forks, the elementary geometrical unit, constitute independent classical degrees of freedom or 'normal modes' of discrete space-time. In the quantized theory, the presence or absence of a fork will be interpreted as elementary excitations of a quantum lattice gas. These excitations will give rise to quantum fluctuations and, more generally, to quantum states representing superpositions of different space-times. Despite using a time foliation, our approach to quantizing space-time is conceptually different from CDT quantum gravity, which is based on a path integral formulation.

Discretization of space-time was introduced as a regularization tool to compute the path integral of general relativity. If one considers discrete space-time as *real*, then forks could be seen as fundamental constituents of space-time, namely as basic geometric structures that connect events. A quantum formulation of this theory would naturally introduce quantum correlations in space-time.

As for the extension of our formalism to higher dimensions, like 2+1 or 3+1, we point out that the fork formalism that we have worked out in 1+1 dimensions, needs additional work and insight in order to apply it to higher dimensions. This can be seen from several facts,

in particular, in higher dimensions we have additional geometrical constraints besides the Euler characteristic and we also have additional Pachner moves. Therefore, this important line of research calls for more elaboration and future work.

Finally, from an experimental perspective, by implementing quantum lattice gas models in the way we have introduced them, one could realize quantum simulations of fluctuating space-time. Such quantum simulators, analog or digital, could be used to simulate models of quantum gravity in modern quantum optics laboratories, e.g. using ultra-cold atoms in optical lattices or ion-traps [19].

Appendix A: Representatives of equivalence classes in all three encodings

To evaluate the path integral of CDT in any dimensions analytically, one has to first single out triangulations which are representatives of their equivalence classes. We demonstrate how this procedure can be done in integer string encoding. First, note that to encode all triangulations of size $V = 2F$ distributed over N spatial slices, one has to generate all strings containing at least N forks and at most F of them containing each element of $\{1 \ldots N\}$ at least once. We then apply symmetry operations (i)-(iii) as illustrated in the following example.

Consider all triangulations containing up to 5 forks distributed over 3 spatial slices. We require at least one fork on every spatial slice; the smallest (in terms of 2-volume) triangulations contain 3 forks. The procedure to find the representatives for the equivalent classes is the following:

1. generate all possible integer strings of lengths 3, 4 and 5, with elements in $S_i \in \{1, 2, 3\}$;

2. consider only strings that contain every element $\{1, 2, 3\}$ at least once (so that only connected triangulations are taken into account);

3. apply symmetry operations to single out unique triangulations / integer strings: if two integer strings are related by a sequence of the symmetry operations (i–ii) they are in the same equivalence class. One representative of each equivalence class give all unique histories / triangulations. It is best to demonstrate this at hand of an example. What are the representatives of triangulations containing 4 forks distributed over 3 spatial slices? From basic combinatorics we are left with nine valid integer string encodings: $S_1 = (1, 2, 3, 3)$, $S_2 = (1, 3, 2, 3)$, $S_3 = (1, 3, 3, 2)$, $S_4 = (1, 2, 2, 3)$, $S_5 = (1, 2, 3, 2)$, $S_6 = (1, 3, 2, 2)$, $S_7 = (1, 1, 2, 3)$, $S_8 = (1, 1, 3, 2)$, and $S_9 = (1, 2, 1, 3)$. Let us to begin with focus on the dual triangulations with one extra d-fork on the third row, i.e. S_1, S_2 and S_3. Utilizing the symmetry operations we can show that $S_2 = (1, 3, 2, 3) = (3, 1, 2, 3) = (1, 2, 3, 3) = S_1$, and $S_3 = (1, 3, 3, 2) = (3, 1, 3, 2) = (3, 3, 1, 2) = (3, 1, 2, 3) = (1, 2, 3, 3) = S_1$, and consequently S_2 and S_3 are redundant, and S_1 is a representative. Similarly we get $S_6 = (1, 3, 2, 2) = (3, 1, 2, 2) = (1, 2, 2, 3) = S_4$, $S_8 = (1, 1, 3, 2) = (1, 3, 1, 2) = (3, 1, 1, 2) = (1, 1, 2, 3) = S_7$, and $S_9 = (1, 2, 3, 1) = (1, 1, 2, 3) = S_7$. Altogether we can for example single out the following representatives: S_1, S_4, S_5, and S_7.

Next we present the binary and the integer description for this set of triangulations explicitly. For the binary description we need to consider arrays of size $N \times \tilde{F} - N + 1 = 3 \times 3$ (see section II in the main text).

1. Representatives for 3 forks distributed over 3 spatial slices

There is only one way to distribute 3 forks on 3 spatial slices, such that

$$\tilde{S}_1 = (1, 2, 3) \quad \rightarrow \quad \tilde{S}_1^{\mathrm{E}} = (1, 2, 3,\ 0, 0, 0,\ 0, 0, 0) \quad \rightarrow \quad \lambda(\tilde{S}_1^{\mathrm{E}}) = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad \text{see FIG. 7(1)} \tag{A1}$$
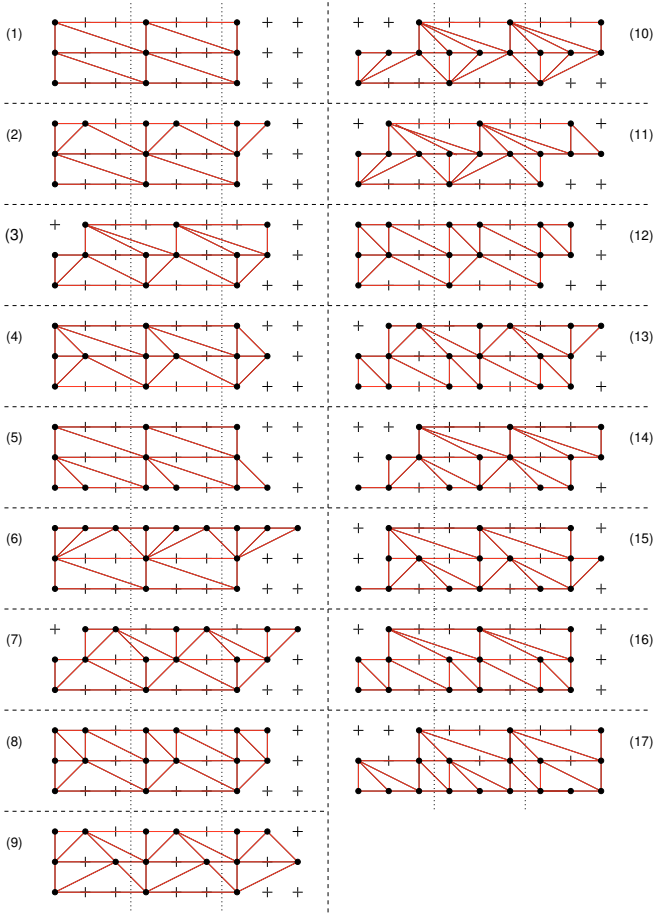
FIG. 7: The representatives of the equivalence classes for all triangulations up to 5 forks distributed on 3 spatial slices are shown. Notice that because of the periodic boundary conditions on the spatial slices, it is advantageous to display three periods (indicated by the dotted vertical lines).

2. Representatives for 4 forks distributed over 3 spatial slices

After applying the symmetry operations we are left with four representatives:

$$\tilde{S}_2 = (1,2,3,3) \quad \rightarrow \tilde{S}_2^{\mathrm{E}} = (1,2,3,\ 0,0,3,\ 0,0,0) \quad \rightarrow \lambda(\tilde{S}_2^{\mathrm{E}}) = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad \text{see FIG. } 7(2); \tag{A2}$$

$$\tilde{S}_3 = (1,2,2,3) \quad \rightarrow \tilde{S}_3^{\mathrm{E}} = (1,2,0,\ 0,2,3,\ 0,0,0) \quad \rightarrow \lambda(\tilde{S}_3^{\mathrm{E}}) = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad \text{see FIG. } 7(3); \tag{A3}$$

$$\tilde{S}_4 = (1,2,3,2) \quad \rightarrow \tilde{S}_4^{\mathrm{E}} = (1,2,3,\ 0,2,0,\ 0,0,0) \quad \rightarrow \lambda(\tilde{S}_4^{\mathrm{E}}) = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad \text{see FIG. } 7(4) \tag{A4}$$

$$\tilde{S}_5 = (1,1,2,3) \quad \rightarrow \tilde{S}_5^{\mathrm{E}} = (1,2,3,\ 1,0,0,\ 0,0,0) \quad \rightarrow \lambda(\tilde{S}_5^{\mathrm{E}}) = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}, \quad \text{see FIG. } 7(5). \tag{A5}$$

3.   Representatives for 5 forks distributed over 3 spatial slices

After applying the symmetry operations we are left with 12 representatives:

$$\tilde{S}_6 = (1,2,3,3,3) \quad \rightarrow \quad \tilde{S}_6^{\mathrm{E}} = (1,2,3,\ 0,0,3,\ 0,0,3) \quad \rightarrow \quad \lambda(\tilde{S}_2^{\mathrm{E}}) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad \text{see FIG. 7(6);} \tag{A6}$$

$$\tilde{S}_7 = (1,2,2,3,3) \quad \rightarrow \quad \tilde{S}_7^{\mathrm{E}} = (1,2,0,\ 0,2,3,\ 0,0,3) \quad \rightarrow \quad \lambda(\tilde{S}_7^{\mathrm{E}}) = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad \text{see FIG. 7(7);} \tag{A7}$$

$$\tilde{S}_8 = (1,2,3,2,3) \quad \rightarrow \quad \tilde{S}_8^{\mathrm{E}} = (1,2,3,\ 0,2,3,\ 0,0,0) \quad \rightarrow \quad \lambda(\tilde{S}_8^{\mathrm{E}}) = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad \text{see FIG. 7(8);} \tag{A8}$$

$$\tilde{S}_9 = (1,2,3,3,2) \quad \rightarrow \quad \tilde{S}_9^{\mathrm{E}} = (1,2,3,\ 0,0,3,\ 0,2,0) \quad \rightarrow \quad \lambda(\tilde{S}_9^{\mathrm{E}}) = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \quad \text{see FIG. 7(9);} \tag{A9}$$

$$\tilde{S}_{10} = (1,2,2,2,3) \quad \rightarrow \quad \tilde{S}_{10}^{\mathrm{E}} = (1,2,0,\ 0,2,0,\ 0,2,3) \quad \rightarrow \quad \lambda(\tilde{S}_{10}^{\mathrm{E}}) = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \quad \text{see FIG. 7(10);} \tag{A10}$$

$$\tilde{S}_{11} = (1,2,2,3,2) \quad \rightarrow \quad \tilde{S}_{11}^{\mathrm{E}} = (1,2,0,\ 0,2,3,\ 0,2,0) \quad \rightarrow \quad \lambda(\tilde{S}_2^{\mathrm{E}}) = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \quad \text{see FIG. 7(11);} \tag{A11}$$

$$\tilde{S}_{12} = (1,2,3,2,3) \quad \rightarrow \quad \tilde{S}_{12}^{\mathrm{E}} = (1,2,3,\ 0,2,3,\ 0,0,0) \quad \rightarrow \quad \lambda(\tilde{S}_{12}^{\mathrm{E}}) = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad \text{see FIG. 7(12);} \tag{A12}$$

$$\tilde{S}_{13} = (1,1,2,3,3) \quad \rightarrow \quad \tilde{S}_{13}^{\mathrm{E}} = (1,0,0,\ 1,2,3,\ 0,0,3) \quad \rightarrow \quad \lambda(\tilde{S}_{13}^{\mathrm{E}}) = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{pmatrix}, \quad \text{see FIG. 7(13);} \tag{A13}$$

$$\tilde{S}_{14} = (1,1,2,2,3) \quad \rightarrow \quad \tilde{S}_{14}^{\mathrm{E}} = (1,0,0,\ 1,2,0,\ 0,2,3) \quad \rightarrow \quad \lambda(\tilde{S}_{14}^{\mathrm{E}}) = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix}, \quad \text{see FIG. 7(14);} \tag{A14}$$

$$\tilde{S}_{15} = (1,1,2,3,2) \quad \rightarrow \quad \tilde{S}_{15}^{\mathrm{E}} = (1,0,0,\ 1,2,3,\ 0,2,0) \quad \rightarrow \quad \lambda(\tilde{S}_{15}^{\mathrm{E}}) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix}, \quad \text{see FIG. 7(15);} \tag{A15}$$

$$\tilde{S}_{16} = (1,2,1,2,3) \quad \rightarrow \quad \tilde{S}_{16}^{\mathrm{E}} = (1,2,0,\ 1,2,3,\ 0,0,0) \quad \rightarrow \quad \lambda(\tilde{S}_{16}^{\mathrm{E}}) = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{pmatrix}, \quad \text{see FIG. 7(16);} \tag{A16}$$

$$\tilde{S}_{17} = (1,1,1,2,3) \quad \rightarrow \quad \tilde{S}_{17}^{\mathrm{E}} = (1,0,0,\ 1,0,0,\ 1,2,3) \quad \rightarrow \quad \lambda(\tilde{S}_{17}^{\mathrm{E}}) = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \quad \text{see FIG. 7(17);} \tag{A17}$$

[1] J. Ambjørn, J. Jurkiewicz, and R. Loll, Phys. Rev. Lett. **85**, 924 (2000); *ibid.*, Phys. Rev. D **64** 044011 (2001).

[2] H. E. Stanley, *Introduction to phase transitions and crit-*

ical phenomena, Oxford University Press (1987).

[3] T. Regge, General relativity without coordinates, *Nuovo Cim.*, 19:558–571, 1961.

[4] J. Ambjørn, J. Jurkiewicz, and R. Loll, Phys. Rev. Lett. **93**,131301 (2004).

[5] U. Pachner, European Journal of Combinatorics, **12**, 129 (1991).

[6] A. Kempf, Phys. Rev. Lett. 103, 231301 (2009).

[7] A. Kempf, arxiv:1302.3680 (2013).

[8] F. David, Simplicial quantum gravity and random lattices (Elsevier, Amsterdam, 1995), Session LVII, p. 679; *ibid.* Nucl. Phys. B **257**, 45 (1985); V. Kazakov, Phys. Lett. B **150**, 282 (1985).

[9] J. Ambjørn and R. Loll. Nucl. Phys. B **536**, 407 (1998).

[10] D. Benedetti, R. Loll, and F. Zamponi, Phys. Rev. D **76**, 104022 (2007).

[11] P. Hořava, Phys. Rev. Lett. **102**, 161301 (2009).

[12] T. P. Sotiriou, M. Visser, and S. Weinfurtner, Phys. Rev. Lett. **107**, 131303 (2011).

[13] M. Reuter and F. Saueressig, JHEP **1112**, 012 (2011).

[14] P. Di Francesco, E. Guitter, C. Kristjansen, Nucl. Phys. B **567**, 515 (2000).

[15] L. Bombelli, J. Lee, D. Meyer and R. Sorkin, Phys. Rev. Lett. **59**, 521 (1987).

[16] T. Konopka, F. Markopolou and S. Severini, Phys. Rev. D **77**, 104029 (2008).

[17] B. Z. Foster, Th. Jacobson, JHEP 0408 (2004) 024

[18] B. Dittrich and R. Loll, Class. Quantum Grav. **23**, 3849 (2006).

[19] H. Weimer, M. Müller, I. Lesanovsky, P. Zoller, H. P. Büchler, Nat. Phys. **6**, 382 (2010); I. Bloch, J. Dalibard and S. Nascimbène, *ibid.* **8**, 267 (2012); R. Blatt and Ch. Roos, *ibid.* **8**, 277 (2012).