Globally and locally causal dynamical triangulations

Jordan, S.

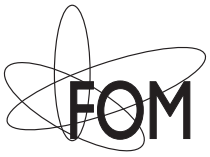
2013, Dissertation

Globally and Locally Causal Dynamical Triangulations

Cover: An illustration of a quantum universe as a superposition of curved spacetimes.

Printed by Ipskamp Drukkers BV
ISBN: 978-94-6191-787-4

This work is part of the research programme of the Foundation for Fundamental Research on Matter (FOM), which is part of the Netherlands Organisation for Scientific Research (NWO).



This work was sponsored by NWO Exacte Wetenschappen (Physical Sciences) for the use of supercomputer facilities, with financial support from the Nederlandse Organisatie voor Wetenschappelijk Onderzoek (Netherlands Organization for Scientific Research, NWO).

Globally and Locally Causal Dynamical Triangulations

Proefschrift

ter verkrijging van de graad van doctor
aan de Radboud Universiteit Nijmegen
op gezag van de rector magnificus
prof. mr. S.C.J.J. Kortmann,
volgens besluit van het college van decanen
in het openbaar te verdedigen
op woensdag 16 oktober 2013
om 12.30 uur precies

door

Samo Jordan

geboren op 6 maart 1975
te Chur, Zwitserland

Promotor: Prof. dr. R. Loll

Manuscriptcommissie:
Prof. dr. R.H.P. Kleiss
Prof. dr. J. Ambjørn (University of Copenhagen)
Prof. dr. G.T. Barkema (Universiteit Utrecht)
Prof. dr. J. Smit (Universiteit van Amsterdam)
Dr. F. Saueressig

Contents

1 Introduction

The beginning of the twentieth century brought a paradigm shift in two major areas in physics. On the one hand it was discovered that physics on very small length scales is governed by laws which were completely new and unanticipated at that time. The famous double-slit experiment revealed that physical objects sometimes behave like a particle and sometimes like a wave, exhibiting properties like superposition and interference. More disturbing was the discovery that the outcome of individual experiments could no longer be predicted. These newly found laws of *quantum physics* only allowed to make predictions of probabilistic nature.

The second revolution changed our understanding of physics at very large length scales. This is the domain ruled by the laws of gravitation which since Newton were understood to be force laws, similar to the laws which govern the physics of classical electrostatics. When Einstein published the theory of special relativity in 1905 he quickly realized that the upper limit for velocities given by the speed of light was in contradiction with the Newtonian laws of gravity. About ten years later he presented the theory of *general relativity* based on the revolutionary new idea that spacetime is curved due to the presence of masses.

Both quantum physics and general relativity have developed into mature and well-tested theories over time. The former has evolved into a set of quantum field theories which describe physics at small length scales and at high velocities where the laws of special relativity become important. They form the backbone of the Standard Model of particle physics which describes three of the four fundamental forces in nature, namely, the electromagnetic, the weak and the strong nuclear forces. The Standard Model has been verified in countless experiments with high precision, including the recent discovery of the Higgs boson [1].

The first experimental verification of general relativity happened in 1919 during a solar eclipse by observing the bending of light as predicted by the theory. Since then a large number of tests have been performed and none of them has shown to be incompatible with general relativity [2]. The theory also predicts the presence of gravitational waves for which so far only indirect evidence exists, based on the study of binary pulsars [3, 4]. The sensitivity of gravitational wave detectors continues to increase and there is

confidence that a direct detection of gravitational waves will take place in the near future [5].

The gravitational force is the weakest of the four fundamental forces and its dominance at large scales is a consequence of its long-range character and its unipolar nature as compared to the bipolar electromagnetic force. The relative weakness of gravity has the consequence that it does not contribute to scattering experiments which take place at microscopic scales in the quantum regime. The length scale at which the strength of the gravitational force becomes comparable to the other forces is governed by the Planck length $\ell_P = 1.62 \times 10^{-35} m$ which is currently completely inaccessible by experiments.

Although the combination of gravity with quantum physics is not obviously needed to explain current experiments, it is nevertheless desirable to understand the physics, which involves both ingredients, in terms of a theory of *quantum gravity*. Initially, the hope was that the highly successful quantizations of the other fundamental forces could be adapted to the case of gravity in a straightforward way. These first attempts quickly faced the problem that gravity behaves differently under renormalization, the standard procedure in any quantum field theory to describe the change of physics as a function of the energy scale. More precisely, it turned out that in order to predict the outcome of a quantum gravitational experiment at arbitrary energies, one needed first to make an infinite number of measurements and then to feed these results into the measurement procedure [6, 7]. This so-called *non-renormalizable* theory was thus not predictive at all.

These problems led to a dismissal of the standard quantum field theoretic approach of gravity and it was believed that there must be some radically new input. String theory was developed based on the idea that the fundamental constituents of nature are one-dimensional objects (see [8, 9, 10, 11] for textbooks). Another popular approach is loop quantum gravity which attempts to perform a canonical quantization using special variables based on tetrads and holonomies [12, 13]. So far all proposals for a theory of quantum gravity are facing significant problems, thus the problem of reconciling general relativity with quantum field theory is still far from being solved.

In 1979 Weinberg suggested that gravity could be renormalizable in a nonperturbative sense [14]. More precisely, if there existed an ultraviolet fixed point of the renormalization group with a finite number of relevant directions, gravity could be described at all scales by a quantum field theory and still be predictive. This so-called *asymptotic safety* scenario has spawned a line of research, with the goal of finding UV fixed points in theories of quantum gravity. The existence of such a UV fixed point is suggested by a $(2+\varepsilon)$-expansion in the dimensionality of spacetime, starting from the observation that two-dimensional gravity is formally a renormalizable theory [15, 16, 17]. With the advent of functional renormalization techniques, the search for UV fixed points could be extended to gravity in four dimensions [18] and numerous results from various truncated versions of the full theory provide evidence that the asymptotic safety scenario may indeed be realized [19, 20, 21, 22, 23].

Independent evidence for the asymptotic safety of gravity, if it is realized, could come from a nonperturbative path integral quantization of gravity. There the key concepts of quantum physics and general relativity, *superposition* and *curved spacetimes*, are combined

in a straightforward way by studying *superpositions of curved spacetimes*. This line of research has been pursued in various lattice approaches of quantum gravity, for example in Regge calculus, dynamical triangulations and spin foam models [24, 25, 26].

Regge calculus and dynamical triangulations implement the path integral of gravity as a sum over piecewise linear geometries which are composed of elementary simplicial building blocks. We are focusing here on dynamical triangulations (DT), where the geometrical information is encoded in the way how the building blocks are glued together. In the original formulation of DT the elementary building blocks were taken to be flat simplices with Euclidean signature and the action was taken to be the Regge action, a straightforward discretization of the Euclidean Einstein-Hilbert action that was originally introduced in [27].

Numerical studies of the phase structure of four-dimensional DT showed the presence of two phases of geometry, a crumpled phase with infinite Hausdorff dimension and a polymeric phase with a Hausdorff dimension of two [28, 29, 30]. The phase transition was first believed to be of higher order, but later simulations with larger systems clearly revealed the first-order nature of the transition [31, 32]. Since the presence of a critical point is deemed necessary in order to define a continuum limit, the researchers gradually lost interest in the original formulation of DT.

To overcome these problems, a modification of DT was suggested in 1998 based on the idea that one should sum over *Lorentzian* geometries [33]. In this new model called *Causal Dynamical Triangulations* (CDT) all piecewise linear geometries are composed of Lorentzian building blocks and causality is enforced kinematically by the presence of a distinguished foliation and the requirement that the spatial leaves of this foliation all have the same topology. The numerical investigation of the phase structure in 3+1 dimensions revealed remarkable differences to the original DT model. A new phase was found containing extended geometries with Hausdorff dimension four and with a scale-dependent spectral dimension flowing from four at large scales to approximately two at small scales [34, 35]. The latter effect has since been observed in various approaches of quantum gravity (see [36] and references therein).

The emergence of a four-dimensional universe motivated a detailed analysis of the effective action for the scale factor of the universe. It was found that the effective action could be matched very well to a minisuperspace action describing a de Sitter spacetime with linearized quantum fluctuations [37]. This result is remarkable since no background geometry has ever been put in by hand and CDT thus passes a nontrivial consistency check with respect to its infrared behavior.

Ultimately a theory of quantum gravity should not only be shown to have the correct low-energy limit, but it should also provide new insights into ultraviolet physics. A pressing UV-related question with respect to CDT is whether it is possible to define a continuum limit in the standard sense of regularized quantum field theories. In order to address this question, the phase transitions need to be analyzed and their order has to be determined. Chapter 3 in this thesis is devoted to the study of the phase transitions in CDT in 3+1 dimensions. We present results which strongly suggest that one of the phase transitions is second order and thus provides natural candidate fixed points to define a

continuum limit.

The presence of a distinguished foliation, a key ingredient of CDT, is also a characteristic feature of a new class of theories of gravity called *Hořava-Lifshitz gravity* (HLG), originally introduced in [38] (see [39, 40] for reviews). In this continuum formulation, the fundamental diffeomorphism symmetry of general relativity is replaced by the reduced symmetry of foliation-preserving diffeomorphisms. The motivation behind this construction is that one would like to add higher-order spatial derivatives to the gravitational action to render the theory renormalizable. In order to make the action Lorentz-invariant, it would be necessary to also add higher-order derivatives in time, which is known to lead to non-unitary quantum theories. In order to define a unitary and renormalizable theory of gravity, HLG postulates an anisotropic scaling of space and time at short distances and uses gravitational actions containing higher-order spatial derivatives and no more than two derivatives in time. Such a theory could still be compatible with observations, if the Lorentz symmetry violations are sufficiently suppressed at low energies.

The calculation of the spectral dimension d_s in HLG in 3+1 dimensions showed that it is $d_s = 4$ at macroscopic scales and $d_s = 2$ in the ultraviolet regime [41]. CDT and HLG gravity therefore not only share the presence of a foliation, but also the same qualitative behavior of the spectral dimension. These similarities led us to ask whether there are more profound connections between CDT and HLG. In chapter 4 we show that the phase diagram of CDT in 3+1 dimensions shows remarkable similarities to the generic Lifshitz phase diagram underlying HLG, which hints at the possibility that a suitable continuum limit of CDT may exhibit anisotropic scaling of space and time just like in HLG. The presence of a second-order phase transition line may even allow for both isotropic and anisotropic UV completions. If such a scenario was realized, CDT could serve as a unifying nonperturbative framework for anisotropic as well as isotropic theories of quantum gravity.

It is natural to ask what conditions need to be satisfied such that a physically interesting phase of geometry is present in a model of dynamical triangulations. One possibility to address this question is to consider alternative actions. In [42] the phase structure of CDT in 2+1 dimensions with a bare action of Hořava-Lifshitz type has been investigated with the result that a de Sitter phase still seems to be present among other new phases of geometry. In a different line of research, the original Euclidean dynamical triangulation model in 3+1 dimensions was modified by adding a nontrivial measure term which leads to the appearance of a new 'crinkled' phase [43]. Recently this model was newly investigated and the spectral dimension in the crinkled phase was found to run from $d_s = 4$ at low energies to approximately $d_s = 2$ at high energies [44], where it should be emphasized that $d_s = 4$ is not in itself an indicator of extended, semiclassical gravity.

CDT enforces causality by means of a distinguished foliation. One can therefore ask whether it is possible to relax this particular implementation of causality. In the chapters 5-7 of this thesis we present a new model of dynamical triangulations which we call *locally causal dynamical triangulations*, where each triangulation is built from Lorentzian building blocks and where the gluing rules are relaxed compared to CDT to allow for

triangulations that do not have a distinguished foliation. More precisely, we require the triangulations to be 'locally causal' in the sense that there exists a well-defined light cone at every point in the triangulation. This requirement implies that the flow of time is locally well-defined. In addition to local causality we also require that the flow of time can be consistently extended to the full triangulation. These two conditions alone do not explicitly exclude the presence of closed timelike curves. By choosing a particular (fixed) spacetime topology it is possible to choose a configuration space where the triangulations do not contain such curves.

We have implemented a simulation software for the locally causal DT model in 2+1 dimensions. In chapter 6 we present the results of our numerical investigations, where we have chosen the 3-sphere as spacetime topology. We show that there exists a region in the phase diagram where CDT-like triangulations dominate the regularized path integral. Furthermore we demonstrate the emergence of genuinely three-dimensional geometries, whose volume distribution excellently fit the volume profile of the classical three-sphere which is the Euclidean version of 2+1 dimensional de Sitter spacetime. On the basis of this result we conjecture that a 3+1 dimensional version of the locally causal DT model may pass the low-energy consistency check in the same way as standard CDT.

In the following chapter we present a compact review of standard CDT, including a discussion of the phase structure in 3+1 dimensions. In chapter 3 we present our analysis of the phase transitions in 3+1 dimensional CDT with the primary result that one of the transitions is of second order. Chapter 4 explores the connections between CDT and Hořava-Lifshitz gravity, based on the similarities between the phase diagrams of both models. In chapter 5 we introduce the theory of locally causal CDT in 2+1 dimensions, followed by a presentation of the numerical results and their implications in chapter 6. Chapter 7 contains an extensive discussion of the implementation of the Monte Carlo simulation, with the main emphasis on the Monte Carlo moves. In chapter 8 we summarize our results. Appendix A contains the documentation of the simulation software for the locally causal DT model, including detailed instructions on how to compile and use the simulation software.

2 Causal Dynamical Triangulations

In this chapter we present a compact review of Causal Dynamical Triangulations, which is a proposal for a theory of quantum gravity that uses a lattice regularization to define the gravitational path integral. This review is mainly based on standard CDT literature in 1+1 dimensions [33], 2+1 dimensions [45] and 3+1 dimensions [34, 35, 37], supplemented with newly created illustrations. For more extensive CDT reviews we refer the reader to [46, 47, 48]. We begin with explaining the theoretical background and afterwards focus on the phase structure of CDT in 3+1 dimensions to prepare the discussion of the phase transitions in the following chapter.

2.1 Theoretical background

The central object of interest in the Dynamical Triangulation (DT) approach is the gravitational path integral, which in the continuum can be formally expressed as

$$Z(G,\Lambda) = \int_{\text{geometries}} \mathscr{D}[g] \exp(iS_{\text{EH}}[g]). \tag{2.1}$$

$S_{\text{EH}}[g]$ is the Einstein-Hilbert action as a functional of the metric g, $\mathscr{D}[g]$ is a measure on the space of geometries (equivalence classes of metrics under the action of the diffeomorphism group), G is the Newton coupling and Λ is the cosmological constant. To properly define the path integral it needs to be regularized, which is done by means of a discretization of spacetime. There is no unique way of performing this discretization. First the elementary building blocks need to be chosen. For example, in two dimensions we can choose to work with triangles, quadrilaterals or other polygons.

Eq. (2.1) represents a sum over curved spacetimes, therefore we need to have a way of encoding curvature. On the left of Fig. 2.1 a flat disc has been triangulated using equilateral triangles, and then one of these triangles has been removed. The two sides of the removed triangle form the *deficit angle* θ. By gluing the two sides of the angle together we get a cone, as shown on the right of the figure. If we parallel transport a vector by circling once around the cone, the vector gets rotated by θ and therefore the deficit angle characterizes the local curvature at the tip of the cone [27].

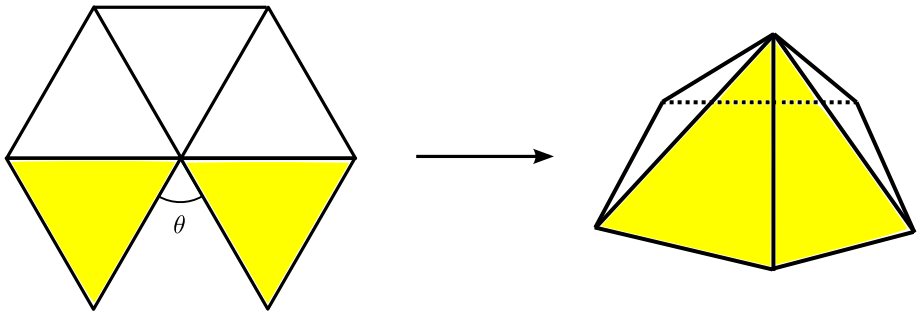


Figure 2.1: A disc in the flat plane is triangulated using equilateral triangles, then one of them is removed (left). By gluing the two sides of the angle θ together we get a cone with a positive deficit angle (right).

By removing more than one triangle, the deficit angle and therefore the curvature increases. On the other hand, one can also consider situations where more than six triangles share one vertex. This corresponds to a geometry with negative curvature. The number of triangles around a vertex is not limited from above while it is obviously limited from below. The precise lower limit is not uniquely fixed and requires another choice to be made. If the number of triangles around a vertex is two, then both triangles share the same vertices. If it is one, then two of the vertices of the triangle are identical. Depending on whether one accepts one or both of these types of degeneracies, the minimal number of triangles around a vertex is 1, 2 or 3.

In the Dynamical Triangulations approach in two dimensions each geometry is represented by a triangulation consisting of equilateral and flat triangles. The number of triangles around each vertex characterizes the local curvature at that vertex. The sum over geometries in the continuum path integral (2.1) becomes a sum over triangulations. Here we have to make a choice because we do not have any laws which dictate the precise definition of the ensemble of all triangulations to be summed over. It is not clear a priori whether one should sum over topologies or work with a fixed topology, and in the latter case one needs to choose a specific topology. Furthermore, one can choose to work with building blocks having Euclidean or Lorentzian signature. The imposition of physical principles such as causality also influences the precise nature of the ensemble. A priori, one does not know which of these choices affect the universal properties of the model at low energies.

Fig. 2.2 shows two examples of two-dimensional triangulations. On the left we see an example of a Euclidean triangulation whose building blocks are equilateral triangles. These do not appear equilateral in the illustration because it is impossible to flatten out a curved surface and preserve the edge lengths at the same time. The metric inside each triangle is uniquely fixed by the length of its edges and its signature is Euclidean. On the right we see a triangulation consisting of Lorentzian triangles with two timelike links and one spacelike link. Also here, the interior geometry is uniquely fixed by the edge lengths. The set of all spacelike links forms a foliation of the geometry. This is the primary ingredient of *Causal Dynamical Triangulations* (CDT).

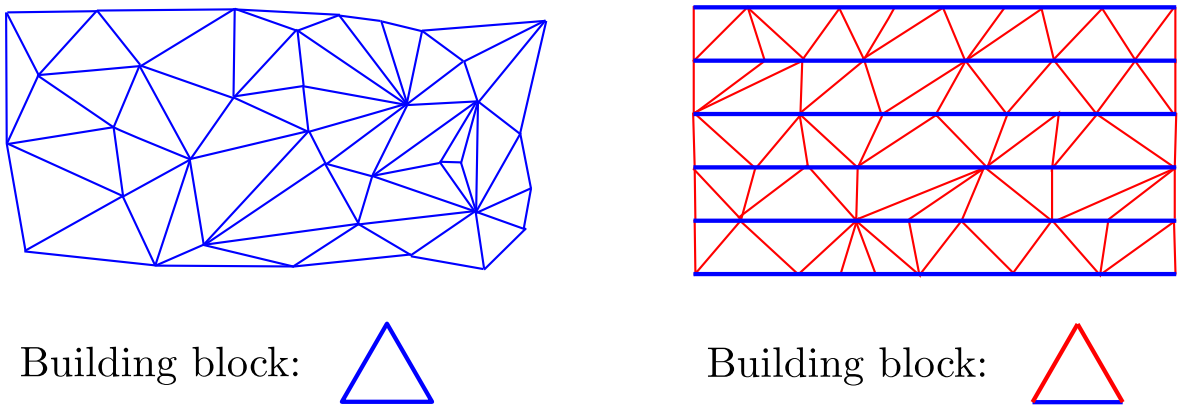


Figure 2.2: A Euclidean triangulation in $d = 2$, consisting of equilateral triangles with Euclidean signature (left), and a foliated causal triangulation in $d = 1 + 1$, containing triangles with two timelike links and one spacelike link (right). Timelike links are shown in red, spacelike ones in blue.

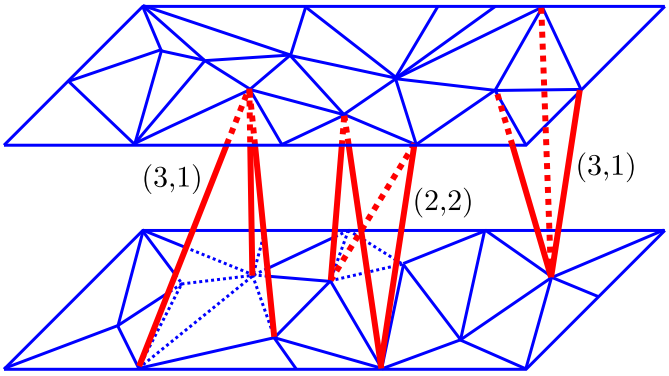


Figure 2.3: Two spatial slices in a foliated triangulation in 2+1 dimensions. Tetrahedra of both types are shown.

The idea of summing over foliated Lorentzian triangulations can easily be extended to higher dimensions. Fig. 2.3 shows a part of a 2+1 dimensional causal triangulation. Each leaf of the foliation forms a two-dimensional triangulation of the same fixed topology $\mathscr{T}$, consisting of equilateral spacelike triangles with link length a_s. Adjacent spatial surfaces are connected using Lorentzian tetrahedra to form a 2+1-dimensional simplicial manifold with spacetime topology $\mathscr{T} \times [0, 1]$. In simulations, time is often compactified for reasons of simplicity. The topology then becomes $\mathscr{T} \times S^1$.

For each Lorentzian tetrahedron the types of its links are completely determined by the foliation. If a link is contained in a spatial surface it is spacelike, if it connects two adjacent spatial surfaces it is timelike. The squared lengths of both types of links is allowed to differ, so that we can write $a_t^2 = -\alpha a_s^2$ with a_t^2 being the squared length of the timelike links. The factor of proportionality α is called the *asymmetry parameter*. In 2+1 dimensions there are two types of tetrahedra depending on how many of its links are spacelike. The (3,1) tetrahedron has three spacelike links forming a spacelike triangle while the (2,2)

tetrahedron contains only two spacelike links. The notation (i,j) means that i vertices of the tetrahedron are located on one spatial slice and the remaining j vertices on an adjacent one.

CDT in 3+1 dimensions can be defined in a similar way. The leaves of the foliation are then three-dimensional Euclidean triangulations of fixed topology. Neighboring slices are connected using Lorentzian 4-simplices. There are two types of these building blocks, denoted by (4,1) and (3,2), depending on how the vertices are distributed among the two spatial slices. By labeling the foliation with increasing integers we get a time variable for free, thus every vertex is automatically assigned a discrete time.

The action in the path integral (2.1) needs to be discretized as well. In the case of the Einstein-Hilbert action there exists a straightforward discretization called the Regge action [27], but more complicated actions can be discretized as well, see for example [42]. CDT with the Regge action can be solved analytically in two dimensions [33], but already the three-dimensional model has only been solved partially and for restricted classes of triangulations [49]. In four dimensions analytical methods are mostly unavailable and one must resort to Monte Carlo simulations to extract physical results.

In order to analyze the dynamics of CDT using simulations, a Wick rotation is applied to both the action and the triangulations. This is achieved by performing an analytic continuation of the asymmetry parameter, by rotating it in the lower half of the complex plane such that α is mapped to $-\alpha$. As a consequence, the gravitational path integral (2.1) turns into a statistical partition function of the form

$$Z = \sum_{T \in \mathscr{C}} \frac{1}{C(T)} \exp(-S^{\text{eucl}}_{\text{Regge}}(T)).$$

$\mathscr{C}$ is the space of all causal triangulations, $S^{\text{eucl}}_{\text{Regge}}$ is the Wick-rotated Regge action and $1/C(T)$ is the discrete analog of the path integral measure, with $C(T)$ being the order of the automorphism group of the triangulation T. By explicitly evaluating the Wick-rotated Regge action on foliated triangulations it acquires the simple form

$$S^{\text{eucl}}_{\text{Regge}} = -(\kappa_0 + 6\Delta)N_0 + \kappa_4 N_4 + \Delta(N_4 + N_4^{(4,1)}),$$

where N_0, N_4 and $N_4^{(4,1)}$ denote the numbers of vertices, four-simplices and four-simplices of type (4,1). The three couplings κ_0, κ_4 and Δ can be expressed as functions of the bare gravitational coupling, the bare cosmological constant and the asymmetry parameter α introduced above. By defining $\widetilde{\kappa}_4 = \kappa_4 + \Delta$ one obtains the version of the Euclideanized Regge action which is ultimately implemented in the computer simulations, namely,

$$S^{\text{eucl}}_{Regge} = -\kappa_0 N_0 + \widetilde{\kappa}_4 N_4 + \Delta(N_4^{(4,1)} - 6N_0), \tag{2.2}$$

where we have grouped together the terms proportional to Δ. We denote the quantity conjugate to Δ as $\text{conj}(\Delta) := N_4^{(4,1)} - 6N_0$. This expression will reappear in the discussion of the phase transitions in chapter 3.

For practical reasons the total system size N_4 is kept approximately fixed in the simulation and measurements are taken whenever the system size exactly equals N_4. This

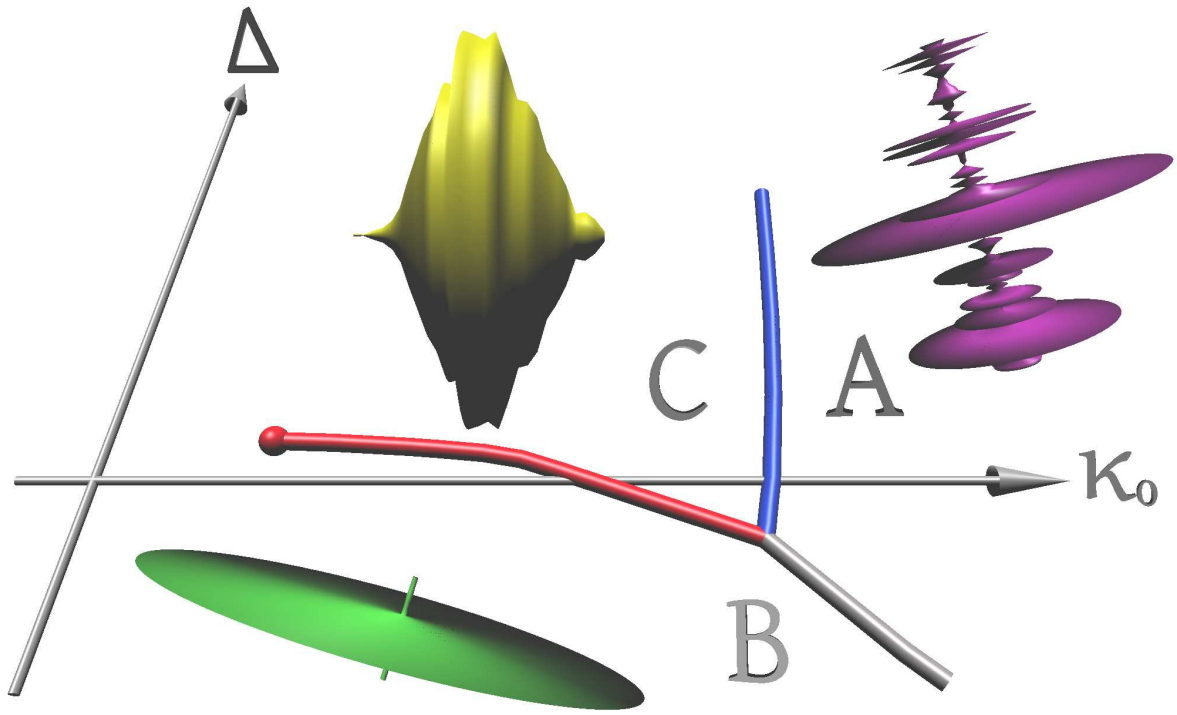


Figure 2.4: Visualization of the phase diagram of CDT in 3+1 dimensions, illustrating the volume profiles characterizing the three observed phases A, B and C.

amounts to simulating the canonical ensemble which is related to the grand canonical ensemble by a Laplace transformation [35]. We are left with two couplings κ_0 and Δ, which span the phase diagram of the system.

2.2 The phase structure of CDT in 3+1 dimensions

Fig. 2.4 shows a visualization of the phase diagram of CDT in 3+1 dimensions. It contains three phases which were originally labeled by A, B and C in [35]. The representative geometries in each of these phases can be distinguished by looking at their volume profile $N_3(t)$, which shows how the spatial extent of the geometries changes as a function of the canonical time variable provided by the foliation. More precisely, the profile is generated by counting the number of spacelike tetrahedra in every leaf of the foliation and by plotting these numbers as a function of the time variable. In Fig. 2.4 these volume profiles have been rotated around the t-axis.

In phase A we observe a sequence of mostly uncorrelated spatial slices. This phase is the analog of the branched polymer phase in Euclidean Dynamical Triangulations which appears there at large values of the inverse Newton coupling [28, 29, 30]. The phase B is dominated by geometries where almost all the 4-simplices are found in a slice of minimal time extension $\Delta t = 2$. In addition, almost all the spacelike tetrahedra are found in one spatial slice thus the volume profile looks similar to a delta function. Another

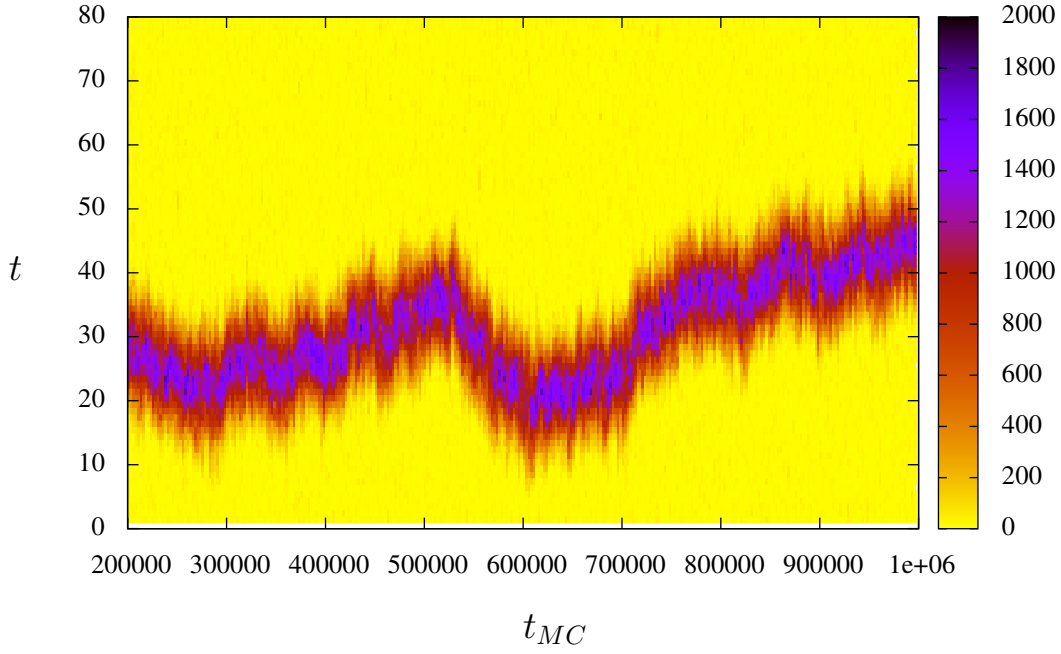


Figure 2.5: An illustration of the Monte Carlo evolution inside phase C, illustrating the random walk of the extended geometry along the time axis.

property of these geometries is the appearance of two 'supervertices' which are vertices with extremely high coordination number. If the dominant spatial slice is located at $t = t_0$, the two supervertices are found at $t = t_0 - 1$ and $t = t_0 + 1$. Most 4-simplices connect one of these two vertices with a spatial triangle in the large spatial slice. The phase B can be seen as an analog of the crumpled phase in Euclidean Dynamical Triangulations which is also characterized by the presence of supervertices [28, 29, 30].

In phase C we observe that the time axis is effectively partitioned into two parts, each with non-minimal time extension. Most 4-simplices are located in one part and form an extended geometry with a characteristic blob-shaped volume distribution. The remaining 4-simplices form a thin tube of minimal spatial extent. The latter is interpreted as a lattice artifact because the kinematical constraints prevent the tube from vanishing completely. Therefore this part has been cut away from the visualization in Fig. 2.4. The measurement of the macroscopic Hausdorff dimension, obtained using a finite-size scaling analysis of the volume profile, shows that the extended part of the geometries in phase C is four-dimensional [50].

Observables such as the macroscopic dimension are calculated by computing the dimension for each individual triangulation sampled during the Monte Carlo process and by averaging over all these results. There is an additional complication in the case of observables such as the volume distribution, which involves the time variable. The geometries in phase C perform a slow random walk along the t-axis, as shown in Fig. 2.5. The horizontal axis represents the Monte Carlo time t_{MC} in units of sweeps, with one sweep corresponding to a million attempted Monte Carlo moves. On the vertical axis the geometric time t provided by the foliation is shown. The color of the pixel at the loca-

tion (t_{MC}, t) encodes the spatial volume at geometric time t in the triangulation sampled at Monte Carlo time t_{MC}.

The random walk is a consequence of time translation symmetry, since by our choice of periodic boundary condition in the time direction, there is a priori no distinguished location on the time axis. In order to compute a meaningful average volume distribution, the random walk needs to be filtered away by aligning the individual distributions before averaging over them. A straightforward (but not unique) way of achieving this is to superimpose the distributions such that their center of volume agrees. This averaging procedure has been employed in [37] to study the average volume distributions and their fluctuations in phase C, leading to the conclusion that the low-energy limit of CDT in that phase is compatible with standard quantum cosmology.

3

Phase transitions of CDT in 3+1 dimensions

This chapter is based on the following publications:

A second-order phase transition in CDT (with J. Ambjørn, J, Jurkiewicz, R. Loll), Physical Review Letters 107 (2011) 211303, arXiv:1008.3932 [hep-th]

Second- and First-Order Phase Transitions in CDT (with J. Ambjørn, J, Jurkiewicz, R. Loll), Physical Review D85 (2012) 124044, arXiv:1205.1229 [hep-th]

A crucial and nontrivial requirement for all nonperturbative approaches of quantum gravity is that they must reproduce the well-established classical behavior of gravitational physics in the low-energy limit. In this respect CDT has had some remarkable successes. Numerical simulations of the model have demonstrated the dynamical emergence of a four-dimensional classical universe as the ground state of the system, even though no background was ever put in by hand [50]. In addition, a more detailed analysis revealed that the effective action for the scale factor of the universe can be matched to a minisuperspace action describing a quantum de Sitter spacetime [37].

Now that CDT has shown a promising infrared limit, it is time to focus our view towards UV physics. An important question is, whether it is possible to define a continuum limit of CDT in 3+1 dimensions by making the discretization scale arbitrarily small. To address this question we need to study the phase diagram of CDT and in particular the properties of the phase transitions. In traditional systems with static lattices and local field degrees of freedom, a continuum limit is defined by an approach to a second-order phase transition. The main purpose of this chapter is to demonstrate that there is strong evidence for the presence of a second-order transition in CDT in 3+1 dimensions.

We begin by presenting precision measurements of the phase transition locations which yield a quantitative phase diagram. Afterwards we explain the methods that we use to determine the order of the phase transitions. We then apply these methods to the A-C transition which we determine to be first order. Afterwards we present the results of the analysis of the B-C transition which strongly suggest that it is second order. Finally we summarize our results and discuss possibilities for future research.

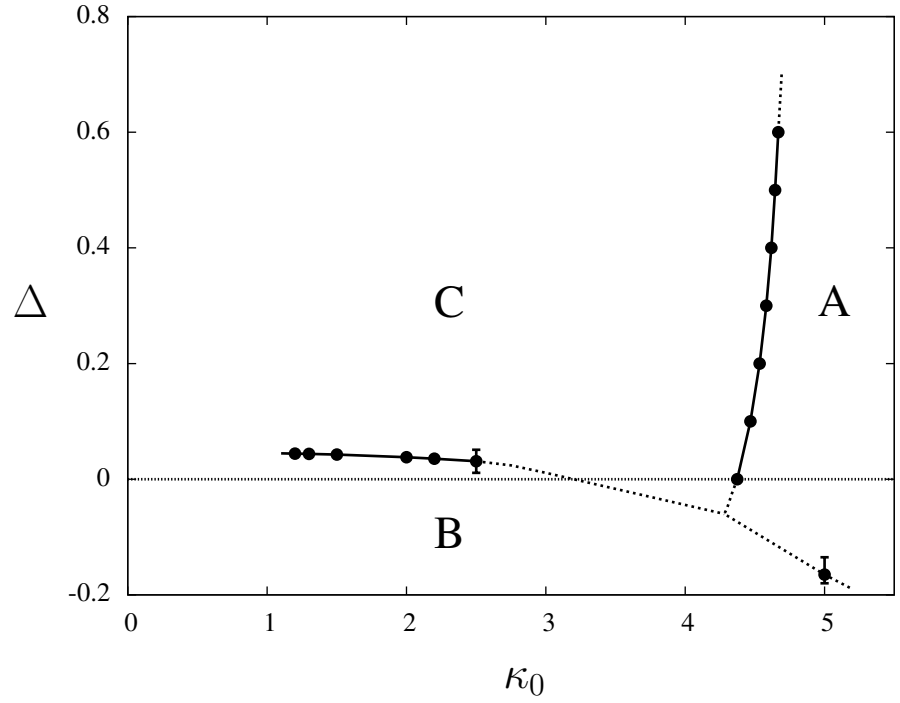


Figure 3.1: The phase diagram of CDT in 3+1 dimensions for a system size $N_4 = 80000$. The large dots represent actual measurements.

3.1 The phase diagram

The first qualitative description of the CDT phase diagram appeared in [35]. In order to prepare a detailed investigation of the phase transitions we have run new simulations using a system size of $N_4 = 80000$ and a fixed spacetime topology of $S^3 \times S^1$ in order to determine the precise locations of the transitions points with the aim of producing a quantitative phase diagram. Figure 3.1 shows the result of our analysis. The large dots represent actual measurements and the solid lines are linear interpolations. The dashed lines mark a region in the phase diagram where technical difficulties have prevented us from making precise measurements. We will return to this issue later on.

Let us comment on some technical issues with regard to measuring the location of a phase transition. The very notion of a phase transition point is ambiguous for finite systems because, strictly speaking, phase transitions can only occur in the limit of infinite size. For example, consider the susceptibility $\chi_{\mathscr{O}} = \langle \mathscr{O}^2 \rangle - \langle \mathscr{O} \rangle^2$ associated with some observable $\mathscr{O}$. Plotting the function $\chi_{\mathscr{O}}(\kappa_0 = \text{const}, \Delta)$ near the B-C transition, one will find that it has a maximum at some value Δ_c. The location of this maximum can be used as a definition of the transition point, but one must keep in mind that its precise value will in general depend weakly on $\mathscr{O}$.

Another definition comes from studying histograms of suitable observables. Consider the Monte Carlo time evolution of the quantity N_0 near the A-C transition shown in Fig. 3.2 (top left). We observe that N_0 for some time fluctuates around one value, then makes a transition to another value, around which it fluctuates for a while, before flip-

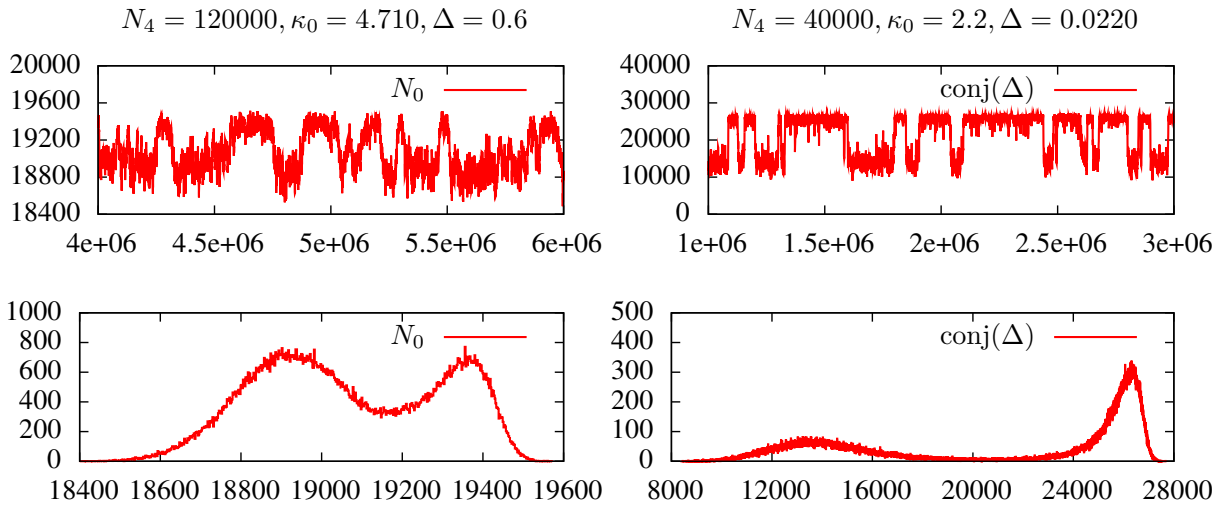


Figure 3.2: Monte Carlo time evolution (i) of N_0 at the A-C transition (top left) with associated histogram (bottom left), (ii) of conj(Δ) at the B-C transition (top right) with associated histogram (bottom right).

ping back to the first value again. This behavior is characteristic of a phase transition with the two different states corresponding to the two phases involved. The associated histogram of N_0 (Fig. 3.2, bottom left) shows a double-peak structure. Varying κ_0 while holding Δ fixed will change the relative heights of the two peaks. We can now fine-tune κ_0 so that both peaks have the same height. This gives us another definition of the phase transition point, although often not the most practical one, since the fine-tuning can be very time-consuming. Also there may be regions of coupling constant space or volume sizes where the flipping between the two states is not strong enough, in which case the histogram will not exhibit two peaks but look more like a deformed Gaussian distribution.

As we move towards $N_4 \to \infty$, all of these definitions will converge to one and the same phase transition point at infinite four-volume. However, what is important for our purposes is the observation that already for moderately large system sizes those definitions will produce values so close to each other as to be practically indistinguishable on a plot like that of Fig. 3.1. In order to determine the phase diagram it is therefore sufficient to fine-tune the parameters until a flipping of phases is observed. Not all observables are equally suited for this kind of analysis, because the flipping between phases can occur with different amplitudes for different observables. A good choice is usually the quantity conjugate to the coupling that needs to be fine-tuned. The form of the action (2.2) suggests using the quantity N_0 for the A-C transition and $\text{conj}(\Delta) = N_4^{(4,1)} - 6N_0$ for the B-C transition. The Monte Carlo time evolution and associated histogram for the latter are shown in Fig. 3.2 on the right (top and bottom).

It is instructive to track how the phase transition signature changes as one moves along the phase transition lines while holding the system size $N_4 = 80.000$ fixed. Along the A-C transition we have not observed any appreciable change, although we have not

investigated the immediate neighborhood of the triple point where all three phases meet. The situation for the B-C transition is rather different. Let us start on the B-C line at the point $\kappa_0 = 2.2$ and move in both directions along the line. As we move to the left, the jump in conj(Δ) associated with the phase flip decreases, and around $\kappa_0 = 1.0$ no observable signature of a phase transition remains. We conclude that the B-C transition line has an end point which for $N_4 = 80.000$ is located around $\kappa = 1.0$. Its precise location remains to be determined.

Conversely, when moving to the right from $\kappa_0 = 2.2$, we observe that the jump in conj(Δ) grows rather quickly, until at $\kappa_0 = 2.5$ we no longer observe the phase flipping, although the reason for this is very different from the situation encountered at the left end point. The increased size of the jump in conj(Δ) is accompanied by a deepening of the vertical gap separating the two peaks of the histogram of conj(Δ). As it becomes deeper, the frequency of the phase flipping decreases significantly, until it becomes so small that no flipping is observed during the entire simulation. This indicates that the simulation becomes stuck in a metastable state and does not reach thermal equilibrium. In Fig. 3.1 this region is marked by a dashed line; here conventional methods are insufficient to measure the phase transition location with acceptable accuracy.

In principle there exists a method which offers a solution for these kind of problems and which is known under the name 'multicanonical Monte Carlo simulations' [51]. In conventional Monte Carlo simulations the states of the ensemble are sampled such that they appear with the Boltzmann probabilities. To calculate observables it is then sufficient to calculate a simple unweighted average. It is possible to generalize the standard Monte Carlo method to sample the states with another probability distribution if the formula for calculating the observables is adapted accordingly. The main idea of multicanonical Monte Carlo simulations is to choose the probability distribution in such a way that the resulting histogram becomes flat in the region of interest. It is clear that in our particular case the replacement of a sharp double peak histogram by a flat histogram would remove the metastability.

The main challenge with this approach is that the desired probability distribution is not known beforehand. Before the actual simulation can be run it is therefore necessary to run a separate program which attempts to generate the desired probability distribution. Such algorithms have been described for example in [51] and [52]. These are iterative procedures which consist of a large number of short Monte Carlo simulations where the current probability distribution is modified after every run according to special rules which are designed in such a way that the probability distribution should ultimately converge to a distribution which produces a flat histogram.

Unfortunately our attempts of using this technique were unsuccessful. The algorithm to generate a probability distribution seemed to work in principle although it was extremely slow. At some locations on the B-C phase transition the distributions only converged after more than a year of running time. In the end it turned out that the quality of the generated distributions was insufficient to remove the metastabilities. A possible reason for the ineffectiveness of the multicanonical method may be that simulations of dynamical triangulations generically run much slower than conventional simulations on

fixed lattices. This also affects the algorithm which generates the probability distributions since it runs a large series of short Monte Carlo simulations. It may be that the running time for these individual runs was too small for the algorithm to work properly.

3.2 How to measure the order of phase transitions

Measuring the order of phase transitions requires some care, as is illustrated by the history of dynamical triangulations. The quantum gravity model based on four-dimensional *Euclidean* triangulations (which lack the foliation present in CDT) has a phase transition, which initially was determined to be of second order [28, 30, 53]. However, later studies with larger systems established instead that the phase transition is of first order [31, 32].

Criteria to distinguish between first- and second-order transitions have been summarized conveniently in [54], although care should be taken in extrapolating from conventional systems, simulated on static lattices, to dynamical triangulations where the lattice itself is dynamical. Let us begin by considering the situation of infinite system size. In this case first-order transitions are uniquely characterized by the existence of observables (first derivatives of the free energy) that are discontinuous across the transition. In addition, the size of the fluctuations relative to the average tends to zero in the infinite-volume limit, implying that the probability distribution of such an observable approaches the sum of two delta-function distributions. The displacement between the two peaks is precisely the difference between the two equilibrium values of the observable.

The probability distribution of an observable in the infinite-volume limit can be understood as arising through a limiting process of the corresponding distributions for finite volume. Their measured counterparts are histograms of the kind displayed in Fig. 3.2. It is sometimes stated in the literature that a double-peak structure of a histogram signals a first-order transition, but this is misleading and can lead to wrong conclusions. Rather, the existence of a double-peak histogram allows one to confirm the first-order nature of a transition by considering a sequence of histograms for increasing system size. We are dealing with a first-order transition whenever the double-peak structure becomes more pronounced with increasing volume. Quantitatively, this means that the vertical gap associated with the double peak increases. By 'vertical gap' we mean the difference between the peak heights (assuming they are equal) and the height of the minimum in between them. How the vertical gap changes with system size is the method of choice to confirm the first-order nature of a transition, provided one can simulate sufficiently large systems, which develop a double-peak structure.

One could in principle try to use the same method to conclude that a transition is of second or higher order by measuring the vertical gap as a function of inverse system size and extrapolating to the limit of infinite volume, where it should vanish. However, because it is often difficult to measure this gap with good accuracy, showing conclusively that it really goes to zero in the limit can be expected to be very difficult. Fortunately, there are better criteria at hand, involving the measurement of critical exponents, which one can use to establish that a given transition is of higher order.

One such exponent measures the shift of a transition point with system size. Recall first how this works for a conventional lattice system such as the Ising model with volume $V = L^d$ where d is the system's dimension [55]. Considering the temperature-driven phase transition of the Ising model and using the location of the maximum of the magnetic susceptibility to define a transition point $\beta^c(V)$, one finds a power-law behavior

$$|\beta^c(\infty) - \beta^c(V)| \propto V^{-1/\nu d} \tag{3.1}$$

for sufficiently large system size. The exponent ν governs the increase of the correlation length in a second-order transition as one moves towards the critical point $\beta^c(\infty)$ on an infinite lattice. For first-order transitions there is no correlation length and one expects the specific scaling [54]

$$|\beta^c(\infty) - \beta^c(V)| \propto V^{-1/\tilde{\nu}}, \quad \tilde{\nu} = 1. \tag{3.2}$$

A sufficiently strong violation of $\tilde{\nu} = 1$ therefore signals the presence of a second-order transition.

To work with the criteria (3.1) or (3.2) one needs some way of judging whether the system sizes under consideration are large enough. Given N data points ordered by system size, we can make M fits where the i-th fit, $i \in \{1, \ldots, M\}$, is made by using the restricted set of data points with labels $i, \ldots, N$. If the corresponding sequence of exponents is drifting, it can be an indication that the system sizes are not large enough. Obviously, this method is only useful when one has sufficiently many data points.

Another quantity of interest is the so-called Binder cumulant associated with an observable $\mathscr{O}$, which may be defined as [54]

$$B_{\mathscr{O}} = \frac{1}{3}\left(1 - \frac{\langle \mathscr{O}^4 \rangle}{\langle \mathscr{O}^2 \rangle^2}\right) = -\frac{1}{3}\,\frac{\langle (\mathscr{O}^2)^2 \rangle - \langle \mathscr{O}^2 \rangle^2}{\langle \mathscr{O}^2 \rangle^2}, \tag{3.3}$$

and which is always nonpositive. Considering $B_{\mathscr{O}}$ as a function of the couplings, its local minima are at transition points, where the fluctuations are largest. We can measure these minima for different system sizes and by extrapolation determine $B_{\mathscr{O}}^{\min}(1/N_4 = 0)$. This quantity is zero when the probability distribution of $\mathscr{O}$ approaches a delta function around an expectation value $\langle \mathscr{O} \rangle$ in the infinite-volume limit, as is expected at a second-order transition. On the other hand, many aspects of first-order transitions are described well by approximating the histogram of the observable with a superposition of two distributions centred at the expectation values $\langle \mathscr{O} \rangle = \mathscr{O}_1$ and $\langle \mathscr{O} \rangle = \mathscr{O}_2$ in the two phases. Again, if these distributions approach delta functions in the infinite-volume limit, the minimum of $B_{\mathscr{O}}$ will be given by

$$B_{\mathscr{O}}^{\min}(1/N_4 = 0) = -\frac{(\mathscr{O}_1^2 - \mathscr{O}_2^2)^2}{12\,\mathscr{O}_1^2 \mathscr{O}_2^2}, \tag{3.4}$$

which is obtained for a value of the coupling constants where the relative strength of the delta functions at $\mathscr{O}_1$ and $\mathscr{O}_2$ is given by $\mathscr{O}_2^2/(\mathscr{O}_1^2 + \mathscr{O}_2^2)$ and $\mathscr{O}_1^2/(\mathscr{O}_1^2 + \mathscr{O}_2^2)$ respectively. We conclude that a sufficiently strong deviation of $B_{\mathscr{O}}^{\min}$ from zero signals a first-order phase

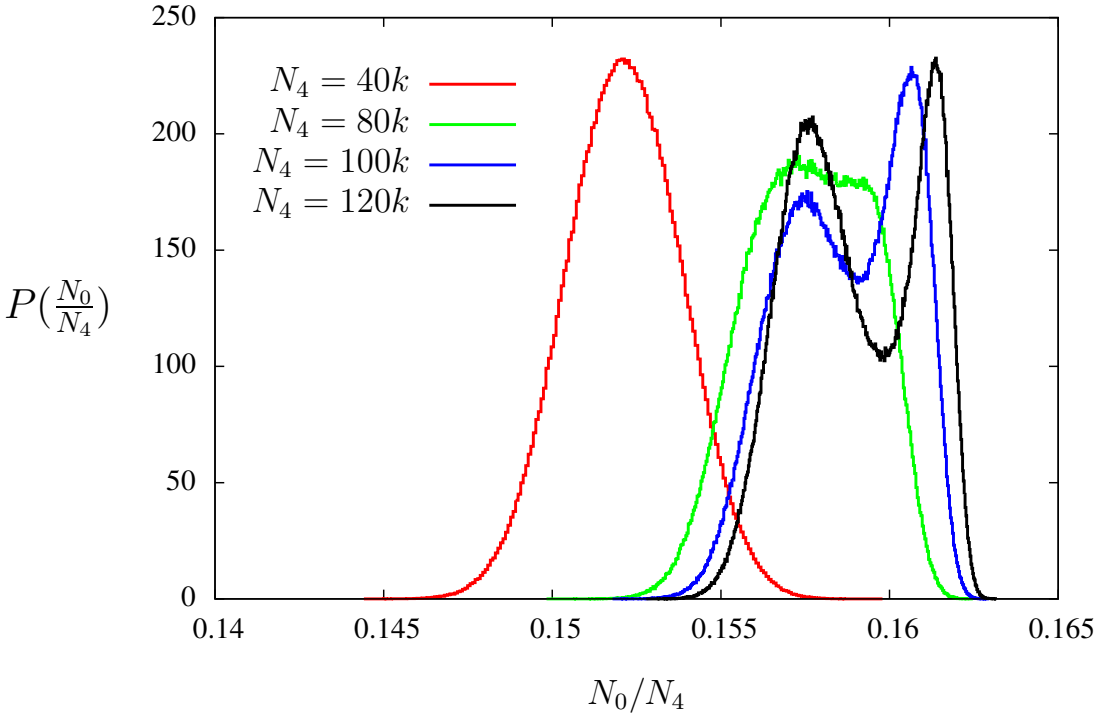


Figure 3.3: The histograms of N_0 at the A-C transition for four different system sizes, normalized to obtain probability distributions which means that the area under each histogram is equal to one. [1]

transition. Using the value of the Binder cumulant to make the case for a second-order transition is more difficult, because a weak first-order transition may show a convergence to a value close to zero.

There are alternative methods, involving the measurement of other critical exponents, which can help to determine the order of a phase transition, but their translation to dynamical triangulations leads to ambiguities. We will therefore proceed by applying the methods described above to the A-C and B-C transitions found in the CDT system. The third transition (A-B) is currently of minor interest, since it bounds two phases that most likely have no relevance for continuum physics.

3.3 The order of the A-C transition

The simulations for the analysis of the A-C transition were run at $\Delta = 0.6$ with system sizes ranging from $40k$ to $150k$. The data presented in the following are the results of simulations where the number of sweeps was approximately $5 \cdot 10^6$ with one sweep representing one million attempted Monte Carlo moves.

Our simulations suffer from a slow convergence of observables which means that the

[1] This figure corrects Fig. 4 in Ref. [56] where the two histograms with largest system size have been labeled incorrectly.

standard error algorithms produce error values which significantly underestimate the true uncertainties of the measurements. In all our measurements we have attempted to obtain more realistic error values by systematically studying the convergence of observables. Given a set of data samples at Monte Carlo times $\tau_1, \ldots, \tau_n$, we can define a time-dependent average $\langle O \rangle (\bar{\tau})$ by including only those data samples for which $\tau_1 \leq \tau \leq \bar{\tau}$. We then plot this quantity as a function of $\bar{\tau}$ and try to extract reasonable error values. This method introduces a degree of subjectivity, but in our opinion produces more realistic error estimates.

We first perform a histogram analysis. Fig. 3.3 shows the histograms for the observable N_0/N_4 at four different system sizes. The histogram at $N_4 = 40k$ assumes an approximate Gaussian shape which gets distorted at $N_4 = 80k$, then a double peak starts to emerge at $N_4 = 100k$ which becomes more pronounced at $N_4 = 120k$. For the two larger volumes, the vertical gap associated with the double-peak clearly increases, while the mutual distance of the peaks stays approximately the same. This behavior is a clear signature of a first-order transition. The emergence of the double-peak shape seems to occur somewhere between $N_4 = 80k$ and $N_4 = 100k$. This may be compared with the analysis performed in the context of four-dimensional Euclidean dynamical triangulations, where the double peak emerged between $N_4 = 16k$ and $N_4 = 32k$ [31]. The more rigid structure imposed on the triangulations by the causality conditions of CDT may be responsible for the shift of the appearance of the first-order signal to somewhat larger volumes.

Let us try to find additional evidence for the first-order nature of the A-C transition by first measuring the shift exponent $\tilde{\nu}$ defined in eq. (3.2). For sufficiently large system sizes we expect a power-law behavior

$$\kappa_0^c(N_4) = \kappa_0^c(\infty) - C N_4^{-1/\tilde{\nu}}, \tag{3.5}$$

where $\kappa_0^c(N_4)$ denotes the transition point at system size N_4 and C is a proportionality factor. We have already mentioned that the notion of a transition point is ambiguous for finite system sizes. For our present purposes, we define $\kappa_0^c(N_4)$ as the location of the maximum of the susceptibility $\chi_{N_0} = \langle N_0^2 \rangle - \langle N_0 \rangle^2$. To measure this maximum we use an extrapolation method due to Ferrenberg and Swendsen [57], also known as histogram method [55].

The best fit using all data points gives $\tilde{\nu} = 1.11(2)$. Unfortunately, the three-parameter fit (3.5) is not very stable when removing the points with the lowest values of N_4. Taking this into account it becomes clear that the error bar of 0.02 significantly underestimates the error associated with the determination of $\tilde{\nu}$ from the data. It indicates the presence of subleading terms which for the volume range studied interfere with the pure power-law behavior of (3.5).

We have made another three-parameter fit to the generic functional form of a large-N_4 expansion of $\kappa_0^c(N_4)$, valid at a first-order transition [54], namely,

$$\kappa_0^c(N_4) = \kappa_0^c(\infty) - C/N_4 - D/N_4^2 + O(1/N_4^3). \tag{3.6}$$

Fig. 3.4 shows the measured data points and the best fit based on the expansion (3.6). The errors associated with individual data points are too small to be visualized in a sensible

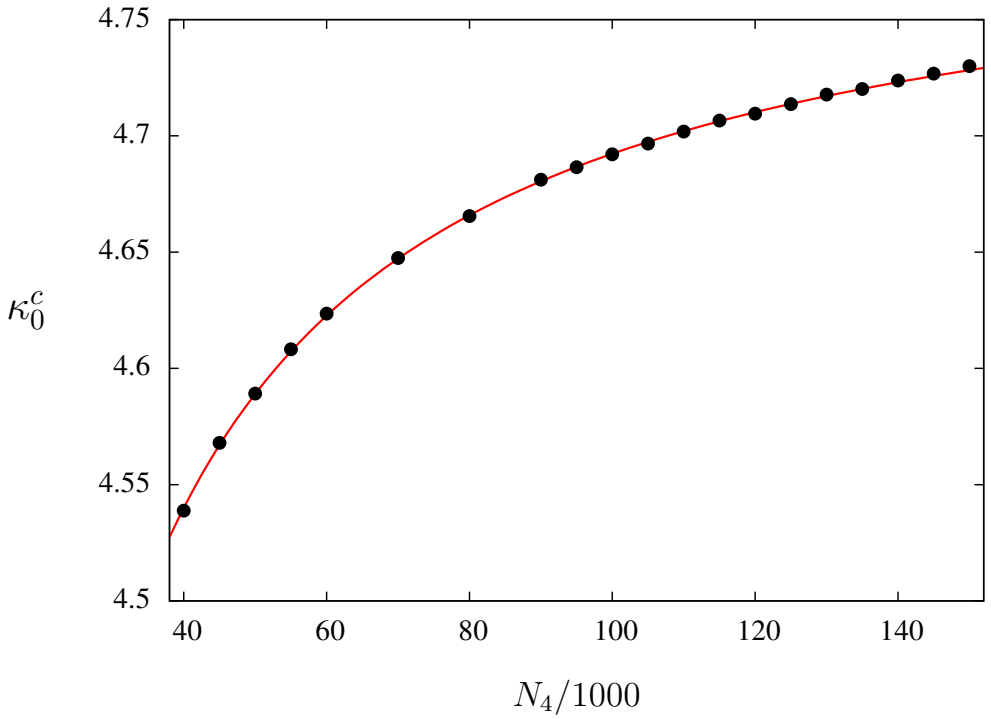


Figure 3.4: The measured A-C transition points $\kappa_0^c(N_4)$ at $\Delta = 0.6$ for different system sizes N_4, together with a best fit to the inverse-volume expansion eq. (3.6).

way. The quality of this fit is comparable with that obtained from fitting to the power law (3.5). To summarize, we can say with confidence that the shift data are compatible with a first-order transition. However, it is also clear that if one wanted to go beyond this statement and nail down the value of the critical exponent $\tilde{\nu}$ with better precision, one would need to consider larger system sizes.

Our next and final step in analyzing the A-C transition will be to consider the Binder cumulant for N_0,

$$B_{N_0} = \frac{1}{3}\left(1 - \frac{\langle N_0^4 \rangle}{\langle N_0^2 \rangle^2}\right), \tag{3.7}$$

which has a local minimum $B_{N_0}^{\min}$ at the A-C phase transition. Like in the case of the shift exponent $\tilde{\nu}$, we use the histogram method to determine this minimum. In Fig. 3.5 we have plotted $B_{N_0}^{\min}$ as a function of system size. The error bars are much larger than for the measurement of the shift exponent, where we measured the *location* of the susceptibility maximum and not the maximum value itself. The plot shows clearly that $B_{N_0}^{\min}$ moves away from zero as one goes to large system sizes, reconfirming the first-order nature of the transition.

By contrast, when going to smaller volumes one finds a different behavior where $B_{N_0}^{\min}$ increases before reaching a local maximum. Evidently, for these small volumes the system lies outside the scaling region where quantities like $B_{N_0}^{\min}$ are expected to behave according to a power law. Looking at the plot, a rough estimate of the onset of the scaling region seems to be around or above $N_4 = 60k$. Comparing with Fig. 3.3, this may be

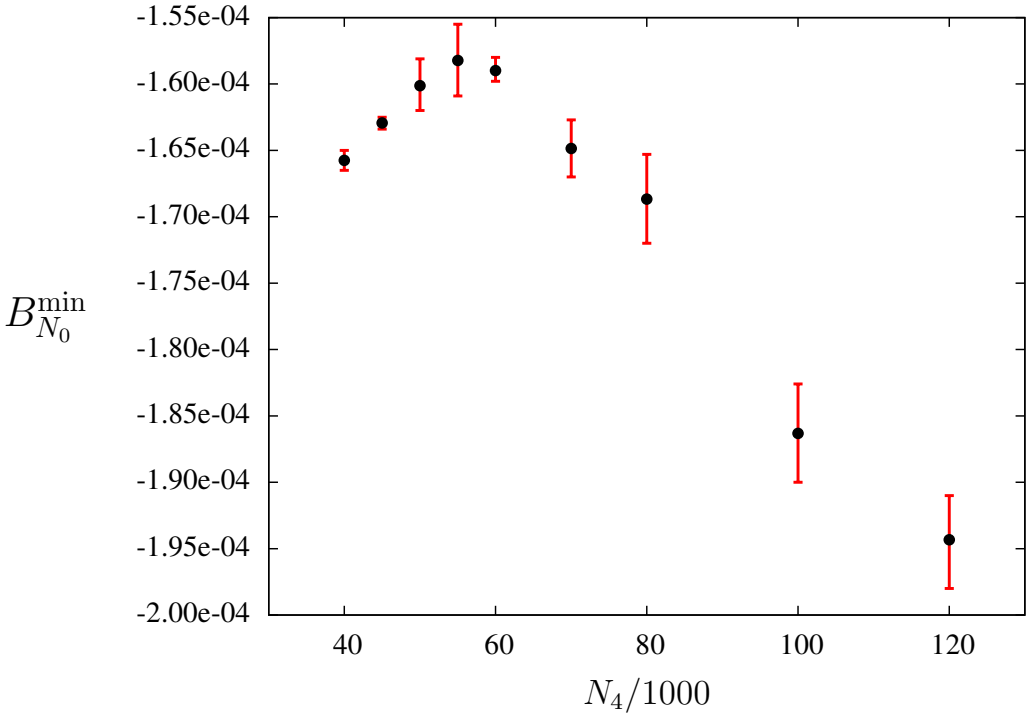


Figure 3.5: The dependence of the minimum $B^{\min}_{N_0}$ of the Binder cumulant B_{N_0} on the system size N_4 at the A-C transition and for $\Delta = 0.6$.

correlated with the observed emergence of the double-peak structure in the histograms.

3.4 The order of the B-C transition

Continuing our investigation of the order of phase transitions in CDT quantum gravity, we now turn to the B-C transition. In this case we have fixed the inverse gravitational coupling to $\kappa_0 = 2.2$ and analysed the system at sizes between $40k$ and $160k$. The number of sweeps used was approximately $2.5 \cdot 10^6$ with one sweep again corresponding to one million attempted Monte Carlo moves. Fig. 3.6 shows the histograms of the quantity $\mathrm{conj}(\Delta) = N_4^{(4,1)} - 6N_0$ for three different system sizes. The situation differs from that of the A-C transition in that we observe a double-peak structure for all four-volumes. The peaks are most pronounced for smaller volume and appear to be merging when the volume is increased. The plot shows that their mutual distance decreases, roughly like $1/N_4$. This is a first indication that the B-C transition may not be a first-order transition.

We proceed by measuring the shift exponent for Δ. Analogous to what we did for the A-C transition, we use the formula

$$\Delta^c(N_4) = \Delta^c(\infty) - CN_4^{-1/\tilde{\nu}}. \tag{3.8}$$

Δ^c is defined as the location of the maximum of the susceptibility $\chi_{\mathrm{conj}(\Delta)} = \langle \mathrm{conj}(\Delta)^2 \rangle - \langle \mathrm{conj}(\Delta) \rangle^2$. The data points and the best fit are displayed in Fig. 3.7. We have not included

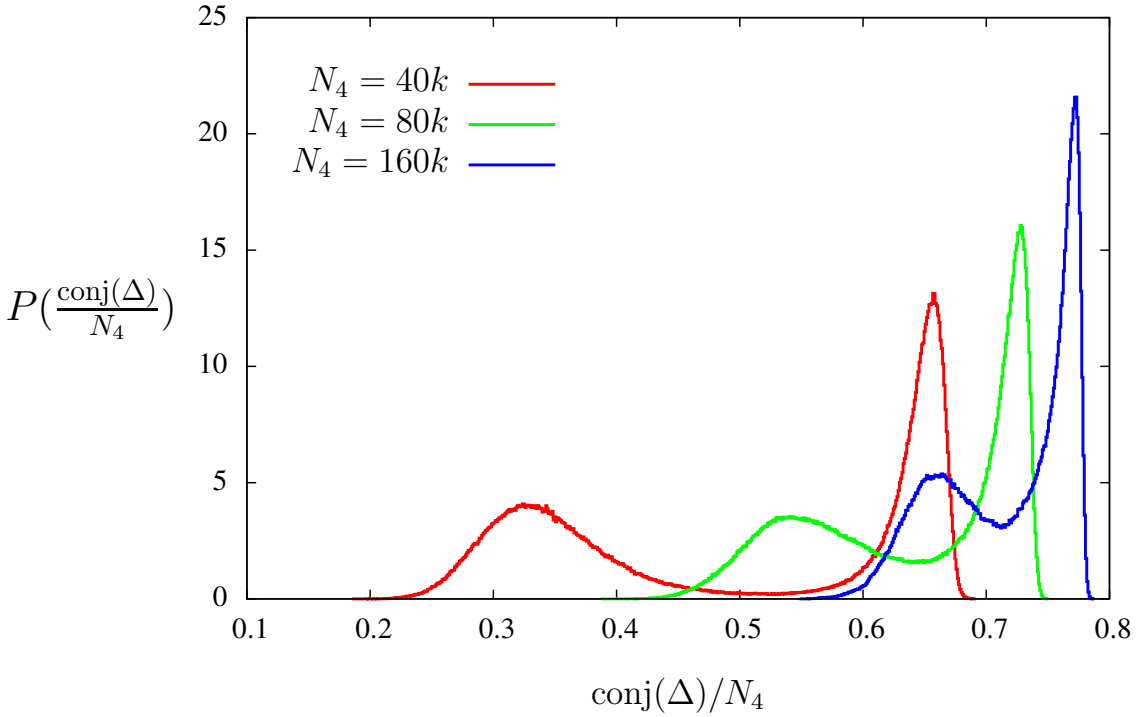


Figure 3.6: The histograms of conj(Δ) at the B-C transition for three different system sizes. The histograms have been normalized to obtain probability distributions such that the area under each histogram equals to one.

any error bars because they turned out to be too small. The best fit through all data points yields $\tilde{\nu} = 2.39(3)$. To judge whether our range of system sizes lies inside the scaling region we have again performed a sequence of fits by successively removing the data points with the lowest four-volume. The corresponding values for $\tilde{\nu}$ are $2.39(3)$, $2.51(3)$, $2.49(3)$ and $2.51(5)$, where the last fit was done with all but five data points removed. Again the error bars are based on making cuts in the sampled data as described above when discussing the A-C transition. The sequence suggests that the data point with the lowest four-volume lies outside the scaling region. Removing it from the fit we get

$$\tilde{\nu} = 2.51(3). \tag{3.9}$$

This result makes a strong case for a second-order transition, since the prediction $\tilde{\nu} = 1$ for a first-order transition is clearly violated.

Let us finally consider how the minimum of the Binder cumulant $B_{\mathcal{O}}$ of definition (3.3), with $\mathcal{O} = \mathrm{conj}(\Delta)$, depends on the system size. In Fig. 3.8 we show $B^{\mathrm{min}}_{\mathrm{conj}(\Delta)}$ as a function of *inverse* system size. The errors turned out to be much smaller than for the corresponding measurements at the A-C transition, with error magnitude approximately equal to the radius of the dots in the plot. Inside the scaling region the minimum of the Binder cumulant is expected to behave like a power-law function. To determine whether our data points lie inside this scaling region, we have again performed a sequence of fits by successively removing the data points with the lowest four-volume. We have found

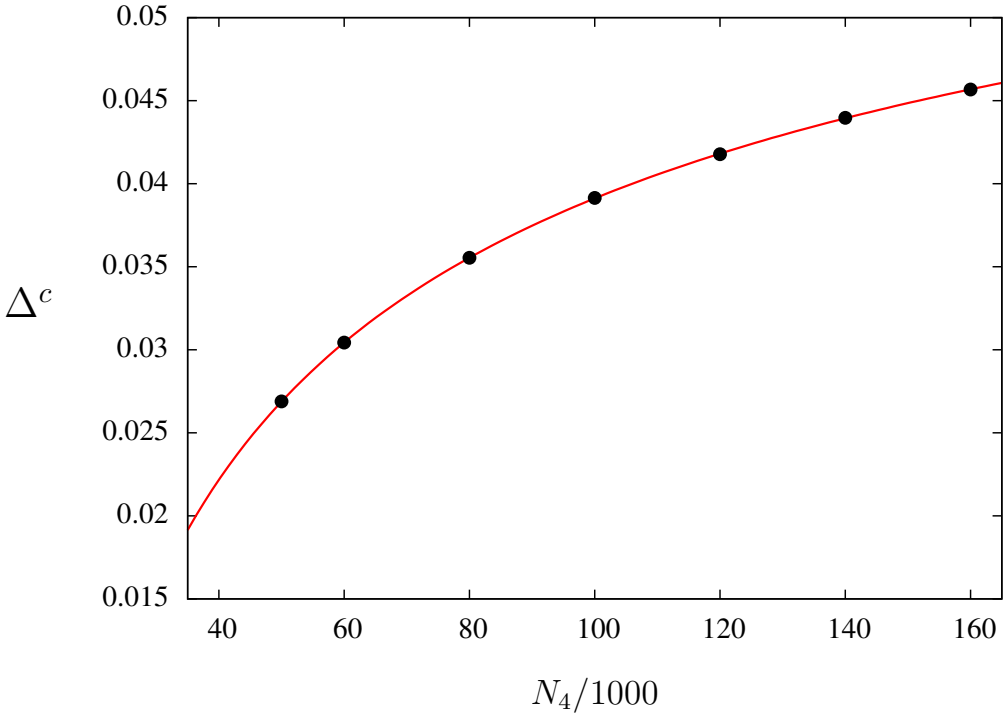


Figure 3.7: The measured B-C transition points $\Delta^c(N_4)$ at $\kappa_0 = 2.2$ for different system sizes N_4, with a best fit to eq. (3.8) to determine the shift exponent $\tilde{\nu}$.

Observable $\mathscr{O}$	$B^{\min}_{\mathscr{O}}(N_4 \to \infty)$
conj(Δ)	$-0.003(4)$
$N_4^{(4,1)}$	$-0.001(3)$
N_2	$-0.000\,000\,1(3)$
N_1	$-0.000\,003(7)$
N_0	$0.000\,0(3)$

Table 3.1: The measurements of $B^{\min}_{\mathscr{O}}(N_4 \to \infty)$ for various observables $\mathscr{O}$, where N_k denotes the number of k-dimensional (sub-)simplices in the triangulation.

that the data points at $N_4 = 40k$ and $N_4 = 50k$ both lie outside the scaling region. The curve displayed in Fig. 3.8 corresponds to the fit with those two points removed.

Table 3.1 collects the results of measuring $B^{\min}_{\mathscr{O}}(N_4 \to \infty)$ for five different observables $\mathscr{O}$. All of these observables exhibit a phase flipping at the transition with associated double-peak histograms. The interpolation to $N_4 \to \infty$ is made by assuming that the quantities $B^{\min}_{\mathscr{O}}(N_4)$ for sufficiently large volumes have a scaling behavior like in eqs. (3.5) and (3.8), namely,

$$B^{\min}_{\mathscr{O}}(N_4) = B^{\min}_{\mathscr{O}}(N_4 \to \infty) - C(\mathscr{O})/N_4^{\tilde{\rho}(\mathscr{O})}, \tag{3.10}$$

with $\mathscr{O}$-dependent exponents $\tilde{\rho}(\mathscr{O})$ and coefficients $C(\mathscr{O})$. If we had to rely on the Binder cumulant measurements alone to determine the order of the B-C transition, we would

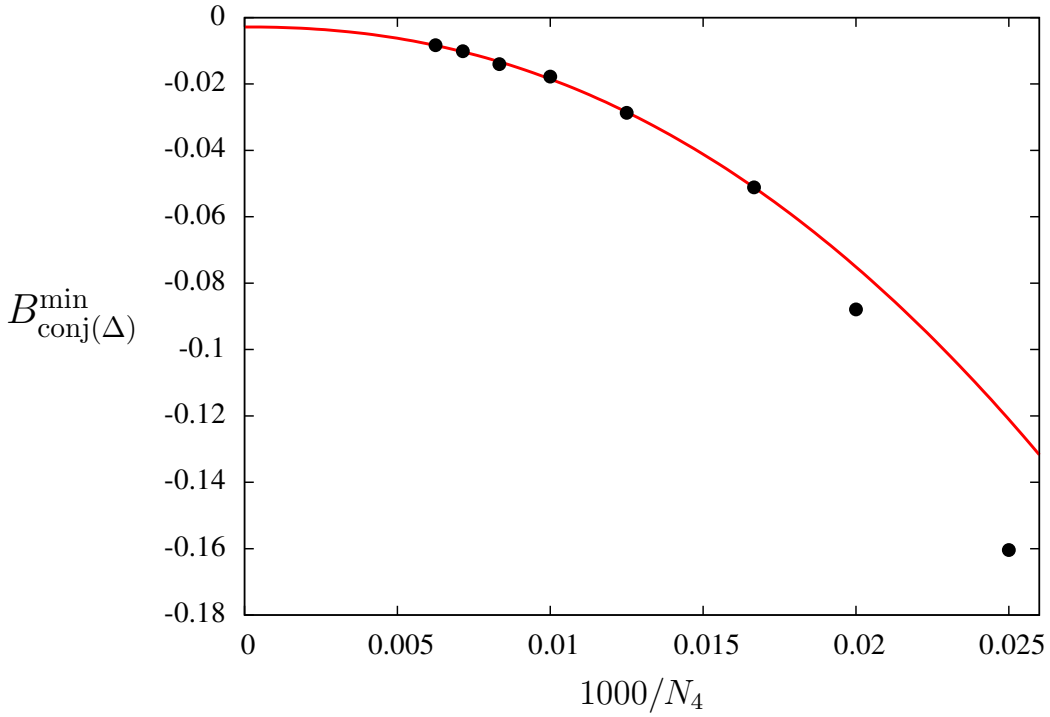


Figure 3.8: The dependence of the minimum $B^{\min}_{\mathrm{conj}(\Delta)}$ of the Binder cumulant $B_{\mathrm{conj}(\Delta)}$ on the (inverse) system size N_4^{-1}, at the B-C transition and for $\kappa_0 = 2.2$ (The fit excludes the two points on the right).

encounter the already mentioned problem of how to distinguish between a second-order transition and a first-order transition where $B^{\min}_{\mathscr{O}}$ converges to a value close to zero. In the case at hand, in addition to the evidence already presented in favor of a second-order transition, the size and quality of the measurement errors for the minimum of the Binder cumulants are such that we can conclude with confidence that the results displayed in Table 3.1 are certainly *consistent* with a limiting value of zero for $N_4 \to \infty$. This would be true even if the errors overestimated the true uncertainties for example by a factor of two.

3.5 Discussion and outlook

We set out with the goal to determine the order of the two physically relevant phase transitions in CDT. For conventional systems on static lattices, describing physics on a fixed background geometry, the methods for doing so are well established. It is not clear a priori whether all of them are applicable for systems based on dynamical lattices, such as CDT, which reflect the dynamical character of the quantum-gravitational, geometric degrees of freedom they aim to describe. Evidence that standard methods can be adapted rather straightforwardly to systems of dynamical geometry comes from Euclidean dynamically triangulated (DT) systems. This has been demonstrated for two-dimensional

Euclidean quantum gravity (coupled to matter) where extensive computer simulations [58, 59] are in agreement with Liouville quantum gravity, which can be solved analytically. It is also true in four-dimensional DT. Although this model does not appear to describe four-dimensional quantum gravity, it *does* exhibit a well-defined scaling behavior in the so-called branched-polymer phase and finite-size scaling works very well [53].

However, since causal dynamical triangulations differ from their purely Euclidean counterparts both in their set-up and (hopefully) the continuum physics they describe, we decided to not take the validity of these methods for granted. Rather, we used several independent criteria for determining the transition orders and checked whether the results were mutually consistent.

Our analysis consisted of three parts: the histogram analysis, the measurement of the shift exponent and the analysis of Binder cumulants. The histogram analysis was sufficient to confirm the first-order nature of the A-C transition. It probes the definition of a first-order transition in a direct way and is therefore the method of choice to confirm the first-order nature of a transition. Measurements of the Binder cumulants also pointed rather unambiguously to a first-order transition. The absolute values of the minima were increasing for increasing volume N_4, contrary to what one would expect for a second-order transition. Measurements of the shift exponent, which governs the dependence of transition points on the system size, were less conclusive. The results were compatible with $\tilde{\nu} = 1$, valid at a first-order transition. However, since the stability of the curve fits under successive removal of data points with small volume was not particularly good, we did not succeed in *determining* the exponent $\tilde{\nu}$ convincingly from the data, but only in establishing its consistency with the first-order value.

The corresponding analysis of the B-C transition gave a very different picture. The histogram analysis showed a double-peak structure, but without the characteristic scaling behavior of a first-order transition. The distance between the two peaks diminishes with increasing N_4. The measurement of the shift exponent gave the result $\tilde{\nu} = 2.51(3)$ which strongly violates the prediction $\tilde{\nu} = 1$ for a first-order transition. In addition, the results of the Binder cumulant analysis were also clearly consistent with the presence of a second-order B-C transition.

In summary, we have found strong evidence that in CDT the A-C transition line is first order, while the B-C transition line is second order. The latter result is both remarkable and attractive. It opens the door to studying critical phenomena in CDT and to defining a continuum limit for vanishing lattice spacing (UV cutoff). A discussion of the effective gravitational coupling constant and its dependence on the bare couplings of the model inside phase C was initiated in [37]. In view of the results presented here, a future task will be to extend this analysis by studying the flow of this coupling constant when approaching the second-order B-C phase transition line. Apart from the intrinsic interest in understanding the behavior of quantum-gravitational observables in this limit, which appears to be associated with probing (sub-)Planckian physics, this may enable one to make a more explicit connection with the asymptotic safety scenario, as outlined recently in [48], as well as anisotropic gravity models of Hořava-Lifshitz type [38].

4

Connections between CDT and Hořava-Lifshitz gravity

This chapter is based on the following publication:

CDT meets Horava-Lifshitz gravity (with J. Ambjørn, A. Görlich, J, Jurkiewicz, R. Loll), Physics Letters B690 (2010) 413-419, arXiv:1002.3298 [hep-th]

The perturbative non-renormalizability of the theory of gravity is already apparent at the power-counting level, being essentially a consequence of the negative mass dimension of the gravitational coupling. It has been known for a long time already that adding spatial derivatives to the Einstein-Hilbert action can render the theory renormalizable. If the effects of these modifications at low energies are sufficiently weak to preserve the compatibility of such a theory with observations, the problem of non-renormalizability could be circumvented.

In order to make an action containing higher spatial derivatives Lorentz-invariant, it is necessary to add terms with higher time derivatives as well. As a consequence of the theorem of Ostrogradski, theories with higher time derivatives develop instabilities due to ghost-like kinetic terms having the wrong sign (see [60] for an extensive discussion of Ostrogradski's theorem and its consequences). Unitarity and Lorentz invariance thus appear to be mutually exclusive when higher spatial derivatives are present in the action.

In 2009 P. Hořava presented a new theory of gravity which is both renormalizable and unitary but where the requirement of Lorentz-invariance is given up at the fundamental level [38]. This modified theory could in principle still be compatible with observations if the renormalization group flow would carry it sufficiently close to general relativity at low energies. The theory and its generalizations are conventionally known under the name *Hořava-Lifshitz gravity* (see [39, 40] for reviews).

A first connection between Hořava-Lifshitz gravity (HLG) and Causal Dynamical Triangulations (CDT) has been observed in [41], where the spectral dimension has been calculated in the context of Hořava-Lifshitz gravity in arbitrary dimensions. In 3+1 dimensions it was shown that the spectral dimension acquires the value $d_s = 4$ at macroscopic scales and $d_s = 2$ in the ultraviolet regime. A qualitatively similar behavior was found numerically in the context of 3+1 dimensional CDT a few years earlier [61], a similar agreement has been shown using functional renormalization group methods in the

continuum [62].

In this chapter we show that the phase diagram of CDT in 3+1 dimensions, which we have presented in the previous chapter, shows remarkable similarities with the generic Lifshitz phase diagram underlying HLG. It is thus conceivable that a putative continuum limit of CDT can exhibit anisotropic scaling of space and time just like in HLG. The presence of a second-order phase transition line in CDT furthermore allows for the possibility that there exist both isotropic and anisotropic fixed points, such that CDT could potentially provide a unifying nonperturbative framework for anisotropic as well as isotropic theories of quantum gravity.

We begin by providing a quick review of HLG based on [39, 40] and continue in the following section by highlighting the similarities between the CDT phase diagram and the Lifshitz phase diagram. We conclude with a discussion of our findings.

4.1 A quick review of Hořava-Lifshitz gravity

The primary new ingredient of Hořava-Lifshitz gravity (HLG), namely the anisotropic scaling of space and time, can already be illustrated in the context of the much simpler theory of a Lifshitz scalar. This is a standard quantum field theory of a massive scalar field on a flat background with the action

$$S = \int \left(\dot{\phi}^2 - \phi D \phi - m^2 \phi^2 + \sum_{n=1}^{N} g_n \phi^n \right) dt d^d x,$$

where the differential operator D is given by

$$D = -c^2 \Delta + \ldots + (-\Delta)^z.$$

If $z = 1$ D is the standard Laplace operator and the theory reduces to a standard relativistic scalar field theory. Now let us assume anisotropic scaling between space and time using the engineering dimensions

$$[dx] = [\kappa]^{-1} \quad \text{and} \quad [dt] = [\kappa]^{-z}, \tag{4.1}$$

with κ being an arbitrary quantity having the dimension of momentum. As a consequence the speed of light becomes dimensionful and we get $[D] = [\kappa]^{2z}$. Therefore the operator Δ^z becomes marginal while the lower-order Δ^n operators become relevant. By using the fact that the action is dimensionless we deduce $[\phi] = [\kappa]^{(d-z)/2}$ which in turn implies $[g_n] = [\kappa]^{d+z-n(d-z)/2}$. If we choose z such that $z \geq d$ then we have $[g_n] \geq 0 \ \forall n$. We conclude that this theory is power-counting renormalizable for $z \geq d$.

In order to apply these ideas to gravity it is convenient to work within the Hamiltonian formulation of general relativity based on the ADM decomposition of spacetime,

$$ds^2 = -N^2 c^2 dt^2 + g_{ij}(dx^i - N^i dt)(dx^j - N^j dt). \tag{4.2}$$

In this framework the spacetime is foliated into a family of non-intersecting spacelike hypersurfaces with induced 3-metric g_{ij}. The scalar *lapse function N* describes how proper

time scales with respect to the coordinate time t, and the *shift vector* N^i encodes how infinitesimally adjacent spatial slices are shifted with respect to each other in the spatial directions.

The Einstein-Hilbert action of general relativity is invariant under general diffeomorphisms. Anisotropic scaling of space and time is clearly incompatible with the requirement that the gravitational action respects full diffeomorphism symmetry. Therefore we will only consider the smaller set of foliation-preserving diffeomorphisms

$$t \to t + \chi^0(t), \quad x^i \to x^i + \chi^i(t, \mathbf{x}).$$

The following ansatz for the gravitational action respects the new symmetry:

$$S = k \int \sqrt{g} N \left((K_{ij} K^{ij} - \lambda K^2) + (-\gamma R^{(3)} + 2\Lambda + V(g_{ij})) \right) d^3x dt, \tag{4.3}$$

$$K_{ij} = \frac{1}{2N}(-\dot{g}_{ij} + \nabla_i N_j + \nabla_j N_i). \tag{4.4}$$

K_{ij} is the usual extrinsic curvature of a spatial slice with respect to the spacetime in which it is embedded, $k \propto 1/G$ is an overall proportionality factor, λ is a new coupling in the kinetic part of the action, γ is a new parameter related to a relative scaling between time and the spatial directions, and $V(g_{ij})$ is a potential term. In the case $\lambda = \gamma = 1$, $V = 0$ the Einstein-Hilbert action with a cosmological term is recovered.

Now we impose anisotropic scaling of space and time according to (4.1). Taking the metric to be dimensionless and using (4.2), we find $[N] = [1]$, $[N^i] = [\kappa]^{z-1}$, $[c] = [\kappa]^{z-1}$ and $[ds] = [\kappa]^{-1}$. Furthermore from the definition of the extrinsic curvature we have $[K] = [\kappa]^z$. Those couplings g_{2z} in the potential term in the action that belong to a term containing $2z$ derivatives have the dimension $[g_{2z}] = [\kappa]^{(d-z)}$ because of the dimensionless nature of the action. In analogy to the Lifshitz scalar the theory becomes power-counting renormalizable if $z = d$ and if terms with $2z$ derivatives are included in the action.

If the potential term in the action is unrestricted, the number of terms compatible with the symmetry of the action becomes very large, making it difficult to perform computations. In the original formulation of HLG in [38] the potential term was restricted to be of the form

$$V = \frac{1}{\sqrt{g}} E^{ij} G_{ijkl} E^{kl},$$

where the quantity E^{ij} is required to follow from a variational principle,

$$\sqrt{g} E^{ij} = \frac{\delta W[g_{kl}]}{\delta g_{ij}},$$

for some action W. G_{ijkl} is the inverse of the generalized Wheeler de Witt metric

$$G^{ijkl} = \frac{1}{2}(g^{ik} g^{jl} + g^{il} g^{jk}) - \lambda g^{ij} g^{kl}. \tag{4.5}$$

This condition has been dubbed *detailed balance* and is inspired by condensed matter systems. For this particular form of the potential, the bare cosmological constant necessarily

has the wrong sign [63]. Therefore this condition is usually dropped and more general versions of HLG are considered.

Another possibility to restrict the potential is to require the lapse function to not depend on the spatial coordinate: $N = N(t)$. This version of HLG is called *projectable Hořava-Lifshitz gravity*. It turns out that the potential now depends only on the metric and its spatial derivatives. In the simplest case, by including only terms with up to six spatial derivatives, the action decomposes into an Einstein-Hilbert part and a Lorentz-violating part with eight terms. The cosmological constant turns out to be unrestricted compared to the situation with the detailed balance condition enabled. The main problem with this version of the theory is the appearance of a scalar propagating degree of freedom, which is unstable or at odds with observations at low energies [64, 65, 66]. Possible resolutions of these problems have been investigated by relaxing the projectability constraint [67], but a fully convincing solution of the scalar mode problem has not yet been achieved.

4.2 Relation between CDT and Hořava-Lifshitz gravity

A common key ingredient in both CDT and Hořava-Lifshitz gravity is a global time foliation, with the difference that in CDT this is not directly associated with a violation of diffeomorphism symmetry, since the dynamics is defined directly on the quotient space of metrics modulo diffeomorphisms. This raises the question whether new insights can be gained by analyzing and interpreting CDT quantum gravity in a generalized, anisotropic framework along the lines of Hořava-Lifshitz gravity. The reference frame until now has been a covariant one, assuming that any UV fixed point found in the CDT formulation could be identified with fixed points found in the covariant renormalization group approach, appealing to the general sparseness of fixed points.

At the same time, we have presented general arguments in favor of a reflection-positive transfer matrix in the (Euclideanized version of) CDT [68, 34]. Thus the conditions for a unitary quantum field theory at the UV fixed point are also met. The philosophy behind formulating gravity at a Lifshitz point was that unitarity in a theory of quantum gravity should be the prime requirement, rather than treating space and time on the (almost) equal footing required by special relativity. We conclude that the CDT approach not only shares the time-foliated structure of spacetime, but also the enforcement of unitarity by construction with Hořava-Lifshitz gravity.

This led us to asking whether CDT may be able to capture aspects of the latter, despite the fact that no higher-order spatial derivative terms are put in by hand in the CDT action. Some support for this idea comes from the fact that one UV result which can be compared explicitly, namely, the nontrivial value of the spectral dimension of quantum spacetime, appears to coincide in both approaches [41, 61]. Interestingly, also the renormalization group approach was able to reproduce the same finding in the case of 3+1 dimensions, after the spectral dimension had first been measured in simulations of CDT quantum gravity, a result taken at the time as possible corroboration of the equivalence between the CDT and RG approaches [69].

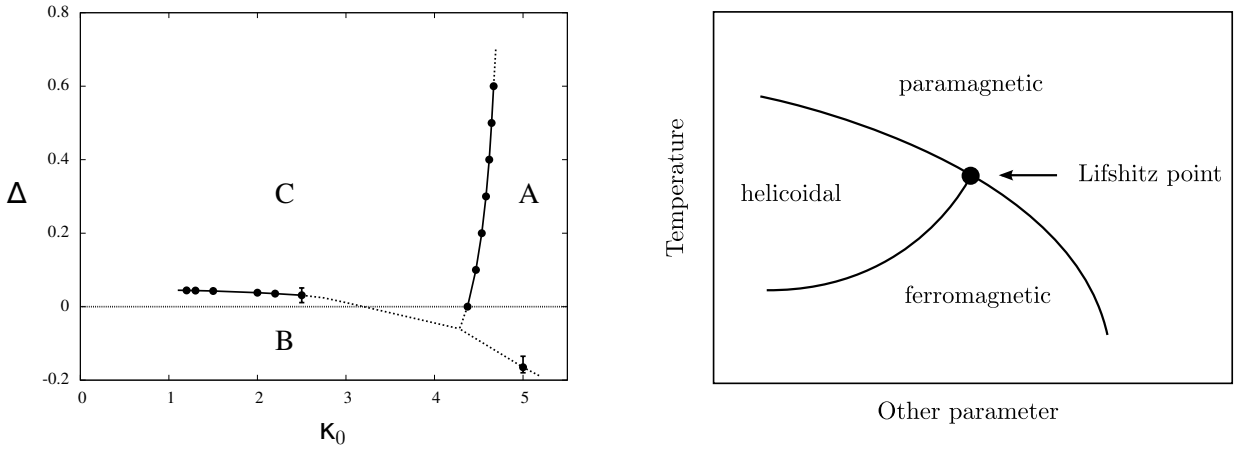


Figure 4.1: The phase diagram of CDT in 3+1 dimensions (left). A qualitative illustration of the Lifshitz phase diagram for a magnetic system, presented in [71] (right).

As we describe in the following, there exist remarkable similarities between the phase diagram of CDT and the phase diagram of theories with a Lifshitz point. Let us first recapitulate the phase structure of CDT in 3+1 dimensions, as shown on the left of Fig. 4.1. There are three phases, labeled A, B and C in [35]. In phase C, which had our main interest in [37, 70], we observed a genuine four-dimensional universe in the sense that as a function of the continuum four-volume V_4 (linearly related to the number of four-simplices), the time extent scaled as $V_4^{1/4}$ and the spatial volume as $V_4^{3/4}$. Moving into phase A, these scaling relations break down. Instead, we observe a number of small universes arranged along the time direction like "pearls on a string", if somewhat uneven in size. They can split or merge along the time direction as a function of the Monte Carlo time used in the simulations. These universes are connected by thin "necks", i.e. slices of constant integer time t_n, where the spatial S^3-universes are at or close to the smallest three-volume permitted (consisting of five tetrahedra glued together), to prevent "time" from becoming disconnected.

By contrast, phase B is characterized by the "vanishing" of the time direction, in the sense that only one spatial hypersurface has a three-volume appreciably larger than the minimal cut-off size of five just mentioned. One might be tempted to conclude that the resulting universe is three-dimensional, just lacking the time direction of the extended universe found in phase C. However, the situation is more involved; although we have a large three-volume collected at a single spatial hypersurface, the corresponding spatial universe has almost no extension. This follows from the fact (ascertained through measurement) that it is possible to get in just a few steps from one tetrahedron to any other by moving along the centres of neighboring tetrahedra or, alternatively, from one vertex to any other along a chain of links. The Hausdorff dimension is therefore quite high, and possibly infinite. Let us assume for the moment that it is indeed infinite; then the universe in phase B has neither time nor spatial extension, and there is no geometry in any classical sense.

We can now give the following qualitative characterization of the three phases in

terms of what we will provisionally call "average geometry". The universe of phase C exhibits a classical four-dimensional background geometry on large scales, such that $\langle geometry \rangle \neq 0$. One may even argue that $\langle geometry \rangle = const.$ in view of the fact that according to the mini-superspace analysis of [37, 70, 72] and allowing for a finite rescaling of the renormalized proper time, the universe can be identified with the round four-sphere, a maximally symmetric de Sitter space of constant scalar curvature. By contrast, in phase B the universe presumably has no extension or trace of classicality, corresponding to $\langle geometry \rangle = 0$. Lastly, in phase A, the geometry of the universe appears to be "oscillating" in the time direction. The three phases are separated by three phase transition lines which meet in a triple point as illustrated in Fig. 4.1 on the left.

We have chosen this particular qualitative description to match precisely that of a Lifshitz phase diagram, shown on the right of Fig. 4.1, originally presented in [71]. In an effective Lifshitz theory, the Landau free energy density $F(x)$ as function of an order parameter $\phi(x)$ takes the form[1]

$$F(x) = a_2\phi(x)^2 + a_4\phi(x)^4 + a_6\phi(x)^6 + \ldots + c_2(\partial_\alpha\phi)^2 + d_2(\partial_\beta\phi)^2 + e_2(\partial_\beta^2\phi)^2 + \ldots, \qquad (4.6)$$

where for a d-dimensional system $\alpha = m+1, \ldots, d$, $\beta = 1, \ldots, m$. Distinguishing between "α"- and "β"-directions allows one to take anisotropic behavior into account. For a usual system, $m = 0$ and a phase transition can occur when a_2 passes through zero (say, as a function of temperature). For $a_2 > 0$ we have $\phi = 0$, while for $a_2 < 0$ we have $|\phi| > 0$ (always assuming $a_4 > 0$). However, one also has a transition when anisotropy is present ($m > 0$) and d_2 passes through zero. For negative d_2 one can then have an oscillating behavior of ϕ in the m "β"-directions. Depending on the sign of a_2, the transition to this so-called modulated, helicoidal or helical phase can occur either from the phase where $\phi = 0$, or from the phase where $|\phi| > 0$. We conclude that the phases C, B, and A of CDT quantum gravity, depicted in Fig. 4.1 on the left, can be put into a one-to-one correspondence with the ferromagnetic, paramagnetic and helical phases of the Lifshitz phase diagram[2]. The triple point where the three phases meet is the so-called Lifshitz point.

The critical dimension beyond which the mean-field Lifshitz theory alluded to above is believed to be valid is $d_c = 4 + m/2$. In lower dimensions, the fluctuations play an important role and so does the number of components of the field ϕ. This does not necessarily affect the general structure of the phase diagram, but can alter the order of the transitions. Without entering into the details of the rather complex general situation, let us just mention that for $m = 1$ fluctuations will often turn the ferromagnetic-helical transition (corresponding to the A-C transition in CDT) into a first-order transition [74]. Likewise, most often the transition between the paramagnetic and the ferromagnetic phase (corresponding to the B-C transition in CDT) is of second order. In chapter 3 we have presented the results of our analysis of the CDT phase transitions, where we have found that the A-C and B-C transitions are indeed of first and second order respectively.

[1] see, for example, [73] for an introduction to the content and scope of "Landau theory"

[2] For definiteness, we are using here a "magnetic" language for the Lifshitz diagram. However, the Lifshitz diagram can also describe a variety of other systems, for instance, liquid crystals.

When we approach the B-C phase transition line from phase C by varying the coupling constant Δ, the time extent of the universe becomes smaller, when measured in units of discrete time steps (recall that we are keeping the four-volume constant). However, as argued in [75], the real time extent does not change all that much, since there is a compensating effect of "stretching" of individual four-simplices in the time direction due to an increase in α, the relative scaling between space- and time-like lattice spacings defined by $a_t^2 = \alpha a_s^2$ [34], and set by hand. (As mentioned above, for given values of κ_0 and κ_4 one can calculate Δ as a function of α.) In addition, one needs to keep in mind that the precise effective action underlying the form of the phase diagram, shown in Fig. 4.1 on the left, is truly nonperturbative, since it results from the interplay of the bare action and the measure (i.e. the entropy of the geometries as a function of N_0, $N_4^{(3,2)}$, $N_4^{(4,1)}$), and is therefore difficult to calculate. As a consequence, it may happen that the effective action changes *form* when we change κ_0 or even Δ. Since the CDT geometries, unlike their Euclidean counterparts, explicitly break the isotropy between space and time, a natural deformation could be "Hořava-Lifshitz-like". In this case a potential effective Euclidean continuum action, including the measure, and expressed in terms of standard metric variables could be of the form

$$S_H = \frac{1}{G}\int d^3x\, dt\, N\sqrt{g}\Big((K_{ij}K^{ij} - \lambda K^2) + (-\gamma R^{(3)} + 2\Lambda + V(g_{ij})\Big), \tag{4.7}$$

where K_{ij} denotes the extrinsic curvature, g_{ij} the three-metric on the spatial slices, $R^{(3)}$ the corresponding 3d scalar curvature, N the lapse function, and finally V a "potential" which in Hořava's formulation would contain higher orders of spatial derivatives, potentially rendering S_H renormalizable. The kinetic term depending on the extrinsic curvature is the most general such term which is at most second order in time derivatives and consistent with spatial diffeomorphism invariance. The parameter λ appears in the (generalized) DeWitt metric, which defines an ultralocal metric on the classical space of all three-metrics.[3] The parameter γ can be related to a relative scaling between time and spatial directions. When $\lambda = \gamma = 1$ and $V = 0$ we recover the standard (Euclidean) Einstein-Hilbert action.

Making a simple mini-superspace ansatz with compact spherical slices, which assumes homogeneity and isotropy of the spatial three-metric g_{ij}, and fixing the lapse to $N = 1$, the Euclidean action (4.7) becomes a function of the scale factor $a(t)$ (see also [76, 77, 78]), that is,

$$S_{mini} = \frac{2\pi^2}{G}\int dt\, a(t)^3\Big(3(1-3\lambda)\,\frac{\dot{a}^2}{a^2} - \gamma\,\frac{6}{a} + 2\Lambda + \tilde{V}(a)\Big). \tag{4.8}$$

The first three terms in the parentheses define the IR limit, while the potential term $\tilde{V}(a)$

[3] The value of λ governs the signature of the generalized DeWitt metric

$$G_\lambda^{ijkl} = \frac{1}{2}(g^{ik}g^{jl} + g^{il}g^{jk}) - \lambda g^{ij}g^{kl},$$

which is positive definite for $\lambda < 1/3$, indefinite for $\lambda = 1/3$ and negative definite for $\lambda > 1/3$.

contains inverse powers of the scale factor a coming from possible higher-order spatial derivative terms.

The results reported in [37, 70, 72] are compatible with the functional form of the mini-superspace action (4.8), but we were not able to determine $\tilde{V}(a)$, which could be important for small values of the scale factor. As outlined in [37], because of the nature of the dependence of the *renormalized* coupling constants on the bare parameters κ_0, Δ, resolving shorter distances seems to necessitate a closer approach to the phase transition lines, with UV behavior found along those lines. The A-C line is first order and thus not of interest. The order of the A-B line has not been determined, but since it cannot be approached from inside phase C which has good IR properties, it is currently not of interest either. This leaves the B-C transition line, which according to our analysis presented in chapter 3 is indeed of second order and thus may allow us to probe the ultraviolet regime.

One defining aspect of Hořava-Lifshitz gravity is the assumption that the scaling dimensions of space and time differ in the ultraviolet regime. This difference is used to construct a theory containing higher spatial derivatives in such a way that it is renormalizable. How would one observe such a difference in the present lattice approach? Consider a universe of time extent T, spatial extension L and total four-volume $V_4(T,L)$. By measuring T and L we can establish the mutual relations

$$T \propto V_4^{1/d_t}, \quad L \propto \left(V_4^{1-1/d_t}\right)^{1/d_s} \propto T^{(d_t-1)/d_s}.$$

Well inside phase C we measured $d_t = 4$ and $d_s = 3$, in agreement with what is expected for an ordinary four-dimensional space-time. If the dimension [T] of time was z times the dimension [L] of length we would have

$$z = \frac{d_s}{d_t - 1}.$$

We have seen that well inside phase B both d_s and d_t must be large, if not infinite. If the B-C phase transition is second order, it may happen that z goes to a value different from 1 when we approach the transition line. To investigate this possibility, we have tried to determine z as a function of the parameter Δ as $\Delta \to 0$. For $\Delta > 0.3$ one obtains convincingly $d_t \approx 4$ and $d_s \approx 3$ and thus $z \approx 1$, but for smaller Δ the quality of our results does not allow for any definite statements. Autocorrelation times seem to become very long and there may be large finite-volume effects, which obscure the measurements.

4.3 Discussion

We have shown that the CDT phase diagram bears a striking resemblance to a Lifshitz phase diagram if we identify "average geometry" with the Lifshitz field ϕ in the heuristic sense discussed above. In our earlier papers, we tentatively interpreted phase A as being dominated by the wrong-sign kinetic term of the conformal factor of the continuum Euclidean Einstein-Hilbert action. It appears to be a Lorentzian remnant of a degeneracy

found in the old Euclidean quantum gravity model based on (Euclidean) dynamical triangulations, which for large values of κ_0 exhibits a branched-polymer phase, likewise interpreted as caused by the dominance of the conformal factor (see, for example, [28, 53]).

Eq. (4.6) strongly suggests an identification of the Lifshitz field ϕ with the conformal factor or some function thereof, such that the transition from phase C to phase A in the Lifshitz diagram is associated with a sign swap of the corresponding kinetic term from negative to positive. An effective action for the conformal mode coming out of a nonperturbative gravitational path integral would consist of (i) a contribution from the bare action (where the kinetic conformal term has the "wrong", negative sign), (ii) a contribution from the measure, and (iii) contributions from integrating out other field components and, where applicable, other matter fields. It has been argued previously that the Faddeev-Popov determinants contributing to the gravitational path integral after gauge-fixing contribute effectively to the conformal kinetic term with the opposite, positive sign [79, 80]. For example, when working in proper-time gauge, to imitate the time-slicing of CDT, Euclidean metrics can be decomposed according to[4]

$$ds^2 = d\tau^2 + \mathrm{e}^{2\phi(\tau,x)} g_{ij}(\tau,x) dx^i dx^j, \tag{4.9}$$

giving rise to a term $-1/G^{(b)}\mathrm{e}^{3\phi}\sqrt{\det g}(\partial_\tau\phi)^2$ in the bare gravity Lagrangian density, with $G^{(b)}$ being the bare Newton's constant. According to [80], one expects that the leading contribution from the associated Faddeev-Popov determinant has the same functional form, but with a plus instead of a minus sign, and with a different dependence on $G^{(b)}$. The presence of contributions of opposite sign to the effective action for the conformal mode $\phi(\tau,x)$ can therefore lead to two different behaviors, depending on the value of $G^{(b)}$. Identifying the κ_0 of our lattice formulation – an a priori freely specifiable parameter – with the inverse of $G^{(b)}$, this mechanism can account exactly for the observed behavior with regard to the transition between phases A and C.[5]

Engaging in a bit of speculation, an interesting possibility would be if the triple point in the CDT phase diagram corresponded to an asymmetric scaling between space and time, like in the Hořava model, while the end point of the B-C line represented an isotropic point associated with the RG asymptotic safety picture. Deciding which scenario is actually realized will require substantially longer simulations or improved updating algorithms, something we are working on presently. – The analysis of the phase structure in the framework of causal dynamical triangulations we have performed here may turn out to provide a universal template for understanding nonperturbative theories of higher-dimensional, dynamical geometry, including "true" quantum gravity.

[4]The conformal decomposition of the spatial three-metric is essentially unique if one requires g_{ij} to have constant scalar curvature.

[5]Related mechanisms have earlier been considered in the context of purely Euclidean dynamical triangulations by J. Smit [81].

5

Locally causal DT in 2+1 dimensions: Theory

The original model of dynamical triangulations was based on a Euclidean path integral. Thus, its regularized configuration space consisted of all triangulations composed of flat building blocks, where the metric in the interior has Euclidean signature. It was hoped that such a model, formulated in four dimensions, could serve as a quantum theory of gravity. However, numerical investigations showed that the model does not have an infrared limit which is compatible with observations [28, 29, 30]. In addition, it was found that the model does not allow a continuum limit to be taken [31, 32].

Causal dynamical triangulations (CDT) have been introduced in 1998 to address these problems [33]. The central new idea in CDT is to incorporate causality into the model. The configuration space now contains all triangulations which consist of flat pieces of Minkowski space and are equipped with a distinguished foliation with a fixed number of spatial slices. Numerical simulations in 3+1 dimensions showed that these modifications lead to a different dynamics compared to the older Euclidean triangulations. In particular, CDT in 3+1 dimensions has been shown to contain a phase of geometry with extended four-dimensional geometries, and a close connection between the infrared sector of CDT and de Sitter cosmology in the continuum could be established [34, 35, 37]. Furthermore, as we have shown in chapter 3, CDT possesses a second-order phase transition, which is a necessary requirement for the model to possess a continuum limit.

The particular way in which CDT incorporates causality is not unique and it is worthwhile to consider other possibilities. In this chapter we present a new model of dynamical triangulations which implements a sum over all Lorentzian triangulations. We use the term 'Lorentzian triangulation' in the sense that the simplicial manifold, constructed using flat pieces of Minkowski space, should admit a Lorentzian metric. This is trivially satisfied inside the building blocks. If these building blocks were glued to each other in a random way, light cone defects would typically occur at the boundaries of the building blocks. We therefore require the triangulation to be *locally causal* in the sense that at every point in the triangulation there exists a well-defined light cone. Local causality at a particular point in the triangulation ensures that the flow of time is well-defined in a local neighborhood of that point and we additionally require that the flow of time can be consistently extended to the whole triangulation. In the continuum, the existence of

a non-vanishing vector field is a necessary and sufficient requirement for a manifold to admit a Lorentzian metric [82]. Our second condition for the new DT model can be seen as a discrete analog of the continuum condition.

Our definition of the new model does at least in principle not exclude the presence of closed timelike curves. However, we also restrict ourselves to a fixed topology for all the triangulations in the configuration space, just like it is done in CDT. The particular choice of topology may prevent closed timelike curves from appearing. This is indeed the case for the numerical simulations in 2+1 dimensions, which we have performed and where we have chosen the 3-sphere as spacetime topology. Our numerical analysis will be discussed in chapter 6.

The most prominent difference between CDT and the new model, which we call *locally causal dynamical triangulations*, concerns the presence and absence respectively of a *distinguished* foliation. The spatial structure of a triangulation in CDT, consisting of all spacelike subsimplices of codimension one, forms a foliation with fixed number of leaves. By labeling the leaves with consecutive integers we get a time variable for free. It allows us to define a propagator which connects a spatial triangulation at time t with another spatial triangulation at time $t+1$. In 1+1 dimensions the propagator has been calculated analytically [33]. An additional merit of having such a time variable is that it allows us to construct observables like the volume profile, which measure the distribution of spatial volume as a function of time. The analysis of these volume profiles in 3+1 dimensions has been crucial to establish the connection between the infrared sector CDT and the de Sitter cosmology in the continuum [37].

In the locally causal DT model, typical triangulations have a more complicated spatial structure, therefore we do not have a similar distinguished time variable at our disposal. As we shall demonstrate in chapter 6, it is nevertheless possible to construct a meaningful time variable, which in the special case of CDT-compatible triangulations agrees with the canonical definition of time. In chapter 6 we show that the construction of a volume profile observable is indeed possible and we use this observable to compare the dynamics of CDT and the locally causal DT model using numerical simulations in 2+1 dimensions.

It is also interesting to compare CDT and the locally causal DT model with respect to 'background independence'. It appears to be difficult to make this term precise and there does not seem to exist a universally accepted definition [83]. With this remark of caution in mind, we can nevertheless try to make some meaningful statements which apply to our situation. Anderson has attempted to define a background-independent theory as one without 'absolute structures' [84]. Intuitively speaking, absolute structures are entities which influence the physical behavior of a system but are not being influenced by this behavior. It is difficult to make a clear separation between these absolute structures and dynamical objects, because the former can be disguised as the latter. In this context it is natural to ask whether the distinguished foliation of CDT qualifies as an absolute structure. It is indeed unchangeable, and all triangulations share the same foliation in the sense that we can superimpose all the triangulations of the ensemble by virtue of the canonical time variable. On the other hand it is not a priori clear whether the distinguished foliation affects the dynamics in the sense that its absence leads to a different

dynamics compared to CDT. It it does, then we can say that the locally causal DT model implements a stronger version of background independence compared to CDT. Our analysis presented in this and the following chapter thus aims to clarify the situation.

We have decided to focus our view on the 2+1 dimensional model for several reasons. Experience from CDT indicates that the study of the dynamics of large-scale properties in 2+1 dimensions may already be sufficient to get a good estimate of the behavior of the 3+1 dimensional model [45, 85]. Furthermore, as we will describe in chapter 7, the locally causal DT model requires the implementation of new Monte Carlo moves which are significantly more difficult to implement compared to the generalized Pachner moves used in CDT. This makes the writing of a simulation software for the new model very challenging already in 2+1 dimensions. At the same time, the increased complexity of the simulation software leads to longer simulation times compared to CDT. It is currently not even clear whether simulations of the new model in 3+1 dimensions could be performed with acceptable running times using contemporary simulation hardware.

Before we embark on exploring the locally causal DT model we briefly mention other work in which the relaxation of the foliation condition in CDT has been addressed. A 'soft' way of relaxing the foliation has been studied in [86, 87], where the timelike links have been allowed to have a varying length under certain conditions. In this approach the foliation is still there, since the connectivity of the underlying graph is unchanged, but the individual leaves of the foliation are not placed at equidistant intervals. A more radical proposal has been made in [87], where the spatial sections in CDT are allowed to be 'punctured' using new building blocks which connect more than two spatial sections at once. This work can be seen as an early precursor of our work which we will describe in the following chapters. The main difference is that the model described in [87] does not satisfy local causality.

In the present chapter we present the theoretical aspects of the locally causal DT model while the following two chapters are devoted to the numerical results and the implementation of the model. We begin with a compact description of the model in 1+1 dimensions before we move on to discuss the kinematical basics of the 2+1 dimensional model. Afterwards we discuss the new kinematical structures appearing in the new model, followed by a discussion of kinematical constraints. We conclude with a detailed analysis of actions and the Wick rotation.

5.1 A quick look at locally causal DT in 1+1 dimensions

We begin by choosing our elementary flat building blocks. Figure 5.1 shows the four types of triangles which can be built using spacelike and timelike links. Here and in the following we will assume that the squared lengths of all links of a specific type are equal. We use the notation $\ell_t^2 = -\alpha\ell_s^2$ where ℓ_s^2 and ℓ_t^2 are the squared lengths of the spacelike and timelike links respectively. The proportionality factor α is called the 'asymmetry parameter'. By calculating the metric inside the triangles one finds that there are two triangles with Lorentzian signature (-+), the ones of type sst and stt, where s stands for

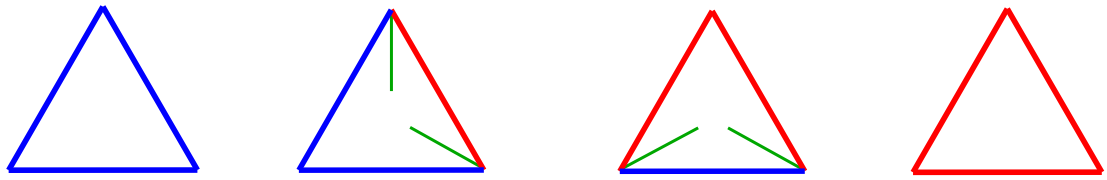


Figure 5.1: The four types of triangles which can be built from two link types, each type having a fixed length. Spacelike links are drawn in blue, timelike links in red. The green lines inside the Lorentzian triangles visualize light rays which connect to the vertices.

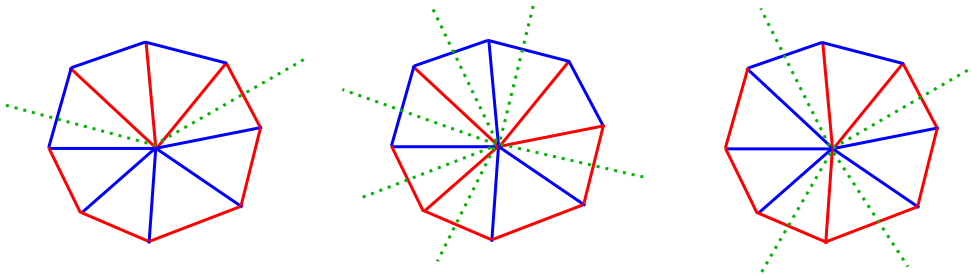


Figure 5.2: Three vertex neighborhoods in 1+1 dimensions. The dotted lines represent light cone crossings. Only the rightmost situation is locally causal with precisely four light cone crossings.

a spacelike edge and t for a timelike edge. This statement holds for all possible values of ℓ_s and ℓ_t. Since we aim to build a model of dynamical triangulations which respects local causality we are restricted to only using triangles with Lorentzian signature. We note that in CDT only the triangle of type stt is used.

The use of Lorentzian building blocks is not a sufficient condition for the triangulation to be locally causal. Violations of local causality can occur at the vertices of the triangulation. To see this we need to have more information about the light cone structure close to the vertices of the Lorentzian triangles. This is conveniently achieved by calculating all the angles which are complex numbers of the form $n \cdot \pi/2 + i \cdot x$ $(-\infty < x < \infty)$ where n gives the number of light cone crossings that occur when circling around the vertex from one side of the angle to the other side [88]. Figure 5.1 visualizes the light cone structure of both Lorentzian triangles.

After gluing the Lorentzian triangles together we can have a closer look at the neighborhood of individual vertices. Figure 5.2 shows three examples of such neighborhoods. If we make a full rotation around each of the three vertices we encounter a different number of light cone crossings. It is now clear that in order to have a locally causal triangulation we have to require that the number of light cone crossings around every vertex has to be exactly four.

Looking at the two Lorentzian building blocks in Fig. 5.1 we see that all angles whose sides are of the same type do not have a light cone in between them while the other angles have exactly one light cone in between them. The local causality constraint therefore translates into the condition that when we circle around a vertex, the type of the radial

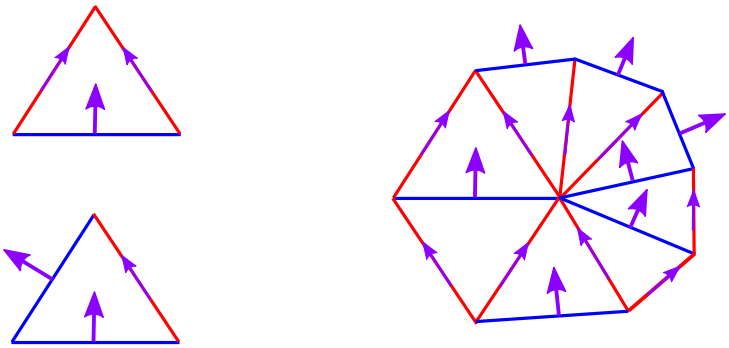


Figure 5.3: The two Lorentzian triangles with chosen time orientation as explained in the text (left). The time orientation of a single link determines the time orientation of all other links in the surrounding (right).

links must change exactly four times from timelike to spacelike or vice versa. In particular, in 1+1 dimensions we always have two groups of radial spacelike links which are located in two opposite sections of the light cone.

In CDT the flow of time is very easy to understand since the spatial sections are labeled by a discrete time coordinate. For locally causal triangulations we have to proceed in a different way to keep track of the overall time direction. We add a time orientation to a timelike link by choosing one of its vertices as the past vertex and we add a time orientation to a spacelike link by choosing one of the two triangles which contain the link as the past triangle. It is then easy to add a time orientation to a Lorentzian triangle by time-orienting one of its links. The light cone structure inside the triangle then uniquely defines the time orientation of the other two links. The left part of Fig. 5.3 shows both Lorentzian triangles with a specific choice of time orientation.

Given a time-oriented Lorentzian triangle the time orientation of every neighbor triangle is also uniquely defined by consistency. If the local causality constraint is satisfied one can orient the neighborhood of a vertex as shown in Fig. 5.3 on the right. It is easy to build examples of vertex neighborhoods which violate local causality and where it is impossible to consistently define a time orientation. The leftmost vertex neighborhood in Fig. 5.2 is such an example. Even if the local causality constraint is satisfied it is not always possible to extend the time orientation to the full triangulation. For example the process will in general fail on nonorientable manifolds. If the full triangulation cannot be time-oriented consistently, it clearly does not admit a Lorentzian metric. Therefore, in addition to the local causality constraint, we require as a second condition that the full triangulation can be time-oriented consistently.

A standard way to enumerate all the triangles in a given triangulation is to use the well-known breadth-first traversal algorithm. One starts from some initial triangle and visits all its neighbors. For each of these neighbors one visits those of their neighbors which have not yet been visited. The process is iterated until all triangles have been visited. Figure 5.4 illustrates this process on the left. We can use the breadth-first traversal algorithm to time-orient a triangulation. We begin by assigning one of the two possible

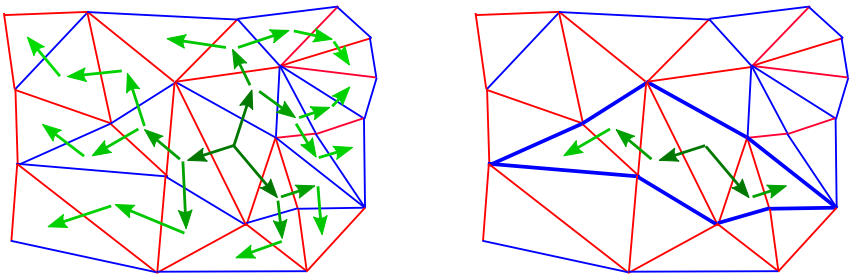


Figure 5.4: The breadth-first traversal algorithm to enumerate all triangles in a triangulation (left). When imposing the additional constraint of never crossing spacelike links, the set of all triangles that is reached by the process forms a bubble (right).

Figure 5.5: A source and a sink of time.

time orientations to the initial triangle. The time orientations of the neighbor triangles are uniquely defined by requiring the time orientation of the shared link to be consistent. Thus we time-orient the neighbors and move to the next triangles according to the breadth-first traversal algorithm. Whenever a new triangle is being time-oriented it can happen that some of its links already have a time orientation induced by the time orientation of one of the neighbor triangles. We then check whether the time orientation of the new triangle is consistent with the time orientations of those links that are already time-oriented. If this is not the case, the triangulation cannot be time-oriented.

We can slightly relax the local causality constraint if we allow for the presence of a limited number of 'sources' and 'sinks' of time. Consider all timelike links emanating from a particular vertex. If the time orientation of all these links points outwards we say that the vertex is a source of time. If they point inwards the vertex is a sink of time. Figure 5.5 illustrates both cases. These sources and sinks of time are useful to define certain boundary conditions. In chapter 6 we will discuss simulations of the locally causal DT model in 2+1 dimensions using the 3-sphere topology. There we place a source and a sink of time at the two poles of the 3-sphere.

Bubbles and bubble graphs

Consider again the breadth-first traversal algorithm illustrated in Fig. 5.4 on the left. We can define a modified version of the algorithm by imposing the additional constraint that one must never cross a spacelike link. This process is shown on the right side of the fig-

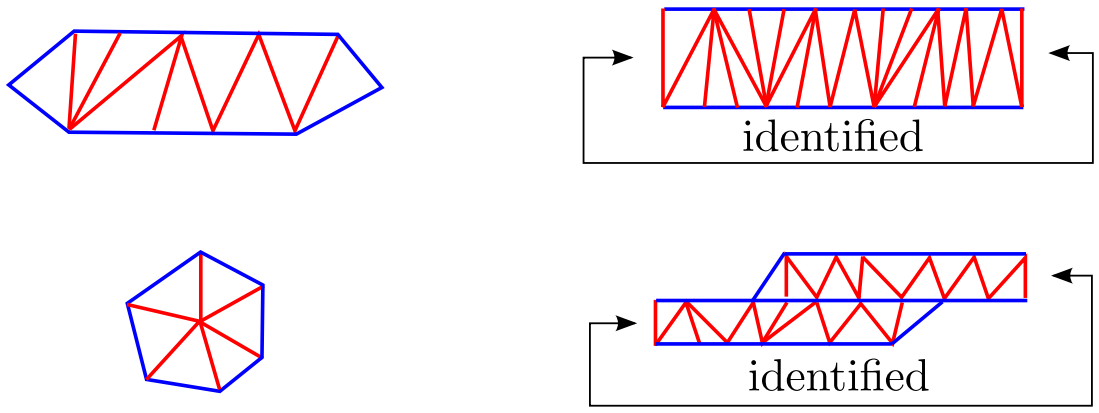


Figure 5.6: Various types of bubbles, which occur in 1+1 dimensional locally causal triangulations without boundary.

ure. We call the set of triangles which is reached by the second process a *bubble*. There exist various types of bubbles as illustrated in Fig. 5.6. We have only included those bubbles which appear in triangulations without boundary. If boundaries are present, there will be other types, which can be obtained by making cuts through some of the bubbles shown in the figure. On the left top we see a bubble consisting of a strip of stt triangles ending on both ends in a sst triangle. The right top bubble is nothing but the familiar strip of unit time extension known from CDT which is bounded by two disconnected spatial sections. On the left bottom we see a bubble with a source or sink of time in the center. A bubble can even overlap itself as shown at the right bottom. This last example demonstrates that bubbles can indeed contain spacelike links which are not located on the boundary of the bubble. Note that both strips shown on the right of the figure are always global in the sense that if one such strip is removed from the triangulation, the remaining sub-triangulation consists of two disconnected parts (assuming non-periodic boundary conditions in time direction).

It is clear that every triangle belongs to a single bubble and therefore the full triangulation can be seen as being composed of bubbles glued to each other. It is thus natural to visualize the set of bubbles as a graph where nodes represent bubbles and edges the interfaces between bubbles. If we time-orient the triangulation, the graph becomes a directed graph where the direction of each edge indicates the flow of time between the two bubbles which are connected by the edge. We call this directed graph a *bubble graph*. Figure 5.7 shows examples of bubble graphs for a CDT-compatible triangulation and for a triangulation which does not have a distinguished foliation. In chapter 7 we will use bubble graphs to implement one of the Monte Carlo moves for the 2+1 dimensional model.

The bubble labeled with 5 is a self-overlapping bubble which means that it is its own neighbor. The bubble graph thus contains a cycle of length one. By inserting bubbles at the location where the bubble labeled with 5 meets itself we can easily create cycles of greater length. It is clear that such cycles in the bubble graph do not imply that closed timelike curves exist.

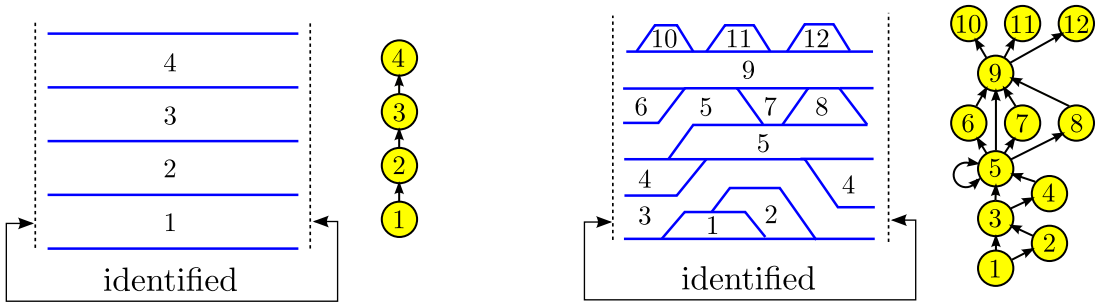


Figure 5.7: Examples of triangulations and their associated bubble graphs. Only spacelike links are drawn in blue. The left-hand side shows a CDT-like triangulation, the right-hand side shows a triangulation which does not have a distinguished foliation.

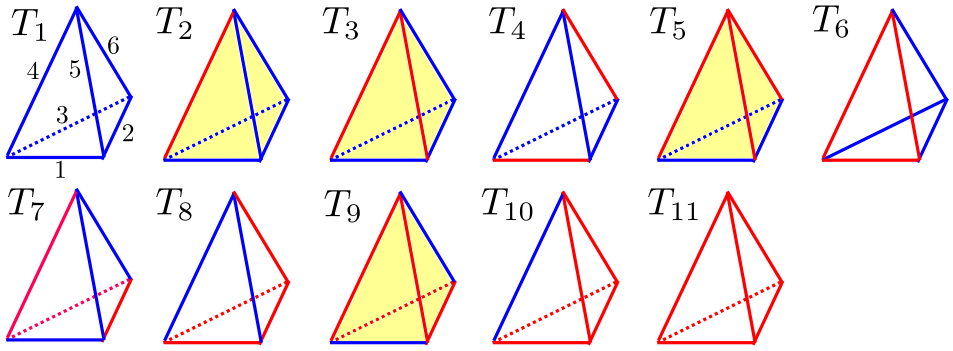


Figure 5.8: All tetrahedra in 2+1 dimensions, built from timelike and spacelike links with fixed squared lengths, allowing for all signatures. The tetrahedra highlighted in yellow have Lorentzian signature, the signature of the T_7 tetrahedron is Lorentzian only for $\alpha < 1$. The labeling shown at the first tetrahedron is the same for all the other tetrahedra.

5.2 Kinematics in 2+1 dimensions

In order to build a model of locally causal dynamical triangulations in 2+1 dimensions we first need to determine the flat building blocks which have the correct Lorentzian signature (-++). We use the notation (-++) to make clear that we are only interested in geometries with one timelike and two spacelike directions. Figure 5.8 shows all types of tetrahedra which can be built using spacelike and timelike links with fixed squared lengths. We allow the squared lengths of spacelike and timelike links to be different: $\ell_t^2 = -\alpha \ell_s^2$. The asymmetry parameter $\alpha > 0$ conveniently parametrizes the amount of anisotropy between space and time. Throughout the rest of the document we will set $\ell_s = 1$. By calculating the metric in the interior of each tetrahedron in Fig. 5.8 we find that the tetrahedra of type T_2, T_3, T_5 and T_9 have Lorentzian (-++) signature and that the signature of the tetrahedron of type T_7 depends on the asymmetry parameter, it is Lorentzian (-++) only for $\alpha < 1$. We note that in CDT only the tetrahedra T_5 and T_9 are used.

Model	Allowed building blocks	Constraint on α
Type 1	T_2, T_3, T_5, T_9	$1/2 < \alpha < 3$
Type 2	T_2, T_3, T_5, T_7, T_9	$1/2 < \alpha < 1$

Table 5.1: The two models of causal triangulations. The remainder of the article discusses the Type 1 model.

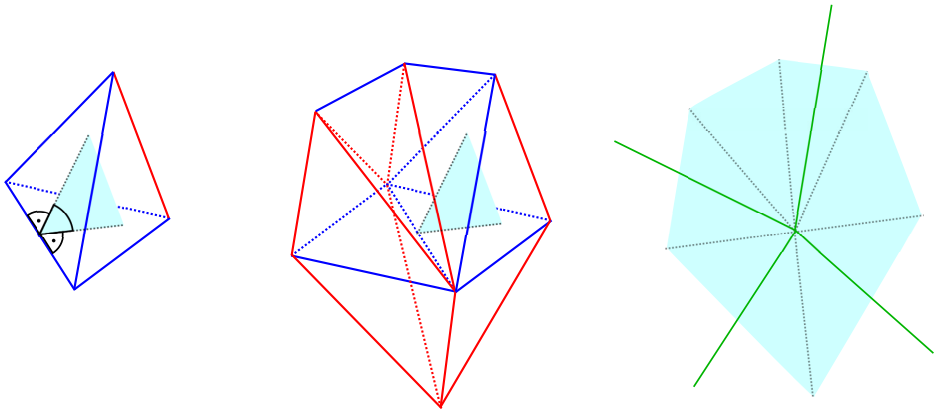


Figure 5.9: On the left, an illustration of a dihedral angle in a tetrahedron is shown. In the center we have the star of the link which is associated to the dihedral angle. The dihedral angles of all tetrahedra in the link star determine the light cones around the link, as shown on the right-hand side. The green lines represent light cone crossings.

The asymmetry parameter α spans one direction in the parameter space of the regularized model. We will see later that the existence of a well-defined Wick rotation implies the condition $1/2 < \alpha < 3$. We can thus define two sub-models of locally causal dynamical triangulations with different parameter spaces. One model contains only the tetrahedra of type T_2, T_3, T_5 and T_9 and we have $1/2 < \alpha < 3$. The second model additionally contains the tetrahedron of type T_7 and the constraint on the asymmetry parameter is $1/2 < \alpha < 1$. We denote these two models by Type 1 and Type 2 respectively. See table 5.1 for a compact summary of our findings. Both models differ substantially in their kinematical properties. In this document we discuss the Type 1 model and leave the thorough analysis of the Type 2 model for future research.

In 2+1 dimensions violations of local causality can occur at the links and possibly at the vertices of the triangulation. To understand the light cone structure around a link we determine the dihedral angles in a tetrahedron. For every link we consider the plane which is orthogonal to the link and which goes through the link center. The intersection of that plane with the tetrahedron then defines the dihedral angle associated to the link as shown in Fig. 5.9 on the left. We use the term 'Euclidean dihedral angle' or 'Lorentzian dihedral angle', depending on the signature of the metric inside the plane. For each link

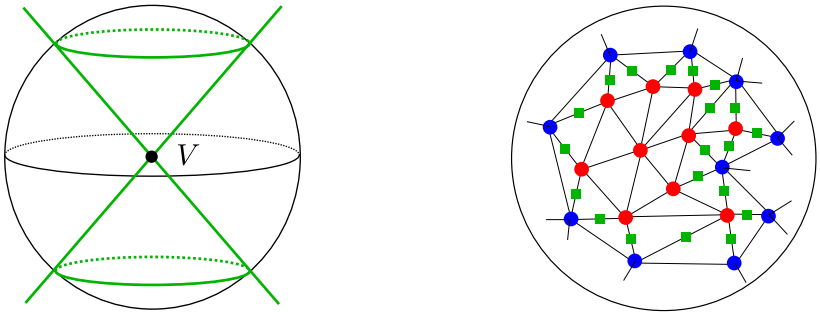


Figure 5.10: If we intersect the boundary of a vertex neighborhood with the light cone associated to the vertex we get two disconnected circles, as shown on the left-hand side. The right-hand side displays a part of the surface triangulation of the vertex neighborhood, showing timelike radial links as red dots, spacelike ones as blue dots and light cone crossings as green dots.

we then consider its star consisting of all tetrahedra which contain the link, as shown in the center of the figure. The plane segments spanned by all the dihedral angles associated to the chosen link form a new plane shown on the right side of the figure. By circling around the center of that plane we effectively circle around the link and can thus study the light cone structure around the link in a two-dimensional setting.

There are two cases to be distinguished here. If the link in the center of the star is timelike, the metric inside the plane has Euclidean signature and we do not get any causality conditions. If on the other hand the link is spacelike, the plane is Lorentzian. In this case we need to calculate all the dihedral angles associated to the link to determine the number of light cone crossings which are encountered by circling around the link. Just like in the 1+1 dimensional case the angles are of the form $n \cdot \pi/2 + i \cdot x$ $(-\infty < x < \infty)$ where n gives us the desired number of light cone crossings. We require that the number of these crossings is exactly four and say that the triangulation satisfies *link causality* if the condition is satisfied for every spacelike link.

Link causality guarantees that we have regular light cones everywhere in the triangulation except possibly at the vertices. If we intersect the light cone of a vertex V with the surface of its neighborhood we get two disconnected (topological) circles if and only if local causality holds at V, see Fig. 5.10 on the left. Let us now position ourselves above the north pole of the unit ball and look down onto the upper hemisphere, shown on the right of the figure. We see radial links connected to V which can be timelike (red dots) or spacelike (blue dots). Going back to Fig. 5.1, we see that all intersection points of lightlike rays with the surface triangulation are found on links that connect red dots with blue dots. These intersection points are visualized as green dots in the figure. If we cut all links which contain intersections of lightlike rays, the surface triangulation breaks up into a number of connected components. If we have two such components containing red vertices and one component containing blue vertices, we say that *vertex causality* holds at V. If the condition is true for all vertices, we say that the triangulation satisfies vertex causality.

Tetrahedron	Links	$\cos(\Theta)$	$\sin(\Theta)$	Lightcone crossings
T_2	1,3,5,6	$\frac{i}{\sqrt{3}}\sqrt{\frac{\alpha}{4+\alpha}}$	$\frac{2}{\sqrt{3}}\frac{\sqrt{3+\alpha}}{\sqrt{4+\alpha}}$	1
	2	$1+\frac{2\alpha}{3}$	$\frac{2i\sqrt{\alpha(3+\alpha)}}{3}$	0
	4	$\frac{2+\alpha}{4+\alpha}$	$\frac{2\sqrt{3+\alpha}}{4+\alpha}$	–
T_3	1	$\frac{i(1+2\alpha)}{\sqrt{3+12\alpha}}$	$2\sqrt{\frac{1+\alpha(4+\alpha)}{3+12\alpha}}$	1
	2,3	$\frac{2+\alpha}{\sqrt{3}\sqrt{-\alpha(4+\alpha)}}$	$\frac{2}{\sqrt{3}}\sqrt{\frac{1+\alpha(4+\alpha)}{\alpha(4+\alpha)}}$	1
	4,5	$\frac{1}{\sqrt{17+\frac{4}{\alpha}+4\alpha}}$	$\frac{2}{\sqrt{4+\frac{\alpha}{1+\alpha(4+\alpha)}}}$	–
	6	$\frac{2+\alpha(4+\alpha)}{\alpha(4+\alpha)}$	$-\frac{2i\sqrt{1+\alpha(4+\alpha)}}{\alpha(4+\alpha)}$	0
T_5	1,2,3	$-\frac{i}{\sqrt{3}\sqrt{1+4\alpha}}$	$\frac{2}{\sqrt{3}}\frac{\sqrt{1+3\alpha}}{\sqrt{1+4\alpha}}$	1
	4,5,6	$\frac{1+2\alpha}{1+4\alpha}$	$\frac{2\sqrt{\alpha(1+3\alpha)}}{1+4\alpha}$	–
T_7	1,6	$\frac{2+3\alpha}{\sqrt{\alpha(4+\alpha)(1+4\alpha)}}$	$-\frac{2\sqrt{(-1+\alpha)(1+\alpha(3+\alpha))}}{\sqrt{\alpha(4+\alpha)(1+4\alpha)}}$	0
	2,4	$\frac{3+2\alpha}{\sqrt{17+\frac{4}{\alpha}+4\alpha}}$	$2\sqrt{-\frac{(-1+\alpha)(1+\alpha(3+\alpha))}{(4+\alpha)(1+4\alpha)}}$	–
	3	$\frac{-1-2\alpha(1+\alpha)}{1+4\alpha}$	$\frac{2\alpha\sqrt{2+\frac{1}{a}-2\alpha-\alpha^2}}{1+4\alpha}$	–
	5	$-\frac{2+\alpha(2+\alpha)}{\alpha(4+\alpha)}$	$-\frac{2\sqrt{(-1+\alpha)(1+\alpha(3+\alpha))}}{\alpha(4+\alpha)}$	2
T_9	1,6	$\frac{3+4\alpha}{1+4\alpha}$	$-\frac{2i\sqrt{2+4\alpha}}{1+4\alpha}$	0
	2,3,4,5	$-\frac{1}{1+4\alpha}$	$\frac{2\sqrt{2}\sqrt{\alpha(1+2\alpha)}}{1+4\alpha}$	–

Table 5.2: Dihedral angles for all tetrahedra of the Type 1 and Type 2 model. The link numbers refer to the numbering shown in Fig. 5.8. For Lorentzian angles the number of light cone crossings is shown.

We have not found any Monte Carlo moves which destroy vertex causality but maintain link causality. Also, we have not been able to explicitly construct any triangulation which satisfies link causality but not vertex causality. Thus it may be possible that link causality actually implies vertex causality, but we currently do not have a proof of this conjecture.

We will now explore the consequences of local causality. For that, we need to know for all Lorentzian dihedral angles how many light cone crossings they contain. This information is displayed in table 5.2, where we have also included the analytic expressions for the cosine and sine of the angles because they will be needed later to compute the action. By analyzing the information in the table we can reformulate the local causality

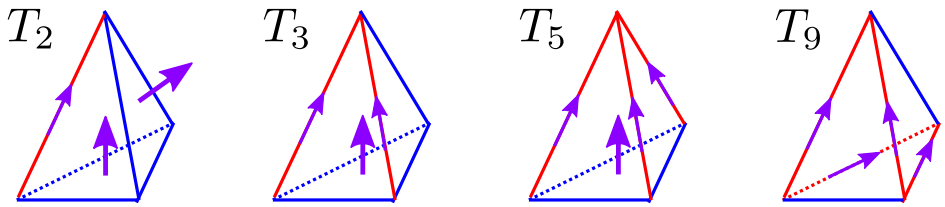


Figure 5.11: The four tetrahedra of the Type 1 model equipped with a time orientation.

condition for the Type 1 model as follows.

If we circle around a spacelike link we encounter a sequence of triangles which contain the link. To every dihedral angle we naturally associate two of these triangles, namely, those which belong to the same tetrahedron as the angle. With the help of table 5.2 we find that whenever exactly one of the two triangles is spacelike, the angle contains one light cone crossing, otherwise it contains none. Thus we encounter crossings whenever the type of triangle changes from spacelike to non-spacelike. The local causality constraint therefore implies that the type of triangle changes exactly four times between spacelike and non-spacelike when we circle around a spacelike link.

We proceed as in 1+1 dimensions to study how to time-orient a triangulation. We add a time orientation to a timelike link by choosing one of its vertices as the past vertex and we add a time orientation to a spacelike triangle by choosing one of the two tetrahedra which contain the triangle as the past tetrahedron. The time orientation of one timelike link or of one spacelike triangle of a chosen tetrahedron determines the time orientation of all other timelike links and spacelike triangles of the same tetrahedron. Thus we end up with a time orientation for a single tetrahedron as shown in Fig. 5.11 for all tetrahedra of the Type 1 model. Just like in 1+1 dimensions the local causality constraint does not guarantee that the time orientation can be extended to the full triangulation. Therefore we require in addition to the local causality condition that the triangulation can be time-oriented consistently. In section 5.1 we have described an algorithm to time-orient a 1+1 dimensional triangulation. This algorithm can easily be generalized to time-orient a 2+1 dimensional triangulation.

We conclude this section with the remark that sources and sinks of time can be defined in a completely analogous way as in 1+1 dimensions. As already mentioned in section 5.1, we will use these sources and sinks of time in chapter 6 to implement particular boundary conditions. In particular we will use the 3-sphere topology and place a source and sink of time at the two poles of the 3-sphere.

5.3 Bubbles, pinchings and rings

We define a *bubble* using the modified breadth-first traversal algorithm which we described in section 5.1 and illustrated in Fig. 5.4 for the 1+1 dimensional case. As an intuitive picture one can imagine inserting a source of water at a particular tetrahedron and letting the water flow out. If we picture spacelike triangles as walls which water cannot

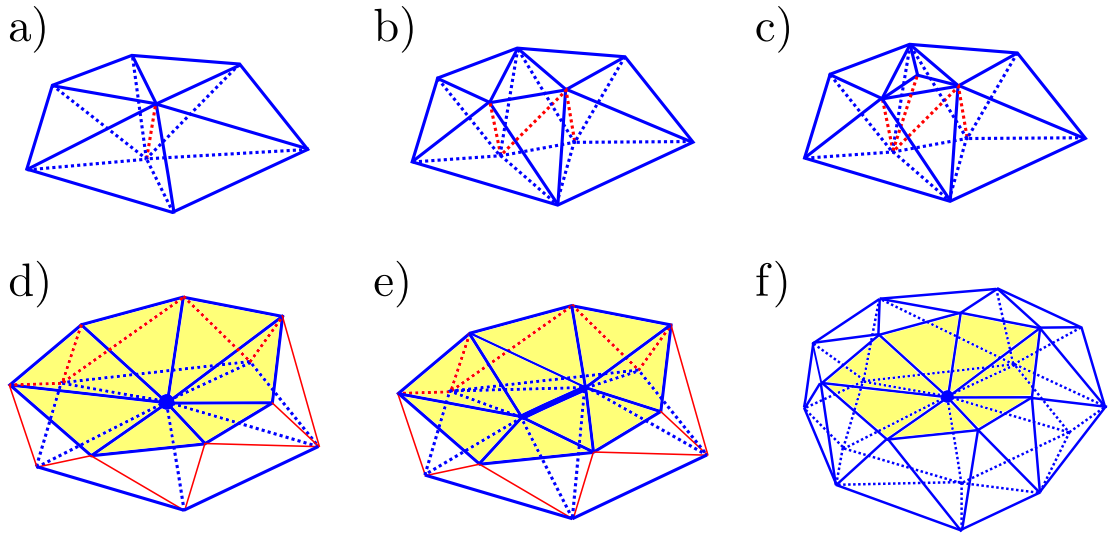


Figure 5.12: a) A ring of tetrahedra of type T_2 forming a simple bubble. b) By inserting tetrahedra of type T_3 we can form more complicated bubbles. c) More general bubbles also contain other tetrahedra types. d) A ring of tetrahedra of type T_3 forms a pinching where two spatial discs meet in a single vertex. e) By adding two tetrahedra of type T_2 we get the situation where two spatial sections meet in a link. f) Bubbles and pinchings can occur in combination (timelike links are omitted here).

penetrate, the water will finally be contained in a subset of the triangulation which we call a bubble. The whole triangulation can be seen as being composed of bubbles glued to each other and can thus be displayed using bubble graphs which we introduced in section 5.1 and visualized in Fig. 5.7 for the 1+1 dimensional case.

By having a closer look at the four types of tetrahedra of the Type 1 model (see Fig. 5.11) we see that only the tetrahedra of type T_2 and T_3 contain triangles with exactly two spacelike edges. Furthermore, we find that both tetrahedra contain exactly two of these triangles. If we glue two of these tetrahedra along such a triangle, the resulting simplicial complex will again have two of these triangles on its boundary. By iterating the gluing procedure we end up having a ring of tetrahedra of type T_2 and T_3. In triangulations without boundary, the set of all tetrahedra of type T_2 and T_3 thus organizes itself in a collection of rings. In triangulations with boundary partial rings are also possible.

If the ring only contains tetrahedra of type T_2 we get a simple bubble as shown in Fig. 5.12 a). It consists of two spatial discs with identical structure and a timelike link in the interior. By collapsing this link the two discs get identified and the bubble thus collapses to a disc. This is one of the Monte Carlo moves which we will describe in section 7.3.2. We can form more complicated bubbles by inserting tetrahedra of type T_3 into the ring of tetrahedra of type T_2, as displayed in Fig. 5.12 b). Figure 5.12 c) shows a bubble which not only contains a ring of tetrahedra of types T_2 and T_3 but also contains a tetrahedron of type T_5 which is enclosed by the ring. More general bubbles consist of an outer ring of tetrahedra T_2 and T_3 and an enclosed inner part consisting of tetrahedra of the other two types.

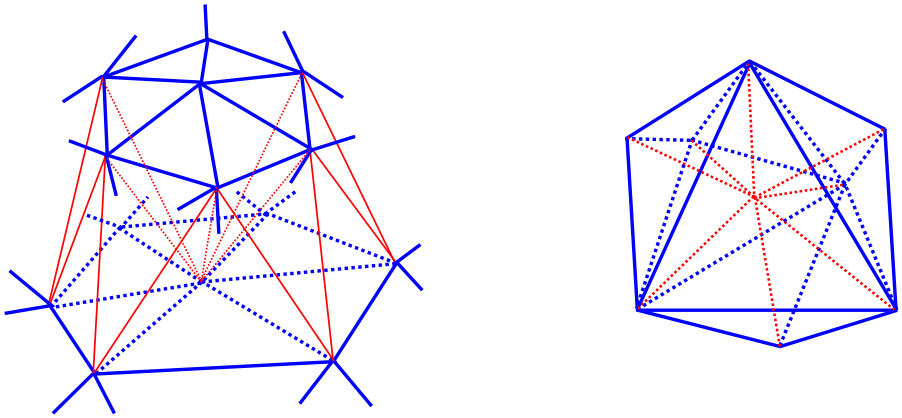


Figure 5.13: The left side shows a part of a thick slice known from CDT. The triangulation enclosed by two spatial sections forms a bubble containing tetrahedra of types T_5 and T_9. Some timelike links have been omitted. The right side shows a bubble containing a source or sink of time.

We can also start with a ring of tetrahedra of type T_3. Figure 5.12 d) shows a visualization of this situation. The spacelike triangles marked in yellow form a spatial disc. Furthermore, we have another set of spacelike triangles situated below the yellow disc which forms a second spatial disc. Both discs meet in a single vertex and we say that there is a *pinching* at that particular vertex. In section 7.3.1 we will describe a Monte Carlo move which creates such a pinching. We can generalize the situation by inserting tetrahedra of type T_2 into the existing ring, as shown in Fig. 5.12 e). The effect is that the two spatial discs now intersect in a link rather than a vertex.

Bubbles and pinchings can occur in combination to create even more complicated structures. Figure 5.12 f) shows an example built solely of tetrahedra of types T_2 and T_3. The spatial structure can still be interpreted as consisting of two spatial discs which not only meet at their boundaries but also at another vertex. If the pinching at the center is modified such that the local intersection of the two spatial sections is a triangle, we get a bubble with the topology of a torus. It is not difficult to check that all these structures are compatible with local causality.

Just like in 1+1 dimensions bubbles can be self-overlapping. Such bubbles have interior spacelike triangles. In 1+1 dimensions these self-overlapping bubbles are only possible if space has the topology of a circle so that the bubble can wind around the circle and overlap itself (see Fig. 5.6, bottom right). If we imagine removing the bubble from the triangulation, the remaining triangulation consists of two disconnected parts, assuming non-periodic boundary conditions in time. A similar criterion can be defined in 2+1 dimensions. We call a 2+1 dimensional bubble *globally self-overlapping* if it is self-overlapping and if its removal breaks the triangulation in two disconnected parts, again assuming nonperiodic boundary conditions in time. In 2+1 dimensions there exist other types of self-overlapping bubbles. For example, a bubble can be locally twisted to overlap with itself.

All the bubbles discussed so far contain tetrahedra of types T_2 and T_3. There also exist

bubbles which do not contain any of these tetrahedra. One example is familiar from CDT where the whole triangulation consists of several thick slices, each of them enclosed by two spatial sections at adjacent time steps (see Fig. 5.13, left). Another example is shown on the right of the figure. This bubble consists solely of tetrahedra of type T_5 which all meet at one vertex. The center vertex is necessarily a source or sink of time, depending on whether the (time-oriented) timelike links are all outward- or inward-pointing.

5.4 Kinematical constraints

The simplest type of information that one can extract from a triangulation is the number of subsimplices of the various types. We have seen that we have four types of tetrahedra in the Type 1 model of locally causal dynamical triangulations and in total ten subsimplex types. We use the following notation to count these objects:

$$
\begin{aligned}
N_0 &= \text{number of vertices} \\
N_1^s &= \text{number of spacelike links} \\
N_1^t &= \text{number of timelike links} \\
N_1 &= N_1^s + N_1^t \\
N_2^{sss} &= \text{number of triangles with three spacelike links} \\
N_2^{tss} &= \text{number of triangles with two spacelike links} \\
N_2^{tts} &= \text{number of triangles with one spacelike link} \\
N_2 &= N_2^{sss} + N_2^{tss} + N_2^{tts} \\
N_3^{T_2} &= \text{number of tetrahedra of type } T_2 \\
N_3^{T_3} &= \text{number of tetrahedra of type } T_3 \\
N_3^{T_5} &= \text{number of tetrahedra of type } T_5 \\
N_3^{T_9} &= \text{number of tetrahedra of type } T_9 \\
N_3 &= N_3^{T_2} + N_3^{T_3} + N_3^{T_5} + N_3^{T_9}
\end{aligned}
$$

There exist relations between these numbers which in the case of CDT have been presented in [34]. Here we repeat the analysis for the ensemble of locally causal triangulations of Type 1. The first identity is the well-known Euler formula

$$N_0 - N_1 + N_2 - N_3 = \chi$$

where χ is the Euler characteristic of the simplicial manifold. Since every tetrahedron contains four triangles and every triangle is shared by two tetrahedra we also have the constraint

$$N_2 = 2N_3 .$$

These two relations are shared with Euclidean triangulations and with CDT. In the latter we then have the foliation constraint $2N_2^{sss} = N_3^{T_5}$, which expresses the fact that in CDT

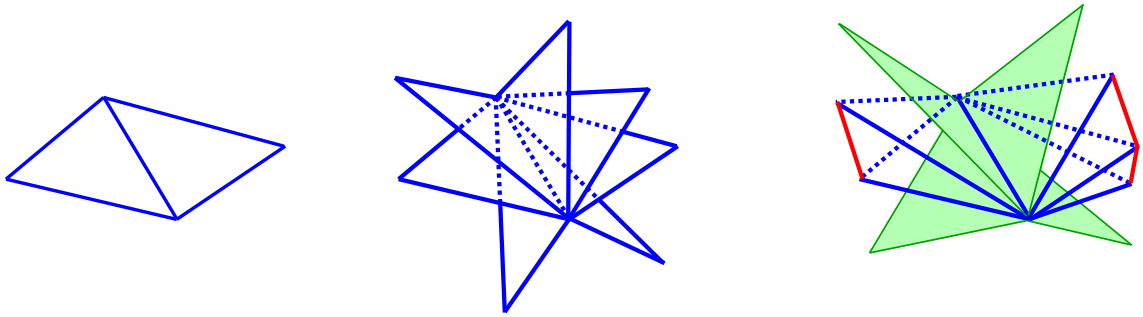


Figure 5.14: In 2+1 dimensional CDT every spacelike link is shared by two spacelike triangles (left). In more general situations we have spacelike links shared by more than two spacelike triangles (center). In causal situations the number of spacelike triangles around a generic spacelike link is determined by the number of tetrahedra of type T_2 around the link (right). The light cone structure is visualized in green.

every spacelike triangle is shared by two tetrahedra of type T_5 while every such tetrahedron contains exactly one spacelike triangle. The analogous relation in locally causal triangulations is

$$2N_2^{sss} = 2N_3^{T_2} + N_3^{T_3} + N_3^{T_5}.$$

This is easily understood by counting all the spacelike faces in the triangulation, which is the right-hand side of the relation, and noting that the number of spacelike triangles is half the number of spacelike faces.

In CDT we have two further constraints which describe the topology of the spatial sections. One is the Euler constraint $N_0 - N_1^s + N_2^{sss} = t\tilde{\chi}$, where $\tilde{\chi}$ is the Euler characteristic for the spatial sections and t is the number of these spatial sections. This constraint does not exist for locally causal triangulations where we have Monte Carlo moves which change the quantity $N_0 - N_1^s + N_2^{sss}$. Furthermore, in CDT every spacelike triangle contains three spacelike links and every spacelike link is shared by two spacelike triangles, yielding the relation $N_1^s = (3/2)N_2^{sss}$. This constraint expresses the fact that spatial sections in 2+1 dimensional CDT locally have the topology of a disc.

This is no longer true when the foliation constraint is relaxed, as illustrated in Fig. 5.14, therefore the constraint $N_1^s = (3/2)N_2^{sss}$ does not hold in the more general case. In locally causal triangulations there exists an analogue of this constraint which can be seen as follows. Let us first cut the whole triangulation at every spacelike link in such a way that each of these links belongs to precisely one spacelike triangle. In the CDT case this means that every spacelike link is effectively duplicated while in the more general case as many copies of the same link will be produced as there exist spacelike triangles around the original link. The total number of spacelike links is now the number of spacelike triangles multiplied by three.

Now we attempt to glue the links together in order to restore the original triangulation. We do this in two steps. Looking at the right-hand side of Fig. 5.14 we note that the spacelike triangles around the center link form two groups located in opposite sectors with respect to the light cone structure. Let n_L and n_R be the number of spacelike trian-

CDT	Locally causal triangulations
$N_0 - N_1 + N_2 - N_3 = \chi$	$N_0 - N_1 + N_2 - N_3 = \chi$
$N_2 = 2N_3$	$N_2 = 2N_3$
$2N_2^{sss} = N_3^{T_5}$	$2N_2^{sss} = 2N_3^{T_2} + N_3^{T_3} + N_3^{T_5}$
$N_0 - N_1^s + N_2^{sss} = t\tilde{\chi}$	$-$
$N_1^s = (3/2)N_2^{sss}$	$N_1^s = \frac{1}{2}(3N_2^{sss} - N_3^{T_2})$
$-$	$N_2^{sst} = N_3^{T_2} + N_3^{T_3}$

Table 5.3: The kinematical constraints in CDT and locally causal DT. The N_i variables have been defined at the beginning of this section.

gles in the left and right sector and let $n_{T_2,l}$ and $n_{T_2,r}$ be the number of tetrahedra of type T_2 in the left and right sector respectively. We have $n_L = n_{T_2,l} + 1$ and $n_R = n_{T_2,r} + 1$ as a consequence of the local causality constraint. This also implies that the only tetrahedra which appear in these two sectors are those of type T_2.

In the cut triangulation we have n_L copies of the center link belonging to spacelike triangles in the left sector and n_R copies belonging to spacelike triangles in the right sector. Now we glue all the links in the first group together and do the same for those in the right group. The n_L links in the left sector are thus replaced by one link and the same happens in the right sector. By doing this in the whole triangulation the number of spacelike links is reduced by $N_3^{T_2}$ which is the number of tetrahedra of T_2 in the whole triangulation.

In the situation on the right of Fig. 5.14 we now have one spacelike link glued to the left triangles and one spacelike link glued to the right triangles. After gluing these two links together and repeating the procedure everywhere in the whole triangulation the number of spacelike links gets halved and the original triangulation is restored. Thus we have found the constraint

$$N_1^s = \frac{1}{2}(3N_2^{sss} - N_3^{T_2}).$$

In locally causal triangulations there exists a constraint which has no analogue in CDT. Looking at the tetrahedra of the Type 1 model we observe that the tetrahedra of type T_2 and T_3 contain two triangles with exactly two spacelike edges while the other tetrahedra types do not contain any of them. It is clear that every sst triangle is shared by two tetrahedra where each of them can be of type T_2 or T_3. Therefore we have the constraint

$$N_2^{sst} = N_3^{T_2} + N_3^{T_3}.$$

We have checked that the Monte Carlo moves for the locally causal DT model, de-

scribed in chapter 7, are not compatible with the existence of any other constraint. Table 5.3 summarizes all constraints for CDT and the locally causal DT model. We have thus five constraints in each case. In the locally causal ensemble there are ten subsimplex variables and thus we conclude that five of them are independent. In the CDT ensemble we have seven subsimplex variables and only two of them are independent. These findings will have important consequences as we will see in the following section.

5.5 Actions and the Wick rotation

The path integral of gravity (2.1) assigns every configuration a weight proportional to $\exp(iS)$ where S is a *bare* action. This action is typically chosen to be the discretized Einstein-Hilbert action, also known as the Regge action, but other choices are possible as well. CDT with bare actions of Hořava-Lifshitz type has for example been investigated in [42]. We have chosen to work with the Regge action, which is also used in most works on CDT. We can therefore attribute any differences in the dynamical properties between CDT and the locally causal model to the enlargement of the ensemble, thus allowing us to effectively compare the two ensembles.

The Regge action was originally introduced in [27] and has been concretely worked out for CDT in [34]. In 2+1 dimensions the action reads

$$S_{\text{Regge}} = k \sum_{l \in \mathcal{N}^{\text{s}}} \text{Vol}(l) \frac{1}{i} \left(2\pi - \sum_{\substack{\text{tetrahedra} \\ \text{at l}}} \Theta \right) + k \sum_{l \in \mathcal{N}^{\text{t}}} \text{Vol}(l) \left(2\pi - \sum_{\substack{\text{tetrahedra} \\ \text{at l}}} \Theta \right) - \lambda \sum_{\text{tetrahedra t}} \text{Vol}(t). \tag{5.1}$$

$\mathcal{N}^s$ and $\mathcal{N}^t$ are the sets of spacelike and timelike links, k and λ are the gravitational and cosmological constants up to a constant factor and $\sum\Theta$ is the sum of all dihedral angles around the link l. In [34] it has been shown that an analytic continuation of the asymmetry parameter $\alpha \to -\alpha$ through the lower half of the complex plane defines a non-perturbative Wick rotation which sends the weights $\exp(iS)$ to $\exp(-S^{\text{eucl}})$ and thus allows a numerical analysis of the gravitational path integral. In 2+1 dimensions the map was shown to exist under the condition $\alpha > 1/2$. In the following we repeat this analysis for the ensemble of locally causal triangulations.

Let us study the behavior of each of the three terms in formula (5.1) under the map $\alpha \to -\alpha$. In the first term we have $\text{Vol}(l) = 1$ because the link is spacelike. The plane orthogonal to the link has Lorentzian signature thus if we circle around the link we encounter four light cone crossings as a consequence of the local causality constraint. Each crossing contributes a real part of $\pi/2$ to the total dihedral angle such that the deficit angle becomes purely imaginary. The Lorentzian plane changes into a Euclidean plane under $\alpha \to -\alpha$ and the deficit angle of a Euclidean plane is real. Therefore we conclude that the first term changes from real to purely imaginary under $\alpha \to -\alpha$.

In the second term the plane orthogonal to the timelike link is Euclidean and stays Euclidean after applying the map $\alpha \to -\alpha$. On the other hand we have $\text{Vol}(l) = \sqrt{\alpha}$ which

Tetrahedron	Volume	Wick rotation condition
T_2	$\frac{1}{12}\sqrt{\alpha(3+\alpha)}$	$\alpha < 3$
T_3	$\frac{1}{12}\sqrt{1+4\alpha+\alpha^2}$	$2-\sqrt{3} < \alpha < 2+\sqrt{3}$
T_5	$\frac{1}{12}\sqrt{1+3\alpha}$	$\alpha > \frac{1}{3}$
T_9	$\frac{1}{6}\sqrt{\frac{1}{2}+\alpha}$	$\alpha > \frac{1}{2}$

Table 5.4: Volumes of the tetrahedra in the Type 1 model and the conditions under which the expression acquires a factor of i under the analytic continuation $\alpha \to -\alpha$.

acquires a factor of $-i$ under the analytic continuation and therefore the second term also changes from real to purely imaginary. For the third term we need the volumes of the tetrahedra which are shown in table 5.4. The third column shows the condition on α such that the expressions for the volumes change from real to imaginary. The conclusion of the whole analysis is that there exists a Wick rotation in the ensemble of locally causal triangulations under the condition

$$\frac{1}{2} < \alpha < 3.$$

We note that the condition is stronger than the analogous condition $\alpha > 1/2$ for CDT.

By explicitly evaluating (5.1) using the expressions in table 5.2 and table 5.4 and after applying the Wick rotation the action acquires the form

$$S^{\text{eucl}} = \sum_i c_i(k,\lambda,\alpha) n_i \tag{5.2}$$
$$n_i \in \{N_0, N_1^s, N_1^t, N_2^{sss}, N_2^{tss}, N_2^{tts}, N_3^{T_2}, N_3^{T_3}, N_3^{T_5}, N_3^{T_9}\}.$$

In section 5.4 we have seen that there exist five relations among the n_i such that five of these variables are independent. This means that the phase space of the Regge actions spanned by k, λ and α forms a three-dimensional subspace in a five-dimensional space of more general actions. In the case of the CDT ensemble the number of independent n_i variables is two, with the consequence that all actions linear in the subsimplex count variables are of Regge type. This means that in the CDT case one can simulate the action (5.2) and essentially forget about the complicated relations between the original couplings and the new couplings.

Since we aim to simulate the locally causal ensemble with the Regge action we will work within the three-dimensional subspace spanned by k, λ and α and thus we need the explicit expressions for the c_i. One consequence of the kinematical constraints is that one can reformulate the action (5.2) such that only five terms are nonzero. We can even choose which of the n_i variables we would like to have in the action. We have decided to use the following form for the action:

$$S^{\text{eucl}} = \widetilde{c}_1 N_0 + \widetilde{c}_2 N_3 + \widetilde{c}_3 N_3^{T_2} + \widetilde{c}_4 N_3^{T_3} + \widetilde{c}_5 N_3^{T_5}. \tag{5.3}$$

$$
\begin{aligned}
\widetilde{c_1} =& -2k\pi\sqrt{-\alpha}\\
\widetilde{c_2} =& -2k\pi\sqrt{-\alpha}+\frac{1}{12}\lambda\sqrt{-2-4\alpha}+4k\sqrt{-\alpha}\arccos\left(\frac{1}{-1-4\alpha}\right)\\
&+2k\arccos\left(\frac{3+4\alpha}{1+4\alpha}\right)\\
\widetilde{c_3} =& -2k\pi+2k\pi\sqrt{-\alpha}-\frac{1}{12}\lambda\sqrt{-2-4\alpha}+\frac{1}{12}\lambda\sqrt{-\alpha(3+\alpha)}\\
&-4k\sqrt{-\alpha}\arccos\left(\frac{1}{-1-4\alpha}\right)+k\arccos\left(1+\frac{2\alpha}{3}\right)+4k\arccos\left(\frac{\sqrt{-\frac{\alpha}{4+\alpha}}}{\sqrt{3}}\right)\\
&+k\sqrt{-\alpha}\arccos\left(\frac{2+\alpha}{4+\alpha}\right)-2k\arccos\left(\frac{3+4\alpha}{1+4\alpha}\right)\\
\widetilde{c_4} =& -\frac{3}{2}k\pi+\frac{3}{2}k\pi\sqrt{-\alpha}-\frac{1}{12}\lambda\sqrt{-2-4\alpha}+\frac{1}{12}\lambda\sqrt{-1-4\alpha-\alpha^2}\\
&-4k\sqrt{-\alpha}\arccos\left(\frac{1}{-1-4\alpha}\right)+2k\arccos\left(\frac{2+\alpha}{\sqrt{3}\sqrt{-\alpha(4+\alpha)}}\right)\\
&+k\arccos\left(-\frac{1+2\alpha}{\sqrt{3}\sqrt{-1-4\alpha}}\right)-2k\arccos\left(\frac{3+4\alpha}{1+4\alpha}\right)\\
&+2k\sqrt{-\alpha}\arccos\left(\frac{1}{\sqrt{17+\frac{4}{a}+4\alpha}}\right)+k\arccos\left(\frac{2+4\alpha+\alpha^2}{4\alpha+\alpha^2}\right)\\
\widetilde{c_5} =& -4k\sqrt{-\alpha}\arccos\left(\frac{1}{-1-4\alpha}\right)+\frac{1}{12}\Bigg(-18k\pi+18k\pi\sqrt{-\alpha}-\lambda\sqrt{-2-4\alpha}\\
&+\lambda\sqrt{-1-3\alpha}+36k\arccos\left(\frac{1}{\sqrt{3}\sqrt{-1-4\alpha}}\right)+36k\sqrt{-a}\arccos\left(\frac{1+2\alpha}{1+4\alpha}\right)\\
&-24k\arccos\left(\frac{3+4\alpha}{1+4\alpha}\right)\Bigg)
\end{aligned}
$$

Table 5.5: The explicit expressions for the coefficients in the formula (5.3) such that the resulting action becomes identical to the Wick-rotated version of the Regge action (5.1).

Table 5.5 shows the lengthy expressions for the coefficients $\widetilde{c_i}$ which make (5.3) coincide with the Wick-rotated version of the Regge action (5.1).

In the special case $\alpha = -1$ and after some manipulations using the kinematical constraints, the action (5.3) using the coefficients in table 5.5 evaluates to

$$S^{\text{eucl}} = -2k\pi N_1 + \left(\frac{\lambda}{6\sqrt{2}} + 6k\arccos\frac{1}{3}\right)N_3, \tag{5.4}$$

which is identical to the Wick-rotated action of Euclidean dynamically triangulated gravity in three dimensions. We have also checked that if we set $N_3^{T_2} = N_3^{T_3} = 0$ our action can be reformulated to precisely match formula (34) in Ref. [34] which is the Regge action evaluated on CDT-compatible triangulations in 2+1 dimensions.

6

Locally causal DT in 2+1 dimensions: Simulations

After having introduced a 2+1 dimensional model of locally causal dynamical triangulations in the previous chapter we now set out to analyze the dynamics of the theory. The complicated nature of dynamical triangulation models in higher dimensions makes it virtually impossible to derive exact analytical results. Therefore we need to use Monte Carlo simulations to measure observables. To this end we have developed a new simulation software to analyze the dynamics of the locally causal DT model.

We are particularly interested in comparing with the dynamics of CDT in 2+1 dimensions, which has been investigated in [45] and [85]. It was found that there exists a phase of geometry containing three-dimensional extended geometries whose classical limit strongly resembles a Euclidean de Sitter spacetime in 2+1 dimensions. These results are analogous to the results found in 3+1 dimensions [37]. This shows that at least some large-scale physical aspects of the theory in 3+1 dimensions are already reflected in 2+1 dimensions. The results presented in this chapter may give us important clues about what we can expect from a 3+1 dimensional model of locally causal dynamical triangulations.

We begin with a discussion of various choices which have to be made in order to precisely define the configuration space. After discussing some details about running the simulations, we proceed by presenting numerical results. We will demonstrate the presence of two phases of geometry and show that almost strictly foliated states emerge in one of these phases close to the phase boundary. Then we perform a refined analysis in this phase using tetrahedron distributions which show that the triangulations are 'weakly foliated'. We conclude with a detailed analysis of volume distributions which provide strong evidence that the geometries are three-dimensional and that their classical limit is compatible with the 3-sphere, which is the Euclidean version of de Sitter spacetime in 2+1 dimensions.

6.1 The choice of configuration space

Ergodicity of Monte Carlo moves

Let us denote by $\mathscr{C}$ the configuration space containing all locally causal triangulations of Type 1. As explained in chapter 5, $\mathscr{C}$ is the space of all triangulations that can be built using tetrahedra of type T_2, T_3, T_5 and T_9, which satisfy local causality and which can be consistently time-oriented using the method described in sections 5.1 and 5.2. In order to explore this space we need Monte Carlo moves, thus let us denote the space of configurations which is reachable by the Monte Carlo moves of our simulations with $\mathscr{C}_{\mathrm{MC}}$. We have $\mathscr{C}_{\mathrm{MC}} \subset \mathscr{C}$ by construction and if $\mathscr{C}_{\mathrm{MC}} = \mathscr{C}$, the Monte Carlo moves are said to be *ergodic* in $\mathscr{C}$. Ergodicity is a key requirement for a Monte Carlo simulation which is supposed to sample all the states of a given configuration space with the Boltzmann probability.

In statistical models we generically have the phenomenon of *universality* which means that the continuum behavior of the model does not depend on the regularization details. More precisely, the phase diagram of such models decomposes into several phases where each phase belongs to a distinct universality class. In the case of dynamical triangulations we do not have any a priori arguments which would dictate the precise definition of the 'correct' configuration space. In this sense, the study of dynamical triangulations is not the search for a particular configuration space but the search for a universality class whose low-energy behavior is compatible with known physics.

If in our example we have $\mathscr{C}_{\mathrm{MC}} \neq \mathscr{C}$, we are simply talking about two different configuration spaces which may or may not belong to the same universality class. There is no a priori reason to declare that $\mathscr{C}$ is more 'correct' than $\mathscr{C}_{\mathrm{MC}}$. We simply need to be aware that we may not be able to extrapolate results extracted from $\mathscr{C}_{\mathrm{MC}}$ to the larger ensemble $\mathscr{C}$.

In our case we indeed are in the situation that we do not know whether our Monte Carlo moves are ergodic in the space of all locally causal triangulations of Type 1. It seems very difficult to carry out a proof of ergodicity. In addition, as we will describe below, we will have to forbid certain Monte Carlo moves to overcome technical difficulties. Therefore our whole numerical analysis aims to discover the universal properties of $\mathscr{C}_{\mathrm{MC}}$, leaving the question open whether $\mathscr{C}_{\mathrm{MC}}$ and $\mathscr{C}$ share the same universal properties or not.

Definining our ensemble

Most simulations of CDT use a fixed spacetime topology of the form of a direct product, where the topology of the spatial slices is fixed and compact and time is compactified to a circle. The latter prescription is at first not compatible with global causality because of the possibility of closed timelike curves. This choice is justified a posteriori by observing that the effective time extension of the emergent geometry in the physically interesting region of the phase diagram assumes a constant value independent of the total number of time steps, if the latter is chosen sufficiently large [37]. One could say that a source

and a sink of time emerge dynamically. The choice of cyclic time thus appears like a boundary condition which does not affect the observables of interest.

In simulations of CDT the number of time steps is fixed. Since CDT-compatible triangulations form a subset of all locally causal triangulations the question arises whether it is possible to alter the number of time steps of a CDT-compatible triangulation using a Monte Carlo process which at an intermediate stage goes through states which do not satisfy the foliation constraint of CDT. The answer is yes and the details will be described in section 7.4. If there exists a region in the phase diagram where the configurations are close to foliated, the notion of 'number of time steps' makes sense and one can ask which equilibrium value is found after thermalization.

While running early test simulations in the ensemble of locally causal triangulations with time compactified we have indeed found states which are approximately foliated, but the number of time steps would not thermalize properly. More precisely, sometimes the simulation would end up with a state having many time steps and at other times the number of time steps would be close to minimal. The reason for these problems is that the processes which alter the number of time steps are suppressed if the spatial extent of the triangulation is large. In the triangulations which were generated during these simulations, the smallest spatial extent turned out to be at least half of the largest spatial extent. Therefore the simulation was not able to alter the number of time steps except very early during the thermalization process.

We have been able to circumvent the problem by not compactifying time. More precisely, we work with the spacetime topology of a 3-sphere and place a source and a sink of time at the two poles. Both the source and sink of time can be seen as degenerate spacelike boundaries. As described above, in the case of CDT it was observed that in the physically interesting region a source and a sink of time emerges dynamically and that the topology effectively changes from $S^2 \times S^1$ to S^3. This provides another motivation for us to choose the 3-sphere as fixed topology.

Now we discuss another technical issue which appeared while running early test simulations: we observed that the simulations often would end up in 'frozen' states where virtually no progress can be made using the implemented Monte Carlo moves. The problem could be traced back to the appearance of globally self-overlapping bubbles, which we have defined in section 5.3. We have attempted to find additional moves to overcome these problems but without success. We therefore decided to reduce the ensemble using a mechanism which prevents these globally self-overlapping bubbles from appearing. We found that if we forbid all moves which merge or split bubbles, these problematic structures do not form. We observed indeed that the simulations behaved much better after the alteration although, as we shall discuss later, there are still regions in the phase diagram where the simulation does not thermalize sufficiently well.

Finally, we have decided to use combinatorial triangulations[1] since most work on CDT also uses them, thus allowing for a better comparison of both ensembles. Now we are ready to summarize our definition of the configuration space. Our ensemble is de-

[1] A d-dimensional triangulation is called a 'combinatorial triangulation' if every d-simplex has $d+1$ distinct vertices and $d+1$ distinct neighbors.

fined by the Monte Carlo moves described in chapter 7, which are chosen to be compatible with the locally causal dynamical triangulation model of Type 1, with combinatorial triangulations and without globally self-overlapping bubbles. The moves preserve local and global topology and we choose an initial configuration which has the topology of a 3-sphere.

6.2 Numerical setup

We have developed the Monte Carlo simulation software using C++ as programming language and taking advantage of object-oriented design principles to incorporate modularity and flexibility into the software. We expected beforehand that the complexity of the software would be higher compared to simulations of CDT. It turned out that the complexity even surpassed our expectations. It was thus also necessary to develop sophisticated testing procedures in order to test the correctness of the simulation.

In the volume term of the Regge action (5.1), the bare cosmological constant λ and the total system size appear as conjugate variables. Since we are interested in the infinite-volume behavior of our system we have the choice of whether to work in the grand-canonical or the canonical ensemble. In the former, λ is kept fixed and the system size is fluctuating, while in the latter the situation is reversed. We follow the choice made in [35] to simulate the canonical ensemble.

Most of our Monte Carlo moves change the system size and therefore we cannot keep the number of tetrahedra exactly fixed. The standard way of solving this problem is to add a potential to the action which forces the system size to fluctuate around a prescribed value. One example is a cusp-shaped potential of the form

$$\delta S = \varepsilon |N_3 - \widetilde{N_3}|$$

where $\widetilde{N_3}$ is the desired system size around which the system should fluctuate. A modified version of such a potential was used in [35]. We have chosen to work with a quadratic potential of the form

$$\delta S = \varepsilon (N_3 - \widetilde{N_3})^2$$

which was used, for example, in [31] and [89]. Our experiments with these two options gave a slight preference to the latter option in situations where the simulation experiences difficulties to decouple from the initial condition.

In the absence of a volume-fixing potential the bare coupling λ controls the average system size, therefore to achieve a desired system size the coupling needs to be fine-tuned. If λ is below this critical value the system size will grow, if it is above the critical value the system size will shrink. Let us assume the former and enable the volume-fixing term. The triangulation then experiences two 'forces', one coming from λ which drives the system size upwards and a balancing force coming from the potential which pulls the system size back towards the prescribed value.

We aim to minimize these 'forces' by tuning the value of λ periodically such that it fluctuates around the critical value. The latter is not known but it is clear that if λ is too

small the simulation will mostly stay on the upper part of the potential and vice versa. By making sure that the simulation spends about half of its time on each side of the potential we effectively achieve the desired fine-tuning of λ. [2]

Before measurements can be made, the simulations need to undergo a phase of thermalization. The time it takes for thermalization is not known beforehand and therefore we need criteria to judge when the simulation is close enough to equilibrium. During the thermalization process our simulation periodically prints out measured values of various observables. If we plot these values as a function of simulation time, they normally start drifting in one direction until they acquire some stable value. We use the stability of all of these observables as a necessary condition for the thermalization to be sufficient.

The main problem with this criterion is that there could be other reasons why the observables stabilize. One possibility is that the simulation reaches a metastable state where it stays for a long time until it decays to the equilibrium state or even to another metastable state. For example, in section 3.1 we have reported the appearance of such states in the context of 3+1 dimensional CDT close to the B-C phase transition.

There could also be algorithmic reasons for why observables stabilize to values which are not their equilibrium values. As we have described in section 6.1, we have observed that simulations which allow for the presence of globally self-overlapping bubbles end up in states where almost no progress is made by executing Monte Carlo moves. Such triangulations are special cases of a more general class of problematic triangulations which have the property of being 'incompressible', a notion which we will explain in the following.

As we have described above, in the absence of a volume-fixing term the average system size is a function of the coupling λ. Let us assume we run a simulation. As soon as we observe the stabilization of observables, we disable the volume-fixing and set λ to a value significantly higher compared to the value it had before. Normally one would expect the triangulation to shrink to a state with almost minimal size in a short amount of time, because a value of λ which is above the critical value acts like a pressure force which attempts to shrink the triangulation. If this shrinking process indeed happens we say that the triangulation is *compressible*.

It may happen that the triangulation does not shrink to a minimal size, in which case we say that the triangulation is *incompressible*. One then typically observes that the number of accepted Monte Carlo moves goes down significantly until no substantial progress is made. In such a situation the Monte Carlo process fails to reach equilibrium. The canonical solution for this problem is to find and implement new Monte Carlo moves which increase the mobility of the simulation, but this may not always be possible or feasible. In our situation we have improved the situation by forbidding the presence of globally self-overlapping bubbles, but the problem was not entirely eliminated. Therefore, to judge the quality of thermalization, we test the compressibility of a triangulation in addition to testing the stability of observables.

After the thermalization phase the actual measurements are being performed. Since

[2] This tuning technique was already implemented in earlier simulations of CDT but was to our knowledge never documented in detail.

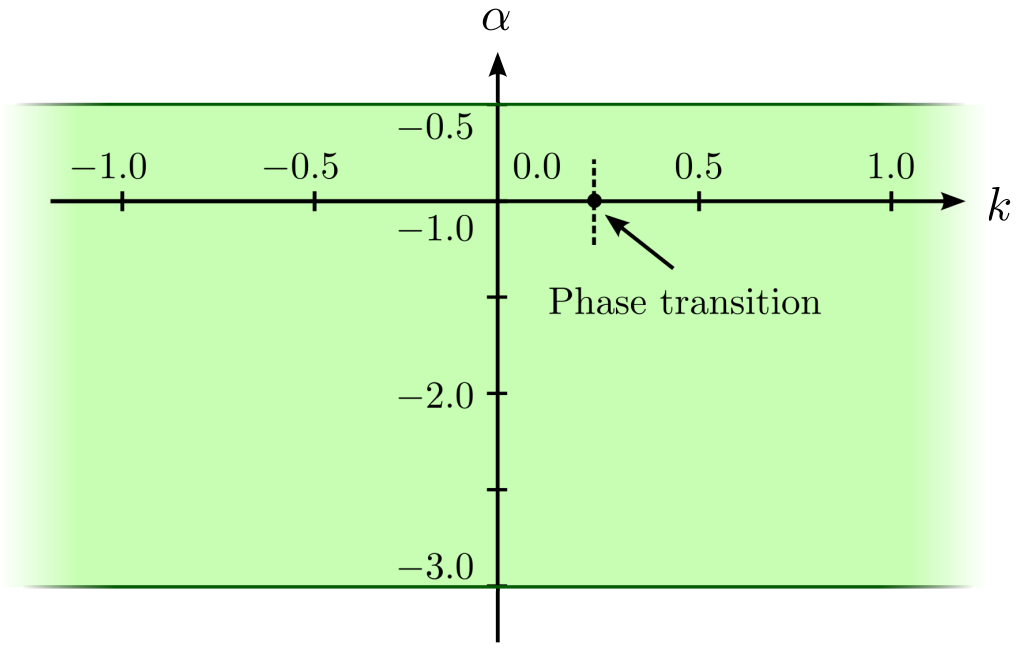


Figure 6.1: The phase diagram of the locally causal DT model of Type 1, defined by the strip $-1/2 > \alpha > -3$, visualized in green. Our investigation has probed the phase space at points on the k-axis ($\alpha = -1$) and on the α-axis ($k = 0$).

we want to extract information about the canonical ensemble we need to deal with the fact that the system size is not exactly constant during the simulation. This problem can easily be solved by only performing measurements whenever the system size exactly matches the prescribed value.

6.3 Exploring the phase diagram

The Regge action (5.1) contains two couplings, k and λ, which are proportional to the inverse bare Newton coupling and the bare cosmological constant. When evaluating the action on locally causal triangulations (which include CDT-compatible triangulations) a third coupling naturally appears, which is the asymmetry parameter α. These three couplings span a three-dimensional space of bare actions. As we have described previously, λ is fine-tuned to a critical value which corresponds to a particular system size. Since we keep the system size approximately fixed we essentially work with a two-dimensional phase diagram spanned by k and α. As explained in section 5.5, the existence of a Wick rotation limits the allowed values for α to the region $1/2 < \alpha < 3$. The Wick rotation maps this range to $-3 < \alpha < -1/2$, which defines the phase diagram of the locally causal DT model of Type 1, which we will explore in the following. Figure 6.1 shows a visualization of the phase diagram.

In the following we analyze the dynamics of our model on the k-axis and on the α-axis. While the simulations work well on the k-axis we encountered difficulties when ex-

ploring the α-axis in the region $-3 < \alpha < -1$. As one moves away from $\alpha = -1$ towards $\alpha = -3$ the triangulations become increasingly incompressible and the thermalization time increases rapidly. A closer analysis revealed that the severity of the problems correlates with the presence of bubbles with complicated internal structure. These problems currently force us to restrict our focus to the region $-1 < \alpha < -1/2$.

Bounds on the vertex density

In 2+1 dimensions there exist kinematical bounds on the vertex density N_0/N_3 and the link density N_1/N_3. In the case of CDT the link density bounds have been analyzed in [34]. It was found that the relation $1 \leq N_1/N_3 \leq 5/4$ holds for CDT while the Euclidean triangulations satisfy the weaker constraint $1 \leq N_1/N_3 \leq 4/3$. Using the kinematical constraints in table 5.3 one finds that $N_0/N_3 = N_1/N_3 - 1$ in the infinite-volume limit. This implies that the link density constraints translate into the vertex density constraints $0 \leq N_0/N_3 \leq 1/4$ for CDT and $0 \leq N_0/N_3 \leq 1/3$ for Euclidean triangulations.

In the case of CDT the link density constraint can be derived using the kinematical constraints. The derivation involves the spatial Euler constraint which is not present in the ensemble of locally causal triangulations. To find the analogous constraint in the locally causal case we follow [34] by considering all those Monte Carlo moves which create a vertex. All of them change the number of tetrahedra by some amount ΔN_3. Then we take all the moves in this set for which ΔN_3 acquires the minimal value. If we start with a minimal triangulation and then act with only the selected moves, the vertex density and thus also the link density is maximized when we approach the infinite-volume limit.

In the case of locally causal triangulations we have two Monte Carlo moves which create one vertex and three tetrahedra, namely, the bubble move and the polar move which will be described in sections 7.3.2 and 7.4. Both moves are unconstrained moves which can always be executed. Therefore we find that in the locally causal model the vertex and link densities satisfy the constraints

$$0 \leq \frac{N_0}{N_3} \leq \frac{1}{3},$$
$$1 \leq \frac{N_1}{N_3} \leq \frac{4}{3}.$$

These relations agree with those of Euclidean triangulations but, as we shall see, the configurations which saturate these constraints differ substantially. Note that in a locally causal model using degenerate triangulations the constraints will be weaker, since bubbles with less than three tetrahedra exist there.

Figure 6.2 shows the measurements of the average vertex densities for various values of k using simulations with $N_3 = 40000$ and $\alpha = -1$. The vertex density increases monotonically with k, which is not surprising since larger values of k make the creation of vertices cheaper in terms of the action. This can be easily seen by rewriting eq. (5.4) using the kinematical constraints, which gives $S^{\text{eucl}} = -2k\pi N_0$ plus a term proportional to N_3. We see that the curve approaches the upper kinematical bound of $1/3$ for large values of k. We also see the clear sign of a phase transition between $k = 0.24$ and $k = 0.28$,

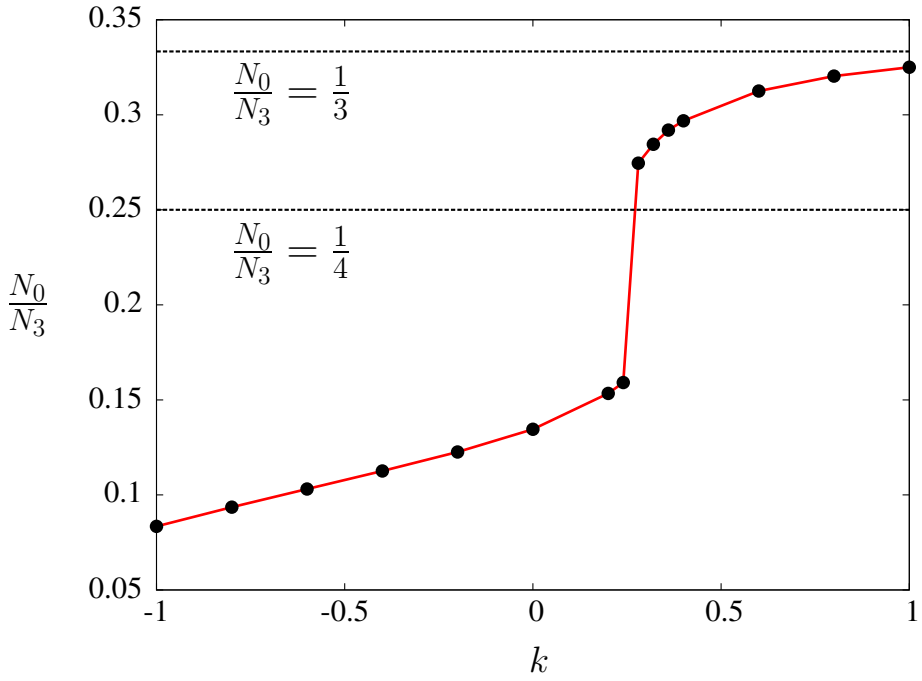


Figure 6.2: Measurement of the average vertex density as function of k, for a system size $N_3 = 40000$ and for $\alpha = -1$. The dots represent actual measurements, the lines are linear interpolations. The dashed lines mark the kinematical bounds for Euclidean triangulations and for CDT.

connecting a phase of low vertex density with another phase of high vertex density. The analogous measurements for $k = 0$ and varying α show that the vertex densities are approximately constant at low values and there is no sign of a phase transition there. Note that, as we have described previously, for technical reasons we only investigate the region $-1 < \alpha < -1/2$ inside the phase diagram. It is of course possible that there exist other phase transitions in the complementary region of the phase diagram.

Emergence of foliated triangulations

Foliated causal triangulations form a subset of locally causal triangulations characterized by the condition $N_3^{T_2} = N_3^{T_3} = 0$.[3] Therefore by plotting the number of tetrahedra of these two types as function of the couplings we can look for regions in the phase diagram where foliated triangulations emerge dynamically. Figure 6.3 shows the number of tetrahedra of all types, averaged over the sampled states in the ensemble, as function of the coupling k. We see that on the left, in the phase with low vertex density the building blocks of foliated triangulations dominate. However, also the other tetrahedra

[3] With periodic boundary conditions in time one could in principle construct a triangulation with $N_3^{T_2} = N_3^{T_3} = 0$ consisting of one single bubble winding around both space and time which clearly is not foliated. These structures are not part of our chosen ensemble because time is not compactified in our case and because we do not allow globally self-overlapping bubbles.

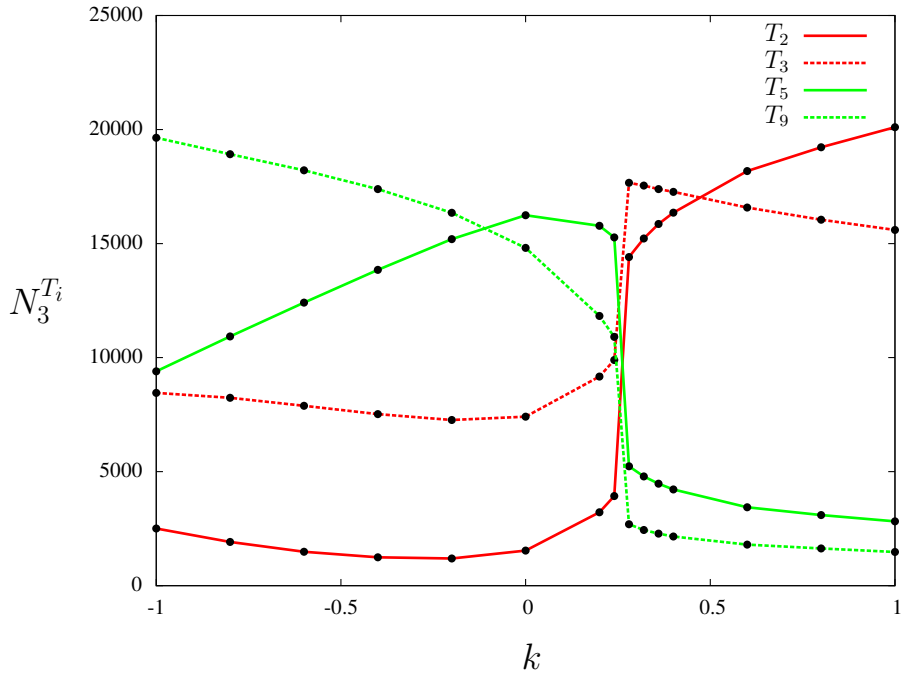


Figure 6.3: The number of tetrahedra of each type, averaged over the sampled triangulations, as function of the coupling k, at $\alpha = -1$.

types appear in significant numbers, therefore foliated triangulations do not appear on the k-axis.

In section 5.3 we have discussed that the tetrahedra of type T_2 and T_3 always form rings in the Type 1 model. Rings that only contain tetrahedra of type T_2 are special types of bubbles, which we call 'simple bubbles'. These bubbles are bounded by two spacelike discs, each of them having exactly one interior vertex. Rings which only contain tetrahedra of type T_3 form pinchings. On the left side of the figure we find $\langle N_3^{T_3} \rangle > \langle N_3^{T_2} \rangle$, which shows that pinchings dominate over simple bubbles, while on the right side in the other phase we have $\langle N_3^{T_2} \rangle > \langle N_3^{T_3} \rangle$, indicating a domination of simple bubbles.

Figure 6.4 shows the expectation values of the number of tetrahedra as function of α. Note that the phase boundary $\alpha = -0.5$ does not belong to the phase diagram since the Wick rotation does not exist for $\alpha = -0.5$. The measurements represented by the rightmost dots in the figure have been performed at $\alpha = -0.52$. We find that both $N_3^{T_2}$ and $N_3^{T_3}$ approach zero as we move towards the phase boundary. At $\alpha = -0.52$ we have measured $\langle N_3^{T_2} \rangle \approx 2.9$ and $\langle N_3^{T_3} \rangle \approx 14.3$, such that in the whole system consisting of 40000 tetrahedra almost none of the building blocks belong to the types T_2 and T_3.

We conclude that the states appearing close to $\alpha = -0.5$ are almost perfectly foliated and belong to a phase with low vertex density. On the effective level the dynamics should therefore be very close to the known dynamics of 2+1 dimensional foliated triangulations in the extended phase [45]. We therefore expect these states close to the phase boundary to be macroscopically extended with a characteristic blob-shaped volume distribution. We will confirm these expectations later on.

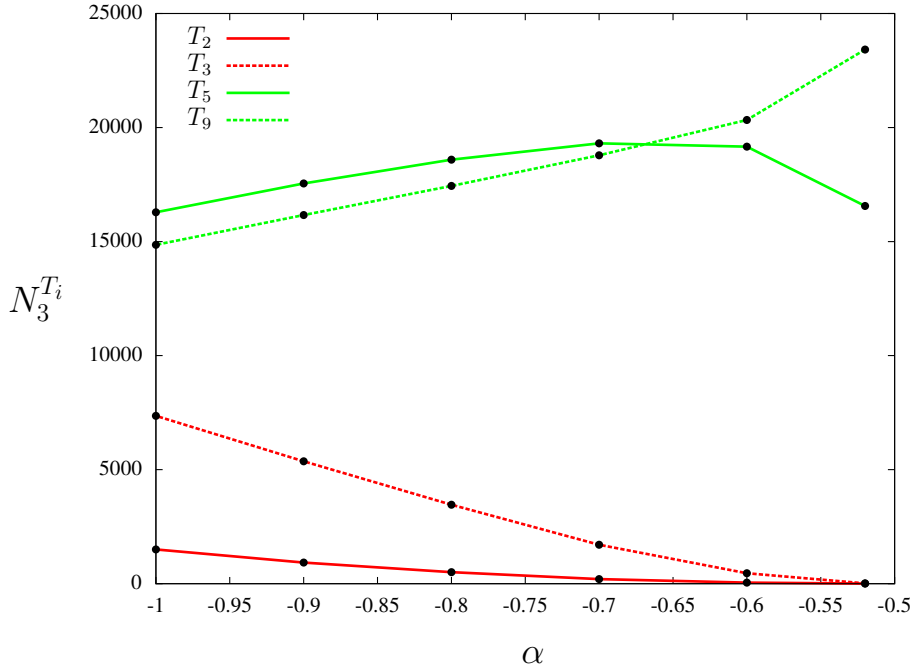


Figure 6.4: The number of tetrahedra of each type, averaged over the sampled triangulations, as function of the coupling α.

6.4 Tetrahedron distributions

Defining a time coordinate

By labeling the leaves of a foliation one immediately gets a time coordinate which subsequently can be used to analyze distributions of tetrahedra and volumes of spatial sections as a function of that time coordinate. In locally causal triangulations generic configurations are not foliated and therefore there is no canonical time coordinate. Our next goal is to define a time coordinate for locally causal triangulations which in the special case of CDT reduces to the canonical time coordinate coming from the foliation. This compatibility requirement does not guarantee that the time coordinate is useful for locally causal triangulations. In the following we present an algorithm which gives us a time coordinate that contains relevant information for both foliated and nonfoliated configurations.

In the following we assume that the spacetime topology is that of a 3-sphere and that the triangulation has been time-oriented using the method described in sections 5.1 and 5.2. For every vertex we consider the set of all future-oriented paths connecting the vertex with the sink of time. The number of links in each path defines a distance between the vertex and the sink of time. We average this quantity over all paths to calculate an average distance which we call d_f. We do the same for all past-oriented paths, giving another average distance d_p. Finally we define the time coordinate to be $t = d_f - d_p$.

Since the number of future-oriented (past-oriented) paths between a vertex and the corresponding source (sink) of time can become very large, we have decided to use a modified algorithm which calculates t in an approximate fashion. For every vertex we

construct N_p future-oriented paths using a random process where we iteratively jump from a vertex to a randomly chosen future neighbor until we reach the sink of time. Afterwards we do the same for the past direction and finally calculate the time coordinate as the difference of both average distances. N_p is an empiric parameter which should be chosen large enough such that any further increase only marginally changes the final outcome. In our simulations we have chosen $N_p = 100$.

In CDT the distance between a vertex and one of the poles defined in this way does not depend on the path taken, thus t is an integer. If we compare the values of t for two vertices which are neighbors and connected by a timelike link we find that the t-values for both vertices differs by two. Therefore our time coordinate induces a labeling of spatial sections which is identical to the canonical labeling in CDT up to a trivial factor of two which we in principle could normalize away.

We have experimented with other notions of time. For example, one could define a time coordinate by measuring the shortest distances from each vertex to the two poles. The tetrahedron distributions which we will present below contain less useful information when the shortest distance prescription is used. We essentially observe that relevant structures which are clearly visible when the average distance prescription is used become smeared out.

Weakly foliated triangulations

We have seen that foliated configurations emerge close to the boundary of the phase diagram. As we move away from the boundary the configurations become less foliated. A strict foliation is achieved if $N_3^{T_2} = N_3^{T_3} = 0$, but it is not clear how to translate nonzero values to a measure of foliatedness of a triangulation. We would like to have a more refined observable which tells us how foliated a triangulation is. In the following we use tetrahedron distributions based on the time coordinate introduced above as a qualitative tool to investigate the foliatedness of a triangulation.

Let us assume that all the vertices have been assigned a time coordinate using the algorithm described above. For each tetrahedron we calculate the sum of all the time coordinates of its vertices and then round the value to the nearest integer. This value is then assigned to the tetrahedron as a new time coordinate. It is clear that the normalization of this time coordinate is different from the normalization of the vertex time coordinate, but that does not matter as long as we do not use both time coordinates at the same time.

For each value of the tetrahedron time coordinate we can now count the number of tetrahedra which share this value. Let us first consider the foliated case. There the tetrahedra organize themselves in a sequence of layers with the structure $\cdots - \mathscr{T}_5^f - \mathscr{T}_9 - \mathscr{T}_5^p - \mathscr{T}_5^f - \cdots$ as explained in [45]. Every $\mathscr{T}_5^f - \mathscr{T}_9 - \mathscr{T}_5^p$-block forms a so-called thick slice between two spatial sections. In a tetrahedron of type T_5 we have three vertices which are shared by the spacelike triangle and a fourth vertex which is connected to the three other vertices using timelike links. $\mathscr{T}_5^f$ ($\mathscr{T}_5^p$) denote the layer consisting of tetrahedra of type T_5 where the fourth vertex has larger (smaller) time coordinate than the other three vertices. The number of tetrahedra in two adjacent T_5 layers are equal, because

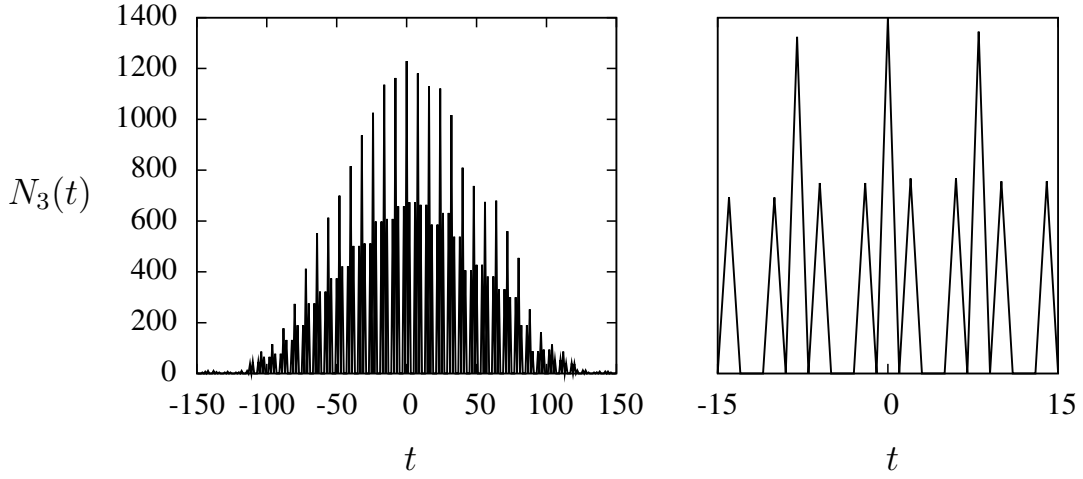


Figure 6.5: The tetrahedron distribution of a single strictly foliated triangulation, extracted from a simulation with foliation constraint enabled. The right figure is a magnification of the center region in the left figure.

each member of one layer neighbors precisely one member of the other layer through the common spacelike triangle.

Now consider two neighboring tetrahedra, one belonging to $\mathscr{T}_5^f$ and the other to $\mathscr{T}_9$. Both tetrahedra have three vertices in common. The time coordinates of the remaining two vertices differ by two, taking into account our normalization. Therefore the time coordinates of both tetrahedra also differ by two. The same is true if we consider two neighbors belonging to $\mathscr{T}_5^p$ and $\mathscr{T}_9$. If two neighboring tetrahedra are of type T_5, the time coordinate of the two non-shared vertices differs by four. We conclude that if we plot the tetrahedron distribution of a foliated state we encounter a sequence of blocks, where each block labeled by integer j has the structure $N_3^{T^f_{5,i}} - 0 - N_3^{T_{9,i}} - 0 - N_3^{T^p_{5,i}} - 0 - 0 - 0$ and where we have $N_3^{T^p_{5,i}} = N_3^{T^f_{5,i+1}}$.

Figure 6.5 shows the measured tetrahedron distribution of one single strictly foliated triangulation which we have generated by running our simulation with foliation constraint enabled. On the left side we see that the distribution appears as the combination of two blob-shaped distributions, one formed by the $\mathscr{T}_5$ layers and the other by the $\mathscr{T}_9$ layers. On the right side the center part of the distribution is shown in detail, revealing the presence of the sequence which we have derived above.

Now we move back to our simulations of the locally causal model and focus on triangulations which are a bit further away from the phase boundary. Figure 6.6 shows the tetrahedron distribution of a single triangulation extracted from a simulation at $k = 0.0$, $\alpha = -0.7$ with $N_3 = 40000$. When comparing the plot with Fig. 6.5 we observe several differences. On the right of Fig. 6.5 we see a succession of blocks, each of them representing one thick slice in the foliated triangulation, where each thick slice further consists of three layers which we previously have denoted by $\mathscr{T}_5^f$, $\mathscr{T}_9$ and $\mathscr{T}_5^p$.

In Fig. 6.6 we still see remnants of the three-peak structure of each block, but there is a

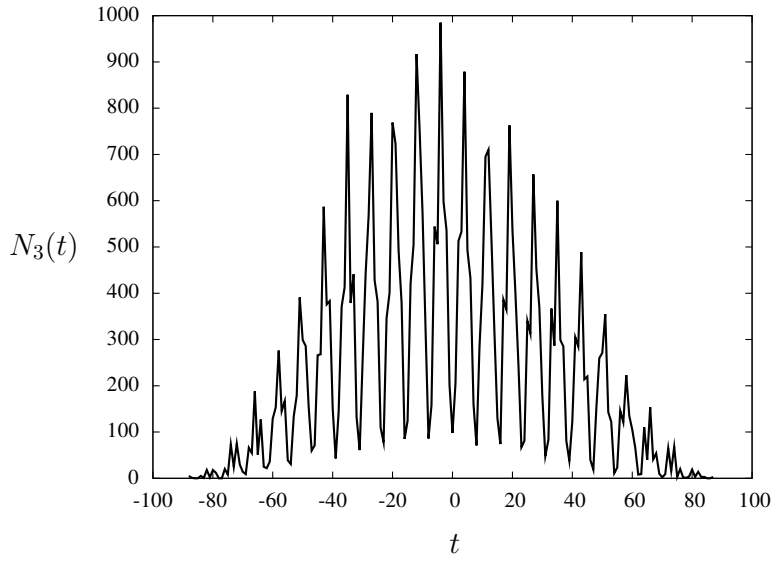


Figure 6.6: The tetrahedron distribution of a single triangulation drawn from a simulation of the locally causal model at $k = 0.0,\ \alpha = -0.7$.

tendency for these structures to become blurred. This blurring effect most likely depends both on changes in the actual triangulation and on the precise algorithm which is used to define the time coordinate and the tetrahedron distribution. Therefore the relevant information lies not in the precise structure of the peaks, but in the overall pattern formed by the succession of all these peaks. Every peak in Fig. 6.6 thus corresponds to one three-peaked block in Fig. 6.5 and describes a structure which resembles a thick slice in a foliated triangulation.

If we again go back to Fig. 6.5 on the right, we also note the gaps between the blocks which mark the presence of the spatial sections in the foliated triangulation. In Fig. 6.6 we note that a new type of layer has emerged at the places where the gaps used to be. These new layers can be interpreted as 'decorated spatial sections' which arise from purely spatial sections by the creation of bubbles. One could say that the spatial sections have acquired a non-zero thickness.

Based on these findings we can interpret the pattern shown in Fig. 6.6 as a triangulation where decorated spatial sections alternate with modified thick slices. We say that a triangulation exhibiting such a structure is *weakly foliated*. This is obviously not a sharp definition as it is not clear at which point a weakly foliated triangulation changes into a truly nonfoliated triangulation. We take here an operational point of view and consider a triangulation to be weakly foliated whenever the tetrahedron distribution shows the characteristic alternation pattern.

It is interesting to study how the foliatedness of a triangulation changes as one moves around in the phase diagram. Figure 6.7 shows the tetrahedron distributions of single triangulations extracted from simulations at $k = 0.0$ for various choices of α. We find an almost strict foliation at $\alpha = -0.52$ which gradually weakens as we move towards $\alpha = -1.0$. It would be even more interesting to continue beyond $\alpha = -1.0$ towards the other

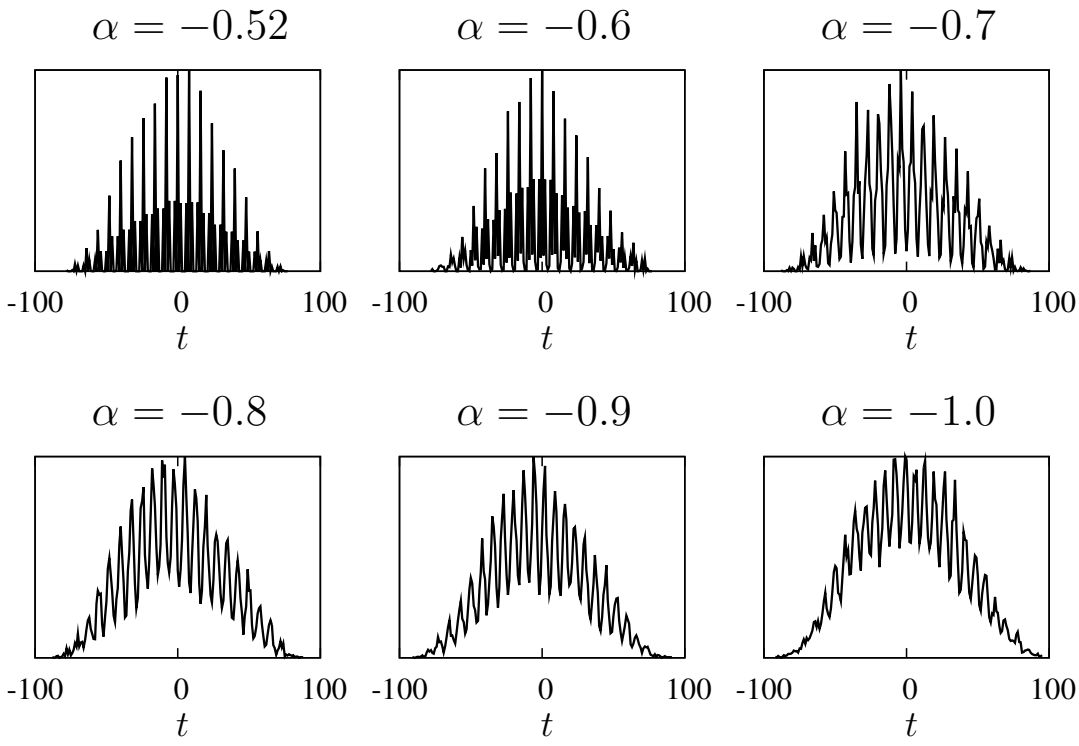


Figure 6.7: The tetrahedron distributions for single triangulations, drawn from simulations at $k = 0.0$ for various choices of α.

phase boundary, to see whether the alternating pattern in the tetrahedron distribution ceases to exist at some location in the phase diagram, possibly indicating a breakdown of the weak foliation. As we have discussed above, technical issues currently prevent us from exploring the region $-3.0 < \alpha < -1.0$.

We have performed a similar analysis on the k-axis. We observe that the alternation pattern becomes even less pronounced as we move from $k = 0.0$ towards $k = -1.0$, indicating a further weakening of the foliation. If we go from $k = 0.0$ in the other direction towards the phase transition the data quality decreases significantly, in such a way that interpretations based on the tetrahedron distributions become unreliable. We can now summarize the discussion in this section by stating that all investigated states in the phase of low vertex density exhibit a weak foliation which becomes an almost strict foliation as one approaches the phase boundary at $k = 0.0,\ \alpha = -0.5$.

6.5 Volume distributions

In $d = D + 1$ dimensional CDT it is easy to measure the distribution of spatial volume by simply counting the number of D-simplices in each spatial section of integer time and plotting this number against the canonical time coordinate. It has been shown that the average volume distribution in 3+1 dimensional CDT very accurately agrees with the profile of the round four-sphere which is nothing else than the Wick-rotated version of

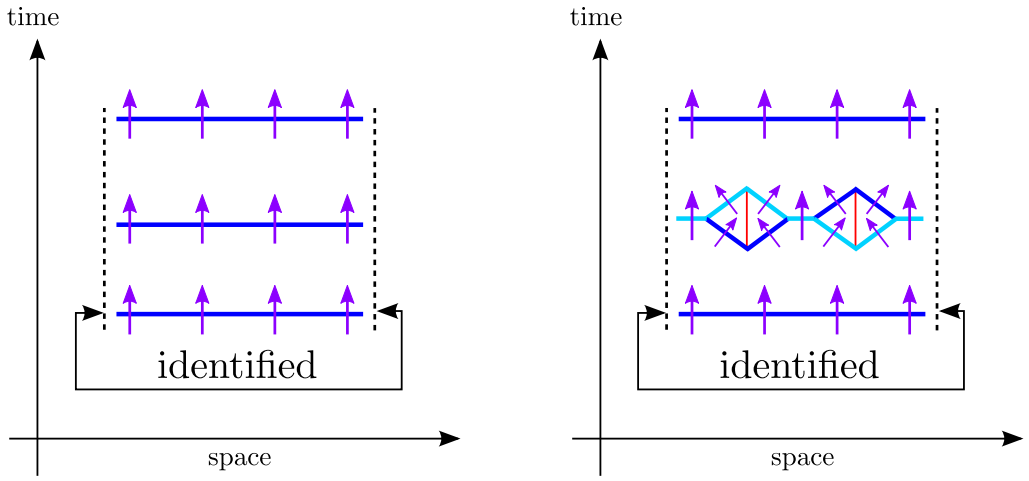


Figure 6.8: A 1+1 dimensional illustration of spatial structures of a foliated triangulation (left) and of a locally causal triangulation (right). The time orientations of spatial links are marked with purple arrows. The structure in light blue is a spatial slice as defined in the text.

the de Sitter spacetime [37]. An analogous result has been found in 2+1 dimensions in [45] and [85] with the emergence of a three-dimensional de Sitter space on large scales. The volume distribution is therefore an important large-scale observable because it provides a nontrivial condition which must be satisfied by the emergent geometry in order to be compatible with the global structure of the universe.

Spatial slices in locally causal triangulations of Type 1

In order to implement a volume distribution for locally causal triangulations we need to define spatial slices. For illustrative purposes, it is convenient to discuss this in 1+1 dimensions first. In the following we assume non-periodic boundary conditions in time direction. The left side of Fig. 6.8 shows the spatial structure of a time-oriented foliated triangulation. The set of all spatial links is organized in a fixed number of connected components, each of them being called a spatial slice and having circular topology. If we imagine cutting the triangulation along one of these spatial slices, the triangulation breaks into two disconnected pieces. In addition, time flows consistently from one subtriangulation to the other.

These observations allow us to define a spatial slice in an alternative way which generalizes to locally causal triangulations. Namely, we define a *spatial slice* to be a set of spatial links with circular topology such that if we cut along the slice, the triangulation breaks into two disconnected parts and time flows consistently from one part into the other. On the right of Fig. 6.8 the central slice has been replaced by a 'decorated spatial section' with two bubbles. There are four different spatial slices in this spatial section; one of them is visualized in light blue. These four slices are obviously not disjoint. Note that in general not every spacelike link belongs to a spatial slice. In section 5.3 we have

described self-overlapping bubbles which contain interior spacelike links. See Fig. 5.6 on the right bottom for an example which shows that these interior spacelike links indeed do not belong to a spatial slice.

In 2+1 dimensions we define a spatial slice in a completely analogous way as the set of spatial triangles of a fixed topology $\mathscr{T}$ such that if we cut along the slice, the triangulation breaks into two disconnected parts and time flows consistently from one part into the other. Since we use the 3-sphere spacetime topology in our simulations, the topology $\mathscr{T}$ in our case is the topology of a 2-sphere.

Creating and averaging volume distributions

If we again look at Fig. 6.8 on the right it becomes clear that if we place n bubbles next to each other, the number of spatial slices in the middle area will become 2^n. This illustrates that the number of spatial slices in locally causal triangulations can become so large that a complete enumeration of all slices becomes unfeasible. The situation is even worse in 2+1 dimensions. Therefore we have developed an algorithm which samples a subset of the whole set of spatial slices using statistical methods. The details of this algorithm are spelled out in section 7.5.

In order to build a volume distribution we first assign a time coordinate to each vertex using the algorithm described in section 6.4. Then we determine a set containing N_s spatial slices using the algorithm mentioned in the previous paragraph. By experimenting with the number N_s we can attempt to find a good compromise between getting more statistics and preventing too much slowdown of the Monte Carlo process. In our simulations we have used $N_s = 1000$. Now we assign a time coordinate to every spatial slice by taking the average of the time coordinates of all the vertices which are contained in the slice. Then we count the number of spacelike triangles in each slice and plot this number against the new time coordinate.

Figure 6.9 shows an example of a volume distribution of a single triangulation, drawn from a simulation at $k = 0.2,\ \alpha = -1.0$ in the low vertex density phase. Note that the normalization of the time coordinate is different from the one used for the tetrahedron distributions in the previous section. Every dot in the figure represents one single spatial slice. We observe the presence of dot clusters, which represent groups of spatial slices very close together with slightly varying volume. These structures are readily interpreted as 'decorated spatial sections' and the gaps in between can be interpreted as modified thick slices which are bounded by two spatial slices. These are the characteristic signs of a weakly foliated triangulation which we have introduced during the discussion of tetrahedron distributions.

Our next goal is to create an average volume distribution by superimposing the distributions of the individually sampled triangulations. There is no unique way to perform such an averaging, and therefore one should in principle investigate different methods and study how the final results depend on the chosen method. In the case of CDT this issue has been discussed in [37]. There one exploits the typical blob-shape of the individual volume distributions in the extended phase and superimposes them essentially by

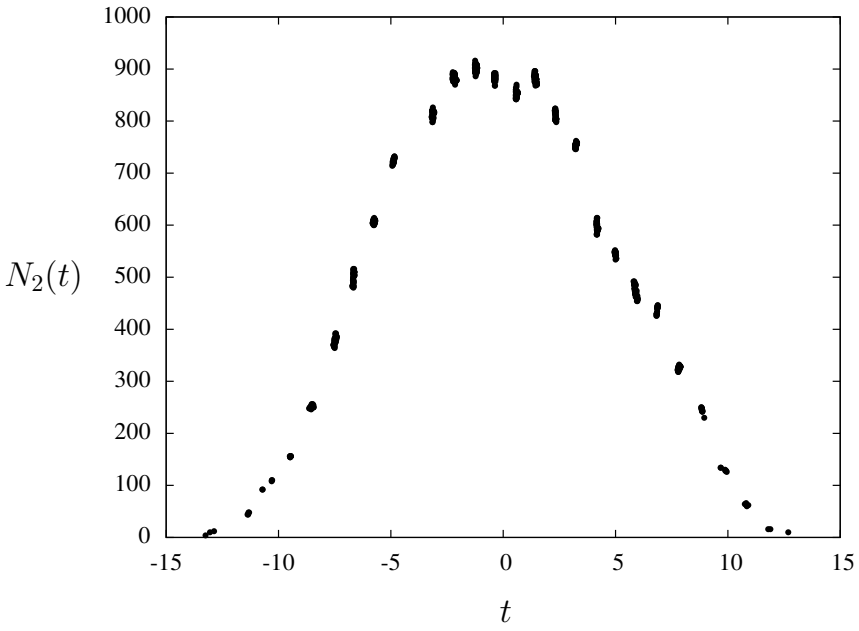


Figure 6.9: An example of a volume distribution of a single locally causal triangulation ($k = 0.2,\ \alpha = -1.0$) exhibiting a weak foliation.

matching their center of volume.

In the case of locally causal triangulations we face additional problems. The naive way of calculating the center of volume using $t_{cv} = \sum t\, N_2(t) / \sum N_2(t)$ would lead to a biased result. Our statistical method to generate a set of spatial slices does not produce perfectly balanced distributions. If, for example, more slices are sampled on the left side of the volume distribution and less slices are sampled on the right side, the center of volume becomes biased towards small time values. We need a more sophisticated averaging procedure which we explain in the following.

First we generate a grid on the t-axis which is significantly finer than the time distance between two dot clusters. Then we run an algorithm to detect the dot clusters and for every cluster calculate its average t-coordinate and volume. The whole dot cluster is then replaced by one single dot which is placed according to the calculated average values. Afterwards we use the well known method of splines to generate a function which smoothly interpolates between all the dots.

Now we repeat this procedure for all sampled triangulations. More precisely, we first duplicate each state in the sampled ensemble and perform a time-reversal operation on one member of each pair. This simply adds more statistics and is compatible with running a simulation with a global time-reversal move enabled. We end up having a set of functions which represent all the individual volume distributions. The advantage of having a set of functions is that we can easily calculate an average function once we have figured out how to horizontally align the individual functions.

To align the shape functions we calculate their center of volume and then shift the functions horizontally until their center of volume agrees. This is essentially the method suggested in [37] for foliated triangulations. We have also experimented with an alterna-

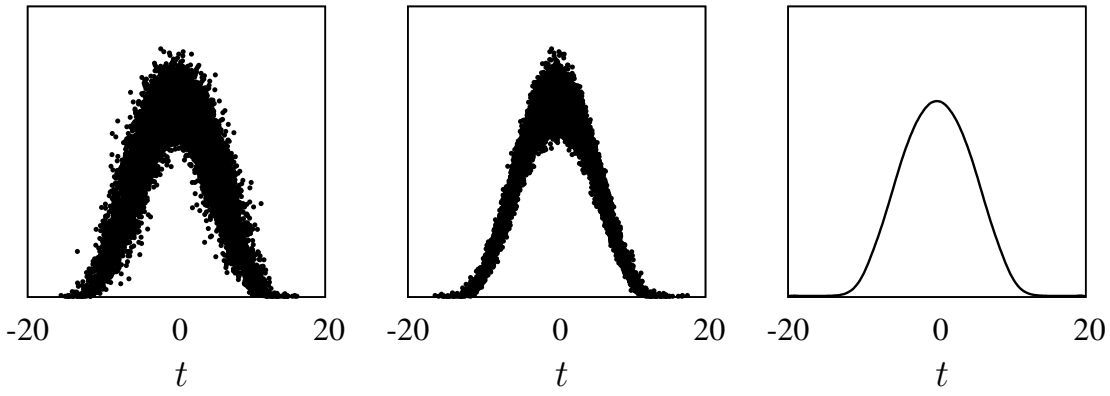


Figure 6.10: The left and center illustrations show a superposition of volume distributions. On the left no alignment has been performed while in the middle the center of mass of the volume distributions has been matched. The right picture shows the averaged volume distribution resulting from the averaging process described in the text.

tive alignment algorithm which attempts to use the visual information contained in the shape to produce a best overlap of the functions. In the cases we have investigated both methods produce virtually indistinguishable results.

Figure 6.10 illustrates the effect of the alignment algorithm. The left picture shows the naive superposition of volume distributions. In the center image the distributions have been horizontally aligned using the information which is computed during the averaging procedure. The right illustration shows the final result of the procedure which is the averaged volume distribution.

A phase of extended geometries

Figure 6.11 shows the average volume distributions for various values of the coupling k and for $\alpha = -1.0$. The figure shows that in all cases the average geometry is macroscopically extended and the average volume distributions show a characteristic blob-shape which strongly resembles the shapes known from CDT in the physically interesting phase [45]. We will perform a more detailed analysis of the average volume distributions further below.

We notice in Fig. 6.11 that with increasing k the time extension of the average geometry increases. It is interesting to examine the volume distributions close to the phase transition in detail. In Fig. 6.12 we have again plotted the distributions associated to $k = 0.0$, $k = 0.2$ and $k = 0.24$. We have enlarged the scale on the t-axis and have also slightly offset the t-axis in vertical direction. We observe that as one approaches the phase transition the emergent geometry develops a tail at both ends of the volume profile. This tail is similar to the tube observed in simulations of CDT. There the tube is interpreted as an algorithmic artifact because the manifold constraints and the fixed number of spatial slices prevent the tail to shrink to zero size. In the locally causal case the situation is less clear because the tail develops spontaneously as we move from $k = 0.0$ towards the phase transition.

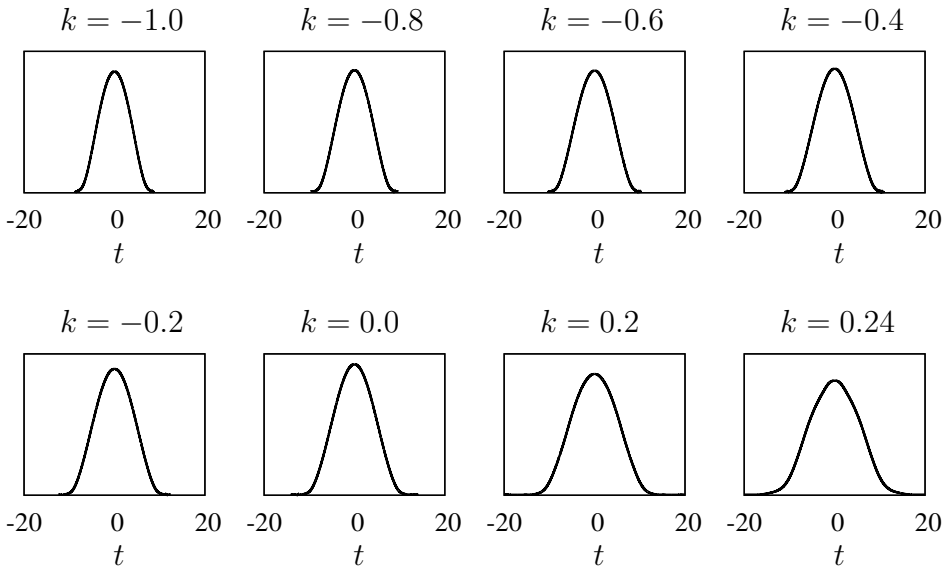


Figure 6.11: The average volume distributions measured at $\alpha = -1.0$ for various values of k. The scale on the vertical axis is the same for all plots. Some of these volume distributions have a tail which is not visible here, but which will be shown in Fig. 6.12.

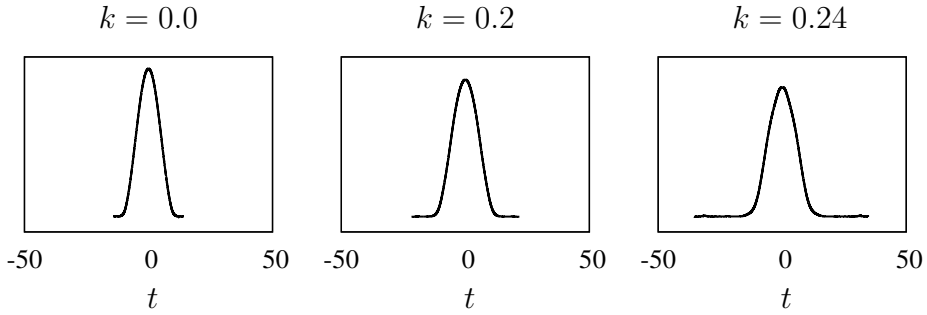


Figure 6.12: The final three average volume distributions in Fig. 6.11 are shown here again but with a different scaling on the t-axis and with a small displacement of the t-axis to make the tail of the distributions better visible.

At this point it is interesting to study the qualitative change of the volume distribution as one crosses the phase transition. Figure 6.13 shows the volume distribution of a single triangulation drawn from a simulation at $k = 0.4,\ \alpha = -1.0$ with 40000 tetrahedra. We observe that the vast majority of spatial slices have minimal or almost minimal size. The minimal size for a spatial slice in 2+1 dimensions is four. The whole triangulation thus forms one single very long tube with minimal spatial extent almost everywhere. We have checked at this point in the phase diagram that the time extension of the tube scales linearly with the system size. We are thus led to the interpretation that the average geometry in the infinite-volume limit is a one-dimensional purely timelike string.

We can now summarize our findings. At all the investigated locations we have found average geometries which are macroscopically extended with a blob-shaped volume distribution. The time extension of the average geometry increases with increasing k. As

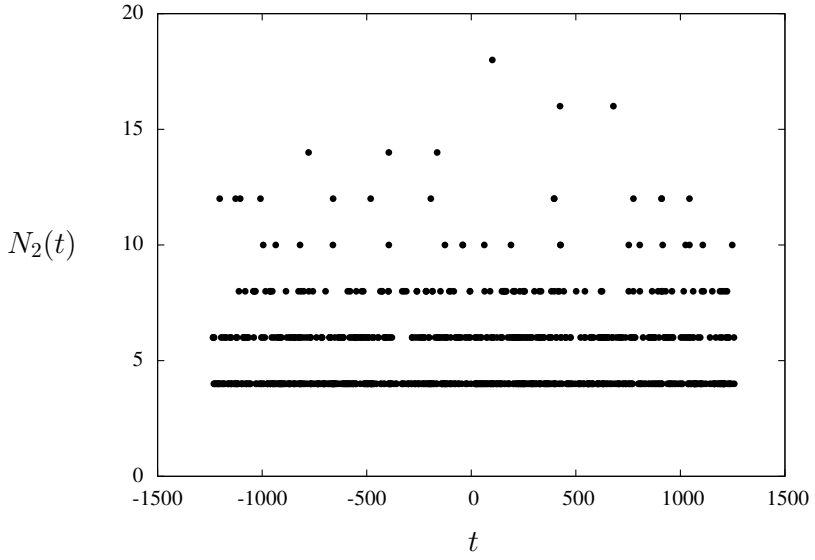


Figure 6.13: The volume distribution of a single triangulation extracted from a simulation at $k = 0.4,\ \alpha = -1.0,\ N_3 = 40000$ in the high vertex density phase.

one approaches the phase transition the geometry spontaneously develops a tail. On the other side of the transition we find geometries forming a long timelike tube.

Evidence for $d = 3$ from finite-size scaling

In order to get more information from the average volume distributions we have run another series of simulations. We have chosen six points on the k-axis (ranging from $k = -1.0$ to $k = 0.0$) and six on the α-axis (ranging from $\alpha = -0.52$ to $\alpha = -1.0$). At every point we have run four simulations with different system sizes ($N_3 = 40000$, 80000, 120000 and 160000).

On the left of Fig. 6.14 we see the average volume distributions extracted from the four simulations at $k = 0.0,\ \alpha = -1.0$. Now we perform a finite-size scaling analysis of these distributions in an analogous way to what was done in [37] for foliated triangulations in 3+1 dimensions. Assuming the average geometry has a macroscopic dimension d we expect that time intervals scale as $N_3^{1/d}$ and that spatial volumes scale as $N_3^{(d-1)/d}$. If we plot the distributions with axes rescaled accordingly, they should fall on top of each other.

In order to find an estimate for d we run an algorithm which scans through an interval of d-values by making steps of $\Delta d = 0.005$, which for each d-value measures how well the distributions overlap themselves. We use a standard least-squares measure with appropriate normalization to measure the quality of overlap. The value of d which minimizes the measure is taken as an estimate for the mascroscopic dimension. In our particular case the algorithm yields a best estimate of $d = 2.91$ at $k = 0.0$. The plot on the right of Fig. 6.14 shows all four distributions with axes rescaled using the best estimate for the dimension. We see that the overlap is virtually perfect.

We have repeated the analysis for the other points in the phase diagram. Figure 6.15

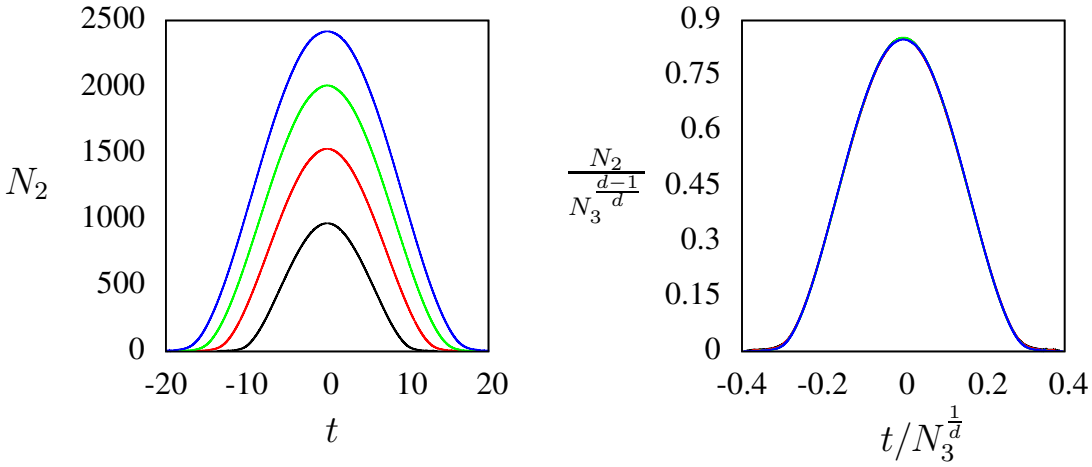


Figure 6.14: The average volume distributions at $k = 0.0$, $\alpha = -1.0$ for four different system sizes ($N_3 = 40000$, 80000, 120000 and 160000). On the left the axes are unscaled while on the right they have been rescaled with $d = 2.91$ to achieve a best overlap of the distributions.

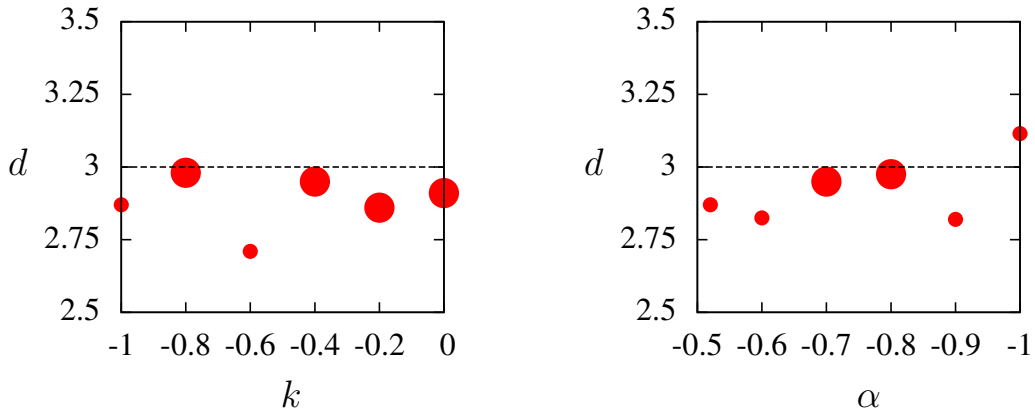


Figure 6.15: Estimates for the macroscopic dimension from finite-size scaling. The left measurements are performed on the k-axis with $\alpha = -1.0$ and the right ones on the α-axis with $k = 0.0$. The large dots represent measurements where the volume distributions overlap with excellent quality, the smaller dots stand for lesser quality.

shows the calculated estimates for the macroscopic dimension. The measurements on the k-axis are shown on the left and those on the α-axis on the right. Those measurements where the overlap has excellent quality are shown with large dots, the remaining ones with small dots. We observe that all of the six high-quality measurements yield macroscopic dimensions between $d = 2.85$ and $d = 3.00$ while the values coming from the other six measurements are more spread.

The plots shown in Fig. 6.15 do not contain error bars. The standard bootstrap algorithm gave errors which underestimate the true errors by at least one order of magnitude, which indicates that there are significant systematic errors not accounted for by the bootstrap method. In order to investigate this issue we have studied how the dimension

observable evolves with Monte Carlo time. More precisely, we can define a dimension $d(t_{MC})$ by cutting away all Monte Carlo data with $t > t_{MC}$ before performing all calculations. The quantity $d(t_{MC})$ can be plotted as function of the Monte Carlo time t_{MC} to study the convergence of the observable. From studying these plots it appears that the convergence of the dimension observable is very slow, possibly delayed by metastabilities.

Furthermore, the calculation of the dimension observable is highly nontrivial and there are several algorithmic choices which affect the final result. First we needed to define a time coordinate, which we did by calculating the average distance between a vertex and the two poles of the 3-sphere. To assign a time coordinate to the spatial slices we averaged over the time coordinates of the vertices in the slice. Finally, we ran a sophisticated averaging algorithm to produce the final distributions. This whole process is highly nonunique and therefore it is possible that these effects contribute to the systematic errors which are not captured by standard error algorithms such as the bootstrap method.

These problems make clear that one single dimension measurement does not provide sufficient evidence to support the $d = 3$ hypothesis. On the other hand, all twelve results combined strongly suggest that the average geometries in the low vertex density phase are three-dimensional. In the following part we will see that the functional form of the volume distributions is also compatible with $d = 3$.

Comparison with the 3-sphere

In [37] the average volume distributions of foliated triangulations in 3+1 dimensions have been compared to the volume profile of the 4-sphere which is the Wick-rotated version of the de Sitter spacetime. More precisely, the distributions coming from the simulations have been compared to a volume profile of the form $V_3(t) = a\cos^3(bt)$, which is the profile of a 4-sphere and where a and b depend on the overall size of the 4-sphere and on a finite relative scaling between the spatial and the timelike extent. It was found that the $\cos^3$ profile can be fitted with high accuracy to the average volume distributions extracted from the simulations.

In the following we will attempt a similar analysis with the average volume distributions from our simulations of locally causal triangulations. The volume profile of the 3-sphere has the functional form $V_2(t) = a\cos^2(bt)$ where a and b are constants. We have chosen those locations in the phase diagram where the rescaled average volume distributions overlap with excellent quality and have attempted to fit the curves using the $\cos^2$-ansatz with two free parameters.

Figure 6.16 shows the result at the point $k = 0.0,\ \alpha = -1.0$. It is clear that only the part of the fit function which is between the two minima is relevant for the comparison. We see that the $\cos^2$-ansatz fits the average volume distributions almost perfectly, except at both ends, where the simulation data show a small tail which is not present in the fit function once we cut away the parts outside the two minima. We have already observed earlier and illustrated in Fig. 6.12 that the average geometries develop a tail as one moves towards the phase transition.

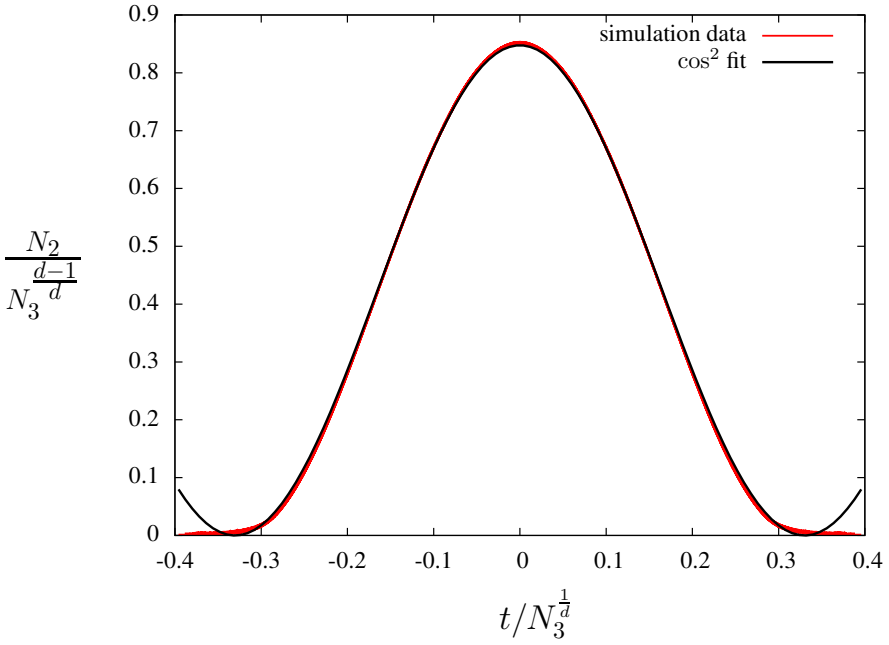


Figure 6.16: The rescaled average volume distributions at $k = 0.0,\ \alpha = -1.0$ and the best fit using the $\cos^2$-ansatz, corresponding to $d = 2.91$.

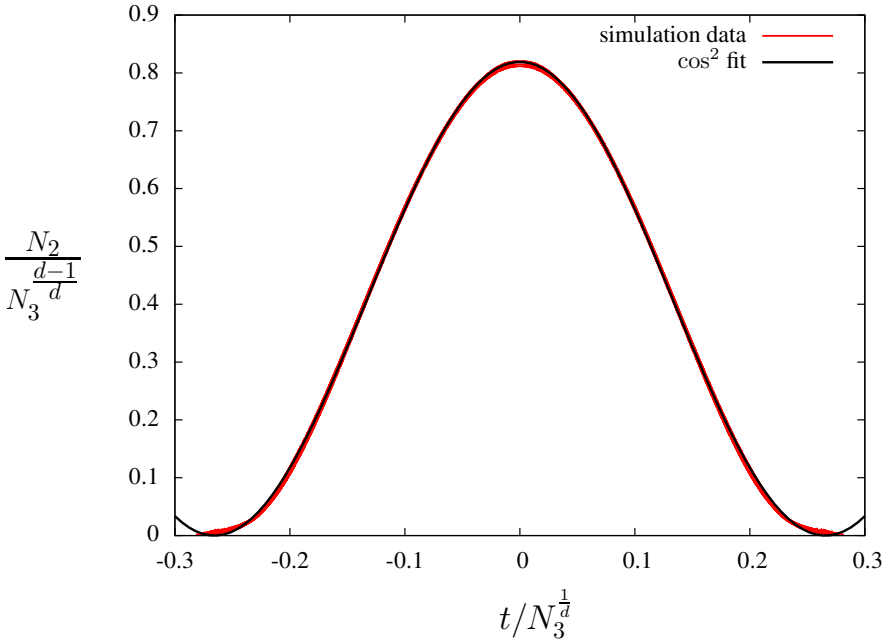


Figure 6.17: The rescaled average volume distributions at $k = -0.8,\ \alpha = -1.0$ and the best fit using the $\cos^2$-ansatz, corresponding to $d = 2.98$.

It is interesting to consider a situation which is further away from the phase transition. Figure 6.17 shows the result of the same analysis for the phase diagram point $k = -0.8,\ \alpha = -1.0$. We again observe an almost almost perfect fit in the region of non-minimal spatial extent, but now even the time extension of the average geometry and

the fit function agree, once we cut away the part of the $\cos^2$-function outside the two minima. The quality of the fit becomes slightly reduced towards both ends, which is not surprising because discretization effects become large when the spatial extent is small.

6.6 Summary

In this and the preceding chapter we have presented a new model of dynamical triangulations in 2+1 dimensions, which is locally causal and where no foliation is enforced at the kinematical level. This new model of locally causal triangulations has a richer kinematical structure compared to its foliated counterpart, with new objects such as bubbles and pinchings of spatial sections. The number of subsimplex types has increased from seven to ten while the number of kinematical constraints has stayed at five. The dimensionality of the space of actions linear in the subsimplex counts has thus increased from two to five, providing an a priori significantly enlarged phase space to be explored.

We have concentrated our efforts on the subspace characterized by actions of Regge type which are straightforward discretizations of the Einstein-Hilbert action. After fixing the total system size, this subspace becomes two-dimensional and is spanned by the inverse Newton coupling and the coupling α, which characterizes the anisotropy between space and time in the regularized sector of the theory. We have derived the condition $1/2 < \alpha < 3$ for the existence of a Wick rotation which implies that the phase diagram has two boundaries.

We have chosen to reduce the full ensemble of locally causal triangulations because the presence of particular kinematical structures of global nature prevented the unconstrained simulations from proper thermalization. This reduction allowed us to investigate the region $1/2 < \alpha \leq 1$ of the phase diagram, while we still observed severe thermalization problems in the complementary region. We have run our simulations with the spacetime topology of the 3-sphere with a source and a sink of (Euclidean) time at the two poles.

We have found two phases of geometry with low and high vertex density. The analysis of tetrahedron distributions shows that the triangulations in the low vertex density phase are weakly foliated and we observe the emergence of almost perfectly foliated states close to the boundary $\alpha = 1/2$ of the phase diagram. We have constructed a volume distribution observable and an averaging procedure to study the average volume profiles of the emergent geometries in the weakly foliated phase. A finite-size scaling analysis strongly suggests that the geometries are three-dimensional. We have also fitted the profiles with a $\cos^2$-ansatz to compare them with the geometry of the classical 3-sphere and found an almost perfect agreement. We have repeated the analysis for various points in the phase diagram with consistent results.

These results indicate that the low vertex density phases of locally causal DT and CDT in 2+1 dimensions have a similar low-energy behavior. It leads us to conjecture that the same may be true in 3+1 dimensions and that the model of locally causal triangulations may be as compatible with cosmological observations as CDT. If these expectations

are verified we can conclude that the foliation is not strictly necessary to get interesting physics out of a model of dynamical triangulations.

One of the new challenges coming with the locally causal triangulations are the increased algorithmic dependences when measuring observables. In order to measure the average volume profile we have developed several nontrivial and nonunique algorithms. First we have introduced a time labeling by measuring the average distance between each vertex and the two poles of the 3-sphere. Then we have used another algorithm which generates a set of spatial slices using statistical methods. Afterwards, each slice has been assigned a time by averaging over the times of the individual vertices. Finally, to produce the averaged volume profiles we have used a sophisticated averaging procedure which involves spline interpolation and center-of-volume matching. All these algorithmic choices make it difficult to define sensible errors which accurately quantify the uncertainty of the measurement.

The study of these algorithmic dependences is certainly an important task for future research. Another important issue is how to solve the remaining technical problems which have prevented us from exploring the region $1 < \alpha < 3$ in the phase diagram. One possibility is to implement even more Monte Carlo moves which increase the mobility of the Monte Carlo process in order to improve thermalization. If these problems can be overcome we can start exploring the full phase diagram and possibly even go beyond the subspace of Regge actions.

Ultimately we would like to analyse the locally causal model in 3+1 dimensions. For several reasons this is highly challenging. The Monte Carlo moves for 3+1 dimensions will be composed of generalizations of those from 2+1 dimensions and possibly new types of moves. We have tried to develop the 2+1 moves in such a way that they can be generalized to higher dimensions without too much effort. At this stage, we have not been able to do so completely, as will be explained in the following chapter.

Another problem is that the sheer complexity of the implementation of the locally causal model makes the simulation run slower compared to the foliated case. In 2+1 dimensions the performance was sufficient to get very good results for observables such as the volume distribution. One can expect that the situation gets worse in 3+1 dimensions, where even simulations of CDT typically require running times of the order of several weeks or even months on contemporary hardware. On the positive side it may be possible that the finite-size effects are reduced in the locally causal model, based on the general experience that less constrained systems have smaller finite-size effects.

7

Locally causal DT in 2+1 dimensions: Implementation

A key requirement of any scientific analysis is *reproducibility* and therefore an important aspect of scientific publications is to provide sufficient information to allow other researchers to reproduce the results. The latter condition becomes problematic when numerical results are published which depend on sophisticated software which was specifically built to investigate the problem at hand. In this case it is even more imperative to provide extensive information which allows other researchers to rebuild the software in a reasonable amount of time such that the original results can be verified independently. In the case of CDT in higher dimensions such a verification has indeed been done [90, 91].

The model of locally causal dynamical triangulations is significantly more complicated compared to CDT and the implementation of the associated simulation software has been highly challenging even in 2+1 dimensions. In addition to generalizations of the Monte Carlo moves known from CDT it was necessary to consider moves which collapse of a link. The software implementation of these new moves is highly nontrivial and we devote much space in this chapter to explaining the key algorithms which are required to implement these Monte Carlo moves.

We begin with a review of the Monte Carlo method and elaborate on some of its aspects in greater detail to facilitate the discussion of the complex moves later on. Then we introduce generalizations of those moves which are implemented in simulations of CDT. Afterwards we discuss three types of link collapse moves, including the graph algorithms which form the backbone for the implementation of these moves. Then we describe a special type of move which operates close to sources and sinks of time. Finally we describe the details of an algorithm which is needed to generate volume distributions.

7.1 The Monte Carlo method

We use Monte Carlo simulations to investigate the dynamical properties of locally causal triangulations of Type 1. Thus we generate a sequence of configurations (triangulations) by applying Monte Carlo moves in such a way that the configurations appear with the Boltzmann probability. To prepare the discussion of our moves we first need to recapitulate some of the basics of the Monte Carlo method. We follow the excellent book by Newman and Barkema[1] [55].

Detailed balance

The two most important requirements for the algorithm to achieve the desired probability distribution of states are *ergodicity* and *detailed balance*. We have discussed the former already in section 6.1 and have explained that we are simulating a subensemble of the full ensemble of locally causal triangulations of Type 1. We proceed to discuss detailed balance, following [55]. Given two states μ and ν we define the *transition probability* $P(\mu \to \nu)$ as the probability that the state ν is generated if the system was in the state μ before. The condition of detailed balance then reads

$$\exp(-S_\mu)P(\mu \to \nu) = \exp(-S_\nu)P(\nu \to \mu) \tag{7.1}$$

with S_μ being the action evaluated on the state μ. The standard way of implementing detailed balance is to execute the moves in two steps by first proposing a particular move and then by accepting or rejecting the chosen move. Formally speaking we break up the transition probability into two parts:

$$P(\mu \to \nu) = g(\mu \to \nu)A(\mu \to \nu). \tag{7.2}$$

The *trial probability* $g(\mu \to \nu)$ is the probability that the algorithm proposes a move leading to the state ν, if the system was in the state μ before. The object $A(\mu \to \nu)$ is the probability of accepting the proposed move.

By rearranging (7.1) and (7.2), the condition of detailed balance can be put into the following form:

$$\frac{A(\mu \to \nu)}{A(\nu \to \mu)} = \exp(-(S_\nu - S_\mu))\frac{g(\nu \to \mu)}{g(\mu \to \nu)} =: x\ . \tag{7.3}$$

A widely used choice to satisfy this condition is known under the name *Metropolis algorithm*:

$$A(\mu \to \nu) = \begin{cases} x & \text{if } x \leq 1 \\ 1 & \text{otherwise} \end{cases}. \tag{7.4}$$

This condition can be easily implemented by generating a random number $r \in [0,1]$ and by accepting the proposed move if $r \leq x$. In order to do this the quantity x needs to be

[1] We also adopt their use of the word 'state' to describe a configuration, rather than describing an element of a Hilbert space.

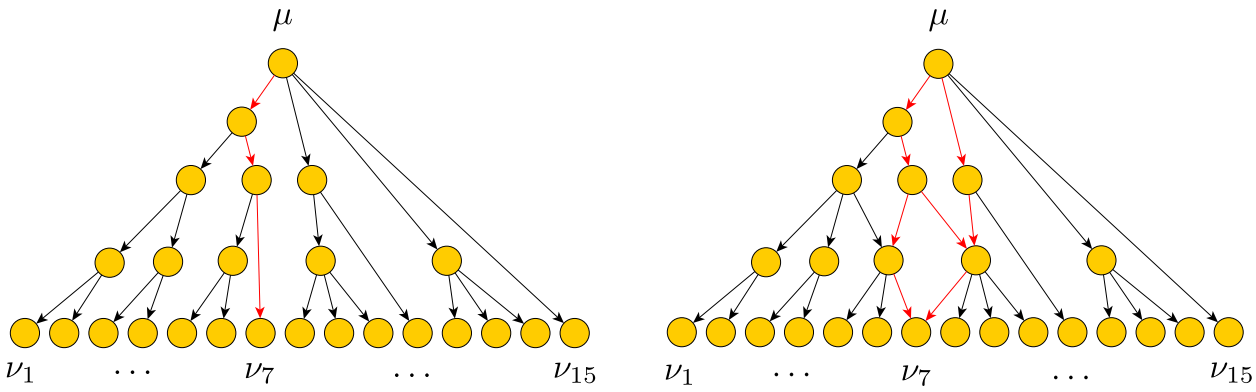


Figure 7.1: The process of proposing trial moves, visualized as directed graphs with nodes representing algorithm states. The two graphs represent two different algorithms.

determined first. In most cases it is easy to measure the action difference $S_\nu - S_\mu$ efficiently. For simple Monte Carlo moves it is also not too difficult to determine the trial probabilities. But as we shall see later on, some of the Monte Carlo moves defining our ensemble are very complex and the process of proposing a particular move and determining the trial probabilities becomes challenging. To prepare the discussion of those complex moves we need to discuss the process of proposing moves in greater detail, which we will do in the following.

Proposing trial moves

The simplest case can be illustrated with the example of the Ising model in two dimensions on a regular lattice where each site (vertex) contains a spin pointing up or down. Let us for simplicity assume a square lattice with N sites per row, then we have a total of N^2 spins. The simplest Monte Carlo move simply flips one particular spin in the whole system. Starting from any state it is clear that the total number of moves is simply N^2. The process of proposing a move consists solely of the choice of the spin to be flipped. The trial probability $g(\mu \to \nu)$ is then $1/N^2$, but since the probability for the reverse move is the same, they both drop out of equation (7.3).

This simple example illustrates that the process of proposing trial moves involves making random choices. In the case of the Ising model only one choice was involved, but in general the number of choices involved can be higher. We can visualize the algorithmic process of proposing a transition $\mu \to \nu_i$ as rooted graph (see Fig. 7.1). The root node at the top represents the state of the algorithm at the beginning of the process. Subsequently the algorithm makes random choices, with every choice leading to a different algorithmic state. At the end of the process one of the final nodes is reached, each of these nodes representing one possible final (ensemble) state ν_i which is connected to the ensemble state μ. For example, if the process ends up at the final node labeled by ν_7 the algorithm has decided to propose the transition $\mu \to \nu_7$.

In Fig. 7.1 we have displayed two different processes. The graph on the left side is a directed tree which means that there only exists one path from the root node to every

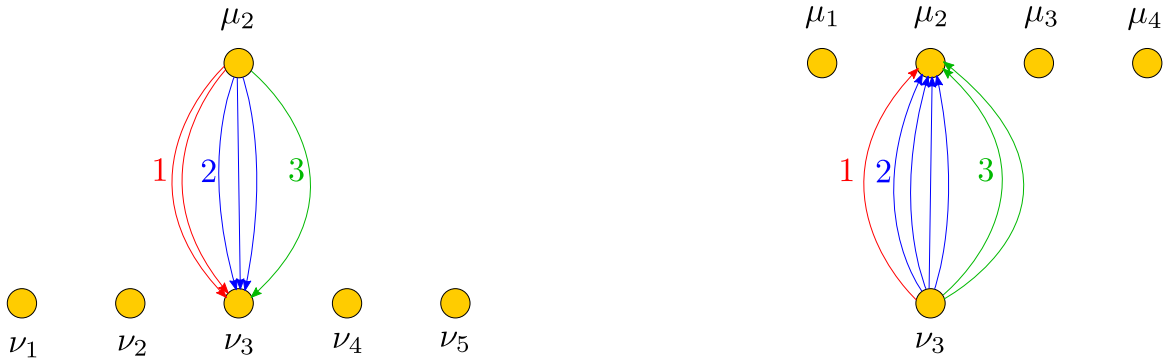


Figure 7.2: The process of proposing a transition $\mu_2 \to \nu_3$ (left) and its inverse (right), showing a symmetry which can be exploited. The paths between μ_2 and ν_3 have been assigned to three groups, labeled by 1,2 and 3. Intermediate nodes and paths not connecting μ_2 and ν_3 have been omitted.

final node. The path connecting the root node with the node ν_7 (marked with red arrows) involves making three choices. The trial probability $g(\mu \to \nu_7)$ is simply the product of the probabilities for each single choice.[2] The graph on the right side is not a directed tree. To compute the trial probability $g(\mu \to \nu_7)$ here, one needs to compute the probability for each path connecting μ with ν_7 and then sum up all the values. Since every path that needs to be taken costs computing time, it is in most cases advantageous to design algorithms which are as tree-like as possible.

Sometimes it is possible to avoid taking all paths from the root node to a particular final node. Let us consider a transition $\mu_2 \to \nu_3$ and its inverse (see Fig. 7.2). In both cases we have grouped all paths connecting μ_2 and ν_3. The precise way how such a grouping is achieved does not matter, except that the number of groups should be the same in both directions. The latter condition allows us to optimize the efficiency of the algorithm as follows. Let us define $P_i(\mu \to \nu)$ as the probability of reaching the state ν starting from the state μ under the condition that the algorithm chooses a path of the group i. Using $P(\mu \to \nu) = \sum_i P_i(\mu \to \nu)$ we see immediately that the condition

$$\exp(-S_\mu) P_i(\mu \to \nu) = \exp(-S_\nu) P_i(\nu \to \mu) \qquad \forall i \tag{7.5}$$

is sufficient for detailed balance (7.1) to be satisfied.[3] We can therefore calculate the trial probabilities without having to sum up the contributions of the paths which belong to another group as the one which has been chosen by the algorithm. The primary challenge to take advantage of this technique is to find a grouping of paths which admits a bijection between the move and its inverse and which is not too complicated to implement. We will demonstrate the use of this technique when we discuss our Monte Carlo moves.

[2] The inverse of this product is sometimes called the Rosenbluth weight and was first introduced in [92].
[3] We thank Timothy Budd for making us aware of this fact.

Computing the action difference

In [34] f-vectors were used as a convenient notation to describe the changes made by individual Monte Carlo moves. We define the f-vector of a triangulation T as

$$f = (N_0, N_1^s, N_1^t, N_2^{sss}, N_2^{tss}, N_2^{tts}, N_3^{T_2}, N_3^{T_3}, N_3^{T_5}, N_3^{T_9}).$$

The elements of the vector count the number of all subsimplex types and we use the same notation here as in section 5.4, namely,

$$\begin{aligned}
N_0 &= \text{number of vertices} \\
N_1^s &= \text{number of spacelike links} \\
N_1^t &= \text{number of timelike links} \\
N_2^{sss} &= \text{number of triangles with three spacelike links} \\
N_2^{tss} &= \text{number of triangles with two spacelike links} \\
N_2^{tts} &= \text{number of triangles with one spacelike link} \\
N_3^{T_2} &= \text{number of tetrahedra of type } T_2 \\
N_3^{T_3} &= \text{number of tetrahedra of type } T_3 \\
N_3^{T_5} &= \text{number of tetrahedra of type } T_5 \\
N_3^{T_9} &= \text{number of tetrahedra of type } T_9.
\end{aligned}$$

Every Monte Carlo move changes the f-vector of the whole triangulation:

$$f' = f + \Delta f.$$

The quantity Δf and the values of all the couplings are sufficient to compute the action difference caused by any Monte Carlo move.

7.2 The generalized Pachner moves

The ensemble of foliated triangulations forms a subset of the ensemble of locally causal triangulations. Therefore it is natural to implement the Monte Carlo moves which were already used for simulations of CDT. These are modified versions of the original Pachner moves for Euclidean triangulations [93, 94]. A description of these modified Pachner moves can be found in [34]. All of these moves change the interior structure of a small compact region of the discrete manifold, leaving the boundary of the region invariant. In the case of CDT the type of all newly created links is uniquely determined. As we shall see, in the case of the larger ensemble of locally causal triangulations there exist some moves where the types of the new links are not determined so that the process of proposing the move involves additional choices.

To facilitate the implementation of these moves it is convenient to introduce some notation. Let n_{bdy} be the number of links belonging to the boundary of the compact

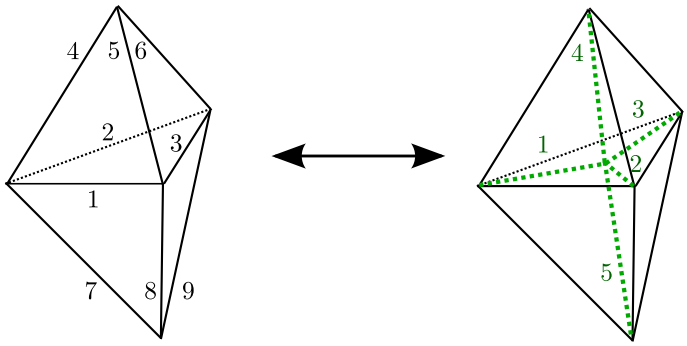


Figure 7.3: The 2-6 move. The labeling of the boundary links on the right-side is the same as on the left-hand side, but has not been drawn to improve visualization. The numbers shown on the right label the interior links which are created by the move.

region. We label all the links from 1 to n_{bdy} and introduce the vector l_{bdy} of size n_{bdy} whose i-th element is either s or t depending on the type of the link which was labeled with i. Spacelike links are marked with s and timelike links with t. A typical example is $l_{\text{bdy}} = (\text{s}, \text{s}, \text{s}, \text{t}, \text{t}, \text{t}, \text{t}, \text{t})$. Likewise we define the vectors $l_{\text{int},1}$ and $l_{\text{int},2}$ to describe the links inside the compact region before resp. after the move.

For every generalized Pachner move there exist many inequivalent link vector combinations, but only a subset of them is compatible with causality. To find out whether a move respects causality one has to check two things. First, all the newly created tetrahedra need to have a valid type according to table 5.1. Second, the neighborhood of every spacelike link needs to be such that one encounters exactly four light cone crossings when circling around that link.

If the link lies on the boundary of the compact region which is changed by the move it is sufficient to show that the number of light cone crossings is preserved, since we can assume that we start from a triangulation which respects local causality. As a consequence it is sufficient to only consider that part of the link neighborhood which is inside the region that is affected by the move. These two simplifications allow for a performance optimization using the technique of precomputing. Before the simulation effectively starts, a procedure is called which performs the causality check for every possible link vector combination and stores the result into a table. Later when the generalized Pachner moves are executed the causality check is done very quickly by consulting the table.

In section 5.2 we have introduced the notions of 'link causality' and 'vertex causality'. They ensure the existence of a well-defined light cone at every link and vertex respectively. We have formulated the conjecture that link causality may imply vertex causality, based on our observation that none of the Monte Carlo moves discussed in this chapter destroys vertex causality while maintaining link causality. Therefore we do not explicitly implement the vertex causality condition, but during the simulation we periodically verify that vertex causality is indeed satisfied at every vertex.

l_{bdy}	$l_{\text{int},2}$	Δf
(s,s,s,t,t,t,t,t)	(s,s,s,t,t)	(1,3,2,2,0,6,0,0,4,0)
(s,s,s,t,t,s,t,t,t)	(s,s,s,t,t)	(1,3,2,2,1,5,0,1,3,0)
(s,s,s,t,s,s,t,t,t)	(s,s,s,t,t)	(1,3,2,2,2,4,0,2,2,0)
(s,s,s,t,t,s,t,s,t)	(s,s,s,t,t)	(1,3,2,2,2,4,0,2,2,0)
(s,s,s,t,t,s,t,t,s)	(s,s,s,t,t)	(1,3,2,2,2,4,0,2,2,0)
(s,s,s,t,s,s,s,t,t)	(s,s,s,t,t)	(1,3,2,2,3,3,0,3,1,0)
(s,s,s,t,s,s,t,s,t)	(s,s,s,t,t)	(1,3,2,2,3,3,0,3,1,0)
(s,s,s,t,s,s,s,s,t)	(s,s,s,t,t)	(1,3,2,2,4,2,0,4,0,0)
(s,s,s,t,s,s,t,s,s)	(s,s,s,t,t)	(1,3,2,2,4,2,0,4,0,0)

Table 7.1: All inequivalent link vector configurations of the 2-6 move which respect the causality constraint. The third column displays the f-vector change caused by the move.

The 2-6 move

Figure 7.3 visualizes how the 2-6 move works. The triangle containing the boundary links 1-3 is deleted and a vertex and five links are created. Each new link connects the new vertex with one of the already existing vertices. The original two tetrahedra get replaced by six new ones. The 2-6 move is the only generalized Pachner move which creates a new vertex. Table 7.1 lists all the inequivalent link vector combinations which are compatible with the causality constraint. We consider two moves to be equivalent if their link vectors are mapped to each other by applying a symmetry transformation to the boundary of the simplicial complex that is changed by the move.

Since the 2-6 move creates five new links, it follows that for every link vector l_{bdy} there are a priori 32 possibilities how to execute the move. Interestingly, it turns out that only one of these possibilities is compatible with the causality constraint. Furthermore, table 7.1 shows that the triangle which contains the boundary links 1,2 and 3 has to be spacelike and that during the move this triangle is subdivided into three spacelike triangles. This fact can be used to simplify the implementation of the move. There are various ways of how to implement the move depending on the available data structures. In any case the computation of the trial probabilities is straightforward and can be done in a similar way as in the case of foliated triangulations.

The 2-3 move

The 2-3 move operates on the same simplicial complex as the 2-6 move. The triangle containing the boundary links 1,2 and 3 is deleted and a new link is created which connects the two vertices that do not belong to the triangle just deleted. Figure 7.4 shows the visualization of the move and table 7.2 displays all equivalent link vectors which

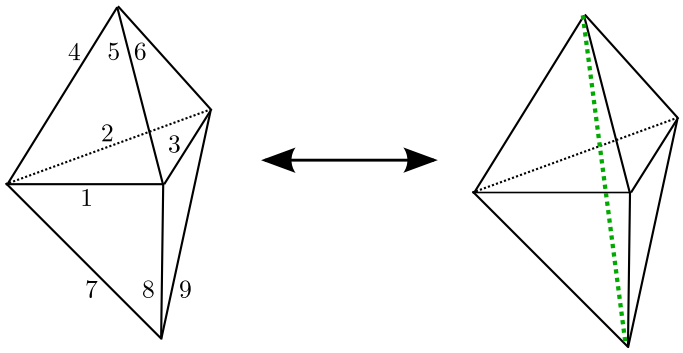


Figure 7.4: The 2-3 move. The labeling of the boundary links on the right side is the same as on the left side.

l_{bdy}	$l_{\text{int},2}$	Δf
(t,s,s,t,s,s,s,t,s)	(t)	(0,0,1,0,0,2,0,0,0,1)
(t,t,s,t,s,s,s,t,t)	(t)	(0,0,1,0,0,2,0,0,0,1)

Table 7.2: All inequivalent link vector configurations of the 2-3 move which respect the causality constraint. The third column displays the f-vector change caused by the move.

are compatible with the causality constraint. The newly created link has to be timelike which can be exploited in the implementation of the inverse move. There are again no particular difficulties concerning the computation of the trial probabilities.

The 4-4 move

The 4-4 move applies to the situation where a link with coordination number four is present, like the interior link in Fig. 7.5 on the left. That link is then deleted and replaced by another link which connects two vertices that do not belong to the link just deleted. Here we encounter a situation which was not present in the case of foliated triangulations in the sense that there are now two possibilities of where to place the new link. Figure 7.5 displays one of these two possibilities, but the link can also be placed such that it connects the two vertices at the top and the bottom.

We deal with the situation as follows. First we choose the link to be deleted and then we make another (binary) choice. Depending on the outcome of this choice, we either execute the move as shown in the figure or we rotate the labeling of the simplicial complex first by 90 degrees before executing the move. The trial probabilities acquire an extra factor of 0.5, but since the same happens for the inverse move, the factor effectively drops out of the detailed balance equation and can be omitted.

Table 7.3 shows all inequivalent link vectors which are compatible with the causality constraint. We have only included the contributions from one of the two variants of the move since both are related by a trivial 90 degree rotation. At the bottom of the table we

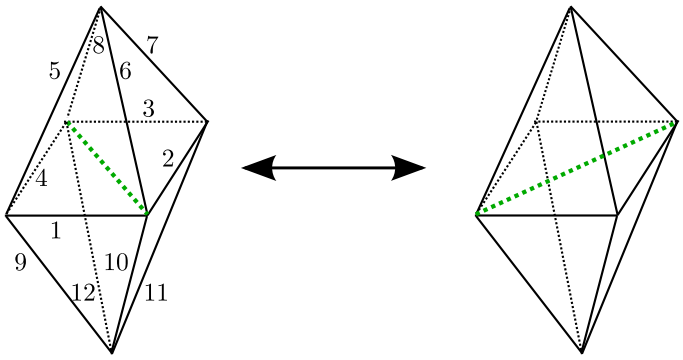


Figure 7.5: The 4-4 move. The labeling of the boundary links on the right side is the same as on the left side.

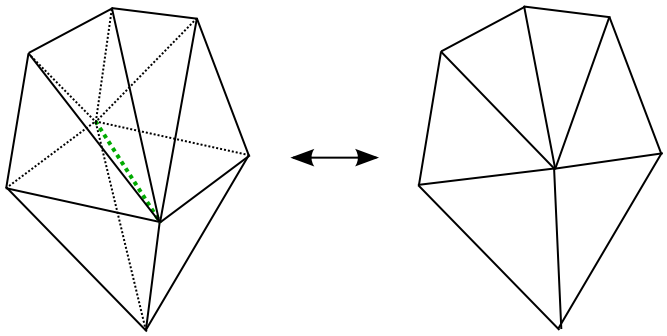


Figure 7.6: The link collapse move.

find two pairs of moves where each member of the pair operates on the same simplicial complex but the type of the new link is different. In such a situation an additional binary choice needs to be made and the trial probability acquires a factor of 0.5. Again the factor can be omitted because it also appears for the inverse move. The table also reveals that some of these moves split or join bubbles. As we have discussed in section 6.1, we have restricted our ensemble by forbidding such moves because we encountered severe thermalization problems using the larger ensemble. We have included these moves in table 7.3 for completeness, but we currently do not use them in our simulations.

7.3 The link collapse moves

As we have seen, the generalized Pachner moves change the interior of a compact subtriangulation leaving its boundary invariant. Another type of move involves collapsing a link to a vertex (see Fig. 7.6). Just like in the case of the generalized Pachner moves, the link collapse move exists in several variants depending on the types of the links which are involved in the move. We have implemented three of these variants and will describe them further below.

One problem common to all the variants of the link collapse move is to determine whether the collapse of the link is allowed. The answer depends on whether one works

l_{bdy}	$l_{\text{int},1}$	$l_{\text{int},2}$	Δf	splits/joins bubbles
(s,s,s,s,t,t,t,s,t,s,t,t)	(s)	(s)	(0,0,0,0,-2,2,0,-2,2,0)	no
(s,s,s,s,t,t,t,s,t,t,t,s)	(s)	(s)	(0,0,0,0,-2,2,0,-2,2,0)	no
(s,s,s,s,s,t,t,s,t,s,t,t)	(s)	(s)	(0,0,0,0,-1,1,0,-1,1,0)	no
(s,s,s,s,s,t,t,s,t,t,t,s)	(s)	(s)	(0,0,0,0,-1,1,0,-1,1,0)	no
(s,s,s,s,t,t,t,s,t,t,t,t)	(s)	(s)	(0,0,0,0,-1,1,0,-1,1,0)	no
(s,s,s,s,s,t,t,s,s,s,t,t)	(s)	(s)	(0,0,0,0,0,0,0,0,0,0)	no
(s,s,s,s,s,t,t,s,s,t,t,s)	(s)	(s)	(0,0,0,0,0,0,0,0,0,0)	no
(s,s,s,s,s,t,t,s,t,s,s,t)	(s)	(s)	(0,0,0,0,0,0,0,0,0,0)	no
(s,s,s,s,s,t,t,s,t,t,t,t)	(s)	(s)	(0,0,0,0,0,0,0,0,0,0)	no
(s,s,s,s,t,t,t,s,s,t,t,t)	(s)	(s)	(0,0,0,0,0,0,0,0,0,0)	no
(s,s,s,s,t,t,t,t,t,t,t,t)	(s)	(s)	(0,0,0,0,0,0,0,0,0,0)	no
(t,s,t,s,s,s,s,s,s,t,t,s)	(t)	(t)	(0,0,0,0,0,0,0,0,0,0)	no
(t,s,t,s,s,t,t,s,s,t,t,s)	(t)	(t)	(0,0,0,0,0,0,0,0,0,0)	no
(t,s,t,s,s,t,t,s,t,s,s,t)	(t)	(t)	(0,0,0,0,0,0,0,0,0,0)	no
(t,s,t,s,s,s,s,s,s,s,s,s)	(s)	(s)	(0,0,0,0,0,0,0,0,0,0)	no
(t,s,t,s,s,s,s,s,s,s,s,s)	(t)	(t)	(0,0,0,0,0,0,0,0,0,0)	no
(t,s,t,s,s,s,s,s,s,s,s,s)	(s)	(t)	(0,-1,1,-2,0,2,-4,4,0,0)	yes
(t,s,t,s,s,s,s,s,s,s,s,s)	(t)	(s)	(0,1,-1,2,0,-2,4,-4,0,0)	yes

Table 7.3: All inequivalent link vector configurations of the 4-4 move which respect the causality constraint. The fourth column displays the f-vector change caused by the move.

with combinatorial or degenerate triangulations. In the latter case several subsimplices can contain the same vertices while this is forbidden in the former case. When using combinatorial triangulations one thus needs to check whether the state after the link collapse move is again a combinatorial triangulation.

Algorithm 1 shows how such a check can be implemented. It is essentially looking for pairs of links and triangles which share the same vertices after the move. It turns out that it is not necessary to look for pairs of tetrahedra sharing the same vertices after the move as we will now explain. Let ℓ be the link to be collapsed, let V_1 and V_2 be its vertices. Let as assume there exists a pair of tetrahedra $\{T_a, T_b\}$ with the vertices $\mathscr{V}_{T_a} = \{V_1, V_a, V_b, V_c\}$ and $\mathscr{V}_{T_b} = \{V_2, V_a, V_b, V_c\}$. If V_a, V_b or V_c is not contained in the star of ℓ, algorithm 1 will find double links and will fail. Therefore we only need to check the case where all these three vertices are contained in the star of ℓ.

If any of the links $\overline{V_aV_b}$, $\overline{V_aV_c}$, $\overline{V_bV_c}$ is not contained in the star of ℓ, algorithm 1 will find double triangles and will also fail. Thus we conclude that we only need to check the case

Input: ℓ=link to be collapsed
Output: **true**, if the link collapse preserves the combinatorial constraint, **false** otherwise
begin
  V_1, V_2=vertices of ℓ, with coord$(V_1) <$ coord(V_2)
  $\mathscr{S}_V$=set of all vertices contained in the star of ℓ
  $\mathscr{S}_L$=set of all links contained in the star of ℓ
  $\mathscr{T}$=set of all tetrahedra containing V_1
  foreach $T \in \mathscr{T}$ **do**
    $\mathscr{S}_T = \{\}$
    $\mathscr{V}_T$=set of all vertices of T, excluding V_1 and V_2
    foreach $V \in \mathscr{V}_T$ **do**
      if $V \notin \mathscr{S}_V$ **then**
        if $\exists$ link $\overline{VV_2}$ **then return** false
      else if $V \in \mathscr{S}_V$ **then**
        $\mathscr{S}_T = \mathscr{S}_T \cup \{V\}$
    $\mathscr{L}$ =set of all links whose vertices are in $\mathscr{S}_T$
    foreach $L \in \mathscr{L}$ **do**
      if $L \notin \mathscr{S}_L$ **then**
        if $\exists$ triangle LV_2 **then return** false
  return true
end

Algorithm 1: Test whether the collapse of a link preserves the combinatorial constraint.

where all of these links are contained in the star of ℓ. Since the three links form a cycle of length three, the coordination number of ℓ needs to be three as well, all of its faces on one side of the star are glued to T_a and the ones on the other side are glued to T_b. But T_a and T_b are also glued to each other through the triangle $V_aV_bV_c$, thus the simplical complex formed by the star of ℓ, T_a and T_b has no boundary. Assuming that the triangulation is connected, the only situation which is not dealt with by algorithm 1 is the one where the complete triangulation consists of five tetrahedra. This situation can be easily checked separately.

7.3.1 The pinching move

The ensemble of locally causal triangulations includes the ensemble of foliated triangulations. Therefore we need Monte Carlo moves which connect foliated triangulations with nonfoliated ones. One example is the pinching move, which we will first explain in 1+1 dimensions to prepare the discussion of the 2+1 dimensional case. Figure 7.7 shows a visualization of the move in 1+1 dimensions. The main idea of the pinching move is to collapse a timelike link to a single vertex. If we start from a foliated configuration, as

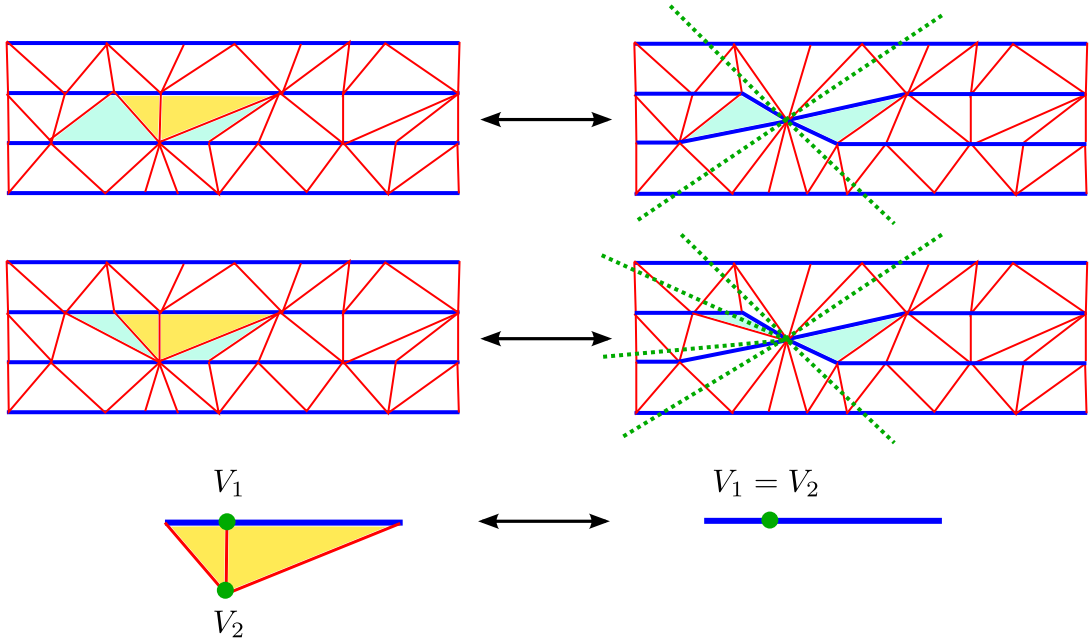


Figure 7.7: The pinching move in 1+1 dimensions. Spacelike links are drawn in blue, timelike links in red. The link $\overline{V_1V_2}$ collapses to a single vertex, creating a pinching. The dotted lines visualize the light cones at the pinching. The first situation respects causality while the second does not.

shown on the left of the illustration, then the result is a triangulation where two spatial slices touch at one vertex, thus creating a pinching which we have introduced in section 5.3.

Not every collapse of a timelike link leads to a triangulation which respects causality. Figure 7.7 illustrates both cases. In the second illustrated situation we see that we encounter six light cone crossings when circling around the pinching. In order to find out under which conditions we get a locally causal triangulation we need to have a closer look at the two triangles which are deleted by the move. These are colored yellow in the illustration. In the locally causal situation both triangles point downwards, but their neighbors, shown in light blue, both point upwards. This is no longer true in the second situation and it is easy to convince oneself that causality is respected if and only if both neighboring triangles point in the opposite direction from the triangles which are being deleted. It is also easy to verify that no further complications arise if we choose a more irregular configuration on the left of Fig. 7.7, as long as we do not change the link types of both yellow triangles. Note that the type of both blue triangles changes from tts to stt.

Let us proceed to the 2+1 dimensional pinching move illustrated in Fig. 7.8. The standard representation of the move in the upper half is complemented by a dual lattice representation in the lower half where dots represent tetrahedra and lines represent the triangles which connect pairs of tetrahedra. We see that the move collapses the link $\overline{V_1V_2}$ with the effect that the two spatial discs shown in the left top of the figure meet at one vertex after the move is executed.

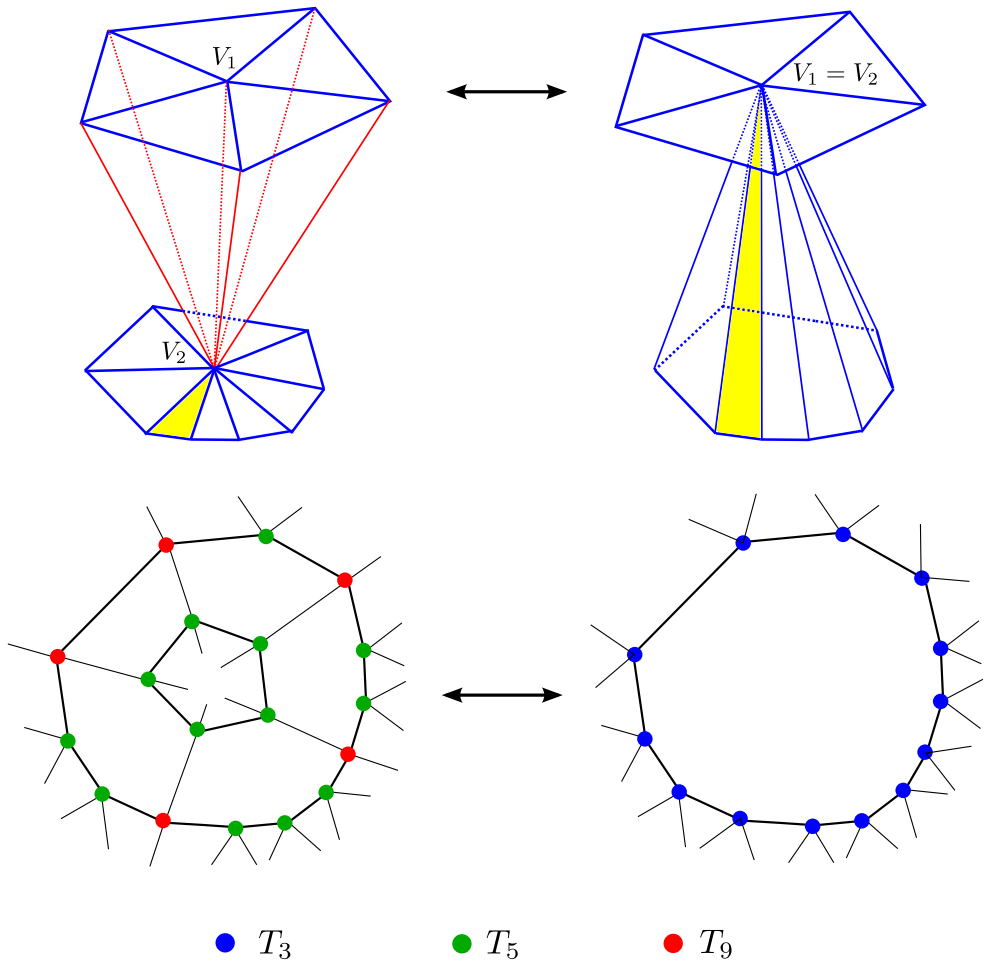


Figure 7.8: The pinching move in 2+1 dimensions. Spacelike links are drawn in blue, timelike links in red. The link $\overline{V_1V_2}$ collapses to a single vertex. The upper half shows the standard representation of the move while the lower half shows a dual lattice representation of the same move.

The coordination number of the link $\overline{V_1V_2}$ is equal to the number of tetrahedra which are deleted by the move. These tetrahedra form a ring in the dual lattice as shown in the left bottom part of the figure. Now we consider the set of all tetrahedra which do not belong to the ring and which share at least one timelike link with any of the tetrahedra in the ring. These tetrahedra form a second ring which encloses the first one.

The pinching move deletes the inner ring leaving only the outer ring. We know from the discussion in section 5.3 that a pinching is equivalent to a ring of tetrahedra of type T_3. Therefore we conclude that the types of the tetrahedra in the outer ring have to change to T_3 after the move is executed. This in turn fixes the types of the tetrahedra before the move is executed. It turns out that all the tetrahedra in the inner ring have to be of type T_5. The direct neighbors in the outer ring have to be of type T_9 and the rest of the tetrahedra have to be of type T_5, as illustrated by Fig. 7.8.

Let us denote by n_{T_5} and n_{T_9} the number of tetrahedra of types T_5 resp. T_9 in both

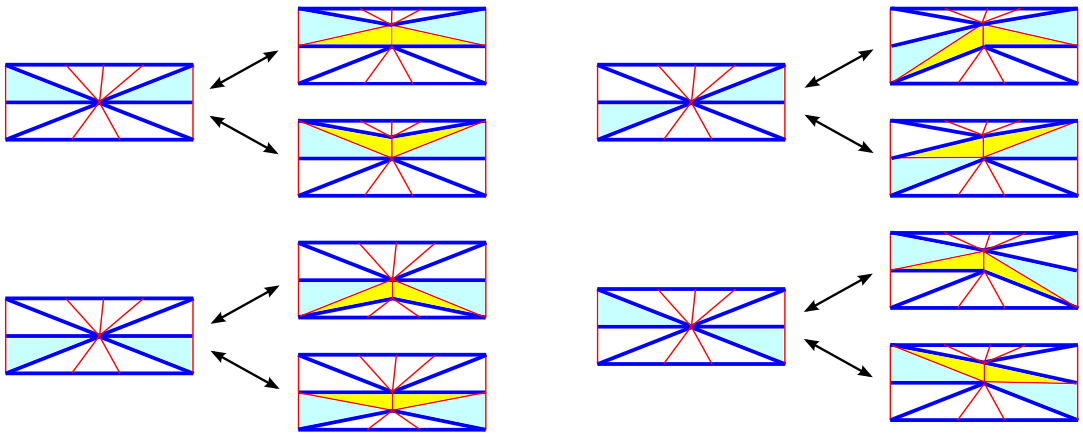


Figure 7.9: The inverse pinching move in 1+1 dimensions, showing various choices that have to be made in the implementation.

rings. The f-vector change associated with the pinching move is then

$$\Delta f = (-1, 0, -n_{T_9} - 1, 0, n_{T_5}, 2n_{T_9} - n_{T_5}, 0, n_{T_5}, -n_{T_5}, -n_{T_9}).$$

To implement the pinching move we conveniently start with the choice of a tetrahedron of type T_5, assuming it is one of those tetrahedra that will be deleted by the move. Then we check whether we have the situation as shown at the left bottom of Fig. 7.8 and reject the move if that is not the case. Afterwards we use algorithm 1 on page 97 to test whether the move connects two combinatorial triangulations. The rest of the implementation is straightforward, except for computing the trial probabilities which we will discuss below.

Now we need to discuss the inverse move which we first do in 1+1 dimensions. Figure 7.9 shows a situation where three spatial sections meet at a vertex. To execute the move we first choose a pair of triangles of type sst where one member of the pair belongs to the left sector and the other member belongs to the right sector as seen from the center vertex. This pair of triangles is shown in light blue in the figure. The inverse move changes the type of both selected triangles from sst to stt, but there are two inequivalent ways of how this can be done. We can create the two new triangles, shown in yellow in the figure, such that they point either up or down.

Only the moves shown on the left of Fig. 7.9 have analogues in 2+1 dimensions. The pairs of triangles shown in light blue correspond precisely to the rings of tetrahedra of type T_3. If $n+1$ spatial sections meet at a single vertex we have n rings of tetrahedra of type T_3. Thus a convenient way to implement the inverse move is to first select a tetrahedron of type T_3 and to reject the move if it does not belong to a ring where all other tetrahedra are also of type T_3. Every such ring contains two spatial discs and the implementation now has to choose one of them and to execute the inverse move as shown in the top part of Fig. 7.8. This second choice is precisely the analogue of the binary choice in 1+1 dimensions which makes the newly created triangles point up or down.

Finally, we discuss the computation of the trial probabilities. The implementation of the pinching move involves only one choice, namely, the choice of a tetrahedron of

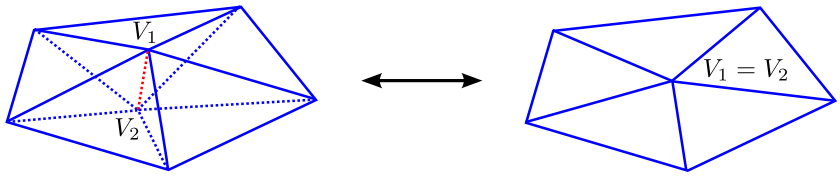


Figure 7.10: The bubble move in 2+1 dimensions. Spacelike links are drawn in blue, timelike links in red. The link $\overline{V_1V_2}$ collapses to a single vertex.

type T_5. This choice determines which tetrahedra are being deleted but then every other tetrahedron from this set could have been chosen yielding the same move. The trial probability thus becomes the number of tetrahedra to be deleted divided by the total number of tetrahedra of type T_5. The situation is similar for the inverse move where a tetrahedron of type T_3 is chosen first. The trial probability is then the number of tetrahedra of type T_3 in the selected ring divided by the total number of tetrahedra of type T_3 multiplied by 0.5, the latter factor accounting for the final binary choice which we have discussed previously.

7.3.2 The bubble move

Figure 7.10 shows an illustration of the bubble move. It is a link collapse move like the pinching move but the types of the involved links are different here. On the left side we see a ring of tetrahedra of type T_2 which forms a small bubble and we see a single timelike link in its interior. Since none of the links on the surface of the bubble change their type it is clear that no tetrahedra belonging to the surrounding of the bubble change their type. In particular the number of tetrahedra of type T_3 is kept invariant by the move. Therefore the bubble move is complementary to the pinching move where rings of tetrahedra of type T_3 are created or deleted and where the number of tetrahedra of type T_2 is kept invariant.

The move collapses the timelike link to a single vertex and as a result the bubble collapses to a spatial disc. The inverse move therefore involves choosing a vertex and a spatial disc with the vertex at its center. The situation is here much more difficult compared to the pinching move because the number of such spatial discs through a given vertex can become large and because there is no simple pattern which describes the set of all these spatial discs. We have to proceed in several steps and will begin with introducing local bubble graphs.

Bubble graphs of unit balls

In section 5.1 we have introduced bubble graphs where each vertex represents a bubble and each link an interface between two bubbles. Now consider the unit ball[4] of a vertex which is a triangulation with spherical boundary. We are interested in the bubble graph

[4]Here and in the following we use the term 'unit ball' to describe a Lorentzian sub-triangulation which is mapped to a Euclidean unit ball by the Wick rotation.

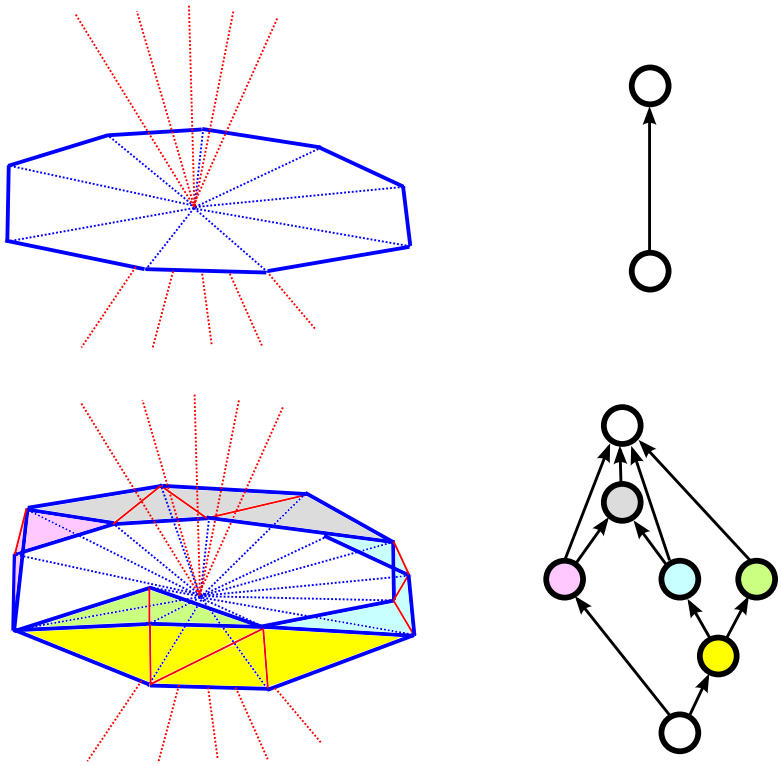


Figure 7.11: The spatial structure around a vertex and the associated local bubble graph. Spacelike links are in blue, timelike ones in red. The top example is known from foliated triangulations, the bottom example is only possible in nonfoliated triangulations.

associated to the unit ball. It is clear that a bubble appearing in such a unit ball is a partial bubble in the sense that it is the result of an intersection of a normal bubble in the full triangulation with the unit ball. It is also possible that several of these partial bubbles are parts of the same bubble in the whole triangulation.

Consider now the top part of Fig. 7.11 which shows the spatial structure around a vertex as it appears in foliated triangulations. In this case the spatial structure is very simple, it consists of a single spatial disc which contains all spacelike links emanating from the center vertex. The radial timelike links form two groups, the ones which are directed to the future and the others which are directed to the past. The associated bubble graph shown at the right top is very simple and consists only of two nodes and the edge connecting the nodes simply represents the spatial disc which is the interface between the future and the past bubble.

The bottom part of the figure shows the spatial structure around a vertex which does not satisfy the foliation condition. The local causality condition dictates that there exist two distinguished bubbles which contain all the radial timelike links just like in the foliated case. This implies that all local bubble graphs have two root nodes, one being a source and the other being a sink with respect to the directions in the graph. In the region between these two special bubbles the structure is now more complicated. We have additional bubbles, whose structure can be nicely seen when considering the surface of

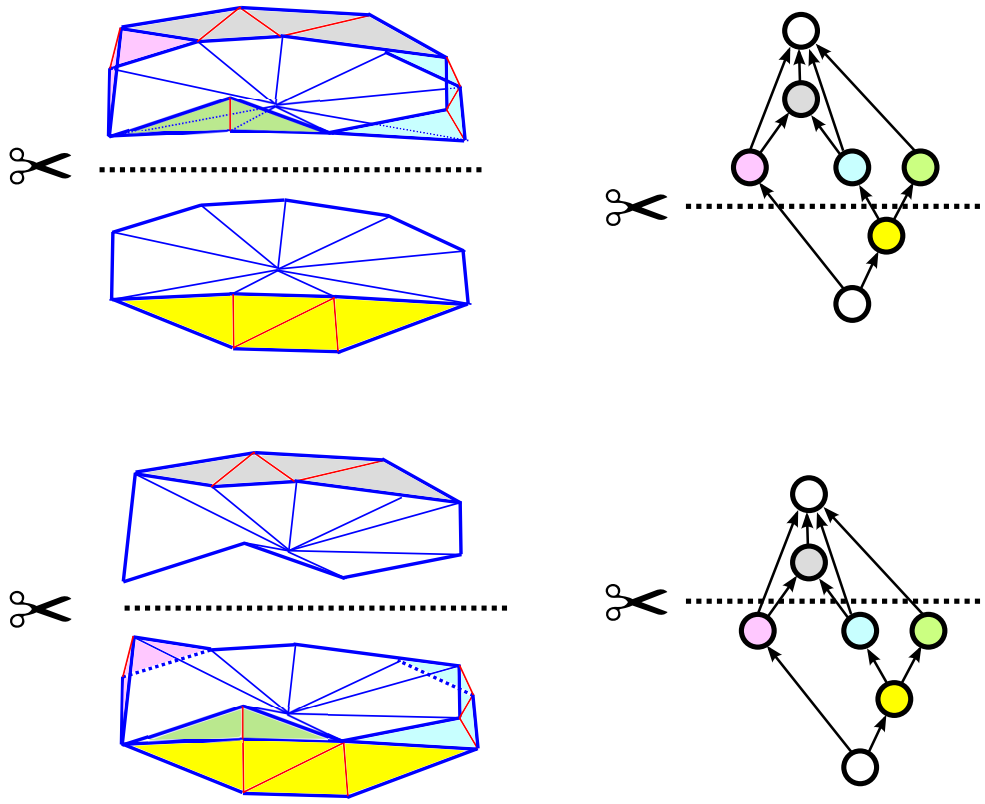


Figure 7.12: The unit ball which was shown at the bottom in Fig. 7.11 is cut in half in two different ways. In both cases the interface between the two remaining parts is a spatial disc. The cut of the unit ball corresponds to a cut of the associated bubble graph, as shown on the right. Only those radial spacelike links are shown which belong to the interface.

the unit ball where these new bubbles appear as 1+1 dimensional bubbles. The bubble graph on the right bottom conveniently visualizes the bubble structure of the unit ball.[5].

Now we take the unit ball from Fig. 7.11 at the bottom and attempt to cut it in half in such a way that the interface between the two remaining parts is a spatial disc. This is most conveniently done by cutting the associated bubble graph in half, as shown in Fig. 7.12. We are interested in cuts such that the time orientation of all spacelike triangles in the interface agrees in the sense that time flows consistently from one half to the other. This condition is necessary because otherwise the bubble move will destroy the local causality. The condition translates into a requirement for the cut of the bubble graph. Namely, the bubble graph has to be cut in such a way that all cut edges consistently point from one half to the other. One can show that under this condition the map between the cuts of the bubble graph and the admissible cuts of the unit ball is in fact one-to-one.

[5]Note that bubble graphs associated to 2+1 dimensional locally causal triangulations are in general not planar graphs. We conveniently use planar bubble graphs for visualization purposes.

Cutting local bubble graphs in two

We have reformulated the problem of finding a spatial disc through a vertex to the problem of finding a cut in the local bubble graph. It is not difficult to write an algorithm which finds one arbitrary cut, but we are interested in an algorithm where all possible cuts are found with a nonzero probability. We also need to be able to efficiently compute the probability that a particular cut is found by the algorithm. In section 7.1 we have discussed the process of proposing trial moves and we have seen that it is advantageous to have an algorithm whose decision graph is as treelike as possible (see also Fig. 7.1). Therefore we would like to avoid the situation that a particular cut is found by several algorithmic paths, because then we would need to execute all these possibilities because the total probability is the sum of the probabilities for each path.

Ideally we would like to have an algorithm where the probability of finding a cut is the same for all the cuts to optimize the acceptance rate of the move. This would require us to enumerate all the cuts, store them in a list and then select a cut with probability $1/N_c$ where N_c is the total number of cuts. In practice such an approach is bound to fail because the number of cuts can become very large and the algorithm would become prohibitively slow.

Bubble graphs are by our definition directed graphs where the direction of each edge indicates the flow of time between the two vertices (bubbles) which are connected by the edge. Bubble graphs can contain cycles[6] as already mentioned in section 5.1. As we will see later, the presence of cycles significantly complicates the algorithm to determine a cut in the local bubble graph. For didactical reasons we therefore first explain a simplified version of the algorithm which assumes that the bubble graph does not contain cycles.

Figure 7.13 visualizes the working of an algorithm to find a cut in a local bubble graph without cycles. First we select the past root node and assign the state 'accepted' to it. All other nodes are put into the 'unassigned' state except for the future root node which is put into the 'rejected' state. Then we determine a set of candidate nodes as follows: for every accepted node we consider those of its future neighbors which are in the 'unassigned' state. For every neighbor we check whether all its past neighbors are in the 'accepted' state. If that is the case, we change its state to 'candidate'. This situation is shown in the first graph in the figure.

To every candidate node we assign the state 'accepted' with a probability p and the state 'rejected' with a probability $1 - p$. We will comment on how to choose a suitable value for p below. Then we determine a new set of candidate nodes and again accept or reject them. The process is iterated until the set of new candidate nodes becomes empty. This situation is shown in the last graph in the figure. At the end we determine all the edges which connect accepted nodes with nodes that have a state different from 'accepted'. This can be done for example using a breadth first traversal algorithm. Every edge represents a set of spacelike triangles and the union of all the spacelike triangles represented by all the edges that were determined at the end forms the spatial disc which is needed as starting point to execute the inverse bubble move.

[6]Here and in the following we always use the term 'cycle' to describe an oriented/directed cycle.

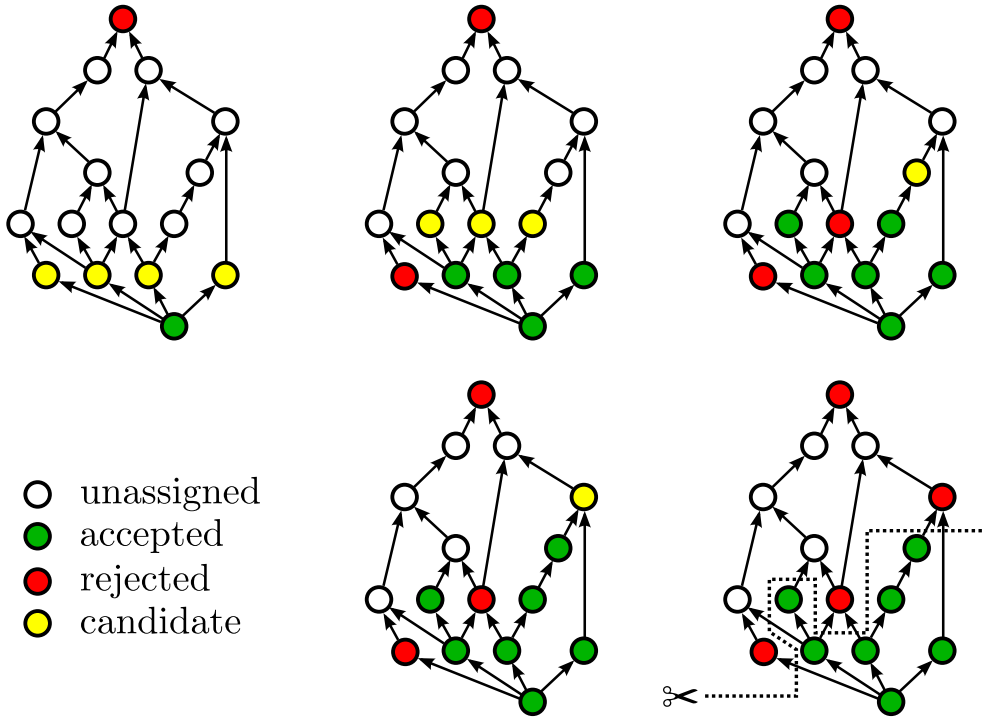


Figure 7.13: A visualization of the algorithm to generate a cut of a local bubble graph without cycles.

The algorithm is designed such that every cut is generated in a unique way and therefore the probability of generating a particular cut is $p^{N_a}(1-p)^{N_r}$ where N_a and N_r are the number of accepted and rejected nodes respectively, excluding both root nodes. In practice we have to deal with an additional problem. Before the algorithm can be executed the local bubble graph must be determined first. This in turn requires to assign a time orientation to the triangulation and there we have a twofold ambiguity because there are two ways to time-orient the triangulation, where both ways are related by globally reversing the flow of time. The consequence is that the choice of a past root node in the cut algorithm is not well defined.

One way to deal with the problem is to internally store the time orientation along with the usual data structures which represent the connectivity of the triangulation. Then the time orientation is well-defined for every configuration which is generated during the Monte Carlo process. The drawback of this approach is that all Monte Carlo moves acquire an additional complexity because the local change of the time orientation needs to be computed.

A second approach which avoids storing the time orientation is to randomly choose one of the two root nodes as starting node for the cut algorithm. We assume that both nodes are chosen with equal probability. Then we run the cut algorithm, but it is clear that the same cut could have been generated starting from the second root node. Therefore we run the algorithm a second time with the alternative starting node. The second pass works slightly differently because the choices are not random anymore but are dictated

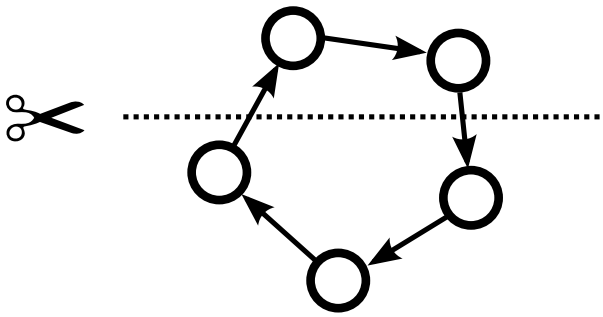


Figure 7.14: A cut through a cycle which demonstrates that cutting cycles always lead to inconsistent time orientations along the cut.

by the outcome of the first pass since the same cut needs to be generated. The total probability of generating the cut is then the average of the probabilities computed in both passes.

The latter approach has another advantage. The probability of a cut being generated depends on the probability p which is used to accept or reject nodes. There is no generic best choice for p because the optimal value depends on the structure of the bubble graph. For small bubble graphs the value of p does not matter too much as long it is not chosen extremely small or large. If the bubble graphs become larger, typical choices such as $p = 1/2$ make cuts that are close to the starting root node appear much more likely compared to cuts that are close to the other root node. By running the algorithm twice, the cuts are found in a more balanced way.

The cut algorithm for bubble graphs with cycles

Bubble graphs generically contain cycles and this is also true for local bubble graphs. If we apply the cut algorithm previously described to a local bubble graph with cycles it will sometimes generate cuts which not only cut the bubble graph in half but which also cut some of the cycles in half. Figure 7.14 illustrates the problem which appears when cycles are cut. The time orientations of the two cut edges are different and thus the spatial disc associated to the complete cut is not valid as a starting point for the inverse bubble move because the move would destroy causality. Therefore we need a cut algorithm which never cuts cycles. In the following we describe the final cut algorithm which we have implemented in our simulation software.

Let us consider a local bubble graph where the new cut algorithm has already progressed such that some nodes have been assigned the states 'accepted' or 'rejected'. We assume that the algorithm started from the past root node. For every node N in the 'unassigned' state we consider the set of all nodes which can be reached by going backwards along the directed edges and which are also in the 'unassigned' state. We denote this set as the *past cone* associated to N. Furthermore we say that a past cone is *irreducible* if it does not contain any past cones that have fewer nodes. It is clear that every irreducible past cone which contains nodes belonging to a cycle in fact contains the complete cycle.

The new cut algorithm determines a list $\mathscr{L}$ of candidate past cones in several steps.

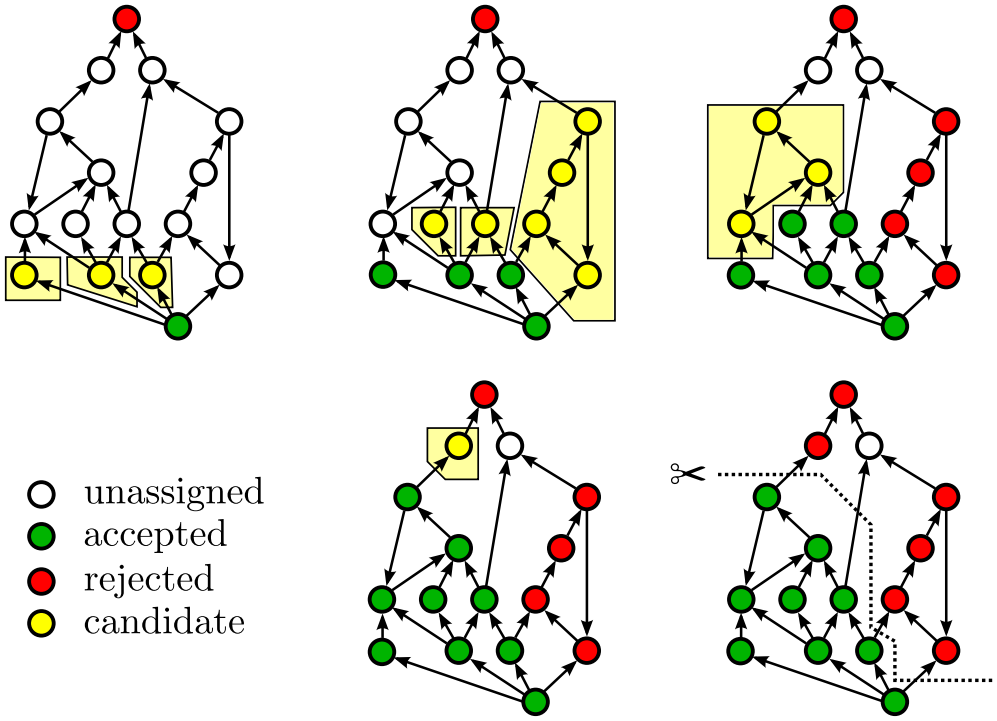


Figure 7.15: A visualization of the algorithm to generate a cut of a local bubble graph without cutting any cycles. The candidate nodes in each highlighted area are accepted or rejected together.

First we set $\mathscr{L} = \{\}$ and then iterate over all unassigned nodes that have at least one past neighbor in the 'accepted' state. For every one of these nodes we check whether any node in its past cone has a past neighbor in the 'rejected' state. If this is not the case we add the past cone to $\mathscr{L}$.

Now we iterate over all elements of $\mathscr{L}$ and remove all past cones which are not irreducible. Some of the remaining irreducible past cones may only differ in their root node but otherwise contain the same nodes. Thus we can see the list as being composed of groups of irreducible past cones where each member in a group contains the same nodes. Now we iterate over $\mathscr{L}$ and remove all but one member of each of these groups. Thus we end up having a list of irreducible and disjoint past cones. For each past cone in $\mathscr{L}$ we accept or reject all of its nodes together with a probability of p resp. $1-p$. Then the process of generating a list of candidate past cones is repeated. If $\mathscr{L}$ stays empty then the algorithm is finished. Figure 7.15 shows a visualization of the final cut algorithm.

To calculate the probability of generating a cut we count how many times we accept or reject all nodes inside a past cone. If these numbers are N_a and N_r then the probability is $p^{N_a}(1-p)^{N_r}$. In our implementation we use $p = 1/2$ and run the algorithm twice, using each root node as a starting node, to deal with the twofold ambiguity of the global time orientation as we have discussed previously. Since we choose the first starting node with probability $1/2$, the total probability of generating the cut is the average of the probabilities computed during each run.

Implementation of the bubble move

We begin with describing the implementation of the bubble move which collapses a timelike link. First we choose a tetrahedron of type T_2 and determine whether it belongs to a ring where all tetrahedra are of type T_2. If this is not the case, the move is rejected. Otherwise we use algorithm 1 on page 1 to test whether the move connects two combinatorial triangulations and reject the move if this is not the case. Then we compute the change of the action using the f-vector difference

$$\Delta f = (-1, -n_{T_2}, -1, -n_{T_2}, -n_{T_2}, 0, -n_{T_2}, 0, 0, 0),$$

where N_{T_2} is the number of tetrahedra in the ring. The rest of the implementation is straightforward except for the computation of the trial probabilities which we discuss next.

The probability of proposing the bubble move is easy to compute. There is only one choice involved, namely, the choice of a tetrahedron of type T_2. This choice determines which ring is being collapsed, but any other tetrahedron in the ring could have been selected resulting in the same move. Therefore the trial probability becomes the number of tetrahedra in the ring divided by the total number of tetrahedra of type T_2.

The real challenge lies in computing the trial probability for the inverse of the bubble move. That probability is determined by the cut algorithm which we have described in the previous section, but now we have the additional complication that we need to know the local surrounding of the collapsed link, which is not directly available before the bubble move has been fully executed. There are two strategies to deal with this problem. One possibility is to execute the bubble move without doing the detailed balance test. Then the probability of the inverse move can be determined by running a modified version of the cut algorithm where the choices are not random but determined by the local surrounding of the link before the move has been executed. Afterwards the detailed balance test can be done. If the outcome is such that the bubble move should not have been executed, the original situation is restored by executing the inverse move.

We are using an alternative method. We have implemented an algorithm which constructs the local bubble graph associated to the link which is being collapsed by the move. Then we have modified the algorithm such that it generates the local bubble graph of the collapsed link. Along the way we also store all the information which is needed by the cut algorithm to generate the correct cut which leads back to the original situation. Then we twice run the modified version of the cut algorithm which is not based on random choices but where all decisions are made such that the correct is found. The computed probability together with additional contributions to be discussed below are finally used to perform the detailed balance test.

Implementation of the inverse bubble move

To execute the inverse link collapse move we first choose a vertex. Then we create the local bubble graph associated to the chosen vertex and run the cut algorithm to find a

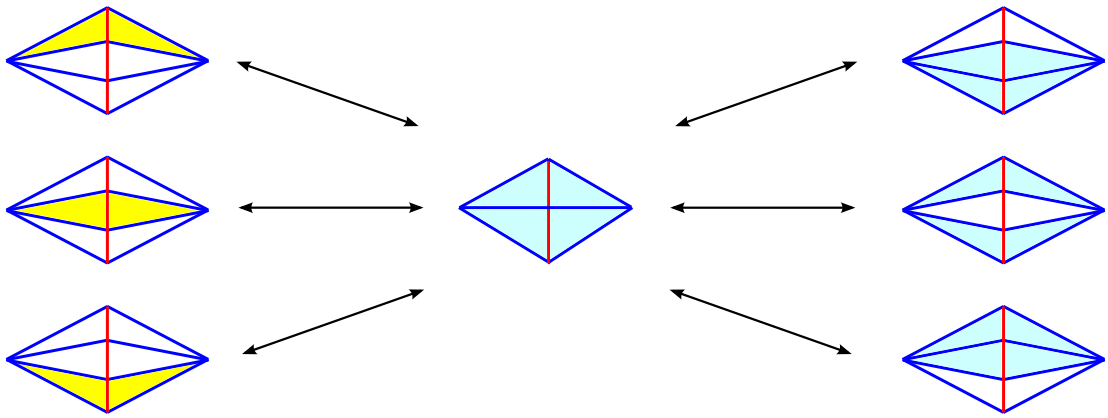


Figure 7.16: A 1+1 dimensional illustration of a situation where the collapse of different links implements the same transition.

cut in the bubble graph. Using the cut we determine the associated spatial disc. Then we compute the change of the action using the f-vector difference

$$\Delta f = (1, n_{T_2}, 1, n_{T_2}, n_{T_2}, 0, n_{T_2}, 0, 0, 0),$$

where n_{T_2} is the number of tetrahedra of type T_2 which are created by the move. This is also equal to the number of triangles in the chosen spatial disc. The trial probability is 1.0 divided by the total number of vertices times the probability to generate the cut using the cut algorithm. The trial probability for the inverse is simply n_{T_2} divided by the number of tetrahedra of type T_2 after the move. Finally we perform the detailed balance test and depending on the outcome complete the move.

Situations with symmetry

Multiple bubbles can be created on top of each other. The left side of Fig. 7.16 shows a situation where three bubbles sit on top of each other. The visualization is for 1+1 dimensions, but can also be seen as a cut through an analogous 2+1 dimensional situation. No matter which of the timelike links we collapse we always end up in the same situation shown in the center. If we then create a bubble in the center situation we have again multiple ways of doing it. The inverse bubble move involves choosing a spatial disc, and the 1+1 dimensional analogue of a spatial disc is a pair of spacelike links. In the center situation we have three pairs of spacelike links which we can choose. For each choice the inverse bubble move leads to the same situation shown on the right.

Normally the computation of the trial probabilities for the illustrated situation would be complicated because we would need to do the computation six times, three times for the collapse move and three times for the inverse move. Fortunately it turns out that by pretending not to know about these symmetric situations we in fact do the right thing. In section 7.1 we have explained that if we can organize all the possible algorithmic paths leading from one situation to the other in groups and if we can find a bijection between

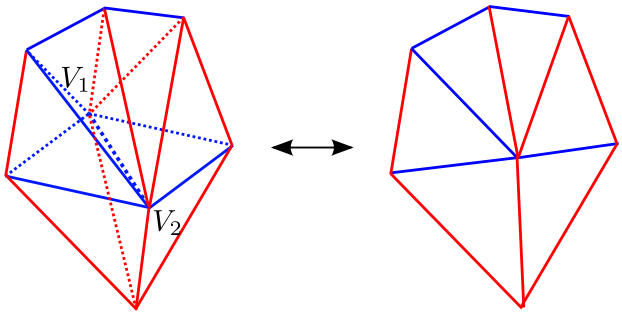


Figure 7.17: An illustration of the link collapse move which collapses a spacelike link. Spacelike links are drawn in blue, timelike links in red.

the groups for the move and its inverse, it is sufficient to implement the detailed balance condition for each pair of groups separately.

It is easy to find such a bijection in the illustrated situation. Each arrow represents an algorithmic path and we put every arrow in its own group, such that all groups contain exactly one algorithmic path. Thus we have three groups for the collapse move and three for the inverse move. We define the bijection by associating the arrows on the same level to each other, thus the top left arrow is associated to the top right arrow and so on. Therefore if we execute the link collapse move, for example, by collapsing the topmost timelike link, we only need to compute the trial probabilities associated with the top left and top right arrows.

7.3.3 The link collapse move for spacelike links

Both the pinching move and the bubble move discussed previously collapse timelike links. Here we present a move which collapses spacelike links. Figure 7.17 shows an example of such a move. Other examples could be constructed by changing the number of tetrahedra around the center link or by changing the types of the links which are part of the link star and which are not identical to the center link. It is clear that the local causality constraint restricts the possible link type patterns.

Let V_1 and V_2 be the two vertices contained by the center link and let $\mathscr{V}$ be the set of vertices which belong to the link star but not to the center link. If for every $V \in \mathscr{V}$ the two links $\overline{VV_1}$ and $\overline{VV_2}$ have the same link type then we say that the link collapse move is 'symmetric'. These symmetric moves form a subset in the class of all possible spacelike link collapse moves. To keep the complexity of the implementation at a manageable level we have restricted ourselves to the symmetric version of this move.

As we have seen in section 5.3 the set of all tetrahedra of types T_2 and T_3 organizes itself in a collection of rings, assuming that our model is of Type 1. The primary function of the spacelike link collapse move is to change the size of these rings. Consider for example the simple bubble illustrated in Fig. 7.10 on the left. It is a ring containing five tetrahedra of type T_2. By collapsing one of the spacelike links which are not connected to the center link we can easily decrease the size of the ring to four, assuming that the local

environment of that spacelike link permits the move. Conversely, it is easy to increase the size of a ring by executing the inverse of the spacelike link collapse move.

The spacelike link collapse move collapses the star of a link to a disc. The inverse move therefore involves choosing a vertex and a disc with the vertex at its center. It is easy to see that none of the triangles in the disc must be spacelike because otherwise the inverse move would produce tetrahedra of type T_1 which have Euclidean signature. Therefore all the triangles in the disc have to be Lorentzian. This is in contrast to the situation for the bubble move where the disc was constrained to contain only spacelike triangles. In section 7.3.2 we have presented an algorithm to choose a spatial disc through a vertex. That algorithm exploits the fact that all spacelike triangles belong to interfaces between bubbles. It is designed such that there is a good chance that it can be scaled up to higher dimensions without too much effort.

Unfortunately, the situation is much more difficult for the spacelike link collapse move. Here we cannot take advantage of structures such as the bubble graph. In fact, we have not been able to find an efficient algorithm which is likely to scale up to higher dimensions and which implements the spacelike link collapse move with sufficient generality. In the following we will discuss an implementation which is efficient and appears to be general enough, but is tied to 2+1 dimensions.

Conditions on the disc imposed by causality

Causality puts constraints on the neighborhood of every spacelike link. In the left situation of Fig. 7.17 we see three types of spacelike links in addition to some timelike ones. There is the center link, the spacelike links which are connected to the center link, and those which are not connected to the center link. In the case of the center link the neighborhood is additionally constrained by the fact that we only consider symmetric link collapse moves. The surface of the link star can be seen as two discs glued to each other along their boundaries, with each disc having V_1 or V_2 as its center. Both discs have identical structure and are equal to the disc which results from the execution of the spacelike link collapse move. The requirement that the link star is causal then translates into the condition that one encounters precisely four light cone crossings when circling around the disc. We denote this condition by C_A.

Let us now consider the spacelike links which are not connected to the center link. The star of every such link is reduced by one tetrahedron during the link collapse. It is easy to see that the number of light cone crossings around such a link stays invariant, thus we do not get any additional constraints. The situation is different for the remaining spacelike links which appear in pairs because we are considering symmetric link collapse moves. The neighborhood of any such spacelike link consists of two parts, one which is contained inside the star of the center link and another part which is located outside of the star. If we circle around the spacelike link we encounter precisely two light cone crossings in each part as a consequence of the local causality condition.

The latter condition translates into an additional condition for our disc. If we circle around any of the radial spacelike links, we require that there are two light cone crossings

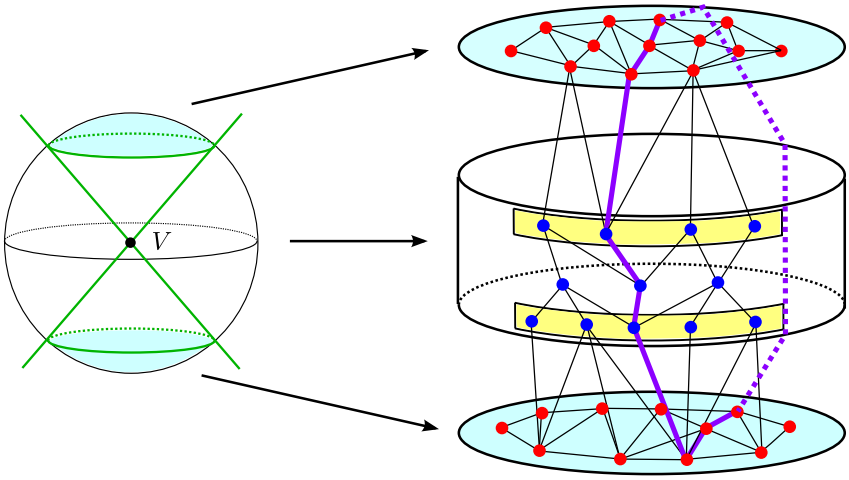


Figure 7.18: A visualization of the reduced surface graph of a unit ball. Blue dots are end points of radial spacelike links, red dots are end points of radial timelike links. An example loop on the graph is visualized in purple.

on each side of the disc. Let us call this condition C_B. In the following we will describe an algorithm to find a disc which satisfies condition C_A while it sometimes violates the condition C_B. Therefore after the disc has been chosen the implementation needs to explicitly check whether the condition C_B is violated and reject the move if the answer is yes.

Choosing a Lorentzian disc

In the following we discuss an algorithm to choose a Lorentzian disc suitable as an initial condition for the inverse link collapse move. Ideally the algorithm would be able to find all the discs, but in practice this goal is not achieved completely. Our algorithm does find 'almost' all the discs which is sufficient for practical purposes. Every disc is found in a unique way such that it is not necessary to execute the algorithm multiple times to compute the total probability of finding the disc.

Finding a non-intersecting disc through a given vertex is equivalent to finding a non-intersecting loop on the surface of the unit ball of the vertex. The general idea of the algorithm is to generate such a non-intersecting loop. The naive solution to this problem would be to select a starting vertex and to perform a self-avoiding random walk on the surface of the unit ball. This approach has several problems. In most of the cases the algorithm would not reach the starting vertex to close the loop, thus leading to a large number of rejected moves. Furthermore, most of the loops would not satisfy the constraints which we have formulated previously. Another problem is that there are many ways of finding the same loop by choosing another vertex on the loop as alternative starting point.

We need a more sophisticated approach which exploits the light cone structure at the center vertex. First we consider the surface of the unit ball of the center vertex. The light

cone structure divides the surface into three regions as shown in Fig. 7.18 on the left side. The future and the past region have the topology of a disc while the region in between, which we call the 'present region', has the topology of an cylinder. Now we consider the triangulation of the surface and remove all spacelike links which belong to spacelike triangles connected to the center of the unit ball. All the removed links necessarily belong to the present region. We call the resulting graph the *reduced surface graph* associated to the center vertex. It is visualized on the right side of the figure.

Our algorithm will attempt to generate a loop in the reduced surface graph. The removal of links is a simple way of implementing the constraint that all triangles belonging to the disc have to be Lorentzian. Looking at the right side of the figure we see that we can also divide the reduced surface graph into three regions. All the vertices in the future and past disc are end points of timelike radial links while the vertices in the middle region belong to spacelike radial links. Those surface links which connect two distinct regions are those which contain an end point of a radial lightlike line. Thus whenever we go from the middle region to the future or past region and vice versa we encounter a light cone crossing.

Let $\mathscr{S}_1$ and $\mathscr{S}_2$ be the set of vertices in the middle region which are connected to vertices in the future and past region respectively. These sets are highlighted in yellow in the figure. In typical situations we have $\mathscr{S}_1 \cap \mathscr{S}_2 \neq \{\}$, and if the local triangulation is foliated we have $\mathscr{S}_1 \equiv \mathscr{S}_2$. We are interested in loops which contain two links that connect the middle region with the future region and which contain two other links that connect the middle region with the past region. In the generic case the loop will therefore contain two distinct vertices $V_1, V_2 \in \mathscr{S}_1$ and two distinct vertices $V_3, V_4 \in \mathscr{S}_2$. In special cases some of the vertices of the first group may be identical to vertices in the other group.

These distinguished vertices divide the loop into four (or less) segments. One segment always traverses the future region, another one traverses the past region, the remaining two traverse the middle region. The length of the latter two can be zero. The basic idea of our disc selection algorithm is to first choose the distinguished vertices and then create the loop segments individually.

Ordering relations

Before we work out the details of the disc selection algorithm, we need to do some preparations. One challenge for the algorithm is to generate the loop in a unique way. To achieve this goal we use ordering relations based on graph invariants. We can assign to every vertex in the reduced surface graph a tuple of numbers, where each number is computed in a deterministic way such that the choice only depends on the reduced surface graph and not on internal labelings in the implementation. These tuples admit an ordering relation which we can use to order the vertices in the reduced surface graph. We use the obvious notation $V_1 < V_2$, $V_1 = V_2$, $V_1 > V_2$ to reflect the ordering. It is easy to extend the ordering relation to sub-graphs by combining the numbers for all the vertices in the sub-graph.

Here we discuss a few examples of these numbers. For every vertex we can calcu-

late the coordination number which is the number of links attached to it. We can also calculate the sum of the coordination numbers of its neighbors. This idea can be generalized further by determining the sum of the squares of the coordination numbers of the neighbors. Or one can modify all of these prescriptions by only counting the number of spacelike links.

In the case that $V_1 = V_2$ for two vertices the implementation may not be able to generate the loop in a unique way and has to reject the move. The likelihood of such a rejection depends on the number of independent graph invariants that are used to define the ordering relation. It is convenient to monitor the number of these rejections during the simulation. If the number is too high, additional graph invariants can be added to reduce the number of rejections.

Selecting the distinguished vertices

Our first step is to choose either the future or the past region of the reduced surface graph by choosing the region which is smaller with respect to the ordering relation. If neither is smaller we reject the move. We use the notation $\mathscr{P}_1$ for the selected polar region and $\mathscr{P}_2$ for the other polar region. We call $\mathscr{S}_1$ the set of vertices in the present region which is connected to $\mathscr{P}_1$ and we use $\mathscr{S}_2$ for the analogous set of vertices connected to $\mathscr{P}_2$.

Now we randomly select a vertex $V_1 \in \mathscr{S}_1$ and define $\mathscr{L} = \{\}$. Then we iterate over all elements $V \in \mathscr{S}_1$ and check whether we have $V > V_1$ and add V to $\mathscr{L}$ if the condition is satisfied. Now we randomly select one member of $\mathscr{L}$ and denote it by V_2. Then we set $\mathscr{L} = \{\}$ and perform a breadth-first traversal algorithm starting at V_1, such that only vertices are visited which belong to the present region. Whenever we encounter a vertex $V \in \mathscr{S}_2$ we add it to $\mathscr{L}$. Then we randomly select one member of $\mathscr{L}$ and denote it by V_3. Now we again set $\mathscr{L} = \{\}$ and repeat the breadth-first traversal algorithm, this time with V_2 as starting point. Every vertex $V \in \mathscr{S}_2$, $V \neq V_3$ is added to $\mathscr{L}$. We randomly select an element in $\mathscr{L}$ and denote it by V_4. If at any stage of this process the set $\mathscr{L} = \{\}$ stays empty, we reject the move.

The vertices V_i, $i \in \{1,4\}$ are the end points of the loop segments that the algorithm will generate later. The condition $V_2 > V_1$ ensures that the same loop could not have been generated if we had chosen V_2 as the first vertex in the process described above. If $V_3 \notin \mathscr{S}_1$, it could not have been chosen as initial vertex, because we chose $\mathscr{P}_1$ as first polar region, and if $V_3 \in \mathscr{S}_1$ we have $V_3 \equiv V_1$. The same argument applies to V_4. Therefore we conclude that there is precisely one algorithmic path leading to the selection of the four vertices V_i, $i \in \{1,4\}$.

The four vertices are the end points of the four loop segments $(V_1 \cdots V_2)$, $(V_1 \cdots V_3)$, $(V_2 \cdots V_4)$ and $(V_3 \cdots V_4)$. The next step in our algorithm is to determine all the intermediate vertices in each loop segment. We will start from one end point and move towards the other which means that we first have to choose where we start. We choose the end point V_i with the smaller index i. The order in which we process the loop segments does not matter. In the following we describe the algorithm to generate a path connecting two vertices. The same algorithm can be used to construct all loop segments.

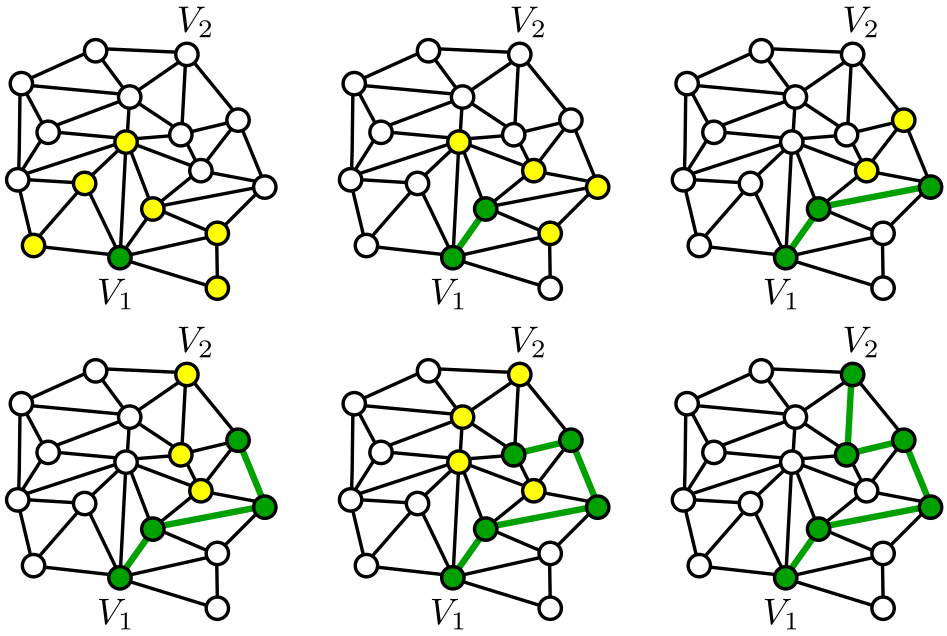


Figure 7.19: A visualization of the algorithm to generate a path between two vertices in a graph. White nodes are unassigned nodes, green nodes are accepted nodes and yellow nodes are candidate nodes.

Constructing paths between two vertices

Figure 7.19 shows a visualization of the algorithm to generate a path between two vertices. The starting vertex is V_1 and the vertex to be reached is V_2. We start by assigning all vertices the state 'unassigned' except for the starting vertex which is assigned the state 'accepted'. Now we generate a list of candidate vertices $\mathscr{L}$ as follows. First we set $\mathscr{L} = \{\}$ and generate a list of all neighbors of the last accepted node, which in the beginning is V_1. For every neighbor N we perform a breadth-first traversal algorithm starting at that neighbor with the constraint that accepted nodes are not visited. If V_2 is visited during this process we add N to $\mathscr{L}$ and also store the shortest distance between N and V_2. At the end of the process $\mathscr{L}$ contains a list of all neighbors of the last accepted node which are able to reach V_2.

Now we randomly select one element of $\mathscr{L}$ and assign the state 'accepted' to it. We have decided to use a non-uniform measure to perform this choice. We use the shortest distance between the candidate node and V_2 and assign a greater weight if the distance is small. This prescription favors short paths between V_1 and V_2. Experience with the simulations indicates that collapse moves often produce small discs and thus our prescription is an effort to optimize the acceptance rate of the move. Now we repeat the generation of a list of candidate nodes and the whole process is iterated until V_2 is accepted.

We run the algorithm four times to construct all four loop segments. All the accepted nodes collected in these four runs form the loop which in turn defines the disc that is used as initial condition by the inverse link collapse move. To calculate the probability of generating the loop we initialize a variable $p = 1.0$ before we make our first random

choices. Then for each random choice which is made with probability P we set $p = p \cdot P$. At the end of the whole process we simply read out the value of p to get the desired probability.

Implementation of the spatial link collapse move

We begin by describing the implementation of the move which collapses a spacelike link. First we choose a spacelike link and determine whether the move is a symmetric link collapse move. If the answer is no, we reject the move. Then we use algorithm 1 on page 1 to test whether the move connects two combinatorial triangulations and reject the move if this is not the case. A careful study of all possible neighborhoods of the center link for symmetric link collapse moves reveals that the f-vector difference has the form

$$\Delta f = -(1, 3 + n_{T_2}, 2 + n_{T_9}, 2 + n_{T_2}, n_{T_2} + n_{T_3}, 6 - n_{T_3} + 2n_{T_9}, n_{T_2}, n_{T_3}, 4 - n_{T_3}, n_{T_9}),$$

where n_{T_2}, n_{T_3} and n_{T_9} are the numbers of tetrahedra of the denoted types that are created by the move. Note the overall minus sign in front. n_{T_2} and n_{T_9} are not bounded from above while we have $1 \leq n_{T_3} \leq 4$. The implementation now computes the action and the trial probabilities, then performs the detailed balance test and executes the move if it is accepted.

The trial probability for proposing the collapse move is simply 1.0 divided by the number of links. The computation of the trial probability for the inverse move is more difficult. First we need to construct the reduced surface graph associated to the vertex which results from collapsing the link. This can be done by simply constructing the reduced surface graph associated to the center link. Then we need to run a modified version of the disc selection algorithm where all decisions are made such that the correct disc is selected which leads back to the original situation. It is crucial that the modified algorithm follows the same algorithmic path as the original algorithm. The computed probability together with additional contributions, which we discuss below when discussing the inverse move, are used to perform the detailed balance test.

Implementation of the inverse spatial link collapse move

To execute the inverse link collapse move we first choose a vertex. Then we create the reduced surface graph associated to the chosen vertex and run the disc selection algorithm to generate a disc. Afterwards we check whether the disc satisfies the condition C_B which we have introduced on page 111. If this is not the case we reject the move. Otherwise we proceed by calculating the action using the f-vector difference

$$\Delta f = (1, 3 + n_{T_2}, 2 + n_{T_9}, 2 + n_{T_2}, n_{T_2} + n_{T_3}, 6 - n_{T_3} + 2n_{T_9}, n_{T_2}, n_{T_3}, 4 - n_{T_3}, n_{T_9})$$

where n_{T_2}, n_{T_3} and n_{T_9} are the numbers of tetrahedra of the denoted types that are created by the move. The trial probability is 1.0 divided by the number of vertices times the probability to generate the disc. The trial probability for the inverse is 1.0 divided by the

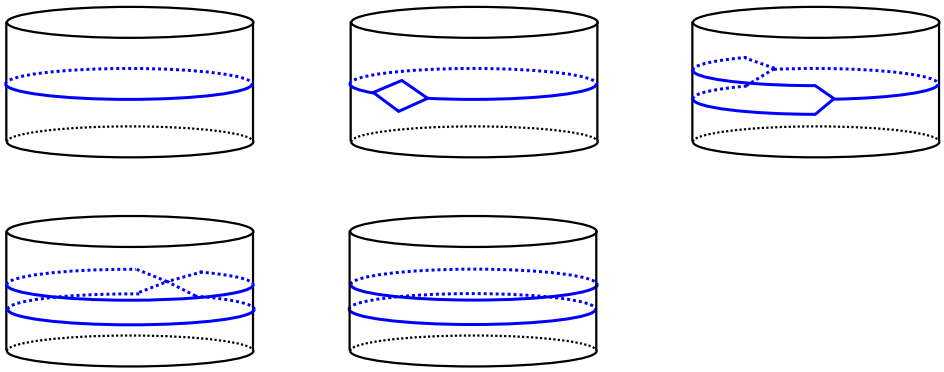


Figure 7.20: A 1+1 dimensional illustration of the process to generate a thick slice. Timelike links are omitted.

number of links after the move. Finally we perform the detailed balance test and execute the move if it is accepted.

We close with a remark concerning symmetric situations. In the discussion of the bubble move we have shown that there exist situations which exhibit a particular type of symmetry in which the naive computation of the trial probabilities becomes complicated. We have argued that by simply ignoring the issue we do the right thing and detailed balance is satisfied. The same remark applies also for the spatial link collapse move. The implementation we have described is designed such that the simplified calculations are applicable.

7.4 The polar move

In simulations of CDT the number of time steps is fixed. Since foliated triangulations form a subset of all causal triangulations the question arises whether it is possible to alter the number of time steps of a foliated triangulation using a Monte Carlo process which at an intermediate step goes through non-foliated states. The answer is yes, as explained in the following.

Figure 7.20 shows a visualization of the process in 1+1 dimensions. Space is compactified with the topology of a circle. Let us focus on one spatial section as shown on the left top of the figure. First we create a bubble and afterwards use other Monte Carlo moves to grow the bubble and shrink the region outside the bubble. The effect is that the two ends of the bubble approach each other until we end up with a pinching, which is finally removed with the inverse pinching move. The process works in an analogous way in 2+1 dimensions.

In the vicinity of sources or sinks of time there exists a second process to generate a thick slice which involves a new type of Monte Carlo move. Figure 7.21 shows the process in 1+1 dimensions. The first graph shows a bubble containing a source of time. Then we execute the new move which subdivides a triangle into three triangles and changes two link types from timelike to spacelike. We call this move the *polar move*.

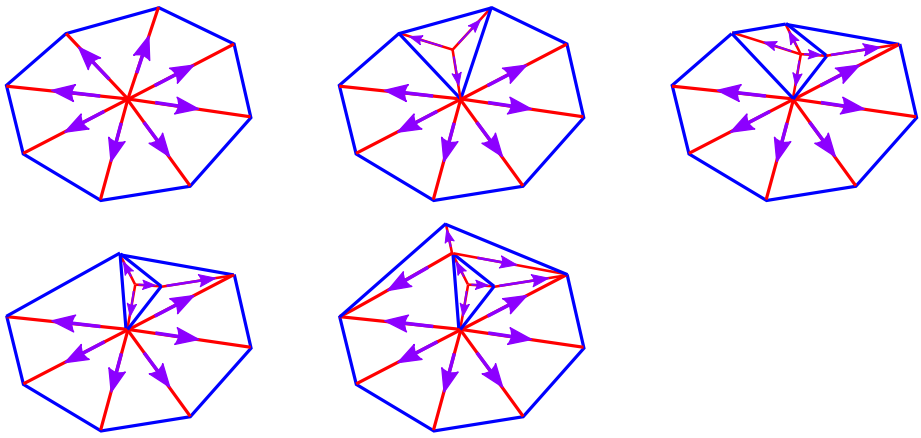


Figure 7.21: A 1+1 dimensional visualization of a second process to generate a thick slice using the polar move.

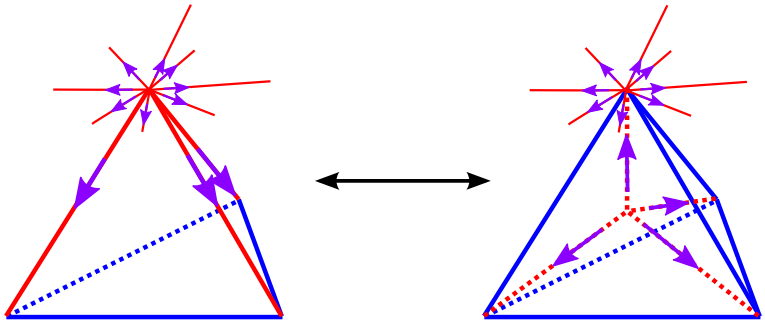


Figure 7.22: The polar move in 2+1 dimensions.

The newly created vertex takes over the role of a source of time. Afterwards we execute a subdivision move, a spatial link collapse move and an inverse pinching move. The result shown in the last graph contains two bubbles, one containing the new source of time and the other having the shape of a thick slice.

The polar move in 2+1 dimensions is illustrated in Fig. 7.22. It subdivides a tetrahedron with a source or sink at one of its vertices and changes the type of all timelike links of the original tetrahedron to spacelike. The newly created vertex becomes the new source or sink of time. The implementation of the polar move first chooses a source or sink of time and afterwards chooses one of the tetrahedra in the star of the chosen vertex. The f-vector difference for the polar move is

$$\Delta f = (1,3,1,3,n_{T_3}+3,-n_{T_3},3,n_{T_3},-n_{T_3}),$$

with n_{T_3} being the number of tetrahedra of type T_3 that are created by the move. The rest of the implementation and the computation of the trial probabilities is straightforward.

The inverse move first selects a source or sink of time and then tests whether the vertex has coordination number four. If this is not the case the move is rejected. Otherwise the four neighboring vertices are determined and one of them is chosen to be the new source or sink of time. Then the implementation tests whether the unit ball of the new

source/sink of time will exclusively contain tetrahedra of type T_5 after the move. If this is not the case the move is rejected. There are no further difficulties concerning the implementation and the computation of the trial probabilities. The f-vector difference is obviously the negative of the one for the other direction.

7.5 Implementation of the volume distribution

In section 6.5 we have introduced spatial slices in nonfoliated triangulations of Type 1 which are essentially nonintersecting sets of spatial triangles that divide the whole triangulation into two disconnected pieces, assuming that time is not compactified. We have also seen that the number of such spatial slices can become very large such that it becomes unfeasible to enumerate all of them. Here we present an algorithm which samples a subset of all spatial slices using statistical methods. This set can then be used to create volume distributions as explained in section 6.5.

In the following we assume that time is not compactified and that we work with the spacetime topology of the 3-sphere. To find one single spatial slice we need to find a cut along spacelike triangles which splits the triangulation in two parts. We can take advantage of the fact that we have already solved a similar problem in order to implement the bubble move, as explained in section 7.3.2. There we have discussed an algorithm which generates a cut of the unit ball of a vertex. The cut algorithm is designed such that it is capable of finding all possible cuts with a nonzero probability. This makes the algorithm useful to generate spatial slices in the whole triangulation.

The main challenge here is to generate a balanced set of spatial slices which means that we would like to sample spatial slices everywhere on the time axis in order to build a useful volume distribution. Therefore we need to study the behavior of the cut algorithm in detail. We briefly recapitulate how the algorithm works and adapt the discussion to the case of finding a cut in the whole triangulation.

The basic idea is to find a cut in the bubble graph of the triangulation. First the bubble containing the source of time is chosen and marked as 'accepted'. Then a set of neighboring nodes is determined according to the rules given in section 7.3.2 and all these nodes are assigned the state 'candidate'. Afterwards a series of random choices is made which assign either the state 'accepted' or 'rejected' to the candidate nodes. Then the process is repeated with selecting a new set of candidate notes. The algorithm terminates as soon as there are no new candidate nodes. See Fig. 7.15 for an illustration of the mechanism.

In order to select a particular cut, the algorithm has to make a number of random choices. Thus we can visualize the working of the algorithm using a rooted graph in a similar way as we did when discussing the mechanism of proposing Monte Carlo moves in section 7.1. Figure 7.1 shows two such decision graphs, a tree-like graph on the left and a non-tree-like one on the right. The root node represents the initial state of the algorithm. Every random choice amounts to stepping from the current node to a neighboring node.

In the following we discuss the particular structure of the decision graph for the cut algorithm. Let us first assume that the cut algorithm is in its initial state which is rep-

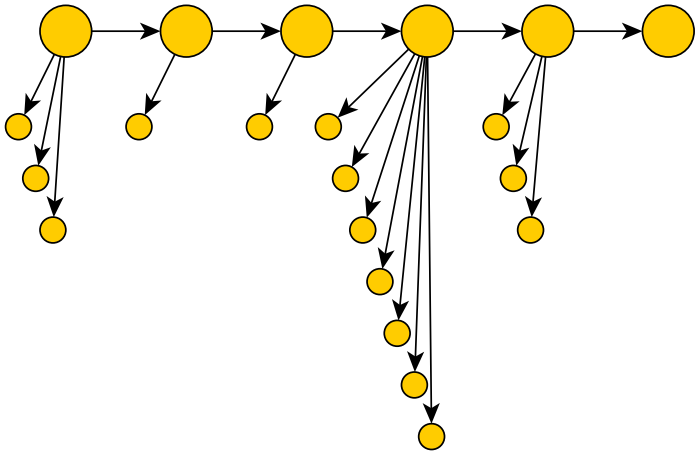


Figure 7.23: A part of the decision graph of the cut algorithm to find spatial slices in a nonfoliated triangulation.

resented in Fig. 7.23 by the left top node. The cut algorithm now determines a set of candidate nodes in the bubble graph and then performs n random choices. Note that n need not be equal to the number of candidate nodes, as explained in section 7.3.2. Since every choice is binary there are in total 2^n new algorithmic states which can be reached from the initial state. Each of these possibilities is represented in the figure by an arrow which connects the left top node to one of its neighbors.

Two of these possibilities are singled out, namely, the ones where all candidate nodes are accepted or rejected. In the latter case the algorithm terminates. The former case is represented in the figure by the arrow which connects the initial node with its horizontal neighbor. Let us assume we make this choice and move to the right. Now we are in a very similar situation as before, except that the number of random choices may be different. If we always decide to accept all candidate nodes, we walk on the horizontal path until we hit the right top node which corresponds to the spatial slice which is closest to the sink of time. The process automatically terminates there.

Let us denote with n_l the number of large nodes in the figure. Now we consider the case that we make $0 \leq m < n_l$ steps to the right. If $m < n_l - 1$, we reject all the candidate nodes and thus terminate the process. Using this prescription we find one of n_l different spatial slices where $m = 0$ represents the spatial slice closest to the source of time and $m = n_l - 1$ the one closest to the sink of time. The set of all spatial slices which can be found in such a way is spread more or less evenly over the t-axis.

It is now clear that all the spatial slices in the triangulation group themselves into n_l equivalence classes where all members of a class are reached by making m steps to the right and afterwards performing random choices which are unconstrained except for the very first one, which is not allowed to lead to the right neighboring node. The set of equivalence classes can be intuitively understood as a partitioning of the whole triangulation, where the parts are time-ordered using m as the ordering parameter.

Let us assume we make m steps to the right starting from the initial node and then perform one random step which does not lead further to the right. We thus end up at one of the small nodes in Fig. 7.23. Assuming that the node is not a final node we consider the sub-graph which has the new node as root node and observe that it has precisely the same structure as the graph shown in the figure. We can therefore simply repeat the same steps as before. By iterating this procedure we finally end up at a final node which is associated to a particular cut of the bubble graph and thus to a particular spatial slice in the whole triangulation.

We have now seen that the whole problem of finding a spatial slice can be reduced to the problem of choosing a path in the subgraph shown in Fig. 7.23. Ideally we would like to make the choices in such a way that all spatial slices finally appear with the same probability. In practice this is not possible since it would require us to enumerate all spatial slices. In the following we explain the details of our algorithm to select a path in the subgraph shown in the figure.

First we perform a deterministic horizontal walk starting on the left top node in the figure and ending at the right top node. Before each step we record the number of binary choices which have to be made in that algorithmic state. We denote these numbers by n_m, $0 \leq m < n_l$, where m is the number of steps made and n_l is the number of large nodes. We assign m as a label to the current node and then continue walking to the right.

Let $p_m = n_m / \sum_{i=0}^{n_l-1} n_i$ and note that $p_{n_l-1} = 0$ because the right top node is a final node. Now we select the large node m with probability p_m and move to that node. Afterwards we perform the n_m binary choices to select one of the outgoing arrows at the chosen node and move to the corresponding neighboring node, which can be both a small or a large node. If the new node is either a final node or a large node, we terminate the algorithm, otherwise we repeat the procedure with the subgraph which has the new node as root node. As soon as we reach a final node the spatial slice is determined and we can use a breadth-first traversal algorithm in the bubble graph to determine all edges which connect the one half of the triangulation with the other half. Every edge is associated to a spatial disc and the union of all these spatial discs forms the chosen spatial slice. The complete algorithm is then finally repeated to generate the desired number of spatial slices.

8 Conclusions

About a hundred years after the revolutionary changes brought by the discovery of quantum physics and general relativity it is still largely unknown how these two pillars of theoretical physics can be combined in a single theory. The number of proposals to tackle this problem has grown steadily over time, and each of these theories have their own virtues and drawbacks. The lack of experiments makes it difficult to even judge a proposal for a theory of quantum gravity, therefore an important task is to search for possibilities how to discriminate between different approaches using experiments. Since the Planck scale, the natural energy scale in quantum gravity, is many orders of magnitude above the energy scales currently accessible by experiments, direct experimental tests of quantum gravity effects are out of reach. Nevertheless it may be possible to test aspects of various proposals of quantum gravity by more indirect methods such as astrophysical observations. This line of research is pursued in the field of 'quantum gravity phenomenology' [95, 96].

In this thesis we have studied one particular approach to quantum gravity called *Causal Dynamical Triangulations* (CDT), which implements the path integral of gravity as a sum over Lorentzian triangulations. It is a fully nonperturbative approach where no background geometry is assumed in the first place. Numerical simulations of CDT in 3+1 dimensions have shown that a four-dimensional background geometry emerges as the ground state of the path integral [37]. It is not difficult to see that this result is truly nonperturbative in nature. The Euclidean Einstein-Hilbert action in the continuum is unbounded from below, featuring a kinetic term with the wrong sign. In contrast, the effective action measured in phase C of the phase diagram of CDT has a kinetic term with the correct sign [37]. This sign flip can be understood as a consequence of an interplay between the bare action and the path integral measure [80].

The fact that CDT relies on a discretization procedure does not imply that there exist fundamental discrete physical objects. The lattice is merely viewed as a quantum field theoretic regulator analogous to dimensional regularization or Pauli Villars regularization, and the lattice spacing assumes the role of an ultraviolet cutoff. To define a continuum theory of quantum gravity the lattice spacing should be sent to zero. More precisely, the goal is to find a suitable fine-tuning of the bare coupling constants such that the lat-

tice spacing goes to zero while the physical length scales are kept invariant. Experience from physical systems with static lattices indicate that the presence of a second-order phase transition is necessary in order to define such a continuum limit. The lack of such a second-order transition in the original formulation of Euclidean dynamical triangulations has been one of the reasons why it has been largely abandoned because there was no prospect of being able to define a continuum theory of quantum gravity.

In chapter 3 we analyzed two lines of phase transitions in 3+1 dimensional CDT. We used three different methods to assess the order of both phase transitions. The first method involved the scaling behavior of histograms of suitable observables. If the couplings were finetuned sufficiently one could observe a flipping between two states, leading to a characteristic double-peak signature of the associated histogram. The second method involved the study of the shift exponent, which describes how the location of the phase transition changes as function of the system size. Finally, we analyzed the dependence of Binder cumulant minima on the system size.

In the case of the A-C transition we observed a double peak histogram which became more pronounced as the system size is increased. This is a clear sign of a first-order transition, which was also confirmed by the results from the other two methods. The histogram analysis applied to the B-C transition did not yield conclusive results. On the other hand the shift exponent was determined to be $\tilde{\nu} = 2.51(3)$, which strongly violates the prediction $\tilde{\nu} = 1$ for a first-order transition. In addition, the analysis of the Binder cumulants supports the second-order nature of the B-C transition. Theses results demonstrate that CDT not only has better infrared properties compared to the original formulation of Euclidean dynamical triangulations, but it also features a second-order transition which may possibly serve to define a continuum limit.

The presence of a preferred foliation is a common ingredient of CDT and *Hořava-Lifshitz gravity* (HLG), which constitutes a new class of theories of gravity designed to be both unitary and renormalizable. In HLG, this is achieved by introducing anisotropic scaling between space and time as well as adding higher-order spatial derivatives to the action. Lorentz invariance is thus explicitly broken and needs to be restored at low energies in order to satisfy the stringent observational constraints on Lorentz symmetry violations. CDT and HLG both share the presence of a preferred foliation, and in both theories the spectral dimension has been shown to be $d_s = 4$ at low energies and $d_s \approx 2$ at high energies. These shared features motivated us to look for more profound connections between CDT and HLG.

In chapter 4 we showed that the phase diagram of CDT in 3+1 dimensions is remarkably similar to the generic Lifshitz phase diagram underlying HLG. By associating the "average geometry" in CDT with the Lifshitz field ϕ we could establish a meaningful one-to-one correspondence between the three phases in each model. It turned out that the order of the phase transitions also matches those found in typical Lifshitz models. Since, according to our analysis in chapter 3, the B-C phase transition in CDT is second order, it may be possible to define a continuum limit by approaching the B-C line. The similarities between CDT and HLG point towards the possibility that a UV fixed point in CDT may be anisotropic in nature, exhibiting a different scaling of space and time. On

the other hand, since the B-C line can be approached anywhere, it may even be possible that there exist both isotropic and anisotropic UV completions. If that was indeed the case, CDT could serve as a unifying nonperturbative framework for anisotropic as well as isotropic theories of quantum gravity.

The central new ingredient in CDT, compared to Euclidean dynamical triangulations, is the presence of a distinguished foliation which ensures that the triangulations are causal. In chapter 5 we introduced a model of *locally causal dynamical triangulations*, where each triangulation is built from Lorentzian building blocks and where the gluing rules are relaxed compared to CDT to allow for triangulations that do not have a distinguished foliation. More precisely, we require that at every point in the triangulation there exists a well-defined light cone and that the locally well-defined flow of time can be consistently extended to the full triangulation.

In chapter 5 we discussed the theoretical aspects of locally causal DT in 2+1 dimensions, with the Regge action and without any nonstandard modification of the measure. The kinematical structure of the new model was shown to be richer compared to CDT, with new objects such as bubbles and pinchings of spatial sections. We showed that there exists a Wick rotation under the condition $1/2 < \alpha < 3$, where α is the asymmetry parameter characterizing the anisotropy between space and time in the regularized sector of the theory. The existence of a Wick rotation is a necessary condition to analyze the model using numerical simulations.

Chapter 6 contains an extensive discussion of the results extracted from Monte Carlo simulations. We have found two phases of geometry, one of them containing extended geometries. These geometries have been shown to be weakly foliated, while almost perfectly foliated states emerge close to the phase boundary $\alpha = 1/2$. A finite-size scaling analysis of the volume distributions strongly suggests that the extended geometries are genuinely three-dimensional. In addition, we have compared the volume distributions to the volume profile of a 3-sphere (the Wick-rotated version of 2+1 dimensional de Sitter spacetime) and have found excellent agreement. The effective large-scale dynamics of the scale factor in the extended phase therefore appears to be similar to the dynamics of CDT in 2+1 dimensions. Therefore it is natural to conjecture that a 3+1 dimensional model of locally causal DT may be compatible to cosmological observations just like CDT in 3+1 dimensions.

While the presence of an extended phase is shared by CDT and the locally causal DT model, we have found significant differences in the overall phase structure. In CDT in 2+1 dimensions, a single phase transition separates the extended phase from a phase featuring uncorrelated spatial slices which contain polymeric structures. In contrast, we find that in the locally causal DT model the polymeric phase is replaced by a long timelike tube with minimal or almost minimal spatial extent everywhere on the time axis. One can therefore expect that a 3+1 dimensional model of locally causal DT may have a different phase structure compared to CDT. If the locally causal model in 3+1 dimensions contains a second-order transition, it could possibly have different properties compared to the second-order transition in CDT which we have extensively discussed in chapter 3.

The implementation of the simulation software has been highly challenging and it

is therefore important to provide extensive material which allows other researchers to independently develop the simulation code in order to reproduce our results. Chapter 7 discusses all the Monte Carlo moves which we have implemented. Among them are generalizations of these moves already used in CDT, plus new moves which implement the collapse of a link. These latter moves are highly nontrivial to implement and we have devoted much space to explaining the key algorithms.

During our research we have been confronted with various problems, some of which remain and will need to be addressed in the future. Difficulties with thermalization forced us to reduce the ensemble by forbidding certain types of global kinematical structures. Even after the reduction, a part of the phase diagram could not be explored. These problems may possibly be overcome by designing new Monte Carlo moves. Another task should be to assess the systematic errors induced by the choice of algorithms when measuring observables. For example, the measurement of volume distributions involves a sequence of several nontrivial algorithms and the dependence of the final results on these algorithmic choices is currently not understood.

There are many ways how this model can be explored, for example, by measuring the various types of dimensions, by analyzing the spatial structures, by exploring the phase transitions, possibly even going beyond the Regge action to analyze the larger space of actions linear in the subsimplex variables. It would also be interesting to investigate the 1+1 dimensional model, possibly with analytical methods and complemented by numerical results. And, finally, we would like to learn more about the 3+1 dimensional model and compare it to CDT in 3+1 dimensions.

A

Software documentation

The chapters 5 - 7 in this thesis are devoted to studying the new model of locally causal dynamical triangulations. A major part of the work on this topic consisted of developing the simulation software to analyze the dynamical properties of the new model in 2+1 dimensions. The programming work turned out to be highly challenging and it took more than a year to implement and test the software until it became reliable to be used for running the simulations, which are discussed in chapter 6. In order to represent the programming part of the project we include here the documentation of the software with detailed instructions on how to compile and use the software.

The software has been developed in C++, taking advantage of object-oriented techniques to make the software modular and flexible. At the time of writing, the latest version of the software contains approximately 15000 lines of code, excluding comment lines. Extensive documentation of the code has been added to make it easier to understand and modify the program. The software currently supports the locally causal DT model of Type 1 in 2+1 dimensions, while the corresponding model of Type 2 is currently not supported. In addition, it is also possible to run the software in CDT-mode.

The software is open source and is available on request. It is released under the MIT license which is an open source license that is less restrictive than the well-known GPL or LPGL licenses. Essentially, the only restriction which is enforced by the MIT license is that the license text must always be included unmodified in every copy of the software. The full license text can be easily found on the internet and is also included in the software package in the file 'LICENSE'.

A.1 Compiling

The software has been developed in C++ and a standard Makefile is present. At the moment only the gcc compiler is supported. The attempt of modifying the Makefile to support other compilers will probably fail because the source code uses some features of the TR1 extension of C++, which is not supported by most compilers. Before attempting to compile the software please make sure that the *zlib* compression library is installed (at least on Linux systems this should always be the case). To compile the software simply run

```
make
```

in the directory where the file 'Makefile' is located. This should produce the executable *dt* in the current directory.

A.2 Usage

The simulation is executed using

```
./dt config-file
```

config-file is a configuration file which controls the behavior of the simulation software. It is described in detail in section A.4. Please note that the software writes several output files to the same directory where the executable is located. Therefore you may want to first create a new directory, copy the executable and the configuration file into the new directory, and run the simulation there.

The software does not spawn a new process and therefore blocks the current shell during its execution, unless it is explicitly started as a background process. If the program is not run as a background process, it is always possible to terminate it by pressing CTRL-C. The software also supports automatic termination after a certain number of Monte Carlo steps or after a certain amount of wall-clock time. In addition, it is possible to schedule the program for termination by modifying the configuration file during the execution of the program. These possibilities will be described later on.

Immediately after starting the program, a log file is created. The file name is given by the *log* field in the [main] section of the configuration file. The log file always contains a short initial part containing some general information about the simulation configuration, followed by a second part which is periodically updated during the simulation. Every entry in the second part displays the current sweep (1 sweep = 10^6 attempted Monte Carlo moves) and a string containing the actual information.

A.3 The output files

The inner core of the simulation software consists of a loop in which Monte Carlo moves are executed. The standard unit to measure the number of Monte Carlo moves is 1 sweep

$= 10^6$ attempted Monte Carlo moves. After a certain number of sweeps, measurements are performed and the results written to output files in standard text format. These measurements are repeated periodically. There are several types of measurements, and each type is associated with its own output file and with own measurement period. Every measurement type has its own section in the configuration file, and every section starts with a line [out_xxxx], where *xxxx* is a string representing that particular measurement type. It is possible to enable/disable each measurement type individually.

The file name of each output file has the following structure: A-B.txt. A is the value of the field *id* which is found in the configuration file section associated to the measurement type. B is the value of the field *id* in the [main] section of the configuration file.

In the following we describe all measurement types. In order to determine the file name associated to the measurement type it is necessary to know the name of the corresponding section in the configuration file. Therefore we always provide this information in the section title or at the beginning of the actual description.

At the beginning of most output files are comment lines starting with a # character. In most cases the last of these lines describes the column structure of the actual measurement data. Typically the first two columns are the current sweep number and the current wall-clock time in seconds. The latter is measured compared to the value of zero at the beginning of the simulation.

A.3.1 Subsimplex counts ([out_general])

Every measurement corresponds to one line. The columns have the following meaning:

sweep current sweep number

time current wall-clock time

N3 number of tetrahedra

N0 number of vertices

N1s number of spacelike links

N1t number of timelike links

N2sss number of spacelike triangles

N2tss number of triangles with one timelike and two spacelike links

N2tts number of triangles with one spacelike and two timelike links

N2ttt number of triangles with three timelike links (currently unused and set to zero)

N3Tx number of tetrahedra of type *x* (see Fig. 5.8 in chapter 5 for a list of all tetrahedron types).

Only the tetrahedron types T2, T3, T5 and T9 are currently supported. T5 corresponds to the 3-1 tetrahedron and T9 to the 2-2 tetrahedron known from CDT. The T2 tetrahedron has one timelike link and the T3 tetrahedron has two timelike links which meet in a single vertex.

A.3.2 Subsimplex distributions ([out_dist])

Foliated triangulations are partitioned into a set of thick slices, therefore it is natural to count all the objects in every slice to produce a distribution of these objects as a function of these slices. We have generalized this idea to also support triangulations which do not have the layered structure of CDT. In this case the triangulation can be partitioned to give a sequence A-B-A-B-..., where A represents a thick slice identical to those found in foliated triangulations and B stands for the sub-triangulation which is located between two neighboring thick slices. Some of the B-type sub-triangulations can be empty. Now we label all these sub-triangulations with an increasing integer starting with 1 and interpret this number as a time variable (later we will describe other notions of time). If two thick slices are directly adjacent, they are assigned adjacent integers numbers, in other words, the empty sub-triangulations are left out. Finally we count all sub-simplices in each sub-triangulation and print these counts together with the associated time. Please note the following remarks:

- Some of the subsimplex counts do not sum up to the global value which is printed in the output file described in section A.3.1. For example this is true for the vertices, which are located on the boundary of each sub-triangulation and are therefore counted more than once when summing up the values for all the sub-triangulations.
- These distributions are only useful for triangulations which are almost strictly foliated. Otherwise the number of sub-triangulations becomes very small and the distribution ceases to provide useful information.

The output file is organized in blocks, where each block is associated to one particular measurement and therefore one particular sweep number. All blocks are separated by an empty line. Every block consists of a number of lines, each of them displaying the number of objects for one sub-triangulation. The column structure is completely analogous to the one described in section A.3.1, except that there is a new column called t (the third column) which is the time associated to the sub-triangulation. The columns 4-22 are therefore analogous to the columns 3-21 of the output file discussed in section A.3.1

A.3.3 Coordination number of vertices [out_vcdist]

The coordination number of a vertex is equal to the number of tetrahedra containing this vertex. The output file is organized in blocks where each block is associated to one particular measurement and therefore one particular sweep number. All blocks are separated by an empty line. The columns have the following meaning:

sweep current sweep number
time current wall-clock time
vcoord an integer value representing a coordination number
count the number of vertices with coordination number *vcoord* (always strictly positive)

Every block is therefore a histogram of all coordination numbers found during the particular measurement.

A.3.4 Tetrahedron distributions ([out_prof])

The measurement type [out_dist] described in section A.3.2 is only useful for triangulations which are (almost) strictly foliated. In order to create a more flexible observable we have defined another notion of time. The triangulation has to satisfy one of the following requirements in order to be compatible with this measurement type:

- The spacetime topology is chosen such that there exists one source and one sink of time, thus time is not periodic. The standard example is the 3-sphere spacetime topology.
- The simulation is run on the restricted ensemble of foliated triangulations with periodic boundary conditions in time.

The time is computed differently for both situations, but both computations locally agree in the case that the triangulation is foliated.

- Case 1:

  Here we have one source and one sink of time. For every vertex we perform a fixed number of random walks forward in time towards the sink and backwards in time towards the source. The number of random walks is configurable, as will be described in section A.4. For every walk we measure the distance to the source and sink and at the end perform an average, which gives two values d_p and d_f. Then we assign the difference $d_f - d_p$ to the vertex as a time variable. Now we loop over all tetrahedra and for each tetrahedron compute the sum of all time values associated to its four vertices. This defines a time variable for the tetrahedron.

- Case 2:

  Here we consider only foliated triangulations. We assign an increasing even integer number to every leaf of the foliation, starting with an arbitrarily chosen spatial slice. This number is then assigned to every vertex in a slice. Finally we compute a time variable for every tetrahedron in the same way as described in case 1.

The output file is organized in blocks where each block is associated to one particular measurement and therefore one particular sweep number. All blocks are separated by an empty line. The columns have the following meaning:

sweep current sweep number
time current wall-clock time
t an integer value representing a time variable
count the number of tetrahedra with time t (positive or zero)

For approximately foliated triangulations the distribution shows an oscillatory behavior, reflecting the alternation between approximate thick slices and decorated thin slices. See chapter 6 for more information on this topic.

A.3.5 Volume distributions ([out_profnew])

In foliated triangulations every spatial slice represents space at a particular time. The number of top-level simplices in every spatial slice is the spatial volume associated to that slice. For example, in 2+1 dimensional foliated triangulations the volume of a spatial slice is equal to the number of spacelike triangles in that slice. The distribution of these volumes as function of the intrinsic time variable is an import observable. In the following we describe an implementation of a volume distribution observable for triangulations which do not have a distinguished foliation.

First we assign a time variable to each vertex. We do this in exactly the same way as described in section A.3.4. Then we randomly generate a subset of all spatial slices in such a way that the slices are evenly distributed along the time direction. The associated algorithm is described in detail in section 7.5. To every spatial slice in the subset we assign a time variable by averaging over all time variables of those vertices which are located on the slice. Finally, for every slice in the subset we print out the time and the volume, the latter being the number of spacelike triangles in the slice.

The output file is organized in blocks where each block is associated to one particular measurement and therefore one particular sweep number. All blocks are separated by an empty line. Every line in each block represents the data for one spatial slice. The columns have the following meaning:

sweep current sweep number
time current wall-clock time
t the time assigned to this spatial slice
count the spatial volume of this spatial slice

In order to get a useful volume distribution observable some nontrivial postprocessing of the raw data is necessary. These steps are described in chapter 6.

A.3.6 Time extension ([out_text])

Given two spatial slices in a foliated triangulation, all future timelike paths connecting both slices have the same length (proper time). If the spacetime topology is that of a 3-sphere, all future timelike paths connecting the source and the sink of time also have the same length time, if the triangulation is foliated. If we consider triangulations which do not have the distinguished foliation, the situation changes and we need a more refined observable to learn more about the timelike extent of the whole system.

In section A.3.4 we have introduced a method to assign a time coordinate to vertices by sending out random walkers and measuring the average distance to the source and sink of time, assuming we work with a spacetime topology with nonperiodic boundary conditions in time. These distances have been denoted by d_p and d_f. While their difference gives us a time variable for the vertex, we can also compute the sum, which gives us the length of the shortest timelike path, which connects the source and the sink of time and which goes through the vertex.

For every vertex we compute the sum $d_p + d_f$ and assign this value to the vertex. In the case of foliated triangulations with periodic boundary conditions, every vertex is assigned the same value, which is the number of leafs of the foliation. For every vertex we then round the computed (floating point) value to the nearest integer and also assign this number to the vertex. Finally, for every integer that appears somewhere, we count how many vertices have been assigned this integer. This produces a histogram of time extension values.

The output file is organized in blocks where each block is associated to one particular measurement and therefore one particular sweep number. The columns have the following meaning:

sweep current sweep number

time current wall-clock time

t an integer value representing an approximate time extension as seen from a vertex

count the number of vertices which agree on the value of *t*

A.3.7 Hausdorff dimension ([out_dimh])

Here we measure the data which is needed to compute the average spacetime Hausdorff dimension. In the first step we attempt to localize a subset of the triangulation with a large spatial extent. All tetrahedra in this subset are then collected in a container. In practice these are all the tetrahedra which share one triangle with a spatial slice of maximal volume.

Afterwards we perform a fixed number of loops (where the fixed number is configurable). In every loop we first randomly choose one of the collected tetrahedra. Then we iteratively construct a partition of the complete triangulation where each element of the partition consists of those tetrahedra which can be reached from the starting tetrahedron by making *r* steps in the dual lattice and for which no shorter path is possible. For each element of the partition we count the number of tetrahedra. After the outer loop is completed we average over all the counts for fixed value of *r*.

The output file is organized in blocks where each block is associated to one particular measurement and therefore one particular sweep number. Every line in a block represents one partition of the triangulation labeled by increasing integer values denoted by *r*. The columns have the following meaning:

sweep current sweep number

time current wall-clock time

r an integer number representing the dual-lattice-distance between the original tetrahedron and the current element of the partition

N3 the number of tetrahedra in the current partition, averaged over the total number of runs

A.3.8 Spectral dimension ([out_dims])

Here we compute the data which is needed to compute the average spacetime spectral dimension. In the first step we determine a subset of the triangulation with a large spatial extent in the same way as described in section A.3.7.

Afterwards we perform a fixed number of loops (where the fixed number is configurable) and in every loop we first randomly choose one of the collected tetrahedra. Then we evolve the discretized scalar diffusion equation using the graph Laplacian as the Laplace operator and using a diffusion constant which at the moment is set to 0.8 (hardcoded). The initial field configuration is a discrete delta distribution centered at the chosen tetrahedron. After every iteration we record the value of the field at the original tetrahedron together with the number of iteration steps elapsed so far, which we call *sigma*. *sigma*=1 corresponds to the initial field configuration. When the outer loop is completed we average over all the values for each value of *sigma*.

The output file is organized in blocks where each block is associated to one particular measurement and therefore one particular sweep number. The columns have the following meaning:

sweep current sweep number

time current wall-clock time

sigma the number of iteration steps in the diffusion process

retprob the value of the scalar field at the original tetrahedron after *sigma* iteration steps, averaged over all runs

ds the spectral dimension, computed using a discretization of the continuum formula $ds = -2(d\log(\text{retprob}(s))/d\log(s))$. Here we use s as abbreviation for *sigma*. We use the following discretization: $ds = -2(\log(\text{retprob}(s+1) - \text{retprob}(s))/(\log(s+1) - \log(s)))$

It is advisable to compute the spectral dimension in a postprocessing step using the *retprob* data, which allows to study different discretizations of the logarithmic derivative and their effect.

The spectral dimension belongs to the most expensive observables with respect to execution time. By tuning some of the configuration parameters, as described later on, a good compromise between accuracy and performance can be achieved.

A.3.9 Spatial Hausdorff dimension ([out_spdimh])

Here we compute the data which is needed to compute the average Hausdorff dimension of spatial slices. First we generate a subset of all spatial slices using the method described in section A.3.5. Then we execute a two-fold nested loop, where the outer loop iterates over a fixed number of these spatial slices while the inner loop iterates over a fixed number of spacelike triangles on the chosen slice. Both of these fixed numbers are

configurable. The chosen triangle is then used as a starting point for a standard Hausdorff dimension measurement, as described in section A.3.7, but adapted to the case where the dimension of the system is reduced by one.

The output file is organized in blocks where each block is associated to one particular measurement and one particular spatial slice. Each block displays the number of triangles at a dual-lattice distance r from the chosen initial triangle, averaged over all runs. In contrast to other measurement types, the sweep number and the wall-clock time are displayed at the beginning of each block in comment lines. Another useful information in the header is *numtri*, which is the number of triangles in the spatial slice. The actual data consists of two numbers per line, the first number being the radius and the second being the number of triangles associated to that radius, averaged over all runs.

A.3.10 Spatial spectral dimension ([out_spdims])

Here we measure the data which is needed to compute the average spectral dimension of spatial slices. First we randomly select one of those spatial slices which has maximal volume. Then we loop over a (configurable) fixed number of spacelike triangles and perform a standard spectral dimension measurement, as described in section A.3.8, but adapted to the case where the dimension of the system is reduced by one. The output file is organized in blocks where each block is associated to one particular measurement and one particular spatial slice. Each block displays the return probability and the spectral dimension as function of the number of iterations, averaged over all runs. In contrast to other measurement types, the sweep number and the wall-clock time are displayed at the beginning of each block in comment lines. Another useful information in the header is *numtri*, which is the number of triangles in the spatial slice. The actual data consists of three numbers per line. The first number is the number of iterations of the diffusion algorithm, defined in the same way as for the spacetime spectral dimension measurement described in section A.3.8. The second and third numbers are the return probability and the spectral dimension, which are defined in the same way as in section A.3.8.

A.3.11 Simplified volume distributions ([out_spvol])

This measurement type describes a simpler volume distribution observable compared to the one described in section A.3.5. First we use the method described in section A.3.2 to create a partition of the whole triangulation, and we assign an increasing integer as a time variable to every element of the partition. For every sub-triangulation in the partition we measure the number of spacelike triangles on its past and its future boundary.

The output file is organized in blocks where each block is associated to one particular measurement and therefore one particular sweep number. All blocks are separated by an empty line. Every line in each block represents the data for one element of the partition. The columns have the following meaning:

sweep current sweep number

time current wall-clock time
t the time assigned to this element of the partition
spvollow number of spacelike triangles on the past boundary of the sub-triangulation
spvolhigh number of spacelike triangles on the future boundary of the sub-triangulation

This measurement type is only useful for (almost) strictly foliated triangulations.

A.3.12 Volume fixing information ([out_volfix])

The simulation software employs a special mechanism to dynamically stabilize the system size, as described in chapter 6. It involves a periodic update of the volume term in the action. This measurement type allows to monitor this process. Every measurement corresponds to one line in the output file. Currently there exist two output formats depending on the selected action:

- Case 1: The action is constrained to be of Regge type

  sweep current sweep number

  time current wall-clock time

  lambda the coupling in front of the volume term

  coeff[N3] The original Regge action is internally converted into a form which is linear in the subsimplex counts. *coeff[N3]* is the coefficient which multiples N3, the total number of tetrahedra.

- Case 2: The action is not constrained to be of Regge type and is defined as a linear combination of subsimplex counts

  sweep current sweep number

  time current wall-clock time

  coeff[N3] the coefficient which multiples N3, the total number of tetrahedra

A.3.13 Bubble volumes ([out_bubvol])

A bubble is defined as a sub-triangulation which is generated using a breadth-first traversal process which does not cross spacelike triangles. The thick slices known from foliated triangulations are examples of such bubbles. The volume of a bubble is the number of tetrahedra contained in the bubble.

The output file is organized in blocks where each block is associated to one particular measurement and therefore one particular sweep number. All blocks are separated by an empty line. Every line in each block displays the distribution of bubble volumes. The columns have the following meaning:

sweep current sweep number

time current wall-clock time
volume an integer representing a bubble volume
count the number of bubbles with the volume *volume* (strictly positive)

A.3.14 Bubble circumferences, variant 1 ([out_bubcirc])

There are several ways of how to define a notion of circumference for bubbles. Here we count the number of tetrahedra of types T2 and T3 and print their sum.

The output file is organized in blocks where each block is associated to one particular measurement and therefore one particular sweep number. All blocks are separated by an empty line. Every line in each block displays the distribution of bubble circumferences. The columns have the following meaning:

sweep current sweep number
time current wall-clock time
circumference an integer representing a bubble circumference
count the number of bubbles with the circumference *circumference* (strictly positive)

A.3.15 Bubble circumferences, variant 2 ([out_bubT2])

Here we use an alternative definition of a bubble circumference, which is simply the number of tetrahedra of type T2 in the bubble.

The output file is organized in blocks where each block is associated to one particular measurement and therefore one particular sweep number. All blocks are separated by an empty line. Every line in each block displays the distribution of bubble circumferences. The columns have the following meaning:

sweep current sweep number
time current wall-clock time
numT2 an integer number representing a bubble circumference
count the number of bubbles with the circumference *numT2* (strictly positive)

A.3.16 Monte Carlo move statistics ([out_stat])

Here we monitor the acceptance rates of all the implemented Monte Carlo moves. Every measurement corresponds to one line in the output file. If some Monte Carlo moves are disabled, the corresponding columns are not displayed in the output file. The columns *xxxacc* represent the number of accepted moves since the last measurement and the columns *xxxrate* represent the fraction of accepted moves compared to all attempted moves since the last measurement.

sweep current sweep number
time current wall-clock time

M26acc # accepted moves (2-6 move)

M62acc # accepted moves (6-2 move)

M23acc # accepted moves (2-3 move)

M32acc # accepted moves (3-2 move)

M44acc # accepted moves (4-4 move)

SLEXacc # accepted moves (inverse of spatial link collapse move)

SLCOacc # accepted moves (spatial link collapse move)

TLEXacc # accepted moves (bubble creation move)

TLCOacc # accepted moves (bubble deletion move)

PINCHacc # accepted moves (pinching creating move)

UNPINacc # accepted moves (pinching deletion move)

M14Pacc # accepted moves (polar move of type 1-4)

M41Pacc # accepted moves (polar move of type 4-1)

LITSacc # accepted moves (link type swap timelike → spacelike)

LISTacc # accepted moves (link type swap spacelike → timelike)

TOTacc # accepted moves (all moves together)

M26acc acceptance rate (2-6 move)

M62acc acceptance rate (6-2 move)

M23acc acceptance rate (2-3 move)

M32acc acceptance rate (3-2 move)

M44acc acceptance rate (4-4 move)

SLEXacc acceptance rate (inverse of spatial link collapse move)

SLCOacc acceptance rate (spatial link collapse move)

TLEXacc acceptance rate (bubble creation move)

TLCOacc acceptance rate (bubble deletion move)

PINCHacc acceptance rate (pinching creating move)

UNPINacc acceptance rate (pinching deletion move)

M14Pacc acceptance rate (polar move of type 1-4)

M41Pacc acceptance rate (polar move of type 4-1)

LITSacc acceptance rate (link type swap timelike → spacelike)

LISTacc acceptance rate (link type swap spacelike → timelike)

A.3.17 Refined statistics on individual Monte Carlo moves

For every Monte Carlo move there exists an additional output file with similar structure. The associated section names are listed in the following:

[out_m26stat] 2-6 move
[out_m62stat] 6-2 move
[out_m23stat] 2-3 move
[out_m32stat] 3-2 move
[out_m44stat] 4-4 move
[out_mslexstat] inverse of spatial link collapse move
[out_mslcostat] spatial link collapse move
[out_mtlexstat] bubble creation move
[out_mtlcostat] bubble deletion move
[out_mpinchstat] pinching creation move
[out_munpinchstat] pinching deletion move
[out_m14polestat] polar move of type 1-4
[out_m41polestat] polar move of type 4-1
[out_mlitsstat] link type swap timelike → spacelike
[out_mliststat] link type swap spacelike → timelike

Every line displays information collected since the last measurement. The column structure depends on the move, but always follows a similar pattern:

sweep current sweep number
time current wall-clock time
xxxatt the number of attempted moves
xxxacc the number of accepted moves
xxxrej1 . . . xxxrejN N fields displaying the number of rejected moves due to a particular reason, the latter being parametrized by the integer in the range [1. . . N].

Here is a list of all the rejection reasons (0 stands for rejections which never happen unless explicitly enabled by modifying the source code)

- **[any]**

  xxxrej1 upper limit for vertices exceeded
  xxxrej2 upper limit for links exceeded
  xxxrej3 upper limit for tetrahedra exceeded

- **[any of out_m26stat, out_m62stat, out_m23stat, out_m32stat, out_m44stat]**

xxxrej4 0
xxxrej5 the randomly chosen local environment does not support this move
xxxrej6 detailed balance rejection

- **[out_m26stat]**

  xxxrej7 the randomly chosen local environment does not support this move

- **[out_m62stat]**

  xxxrej7 the randomly chosen local environment does not support this move
  xxxrej8 creation of a double triangle
  xxxrej9 0

- **[out_m23stat]**

  xxxrej7 creation of a double link

- **[out_m32stat]**

  xxxrej7 the randomly chosen local environment does not support this move
  xxxrej8 creation of a double triangle

- **[out_m44stat]**

  xxxrej7 the randomly chosen local environment does not support this move
  xxxrej8 creation of a double link

- **[out_mslexstat]**

  xxxrej4 the disc selection algorithm failed
  xxxrej5 the move violates local causality
  xxxrej6 the move is identical to a 2-6 move
  xxxrej7 detailed balance rejection
  xxxrej8 0

- **[out_mslcostat]**

  xxxrej4 the randomly chosen local environment does not support this move
  xxxrej5 the move is identical to a 6-2 move
  xxxrej6 the move does not yield a combinatorial triangulation
  xxxrej7 the disc selection algorithm failed

xxxrej8 detailed balance rejection
xxxrej9 0

- **[out_mtlexstat]**

  xxxrej4 detailed balance rejection
  xxxrej5 0

- **[out_mtlcostat]**

  xxxrej4 there are no T2 tetrahedra in the whole triangulation
  xxxrej5 the randomly chosen local environment does not support this move
  xxxrej6 the move does not yield a combinatorial triangulation
  xxxrej7 detailed balance rejection
  xxxrej8 0

- **[out_mpinchstat]**

  xxxrej4 there are no T5 tetrahedra in the whole triangulation
  xxxrej5 the randomly chosen local environment does not support this move
  xxxrej6 the move does not yield a combinatorial triangulation
  xxxrej7 the move violates local causality
  xxxrej8 detailed balance rejection
  xxxrej9 0

- **[out_munpinchstat]**

  xxxrej4 there are no T3 tetrahedra in the whole triangulation
  xxxrej5 the randomly chosen local environment does not support this move
  xxxrej6 detailed balance rejection
  xxxrej7 0

- **[out_m14polestat]**

  xxxrej4 no source or sink of time found
  xxxrej5 detailed balance rejection

- **[out_m41polestat]**

  xxxrej4 no source or sink of time found
  xxxrej5 the randomly chosen local environment does not support this move

xxxrej6 detailed balance rejection

- **[any of out_mlitsstat, out_mliststat]**

  xxxrej4 the randomly chosen local environment does not support this move

  xxxrej5 0

  xxxrej6 the move violates local causality (invalid building blocks)

  xxxrej7 the move violates local causality (invalid light cone structure)

  xxxrej8 detailed balance rejection

A.4 The configuration file

The behavior of the simulation software is controlled by customizing the configuration file. It consists of several sections beginning with an entry [xxx] where *xxx* is a keyword describing the purpose of the section. All other lines in a section have the form 'key = value' where *key* is a fixed keyword associated to a particular property of the simulation software, and *value* is the associated value, which can be of type integer, string or boolean. All values are given without quotes, and boolean values can either be *true* or *false*. In the following we describe the configuration file in detail. A sample configuration file, which has been used for the simulations described in chapter 6, is located in the subdirectory 'simulations/test'.

A.4.1 The [main] section

id This is an arbitrary string, which is used to define the file names of output files. The main purpose is to incorporate some information about the particular simulation run into the file names. Some other elements of the configuration file allow for the possibility to incorporate this identifier by including the string '$id' in the value field.

mode There are four possible values:

'endless' The simulation does not stop until the process is killed or CTRL-C is pressed in the shell.

'counter' The simulation stops when the current sweep number reaches a maximal value (see *numsweeps* below)

'timer' The simulation stops after running for a fixed amount of wall-clock time (see *numseconds* below).

'quit' The simulation stops instantly (the *mode* field is periodically reread, thus changing the value to 'quit' during the simulation is one method of terminating the program).

numseconds The number of seconds (wall-clock time) until the simulation stops, if *mode* is set to 'timer'.

numsweeps The simulation stops when the current sweep number reaches the value *numsweeps* and if *mode* is set to 'counter'.

init There are three possible values:

'build' The simulation starts with a fixed initial triangulation (see *initial* in the [lattice] section how to choose an initial triangulation)

'load' The simulation loads a triangulation from a file. The file name is specified in the *conf* field. See *saveconf* and *conf_out* to learn more about saving a triangulation to a file.

'restore' The simulation loads a saved simulation state from a file. The file name is specified in the *state* field. See *savestate* and *state_out* to learn more about saving the state of a simulation. Note that loading a simulation state has the effect that some properties specified in the configuration file are overridden because the associated information is imported from the file.

reloadperiod : The configuration file is reread every *reloadperiod* sweeps. Only a subset of all properties is updated in such a reloading process. We will list all these properties later on.

checkperiod : The lattice integrity is checked every *checkperiod* sweeps. If any defect is encountered, the simulation stops instantly without saving the triangulation or the simulation state.

saveconf : A boolean value, which is *true* if the triangulation should be saved to a file at regular intervals and at the end of the simulation. The file name is specified in *conf_out*. Along with the simulation data, the current values of the couplings in the action plus the relative weights of the Monte Carlo moves are saved.

saveconfperiod If *saveconf* is *true*, the triangulation is saved every *saveconfperiod* sweeps.

savestate A boolean value, which is *true* if the simulation state should be saved to a file at regular intervals and at the end of the simulation. The file name is specified in *state_out*. Simulation states include the information which is saved when *saveconf* is set to *true*, but more information is stored, such as the current state of the random number generator, the current state of the volume fixing mechanism, and the sweep number. The complete information allows to restore a simulation state, such that the restored simulation should behave in exactly the same way as the original simulation after the information has been saved. There are mainly two situations where simulation states should be preferred over the (smaller) triangulation files:

- Some problem has occurred that needs to be reproduced
- A simulation has been interrupted and needs to be restarted. Then the output files can be configured such that the new data is appended to the old data, and it is desirable that the sweep number is not reset to zero, but set to the value at the end of the previous simulation run.

savestateperiod If *savestate* is *true*, the simulation state is saved every *savestateperiod* sweeps.

conf_load_couplings : This boolean field is only effective if *init* is set to 'load'. If this field is set to *true*, the values of the couplings in the action are also imported, when the triangulation is read from the file. The couplings are also protected from being updated during the regular reloading of the configuration file (see *force_update* in the [action] section to learn how to remove this protection). Note that the coupling in front of the volume term is **not** imported unless the volume fixing mechanism is disabled.

conf_load_weights This boolean field is only effective if *init* is set to *load*. If this field is set to *true*, the relative weights of the Monte Carlo moves are also imported when the triangulation is read from the file. These weights are also protected from being updated during the regular reloading of the configuration file (see *force_update* in the [mcmoves] section to learn how to remove this protection).

conf The name of the file used to load a triangulation, when *init* set to 'load'. This value may contain the string '$id' which is then replaced by the value of the *id* field of the [main] section.

conf_out The file name used to save a triangulation, if *saveconf* is set to *true*. This value may contain the string '$id' which is then replaced by the value of the *id* field of the [main] section. It is allowed to choose the same file name as specified in *conf*.

state The name of the file used to load a simulation state, when *init* set to 'restore'. This value may contain the string '$id' which is then replaced by the value of the *id* field of the [main] section.

state_out The file name used to save a simulation state, if *savestate* is set to *true*. This value may contain the string '$id' which is then replaced by the value of the *id* field of the [main] section. It is allowed to choose the same file name as specified in *state*.

log The name of the log file. This value may contain the string '$id' which is then replaced by the value of the *id* field of the [main] section.

A.4.2 The [random] section

generator : Defines which pseudo random number generator (RNG) to use. Possible choices are:

- 'LFIB4' (RNG created by George Marsaglia)
- 'KISS' (RNG created by George Marsaglia)
- 'MT19937' (RNG created by Makoto Matsumoto and Takuji Nishimura, also known as 'Mersenne Twister')
- 'LFIB_607_273'

All these generators are high quality generators. We tested them using the *TestU01* software and found the LFIB4 generator to perform best, closely followed by the KISS generator.

seed The seed value for the random number generator. If a value of zero is specified, the seed is computed using 'time(NULL)' which makes the seed depend on the current time with the granularity of one second.

A.4.3 The [lattice] section

maxsize The upper limit for the number of tetrahedra.

vfixenabled A boolean value which needs to be set to *true* if the volume fixing mechanism should be enabled. The volume fixing mechanism is essentially a reimplementation of the corresponding mechanism in the CDT simulation software for 3+1 dimensions developed by Andrzej Görlich.

volumefixing There are two possible values:

'cusp' The volume fixing is realized using a cusp-shaped potential. Some properties of the volume fixing mechanism are specified in the section [volumefixing_default].

'parabolic' The volume fixing is realized using a parabolic potential. Some properties of the volume fixing mechanism are specified in the section [volumefixing_parabolic].

See chapter 6 for more details on these potentials.

vfixforceupdate Some properties of the volume fixing are defined in the [volumefixing_xxx] sections. If the *init* field in the [main] section is set to 'restore', these properties are protected from being overwritten during configuration file reloads. If *vfixforceupdate* is set to *true*, the protection is removed and the properties are updated using the values found in the configuration file, when the file is reloaded the next time.

targetsize The volume fixing mechanism aims to let the total number of tetrahedra fluctuate around the value *targetsize*.

initial If the *init* field in the [main] section is set to 'build', the simulation starts with a newly created triangulation. The *initial* field allows to select such a triangulation. Currently there is only one choice:

'cdt_s2min' A foliated triangulation with a fixed number of time steps (configurable, see the [lat_init_cdt] section) and where the spatial geometry is close to minimal.

time_periodic If this boolean field is set to *true*, the initial triangulation is built with time compactified to a circle. If it is set to *false*, the triangulation is built with the spacetime topology of the 3-sphere with one source and one sink of time at the two poles.

A.4.3.1 The [lattice_init_cdt] section

time_extent Specifies the number of time steps in any foliated triangulation that is built to start a new simulation.

A.4.3.2 The [volumefixing_xxx] section

eps The multiplicative factor in front of the volume fixing potential in the action. Useful values are:

- 0.02 (cusp potential)
- 0.00001 (parabolic potential)

window The volume fixing potential measures how often the simulation stays on the upper branch and on the lower branch of the potential. Only events lying outside the interval [*targetsize-window*,*targetsize+window*] are counted. A useful value is 10.

threshold Even with a working volume fixing mechanism the simulation will always spend a bit more time on one half of the potential compared to the other. Therefore a threshold value is needed to decide when the asymmetry is strong enough to justify a modification of the volume term of the action. A useful value is 0.2.

A.4.4 The [latticeconstraints] section

type There are two possible values:

'cdt' The triangulation is required to be foliated at all times. The simulation should then yield results consistent with those results obtained using other CDT simulation software in 2+1 dimensions.

'lorentzian' The triangulation is required to be locally causal of Type 1, thus no foliation is enforced.

allow_bubble_merge If this boolean value is set to *false*, the ensemble of locally causal triangulations of Type 1 is further reduced by disallowing any Monte Carlo move which joins or splits bubbles. This is the setting we used for the simulations described in chapter 6. A value of *true* may lead to severe thermalization problems. It is currently not clear whether enabling these option alters the dynamics significantly.

tetratypes_override If this boolean value is set to *true*, some of the tetrahedra types can be manually disabled (see the [tetratypes] section below).

A.4.4.1 The [tetratypes] section

If this section is omitted then all building blocks which are compatible with the model chosen by the *type* field in the [lattice] section are allowed. The following options are only effective if the field *tetratypes_override* in the [latticeconstraints] section is set to *true*.

t2 *false* if no tetrahedra of type T2 should be allowed
t3 *false* if no tetrahedra of type T3 should be allowed
t5 *false* if no tetrahedra of type T5 should be allowed
t9 *false* if no tetrahedra of type T9 should be allowed

A.4.5 The [action] section

As described in chapter 5, the space of all Regge actions is strictly smaller than the space of all actions linear in the subsimplex count variables, if we consider the model of locally causal triangulations. Therefore one can choose to work in either space. If one works with the Regge action, the original Regge couplings are used to drive the simulation.

action : There are four possible choices:

- **'regge_n0'** The action is constrained to be of Regge type and the action is internally converted into a linear action including a c*N0-term. The [action_regge] section allows to specify the couplings.
- **'regge_n1'** The action is constrained to be of Regge type and the action is internally converted into a linear action including a c*N1-term. The [action_regge] section allows to specify the couplings.
- **'linear_n0'** The action is a generic linear combination using the following subsimplex counts: N0, N3, N3T2, N3T3, N3T5. The [action_linear_n0] section allows to specify the couplings.
- **'linear_n1'** The action is a generic linear combination using the following subsimplex counts: N1, N3, N3T2, N3T3, N3T5. The [action_linear_n1] section allows to specify the couplings.

There is virtually no difference between *regge_n0* and *regge_n1*, except that the log file displays the value of the coupling in front of the N0 term or N1 term, depending on the choice. The distinction may be useful when comparing results obtained using this simulation with results obtained using other CDT simulations, which may either work with a N0 term or with a N1 term. The value of the coupling in front of the corresponding term can then be compared between both simulations.

force_update There are two situations in which the couplings of the action are protected from being updated during the periodic reloading of the configuration file:

- The *init* field in the [main] section is set to 'load' and *conf_load_couplings* in the [main] section is set to 'true'.
- The *init* field in the [main] section is set to 'restore'.

If *force_update* is set to *true*, the protection is removed and the couplings are updated using the values found in the configuration file when the file is reloaded the next time. The coupling in front of the volume term is only updated if the volume fixing mechanism is disabled.

A.4.5.1 The [action_regge] section

This section is only effective if the *action* field in the [action] section is set to 'regge_n0' or 'regge_n1'.

k The value of the coupling which multiplies the curvature term. It is proportional to the inverse bare Newton constant.

lambda The value of the coupling which multiplies the volume term. This value is only taken into account in one of the following situations:

- At the beginning of the simulation if the *init* field in the [main] section is set to 'build'.
- Whenever the configuration file is reloaded and the volume fixing is disabled.

alpha The asymmetry parameter which parametrizes how much the squared lengths of the timelike and spacelike links differ.

volfixstepsize The volume fixing term constantly modifies the volume term. This value allows to control how large these modifications are in each step. A useful value is 0.01.

A.4.5.2 The [action_linear_n0] and [action_linear_n1] sections

These sections are only effective if the *action* field in the [action] section is set to 'linear_n0' resp. 'linear_n1'.

coeff_N0 The coefficient in front of the N0 term (only in the [action_linear_n0] section).

coeff_N1 The coefficient in front of the N1 term (only in the [action_linear_n1] section).

coeff_N3 The coefficient in front of the N3 term. Similar restrictions apply as in the case of *lambda* in the [action_regge] section.

coeff_N3T2 : The coefficient in front of the N3T2 term.

coeff_N3T3 : The coefficient in front of the N3T3 term.

coeff_N3T5 : The coefficient in front of the N3T5 term.

A.4.6 The [mcmoves] section

This section allows to define the relative weights of all the Monte Carlo moves. A higher weight means that the move is proposed more often. A general rule of thumb is that the number of accepted moves of any type should be about equal. If one move is not accepted very often one can compensate by assigning a higher weight to it. In practice the situation is much more complicated, because maximal performance is not the same as optimal acceptance rate. Some moves are much slower than other moves, therefore executing them too often can do more harm than good.

The weights of a move and its inverse can be set independently. Quite often we have the situation that one move is dominantly rejected because of geometrical/topological

reasons while the inverse is dominantly rejected by detailed balance. The latter rejections are typically the most expensive ones, thus it is sometimes advantageous to execute the moves of the first category more often, because this automatically reduces the number of failures of their inverse due to detailed balance.

The best practice is probably to start with established values (examples are given below), to monitor the acceptance rates and perform modifications if necessary.

force_update There are two situations in which the relative weights are protected from being updated during the periodic reloading of the configuration file:

- The *init* field in the [main] section is set to 'load' and *conf_load_weights* in the [main] section is set to 'true'.
- The *init* field in the [main] section is set to 'restore'.

If *force_update* is set to *true*, the protection is removed and the couplings are updated using the values found in the configuration file when the file is reloaded the next time.

weight_xxx The relative weight of the associated move:

weight_pachner_26 2-6 move

weight_pachner_62 6-2 move

weight_pachner_23 2-3 move

weight_pachner_32 3-2 move

weight_pachner_44 4-4 move

weight_tlexpand bubble creation move

weight_tlcollapse bubble deletion move

weight_slexpand inverse of spatial link collapse move

weight_slcollapse spatial link collapse move

weight_pinch pinching creation move

weight_unpinch pinching deletion move

weight_m14pole polar move of type 1-4

weight_m41pole polar move of type 4-1

weight_linkst link type swap spacelike → timelike

weight_linkts link type swap timelike → spacelike

For the simulations described in chapter 6 we used the following values:

```
weight_pachner_26 = 0.3
weight_pachner_62 = 15
weight_pachner_23 = 0.75
```

```
weight_pachner_32 = 0.4
weight_pachner_44 = 0.5
weight_tlexpand = 0.15
weight_tlcollapse = 0.5
weight_slexpand = 0.2
weight_slcollapse = 2
weight_pinch = 0.5
weight_unpinch = 0.5
weight_m14pole = 0.1
weight_m41pole = 0.1
weight_linkst = 0.1
weight_linkts = 0.1
```

A.4.7 The [output] section

Every output file described in section A.3 has one section in the configuration file associated to it, and the relation between the section name and the output file has also been given in section A.3. All section names have the form '[out_xxx]'. Some fields are common to all sections:

id A string which is part of the file name associated to the output file.

append If this boolean field is set to *true*, new data is appended to existing data, if the output file already exists. If it is set to *false*, a new output file is created in any case at the beginning of the simulation.

exact_size If this boolean field is *false*, measurements are always done at the beginning of a sweep, regardless of the actual system size. If it is *true*, the simulation waits until the system size exactly matches the value of the field *targetsize* in the [lattice] section. Only one measurement is done during one sweep. In principle it can happen that the target size is never reached during some sweep (for example this happens if the volume fixing potential is chosen flat enough such that the fluctuations become large), then the measurement is postponed to the next sweep.

period The number of sweeps between two successive measurements.

buffersize The software uses buffered IO to minimize IO traffic. New data is written to an internal buffer of size *buffersize*. As soon as the buffer is full it is written to the file. A typical value is 4096, corresponding to a buffer size of 4KB. For debugging purposes the value can be set to zero, which is equivalent to having unbuffered IO.

Some fields are common to several output files, these are therefore located in the [output] section:

num_layers Some measurement types rely on sampling a fixed number of spatial slices. This field defines how many spatial slices are generated. A typical value is 1000. This field is effective for the following output files:

- out_profnew
- out_dimh
- out_dims
- out_spdimh
- out_spdims

dist_averaging Some measurement types define a time variable by performing a fixed number of random walks. This field specifies how many of these walks are being done. A typical value is 100. This field is effective for the following output files:

- out_prof
- out_profnew
- out_text
- out_dimh
- out_dims
- out_spdimh
- out_spdims

Some sections related to output files contain additional fields. These are discussed in the following.

A.4.7.1 The [out_dimh] section

num_runs The measurement of the spacetime Hausdorff dimension, as described in section A.3.7, involves performing a fixed number of runs. *num_runs* defines this number. A typical value is 100.

A.4.7.2 The [out_dims] section

num_runs The measurement of the spacetime spectral dimension, as described in section A.3.8, involves performing a fixed number of runs. *num_runs* defines this number.

num_sigma The number of iterations for each diffusion process.

Since the spectral dimension measurement is expensive, large values for *num_sigma* typically need to be compensated by using a small value for *num_runs*, such as *num_runs*=1'. Then no averaging over starting positions for the diffusion process is done.

A.4.7.3 The [out_spdimh] section

num_runs The measurement of the spatial Hausdorff dimension, as described in section A.3.9, involves performing a fixed number of runs. *num_runs* defines this number. A typical value is 100.

num_layers The field *num_layers* in the [output] section (not to be confused with this field here) specifies how many spatial slices are generated by the software. This field here defines how many of these generated slices are randomly selected to perform a measurement of the spatial Hausdorff measurement. A typical value is 100.

A.4.7.4 The [out_spdims] section

num_runs The measurement of the spatial spectral dimension, as described in section A.3.10, involves performing a fixed number of runs. *num_runs* defines this number. A typical value is 100.

num_sigma The number of iterations for each diffusion process.

A.4.8 Reloadable properties

The configuration file is reloaded periodically. Not all properties are updated during such a reloading process. In the following we list all those fields of the configuration files, which are associated with reloadable properties. Note that some of these fields may not lead to an update of the corresponding property because the latter are protected from being overwritten in specific situations. As described previously there exist special fields (which typically are called *force_update*) which allow to remove the protection.

- [main]/mode
- [main]/numseconds : This value is still compared with the difference of the current time and the time when the simulation was started.
- [main]/numsweeps
- [main]/reloadperiod
- [main]/checkperiod
- [main]/saveconf
- [main]/saveconfperiod
- [main]/conf_out
- [main]/savestate
- [main]/savestateperiod
- [main]/state_out

- [lattice]/targetsize
- [lattice]/vfixenabled
- [lattice]/vfixforceupdate
- [action]/force_update
- [action_xxx]/<all fields>
- [volumefixing_xxx]/<all fields>
- [mcmoves]/<all fields>

A.5 Using clusters

This software does not employ non-trivial parallelization. The following trivial parallelization strategies can be applied:

- Running multiple simulations at various locations in the phase diagram.
- Running multiple simulations with identical parameters, but different seeds for the random number generator. This requires that the initial triangulation is thermalized. All the simulations then trace out different paths in the space of all configurations and therefore gather statistically independent data. To produce a thermalized configuration, a simulation needs to be run beforehand, and that simulation cannot be parallelized.

Some clusters have restrictions concerning the total running time. To run longer simulations the following strategy can be applied:

- In the [main] section of the configuration file, set the *mode* to 'counter' and *numseconds* to a value, which is a few minutes shorter than the maximal running time on the cluster. This prevents the situation that the simulation is abruptly killed before it is capable of quitting in an ordinary fashion. If measurements are being performed that take much longer than a minute per measurement cycle, more time should be subtracted, because the simulation cannot quit ordinarily during a measurement cycle.
- Configure the output files such that new data is appended to the old data.
- Use simulation states instead of triangulation files, because the sweep number is only saved in the former. This guarantees continuity of the output data, which is very useful when using plotting programs to display the Monte Carlo evolution of observables.

- Using the same file names to load and save simulation states allows to restart the simulations with almost no effort, because no file renamings are necessary. If the first simulation starts with a newly constructed triangulation, some modifications are necessary before starting the next simulation (at least the *init* field in the [main] section needs to be changed).

The typical memory usage per simulation is of the order of a few hundred MBytes for system sizes of the order of 100000 tetrahedra. If many instances of the simulation run on a processor with many cores, the total amount of memory available may become an issue.

The output files can become quite large depending on the periods defined in the configuration file. Since the size grows linearly with wall-clock time, a good strategy is to first run a short test run, measure the file sizes and extrapolate the values to realistic situations. If these values are unacceptably large, some of the periods can be enlarged.

Sometimes it is convenient to use unbuffered IO while testing the simulation, but it is very important to restore the configuration to buffered IO when running the simulation on clusters. Otherwise the performance will be much reduced, often also for other users of the clusters, if no local storage is accessible. If local storage is available, it is advisable to use it, although one should take into account that the data may get lost if the processes are killed.

Never trust the backup system of the cluster.

Bibliography

[1] S. Chatrchyan et al. Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC. *Phys.Lett.*, B716:30–61, 2012, 1207.7235.

[2] S.G. Turyshev. Experimental Tests of General Relativity. *Ann.Rev.Nucl.Part.Sci.*, 58:207–248, 2008, 0806.1731.

[3] R.A. Hulse and J.H. Taylor. Discovery of a pulsar in a binary system. *Astrophys.J.*, 195:L51–L53, 1975.

[4] J.M. Weisberg and J.H. Taylor. Relativistic binary pulsar B1913+16: Thirty years of observations and analysis. *ASP Conf.Ser.*, 328:25, 2005, astro-ph/0407149.

[5] B.S. Sathyaprakash and B.F. Schutz. Physics, Astrophysics and Cosmology with Gravitational Waves. *Living Rev.Rel.*, 12:2, 2009, 0903.0338.

[6] G. 't Hooft and M.J.G. Veltman. One loop divergencies in the theory of gravitation. *Annales Poincare Phys.Theor.*, A20:69–94, 1974.

[7] M.H. Goroff and A. Sagnotti. Quantum gravity at two loops. *Phys.Lett.*, B160:81, 1985.

[8] J. Polchinski. *String theory. Vol. 1: An introduction to the bosonic string*. Cambridge University Press, 1998.

[9] J. Polchinski. *String theory. Vol. 2: Superstring theory and beyond*. Cambridge University Press, 1998.

[10] M.B. Green, J.H. Schwarz, and E. Witten. *Superstring Theory. Vol. 1: Introduction*. Cambridge University Press, 1988.

[11] M.B. Green, J.H. Schwarz, and E. Witten. *Superstring Theory. Vol. 2: Loop Amplitudes, Anomalies and Phenomenology*. Cambridge University Press, 1988.

[12] A. Ashtekar. New Variables for Classical and Quantum Gravity. *Phys.Rev.Lett.*, 57:2244–2247, 1986.

[13] C. Rovelli. Loop Quantum Gravity. *Living Rev.Rel.*, 11:5, 2008.

[14] S. Weinberg. Ultraviolet divergences in quantum theories of gravitation. In *General Relativity: An Einstein centenary survey*, pages 790–831. Cambridge University Press, 1979.

[15] S.M. Christensen and M.J. Duff. Quantum gravity in 2+epsilon dimensions. *Phys.Lett.*, B79:213, 1978.

[16] R. Gastmans, R. Kallosh, and C. Truffin. Quantum gravity near two dimensions. *Nucl.Phys.*, B133:417, 1978.

[17] H. Kawai, Y. Kitazawa, and M. Ninomiya. Ultraviolet stable fixed point and scaling relations in (2+epsilon)-dimensional quantum gravity. *Nucl.Phys.*, B404:684–716, 1993, hep-th/9303123.

[18] M. Reuter. Nonperturbative evolution equation for quantum gravity. *Phys.Rev.*, D57:971–985, 1998, hep-th/9605030.

[19] D.F. Litim. Fixed points of quantum gravity. *Phys.Rev.Lett.*, 92:201301, 2004, hep-th/0312114.

[20] H.W. Hamber and R.M. Williams. Nonlocal effective gravitational field equations and the running of Newton's G. *Phys.Rev.*, D72:044026, 2005, hep-th/0507017.

[21] M. Reuter and F. Saueressig. Functional renormalization group equations, asymptotic safety, and quantum Einstein gravity. In *Geometric and Topological Methods for Quantum Field Theory*. Cambridge University Press, 2010, 0708.1317.

[22] A. Codello, R. Percacci, and C. Rahmede. Investigating the ultraviolet properties of gravity with a Wilsonian renormalization group equation. *Annals Phys.*, 324:414–469, 2009, 0805.2909.

[23] D. Benedetti, P.F. Machado, and F. Saueressig. Asymptotic safety in higher-derivative gravity. *Mod.Phys.Lett.*, A24:2233–2241, 2009, 0901.2984.

[24] M. Roček and R.M. Williams. Quantum Regge calculus. *Phys.Lett.*, B104:31, 1981.

[25] A. Perez. Spin foam models for quantum gravity. *Class.Quant.Grav.*, 20:R43, 2003, gr-qc/0301113.

[26] R. Loll. Discrete Approaches to Quantum Gravity in Four Dimensions. *Living Rev.Rel.*, 1:13, 1998, gr-qc/9805049.

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

[28] J. Ambjørn and J. Jurkiewicz. Four-dimensional simplicial quantum gravity. *Phys.Lett.*, B278:42–50, 1992.

[29] M.E. Agishtein and A.A. Migdal. Critical behavior of dynamically triangulated quantum gravity in four-dimensions. *Nucl.Phys.*, B385:395–412, 1992, hep-lat/9204004.

[30] S. Catterall, J.B. Kogut, and R. Renken. Phase structure of four-dimensional simplicial quantum gravity. *Phys.Lett.*, B328:277–283, 1994, hep-lat/9401026.

[31] P. Bialas, Z. Burda, A. Krzywicki, and B. Petersson. Focusing on the fixed point of 4-D simplicial gravity. *Nucl.Phys.*, B472:293–308, 1996, hep-lat/9601024.

[32] B.V. de Bakker. Further evidence that the transition of 4-D dynamical triangulation is first order. *Phys.Lett.*, B389:238–242, 1996, hep-lat/9603024.

[33] J. Ambjørn and R. Loll. Nonperturbative Lorentzian quantum gravity, causality and topology change. *Nucl.Phys.*, B536:407–434, 1998, hep-th/9805108.

[34] J. Ambjørn, J. Jurkiewicz, and R. Loll. Dynamically triangulating Lorentzian quantum gravity. *Nucl.Phys.*, B610:347–382, 2001, hep-th/0105267.

[35] J. Ambjørn, J. Jurkiewicz, and R. Loll. Reconstructing the universe. *Phys.Rev.*, D72:064014, 2005, hep-th/0505154.

[36] S. Carlip. Spontaneous dimensional reduction in short-distance quantum gravity? In *AIP Conference Proceedings*, volume 1196, pages 72–80, 2009, 0909.3329.

[37] J. Ambjørn, A. Görlich, J. Jurkiewicz, and R. Loll. Nonperturbative quantum de Sitter universe. *Phys. Rev.*, D78:063544, 2008, 0807.4481.

[38] P. Hořava. Quantum gravity at a Lifshitz point. *Phys. Rev. D*, 79:084008, Apr 2009, 0901.3775.

[39] S. Weinfurtner, T.P. Sotiriou, and M. Visser. Projectable Hořava-Lifshitz gravity in a nutshell. *J.Phys.Conf.Ser.*, 222:012054, 2010, 1002.0308.

[40] T.P. Sotiriou. Hořava-Lifshitz gravity: a status report. *J.Phys.Conf.Ser.*, 283:012034, 2011, 1010.3218.

[41] P. Hořava. Spectral Dimension of the Universe in Quantum Gravity at a Lifshitz Point. *Phys.Rev.Lett.*, 102:161301, 2009, 0902.3657.

[42] C. Anderson, S. Carlip, J.H. Cooperman, P. Hořava, R.K. Kommu, and P.R. Zulkowski. Quantizing Hořava-Lifshitz gravity via causal dynamical triangulations. *Phys.Rev.*, D85:044027, 2012, 1111.6634.

[43] S. Bilke, Z. Burda, A. Krzywicki, B. Petersson, J. Tabaczek, and G. Thorleifsson. 4d simplicial quantum gravity: matter fields and the corresponding effective action. *Physics Letters B*, 432(34):279 – 286, 1998, hep-lat/9804011.

[44] J. Laiho and D. Coumbe. Evidence for Asymptotic Safety from Lattice Quantum Gravity. *Phys.Rev.Lett.*, 107:161301, 2011, 1104.5505.

[45] J. Ambjørn, J. Jurkiewicz, and R. Loll. Nonperturbative 3D Lorentzian quantum gravity. *Phys.Rev.*, D64:044011, 2001, hep-th/0011276.

[46] J. Ambjørn, J. Jurkiewicz, and R. Loll. Quantum gravity as sum over spacetimes. *Lect.Notes Phys.*, 807:59–124, 2010, 0906.3947.

[47] J. Ambjørn, A. Görlich, J. Jurkiewicz, and R. Loll. CDT - an entropic theory of quantum gravity. In *Springer Lecture Notes in Physics*, pages 1–49. Springer, 2011, 1007.2560.

[48] J. Ambjørn, A. Görlich, J. Jurkiewicz, and R. Loll. Nonperturbative quantum gravity. *Physics Reports*, 519(45):127 – 210, 2012, 1203.3591.

[49] D. Benedetti, R. Loll, and F. Zamponi. (2+1)-dimensional quantum gravity as the continuum limit of causal dynamical triangulations. *Phys.Rev.*, D76:104022, 2007, 0704.3214.

[50] J. Ambjørn, J. Jurkiewicz, and R. Loll. Emergence of a 4D World from Causal Quantum Gravity. *Phys.Rev.Lett.*, 93:131301, 2004, hep-th/0404156.

[51] B.A. Berg. Introduction to multicanonical Monte Carlo simulations. *Fields Inst. Commun.*, 26:1, 2000, cond-mat/9909236.

[52] Y. Okamoto and U.H.E. Hansmann. Thermodynamics of helix-coil transitions studied by multicanonical algorithms. *J.Phys.Chem.*, 99:11276–11287, 1995, chem-ph/9505006.

[53] J. Ambjørn and J. Jurkiewicz. Scaling in four-dimensional quantum gravity. *Nucl.Phys.*, B451:643–676, 1995, hep-th/9503006.

[54] H. Meyer-Ortmanns. Phase transitions in quantum chromodynamics. *Rev.Mod.Phys.*, 68:473–598, 1996, hep-lat/9608098.

[55] M.E.J. Newman and G.T. Barkema. *Monte Carlo Methods in Statistical Physics*. Oxford University Press, 2002 edition, 1999.

[56] J. Ambjørn, S. Jordan, J. Jurkiewicz, and R. Loll. Second- and first-order phase transitions in causal dynamical triangulations. *Phys.Rev.*, D85:124044, 2012, 1205.1229.

[57] A.M. Ferrenberg and R.H. Swendsen. New Monte Carlo technique for studying phase transitions. *Phys. Rev. Lett.*, 61:2635–2638, 1988.

[58] J. Ambjørn, J. Jurkiewicz, and Y. Watabiki. On the fractal structure of two-dimensional quantum gravity. *Nucl.Phys.*, B454:313–342, 1995, hep-lat/9507014.

[59] J. Ambjørn and K.N. Anagnostopoulos. Quantum geometry of 2D gravity coupled to unitary matter. *Nucl.Phys.*, B497:445–478, 1997, hep-lat/9701006.

[60] R.P. Woodard. Avoiding dark energy with 1/r modifications of gravity. *Lect.Notes Phys.*, 720:403–433, 2007, astro-ph/0601672.

[61] J. Ambjørn, J. Jurkiewicz, and R. Loll. Spectral dimension of the universe. *Phys.Rev.Lett.*, 95:171301, 2005, hep-th/0505113.

[62] M. Reuter and F. Saueressig. Fractal space-times under the microscope: A renormalization group view on Monte Carlo data. *JHEP*, 1112:012, 2011, 1110.5224.

[63] T.P. Sotiriou, M. Visser, and S. Weinfurtner. Phenomenologically viable Lorentz-violating quantum gravity. *Phys.Rev.Lett.*, 102:251601, 2009, 0904.4464.

[64] C. Charmousis, G. Niz, A. Padilla, and P.M. Saffin. Strong coupling in Hořava gravity. *JHEP*, 0908:070, 2009, 0905.2579.

[65] A. Papazoglou and T.P. Sotiriou. Strong coupling in extended Hořava-Lifshitz gravity. *Phys.Lett.*, B685:197–200, 2010, 0911.1299.

[66] K. Koyama and F. Arroja. Pathological behaviour of the scalar graviton in Hořava-Lifshitz gravity. *JHEP*, 1003:061, 2010, 0910.1998.

[67] D. Blas, O. Pujolas, and S. Sibiryakov. Models of non-relativistic quantum gravity: The Good, the bad and the healthy. *JHEP*, 1104:018, 2011, 1007.3503.

[68] J. Ambjørn, J. Jurkiewicz, and R. Loll. A Nonperturbative Lorentzian path integral for gravity. *Phys.Rev.Lett.*, 85:924–927, 2000, hep-th/0002050.

[69] O. Lauscher and M. Reuter. Fractal spacetime structure in asymptotically safe gravity. *JHEP*, 0510:050, 2005, hep-th/0508202.

[70] J. Ambjørn, A. Görlich, J. Jurkiewicz, and R. Loll. Planckian Birth of the Quantum de Sitter Universe. *Phys.Rev.Lett.*, 100:091304, 2008, 0712.2485.

[71] R.M. Hornreich, M. Luban, and S. Shtrikman. Critical behavior at the onset of $\vec{k}$-space instability on the λ line. *Phys. Rev. Lett.*, 35:1678–1681, Dec 1975.

[72] J. Ambjørn, J. Jurkiewicz, and R. Loll. Semiclassical universe from first principles. *Phys.Lett.*, B607:205–213, 2005, hep-th/0411152.

[73] N. Goldenfeld. *Lectures on phase transitions and the renormalization group*. Addison-Wesley, 1992.

[74] A. Michelson. Phase diagrams near the Lifshitz point. I. Uniaxial magnetization. *Phys. Rev. B*, 16:577–584, Jul 1977.

[75] J. Ambjørn, A. Görlich, J. Jurkiewicz, and R. Loll. Geometry of the quantum universe. *Phys.Lett.*, B690:420–426, 2010, 1001.4581.

[76] E. Kiritsis and G. Kofinas. Hořava-Lifshitz cosmology. *Nucl.Phys.*, B821:467–480, 2009, 0904.1334.

[77] R. Brandenberger. Matter Bounce in Hořava-Lifshitz Cosmology. *Phys.Rev.*, D80:043516, 2009, 0904.2835.

[78] G. Calcagni. Cosmology of the Lifshitz universe. *JHEP*, 0909:112, 2009, 0904.0829.

[79] P.O. Mazur and E. Mottola. The path integral measure, conformal factor problem and stability of the ground state of quantum gravity. *Nuclear Physics B*, 341(1):187 – 212, 1990.

[80] A. Dasgupta and R. Loll. A proper-time cure for the conformal sickness in quantum gravity. *Nucl.Phys.*, B606:357–379, 2001, hep-th/0103186.

[81] J. Smit. Remarks on the quantum gravity interpretation of 4D dynamical triangulation. *Nucl.Phys.Proc.Suppl.*, 53:786–790, 1997, hep-lat/9608082.

[82] E. Minguzzi and M. Sánchez. *The causal hierarchy of spacetimes*, page 299–358. ESI Lect. Math. Phys. Eur. Math. Soc., Zürich, 2008.

[83] D. Giulini. Some remarks on the notions of general covariance and background independence. *Lect.Notes Phys.*, 721:105–120, 2007, gr-qc/0603087.

[84] J.L. Anderson. *Principles of Relativity Physics*. Academic Press, New York, 1967.

[85] D. Benedetti and J. Henson. Spectral geometry as a probe of quantum spacetime. *Phys.Rev.*, D80:124036, 2009, 0911.0401.

[86] F. Markopoulou and L. Smolin. Gauge fixing in causal dynamical triangulations. *Nucl.Phys.*, B739:120–130, 2006, hep-th/0409057.

[87] T. Konopka. Foliations and 2+1 causal dynamical triangulation models. *Phys.Rev.*, D73:024023, 2006, hep-th/0505004.

[88] R. Sorkin. Time-evolution problem in regge calculus. *Phys.Rev.*, D12:385–396, 1975.

[89] J. Ambjørn, A. Görlich, J. Jurkiewicz, R. Loll, J. Gizbert-Studnicki, and T. Trzesniewski. The semiclassical limit of causal dynamical triangulations. *Nucl.Phys.*, B849:144–165, 2011, 1102.3929.

[90] J. Laiho. Private communication with R. Loll.

[91] R. Kommu. A validation of causal dynamical triangulations. *Class.Quant.Grav.*, 29:105003, 2012, 1110.6875.

[92] M.N. Rosenbluth and A.W. Rosenbluth. Monte Carlo calculation of the average extension of molecular chains. *J.Chem.Phys.*, 23:356–359, 1955.

[93] U. Pachner. P.L. homeomorphic manifolds are equivalent by elementary shellings. *Eur. J. Comb.*, 12(2):129–145, February 1991.

[94] M. Gross and S. Varsted. Elementary moves and ergodicity in d-dimensional simplicial quantum gravity. *Nuclear Physics B*, 378(12):367 – 380, 1992.

[95] G. Amelino-Camelia. Introduction to quantum-gravity phenomenology. *Lect.Notes Phys.*, 669:59–100, 2005, gr-qc/0412136.

[96] S. Hossenfelder and L. Smolin. Phenomenological quantum gravity. *Phys.Canada*, 66:99–102, 2010, 0911.2761.

Publications

CDT meets Hořava-Lifshitz gravity (with J. Ambjørn, A. Görlich, J, Jurkiewicz, R. Loll), Physics Letters B690 (2010) 413-419, arXiv:1002.3298 [hep-th]

Second-Order Phase Transition in Causal Dynamical Triangulations (with J. Ambjørn, J, Jurkiewicz, R. Loll), Physical Review Letters 107 (2011) 211303, arXiv:1008.3932 [hep-th]

Second- and first-order phase transitions in causal dynamical triangulations (with J. Ambjørn, J, Jurkiewicz, R. Loll), Physical Review D85 (2012) 124044, arXiv:1205.1229 [hep-th]

Causal Dynamical Triangulations without preferred foliation (with R. Loll), Physics Letters B, 2013, in press, arXiv:1305.4582 [hep-th]

De Sitter universe from CDT without preferred foliation (with R. Loll), to appear

Summary

The beginning of the twentieth century brought a paradigm shift in two major areas in physics. On the one hand it was discovered that physics on very small length scales is governed by laws which were completely new and unanticipated at that time. The famous double-slit experiment revealed that physical objects sometimes behave like a particle and sometimes like a wave, exhibiting properties like superposition and interference. More disturbing was the discovery that the outcome of individual experiments could no longer be predicted. These newly found laws of *quantum physics* only allowed to make predictions of probabilistic nature.

The second revolution changed our understanding of physics at very large length scales. This is the domain ruled by the laws of gravitation which since Newton were understood to be force laws, similar to the laws which govern the physics of classical electrostatics. When Einstein published the theory of special relativity in 1905 he quickly realized that the upper limit for velocities given by the speed of light was in contradiction with the Newtonian laws of gravity. About ten years later he presented the theory of *general relativity* based on the revolutionary new idea that spacetime is curved due to the presence of masses.

Both quantum physics and general relativity have developed into mature and well-tested theories over time. The former has evolved into a set of quantum field theories which describe physics at small length scales and at high velocities where the laws of special relativity become important. They form the backbone of the Standard Model of particle physics which describes three of the four fundamental forces in nature, namely, the electromagnetic, the weak and the strong nuclear forces. The Standard Model has been verified in countless experiments with high precision, including the recent discovery of the Higgs boson.

The first experimental verification of general relativity happened in 1919 during a solar eclipse by observing the bending of light as predicted by the theory. Since then a large number of tests have been performed and none of them has shown to be incompatible with general relativity. The theory also predicts the presence of gravitational waves for which so far only indirect evidence exists, based on the study of binary pulsars. The sen-

sitivity of gravitational wave detectors continues to increase and there is confidence that a direct detection of gravitational waves will take place in the near future.

The gravitational force is the weakest of the four fundamental forces and its dominance at large scales is a consequence of its long-range character and its unipolar nature as compared to the bipolar electromagnetic force. The relative weakness of gravity has the consequence that it does not contribute to scattering experiments which take place at microscopic scales in the quantum regime. The length scale at which the strength of the gravitational force becomes comparable to the other forces is governed by the Planck length $\ell_P = 1.62 \times 10^{-35} m$ which is currently completely inaccessible by experiments.

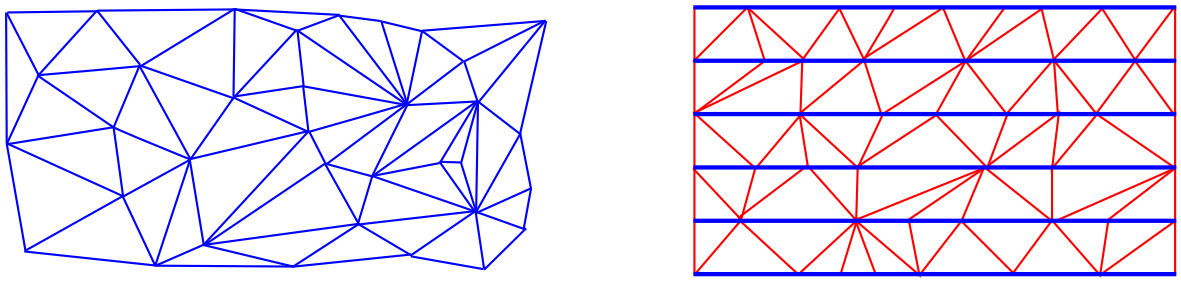
Although the combination of gravity with quantum physics is not obviously needed to explain current experiments, it is nevertheless desirable to understand the physics, which involves both ingredients, in terms of a theory of *quantum gravity*. Several proposals for such a theory have been developed over time, such as string theory, which is based on the idea that the fundamental building blocks of nature are one-dimensional objects and that their vibrations generate the elementary particles of the Standard Model.

In this thesis we have focused on *Causal Dynamical Triangulations* (CDT) which is another proposal for a theory of quantum gravity. In this approach one considers discretized spacetimes which are composed of elementary flat building blocks. The way these objects are glued together determines how curved the spacetime becomes. In this way the basic idea of general relativity is implemented. In practice one considers a large amount of different discretized spacetimes and attempts to superimpose them onto each other, which implements the fundamental idea behind quantum physics. CDT thus performs a *superposition of curved spacetimes*. In chapter 2 we have presented a review of CDT based on standard literature.

The discretization in CDT is in fact only a technical tool which is necessary to actually perform measurements. An important question is whether the lattice can at least in principle be refined until it entirely disappears. If this is possible then we say that the theory has a continuum limit. To find out whether such a limit exists in CDT one has to analyze its phase transitions. In 3+1 dimensional CDT there are three phases of geometry, one of them being physically interesting because it contains four-dimensional geometries in contrast to the geometries in the other phases, where the dimension is either much smaller or much larger than four.

There are two phase transitions which separate the physically interesting phase from a neighboring phase and we call them the A-C transition and the B-C transition. In chapter 3 we have analyzed both phase transitions using three different methods. It turns out that the A-C transition is a first-order transition while the B-C transition is second order. The latter result is important because the presence of a second order transition is a necessary condition for a continuum limit to exist. The theory which preceded CDT, called Euclidean Dynamical Triangulations, did not have such a transition and was therefore abandoned.

In chapter 4 we have explored the connections between CDT and Hořava-Lifshitz gravity (HLG). The latter represents a new theory of gravity, where the fundamental symmetry between space and time is given up in order to improve the high-energy be-



havior of the corresponding quantum theory. CDT and HLG both share the presence of a foliation and in both cases the spectral dimension shows the same qualitative behavior. We show in chapter 4 that the phase diagrams of CDT and HLG show remarkable similarities and we argue that a continuum limit of CDT may be "Hořava-Lifshitz-like".

In the illustration we see two different types of twodimensional triangulations. The left illustration shows an example of a Euclidean triangulation where there is no distinction between space and time. On the right side we see a CDT triangulation where time flows along the red lines. We see that space, represented by the blue lines, forms a layered structure, which we call a foliation. A natural question is whether there is a model which fits in between, which is causal, but not foliated. The answer is yes and the chapters 5-7 are devoted to studying a new model called *locally causal dynamical triangulations*.

In chapter 5 we have presented the theoretical background behind the new model in 2+1 dimensions. In order to analyze the dynamical content of the theory we have developed a new simulation software. Chapter 6 contains an extensive discussion of the results extracted from running simulations. It turns out that the model of locally causal DT contains a phase with geometries which are three-dimensional, just like CDT in 2+1 dimensions. The results also indicate that the locally causal model in 3+1 dimensions (for which the simulation software has not yet been developed) may be compatible with cosmological observations just like CDT in 3+1 dimensions. Finally, in chapter 7 we have provided extensive documentation of algorithms which are needed to actually implement a simulation software for locally causal DT in 2+1 dimensions.

Samenvatting

Het begin van de twintigste eeuw bracht fundamentele veranderingen met zich mee voor twee grote gebieden van de fysica. Aan de ene kant ontdekte men dat de fysica op zeer kleine lengteschalen beschreven wordt door wetten die compleet nieuw en onverwacht waren in die tijd. Het beroemde tweespletenexperiment onthulde dat fysische objecten zich soms gedragen als een deeltje en soms als een golf, met eigenschappen zoals superpositie en interferentie. Nog verrassender was de ontdekking dat het resultaat van individuele experimenten niet langer voorspeld konden worden. Deze nieuwe wetten van de *kwantummechanica* staan alleen maar statistische uitspraken toe.

De tweede revolutie veranderde ons begrip van de fysica op hele grote lengteschalen. Dit is het domein dat gedomineerd wordt door de zwaartekracht die sinds Newton beschreven werd aan de hand krachtwetten, zoals in klassieke elektrostatica. Toen Einstein in 1905 zijn speciale relativiteitstheorie publiceerde, realiseerde hij zich dat de bovenlimiet op snelheden, gegeven door de lichtsnelheid, in tegenspraak zijn met de Newtoniaanse zwaartekrachtswet. Ongeveer tien jaar later presenteerde hij de *algemene relativiteitstheorie*, gebaseerd op het revolutionaire idee dat ruimtetijd gekromd wordt door de aanwezigheid van materie.

Zowel kwantummechanica als de algemene relativiteitstheorie hebben zich ontwikkeld tot volwassen en goed geteste theorieën. Het eerstgenoemde ontwikkelde zich in een verzameling van kwantumveldentheorieën, die de fysica op zeer kleine lengteschalen beschrijft en op hoge snelheden, waar de speciale relativiteitstheorie belangrijk wordt. Deze theorieën vormen de ruggengraat van het Standaard Model van de deeltjesfysica, die drie van de vier fundamentele natuurkrachten beschrijft, namelijk elektromagnetisme, de zwakke interactie en de sterke interactie. Het Standaard Model is geverifieerd door talloze experimenten met hoge precisie, inclusief de recente ontdekking van het Higgs-boson.

De eerste experimentele verificatie van de algemene relativiteitstheorie vond plaats in 1919 gedurende een zonsverduistering door het voorspelde afbuigen van licht te observeren. Sinds die tijd zijn er een groot aantal experimenten uitgevoerd, waarvan er geen enkele de algemene relativiteitstheorie tegenspreekt. De theorie voorspelt ook het be-

staan van gravitationele golven, waarvoor op dit moment alleen indirect bewijs bestaat, gebaseerd op de studie van dubbel-pulsars. De gevoeligheid van gravitationelegolvendetectoren blijft toenemen en men heeft het vertrouwen dat een directe meting van gravitationele golven mogelijk zal zijn in de nabije toekomst.

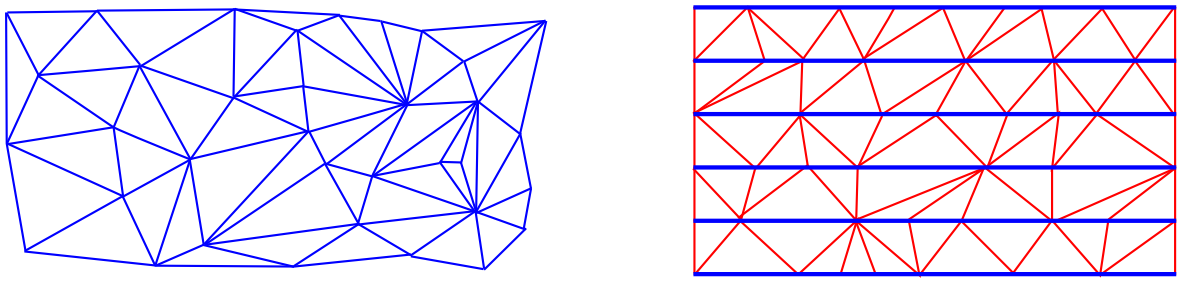
De zwaartekracht is de zwakste van de vier fundamentele krachten en zijn dominantie op grote schalen is een gevolg van het langeafstandskarakter en zijn unipolaire natuur, in tegenstelling tot de bipolaire elektromagnetische kracht. De relatieve zwakte van de zwaartekracht resulteert erin dat zij niet bijdraagt aan verstrooiingsexperimenten die plaatsvinden op microscopische schalen in het kwantumregime. De lengteschalen waarop de zwaartekracht vergelijkbaar wordt met de andere krachten wordt gegeven door de Planck-lengte $\ell_P = 1.62 \times 10^{-35} m$, die op dit moment buiten het bereik van metingen ligt.

Ook al is de combinatie van zwaartekracht met kwantummechanica niet direct nodig voor huidige experimenten, is het desalniettemin wenselijk om de fysica, met beide ingrediënten, te begrijpen in een theorie van *kwantumzwaartekracht*. Verschillende voorstellen voor een dergelijke theorie zijn door de jaren heen ontwikkeld, zoals snaartheorie, dat gebaseerd is op de idee dat de fundamentele bouwstenen van de natuur eendimensionale objecten zijn en dat hun vibraties de elementaire deeltjes generen van het Standaard Model.

In deze thesis focussen we op *Causal Dynamical Triangulations* (CDT), dat een ander voorstel is voor een theorie van kwantumzwaartekracht. In deze aanpak bekijken we discrete ruimtetijden die opgebouwd zijn uit elementaire platte bouwstenen. De manier waarop deze objecten aan elkaar gelijmd worden, bepaalt hoe gekromd de ruimtetijd zal zijn. Op deze manier is het basisprincipe van de algemene relativiteitstheorie ingebouwd. In de praktijk bekijkt men een groot aantal verschillende discrete ruimtetijden die men met behulp van superpositie op elkaar legt, wat zorgt voor de fundamentele idee achter de kwantumfysica. CDT bewerkstelligt dus een *superpositie van gekromde ruimtetijden*. In hoofdstuk 2 presenteren we een overzicht van CDT, gebaseerd op standaardliteratuur.

De discretisatie van CDT is alleen een technisch hulpmiddel dat nodig is om de metingen uit de voeren. Een belangrijke vraag is of het rooster in principe verfijnd kan worden tot het uiteindelijk verdwijnt. Als dit mogelijk is, zeggen we dat de theorie een continuümlimiet heeft. Om uit te vinden of een dergelijke limiet bestaat voor CDT, moet men de faseovergangen analyseren. In 3+1 dimensionale CDT zijn er drie overgangen van geometrie, waarvan er één fysisch interessant is, omdat hij vierdimensionale geometrieën bevat, in plaats van de geometrieën in de andere fasen, waarvan de dimensie ofwel veel groter of veel kleiner is dan vier.

De twee faseovergangen die de fysisch interessante fase onderscheiden van een aanliggende fase noemen we de A-C-overgang en de B-C-overgang. In hoofdstuk 3 hebben we beide faseovergangen geanalyseerd met drie verschillende methoden. Het blijkt dat de A-C-overgang een eerste orde overgang is, terwijl de B-C-overgang van tweede orde is. Dit laatstgenoemde resultaat is belangrijk, omdat de aanwezigheid van een tweede orde faseovergang een eis is voor het bestaan van een continuümlimiet. De voorganger



van CDT, genaamd Euclidian Dynamical Triangulations, had een dergelijke overgang niet en is daardoor niet levensvatbaar.

In hoofdstuk 4 hebben we de connectie tussen CDT en Hořava-Lifshitz gravity (HLG) verkend. Dit laatste vertegenwoordigt een nieuwe zwaartekrachtstheorie, waarin de fundamentele symmetrie tussen ruimte en tijd opgegeven wordt om het hoge-energiegedrag van de corresponderende kwantumtheorie te verbeteren. CDT en HLG delen de aanwezigheid van een gelaagdheid (foliatie) en in beide gevallen geeft de spectrale dimensie hetzelfde kwalitatieve gedrag. In hoofdstuk 4 laten we zien dat de fasediagrammen van CDT en HLG opvallende gelijkenissen vertoont en we geven argumenten waarom de continuümlimiet van CDT misschien "Hořava-Lifshitz-achtig" is.

In de illustratie zien we twee verschillende types van tweedimensionale triangulaties. Het linker plaatje geeft een voorbeeld van een Euclidische triangulatie waar er geen verschil is tussen ruimte en tijd. Aan de rechterkant zien we een CDT-triangulatie waar de tijd loopt langs de rode lijnen. We zien dat ruimte, gerepresenteerd met blauwe lijnen, een gelaagde structuur vormt, die we foliatie noemen. Een natuurlijke vraag is of er een model is wat ertussenin past, dat causaal is maar geen foliatie heeft. Het antwoord is ja en de hoofdstukken 5-7 zijn toegewijd aan de bestudering van dit nieuwe model genaamd *locally causal dynamical triangulations*.

In hoofdstuk 5 presenteren we de theoretische achtergrond achter het nieuwe model in 2+1 dimensies. Om de dynamische inhoud van deze theorie te analyseren, hebben we nieuwe simulatiesoftware geschreven. Hoofdstuk 6 bevat een uitgebreide discussie over de resultaten die uit deze simulaties gehaald zijn. Het blijkt dat het model van lokaal causale DT een fase bevat met geometrieën die driedimensionaal zijn, net zoals CDT in 2+1 dimensies. De resultaten geven ook aan dat het lokaal causale model in 3+1 dimensies (waarvoor de simulatiesoftware nog niet ontwikkeld is) waarschijnlijk in overeenstemming is met kosmologische observaties, net zoals CDT in 3+1 dimensies. Tot slot geven we in hoofdstuk 7 uitgebreide documentatie over de algoritmen de we nodig hadden om de simulatiesoftware voor lokaal causale DT in 2+1 dimensies te implementeren.

Acknowledgments

At the beginning I would like to thank my supervisor, Renate Loll, for giving me the opportunity to work on my PhD project in her group, for guiding me through these four years and for spending a lot of time helping me to achieve a high quality standard in my work. I am especially grateful for the freedom she gave me to explore my own ideas. Another big thanks goes to Jan Ambjørn for all the lively discussions and the fruitful collaboration. His passion for science has inspired and also motivated me to eagerly work on my projects.

In 2009 I had the pleasure of meeting Jerzy Jurkiewicz and his former PhD student Andrzej Görlich. I am grateful for the hospitality during my visit in Cracow and for the help they both gave me at a time when I was just beginning to learn the basics of CDT and its numerical aspects. My other visit in Poland in Zakopane proved equally important, as my first project was born there, set up by Jan and Jerzy.

I spent about three and a half years at the Institute of Theoretical Physics in Utrecht. I was lucky to have my office just next door to Gerard Barkema who often advised me in matters of Monte Carlo simulations and which kindfully agreed to become one of my references for job applications. I am very grateful for all his help and enjoyed all our mutual discussions. I also benefited a lot from discussions with Timothy Budd about both theoretical and numerical aspects of CDT.

For over three years I was gifted with two wonderful office mates, Jasper and Olivier. I highly appreciate our companionship and the nice and calm atmosphere in our office. Thanks to Anne and Juliane for the shared learning sessions and for making it not only a scientific but also a personal event. Thanks to Ben for translating the summary chapter and for many interesting discussions. Thanks to Dražen for the table tennis matches and for the nice talks in between. A special thanks also to Andreas for his companionship not only in Utrecht but also in Nijmegen. Further thanks to Chris, Igor, Paul and Philipp for the nice time that we had together.

Before I moved to the Netherlands I lived in Zurich for more than five years. Some of the friendships made at that time have survived my move to the Netherlands. A big thanks to Colette and Roger for their hospitality during my holiday visits. Furthermore

Acknowledgements

I would like to thank Christian, Katrin, Patrick, Rebecca, Urs Hörler and Urs Mosele for all the enjoyable meetings during my holidays in Switzerland. And finally a big thanks to everyone in my family for all their support and care.

Curriculum Vitae

The author was born on March 6, 1975 in Chur, Switzerland. After the standard primary school education he entered the gymnasium in Chur where he attended a secondary school education with focus on natural sciences. After the successful completion in 1994 he started a study of telecommunications at the University of Applied Sciences in Chur. He focused on aspects of software development and wrote the final thesis in the area of objectoriented databases. In 1998 he concluded his studies by obtaining an engineering degree, which nowadays would be considered equivalent to a bachelor degree. He and a second student received an award for the best final grades.

At the beginning of 1999 he started to work for the Internet provider *deep AG* in Chur as a software engineer. His main project consisted of the development of a new web-based customer and project administration software, including the design and implementation of the underlying database structure. Further projects included the implementation of an application server and the development of an administration tool which allow customers to alter the properties of their email and web hosting accounts.

In 2003 he started to study physics at the University of Zurich while still being part-time employed at the Internet provider. He wrote his bachelor thesis in the area of general relativity and obtained his bachelor degree with a total grade of 5.9 (max: 6.0). In 2008 the author wrote his master thesis in the area of string theory and received his master degree with a total grade of 5.7 (max: 6.0). In 2009 he moved to Utrecht in the Netherlands and started his PhD project under the supervision of Prof. Renate Loll in the area of Causal Dynamical Triangulations. On September 1st in 2012 the whole research group moved to the Radboud University in Nijmegen, where this thesis has been written.