Title: Computational Frameworks for Quantum Gravity and Beyond

Date: Sep 11, 2017 03:30 PM

URL: http://pirsa.org/17090019

Abstract: <p>Causal set quantum gravity, computational methods Series</p>

<p>The Cactus High Performance Computing (HPC) Framework was designed to
facilitate the development of software to simulate astrophysically realistic
inspiral and merger of binary black holes and neutron stars. A major part of the motivation for
its design was the US Binary Black Hole Grand Challenge in the 1990s, which
was an alliance of hundreds of scientists, all attempting to collaborate on
the same code base. In order to handle such a large, diverse collaboration,
the design ended up being extremely general and versatile, able to support
virtually any sort of scientific computation.

I have developed an extensive software suite within the Cactus Framework for
doing large scale computations in discrete quantum gravity, using Causal Sets,
and also spin networks in Loop Quantum Gravity. Recently, evidence is
emerging that ideas from discrete quantum gravity, and consequently this
software, may have much broader application in areas of Data Science,
including social science, computational neuroscience, and drug design.</p>

FOUNDATIONAL QUESTIONS INSTITUTE

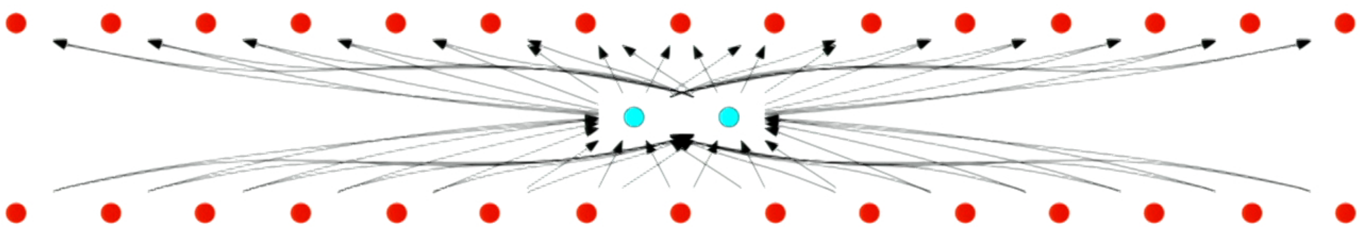
UCSD

Computational Frameworks for Quantum Gravity and Beyond

David Rideout

Department of Mathematics
University of California, San Diego

PI Discussion on Computational Quantum Gravity 11 Sep 2017



Main Points

1. Computers are an important tool for progress in Quantum Gravity
2. High Performance Computing frameworks such as Cactus are a Good Thing for stimulating progress in Quantum Gravity research

Main Points

1. Computers are an ~~important~~ essential tool for progress in Quantum Gravity
2. High Performance Computing frameworks such as **Cactus** are a Good Thing for stimulating progress in Quantum Gravity research
    1. Can build directly from each others' results
    2. Can take advantage of extremely powerful but complicated hardware technologies such as GPUs and Streaming SIMD Extensions (SSE)

Main Points

3. Quantum Gravity is not 'just a tiny subfield of a tiny subfield of an ancient and now mostly irrelevant area of Science'

⇝ Is directly relevant to much current research in Data Science

1. Big Data
2. Computational Neuroscience
3. Medicine & Bioinformatics
4. Emerging quantative approaches to Political and Social Science

Main Points

3. Quantum Gravity is not 'just a tiny subfield of a tiny subfield of an ancient and now mostly irrelevant area of Science'

   ⇝ Is directly relevant to much current research in Data Science

   1. Big Data
   2. Computational Neuroscience
   3. Medicine & Bioinformatics
   4. Emerging quantative approaches to Political and Social Science

Main Points

3. Quantum Gravity is not 'just a tiny subfield of a tiny subfield of an ancient and now mostly irrelevant area of Science'

⇝ Is directly relevant to much current research in Data Science

1. Big Data
2. Computational Neuroscience
3. Medicine & Bioinformatics
4. Emerging quantative approaches to Political and Social Science

Plan of this Talk

1. Clarifications about “Computing”
2. What is a Computational Framework?
3. Quantum Gravity and Spacetime Discreteness
   - Data as Geometry
   - Ordinal Embedding
   - Network Science
4. Results from Computational QG
   - Causal Sets
     - Dynamical Emergence of Continuum
   - Loop Quantum Gravity
5. Conclusions

Some clarification about the term "Computing"

- Want to frame quite general discussion which is nevertheless frank, open, and precise, about computation in relatively new context of quantum gravity — clarifying jargon and avoiding sales pitches.
- In context of quantum gravity, many standard terms aquire ambiguous meanings. I attempt to address some of these issues here:
- *machine* computing (as opposed to computing with pencil and paper etc)

Some clarification about the term "Computing"

- Want to frame quite general discussion which is nevertheless frank, open, and precise, about computation in relatively new context of quantum gravity — clarifying jargon and avoiding sales pitches.
- In context of quantum gravity, many standard terms aquire ambiguous meanings. I attempt to address some of these issues here:
- *machine* computing (as opposed to computing with pencil and paper etc)

Some clarification about the term "Computing"

- Want to frame quite general discussion which is nevertheless frank, open, and precise, about computation in relatively new context of quantum gravity — clarifying jargon and avoiding sales pitches.
- In context of quantum gravity, many standard terms aquire ambiguous meanings. I attempt to address some of these issues here:

- *machine* computing (as opposed to computing with pencil and paper etc)

‘Numerical’ versus ‘Symbolic’ computing

- Aren’t numbers generally represented by symbols?

‘Numerical’ versus ‘Symbolic’ computing

- Aren’t numbers generally represented by symbols?
  ⇝ Not really!
- Usually either:
  1. Use rational approximations
  2. Only explicitly name vanishingly small fraciton of the set of all numbers

'Numerical' versus 'Symbolic' computing

- Aren't numbers generally represented by symbols?
  $\rightsquigarrow$ Not really!
- Usually either:
  1. Use rational approximations
  2. Only explicitly name vanishingly small fraciton of the set of all numbers
- Write symbols to 'simultaneously' represent infinite collection of values

‘Numerical’ versus ‘Symbolic’ computing

- Aren’t numbers generally represented by symbols?
  ⇝ Not really!
- Usually either:
  1. Use rational approximations
  2. Only explicitly name vanishingly small fraciton of the set of all numbers
- Write symbols to ‘simultaneously’ represent infinite collection of values
  ⇝ I think this is what people mean by ‘*symbolic* computing’.
- Here I am primarily interested in the other sort, of ‘*numerical*’ computing.

'Numerical' versus 'Symbolic' computing

- So what is 'numerical' computing?

'Numerical' versus 'Symbolic' computing

- So what is 'numerical' computing?
  $\longrightarrow$ We assign specific values to the 'variables' and compute the implications of those values.

‘Numerical’ versus ‘Symbolic’ computing

- So what is ‘numerical’ computing?
  $\longrightarrow$ We assign specific values to the ‘variables’ and compute the implications of those values.
- Of course in the context of quantum gravity those values may themselves be regarded as symbols
  - a specific poset with five elements
  - a specific rank 3 oriented matroid
  - the name of some molecule

'Numerical' versus 'Symbolic' computing

- So what is 'numerical' computing?
  $\longrightarrow$ We assign specific values to the 'variables' and compute the implications of those values.
- Of course in the context of quantum gravity those values may themselves be regarded as symbols
  - a specific poset with five elements
  - a specific rank 3 oriented matroid
  - the name of some molecule

'Numerical' versus 'Symbolic' computing

- So what is 'numerical' computing?
  $\longrightarrow$ We assign specific values to the 'variables' and compute the implications of those values.
- Of course in the context of quantum gravity those values may themselves be regarded as symbols
  - a specific poset with five elements
  - a specific rank 3 oriented matroid
  - the name of some molecule
- So I have in mind what is traditionally called 'numerical computing', in which the variables take on specific values, even though these values may themselves be symbols.

‘Numerical’ versus ‘Symbolic’ computing

- So what is ‘numerical’ computing?
  $\longrightarrow$ We assign specific values to the ‘variables’ and compute the implications of those values.
- Of course in the context of quantum gravity those values may themselves be regarded as symbols
  - a specific poset with five elements
  - a specific rank 3 oriented matroid
  - the name of some molecule
- So I have in mind what is traditionally called ‘numerical computing’, in which the variables take on specific values, even though these values may themselves be symbols.
- (I am open to suggestions for less ambiguous names for these sorts of computing!)

What I mean by ‘Cactus’

- The Cactus High Performance Computing Framework is primarily a modular API (“application programming interface”).
- Introduces new paradigm in scientific programming:
- Instead of writing ‘standalone’ program in some language,
- one writes modules within the framework,
- in the language of one’s choice (to a degree),
- which then allows them to interoperate with other modules.

What I mean by ‘Cactus’

- The Cactus High Performance Computing Framework is primarily a modular API (“application programming interface”).
- Introduces new paradigm in scientific programming:
- Instead of writing ‘standalone’ program in some language,
- one writes modules within the framework,
- in the language of one’s choice (to a degree),
- which then allows them to interoperate with other modules.

What I mean by ‘Cactus’

- The Cactus High Performance Computing Framework is primarily a modular API (“application programming interface”).
- Introduces new paradigm in scientific programming:
- Instead of writing ‘standalone’ program in some language,
- one writes modules within the framework,
- in the language of one’s choice (to a degree),
- which then allows them to interoperate with other modules. These modules:

What I mean by ‘Cactus’

- The Cactus High Performance Computing Framework is primarily a modular API (“application programming interface”).
- Introduces new paradigm in scientific programming:
- Instead of writing ‘standalone’ program in some language,
- one writes modules within the framework,
- in the language of one’s choice (to a degree),
- which then allows them to interoperate with other modules. These modules:
  - are often written by oneself
  - or others, often experts in different aspects of the problem at hand
  - some experts are extremely adept at eliciting extreme performance from computer hardware

What I mean by 'Cactus'

- The Cactus High Performance Computing Framework is primarily a modular API ("application programming interface").
- Introduces new paradigm in scientific programming:
- Instead of writing 'standalone' program in some language,
- one writes modules within the framework,
- in the language of one's choice (to a degree),
- which then allows them to interoperate with other modules. These modules:
  - are often written by oneself
  - or others, often experts in different aspects of the problem at hand
  - some experts are extremely adept at eliciting extreme performance from computer hardware

What I mean by 'Cactus'

- The Cactus High Performance Computing Framework is primarily a modular API ("application programming interface").
- Introduces new paradigm in scientific programming:
- Instead of writing 'standalone' program in some language,
- one writes modules within the framework,
- in the language of one's choice (to a degree),
- which then allows them to interoperate with other modules. These modules:
  - are often written by oneself
  - or others, often experts in different aspects of the problem at hand
  - some experts are extremely adept at eliciting extreme performance from computer hardware

What I mean by 'Cactus'

- The Cactus High Performance Computing Framework is primarily a modular API ("application programming interface").
- Introduces new paradigm in scientific programming:
- Instead of writing 'standalone' program in some language,
- one writes modules within the framework,
- in the language of one's choice (to a degree),
- which then allows them to interoperate with other modules. These modules:
  - are often written by oneself
  - or others, often experts in different aspects of the problem at hand
  - some experts are extremely adept at eliciting extreme performance from computer hardware
- Beauty is ability to achieve this, all the while maintaining performance as primary design principle

Some thoughts about Software Engineering

- **Performance** versus **ease-of-use**
  - Difficult to have both
  - Interpreted versus compiled languages
  - Interpreters as frameworks
  - 'Playground' versus production code
- Cactus & language independence
- Another reflection on modularity and 'code sharing':

  Have you ever tried to understand and use code you have written 15 years ago?

Some thoughts about Software Engineering

- **Performance** versus **ease-of-use**
  - Difficult to have both
  - Interpreted versus compiled languages
  - Interpreters as frameworks
  - 'Playground' versus production code
- Cactus & language independence
- Another reflection on modularity and 'code sharing':

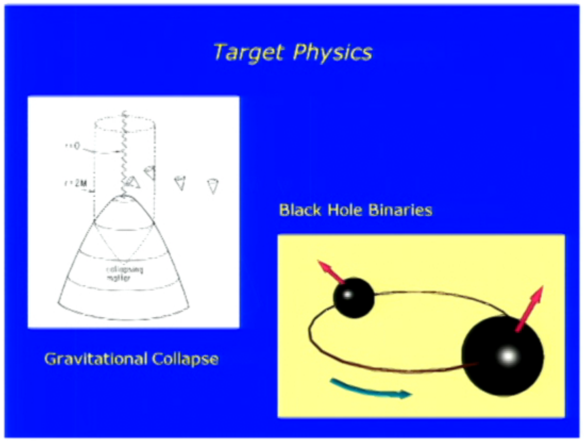
Have you ever tried to understand and use code you have written 15 years ago?

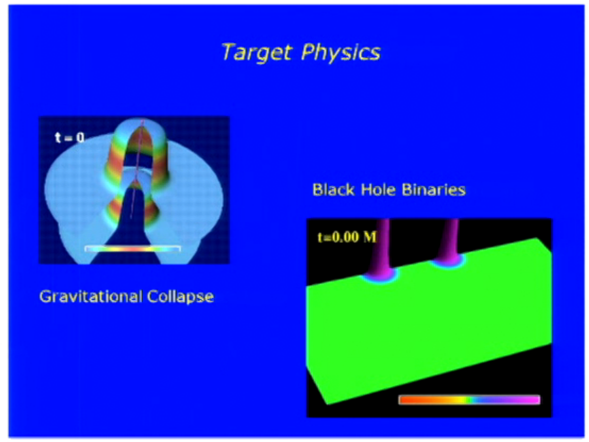
Some thoughts about Software Engineering

- **Performance** versus **ease-of-use**
  - Difficult to have both
  - Interpreted versus compiled languages
  - Interpreters as frameworks
  - ‘Playground’ versus production code
- Cactus & language independence
- Another reflection on modularity and ‘code sharing’:

  Have you ever tried to understand and use code you have written 15 years ago?

The Cactus HPC Framework

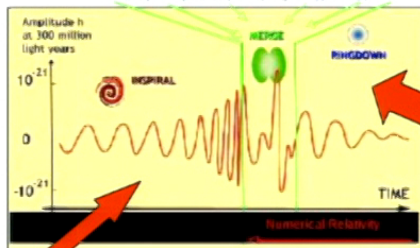




Gravitational Wave Physics

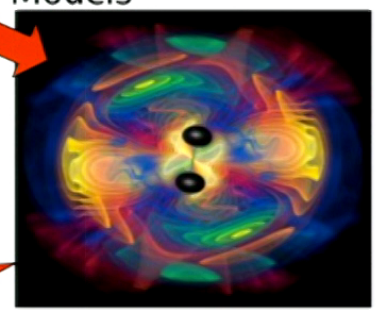
Analysis & Insight

Teraflop Computation, AMR, Elliptic-Hyperbolic, ???



Observations

Models



Einstein Equations: $G_{\mu\nu}(\gamma_{ij}) = 8\pi T_{\mu\nu}$

- Constraint Equations (4 coupled elliptic)
- 12 fully 2nd order evolution equations for γ_{ij}, K_{ij} $(\partial\gamma_{ij}/\partial t)$
- 4 gauge conditions
- GR hydrodynamics for $T_{\mu\nu}$

Collaborative

- Many different sub-parts of complex physics problems, e.g. Numerical Relativity:
  initial data, evolution, horizons, waves, elliptic solvers, AMR, excision, shift conditions
- Need "modular" framework which connects (geographically distributed) experts in each of these areas
- Just one code version !!
- Also:
  - Need not know about the whole code
  - Short startup times for new people
  - Code reuse after people leave

Community: Share common code and experiences, validate results, compare methods.

<u>Example ThornList</u>

ADM_BSSN (Miguel)
FishEye (John B)
Zorro (Manuela/Carlos)
AHFinder (Miguel)
PUGH (Tom)
FlexIO (John S)
IOFlexIO (Thomas R)
BAM_Elliptic (Bernd)
PsiKadelia (Steve)
Time (Gabrielle)
EllBase (Gerd)
Einstein (All)

Portability

- Develop and run on many different architectures
  (laptop, workstations, supercomputers)
- Set up and get going quickly
  (new computer, visits, new job, wherever you get SUs)
- Make immediate use of free resources
- Portability crucial for "Grid Computing"

VARIED RESOURCES AVAILABLE

Origin 2000 (NCSA)
IBM SP4 (NCSA)
Compaq Alpha (PSC)
Linux Cluster (Peyote)
Hitachi SR-8000 (LRZ)
Cray T3E (Garching)
SP4 (ZIB)
Institute Workstations
Linux Laptops
Ed's Mac

Very different architectures, operating systems, compilers and MPI implementations

Large Scale

- Typical BH run needs >100GB of memory:
  - 200 Grid Functions (tensor component functions)
  - 400x400x200 grid
- Typical run makes 3000 iterations with 6000 Flops per grid point:
  - 600 TeraFlops !!
- Output of just one Grid Function at one time step
  - 500 MB (1 TB for 15GF every 50 time steps)
- One computation takes longer than queue times
  - Often need days or weeks or more
- Computing time is a valuable resource
  - One simulation: 5000 to 20000 SUs
  - Need to make each simulation count

Requirements

Parallelism

Optimization

Parallel/Fast IO, Data Management, Visualization

Checkpointing

Interactive monitoring, steering, visualization, portals

Users and Toolkits

- Many numerical relativity groups around the world
  - Over 100 publications
  - Maya, Whisky, Lazarus, ...
- Others:
  - Computational Fluid Dynamics
  - Reservoir simulations (BlackOil)
  - Quantum Gravity
  - Chemical Engineering
  - Crack Propagation
  - Environmental modeling
  - Plasma physics
  - Computer science
  - Astrophysics
  - Cosmology
  - Biology

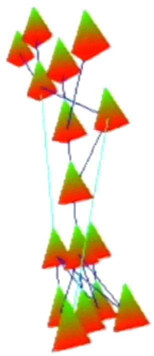
- Toolkits
  - Cactus Computational Toolkit
  - Einstein Toolkit
  - Causal Sets Toolkit
  - CFD Toolkit
  - (Biology Toolkit)
- Teaching
  - Over 30 student thesis/diploms

The Cactus High Performance Computing Framework

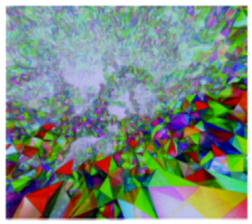
- Modular framework $\rightsquigarrow$ facilitates development of community toolkits
- Language independence $\rightsquigarrow$ easy to use and program
- Testsuites – detects when modifications to code change output!
- Automated benchmarking tools
- Facilitates running on supercomputers :
  - Automatic parallelism
  - Automatic detection of libraries $\longrightarrow$ knows how to build itself on wide variety of architectures
- Abstracts away many complications of large scale computing

Numerous Approaches to QG

Causal Sets

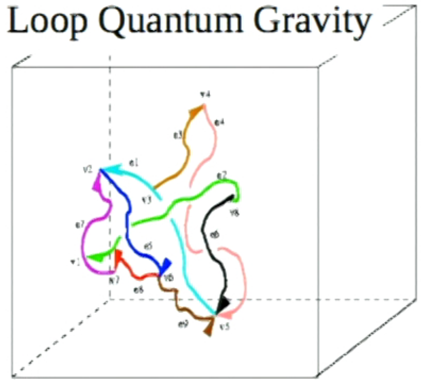


Spin Foams

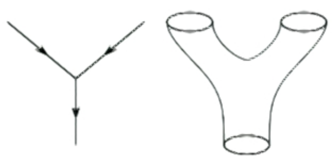


Causal Dynamical Triangulations

Loop Quantum Gravity

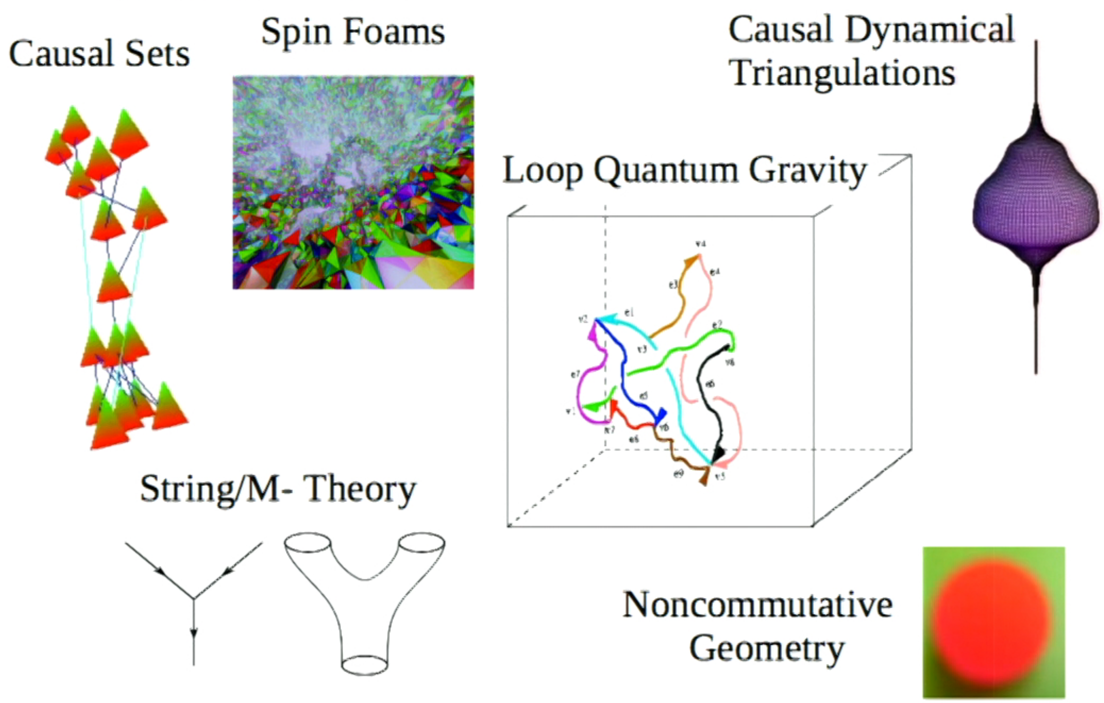


String/M- Theory



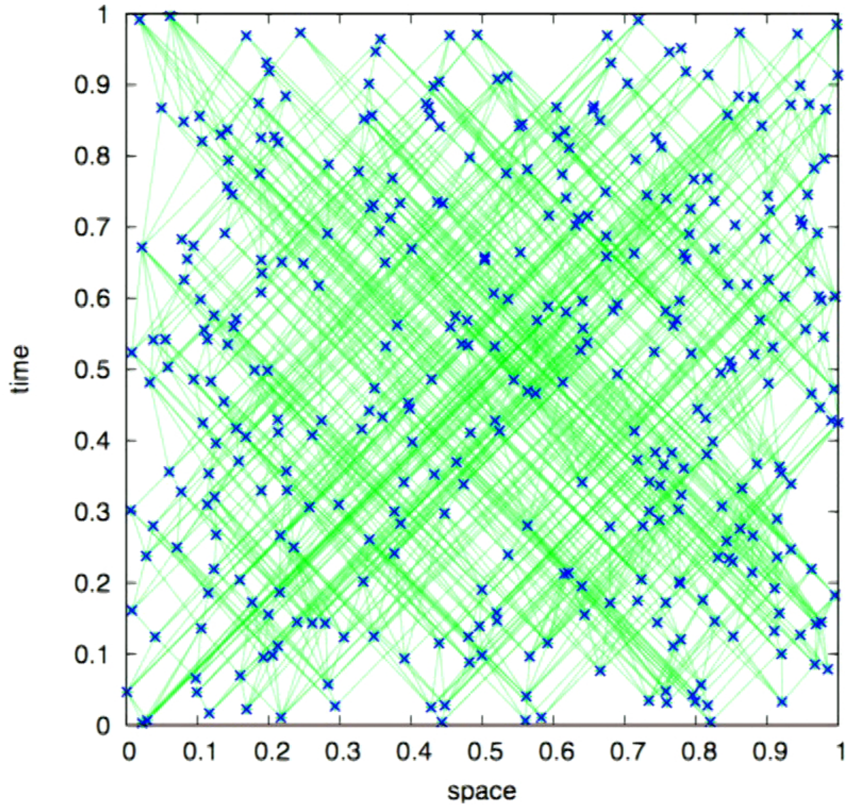
Noncommutative Geometry

Numerous Approaches to QG

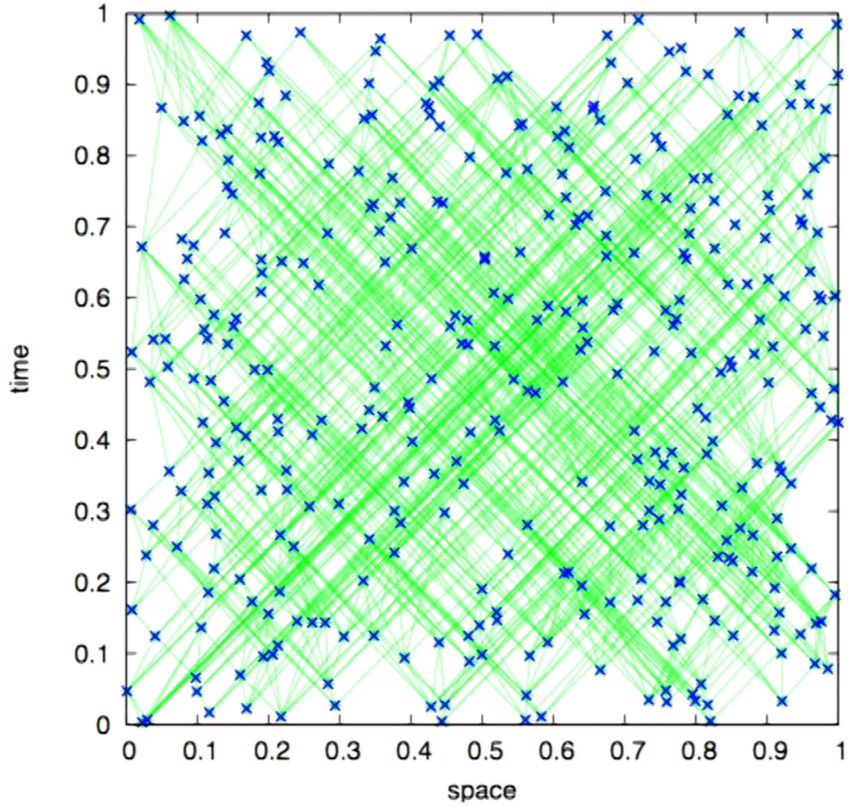


⇝ All predict some form of spacetime discreteness at small length / time scales near the Planck scale

Quantum Gravity and Spacetime Discreteness



Quantum Gravity and Spacetime Discreteness



⇝ Geometry as Data

Data as Geometry

Mobile phone data:

```
1428381900    2063_2_3    4.21812303930583     39    441.0077527068526
1428381900    2063_3_0    4.082151070910959    39    156.6167091635713
1428381900    2064_0_0    4.237465831741913    39    491.3782312381997
1428381900    2064_0_2    5.790193868189464    39    424.217684407336
1428381900    2064_1_1    7.2776012760661 39   758.6005150074806
1428381900    2064_1_3    17.7789638409026     39    2451.383798644420
1428381900    2064_2_0    13.254485197417544   39    1166.864975840631
1428381900    2064_2_2    15.617799462553531   39    1933.535739669483
1428381900    2064_3_1    12.151260867327524   39    783.4986097303604
1428381900    2097_1_1    17.02070525948483    39    1306.805545524564
1428381900    3415_0_3    7.722073037605164    39    700.6958779989764
```

Human protein interaction data:

```
21422   5174    1244    111200  107647  -       RP11-114F7.2    PDZK1   ABCC2
21428   23705   4356    117218  110496  -       -       CADM1   MPP3    BL2|IGSF4|
21434   51678   7064    119675  112921  -       -       MPP6    THOP1   PALS2|VAM-
```

Data as Geometry

Mobile phone data:

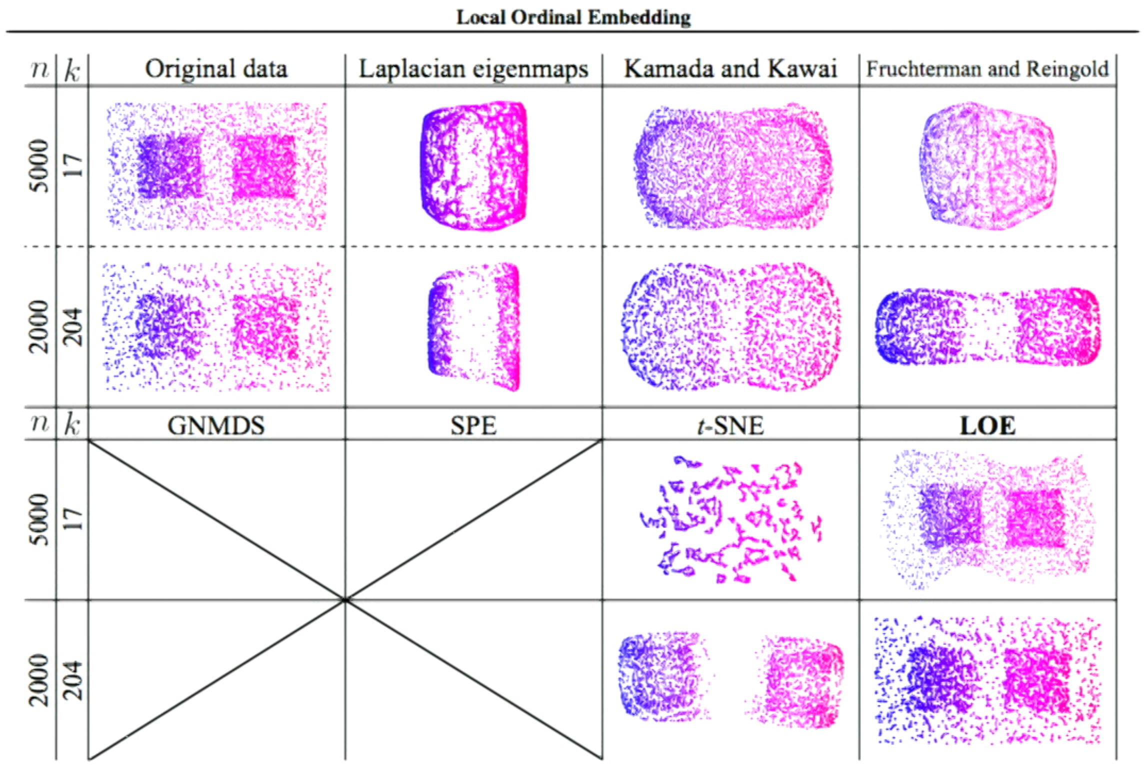
1428381900	2063_2_3	4.21812303930583	39	441.0077527068526
1428381900	2063_3_0	4.082151070910959	39	156.6167091635713
1428381900	2064_0_0	4.237465831741913	39	491.3782312381997
1428381900	2064_0_2	5.790193868189464	39	424.217684407336
1428381900	2064_1_1	7.2776012760661	39	758.6005150074806
1428381900	2064_1_3	17.7789638409026	39	2451.383798644420
1428381900	2064_2_0	13.254485197417544	39	1166.864975840631
1428381900	2064_2_2	15.617799462553531	39	1933.535739669483
1428381900	2064_3_1	12.151260867327524	39	783.4986097303604
1428381900	2097_1_1	17.02070525948483	39	1306.805545524564
1428381900	3415_0_3	7.722073037605164	39	700.6958779989764

Human protein interaction data:

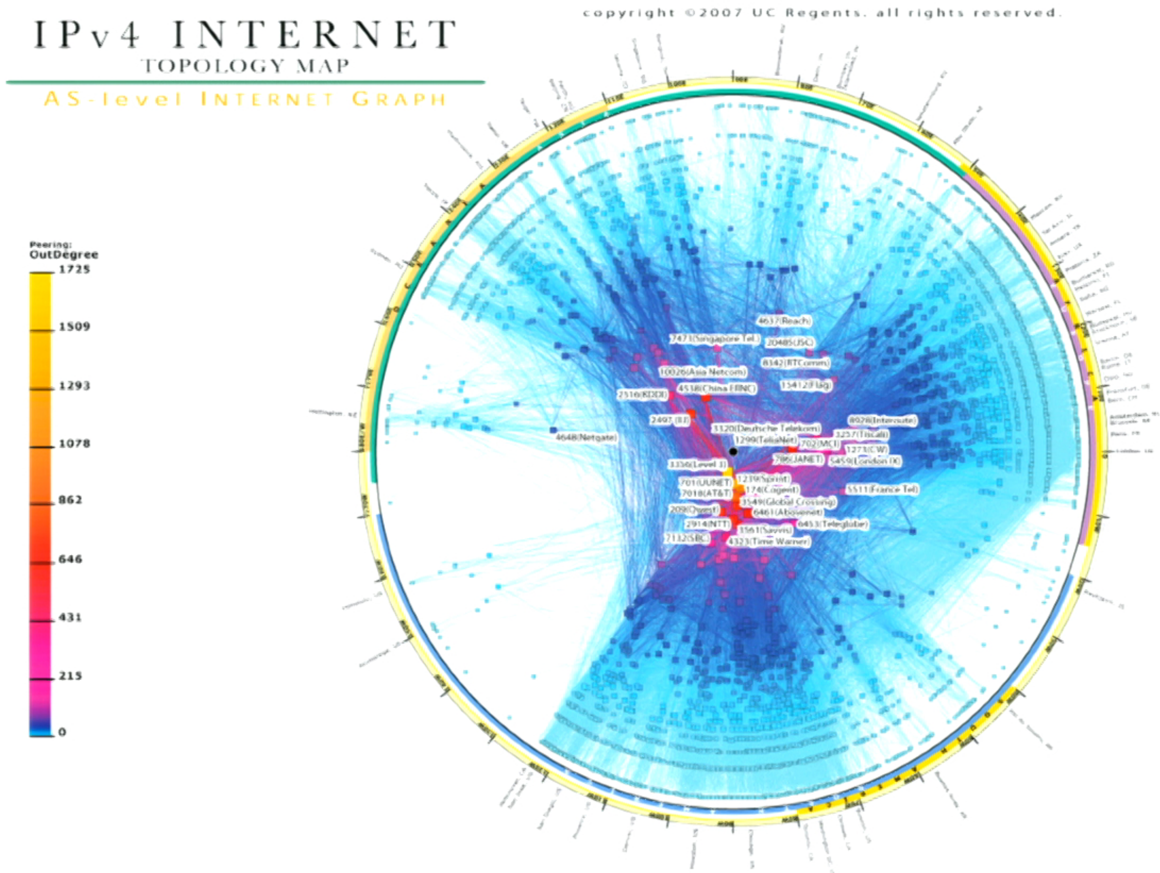
21422	5174	1244	111200	107647	-	RP11-114F7.2		PDZK1	ABCC2
21428	23705	4356	117218	110496	-	-	CADM1	MPP3	BL2|IGSF4|
21434	51678	7064	119675	112921	-	-	MPP6	THOP1	PALS2|VAM-

- Summarize data in some useful way, to draw conclusions and make predictions
- What better summary than a geometric approximation?
  ⇝ View data elements as coordinates of points in some manifold
- **Support Vector Machines** identify geometric properties of boundaries within data

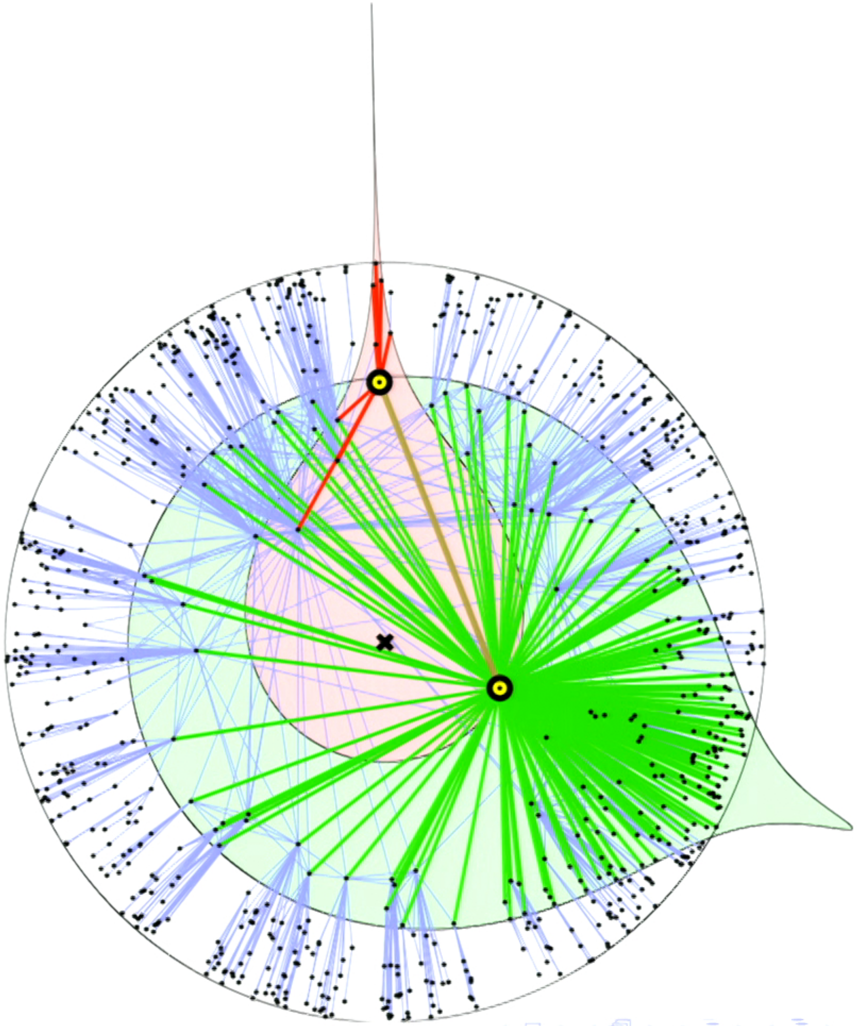
Ordinal Embedding



Network Science



Network Science



Network Science

SCIENTIFIC REPORTS

OPEN

Network Cosmology

Dmitri Krioukov[1], Maksim Kitsak[1], Robert S. Sinkovits[2], David Rideout[3], David Meyer[3] & Marián Boguñá[4]

[1]Cooperative Association for Internet Data Analysis (CAIDA), University of California, San Diego (UCSD), La Jolla, CA 92093, USA, [2]San Diego Supercomputer Center (SDSC), University of California, San Diego (UCSD), La Jolla, CA 92093, USA, [3]Department of Mathematics, University of California, San Diego (UCSD), La Jolla, CA 92093, USA, [4]Departament de Física Fonamental, Universitat de Barcelona, Martí i Franquès 1, 08028 Barcelona, Spain.

SUBJECT AREAS:
STATISTICAL PHYSICS, THERMODYNAMICS AND NONLINEAR DYNAMICS
THEORETICAL PHYSICS
APPLIED PHYSICS
COSMOLOGY

Received
23 July 2012

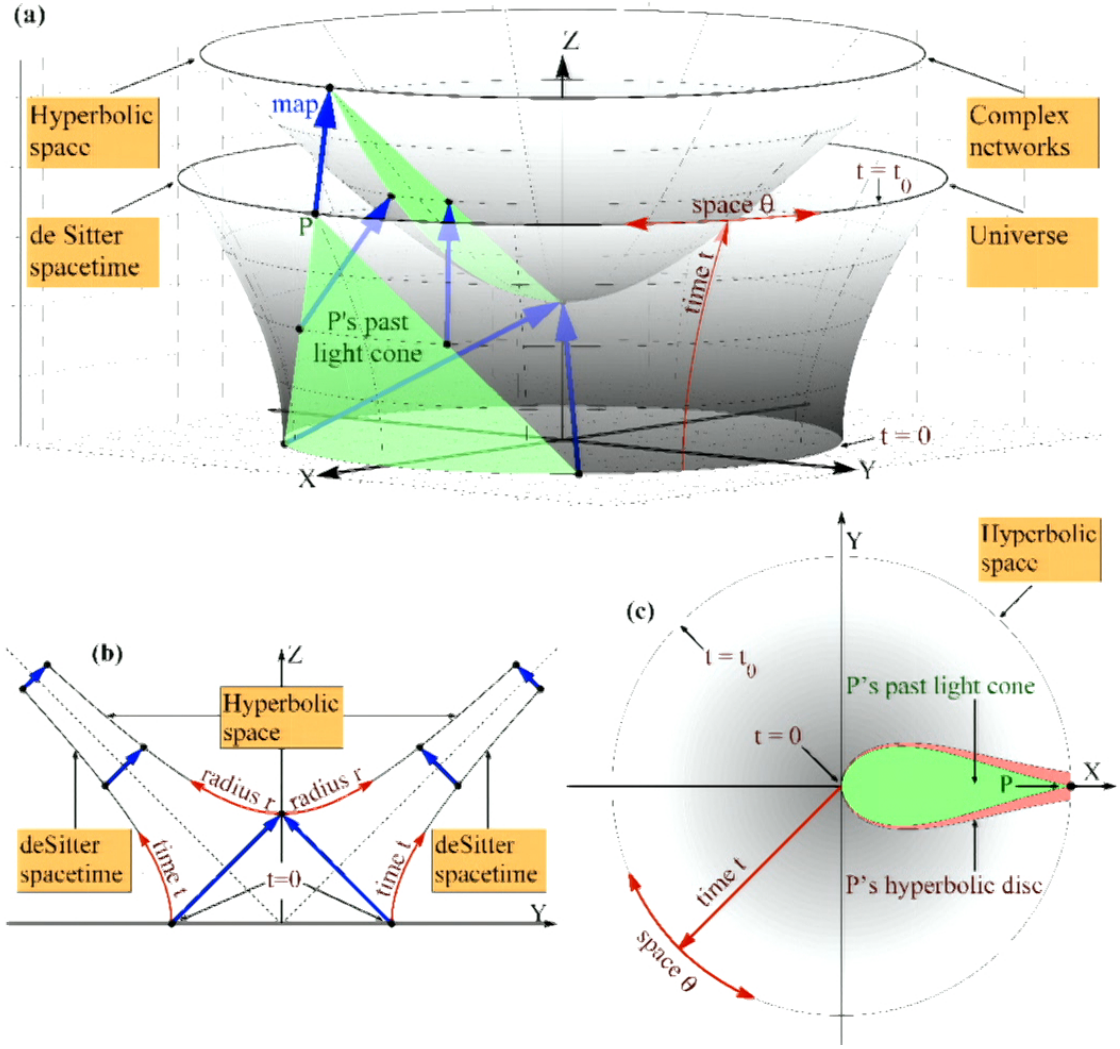
Accepted
3 October 2012

Published
16 November 2012

Prediction and control of the dynamics of complex networks is a central problem in network science. Structural and dynamical similarities of different real networks suggest that some universal laws might accurately describe the dynamics of these networks, albeit the nature and common origin of such laws remain elusive. Here we show that the causal network representing the large-scale structure of spacetime in our accelerating universe is a power-law graph with strong clustering, similar to many complex networks such as the Internet, social, or biological networks. We prove that this structural similarity is a consequence of the asymptotic equivalence between the large-scale growth dynamics of complex networks and causal networks. This equivalence suggests that unexpectedly similar laws govern the dynamics of complex networks and spacetime in the universe, with implications to network science and cosmology.

Physics explains complex phenomena in nature by reducing them to an interplay of simple fundamental laws.

Network Science

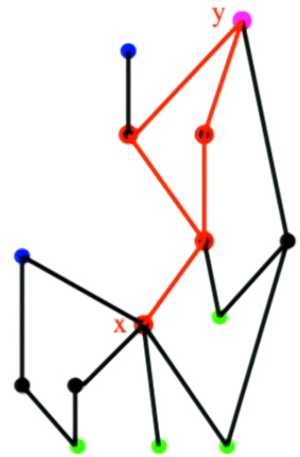


Outline

Causal Sets: Fundamentally Discrete Gravity

Based upon two main observations:

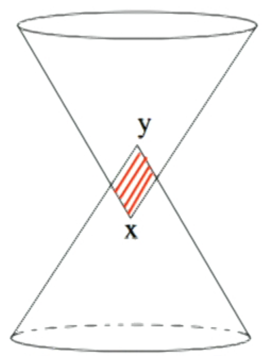
- Richness of causal structure
- Need for discreteness



Properties of discrete causal order $\prec$:

- irreflexive $(x \not\prec x)$
- transitive $(x \prec y \quad \text{and} \quad y \prec z \Rightarrow x \prec z)$
- locally finite $(|\{y | x \prec y \prec z| < \infty)$

Some definitions:



- relation & link
- chain & antichain, height & width
- causal interval or order interval
- maximal & minimal elements

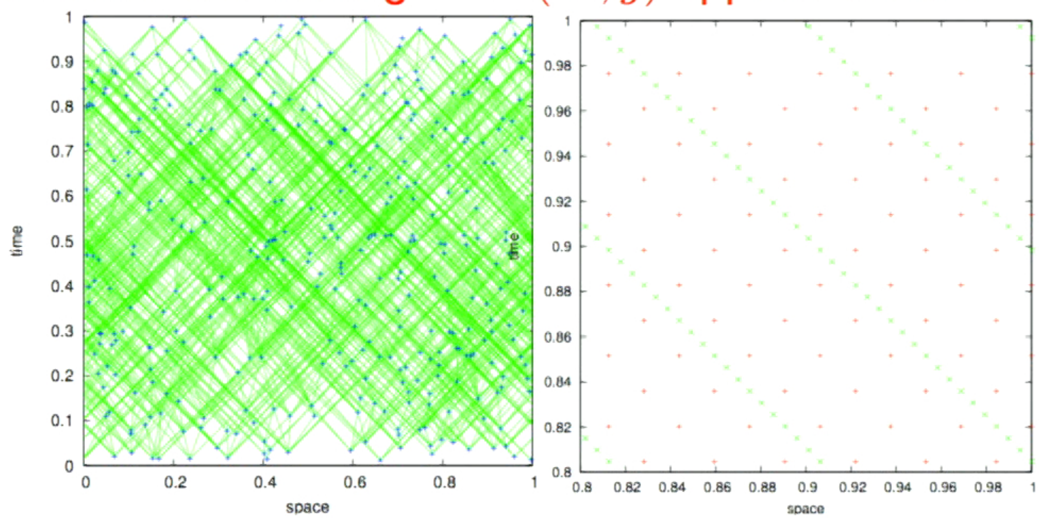
Spacetime Manifold as Emergent Structure

The continuum approximation

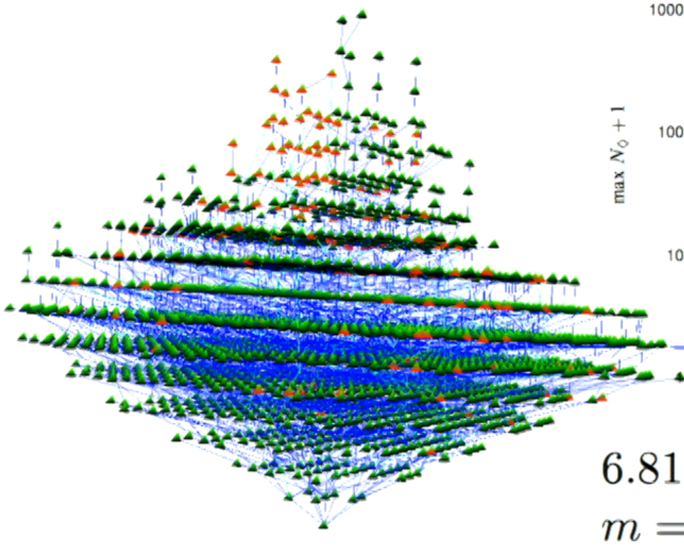
- *Embedding* – order preserving map $\phi : \mathcal{C} \rightarrow (M, g)$

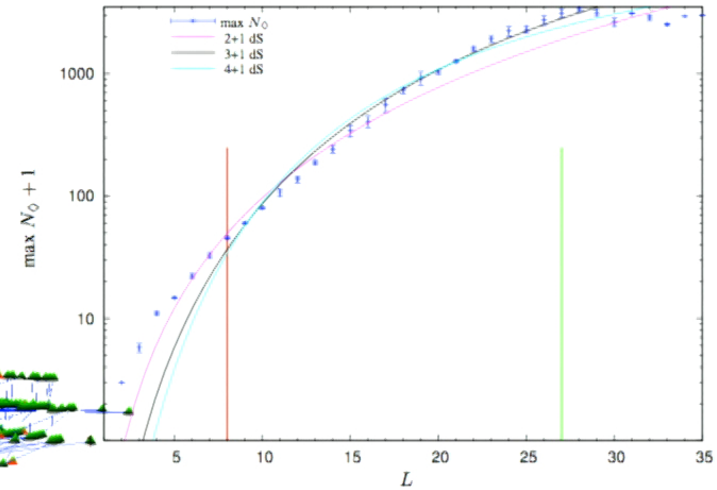
$$x \prec y \Leftrightarrow \phi(x) \prec \phi(y) \;\; \forall x, y \in \mathcal{C}$$

- *Faithful embedding ('Sprinkling')*:
  - "preserves number – volume correspondence"
  - Spacetime does not possess structure at scales smaller than discreteness scale
- $\exists$ faithful embedding $\Longrightarrow$ (M, g) approximates $\mathcal{C}$



Early Universe of Growth Dynamics



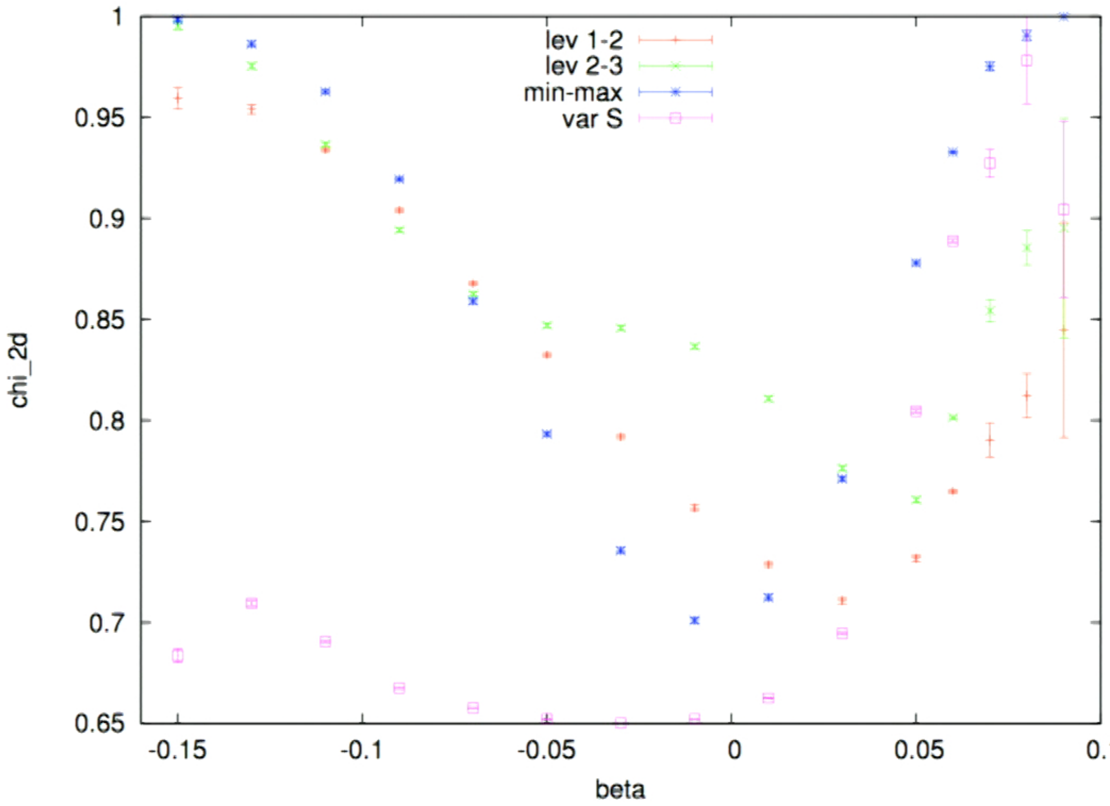


$\ell = 6.81 \pm .72$

$m = 1.926 \pm .023$

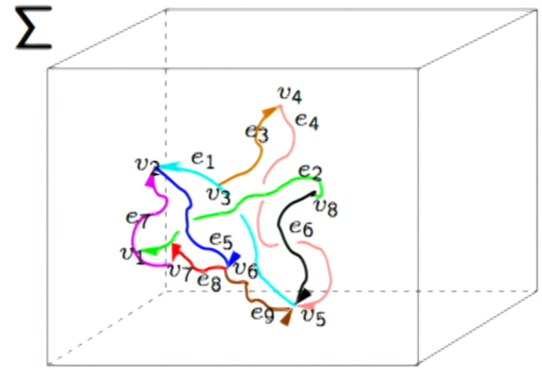
M. Ahmed and DR, *Phys.Rev.D* **81**, 083528 (2010)
arXiv:0909.4771 [gr-qc]

Two Dimensional Behavior for $n = 15$



Loop Quantum Gravity
Kinematical Hilbert Space

Consider



Canonical variables:
holonomies h_{e_I}, fluxes $E_i(S)$

$E(\gamma)\ldots$ edge set of γ
$V(\gamma)\ldots$ vertex set of γ

- Kinematical Hilbert space $\mathcal{H}_{\text{kin},\gamma} = L^2(\bar{\mathcal{A}}_\gamma, d\mu_\gamma)$
- Spin Network Functions

$$T_{\gamma\vec{j}\vec{m}\vec{n}}(A) := \prod_{e\in E(\gamma)} \sqrt{2j_e+1}[\pi_{j_e}(h(A))]_{m_e n_e}$$

- Basis of $\mathcal{H}_{kin,\gamma}$ (Peter & Weyl): $\sqrt{2j+1}\left[\pi_j(h_e)\right]_{m_e n_e} \sim \langle h_e \mid j\ m_e\ ;\ n_e \rangle$
  - Can replace $-\frac{i}{2}E_j \mid j\ m_e\ ;\ n_e \rangle \quad = \quad J_j \mid j\ m_e\ ;\ n_e \rangle$

Action of operators only containing E_j can be expressed as action of usual angular momentum operators acting on a spin system

(Ashtekar-Lewandowski) Volume Operator: Structure

- Classical Volume Expression

$$V(R) = \int_R d^3x \, \sqrt{\det q(x)} = \int_R d^3x \, \sqrt{\left| \frac{1}{3!} \varepsilon^{ijk} \varepsilon_{abc} E_i^a(x) E_j^b(x) E_k^c(x) \right|}$$

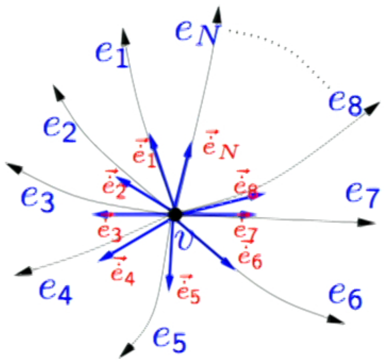
- Structure of the volume operator

$$\hat{V}_\gamma(R) \, f_\gamma \propto \ell_P^3 \sum_{v \in V(\gamma)} \sqrt{\left| \sum_{IJK} \epsilon(I,J,K) \quad \hat{q}_{IJK} \right|} \, f_\gamma$$

graph structure

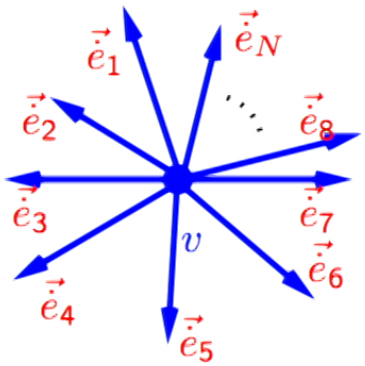
$$\epsilon(I,J,K) = \mathrm{sgn}\big(\det(\dot{e}_I, \dot{e}_J, \dot{e}_K)(v)\big)$$

Graph Structure



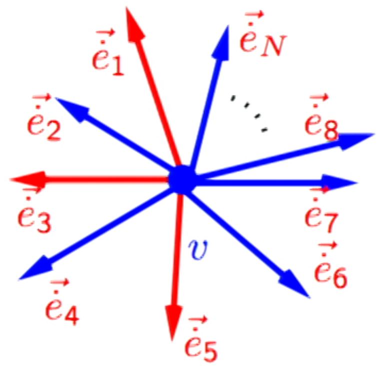
- N edges at vertex v
- with tangent vectors

Graph Structure



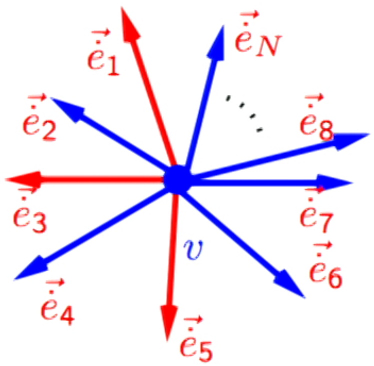
- N edges at vertex v
- with tangent vectors

Graph Structure



- N edges at vertex v
- with tangent vectors
- edge triple e_1, e_3, e_5

Graph Structure



- N edges at vertex v
- with tangent vectors
- edge triple e_1, e_3, e_5

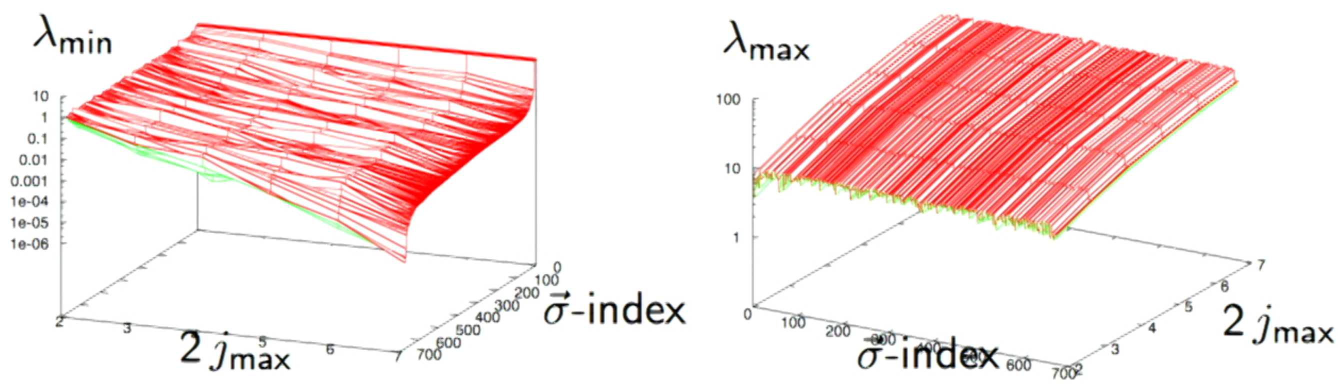
$$\epsilon(1\,3\,5) \;:=\; \mathrm{sgn}\big(\det(\vec{e}_1\,,\,\vec{e}_3\,,\,\vec{e}_5)\big)$$

- General: $\epsilon(L\;M\;N) \;:=\; \mathrm{sgn}\big(\det(\vec{e}_L\,,\,\vec{e}_M\,,\,\vec{e}_N)\big)$
- Assignment of sign (+/-) to every triple of edge tangent vectors

$\rightsquigarrow$ This exact mathematical object is called a *chirotope* in the context of *Oriented Matroid Theory*

What sign combinations, aka chirotopes, will occur at all?

Extremal non-zero eigenvalues at the 7-vertex



Conclusions

1. Computers are arguably an essential tool for progress in Quantum Gravity
2. High Performance Computing frameworks such as **Cactus** are a Good Thing for stimulating progress in Quantum Gravity research
   1. Can build directly from each others' results
   2. Can take advantage of extremely powerful but complicated hardware technologies such as GPUs and Streaming SIMD Extensions (SSE)
3. Quantum Gravity is not 'just a tiny subfield of a tiny subfield of an ancient and now mostly irrelevant area of Science' $\rightsquigarrow$ Is directly relevant to much current research in Data Science

Conclusions

4. Use of the Cactus HPC Framework for Quantum Gravity is leading to important developments
   1. Evidence of continuum-like behavior in the very early universe of sequential growth dynamics for causal sets
   2. Evidence of dimensional reduction in causal sets at small scales, in the full path sum over histories
   3. The ability to explore generic features of LQG which are inaccessible by other methods
      $\rightsquigarrow$ provides understanding of the spectrum of the volume operator at a generic spin network vertex