Computational Exploration of String Theory

Michael R. Douglas

[1] Simons Center / Stony Brook University

AITP 2018 – Aussois, France

Abstract

String theory provides a way to derive the possible laws of physics, and testable predictions, from purely mathematical structures such as complex manifolds and submanifolds, homology groups, group representations, etc. The list of possibilities is finite and in principle could be classified, but the problem is very large.

String theorists have used computational methods to help do this for many years. A pioneering example developed in the early 90's and which is still of central importance is the Kreuzer-Skarke database of reflexive polytopes. Since then many more algorithms and datasets have been developed by string theorists, many of value for pure mathematicians as well.

Our computational tools for working with and managing this information are very primitive. I will suggest a tool – a "formal wiki" – to help string theorists and other mathematical scientists to maintain shared repositories of formally verified mathematical software and data.

Abstract

String theory provides a way to derive the possible laws of physics, and testable predictions, from purely mathematical structures such as complex manifolds and submanifolds, homology groups, group representations, etc. The list of possibilities is finite and in principle could be classified, but the problem is very large.

String theorists have used computational methods to help do this for many years. A pioneering example developed in the early 90's and which is still of central importance is the Kreuzer-Skarke database of reflexive polytopes. Since then many more algorithms and datasets have been developed by string theorists, many of value for pure mathematicians as well.

Our computational tools for working with and managing this information are very primitive. I will suggest a tool – a "formal wiki" – to help string theorists and other mathematical scientists to maintain shared repositories of formally verified mathematical software and data.

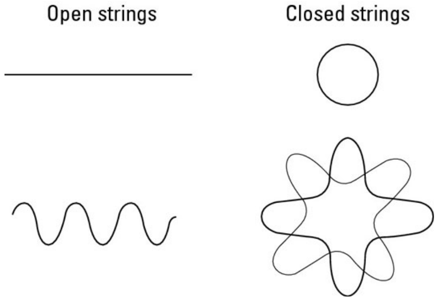
Abstract

String theory provides a way to derive the possible laws of physics, and testable predictions, from purely mathematical structures such as complex manifolds and submanifolds, homology groups, group representations, etc. The list of possibilities is finite and in principle could be classified, but the problem is very large.

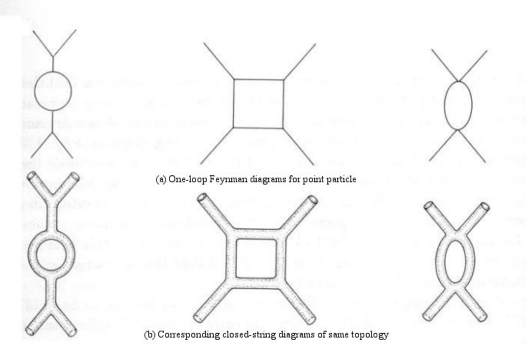
String theorists have used computational methods to help do this for many years. A pioneering example developed in the early 90's and which is still of central importance is the Kreuzer-Skarke database of reflexive polytopes. Since then many more algorithms and datasets have been developed by string theorists, many of value for pure mathematicians as well.

Our computational tools for working with and managing this information are very primitive. I will suggest a tool – a "formal wiki" – to help string theorists and other mathematical scientists to maintain shared repositories of formally verified mathematical software and data.

String theory: particles (electrons, quarks, photons, etc.) are really small loops of string. Different modes of vibration → different particles.
Open string → one direction of vibration → polarization of photon.
Closed strings naturally vibrate in two directions (left and right movers).
Spin two particle → graviton, so string theory naturally contains gravity (general relativity).
In fact string theory is a quantum unified theory of gravity and Yang-Mills theory coupled to matter, all of the fundamental theories which describe known physics.



The unification of general relativity and quantum mechanics has been studied intensively for over 60 years and it is a hard problem. Scaling arguments (theory of renormalizability) tell us that in D space-time dimensions, the strength of gravity grows with decreasing length L as L^{2-D}. In $D > 2$ gravity becomes strong at the Planck scale (10^{-33} cm) meaning that the metric is a strongly fluctuating variable. However the observed space-time metric is almost flat, it is not strongly fluctuating.



In string theory, there is another preferred scale, the string length. Quantum effects are "cut off" at shorter distances, eliminating the strong metric fluctuations so that quantum gravity is consistent with the observed properties of space-time.

String theory cannot be modified – it is a single unified structure (though with many limits which superficially look different). Thus, it is either right or wrong as a candidate fundamental theory.

And, string theory has many other surprising and important properties:

- Maximal symmetry and supersymmetry
- Exceptional structures such as E_8
- Dualities: strong $\leftrightarrow$ weak, gauge $\leftrightarrow$ gravity.

Many of the alternative approaches to quantum gravity turned out to be particular cases of "string/M theory."

String theory has explained some (not yet all) of the mysteries of quantum gravity, such as the entropy of black holes. The study of string theory has also led to breakthroughs on many other physical questions: the origin of quark confinement, symmetry breaking, phase diagrams, etc.

String theory cannot be modified – it is a single unified structure (though with many limits which superficially look different). Thus, it is either right or wrong as a candidate fundamental theory.

And, string theory has many other surprising and important properties:

- Maximal symmetry and supersymmetry
- Exceptional structures such as E_8
- Dualities: strong $\leftrightarrow$ weak, gauge $\leftrightarrow$ gravity.

Many of the alternative approaches to quantum gravity turned out to be particular cases of "string/M theory."

String theory has explained some (not yet all) of the mysteries of quantum gravity, such as the entropy of black holes. The study of string theory has also led to breakthroughs on many other physical questions: the origin of quark confinement, symmetry breaking, phase diagrams, etc.

String theory cannot be modified – it is a single unified structure (though with many limits which superficially look different). Thus, it is either right or wrong as a candidate fundamental theory.

And, string theory has many other surprising and important properties:

- Maximal symmetry and supersymmetry
- Exceptional structures such as E_8
- Dualities: strong $\leftrightarrow$ weak, gauge $\leftrightarrow$ gravity.

Many of the alternative approaches to quantum gravity turned out to be particular cases of “string/M theory.”

String theory has explained some (not yet all) of the mysteries of quantum gravity, such as the entropy of black holes. The study of string theory has also led to breakthroughs on many other physical questions: the origin of quark confinement, symmetry breaking, phase diagrams, etc.

However, it is hard to get additional empirical evidence for or against the claim that string theory is the fundamental theory of our universe.

Strings have higher modes of vibration and producing these would be a strong test. But, the energy required to do this is comparable to the energy scale associated to gravity: the Planck scale, 10^{19} GeV. By comparison, the LHC at CERN produces collisons with energies of $1.3 \cdot 10^4$ GeV. While we cannot directly test the underlying “stringy” nature of matter at these energies, we can hope to discover new particles which would naturally emerge from string theory (or, which would be impossible to describe as strings). The primary examples are the “superpartners” which are predicted by the theory of supersymmetry and fit well with string theory.

These have not yet been discovered and one of the outstanding questions is, does string theory predict an energy scale at which they are likely to be discovered? Some have argued yes, at around 10^5 GeV. Such a collider could be built (probably by 2040), and the prediction tested.

However, it is hard to get additional empirical evidence for or against the claim that string theory is the fundamental theory of our universe.

Strings have higher modes of vibration and producing these would be a strong test. But, the energy required to do this is comparable to the energy scale associated to gravity: the Planck scale, 10^{19} GeV. By comparison, the LHC at CERN produces collisons with energies of $1.3 \cdot 10^4$ GeV. While we cannot directly test the underlying "stringy" nature of matter at these energies, we can hope to discover new particles which would naturally emerge from string theory (or, which would be impossible to describe as strings). The primary examples are the "superpartners" which are predicted by the theory of supersymmetry and fit well with string theory.

These have not yet been discovered and one of the outstanding questions is, does string theory predict an energy scale at which they are likely to be discovered? Some have argued yes, at around 10^5 GeV. Such a collider could be built (probably by 2040), and the prediction tested.

However, it is hard to get additional empirical evidence for or against the claim that string theory is the fundamental theory of our universe.

Strings have higher modes of vibration and producing these would be a strong test. But, the energy required to do this is comparable to the energy scale associated to gravity: the Planck scale, 10^{19} GeV. By comparison, the LHC at CERN produces collisons with energies of $1.3 \cdot 10^4$ GeV. While we cannot directly test the underlying “stringy” nature of matter at these energies, we can hope to discover new particles which would naturally emerge from string theory (or, which would be impossible to describe as strings). The primary examples are the “superpartners” which are predicted by the theory of supersymmetry and fit well with string theory.

These have not yet been discovered and one of the outstanding questions is, does string theory predict an energy scale at which they are likely to be discovered? Some have argued yes, at around 10^5 GeV. Such a collider could be built (probably by 2040), and the prediction tested.

Besides experiments at particle colliders, we can also imagine testing string theory (or alternative proposals for fundamental physics) using astronomical observations, or other physics experiments.

The physics of early cosmology (the "Big Bang") probes higher energies, all the way up to the Planck scale. It leaves observable effects on the cosmic microwave background radiation (CMBR). For example, it has been suggested that "B-modes" of the CMBR would naturally come out of stringy inflation.

There might also be rare processes such as proton decay, or very weakly coupled particles (WIMPs, gravitinos, axions, etc.) which have not yet been seen yet, and which tell us about physics at high energies. It has been suggested that string theory leads to a very large number of weakly coupled light particles ("axions") which might make up the dark matter.

Besides experiments at particle colliders, we can also imagine testing string theory (or alternative proposals for fundamental physics) using astronomical observations, or other physics experiments.

The physics of early cosmology (the “Big Bang”) probes higher energies, all the way up to the Planck scale. It leaves observable effects on the cosmic microwave background radiation (CMBR). For example, it has been suggested that “B-modes” of the CMBR would naturally come out of stringy inflation.

There might also be rare processes such as proton decay, or very weakly coupled particles (WIMPs, gravitinos, axions, etc.) which have not yet been seen yet, and which tell us about physics at high energies. It has been suggested that string theory leads to a very large number of weakly coupled light particles (“axions”) which might make up the dark matter.

Besides experiments at particle colliders, we can also imagine testing string theory (or alternative proposals for fundamental physics) using astronomical observations, or other physics experiments.

The physics of early cosmology (the “Big Bang”) probes higher energies, all the way up to the Planck scale. It leaves observable effects on the cosmic microwave background radiation (CMBR). For example, it has been suggested that “B-modes” of the CMBR would naturally come out of stringy inflation.

There might also be rare processes such as proton decay, or very weakly coupled particles (WIMPs, gravitinos, axions, etc.) which have not yet been seen yet, and which tell us about physics at high energies. It has been suggested that string theory leads to a very large number of weakly coupled light particles (“axions”) which might make up the dark matter.

To make any claims about what new particles or forces might be predicted by string theory, we must first show that we can use it to derive the particles and forces we know about, described by the Standard Model. If this is not possible, we will know that string theory does not describe our universe.

How do we do this? The starting point is to realize that in string theory, spacetime actually has ten dimensions. This is not in contradiction with experience but only if we postulate that six of these dimensions form a compact manifold M of diameter much less than a micron (the scale at which Newton's inverse square law of gravity has been tested).

One then needs to work out the theory of strings vibrating on M. The topology and geometry of M translate into properties of the matter and forces we observe – the spectrum and masses of particles, the fine structure constant and other couplings, etc..

To make any claims about what new particles or forces might be predicted by string theory, we must first show that we can use it to derive the particles and forces we know about, described by the Standard Model. If this is not possible, we will know that string theory does not describe our universe.

How do we do this? The starting point is to realize that in string theory, spacetime actually has ten dimensions. This is not in contradiction with experience but only if we postulate that six of these dimensions form a compact manifold M of diameter much less than a micron (the scale at which Newton's inverse square law of gravity has been tested).

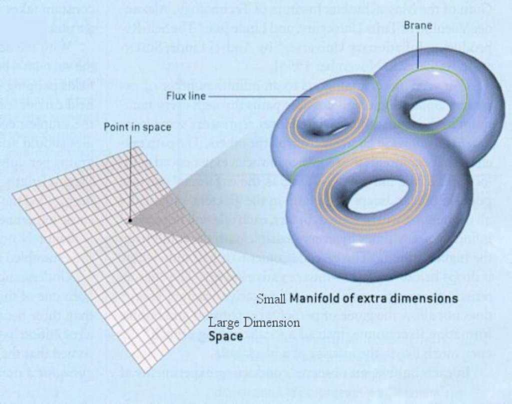
One then needs to work out the theory of strings vibrating on M. The topology and geometry of M translate into properties of the matter and forces we observe – the spectrum and masses of particles, the fine structure constant and other couplings, etc..

To make any claims about what new particles or forces might be predicted by string theory, we must first show that we can use it to derive the particles and forces we know about, described by the Standard Model. If this is not possible, we will know that string theory does not describe our universe.

How do we do this? The starting point is to realize that in string theory, spacetime actually has ten dimensions. This is not in contradiction with experience but only if we postulate that six of these dimensions form a compact manifold M of diameter much less than a micron (the scale at which Newton's inverse square law of gravity has been tested).

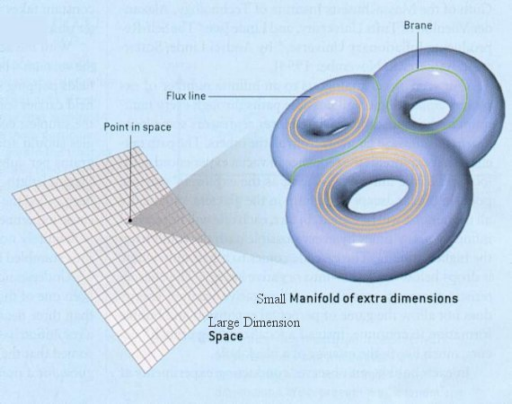
One then needs to work out the theory of strings vibrating on M. The topology and geometry of M translate into properties of the matter and forces we observe – the spectrum and masses of particles, the fine structure constant and other couplings, etc..

There are many possible choices for M – it might be a six-torus, a six-torus quotiented by a discrete group (an “orbifold”), a Calabi-Yau manifold, or many other possibilities. In addition there is additional data (branes, fluxes, *etc.*) to be chosen on M, call such a choice V (we will be a little bit more concrete below).



Each (M, V) can lead to different predictions for observed physics. We do not know *a priori* which of these choices describes our universe. Thus, we need to work out the predictions for many such choices.

There are many possible choices for M – it might be a six-torus, a six-torus quotiented by a discrete group (an “orbifold”), a Calabi-Yau manifold, or many other possibilities. In addition there is additional data (branes, fluxes, *etc.*) to be chosen on M, call such a choice V (we will be a little bit more concrete below).



Each (M, V) can lead to different predictions for observed physics. We do not know *a priori* which of these choices describes our universe. Thus, we need to work out the predictions for many such choices.

The details are a long story but the main point for purposes of this talk is,

> *String theory provides a procedure by which we can construct and classify objects with purely mathematical definitions - manifolds with structure* (M, V) *- and for each one derive a candidate theory of the observable physics in our universe.*

Thus the central problem of theoretical physics – what are the possible fundamental laws of nature – is reduced to pure mathematics.

There are arguments that the number of possibilities for (M, V) is **finite**. If we could show that **none** of the possibilities reproduce the Standard Model (or whatever future physics we discover), we would falsify string theory.

The details are a long story but the main point for purposes of this talk is,

> *String theory provides a procedure by which we can construct and classify objects with purely mathematical definitions - manifolds with structure* (M, V) *- and for each one derive a candidate theory of the observable physics in our universe.*

Thus the central problem of theoretical physics – what are the possible fundamental laws of nature – is reduced to pure mathematics.

There are arguments that the number of possibilities for (M, V) is **finite**. If we could show that **none** of the possibilities reproduce the Standard Model (or whatever future physics we discover), we would falsify string theory.

So far it looks more likely that many of the possible (M, V) reproduce the Standard Model, and that these lead to a variety of further predictions – superpartners of various masses, other extra particles, different models of cosmology etc.

In this case we need to get a sense for the possible (M, V) and possible resulting laws of physics, to know where to look for predictions. Ideally we would derive not just a set $\{(M_i, V_i)\}$ but a probability distribution

$$P(\ (M_i, V_i)\ ) \tag{1}$$

which expresses how likely it is that our universe is described by each of the possibilities. While this may sound crazy, there are proposals for how to do this which are not that different from other derivations of probability distributions in physics.

As an analogy, starting from nuclear physics and the theory of stars, we can derive a theoretical prediction for the abundance of the various elements (hydrogen, helium, lithium etc.) in the universe. This prediction is in good agreement with astronomical observations.

So far it looks more likely that many of the possible (M, V) reproduce the Standard Model, and that these lead to a variety of further predictions – superpartners of various masses, other extra particles, different models of cosmology etc.

In this case we need to get a sense for the possible (M, V) and possible resulting laws of physics, to know where to look for predictions. Ideally we would derive not just a set $\{(\mathrm{M}_i, \mathrm{V}_i)\}$ but a probability distribution

$$\mathrm{P}(\ (\mathrm{M}_i, \mathrm{V}_i)\ ) \tag{1}$$

which expresses how likely it is that our universe is described by each of the possibilities. While this may sound crazy, there are proposals for how to do this which are not that different from other derivations of probability distributions in physics.

As an analogy, starting from nuclear physics and the theory of stars, we can derive a theoretical prediction for the abundance of the various elements (hydrogen, helium, lithium etc.) in the universe. This prediction is in good agreement with astronomical observations.

So far it looks more likely that many of the possible (M, V) reproduce the Standard Model, and that these lead to a variety of further predictions – superpartners of various masses, other extra particles, different models of cosmology etc.

In this case we need to get a sense for the possible (M, V) and possible resulting laws of physics, to know where to look for predictions. Ideally we would derive not just a set $\{(M_i, V_i)\}$ but a probability distribution

$$P(\ (M_i, V_i)\ ) \tag{1}$$

which expresses how likely it is that our universe is described by each of the possibilities. While this may sound crazy, there are proposals for how to do this which are not that different from other derivations of probability distributions in physics.

As an analogy, starting from nuclear physics and the theory of stars, we can derive a theoretical prediction for the abundance of the various elements (hydrogen, helium, lithium etc.) in the universe. This prediction is in good agreement with astronomical observations.

This is all we will say about the physics of string theory. Next we will sketch some computational problems which follow from what we just described and which physicists would really like to have help with. Some basic problems are

- Sample from the possible (M, V) with some probability distribution (the uniform one, or some approximation to P discussed earlier).
- Given an (M, V), compute the resulting spectrum of particles and their interactions.

Of course, string theorists already use computational mathematical tools to work on these problems, and have produced software and databases which are both useful for string theory and of interest to mathematicians. But with our present tools, the full problem is just too big and too hard. We need better tools.

What do I mean by this? Let me spell out a piece of the problem so that the discussion can be more concrete, before explaining.

This is all we will say about the physics of string theory. Next we will sketch some computational problems which follow from what we just described and which physicists would really like to have help with. Some basic problems are

- Sample from the possible (M, V) with some probability distribution (the uniform one, or some approximation to P discussed earlier).
- Given an (M, V), compute the resulting spectrum of particles and their interactions.

Of course, string theorists already use computational mathematical tools to work on these problems, and have produced software and databases which are both useful for string theory and of interest to mathematicians. But with our present tools, the full problem is just too big and too hard. We need better tools.

What do I mean by this? Let me spell out a piece of the problem so that the discussion can be more concrete, before explaining.

Kreuzer-Skarke database

Perhaps the best introduction to computational string theory is to describe the first work of lasting value. This is the Kreuzer-Skarke database of reflexive polytopes, used to classify and work with a particular type of manifold M, the toric Calabi-Yau hypersurfaces.

Let us first give the definition, and then say a few words about where it comes from. A lattice polytope Δ is the convex hull in $\mathbb{R}^{d+1}$ of a finite set of integral points $\nu^{(i)} \in \mathbb{Z}^{d+1}$. Its **dual polytope** is the set

$$\Delta^* \equiv \{y \in \mathbb{R}^{d+1} : x \cdot y \geq -1 \ \forall x \in \Delta\}. \tag{2}$$

A lattice polytope is **reflexive** if its dual is also a lattice polytope.

For fixed d, the set of reflexive lattice polytopes is finite. The Kreuzer-Skarke database lists the 473,800,776 instances for $d = 3$. (See `http://hep.itp.tuwien.ac.at/~kreuzer/CY/`)

Kreuzer-Skarke database

Perhaps the best introduction to computational string theory is to describe the first work of lasting value. This is the Kreuzer-Skarke database of reflexive polytopes, used to classify and work with a particular type of manifold M, the toric Calabi-Yau hypersurfaces.

Let us first give the definition, and then say a few words about where it comes from. A lattice polytope Δ is the convex hull in $\mathbb{R}^{d+1}$ of a finite set of integral points $\nu^{(i)} \in \mathbb{Z}^{d+1}$. Its **dual polytope** is the set

$$\Delta^* \equiv \{y \in \mathbb{R}^{d+1} : x \cdot y \geq -1 \ \forall x \in \Delta\}. \quad (2)$$

A lattice polytope is **reflexive** if its dual is also a lattice polytope.

For fixed d, the set of reflexive lattice polytopes is finite. The Kreuzer-Skarke database lists the 473,800,776 instances for $d = 3$. (See `http://hep.itp.tuwien.ac.at/~kreuzer/CY/`)

Kreuzer-Skarke database

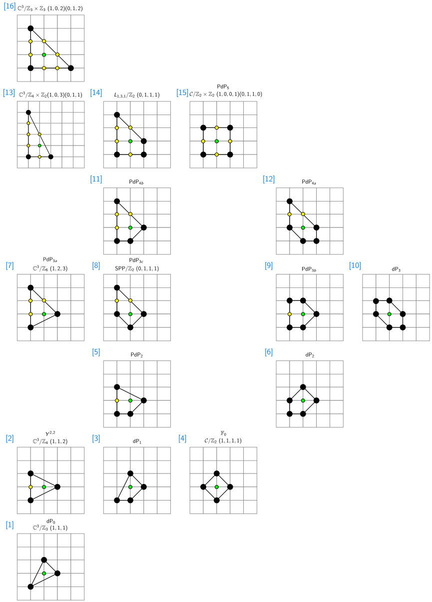
Perhaps the best introduction to computational string theory is to describe the first work of lasting value. This is the Kreuzer-Skarke database of reflexive polytopes, used to classify and work with a particular type of manifold M, the toric Calabi-Yau hypersurfaces.

Let us first give the definition, and then say a few words about where it comes from. A lattice polytope Δ is the convex hull in $\mathbb{R}^{d+1}$ of a finite set of integral points $\nu^{(i)} \in \mathbb{Z}^{d+1}$. Its **dual polytope** is the set

$$\Delta^* \equiv \{y \in \mathbb{R}^{d+1} : x \cdot y \geq -1 \ \forall x \in \Delta\}. \tag{2}$$

A lattice polytope is **reflexive** if its dual is also a lattice polytope.

For fixed d, the set of reflexive lattice polytopes is finite. The Kreuzer-Skarke database lists the 473,800,776 instances for $d = 3$. (See `http://hep.itp.tuwien.ac.at/~kreuzer/CY/`)



From Y.-H. He, R. K. Seong and S.-T. Yau, `arXiv:1704.03462`. The symmetry under reflection across the x-axis relates pairs of dual polytopes.

Why do we care about reflexive polytopes? First, the extra dimensions (in realistic compactifications) must be a six dimensional manifold which approximately satisfies Einstein's equation $R_{ij} = 0$ (Ricci flat). Furthermore we assume there is $N = 1$ supersymmetry in the resulting four dimensional theory. This requires a covariantly constant spinor on M which implies that M is complex and Kähler. A large subset of such M can be obtained as the solution set of a single equation $f(Z) = 0$ in a toric manifold, constructed by gluing charts $(\mathbb{C}^*)^d$ according to a prescription whose data depends on Δ. Then, by Yau's theorem such an M will have a Ricci flat metric iff $c_1(M) = 0$. This will be the case for suitably chosen f if Δ is reflexive.

The condition that Δ^* is also a lattice polytope means that the CY associated to Δ has a second "mirror" CY associated to Δ^*. This "mirror symmetry" was noticed by the physicist Philip Candelas and then explained in these terms by Victor Batyrev.

Why do we care about reflexive polytopes? First, the extra dimensions (in realistic compactifications) must be a six dimensional manifold which approximately satisfies Einstein's equation $R_{ij} = 0$ (Ricci flat). Furthermore we assume there is $N = 1$ supersymmetry in the resulting four dimensional theory. This requires a covariantly constant spinor on M which implies that M is complex and Kähler. A large subset of such M can be obtained as the solution set of a single equation $f(Z) = 0$ in a toric manifold, constructed by gluing charts $(\mathbb{C}^*)^d$ according to a prescription whose data depends on Δ. Then, by Yau's theorem such an M will have a Ricci flat metric iff $c_1(M) = 0$. This will be the case for suitably chosen f if Δ is reflexive.

The condition that Δ^* is also a lattice polytope means that the CY associated to Δ has a second "mirror" CY associated to Δ^*. This "mirror symmetry" was noticed by the physicist Philip Candelas and then explained in these terms by Victor Batyrev.

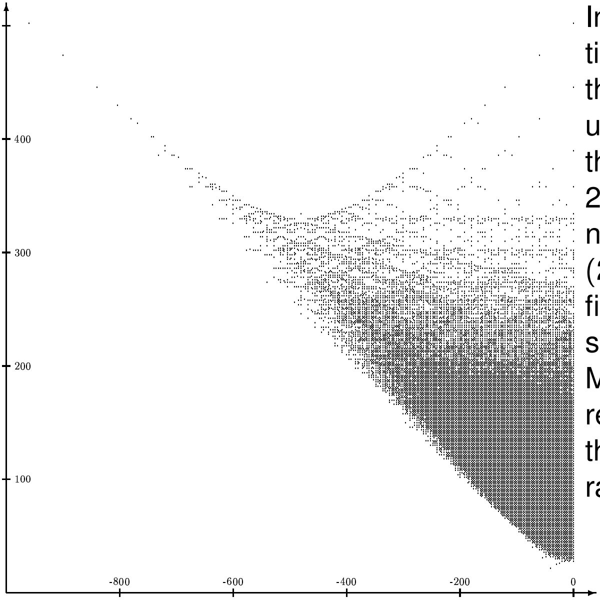


Fig. 1: $h_{11} + h_{12}$ vs. Euler number $\chi = 2(h_{11} - h_{12})$ for all pairs (h_{11}, h_{12}) with $h_{11} \le h_{12}$.

In this figure, the vertical axis is $b^{1,1} + b^{2,1}$, the number of CY moduli. The horizontal axis is the Euler character $\chi = 2b^{1,1} - 2b^{2,1}$, twice the number of generations in $(2,2)$ heterotic compactification. Each dot represents one or more CY_3's. Mirror symmetry is the reflection symmetry with the (omitted) $\chi > 0$ quadrant.

Our point is of course not to delve into these physical and mathematical arguments, but rather to show that a series of natural physical conditions (here, Einstein's equations and supersymmetry) can be translated into mathematics and then into combinatorics.

The successive steps of defining the additional data V and computing the physical predictions can also be translated. The details depend on which limit of string theory we start with: type IIa/IIb/heterotic string, M theory or F theory. Each limit has a preferred class of M leading to realistic physics: the Calabi-Yau threefolds of the Kreuzer-Skarke database are used in type II and heterotic string theory.

In type IIb string theory the first piece of data V is a divisor, (roughly) a submanifold of M of complex codimension 1. It can be defined as the zero set of a function $g(Z) = 0$ on M. In heterotic string theory, the first piece of data V is a holomorphic vector bundle whose structure group is a subgroup of $E_8 \times E_8$. In both cases, this choice determines the gauge group G and matter charges. To reproduce the Standard Model G must contain $SU(3) \times SU(2) \times U(1)$.

Our point is of course not to delve into these physical and mathematical arguments, but rather to show that a series of natural physical conditions (here, Einstein's equations and supersymmetry) can be translated into mathematics and then into combinatorics.

The successive steps of defining the additional data V and computing the physical predictions can also be translated. The details depend on which limit of string theory we start with: type IIa/IIb/heterotic string, M theory or F theory. Each limit has a preferred class of M leading to realistic physics: the Calabi-Yau threefolds of the Kreuzer-Skarke database are used in type II and heterotic string theory.

In type IIb string theory the first piece of data V is a divisor, (roughly) a submanifold of M of complex codimension 1. It can be defined as the zero set of a function $g(Z) = 0$ on M. In heterotic string theory, the first piece of data V is a holomorphic vector bundle whose structure group is a subgroup of $E_8 \times E_8$. In both cases, this choice determines the gauge group G and matter charges. To reproduce the Standard Model G must contain $SU(3) \times SU(2) \times U(1)$.

Our point is of course not to delve into these physical and mathematical arguments, but rather to show that a series of natural physical conditions (here, Einstein's equations and supersymmetry) can be translated into mathematics and then into combinatorics.

The successive steps of defining the additional data V and computing the physical predictions can also be translated. The details depend on which limit of string theory we start with: type IIa/IIb/heterotic string, M theory or F theory. Each limit has a preferred class of M leading to realistic physics: the Calabi-Yau threefolds of the Kreuzer-Skarke database are used in type II and heterotic string theory.

In type IIb string theory the first piece of data V is a divisor, (roughly) a submanifold of M of complex codimension 1. It can be defined as the zero set of a function $g(Z) = 0$ on M. In heterotic string theory, the first piece of data V is a holomorphic vector bundle whose structure group is a subgroup of $E_8 \times E_8$. In both cases, this choice determines the gauge group G and matter charges. To reproduce the Standard Model G must contain $SU(3) \times SU(2) \times U(1)$.

One then continues to make further choices conditioned on the previous choices. For IIb, given a divisor D (roughly, a 4-dimensional submanifold of M), one chooses a vector bundle on D, flux, etc. For heterotic one chooses preferred two-cycles on which to embed "branes," and another type of flux. Eventually one has completed the specification of the geometry.

If we continue this sketch of string compactification, we will get into analytic and numerical details. For example, the relation between "flux" – two elements F, H of $H^3(M, \mathbb{Z})$ – and masses, proceeds through solving the "moduli problem." This is to find a solution of the equation

$$0 = \int_M \sum_i \chi_i^{(2,1)}(t) \wedge \left(F^i + \tau H^i\right) .$$

This can be formulated concretely in terms of period integrals, or Picard-Fuchs equations, and solved using computational tools.

One then continues to make further choices conditioned on the previous choices. For IIb, given a divisor D (roughly, a 4-dimensional submanifold of M), one chooses a vector bundle on D, flux, etc. For heterotic one chooses preferred two-cycles on which to embed "branes," and another type of flux. Eventually one has completed the specification of the geometry.

If we continue this sketch of string compactification, we will get into analytic and numerical details. For example, the relation between "flux" – two elements F, H of $H^3(M, \mathbb{Z})$ – and masses, proceeds through solving the "moduli problem." This is to find a solution of the equation

$$0 = \int_M \sum_i \chi_i^{(2,1)}(t) \wedge \left(F^i + \tau H^i\right) .$$

This can be formulated concretely in terms of period integrals, or Picard-Fuchs equations, and solved using computational tools.

Organizational aspects of the string vacuum problem

- This is a large problem which has been pursued by 100's of people for over 30 years, and it looks like this will continue.
- The problem of constructing a string vacuum has several steps, each involving various mathematical ingredients. The definition of each step depends on the results of the previous steps. For example, given M, there are constructions of the set of vector bundles V over M. But there is no "general construction of V."
- Each step is somewhat intricate and requires theorems and verification. And while each sits in some mathematical field (algebraic geometry, combinatorics, PDE, etc.), the problem as a whole spans disciplines and there is no overall expert.
- Theoretical physicists are experts in **none** of these fields and would rather have the theorems, algorithms and mathematical results provided to them. Of course once a construction is defined, they are the experts at computing the resulting predictions.

Organizational aspects of the string vacuum problem

- This is a large problem which has been pursued by 100's of people for over 30 years, and it looks like this will continue.
- The problem of constructing a string vacuum has several steps, each involving various mathematical ingredients. The definition of each step depends on the results of the previous steps. For example, given M, there are constructions of the set of vector bundles V over M. But there is no "general construction of V."
- Each step is somewhat intricate and requires theorems and verification. And while each sits in some mathematical field (algebraic geometry, combinatorics, PDE, etc.), the problem as a whole spans disciplines and there is no overall expert.
- Theoretical physicists are experts in **none** of these fields and would rather have the theorems, algorithms and mathematical results provided to them. Of course once a construction is defined, they are the experts at computing the resulting predictions.

Organizational aspects of the string vacuum problem

- This is a large problem which has been pursued by 100's of people for over 30 years, and it looks like this will continue.
- The problem of constructing a string vacuum has several steps, each involving various mathematical ingredients. The definition of each step depends on the results of the previous steps. For example, given M, there are constructions of the set of vector bundles V over M. But there is no "general construction of V."
- Each step is somewhat intricate and requires theorems and verification. And while each sits in some mathematical field (algebraic geometry, combinatorics, PDE, etc.), the problem as a whole spans disciplines and there is no overall expert.
- Theoretical physicists are experts in **none** of these fields and would rather have the theorems, algorithms and mathematical results provided to them. Of course once a construction is defined, they are the experts at computing the resulting predictions.

Organizational aspects of the string vacuum problem

- This is a large problem which has been pursued by 100's of people for over 30 years, and it looks like this will continue.
- The problem of constructing a string vacuum has several steps, each involving various mathematical ingredients. The definition of each step depends on the results of the previous steps. For example, given M, there are constructions of the set of vector bundles V over M. But there is no "general construction of V."
- Each step is somewhat intricate and requires theorems and verification. And while each sits in some mathematical field (algebraic geometry, combinatorics, PDE, etc.), the problem as a whole spans disciplines and there is no overall expert.
- Theoretical physicists are experts in **none** of these fields and would rather have the theorems, algorithms and mathematical results provided to them. Of course once a construction is defined, they are the experts at computing the resulting predictions.

Fortunately, pure mathematicians find these problems interesting, not least because there have been very important returns to pure mathematics:

- Mirror symmetry: enumerative formulas, homological mirror symmetry, Bridgeland stability, *etc.*
- New topological and geometric invariants: Gromov-Witten, Seiberg-Witten, Donaldson-Thomas, Gopakumar-Vafa, *etc.*
- Topological field theory and topological quantum gravity, *etc.*

And besides these conceptual developments, string theory has been a valuable source of concrete problems and examples as well.

The contributions of string theorists to these developments have generally not been rigorous mathematics but rather conjectures, new relations, new connections, and new questions which are developed in a loose collaboration between physicists and mathematicians.

Why is this interesting for computational mathematics?

Though I hope some of you are motivated to learn more about string theory and related mathematics, this is not the main goal of my talk.

Rather, it is to argue that computer scientists who work on “higher order” mathematical topics – not just constructing particular proof verifications or algorithms, but who develop the frameworks and software which make this possible – have a unique and important role to play in this project.

Namely, it is to develop the computational framework and platform in which diverse research groups can do their work, integrate it and produce something of lasting value.

Of course this is not particular to string theory – many fields of mathematical science would benefit from having such a platform.

Why is this interesting for computational mathematics?

Though I hope some of you are motivated to learn more about string theory and related mathematics, this is not the main goal of my talk.

Rather, it is to argue that computer scientists who work on "higher order" mathematical topics – not just constructing particular proof verifications or algorithms, but who develop the frameworks and software which make this possible – have a unique and important role to play in this project.

Namely, it is to develop the computational framework and platform in which diverse research groups can do their work, integrate it and produce something of lasting value.

Of course this is not particular to string theory – many fields of mathematical science would benefit from having such a platform.

Why is this interesting for computational mathematics?

Though I hope some of you are motivated to learn more about string theory and related mathematics, this is not the main goal of my talk.

Rather, it is to argue that computer scientists who work on “higher order” mathematical topics – not just constructing particular proof verifications or algorithms, but who develop the frameworks and software which make this possible – have a unique and important role to play in this project.

Namely, it is to develop the computational framework and platform in which diverse research groups can do their work, integrate it and produce something of lasting value.

Of course this is not particular to string theory – many fields of mathematical science would benefit from having such a platform.

Why is this interesting for computational mathematics?

Though I hope some of you are motivated to learn more about string theory and related mathematics, this is not the main goal of my talk.

Rather, it is to argue that computer scientists who work on "higher order" mathematical topics – not just constructing particular proof verifications or algorithms, but who develop the frameworks and software which make this possible – have a unique and important role to play in this project.

Namely, it is to develop the computational framework and platform in which diverse research groups can do their work, integrate it and produce something of lasting value.

Of course this is not particular to string theory – many fields of mathematical science would benefit from having such a platform.

A natural goal for computer scientists is to provide a platform which is general enough to be used throughout the mathematical sciences.

Of course there exist popular platforms for symbolic math, for example Mathematica and Sage. These are a good starting point if rigor is not central.

Mathematica, Sage and the other platforms have large libraries of user-submitted code. In fact the PALP package to work with polytopes and the Kreuzer-Skarke database has been implemented in Sage (mostly by Andrey Novoseltsev). Other standard problems in theoretical physics, such as working with group representations, computing Riemann tensors and checking the equations of general relativity, etc. have many packages devoted to them.

The word “many” already suggests a problem as one might have thought that the best package would have taken over and replaced the others. In practice, theoretical physicists often find it easier to develop their own packages than to learn to use the existing ones.

A natural goal for computer scientists is to provide a platform which is general enough to be used throughout the mathematical sciences.

Of course there exist popular platforms for symbolic math, for example Mathematica and Sage. These are a good starting point if rigor is not central.

Mathematica, Sage and the other platforms have large libraries of user-submitted code. In fact the PALP package to work with polytopes and the Kreuzer-Skarke database has been implemented in Sage (mostly by Andrey Novoseltsev). Other standard problems in theoretical physics, such as working with group representations, computing Riemann tensors and checking the equations of general relativity, etc. have many packages devoted to them.

The word “many” already suggests a problem as one might have thought that the best package would have taken over and replaced the others. In practice, theoretical physicists often find it easier to develop their own packages than to learn to use the existing ones.

A natural goal for computer scientists is to provide a platform which is general enough to be used throughout the mathematical sciences.

Of course there exist popular platforms for symbolic math, for example Mathematica and Sage. These are a good starting point if rigor is not central.

Mathematica, Sage and the other platforms have large libraries of user-submitted code. In fact the PALP package to work with polytopes and the Kreuzer-Skarke database has been implemented in Sage (mostly by Andrey Novoseltsev). Other standard problems in theoretical physics, such as working with group representations, computing Riemann tensors and checking the equations of general relativity, etc. have many packages devoted to them.

The word "many" already suggests a problem as one might have thought that the best package would have taken over and replaced the others. In practice, theoretical physicists often find it easier to develop their own packages than to learn to use the existing ones.

A natural goal for computer scientists is to provide a platform which is general enough to be used throughout the mathematical sciences.

Of course there exist popular platforms for symbolic math, for example Mathematica and Sage. These are a good starting point if rigor is not central.

Mathematica, Sage and the other platforms have large libraries of user-submitted code. In fact the PALP package to work with polytopes and the Kreuzer-Skarke database has been implemented in Sage (mostly by Andrey Novoseltsev). Other standard problems in theoretical physics, such as working with group representations, computing Riemann tensors and checking the equations of general relativity, etc. have many packages devoted to them.

The word “many” already suggests a problem as one might have thought that the best package would have taken over and replaced the others. In practice, theoretical physicists often find it easier to develop their own packages than to learn to use the existing ones.

The PALP package contains about 50 functions, some implementing nontrivial algorithms, but many just “accessors,” for example one to return the number of vertices in a polytope. Its manual is perhaps 40 pages long.

This is a specialized enough application that people do use PALP rather than rewrite it. It surely helps as well that polytopes have a natural presentation with a simple computational representation.

Many mathematical constructs (groups, manifolds, etc.) have many presentations. Working with them effectively requires the ability to convert between presentations, and to derive and work with invariants.

Furthermore, their computational representations involve many arbitrary choices. This leads to a steeper learning curve, and induces a sense in the prospective user that he or she could have made different or better choices. Even once all these details are learned, the arbitrariness creates barriers to communicating results and data between packages, between research groups, etc.

The PALP package contains about 50 functions, some implementing nontrivial algorithms, but many just “accessors,” for example one to return the number of vertices in a polytope. Its manual is perhaps 40 pages long.

This is a specialized enough application that people do use PALP rather than rewrite it. It surely helps as well that polytopes have a natural presentation with a simple computational representation.

Many mathematical constructs (groups, manifolds, etc.) have many presentations. Working with them effectively requires the ability to convert between presentations, and to derive and work with invariants.

Furthermore, their computational representations involve many arbitrary choices. This leads to a steeper learning curve, and induces a sense in the prospective user that he or she could have made different or better choices. Even once all these details are learned, the arbitrariness creates barriers to communicating results and data between packages, between research groups, etc.

The PALP package contains about 50 functions, some implementing nontrivial algorithms, but many just "accessors," for example one to return the number of vertices in a polytope. Its manual is perhaps 40 pages long.

This is a specialized enough application that people do use PALP rather than rewrite it. It surely helps as well that polytopes have a natural presentation with a simple computational representation.

Many mathematical constructs (groups, manifolds, etc.) have many presentations. Working with them effectively requires the ability to convert between presentations, and to derive and work with invariants.

Furthermore, their computational representations involve many arbitrary choices. This leads to a steeper learning curve, and induces a sense in the prospective user that he or she could have made different or better choices. Even once all these details are learned, the arbitrariness creates barriers to communicating results and data between packages, between research groups, etc.

The PALP package contains about 50 functions, some implementing nontrivial algorithms, but many just “accessors,” for example one to return the number of vertices in a polytope. Its manual is perhaps 40 pages long.

This is a specialized enough application that people do use PALP rather than rewrite it. It surely helps as well that polytopes have a natural presentation with a simple computational representation.

Many mathematical constructs (groups, manifolds, etc.) have many presentations. Working with them effectively requires the ability to convert between presentations, and to derive and work with invariants.

Furthermore, their computational representations involve many arbitrary choices. This leads to a steeper learning curve, and induces a sense in the prospective user that he or she could have made different or better choices. Even once all these details are learned, the arbitrariness creates barriers to communicating results and data between packages, between research groups, etc.

Why is there no Wikipedia of formal mathematics?

Asking this question is perhaps the simplest way to make my point. And I apologize if there **is** a Wikipedia of formal mathematics, and that the problem is just one of "marketing." Theoretical physicists would be very happy to use it, and to spread the word to others.

There are many things which make Wikipedia work, but one of the main ones is that people find the assignment of facts to articles reasonably natural (it is by no means unique but they can learn it). And, when several people can make contributions to an article, they are able to integrate them. In practice this process often involves editors – but Wikipedia needs less human editorial effort than one might have thought, given its size and scope.

Of course it is far easier to integrate contributions to a document written for humans, than to integrate contributions to a formal document. Typically a single typo in a formal document renders it "meaningless," at least in the sense prescribed by its syntax and semantics. The choices we discussed lead to far greater problems.

Why is there no Wikipedia of formal mathematics?

Asking this question is perhaps the simplest way to make my point. And I apologize if there **is** a Wikipedia of formal mathematics, and that the problem is just one of "marketing." Theoretical physicists would be very happy to use it, and to spread the word to others.

There are many things which make Wikipedia work, but one of the main ones is that people find the assignment of facts to articles reasonably natural (it is by no means unique but they can learn it). And, when several people can make contributions to an article, they are able to integrate them. In practice this process often involves editors – but Wikipedia needs less human editorial effort than one might have thought, given its size and scope.

Of course it is far easier to integrate contributions to a document written for humans, than to integrate contributions to a formal document. Typically a single typo in a formal document renders it "meaningless," at least in the sense prescribed by its syntax and semantics. The choices we discussed lead to far greater problems.

Why is there no Wikipedia of formal mathematics?

Asking this question is perhaps the simplest way to make my point. And I apologize if there **is** a Wikipedia of formal mathematics, and that the problem is just one of "marketing." Theoretical physicists would be very happy to use it, and to spread the word to others.

There are many things which make Wikipedia work, but one of the main ones is that people find the assignment of facts to articles reasonably natural (it is by no means unique but they can learn it). And, when several people can make contributions to an article, they are able to integrate them. In practice this process often involves editors – but Wikipedia needs less human editorial effort than one might have thought, given its size and scope.

Of course it is far easier to integrate contributions to a document written for humans, than to integrate contributions to a formal document. Typically a single typo in a formal document renders it "meaningless," at least in the sense prescribed by its syntax and semantics. The choices we discussed lead to far greater problems.

While several “benchmark” problems have been proposed for automated mathematical reasoning – finding proofs, verifying proofs, developing a mathematical search engine – I would advocate this one as key:

Problem

Develop a platform which supports shared repositories of formal knowledge, which allows integrating diverse contributions from many sources with minimal human editorial work, and which can be used by practitioners without their needing to know much more than the formal language(s) of the knowledge itself.

In other words, a “wiki for formal language.”
Unlike a wiki for natural language, I think a system to do this needs to have some intelligence – but perhaps only at a level comparable to that needed for verification.

I will conclude by giving a few more desiderata for such a system.

- The analog of an article is a package which defines functions, operations, objects, and specifications. So, if one has a specification S of some class of mathematical objects and operations, it should be possible to send the following query: is there any implementation of S ? The system must be smart enough to discover simple translations between S and the specifications of the existing packages.
  This would address one the basic problems I described, that computational representations of mathematical objects involve much arbitrariness (of data structures, of particulars of function specification, etc.)
- More queries: if S is implemented, what is the translation between my terms, and those of the available implementations? Are there implementations I can download? If my desired operations are computationally intensive, are there servers out there which offer to perform them (perhaps for a fee) ?

- The issues in dealing with multiple presentations of a mathematical object are much deeper. Showing that two presentations define the same class of objects, or giving algorithms to convert between presentations or to compute invariants, can be highly nontrivial mathematics.
  Thus, this aspect of the problem must be an integral part of the framework from the start.

  For example, suppose two groups define objects A and B, and each of A and B accumulates a variety of implementations, algorithms, etc., all similar enough for the system to automatically translate $\mathrm{A} \leftrightarrow \mathrm{A}'$ and $\mathrm{B} \leftrightarrow \mathrm{B}'$.
  Then, some brilliant person shows that there is a highly nontrivial isomorphism between A and B. To what extent can we express this within the framework, and to what extent can this knowledge be used to help answer subsequent queries?

- The issues in dealing with multiple presentations of a mathematical object are much deeper. Showing that two presentations define the same class of objects, or giving algorithms to convert between presentations or to compute invariants, can be highly nontrivial mathematics.
  Thus, this aspect of the problem must be an integral part of the framework from the start.

  For example, suppose two groups define objects A and B, and each of A and B accumulates a variety of implementations, algorithms, etc., all similar enough for the system to automatically translate $\mathrm{A} \leftrightarrow \mathrm{A}'$ and $\mathrm{B} \leftrightarrow \mathrm{B}'$.
  Then, some brilliant person shows that there is a highly nontrivial isomorphism between A and B. To what extent can we express this within the framework, and to what extent can this knowledge be used to help answer subsequent queries?

- The system should accept submissions from any and all, but their value will depend greatly on the extent to which they have sensible specifications and can be verified to efficiently implement them. This needs to be taken into account in answering the queries: if packages P_1, P_2, etc. have specifications which claim to solve problems specified by S, how can the system supply data to the user to allow him (or her, or it) to choose which ones to try, and to be convinced that the results they supply are correct?
- Updates to definitions must not interrupt their use. Except in the simplest cases, changes will probably be made not by modifying an existing package, but by defining a new version with its own specification, and letting the query mechanism redirect new requests to the new version. Thus, there will be an ongoing need to update references to packages, and to verify that these updates are correct.

Further reading, conferences, etc.

Representative recent work on computational string theory:

- Tools for CICYs in F-theory, Anderson *et al*, arXiv:1608.07554
- Scanning the skeleton of the 4D F-theory landscape, Taylor and Wang, arXiv:1710.11235
- Calabi-Yau Volumes and Reflexive Polytopes, He, Seong and S.-T. Yau, arXiv:1704.03462
- Vacuum Selection from Cosmology on Networks of String Geometries, Carifio *et al*, arXiv:1711.06685
- The Minimal SUSY $\mathrm{B}-\mathrm{L}$ Model: Simultaneous Wilson Lines and String Thresholds, Deen, Ovrut and Purves, arXiv:1604.08588
- http://hep.itp.tuwien.ac.at/~kreuzer/CY/

String theory and data science conferences:

- Northeastern, Nov 30-Dec 2, 2017:
  `https://web.northeastern.edu/het/string_data/`
- Munich, March 26-29:
  `https://indico.mpp.mpg.de/event/5578/overview`

Textbooks:

- *String Theory and M-Theory: A Modern Introduction*, K. Becker, M. Becker and J. H. Schwarz.
- *Superstring Theory*, M. B. Green, J. H. Schwarz and E. Witten.

Historically important work:

- W. Lerche, D. Lust, and A. N. Schellekens, Chiral Four-Dimensional Heterotic Strings from Selfdual Lattices, Nucl.Phys. B287, 477, 1987.
- K. Dienes, Statistics on the heterotic landscape, `hep-th/0602286`