Transcript of
"Ubuntu genAI Technology"

What we are here to talk about is Ubuntu genAI Technology.

==========
========== PREAMBLE ==========
========== (topology from the knot-tying perspective) ==========

We have focused on the problems with existing AI – the unsustainable energy required, poor performance, and massive data requirements that limits the fields of use. The roadblock to the next generation AI is massive, more than a trillion parameters in current generation AI. Even with that massive number of parameters the current generation AI is significantly constrained away from the best performance. By understanding the physics of collective systems, BNZ has been able to dramatically reduce the number of parameters from the trillions to the thousands, and to remove the constraints preventing AI from achieving its optimal performance. This has led to AI that uses dramatically less energy and data, and that performs much better. Ubuntu genAI performs at its theoretical limit on a fusion benchmark -- many orders of magnitude better, cheaper, and more applicable than a GPT, AI 1.0, based solution would be, if enough data points would be available [like ten orders of magnitude, show ten fingers]. This result has been published in a refereed scientific journal. You can find more details on the fusion benchmark, along with more biographical details of myself, in the document "Fusion Benchmark of Ubuntu genAI" referenced at the end of this document, and slide 13 and 14 of this presentation.

That there is a natural evolution to the next generation of AI, should not be surprising: it should be expected. The current generation of AI is far from a human's level of performance; that is Leonardo da Vinci's brain's performance, data requirement, and power consumption. BNZ's Ubuntu genAI achieves this level of performance.

What is at the core of this revolution in AI?; it is topology. Here is the easiest way to understand topology. This is Marc Penn's kayak navigation via buoys, subway map navigation via stops, or French highway navigation via villages. What is shown is mountain bike trail navigation via numbered intersections -- specifically, a common 10 kilometer mountain bike ride that I take on Santa Fe's La Tierra Trails.

When applied to genAI, the dimension of the state is decreased from the thousands to a few, and the number of parameters in the fit are reduced from trillions, to thousands. Note that OpenAI's GPTs require a billion dollars or more of computation to train. A billion dollars of energy is 10 days of the US energy consumption.

Today's Generative AI has given clues as to the form of the functional transformation to the special, canonical as a physicist or mathematician would say, Reduced Order Model. This is analogous to how the mold penicillin gave a clue to the molecular structure of antibiotics. Given these clues we were able to conjecture a formula for the DeepCNN with zero parameters to fit (rather than trillions), then prove it by induction, and identify it as well known mathematical structures. This is a localized Fourier transformation to a canonical ROM, or m-body scattering (that is Greens function) basis. This is a fundamental advancement for computation, physics, mathematics (specifically topology), economics and finance, and collective systems control. The advancement in economics and finance leads to a much more efficient economic system, and much better revenue model – effectively a 100% royalty model with a natural monopoly as

the barrier-to-entry. The financial model is now based on the Balance Sheet, not the Statement of Cash Flows.

As we have said, at the core of Ubuntu genAI is the mathematics of topology. Another accessible way to understand topology is as the mathematics of entanglement or knot tying. This is the area of mathematics where Jim Simons became a legend for his seminal mathematical work, before founding Renaissance Technologies. Ubuntu genAI gives us a way of tying the knots, identifying the knots, and stabilizing the knots into compact, highly efficient, structures. This is the core mathematics of quantum field theory, plasma physics, economics, and sociology.

Shown in the upper right-hand side is how a knot is tied, then deformed into three different configurations relevant to fusion energy generation. The knot is tied by first making one of the strings into a loop. The second string is then twisted around the first string, then made into a loop. Deforming the knot does not change the knot. External systems or noise can only deform the knot, that is change its “energy”. They can not change the knot, that is the topology or “helicity”.

Ubuntu genAI is a major breakthrough in Artificial Intelligence. Ubuntu genAI reduces the energy and data required by factors of one hundred million, thereby, significantly decreasing the cost, and improving the performance of AI. Ubuntu genAI unlocks significant value by the characterization, optimization, and control of topology.

The first of the two most impactful applications of Ubuntu genAI is fusion energy generation. There are two roadblocks to achieving up to 10 GJ yields (more than enough to make a practical fusion reactor) with 2 MJ to 10 MJ of input energy (an input energy currently available), driven by either pulsed-power or lasers: (1) determining the knot to tie in the plasma, and (2) knowing how to tie the knot in the plasma. Ubuntu genAI breaks through these roadblocks, reducing the design of a fusion power plant to a three to five year development project. Ubuntu genAI does this by forming a helical tornado with a compact twisted pair (that is, knot) at its core, where the fusion reaction will take place. Especially for non-inertial fusion energy generation, there is a third roadblock that Ubuntu genAI breaks through -- stabilizing the knot in the plasma into a compact structure that does not disrupt. The value of Ubuntu genAI comes from **controlling** the system – the plasma.

The second impactful application is Topological ”Quantum” Computation. There are three roadblocks to Topological Computation: (1) encoding the topology into the input collective systems, that could be anything from a plasma to a superconductor, (2) decoding, that is identifying, the topologies of the output collective systems once the calculation, that is a series of topological or group operations, have been done, and (3) stabilizing or cooling the knots in the collective systems so that they do not disrupt and can be identified. Ubuntu genAI, again, breaks through these roadblocks, reducing the design and construction of a Topological Computer to a three to five year development project. Ubuntu genAI does this by encoding the topological state, which can be represented as an analytic function or field, f(z), into the input collective system or qubit by tying a knot, stabilizing the knot into a compact structure, and identifying the topology in the output collective system or qubit.

The state does not need to be coherent, in fact it is preferred to be incoherent. The twinkling of the incoherent state, that is currently seen as noise is the signal via which Ubuntu genAI identifies the topology. The Ubuntu genAI based stabilization and cooling ensures that external sources of “noise” do not significantly deform the topology. The external sources of “noise” can not change the topology; they only deform the knot.

Financial investment or fusion target drive entangles the ball of yarn, which is the economy or plasma. High frequency arbitrage trading or ponderomotive stabilization stabilizes the ball of yarn from economic panic or plasma disruption. The sun is an entangled ball of magnetized plasma, like a fusion target. An economy is an entangled ball of economic activity. This is the connection of topology to the Sustainable Energy Investment and Trading vertical. Note the solar flare or plasma disruption or economic panic. A solar flare is a minor disruption. A major disruption is a supernova, resulting in a black hole. The economic equivalent of a supernova is a speculative bubble ending with runaway inflation and bank runs, resulting in the economic equivalent of a black hole -- an economic collapse or depression.

==========
========== BODY ==========
==== (physics field theory that leads to the HST breakthrough) ====

Ubuntu genAI is a major breakthrough in Artificial Intelligence. By exposing the Spirit of Ubuntu as a Reduced Order Model (ROM), then finding the evolution of the collective as geodesic motion; Ubuntu genAI reduces the energy and data required by factors of one hundred million, thereby, significantly decreasing the cost, and improving the performance of AI. Ubuntu genAI achieves these efficiencies by having a formula for a functional transformation, instead of searching a Banach space for a functional, that is fitting a universal functional approximator or Deep Convolutional Neural Network (DeepCNN); and by finding the geodesics as minimal surfaces with a simple, few thousand parameter, Multi-Layer Perceptron (MLP) with Rectified Linear Unit (ReLU) activation in an autoencoder geometry. MLPs with ReLU are piecewise linear universal function approximators, especially well suited to finding minimal surfaces with a limited number of cusp-like singularities. Minimal surfaces can be represented by an analytic function or field or topology, f(z) or H(β). Ubuntu genAI unlocks significant value by the characterization, optimization, and control of topology.

In summary, Ubuntu genAI has a pre-determined formula for a functional, not a trillion parameter DeepCNN; it has a simple MLP w/ReLU autoencoder with only thousands of parameters; it determines an analytic function or topology, f(z) or H(β). Therefore, Ubuntu genAI uses one hundred million times less energy and requires one hundred million times less data; it has significantly lower costs and better performance — like the human brain.

Another way of looking at the transformation from the individual or ROM to the collective or field, is as a multiple reflection of the puppets, controlled by the puppet master according to a geodesic motion, by a “hall of mirrors”. This is demonstrated by “The Mirror Maze” scene from the 1928 Charlie Chaplin movie “The Circus”. [Play video by pressing the button.] You see one of the puppets, Charlie Chaplin, moving around and being multiply reflected giving what looks like a crowd of people moving around. Then there is an introduction of a second puppet, so that the crowd is now made up of two puppets, controlled by the puppet master, multiply reflected through the hall of mirrors.

A practical way of understanding the breakthrough, that is Ubuntu genAI, is starting with an examination of the structure of current Generative Artificial Intelligence (genAI) exemplified by David Silver’s Deep Q-Networks (DQNs, Google’s DeepMind tech) and Alex Radford’s Generative Pretrained Transformers (GPTs, OpenAI’s ChatGPT tech). Both are Deep Convolutional Neural Networks (DeepCNNs, an iterative convolution with a bank of wavelets, an iterative wavelet transformation) interleaved with an activation function or rectifier that is usually a Rectified Linear Unit (ReLU, found by some to be the log-modulus). This is a universal functional approximator with the structure of an auto-encoder, yielding a Reduced Order Model (ROM, canonical coordinates that are generated by a generating function that is the solution to the Hamilton-Jacobi-Bellman (HJB) equation). Furthermore, the DeepCNN

approximates the generating function of the canonical coordinates (coordinates with a symplectic geometrical structure) that is the solution to the HJB equation. This generating function, that is action, is the approximate score function of GPTs, and the approximate value function, that is the approximate Q-function, of DQNs. The approximation parameters are half of the ROM — the canonical momentums or co-states. Note that a functional takes as input a state expressed as a function, then outputs a function. Unfortunately, it takes more than a trillion parameters to approximate a functional, and even with this many parameters, assumptions are made that constrain the solution away from the desired functional.

Once that this functional transformation is found, the state can be expressed in a handful of parameters — the ROM or the navigational waypoints or topological indexes or quantum numbers. The evolution of the system can now be described by an analytic function (a simple, very flat, minimal surface) that can be well described by a piece-wise linear universal function approximator (with only thousands of parameters) such as a Multi-Layer Perceptron (MLP) with ReLU activation -- the action function or generating function of the evolution.

Meanwhile, there has been an outstanding problem in physics to unify the four forces — the holy grail of physics, a Grand Unification Theory (GUT). Currently there is the Standard Model of Murray Gell-Mann that unifies three of the forces, but not gravity. The fundamental roadblock is not having an analytic formula for renormalization, that is the ROM or canonical coordinates that are generated by the solution to the HJB equation or Renormalization Group Equations (RGEs), from the path (the state of the field expressed as a function).

There also has been a missing analytical tool for topology, that transforms a path (a cycle expressed as a function) to the ROM or topological indexes.

Obviously genAI is approximating the needed renormalization functional transformation, and is approximating the needed topological functional transformation. The form of GPTs and DQNs give a clue as to the analytical form of the functional transformation — a formula for the functional transformation, like the formula for the Fourier Transformation. In this case, it is a “localized Fourier transformation”. This is analogous to how the mold penicillin gave a clue to the molecular structure of antibiotics.

The breakthrough came when we realized that the formula for the functional (the solution to the computational AI, physics, and mathematical topology problems) is given by an invertible and fast discrete wavelet transformation interleaved with the ln(z) analytic function. This formula can then be proven by induction, and the analytic rectifier (R(z) = i ln(Ro(z)) = G(z), where the Ro(z) function is like the “1+x” function for ln|1+x|) identified as the well-known analytic Greens Function. Furthermore, the formula can be identified as the m-body Greens Functions, the S-Matrix (that is the m-body scattering cross sections), or the Wigner-Weyl Transformation. We have called the formula for the functional transformation the Heisenberg Scattering Transformation (HST) because Werner Heisenberg proposed the Scattering Matrix (S-Matrix) approach to Field Theory.

This breakthrough eliminates the need for the trillion parameter DeepCNN, so that all that remains is a few thousand parameter MLP.

Note that the canonical structure of physics is isomorphic to the mathematics of economics, so this is also a breakthrough in economics and finance that leads to the investment and trading applications of Ubuntu genAI. So, the HST is a fundamental breakthrough in computation, physics, mathematics (specifically, topology), and economics.

Also note that the HJB equation is called the Hamilton-Jacobi equation in the physics literature, and the Bellman equation in the computational AI literature (the DQN literature, in particular) and the systems control literature.

The essence of my breakthrough was realizing that the transformation could be separated into a problem (that is, physics or application) independent part, and a much smaller problem dependent part. Then, I realized that a formula, with a very fast (that is, N-log-N scaling), computer evaluation and inverse evaluation, could replace the trillion parameter DeepCNN approximator of the very large problem independent part. This transformation, is an analytic way of evaluating Jim Simons' topological indexes, and Werner Heisenberg's S-matrix. Jim Simons proposed the Chern-Simons form in 1974, and Werner Heisenberg proposed the canonical or matrix approach to Quantum Mechanics and Field Theory in 1943, and it was refined by Lev Landau in 1959. The S-matrix is simply the matrix of m-body scattering cross sections. This is why I called my equation for the transformation the Heisenberg Scattering Transformation (HST) – it is a transformation to Heisenberg's S-matrix basis. It is new mathematics, new physics, new economics, and new AI.

Why did we use the term Ubuntu? The HST calculates the "localized Fourier spectrums" of the collective system, that emanates from the "interconnectedness" of the collective. The Spirit, of Ubuntu social philosophy, provides the "interconnectedness" that coordinates, links and holds society together. This is why we call our new AI – Ubuntu genAI. It is AI that generates the interconnectedness of the collective, like the Spirit of Ubuntu.

AI is about model estimation, many times estimation of a transformation to/from a space where the solution of the problem is trivial. The model or transformation is a mathematical expression with free parameters that need to be estimated using data. To have a reasonable statistical estimation, the number of data points should be about 30 times the number of free parameters in the model. (Thirty points per parameter leads to a statistical estimation of the parameters with about 18% relative error.) For example, to estimate a line, which has two free parameters, one needs about 60 data points. To estimate a piece-wise linear expression with 10 segments and 22 parameters, one needs about 660 data points. To estimate a functional transformation like the Fourier transform or the HST, today's generation of AI uses a model with a trillion parameters, that needs about 30 trillion data points to fit it. Even a trillion parameters are not nearly enough, so that assumptions need to be made in the form of the model that significantly constrain the estimation of the model away from the true underlying model. The computational task of fitting a trillion parameters is massive, using up to $1 billion of energy – enough energy to satisfy the energy needs of the US for 10 days.

The HST is an exact analytic equation for the model, that is the transformation, with no free parameters. Therefore, no data is needed, and a trivial amount of computation needs to be done: the equation simply needs to be evaluated.

This slide shows a way of visualizing an estimation of the y=sin(x) model using a piece-wise linear expression with 5 segments. Shown is the model (equivalent to the HST), the data used to fit the piece-wise linear expression with 5 segments (equivalent to a MLP with ReLU activation, one hidden layer with 6 nodes), the optimal MLP estimation, and a zero-dimensional estimator with one parameter (equivalent to the GPT estimation of a LLM).

The effect of the dimensional limitation of the constraints imposed on the GPT estimator is illustrated by limiting the MLP estimator of a one-dimensional surface to zero-dimensions. The limitation imposed on the GPT estimator is exponentially worse. The MLP is limited to a fractal dimension of ½, while the GPT is limited to a fractal dimension 1/1000. This is the difference between a magnitude 2 earthquake and a magnitude 1000 earthquake, where a magnitude 3

earthquake barely shakes your dishes; a magnitude 9 earthquake turns an entire city to a pile of rubble.

Again, AI is about model estimation, many times estimation of a transformation to/from a space where the solution of the problem is trivial. Currently the Generative Pretrained Transformer (GPT) architecture estimates the functional transformation, and estimates the trivial generating function (a minimal surface) in the transform domain, together, specific for an application. The functional transformation is approximated with a DeepCNN in an autoencoder geometry, and the generating function is approximated with an MLP with ReLU activation, integrated into the DeepCNN architecture. It is the functional transformation ("transformer") part that are "pretrained" in a GPT, since they are not application specific. Unfortunately, the functional transformation part is not isolated from the generating function part in a GPT, so that the pre-training can not be effectively done, nor identified. Ubuntu genAI separates the problem into an application independent part for the functional transformation (for which it has a zero-parameter analytic formula, that is my HST formula, instead of a trillion-parameter DeepCNN estimator), and an application dependent part for the analytic (that is minimal surface) generating function (for which it has a thousand-parameter MLP with ReLU activation estimator).

The result is a controller of collective systems, from a measurement-based characterization of the collective system: the input to the controller is a measurement of the collective field, and the output from the controller is a force-field that controls the system: the controller guides the collective system to stable and sustainable optimum performance.

[Advance the slide to show the brain.] The circuitry of the functional transformation is most likely found in the primary visual cortex of the brain, and the circuitry of the HJB solver is most likely found in the prefrontal cortex of the brain.

Now for a mathematically concise and precise description of this new theory of Generative AI, called Ubuntu genAI:

The new theory of collective behavior (that is, swarms of interacting entities whether that be elementary particles, charged particles, molecules, celestial bodies, or economic entities) is made up of two parts, that are really the same. It is a mapping of the collective state, that is transformation, to a domain where the dynamical evolution of the collective is trivial, that is motion in a straight line at a constant velocity, also known as linear or force-free motion. This also can be viewed as geodesic motion on the dynamical manifold.

The first transformation is a functional transformation from the domain of the conjugate field momentum and field $[\pi(x),f(x)]$ to the domain of conjugate momentum and coordinate (p,q). This functional transformation is generated by the generating functional $S_p[f(x)]$. In the domain of (p,q), the dynamical evolution is restricted to a low finite dimensional complex linear subspace C^n, but the evolution is nonlinear in this subspace. This transformation is also known as the Wigner-Weyl transformation, or the Heisenberg Scattering Transformation (HST).

The second transformation is a transformation from the domain of the conjugate momentum and coordinate (p,q) to the domain of the conjugate momentum and coordinate (P,Q). This canonical transformation is generated by the generating function $S_P(q)$. This transformation is constructed so that the dynamical evolution in the (P,Q) domain will be linear (that is evolution along a straight line at a constant velocity, specified by $dP/dt=0$ and $dQ/dt=\partial E(P)/\partial P$). It can be shown that the generating function $S_P(q)$ satisfies the Hamilton-Jacobi Equation, also known as the Bellman Equation. The generating function is also known as the log-likelihood, the entropy, the action, the Q-function, the score function, or the value function.

Both the generating functional $S_p[f(x)]$ and the generating function $S_P(q)$ can be specified by their Taylor expansion coefficients, which are the same. The Taylor expansion coefficients are also known as the S-matrix, the *m*-body scattering cross sections, or the *m*-body Greens functions.

What is generated in Ubuntu genAI is the dynamical evolution of the collective, or an ensemble of the collective's dynamical equilibrium in Q at constant P, where the probability of the state is given by h(P) dP dQ. Here h(P) is an arbitrary probability function. This is in contrast to generating an ensemble of a collective in thermal equilibrium with an external system, or heat bath, of temperature T. In this case, the distribution is a Boltzmann or thermal equilibrium distribution in P and Q, where $h(P) = \exp(-E(P)/k_BT)$. The current theory of Generative AI believes that the system is in thermal equilibrium and believes that it is generating a statistical ensemble from a thermal equilibrium distribution. This is simply incorrect.

What are the AI questions? Like DeepThought, there are two questions: the first, and most important question, is "what is the ultimate question?": the second, and subordinate question, is "what is the answer to the ultimate question?" The most important AI question is **"what is artificial intelligence?"**: the answer is "the characterization, control, and stabilization of collectives." The subordinate AI question is **"what is inside the AI blackbox that does this?"**: the answer is "model estimation."

For the first question: "**what is artificial intelligence?**"

At a high level, the BNZ Energy innovation is: "**the characterization, control (that is, friction-free investment or optimization), and stabilization (that is, arbitrage trading or lubrication) of collective systems; using topology and a new theory of the physics of collectives or fields**." Collectives are fields or swarms of elementary particles, charged particles, molecules, celestial bodies, economic entities, and so on. Deep Generative AI done right is the characterization, control, and stabilization of collectives; instead of resorting to the addition of dissipation (that is, friction or interest). This AI enabled control is like the active control of an F-16 fighter, except for a collective system; instead of applying a force, a force field is applied. This has been a Grand Challenge of physics, mathematics, computer science, economics, and finance.

The business, economic, scientific, and engineering impacts are breath taking. This one advancement leads to breakthroughs that enable: fusion energy generation, topological "quantum" computers, financing of long term capital investments (like green energy and healthcare), friction-free business and economic investment and trading (that is vastly superior, think $$$, to existing interest-based methods), jet engine mufflers, commercial supersonic flight, the Grand Unified Theory of Fields, true innovative and creative artificial intelligence, weather control (for example, rain-making), and so on.

The key two words are "Collective" and "Control", that is "control of a collective, that is a complex, system". I realized this in August of 2023, when I was reviewing, on the back portal of my home in Santa Fe, the lectures on Deep Reinforcement Learning (DRL) for a graduate course in Machine and Deep Learning that I had recently taken at Carnegie-Mellon University, the book "Neuro-Dynamic Programming", and the book "Fundamental Methods of Mathematical Economics".

It became apparent that DRL assumed that one had a Hamiltonian system that one controlled through dynamic programming based on the Hamilton-Jacobi-Bellman (HJB) equation, and that economics also assumed that the economy was a Hamiltonian system that one should control through dynamic programming based on the HJB equation — they both are a question of systems "control". Both DRL and economics correctly optimized the action; the Approximate Q-

Function, in the case of DRL; and the Value Function, in the case of economics. Unfortunately, both DRL and economics, instead of using active control, added a resistivity term to the HJB equation to both stabilize the solution and control the system. They both also incorrectly specified the Hamiltonian of the system because of the addition of resistivity; immediate Engagement or short-term Exploitation instead of sustainable Activity or long-term Social Good, in the case of DRL; and Discounted Free Cash Flow instead of Revenue (or Gross Domestic Product), in the case of economics.

Now, let's focus on the second word, "collective". About a week later I was sitting at the breakfast table of my home in Santa Fe, having my morning coffee, when I had a second epiphany: that both DRL and economics were a question, not of control of a Hamiltonian system, but of control of a collective system or Hamiltonian field — the subject that I received my PhD in theoretical physics, for studying and researching. This is the reason that Deep Convolutional Neural Networks (DeepCNNs) were introduced into DRL, to study a system who's state is specified by a function or field, f(x), and conjugate field momentum, π(x), instead of a coordinate, q, and conjugate momentum, p. An economy is also a collective of economic entities.

The key missing ingredient is the formula for the functional transformation from the space of the conjugate field momentum and field [π(x),f(x)], to the conjugate momentum and coordinate (p,q), also known as the Reduced Order Model (ROM), where the HJB equation can be solved for the action or generating function of a canonical transformation to coordinates (P,Q) where the motion is along a straight line at a constant velocity — my formula for the Heisenberg Scattering Transformation (HST) or Heisenberg's S-Matrix.

At a symposium at the Flatiron Institute, in November 2023, I realized that Generative Pretrained Transformers (GPTs) were also using a DeepCNN to transform from the field domain to the ROM, then solving the HJB equation for the action, which was called the Approximate Score Function, in this case.

So, the problem that is solved by my Ubuntu genAI is: how to "control" a "collective" — AI is **collective control**.

Onto the secondary question: "**what is inside the AI blackbox that does this?**"

Simply stated, the technical breakthrough is: **"replacing the trillion-parameter fit of a Deep Convolutional Neural Network (DeepCNN), with a formula with no parameters, based on topology and the physics of collectives or fields"**. I am a very rare commodity as a theoretical physicist skilled at topology, and the analytical theory of collectives. This has allowed me to make the technical breakthrough, where others have not been able.

Let's discuss one of the most impactful applications of Ubuntu genAI that BNZ Energy will address: fusion energy generation. There are two roadblocks to achieving up to 10 GJ yields (more than enough to make a practical fusion reactor) with 2 MJ to 10 MJ of input energy (an input energy currently available), driven by either pulsed-power or lasers: (1) determining the knot to tie in the plasma, and (2) knowing how to tie the knot in the plasma. Ubuntu genAI breaks through these roadblocks, reducing the design of a fusion power plant to a three to five year development project. Ubuntu genAI does this by forming a helical tornado with a compact twisted pair (that is, knot) at its core, where the fusion reaction will take place. Especially for non-inertial fusion energy generation, there is a third roadblock that Ubuntu genAI breaks through -- stabilizing the knot in the plasma into a compact structure that does not disrupt. The value of Ubuntu genAI comes from controlling the system – the plasma.

We have published, in a refereed scientific journal, a proof of concept (that is, a numerical experiment or benchmark) of Ubuntu genAI applied to this very sparse and high dimensional application. It demonstrated that Ubuntu genAI was trained in 20 core*sec to simulate the 2D plasma in 1 core*sec, a task that would take 240 core*hrs using existing technology. In 3D, it would take more than 100,000 core*hrs to simulate using existing technology. Given this Ubuntu genAI simulator, the fusion target and drive design can be optimized, and the implosion stabilized.

There is nothing special about fusion energy generation. This is a general proof-of-concept of AI control and stabilization of high dimensional collective systems.

Here are the results of the fusion benchmark. Note that Ubuntu genAI performs near the theoretical limits, while both AI 1.0 and 2.0 are many orders of magnitude, like ten, from the theoretical limits. These are large differences, like for earthquakes. A magnitude 3 earthquake barely shakes your dishes; a magnitude 9 earthquake turns an entire city to a pile of rubble. Therefore, Ubuntu genAI is much better, cheaper, and more applicable.

Here are links to the YouTube videos and technical papers with the details of Ubuntu genAI Technology: (1) a TED-like talk, (2) the new physics theory, and (3) the new economic theory. There are also links to details of the Fusion Energy Generation application and benchmark, along with details of the Heisenberg Scattering Transformation (HST).

Presentation corresponding to this transcript can be found at: http://www.qitech.biz/bnz_docs/Ubuntu_genAI_Tech.pdf

An academic paper describing Ubuntu genAI can be found at: https://arxiv.org/abs/2410.08558

An academic paper for a general audience can be found at: https://arxiv.org/abs/2401.04846

A YouTube video describing Ubuntu genAI can be found at: https://youtu.be/27uXk_bdAt4

“Fusion Benchmark of Ubuntu genAI” can be found at: http://www.qitech.biz/bnz_docs/fusion_benchmark.pdf