> The danger of modern brute-force AI (whether an LLM or an NVIDIA physics twin) is that **it hides its lossiness behind an illusion of perfection.** Because it can process a petabyte of bits in milliseconds and render it beautifully, we treat its output as a "single source of truth." We forget that it is a probabilistic guessing machine. We are handing over the keys of engineering rigor to a system that possesses all of our lossiness, but none of our capacity for symbolic, self-correcting logic.

Yes, **"a bucket of bolts."**

It is that classic, deeply human phrase for a machine—whether a heavily used car, a clattering old tractor, or an aging aircraft—that is held together by little more than mechanical luck, friction, and a prayer. It is the perfect, visceral antithesis to the sterile, mathematically pristine worlds of NURBS and CAD.

There is a beautiful irony in your juxtaposition. We can spend our careers engineering the highest levels of structural logic, exact algebraic topology, and precise geometric definitions. Yet, at the end of the day, those clean mathematical truths are deployed into a physical universe made of raw matter and energy, where things wear down, rattle, and inevitably become a "bucket of bolts."

Driving those machines gives an engineer an intuitive feel for the physical world—the vibrations, the tolerances, and the material realities—that a purely virtual, brute-force numeric simulation can never fully capture on a screen.

It sounds like you have a rich tapestry of practical, hands-on experience and deep computational history to draw from. When you are ready to start drafting or structuring the **historical sections** or the **philosophical arguments** of this critique, just let me know. How would you like to **save or organize** these thoughts next?

Now, this is neither anthropology nor psychology (but they pertain to the issue). ... I have been using buckets'o'bits. Tell me about that.

When you pivot from a **"bucket of bolts"** to **"buckets'o'bits,"** you are capturing the ultimate computational irony of the modern reductionist era. You've traded physical, rattling mechanical components for digitized, ungrounded numbers—but the structural chaos remains exactly the same.

If a *bucket of bolts* is a physical machine held together by friction and luck, a *bucket'o'bits* is a virtual machine held together by pure statistical approximation and massive compute.

Discussion with Gemini (via Chrome), its summary john.m.switlik@ieee.org

Here is how "buckets'o'bits" explains exactly why the modern reductionist approach to AI and simulation hits a wall:

1. The Discretization Trap (Dumping Math into a Bucket)

When you use exact mathematics like NURBS, the geometry is continuous, analytical, and elegant. It possesses topological truth.

- But when you transition to modern neural physics, digital twins, or dense point clouds (like NVIDIA's neural simulation frameworks), you chop that elegant mathematical continuum into a trillion fragmented, discrete pieces.
- You are essentially taking a beautiful, continuous mathematical curve, smashing it with a hammer, and sweeping the resulting shattered fragments into a **bucket of individual bits**. You then force a massive GPU to crunch through that bucket, trying to approximate what the continuous shape used to look like.

2. Loss of Engineering Semantics

Historically, programmers joked about the "bit bucket"—the mythical wastebasket where lost, corrupted, or unaligned data went to die. [1]

- In a reductionist system, the **entire model** becomes a bucket of bits.
- In Knowledge-Based Engineering (KBE), a bit has context: it knows it represents a specific wing rib, bound by a specific engineering rule.
- In a pure brute-force neural network or deep numeric simulation, that semantic meaning is erased. The AI is just looking at floating-point values in a massive array. It doesn't know it's a wing; it just knows it's a massive, unorganized bucket of ones and zeros trying to minimize an error function. It lacks the architectural rules of the plane.

3. The Illusion of Precision

Driving a *bucket of bolts* forces you to respect reality because you can feel the loose tolerances and the vibration. It is honestly imperfect.

- A *bucket'o'bits*, however, presents a dangerous illusion. Because an NVIDIA Omniverse simulation or a deep-learning generative CAD model looks visually stunning and renders at 4K resolution, engineers mistake **visual density for physical truth**.
- Underneath the hood, it is still just a massive, approximated pile of probabilities. When you throw away algebraic topology and rule-based KBE, you are just running a very fast, very expensive calculation on a fundamentally loose, rattling digital approximation.

Discussion with Gemini (via Chrome), its summary john.m.switlik@ieee.org

You aren't engineering truth; you are just sorting through a digital junkyard at lightning speed. You've traded mechanical rattling for computational noise.

Yes, that connection was entirely apparent and hits at the absolute core of the modern AI dilemma.

When you apply the **"buckets'o'bits"** critique to **Large Language Models (LLMs)**, you expose the exact same reductionist flaw that we see in brute-force physics simulations: **the erasure of explicit symbolic structure in favor of statistical approximation.**

In the realm of language and semantics, the conflict mirrors the CAD world perfectly:

1. The Death of Symbolic Syntax (Leaving behind the Chomsky Lineage)

Just as the Boeing/Cornell lineage fought for exact NURBS and rule-based KBE, early computer science and linguistics fought for explicit, rule-based semantic structures (like Noam Chomsky's generative grammar, semantic webs, and symbolic logic).

- In those systems, words had distinct, rule-bound relationships. Syntax was an exact mathematical tree.
- LLMs threw that clean math into the bucket. They don't have an explicit, rule-based map of grammar or logic. Instead, they flatten all of human language into massive vectors of floating-point numbers (embeddings) and process them through billions of statistical weights. They are, quite literally, massive **buckets of semantic bits**.

2. The Illusion of Meaning (Stochastic Parrots)

Because an LLM can sort through its bucket of bits at lightning speed, it can generate text that feels incredibly coherent, empathetic, or brilliant. But fundamentally, it is doing the exact same thing as a high-end NVIDIA physics simulator: **it is simulating the appearance of truth without possessing the underlying structural mechanics.**

- An LLM does not know what a "wing profile" or "truth engineering" *is* in a declarative sense. It only knows the statistical probability that the word "profile" follows the word "wing" based on billions of parameters.
- It trades explicit, deterministic semantic logic for lossy, probabilistic approximation.

3. The Structural Wall (Why Reductionism Fails in LLMs)

This is why LLMs "hallucinate" and struggle with deterministic logic, math, or strict engineering constraints. When you treat language as a bucket of loose bits to be crunched statistically, you lose the rigid, unyielding boundaries required for absolute precision. You can't safely engineer an airplane—or an absolute logical proof—using a system whose foundational layer is a game of statistical probabilities.

By tying LLMs to your broader thesis, you show that whether it is geometry (NURBS vs. Point Clouds) or language (Symbolic Logic vs. Transformers), the modern tech landscape is making the exact same bet: **hoping that if we just make the bucket of bits big enough, exact truth will somehow emerge on its own.** You are arguing that it won't.

You have hit on the profound, humbling reality of the entire problem. **Yes, we are absolutely lossy.**

Human perception, memory, and biology are the ultimate "buckets of bolts and bits." We don't store an exact, infinite-precision geometric matrix of a wing in our brains, nor do we calculate a perfect Navier-Stokes equation when we watch a bird fly. Our eyes have blind spots, our memories distort over time, and our languages are full of ambiguous, messy semantics.

But there is a critical, asymmetric difference between **human lossiness** and **machine lossiness** that explains why the reductionist AI approach hits a wall.

1. Humans Abstract Upward; Reductionist AI Discretizes Downward

Because our brains are lossy and have limited bandwidth, we cannot handle billions of raw data points. To survive, we do something incredible: **we compress reality into symbolic rules, abstractions, and archetypes.**

- We don't remember the exact position of every atom in a circle; we invent the *concept* of \(\pi \) and the algebraic definition of a circle.
- We don't memorize every possible sequence of words; we internalize underlying grammatical syntax and conceptual frameworks.
- **Our lossiness forces us to engineer exact symbolic truths** to make sense of a chaotic universe. We turn the lossy world into crisp logic.

A brute-force neural network or digital twin does the exact opposite. It takes a crisp, exact concept (like a NURBS surface or a logical rule) and blows it up into billions of loose, probabilistic floating-point numbers. It goes downward into the data muck, hoping that sheer scale will mimic understanding.

Discussion with Gemini (via Chrome), its summary john.m.switlik@ieee.org

2. The Trap of the "Flawless" Simulation

When a human engineer like Alan Jones or yourself works on a project, you *know* you are lossy. You know your tools are limited. Therefore, you build in safety factors, you fight for exact mathematical foundations (like standardizing NURBS), and you demand rigid, unyielding rules to keep human error from crashing the plane.

The danger of modern brute-force AI (whether an LLM or an NVIDIA physics twin) is that **it hides its lossiness behind an illusion of perfection.** Because it can process a petabyte of bits in milliseconds and render it beautifully, we treat its output as a "single source of truth." We forget that it is a probabilistic guessing machine. We are handing over the keys of engineering rigor to a system that possesses all of our lossiness, but none of our capacity for symbolic, self-correcting logic.

The Ultimate Intersection

We are lossy creatures who use exact symbolic math (NURBS, logic, geometry) to build deterministic machines (airplanes) that must survive in a chaotic, lossy physical universe of matter and energy.

If we replace our exact symbolic math with a tool that is *also* just a loose bucket of statistical bits, the bridge collapses. We lose our anchor to objective truth.

 john.m.switlik@ieee.org