Monte Carlo Methods for Integration of Fractal Morphic Energy Number Reductionist Mappings to the, "Reals."

by Parker Emmerson, with thanks to Jehovah, the Living One

Introduction:

Concrete: Inasmuch as I have criticized the necessity, conception, utility, functionality, validity, and actual existence of the so - called, "Real," numbers via the homomorphic, topological methods described in my works, , it is still possible to reduce the fluidity of the symbol game of Quasi - Quanta symbolic entanglement of Energy Numbers for the sake of demonstrating potential graph forming and calculator applications . It is arguable the the, "Real," numbers are not really even real . While if it were up to me, I' d call them something else, it seems the, "consensus," will remain rigid and wrong in their terminology as usual in this realm . The point of this paper is not to show you how good I am at programming, I' m not . The point of this paper is to show you that the beauty and imagination of the functional, homological, topological calculus of Energy Numbers and fractal morphisms can be reduced by numerical methods into a graphble relationship by one or more modal interpretations . I' m sure one more advanced in programming would be able to substantially interpret Energy Numbers in more complex and meaningful patterns and make more advanced graphing analogs to their functionality . Just, the premise of doing so, while most likely flawed, could be potentially fruitful in the sense that we get to generate graphing calculator diagrams in potentially novel ways .

We start by noting the formality of the

function $\tilde{\star} R = \int_0^{\infty} \left(\left(\Psi \cdot \sin^2 \theta \right) + n^{m-1} \right) \cdot \tan t \tan^2 \theta \prod_{\Lambda} d\, h\, d\, \theta$,

Thus, the integral can be performed,

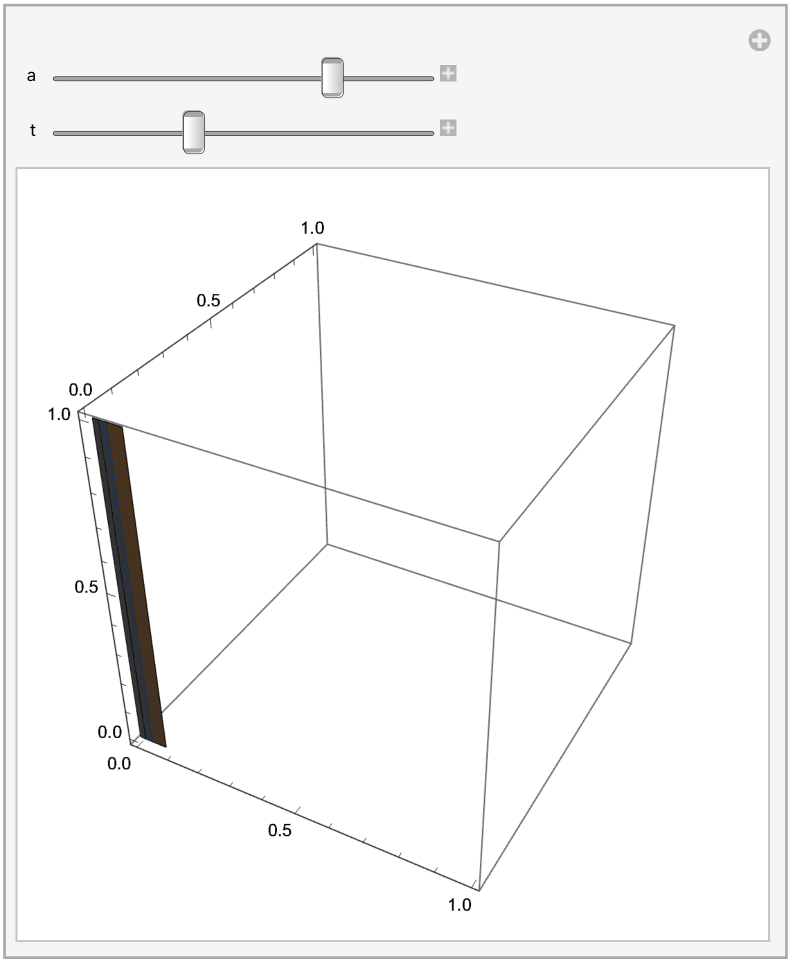
```
Integrate[((Ψ Sin[θ]^2) + n^(m-1)) Tan[t] Tan[θ]^2, {θ, 0, a}]
```

$$-\frac{\left(4\, a\, n^m + 6\, a\, n\, \Psi - n\, \Psi\, \text{Sin}[2\, a] - 4\left(n^m + n\, \Psi\right) \text{Tan}[a]\right) \text{Tan}[t]}{4\, n} \quad \text{if } 2\, \text{Re}[a] \le \pi \,||\, a \notin \mathbb{R}$$

which is graphable :

In[∘]:=
```
Manipulate[ContourPlot3D[-((4 a n^m + 6 a n Ψ - n Ψ Sin[2 a] - 4 (n^m + n Ψ) Tan[a]) Tan[t])/(4 n),
  {m, 0, 1}, {n, 0, 1}, {Ψ, 0, 1}], {a, 0, 1}, {t, 0, 1}]
```

Out[∘]=



Programs:

Upon initial attempts to run the monte carlo simulation on the integral, I was met with a number of problems with recursion :

In[ ]:=
```
mcR = -1 + Sum[D[1, x^j] / (Tan[θ.h[n]] - Ψ), {j, 1, n}];
EData =
 {n, l} ↦ ((Exp[b^(μ - ξ)] / (Exp[n^m] - Exp[l^m])) + Exp[-(1/m) h[n]] * Exp[Tan[t]]);
n1 = 2;
l1 = 0;
θ1 = 0; Ξ1 = 1; Ps1 = 1; b1 = 2; μ1 = 1; ξ1 = 0;
MonteCarloData1 = Reap[Do[θ1 = θ1 + RandomReal[];
      Ξ1 = Ξ1 * RandomReal[];
      Ps1 = Ps1 * RandomReal[];
      b1 = b1 * RandomReal[];
      Ω1 = Ω1 * RandomReal[{0, 1}];
      n1 = n1 + RandomInteger[{1, 10}];
      l1 = l1 + RandomInteger[{1, 10}];
      hn = RandomInteger[{1, 10}];
      Sow[mcR ((b1^(μ1 - ξ1)) / (Tan[t]^2 * Sqrt[Product[h[n] - Ps1, {n, Λ}]])) *
        (Ω1 * EData[n1, l1])], 40]][[2, 1]];

barChart = BarChart[HistogramList[MonteCarloData1, 10][[2]],
   ChartLabels → Placed[HistogramList[MonteCarloData1, 10][[1]], Above],
   AxesLabel → {Style["x", 14, Bold], Style["N", 14, Bold]}, PlotRange → All];

Show[barChart, PlotRange → Full]
```

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of 0.230294 Ω1.

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of Periodic`PeriodicSequencePeriod[−0.17407, n].

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of Periodic`PeriodicLibraryDump`res = Periodic`PeriodicLibraryDump`periodicSequenceHeadDecomposition[−0.17407 + 1. h[n], n, Plus, False].

General: Further output of $RecursionLimit::reclim2 will be suppressed during this calculation.

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of Periodic`PeriodicLibraryDump`res = Periodic`PeriodicLibraryDump`PeriodicSequenceHeadDecomposition[1. (−0.17407 + 1. h[n]), n, Plus, False].

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of RuleCondition[Periodic`PeriodicLibraryDump`res, FreeQ[Periodic`PeriodicLibraryDump`res, $Failed]].

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of Simplify`PWPresentQ[1. (−0.17407 + 1. h[n])] && ! Simplify`PWPresentQ[{{n, 1, Λ}}].

General: Further output of $RecursionLimit::reclim2 will be suppressed during this calculation.

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of Product`ProductPeriodicDump`res1 = Periodic`PeriodicSequenceDecompose[−0.17407 + 1. h[n], n, Plus].

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of RuleCondition[≪1≫].

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of Sum`PiecewiseSumProductDump`res = Sum`PiecewiseSumProductDump`productPiecewiseThread[−0.17407 + 1. h[n], {n, 1, Λ}].

General: Further output of $RecursionLimit::reclim2 will be suppressed during this calculation.

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of
Product`ProductPeriodicDump`res1 = Product`ProductPeriodicDump`PeriodicPower[1. (–0.17407 + 1. h[n]), {n, 1, Λ}].

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of
If[FreeQ[Product`ProductPeriodicDump`res1, $Failed], Throw[Product`ProductPeriodicDump`res1]].

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of
Product`ProductPeriodicDump`res1 = Product`ProductPeriodicDump`PeriodicPlus[1. (–0.17407 + 1. h[n]), {n, 1, Λ}].

General: Further output of $RecursionLimit::reclim2 will be suppressed during this calculation.

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of
Message[Message::msgl, Hold[{$RecursionLimit::reclim2, $RecursionLimit::reclim2, $RecursionLimit::reclim2, General::stop}]].

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of {$RecursionLimit::reclim2}.

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of
{OutputStream[Name: stdout Unique ID: 1]}.

General: Further output of $RecursionLimit::reclim2 will be suppressed during this calculation.

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of
Message[Message::msgl, Hold[{$RecursionLimit::reclim2, $RecursionLimit::reclim2, $RecursionLimit::reclim2, General::stop}]].

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of {$RecursionLimit::reclim2}.

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of
{OutputStream[Name: stdout Unique ID: 1]}.

General: Further output of $RecursionLimit::reclim2 will be suppressed during this calculation.

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of
Message[Message::msgl, Hold[{$RecursionLimit::reclim2, $RecursionLimit::reclim2, $RecursionLimit::reclim2, General::stop}]].

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of {$RecursionLimit::reclim2}.

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of
{OutputStream[Name: stdout Unique ID: 1]}.

Skeleton Key

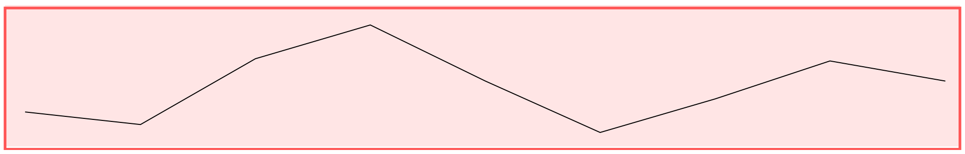
In[∘]:=
```
In[15] :=
 Plot[Evaluate[Integrate[((Ψ Sin[θ]^2) + n^(m - 1)) Tan[t] Tan[θ]^2, {θ, 0, x}]],
  {x, 0, π}, PlotRange → All]

Out[15] = Graphics[{{RGB[0.368417, 0.506779, 0.709798],
    Line[{{0., 0.}, {0.785398, -0.0854466}, {1.5708, 0.362941},
      {2.35619, 0.593001}, {3.14159, 0.211337}, {3.92699, -0.139387},
      {4.71239, 0.0889663}, {5.49779, 0.347876}, {6.28319, 0.211337}}]}}]
```

SetDelayed: Tag In in In[15] is Protected.

Out[∘]=
```
$Failed
```

Set: Tag Out in %15 is Protected.

Out[∘]=



The recursion problems were overcome with :

In[∘]:=
```
mcR = -1 + Sum[D[1, x^j] / (Tan[θ.h[n]] - Ψ), {j, 1, n}];
EData =
  {n, l} → ((Exp[b^(μ - ξ)] / (Exp[n^m] - Exp[l^m])) + Exp[-(1/m) h[n]] * Exp[Tan[t]]);

MonteCarlo[f_, {xmin_, xmax_}, {ymin_, ymax_},
  {θmin_, θmax_}, {nmin_, nmax_}, Ωu_] := Module[{x, y, θ, n},
  Sample[f, {xmin, xmax}, {ymin, ymax}, {θmin, θmax}, {nmin, nmax}, Ωu] * Integrate[
    f * Ωu, {x, xmin, xmax}, {y, ymin, ymax}, {θ, θmin, θmax}, {n, nmin, nmax}]]

MonteCarloData = Reap[Do[θ1 = θ1 + RandomReal[{0, 2 Pi}];
      Ξ1 = Ξ1 * RandomReal[];
      Ps1 = Ps1 * RandomReal[];
      b1 = b1 * RandomReal[];
      Ω1 = Ω1 * RandomReal[{0, 1}];
      n1 = RandomInteger[{1, 10}];
      l1 = RandomInteger[{1, 10}];
      hn = RandomInteger[{1, 10}];
      SampleData1 = mcR ((b1^(μ1 - ξ1)) /
          (Tan[t]^2 * Sqrt[Product[h[n1] - Ps1, {n1, Λ}]])) * (Ω1 * EData[n1, l1]);
      TimeData1 = Round[AbsoluteTime[] - StartTime, 0.1];
      Sow[SampleData1, TimeData1], 40]][[2, 1]];

barChart = BarChart[HistogramList[MonteCarloData, 10][[2]],
   ChartLabels → Placed[HistogramList[MonteCarloData, 10][[1]], Above],
   AxesLabel → {Style["x", 14, Bold], Style["Time (s)", 14, Bold]}, PlotRange → All];

Show[barChart, PlotRange → Full]
```

General: x^j is not a valid variable.

General: x^j is not a valid variable.

General: $x^{Sum`FiniteSumDump`l}$ is not a valid variable.

General: Further output of General::ivar will be suppressed during this calculation.

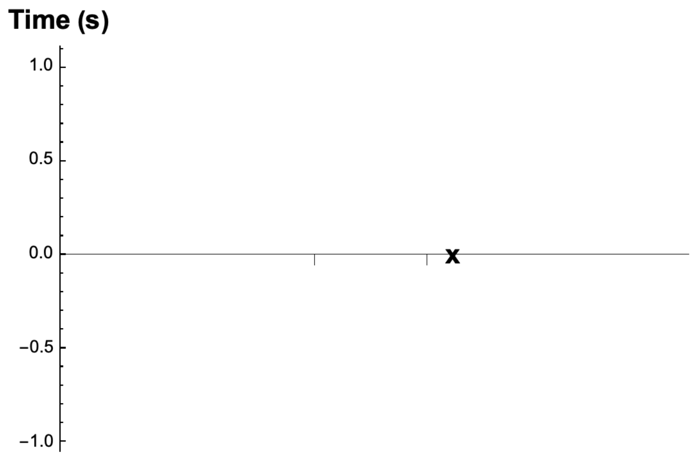
$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of 4.14254 + θ1.

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of 0.139721 Ξ1.

$RecursionLimit: Recursion depth of 1024 exceeded during evaluation of 0.136079 Ps1.

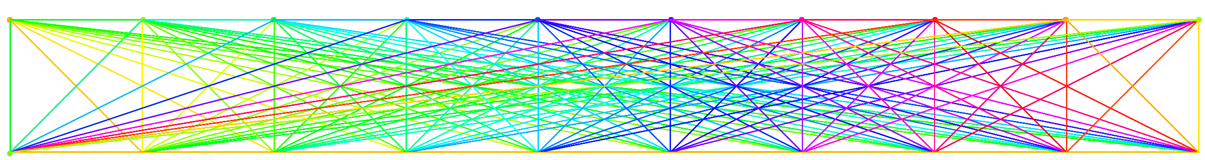
General: Further output of $RecursionLimit::reclim2 will be suppressed during this calculation.

Out[∘]=

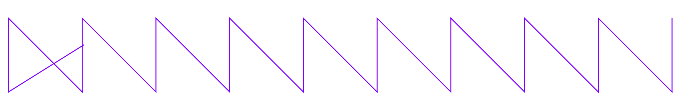


In[∘]:=
```
tVec = {0., 0.785398, 1.5708, 2.35619,
   3.14159, 3.92699, 4.71239, 5.49779, 6.28319, 7.06858};

rVec = Table[If[i == 0, 0., 0.785398], {i, 0, 9}];

cang = Tuples[{tVec, rVec}];
cycol = Append[cang, {0., 0.}];

Graphics[
 Table[{Hue[(4 * i + j + 2) / (4 * Length[cycol] + 2)], Line[{cycol[[i]], cycol[[j]]}]},
  {i, 1, Length[cycol]}, {j, i + 1, Length[cycol]}]]

dydxVec = Table[{If[i == 0, 0., 0.8], If[i == 0, 0., 0.5]}, {i, 0, 9}];

cydcol = PairwiseSum[cang, dydxVec];

Graphics[
 Table[{Hue[(4 * i + j) / (4 * Length[cydcol])], Line[{cydcol[[i]], cydcol[[j]]}]},
  {i, 1, Length[cydcol]}, {j, i + 1, Length[cydcol]}]]
```

Out[∘]=

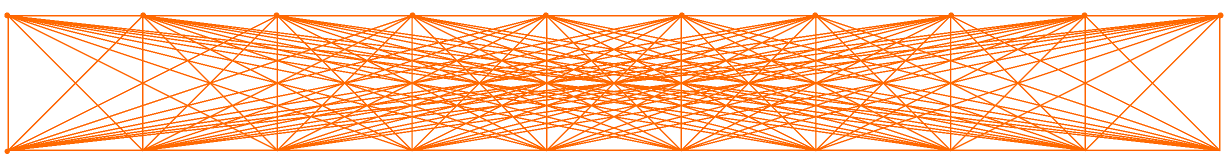


Out[∘]=

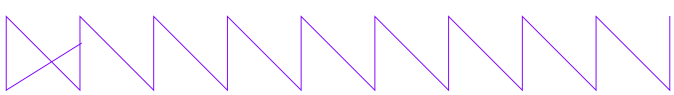


```
In[∘]:= tVec = {0., 0.785398, 1.5708, 2.35619,
    3.14159, 3.92699, 4.71239, 5.49779, 6.28319, 7.06858};

rVec = Table[If[i == 0, 0., 0.785398], {i, 0, 9}];

cang = Tuples[{tVec, rVec}];
cycol = Append[cang, {0., 0.}];

Graphics[Table[{Hue[((√(-1.1294090667581471`*^18 i +
            8.987551787368176`*^16 i^2 + 3.5481432270250993`*^18 Sin[j]^2)) /
        (√(-12.566370614359172` i + i^2 + 39.47841760435743` Sin[j]^2))) /
      (4 * Length[cycol] + 2)], Line[{cycol[[i]], cycol[[j]]}]},
   {i, 1, Length[cycol]}, {j, i + 1, Length[cycol]}]]

dydxVec = Table[{If[i == 0, 0., 0.8], If[i == 0, 0., 0.5]}, {i, 0, 9}];

cydcol = PairwiseSum[cang, dydxVec];

Graphics[
 Table[{Hue[(4 * i + j) / (4 * Length[cydcol])], Line[{cydcol[[i]], cydcol[[j]]}]},
  {i, 1, Length[cydcol]}, {j, i + 1, Length[cydcol]}]]
```

Out[∘]=



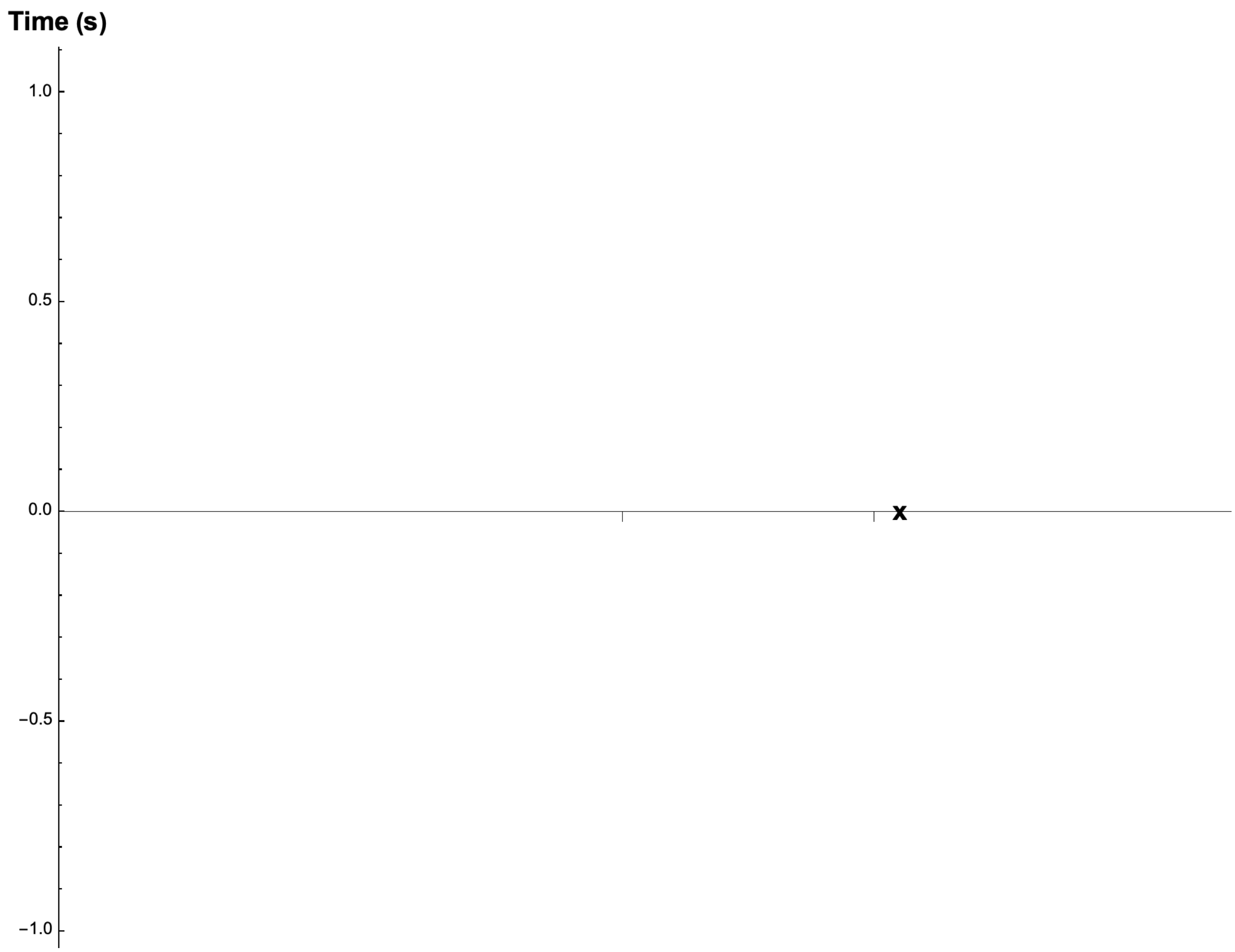
Out[∘]=



Treasure Map

Interpretation :

From the graph, and the zooming in on the graph, we can see that the mark, x, marks a spot that is perpetually, immeasurably close to the line, but sadly, not exactly on the line . This is the dilemma of quantum mechanics, essentially . It is evidentiary of a misconception within many commonly accepted functions of standard calculi in the literature . 1/∞ is too often interpreted as 0. The 1/∞ is not on the line, but infinitessimally close to the line, symbolically, representativly indicated here in this graph by Mathematica . As we zoom in, the spot marked, x, will ever approach exactness with the line, but will never actually be on the line. I would argue that essentially, all statistical interpretations of atomistic, quark, so called, "quantum," phenomena are actually either 1) poor descriptions of the phenomenon being measured due to inadequate measurements, 2) the equations are accurately describing the phenomenon, and the phenomenon of the so called, "material universe," is simply an imprecise simulation of a more precise, linguistic calculus or, 3) The perceptual conception of the phenomenon as "statistical," or, "probablistic," is actually recursively bringing about the bad math, and adjusting our perceptions through more advanced mathematical language will actually entangle the phenomenon into becoming more in line with the language used to describe it once it has been adapted to Energy Number theory .



References :

Morphic Topology of Numeric Energy : A Fractal Morphism of Topological Counting Shows Real Differentiation of Numeric Energy

https : // zenodo.org / record / 7 976 215