Probabilistic Geometry and Information Content

(an Introduction to Corob Theory)

ANPA Conference

Cambridge, UK

by Dr. Douglas J. Matzke

matzke@ieee.org

Aug 15-18, 2002

Abstract

The field of probabilistic geometry has been known about in the field of mathematics for over 50 years. Applying the unintuitive metrics in these high-dimensional spaces to the information arena is conceptually very tricky.

Pentti Kanerva developed computational uses in the mid 80s. Nick Lawrence also rediscovered similar results in the early 90s. His patented computational theory is called *Corobs*, which stands for *Cor*relational-Algorithm *Obj*ects. Recently, the link between quantum theory and Corob Theory was researched under DOD SBIR funding.

This presentation gives an overview of this field including the key concepts of how to implement useful computation, knowing that randomly chosen points are all a standard, equidistant apart (sqrt(N/6)) in a unit N-cube (as N>>3).

History of Geometric Probability

Also known as probabilistic geometry or integral geometry or "continuous combinatorics" and related to the study of invariant measures in Euclidean n-spaces (n+1 invariants in dim=n).

- Ninth Edition of Encyclopedia Britannica article by Crofton
- 1926 brochure by Deltheil
- 1962 book by Kendall and Moran
- 1988 book by Kanerva (Sparse Distributed Memory)
- 1998 Corob Patent & web site by Lawrence Technologies
- Modern books by Klain and Rota

Applications:

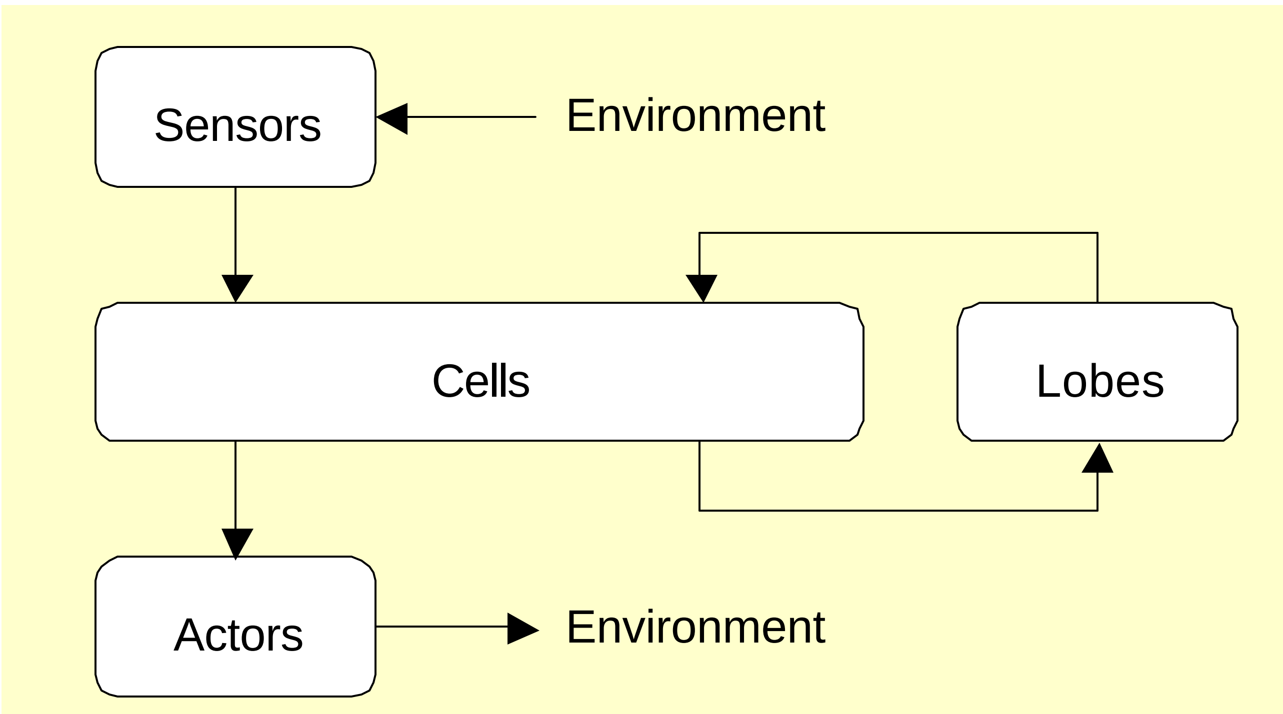
- Buffon Needle Problem (Barbier's solution)
- Crystallography, sampling theory, atomic physics, QC, etc
- Basically the study of actions of Lie groups to sym spaces

Motivated by Neurological Models

Corobs and Synthetic Organisms

Goal: See and do things **like** things previously seen and done

Nerve cells perform a random walk influenced by their input connections/structure. Therefore, *randomness is the key mechanism* of neuronal information.

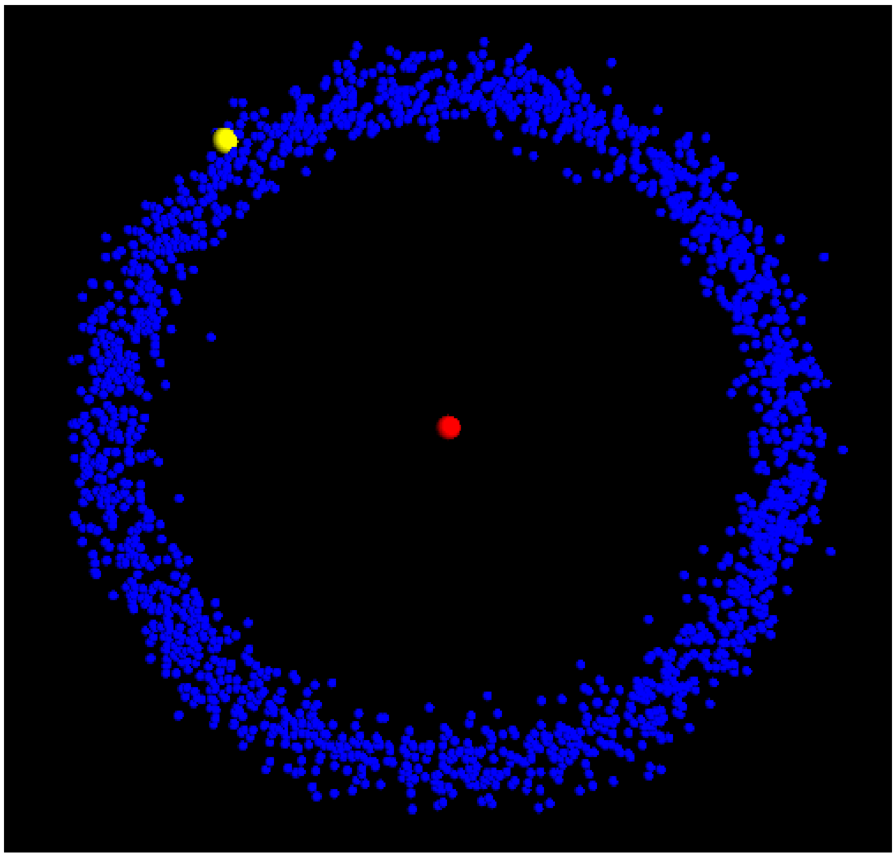


Lobes perform nearness metric computation

Discovering Geometric Probability

- With a *bounded* N-space (N>20)(real or complex)
- Asymmetric (0 to 1) or symmetric (+1 to -1) spaces
- With uniform distribution, randomly pick 2 points
- Compute or measure the Cartesian distance
- Repeat process for 1000s of random points
- Most distances will be a "*standard distance*", which for a unit N-cube is equivalent to $\sqrt{N/6}$ with *constant* standard deviation of $\sqrt{7/120} = 0.214$
- Analytical results produces same (www.LT.com) and patents issued (plus more pending).

Key Concept: Equidistance

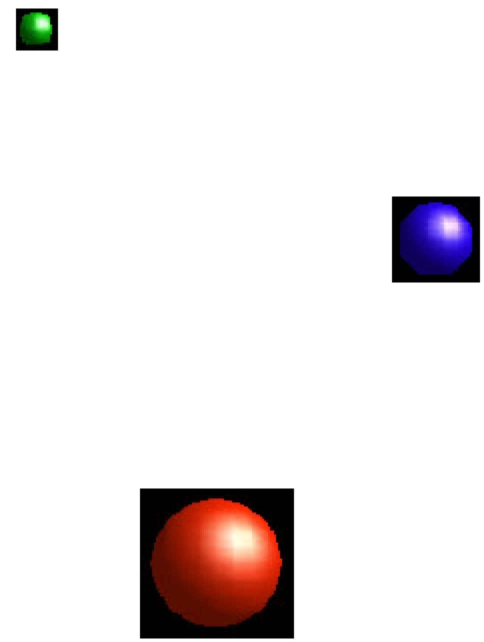


Points are *corobs* = Correlithm Objects

- All points tend to be the *same distance* from the red point.
- If the yellow point were at the center, the blue points would still be the same distance, and the red point would be among them!
- Distance is proportional to the probability of finding that point using a random process
- The more dimensions the larger the “standard distance” but the standard deviation remains a *constant*!

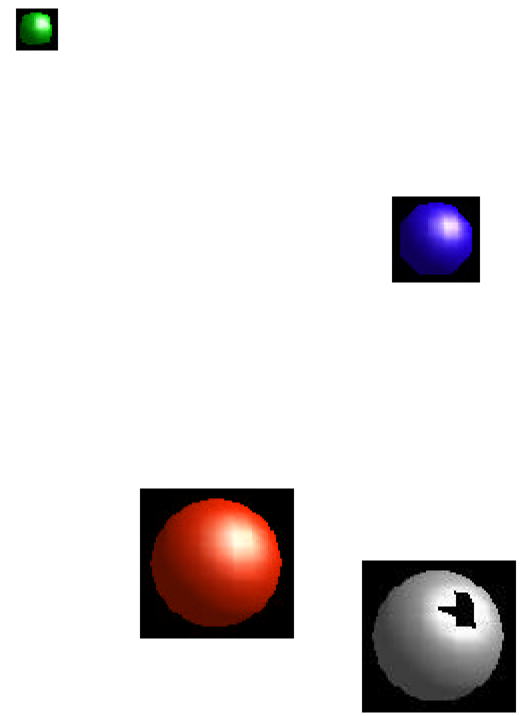
Corob Computing using Soft Tokens

- Data may be associated with Random Points.
- Here 3 data points are associated with Red, Green, and Blue.
- "Soft" because these tokens do not have sharp, brittle boundaries



Nearby Points are Similar

- The (unknown) Grey point is closest to the Red point.
- It is much more likely to be a “noisy” version of the Red point than the Green or Blue points, because it is closer.
- Hence, "soft tokens" or "corobs"
- Naturally robust probabilistic yet error correcting representation



Corob Language: Logic Example

```
import corob_lang        #depends on the corob language python module
define system.gates_and size=30
define subspace.Boolean False True  #randomly thrown soft tokens
```

```
define input.In1 Boolean pattern=(False False True True), degrade=20
define input.In2 Boolean pattern=(False True False True), degrade=20
define bundles.AndOut.OrOut:  #two outputs
       lobe(input=      (In1   In2   : Boolean Boolean), mode=quantize
            education =((False False : False   False)
                        (False True  : False   True)
                        (True  False : False   True)
                        (True  True  : True    True)) )
```

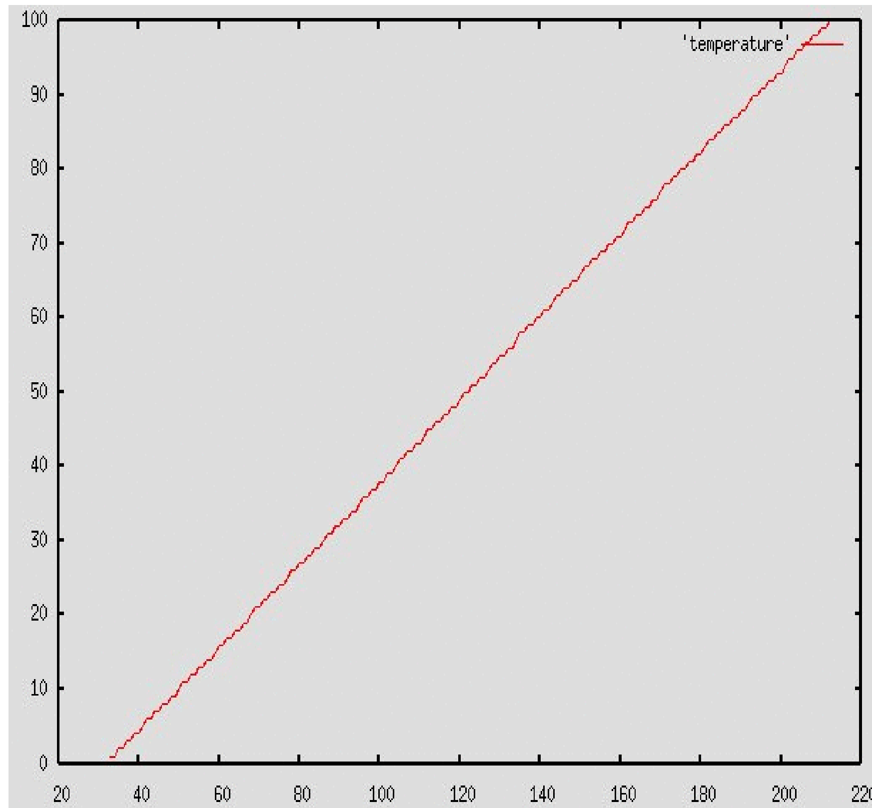
```
expect = "AndOut=(False False False True), OrOut=(False True True True)"
gates.validate_pattern(expect)  #inputs with  0% noise for time=0-3
gates.validate_pattern(expect)  #inputs with 20% noise for time=4-7
gates.validate_pattern(expect)  #inputs with 40% noise for time=8-11
gates.validate_pattern(expect)  #inputs with 60% noise for time=12-15
```

Sensor and Actor Example: Thermometer

```
import corob_lang
define system.temperature
define subset.fahrenheit: 32..212
define subset.centigrade: 0..100
define subspace.comfort: #subjective labels
   (freeze cool perfect warm hot sauna boil)
   drift=20  #string corobs 20% of standist
```

```
define input.thermometer: fahrenheit
define bundle.feeling:
       sensor(     mode=interpolate,
          input=(thermometer:comfort),
          education=((32:freeze),(50:cool),
            (77:perfect),(95:warm),(104:hot),
            (150: sauna),(212:boil)) )
define bundle.centigauge:  #play on words
       actor(      mode=interpolate,
         input=(feeling :centigrade),
         education=((freeze:0),(cool:10),
            (perfect:25),(warm:35),(hot:40),
            (sauna:66),(boil:100)) )
```

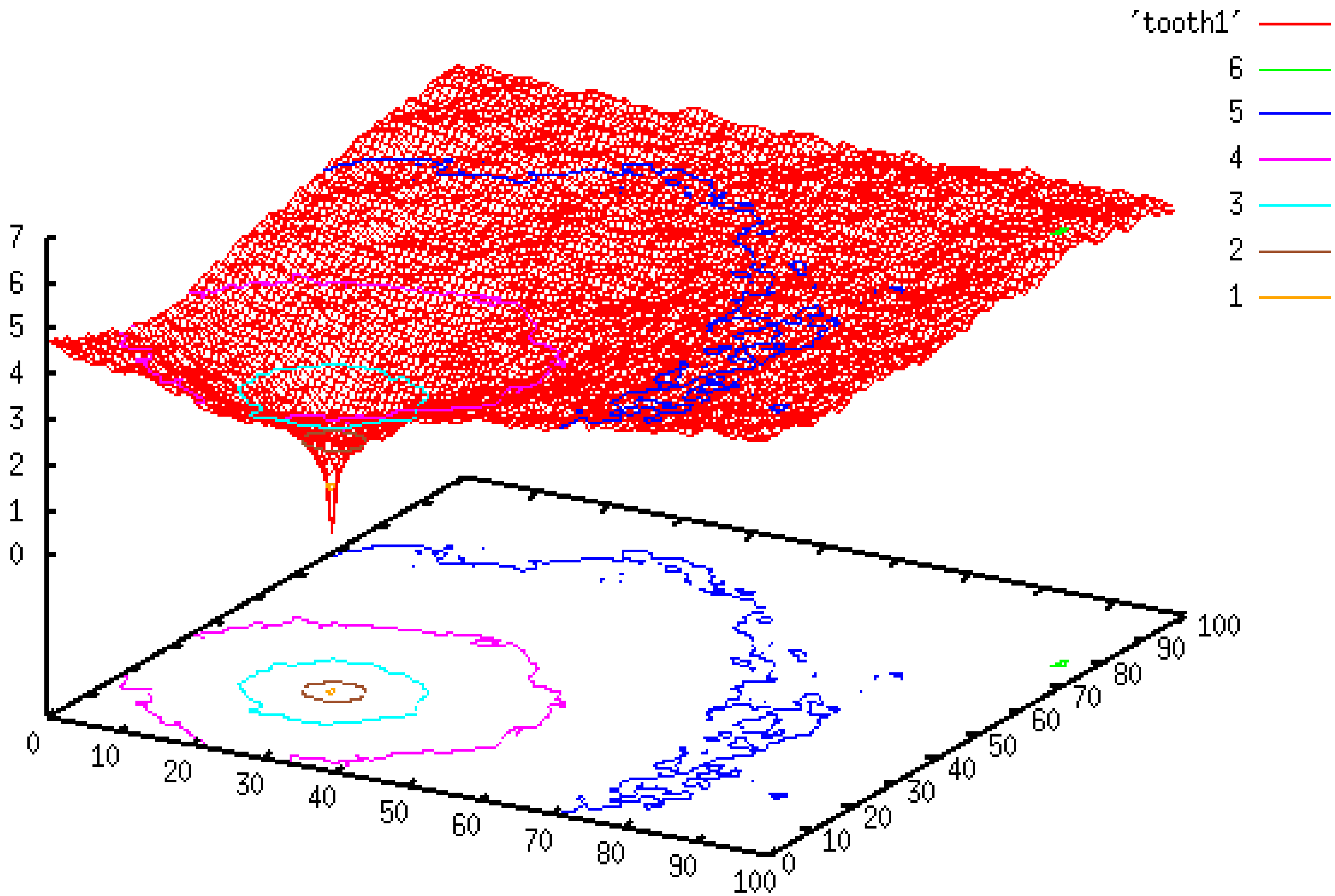
```
#validate, codegen, import & run 181 steps
temperature.run(steps=181)
```



Fahrenheit to Centigrade Conversion

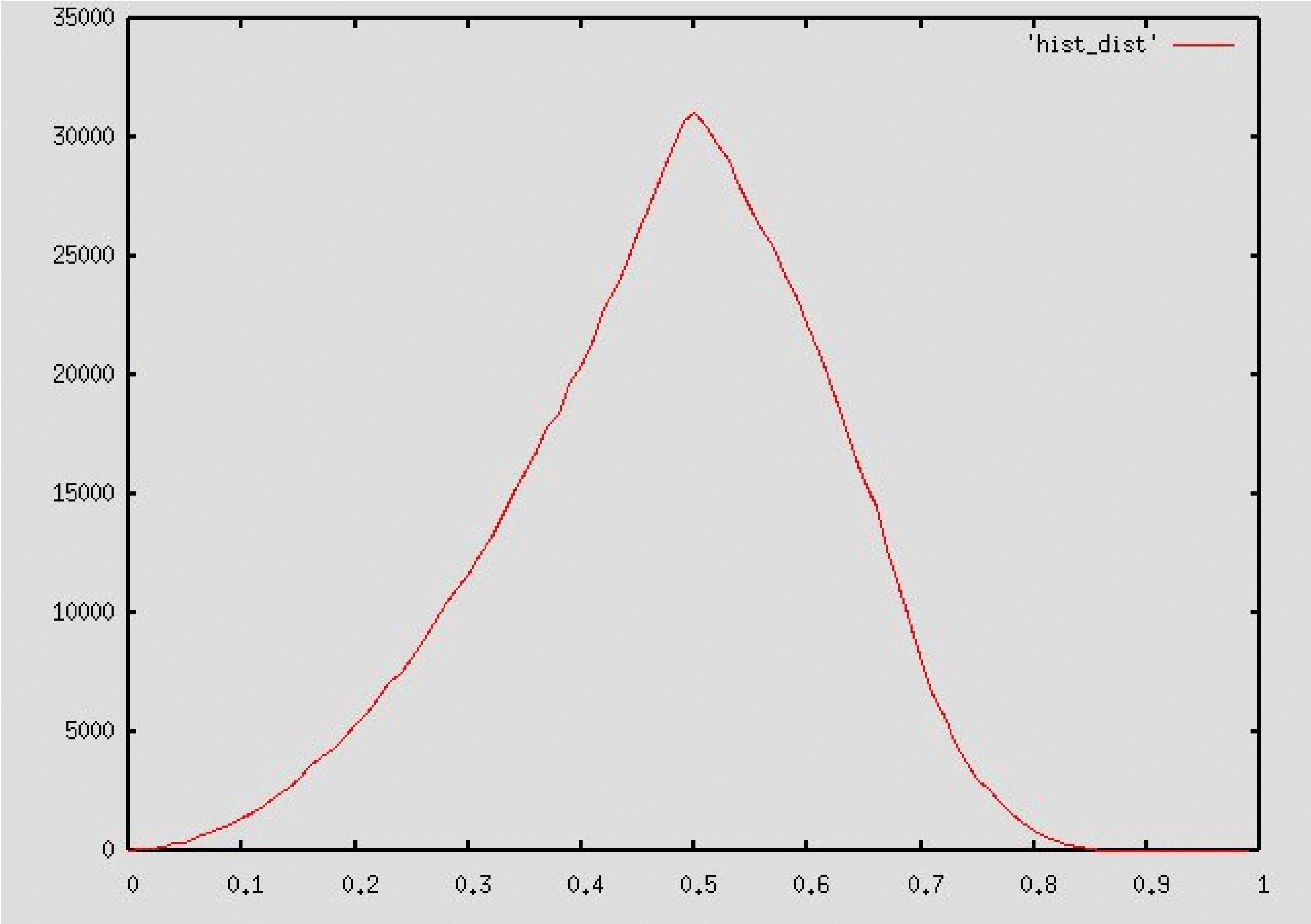
Embedding Continuous Geometries

Using string corobs and toothpicks (patents pending)

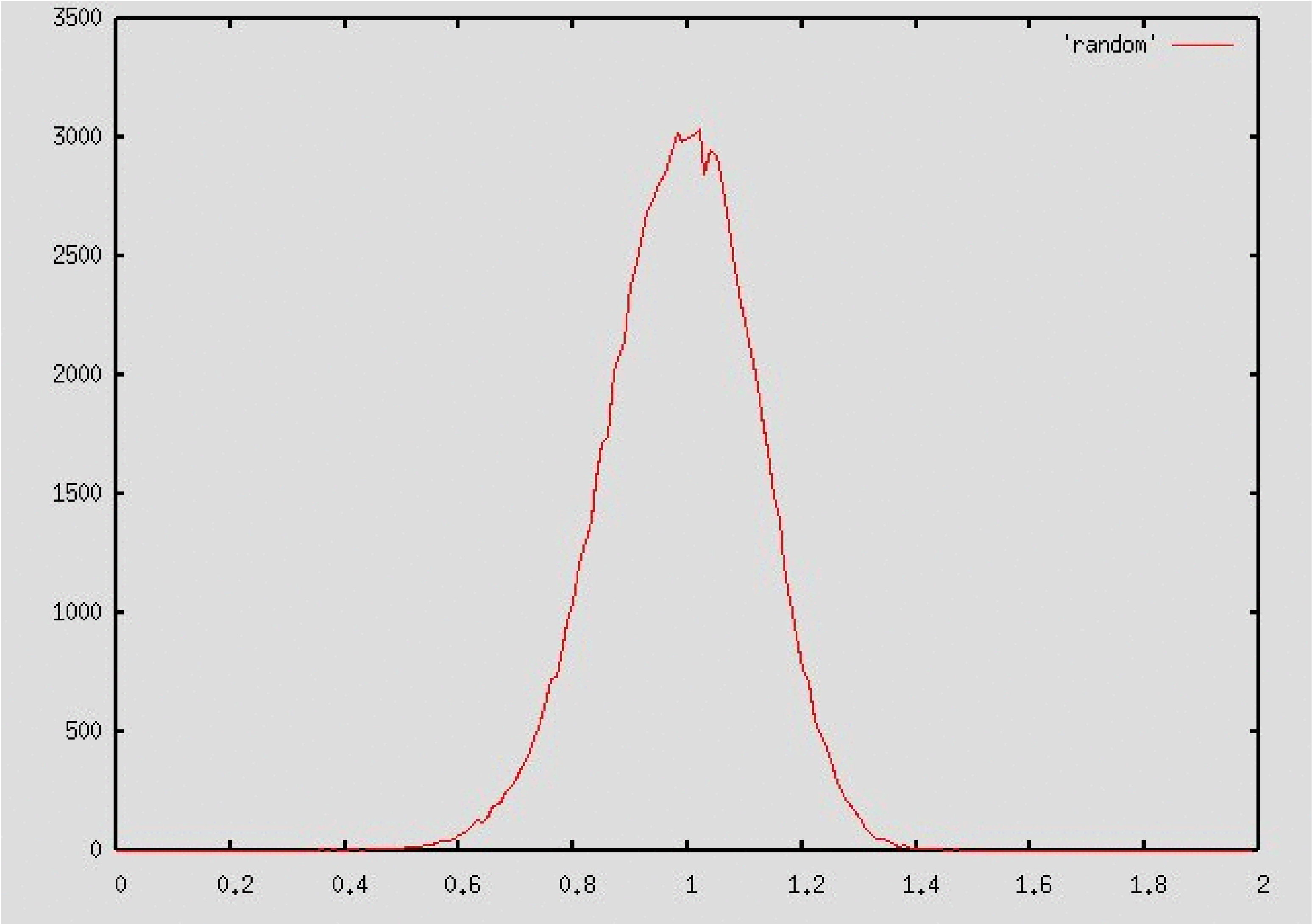


Topological Structure of N-Space

Distance Histogram for N=3

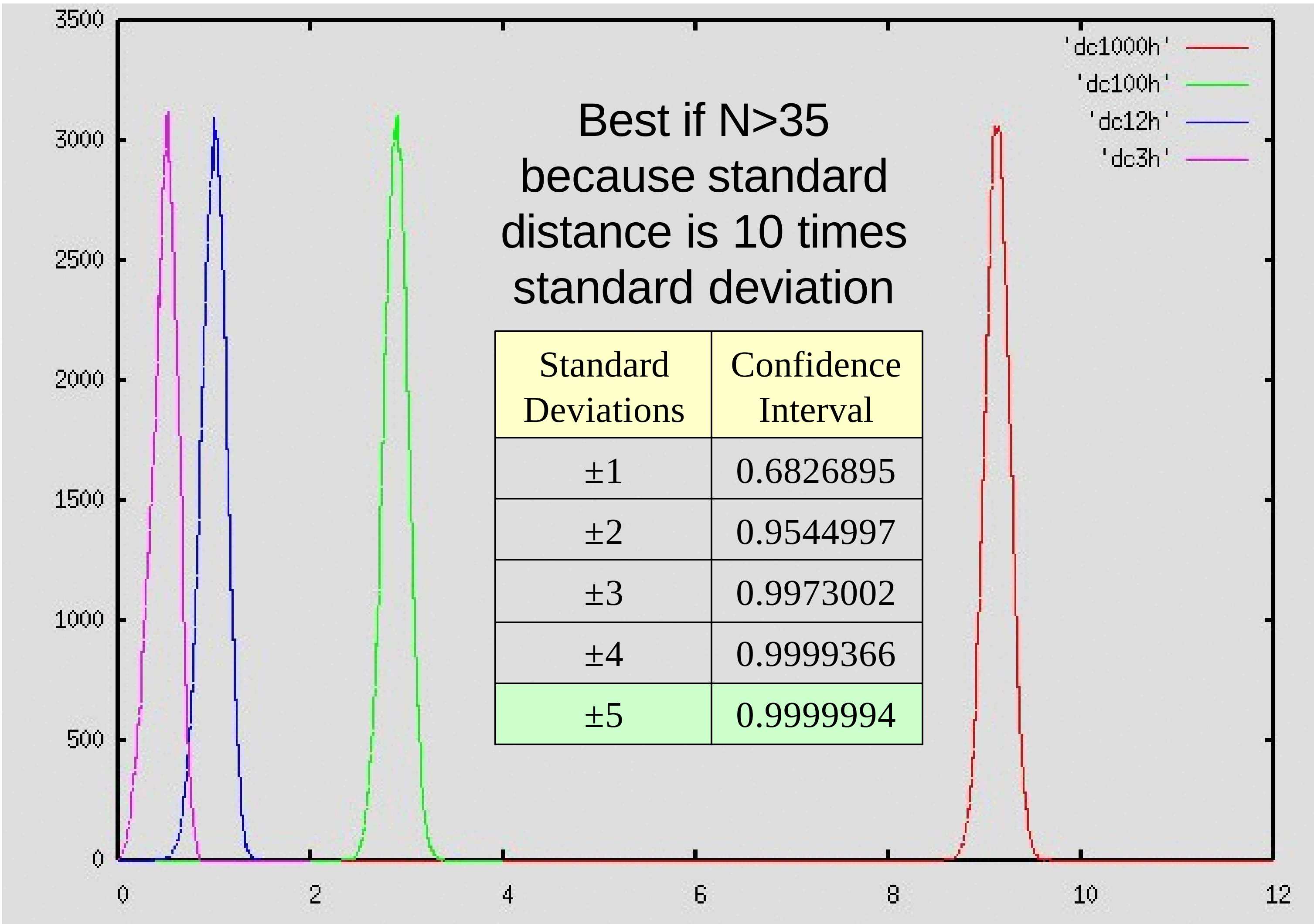


Distance Histogram for N=12



Effects of Constant Standard Deviation

Distance Histograms for N=3,12,100,1000



Best if N>35 because standard distance is 10 times standard deviation

Standard Deviations	Confidence Interval
±1	0.6826895
±2	0.9544997
±3	0.9973002
±4	0.9999366
±5	0.9999994

Distance and Information Content

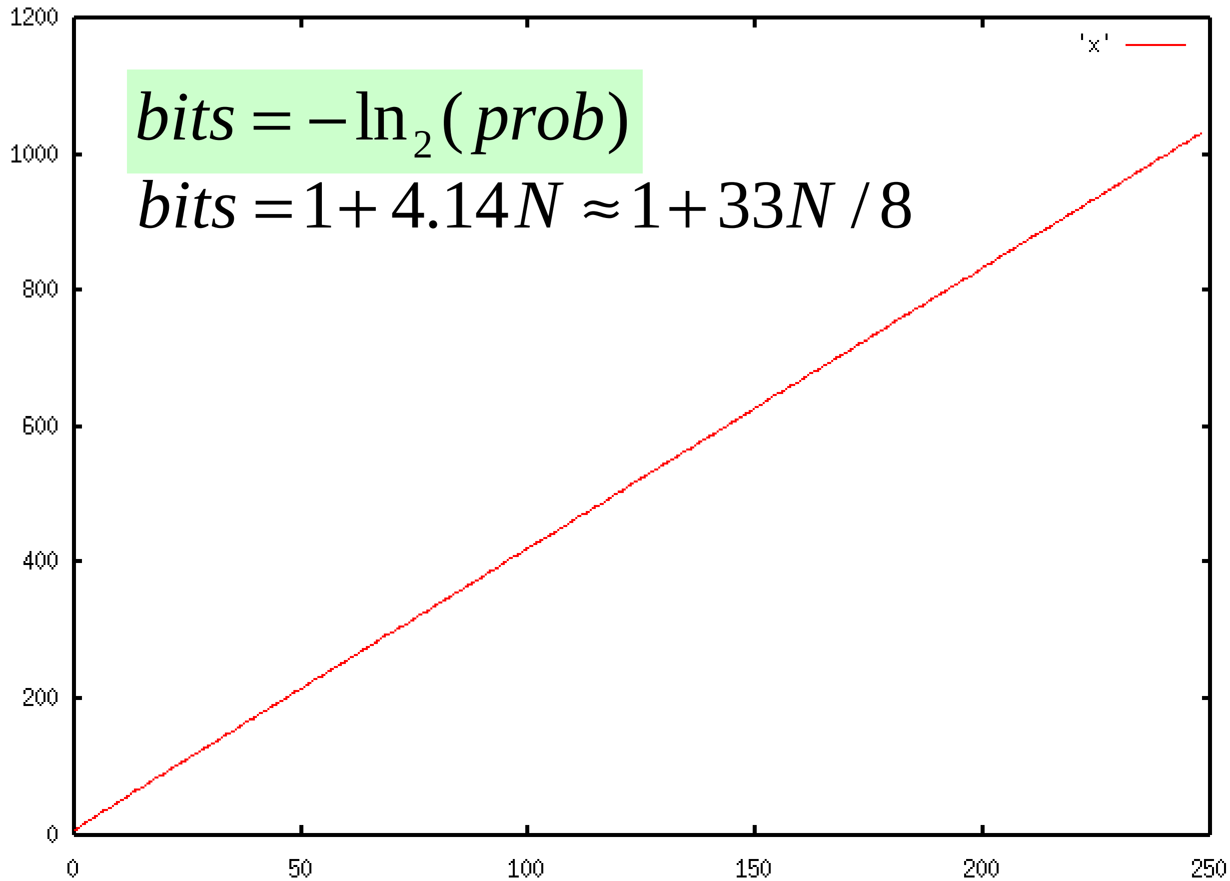
using standard distance normalized by standard deviation

$$prob = erfcc\left(\sqrt{N/6}/\sqrt{7/120} = \sqrt{20N/7}\right)$$

Just over 4 bits per dimension!

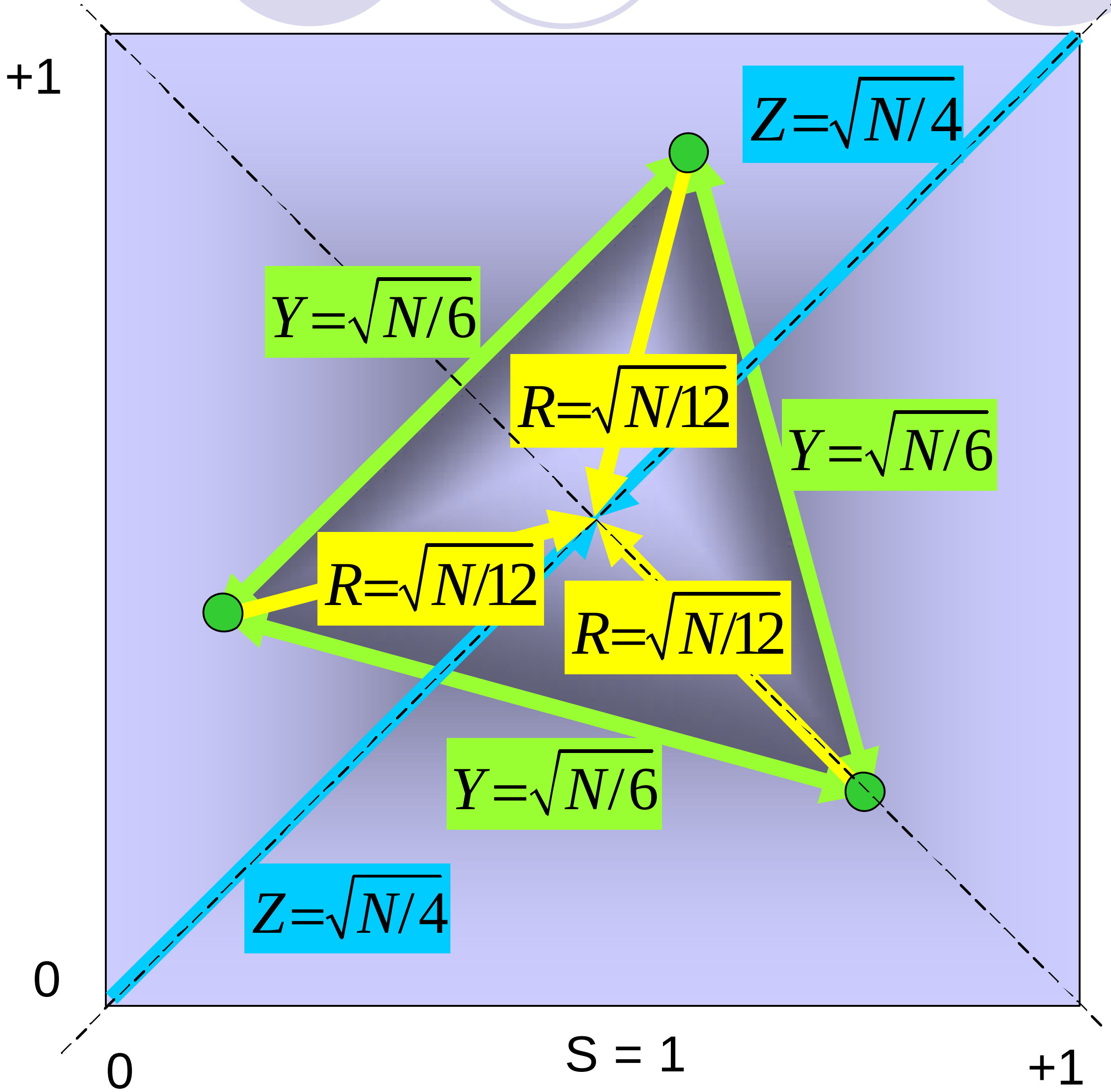
$$bits = -\ln_2(prob)$$

$$bits = 1 + 4.14N \approx 1 + 33N/8$$



Bit content of standard distance for N=1-250

Standard Distance and Standard Radius



Forms an N-dim tetrahedron or N-equihedron

(N-shell not N-sphere)

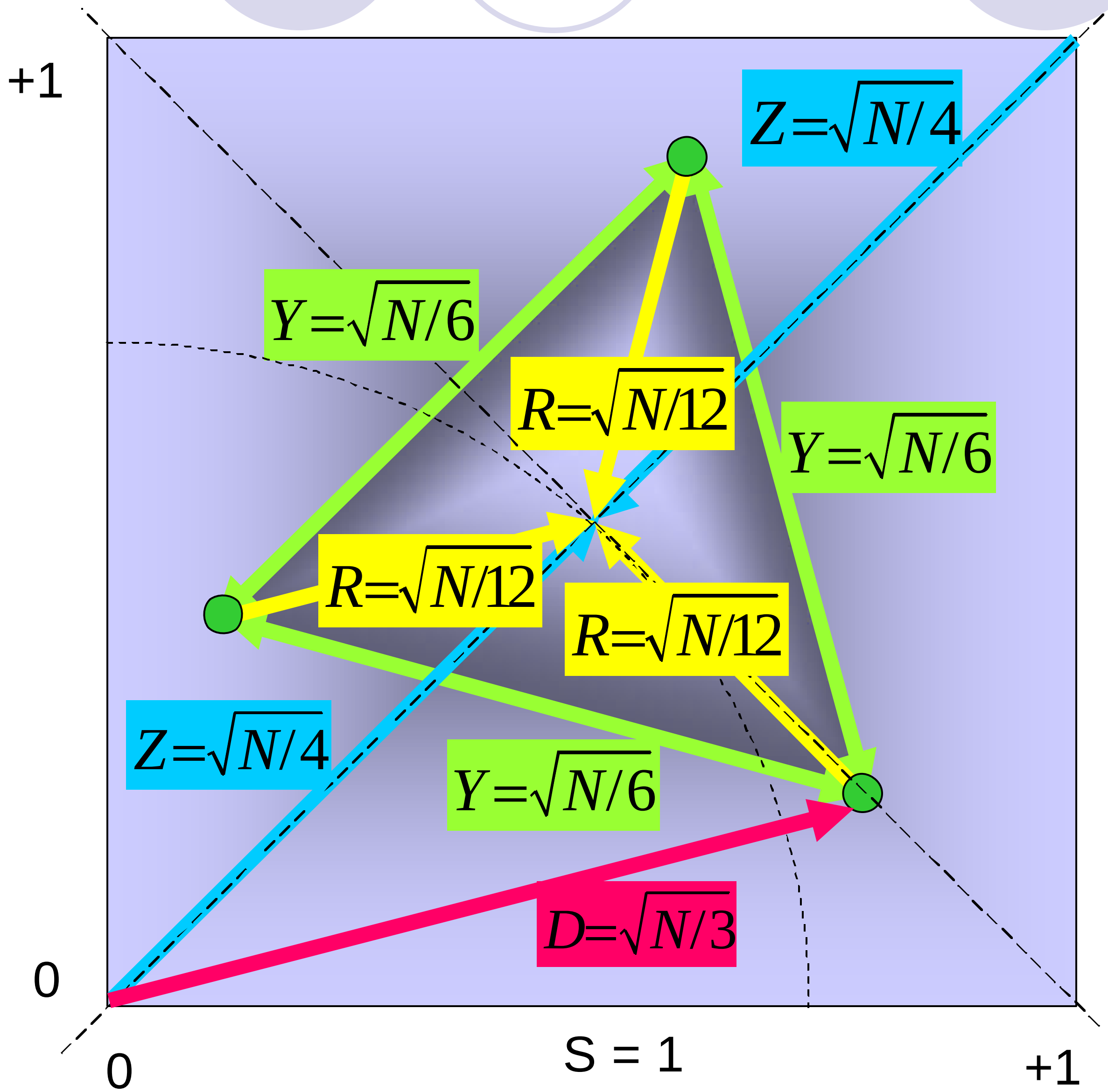
Space Center is point [.5 .5 …]

$$R \cong \sqrt{N/12}$$
$$Y \cong \sqrt{N/6}$$

$$M = \sqrt{N} = Z + Z$$
$$Z = M/2 = \sqrt{N/4}$$

8/15/2002 DJM

Distance from Corner to Random Point

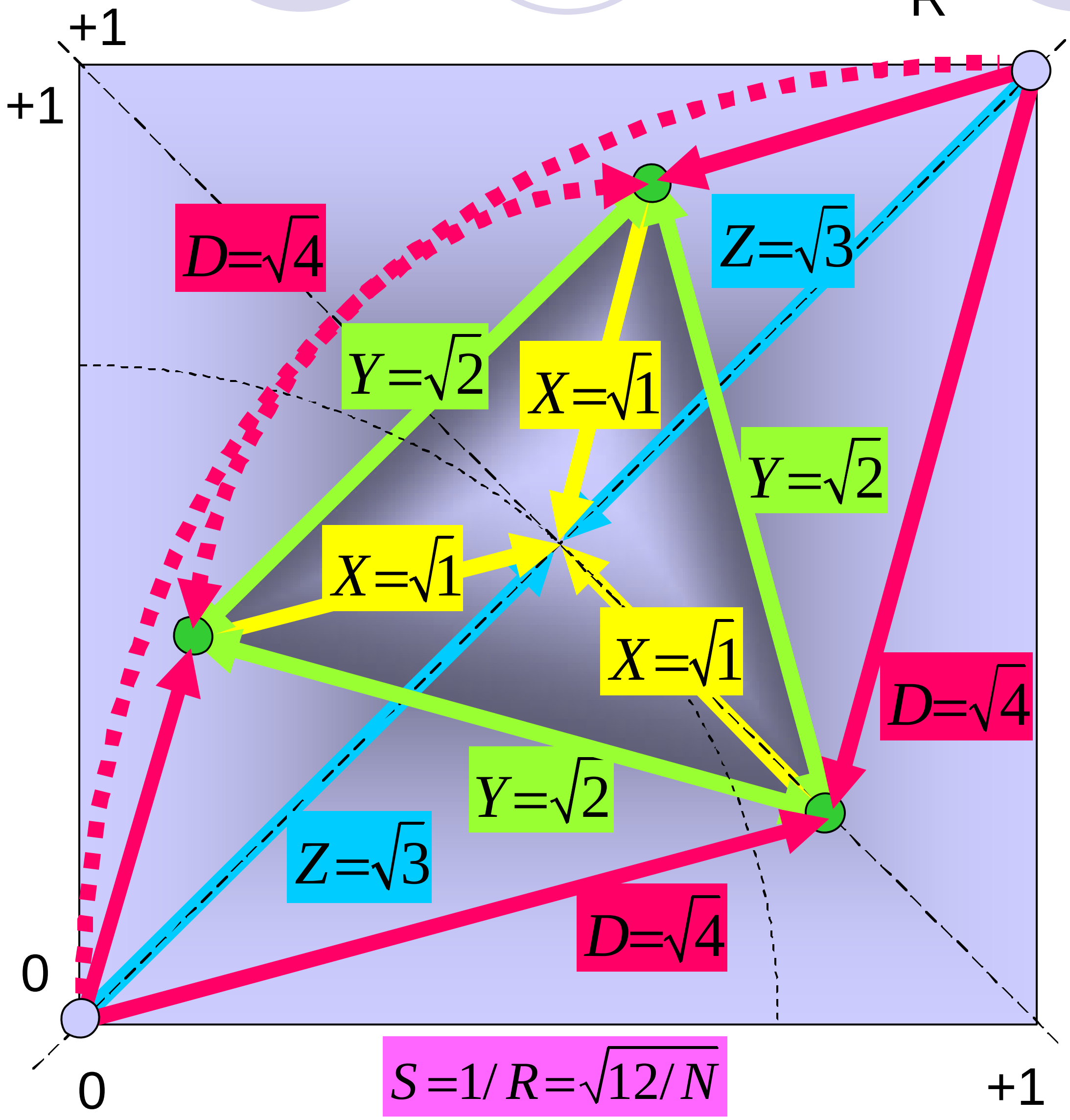


Distance from random corner to a random point is D=2R so call it the diameter D.

Notice equalities:
$Z^2 + R^2 = D^2$ and
$Z^2 + Z^2 = K^2$ where
$K \cong \sqrt{N/2}$ is the Kanerva distance of random corners

Normalized Distances Summary

for unit N_R-cube



$S=1/R=\sqrt{12/N}$

$R=\sqrt{N/12}$

$X=\sqrt{N/12}/R=\sqrt{1}=1$

$Y=\sqrt{N/6}/R=\sqrt{2}$

$Z=\sqrt{N/4}/R=\sqrt{3}$

$D=\sqrt{N/3}/R=\sqrt{4}$

$K=\sqrt{N/2}/R=\sqrt{6}$

$M=\sqrt{N}/R=\sqrt{12}$

$C=\sqrt{N/4}/R=\sqrt{3}$

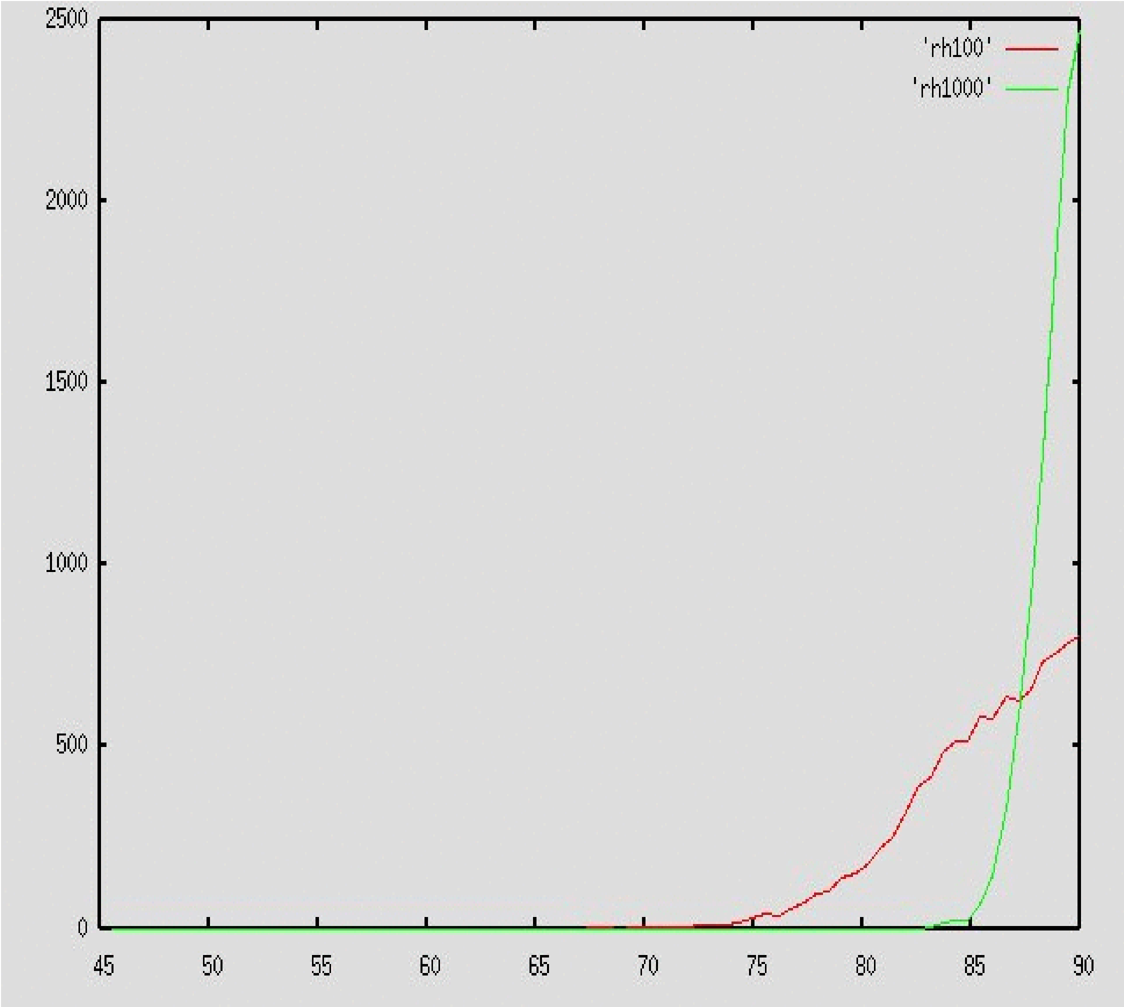
$Stdev_Y=\sqrt{7/120}/R=\sqrt{7/10N}$

$Stdev_R=\sqrt{1/60}/R=\sqrt{1/5N}$

8/15/2002 DJM

Standard Angles from Inner Product

Random points/tokens are nearly orthogonal

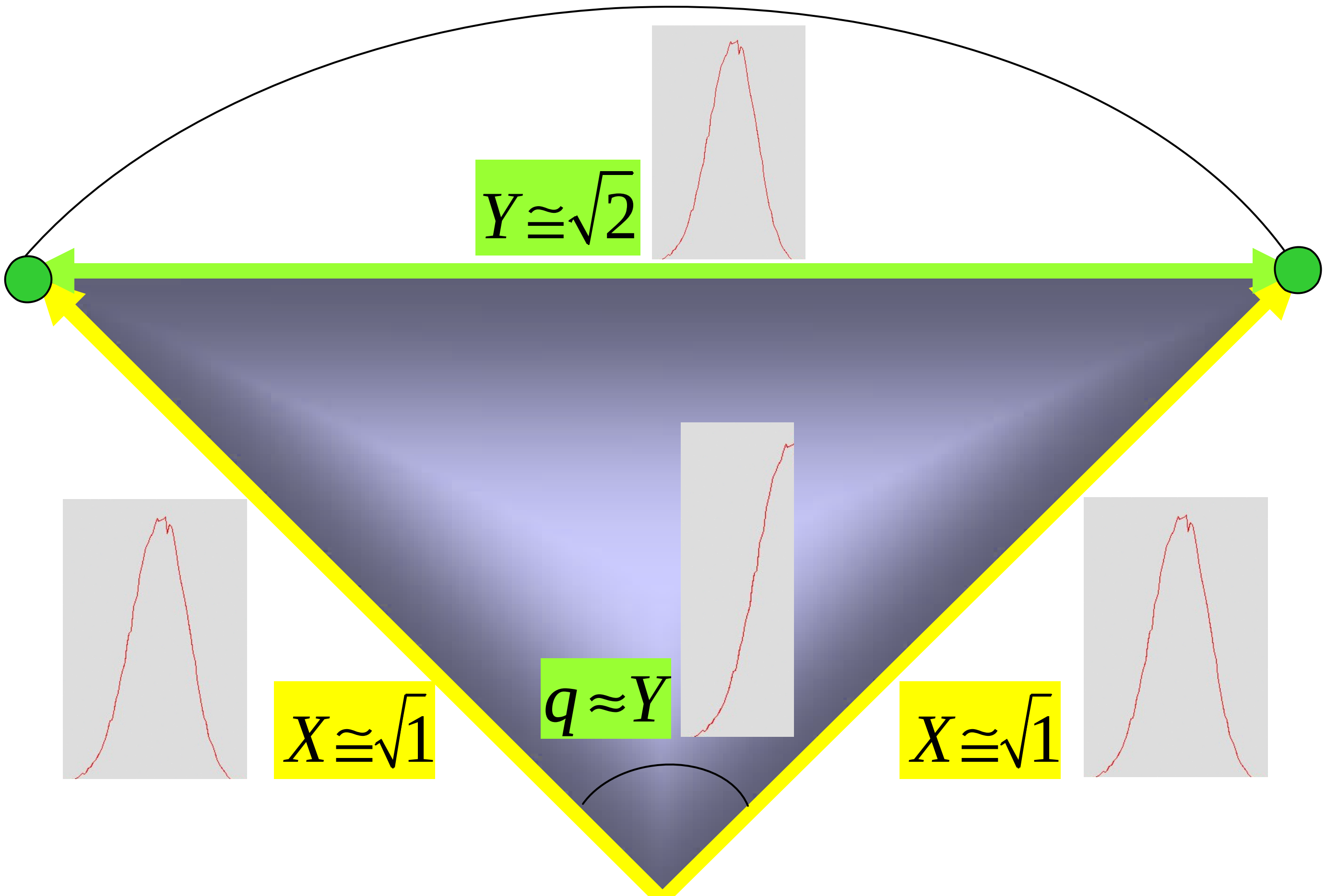


$$q = \cos^{-1}\left(\frac{\langle \boldsymbol{x} | \boldsymbol{y} \rangle}{\|\boldsymbol{x}\| \ \|\boldsymbol{y}\|}\right)$$

Size of N_R	Standard Deviation	
	Inner Prod	As Angle
100	.1000	5.758°
1000	.0315	1.816°
10,000	.0100	0.563°

Size of N_C	Standard Deviation
	As Angle
100	4.05°
1000	1.27°
10,000	0.40°

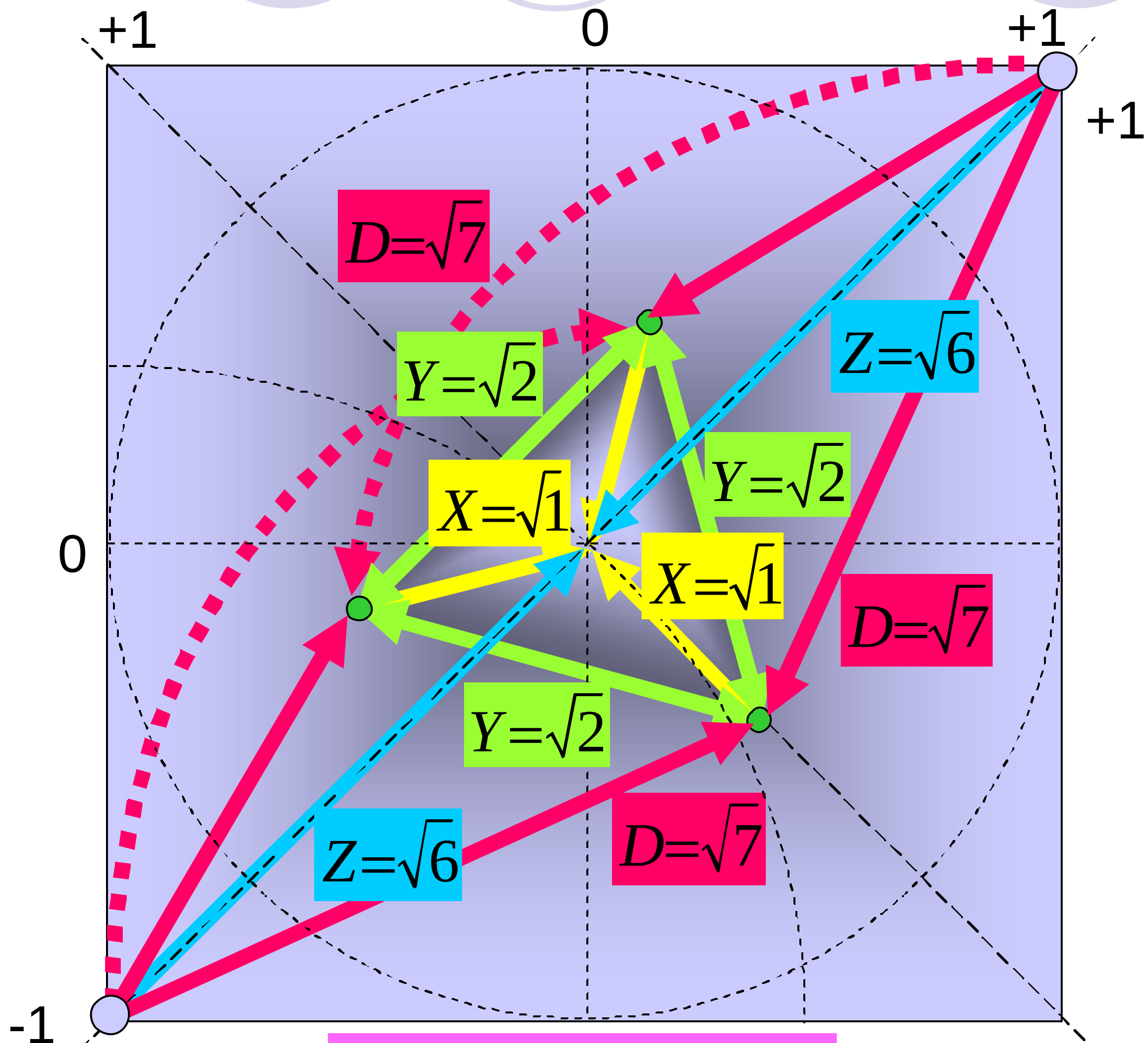
Standard Angle Proportional to Standard Distance



Corob tokens are identical to orthonormal basis states!

Random magnitude and phase for $N=N_c$

Distribution shifts relative to size of bounding box



$S=2/R=\sqrt{4}/R=\sqrt{12/N}$

$R=\sqrt{N/3}$

$X=\sqrt{N/3}/R=\sqrt{1}=1$

$Y=\sqrt{2N/3}/R=\sqrt{2}$

$Z=\sqrt{2N}/R=\sqrt{6}$

$D=\sqrt{7N/3}/R=\sqrt{7}$

$K=\sqrt{4N}/R=\sqrt{12}$

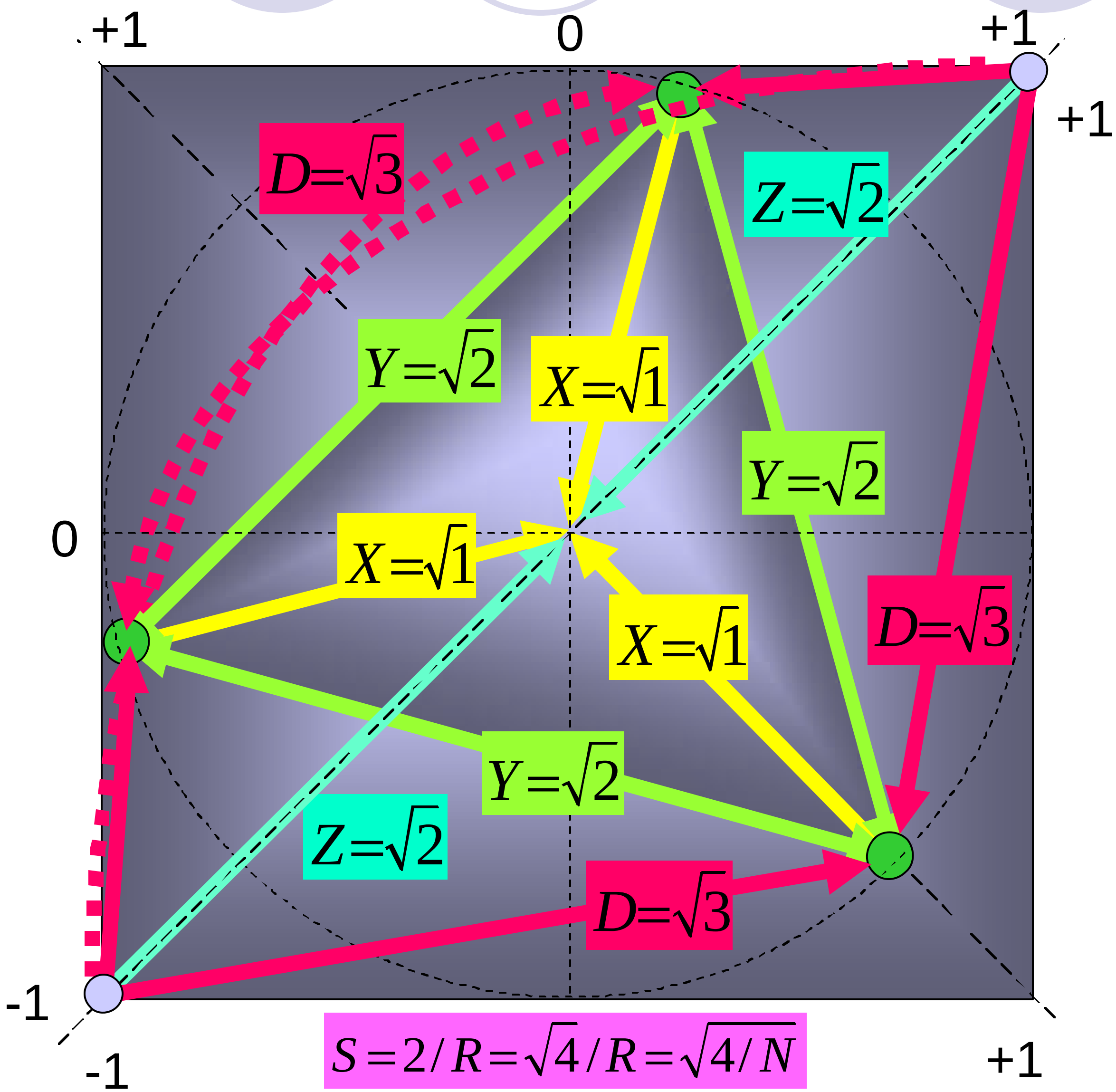
$M=\sqrt{8N}/R=\sqrt{24}$

$C=\sqrt{4N/3}/R=\sqrt{4}=2$

$D^2=\sqrt{6}^2+\sqrt{1}^2=\sqrt{7}^2$

8/15/2002 DJM

Random phase and magnitude=1 for N_c



$$R=\sqrt{N}$$

$$X=\sqrt{N}/R=\sqrt{1}=1$$

$$Y=\sqrt{2N}/R=\sqrt{2}$$

$$Z=\sqrt{2N}/R=\sqrt{2}$$

$$D=\sqrt{3N}/R=\sqrt{3}$$

$$K=\sqrt{4N}/R=\sqrt{4}=2$$

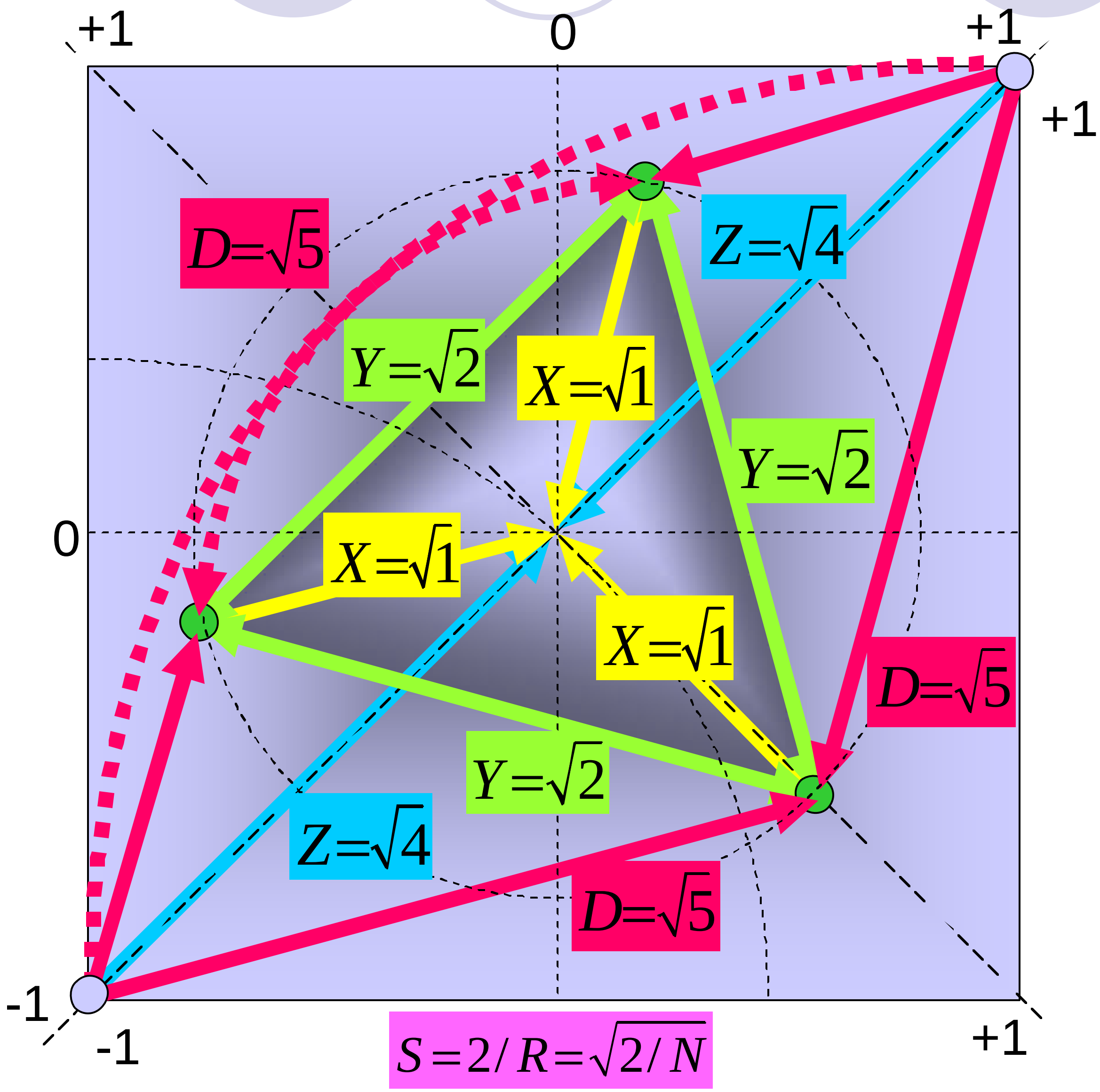
$$M=\sqrt{8N}/R=\sqrt{8}$$

$$C=\sqrt{4N}/R=\sqrt{4}=2$$

$$D^2=\sqrt{2}^2+\sqrt{1}^2=\sqrt{3}^2$$

8/15/2002 DJM

Qubits with Random phase for N_q



$$R=\sqrt{N}$$

$$X=\sqrt{N}/R=\sqrt{1}=1$$

$$Y=\sqrt{2N}/R=\sqrt{2}$$

$$Z=\sqrt{4N}/R=\sqrt{4}=2$$

$$D=\sqrt{5N}/R=\sqrt{5}$$

$$K=\sqrt{8N}/R=\sqrt{8}$$

$$M=\sqrt{16N}/R=\sqrt{16}=4$$

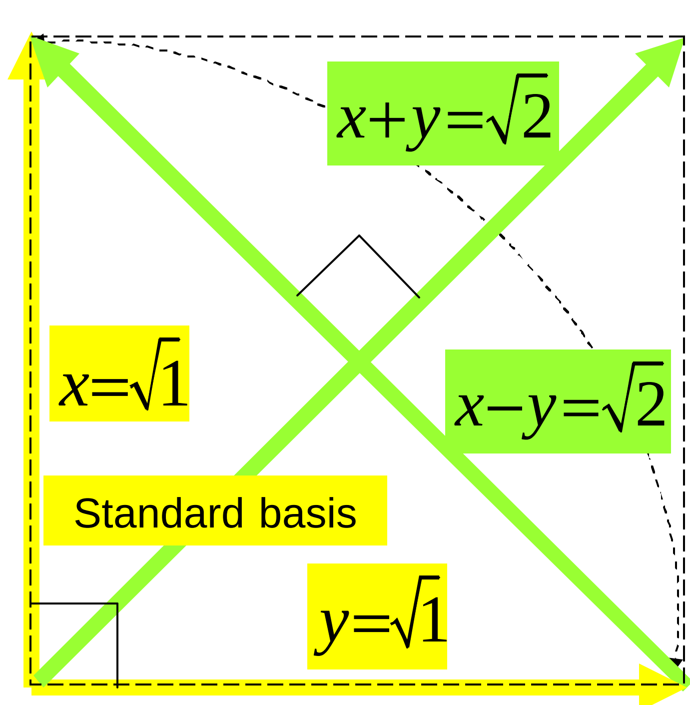
$$C=\sqrt{4N}/R=\sqrt{4}=2$$

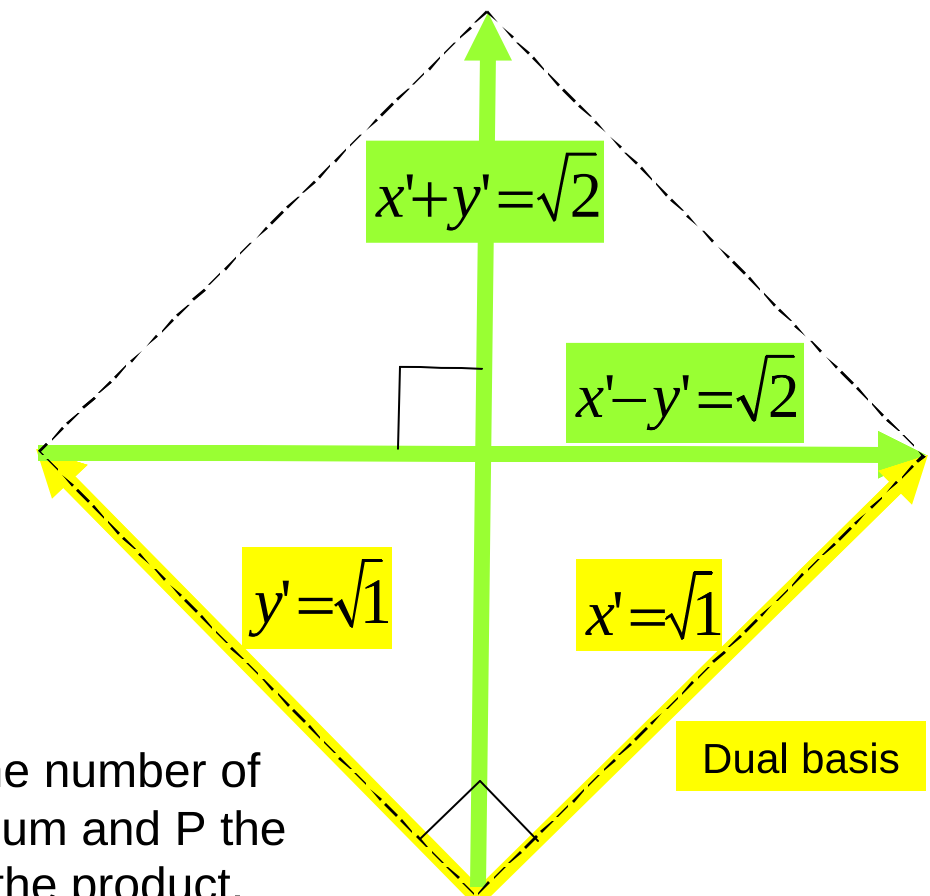
$$D^2=\sqrt{4}^2+\sqrt{1}^2=\sqrt{5}^2$$

8/15/2002 DJM

Normalizing Vector Add/Mult

Corob equivalence to unitarity constraint





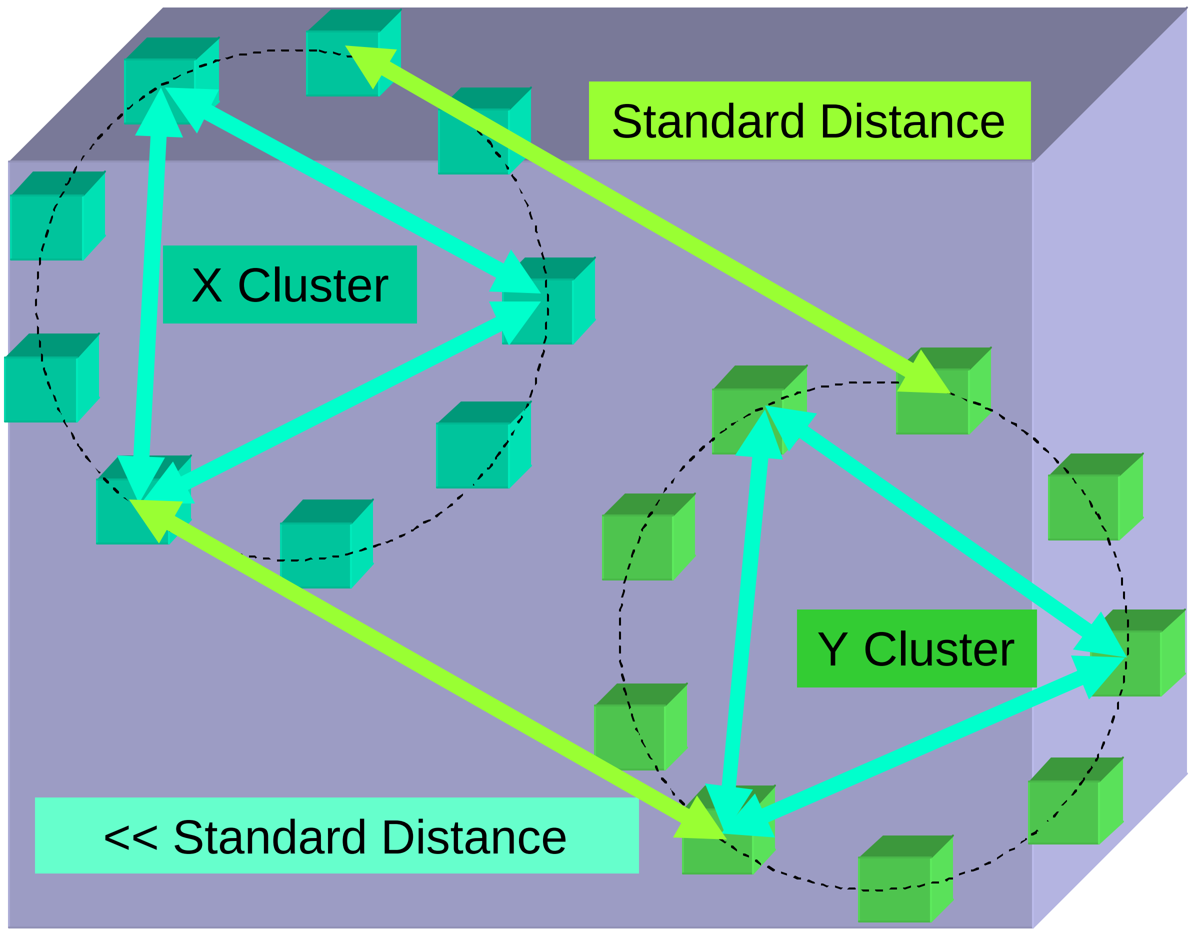
$$sum = \frac{x \pm y \pm z \pm \ldots}{\sqrt{S}}$$

Where S is the number of terms in the sum and P the # of terms in the product.

The bounds on the sumation space increases by sqrt(S)

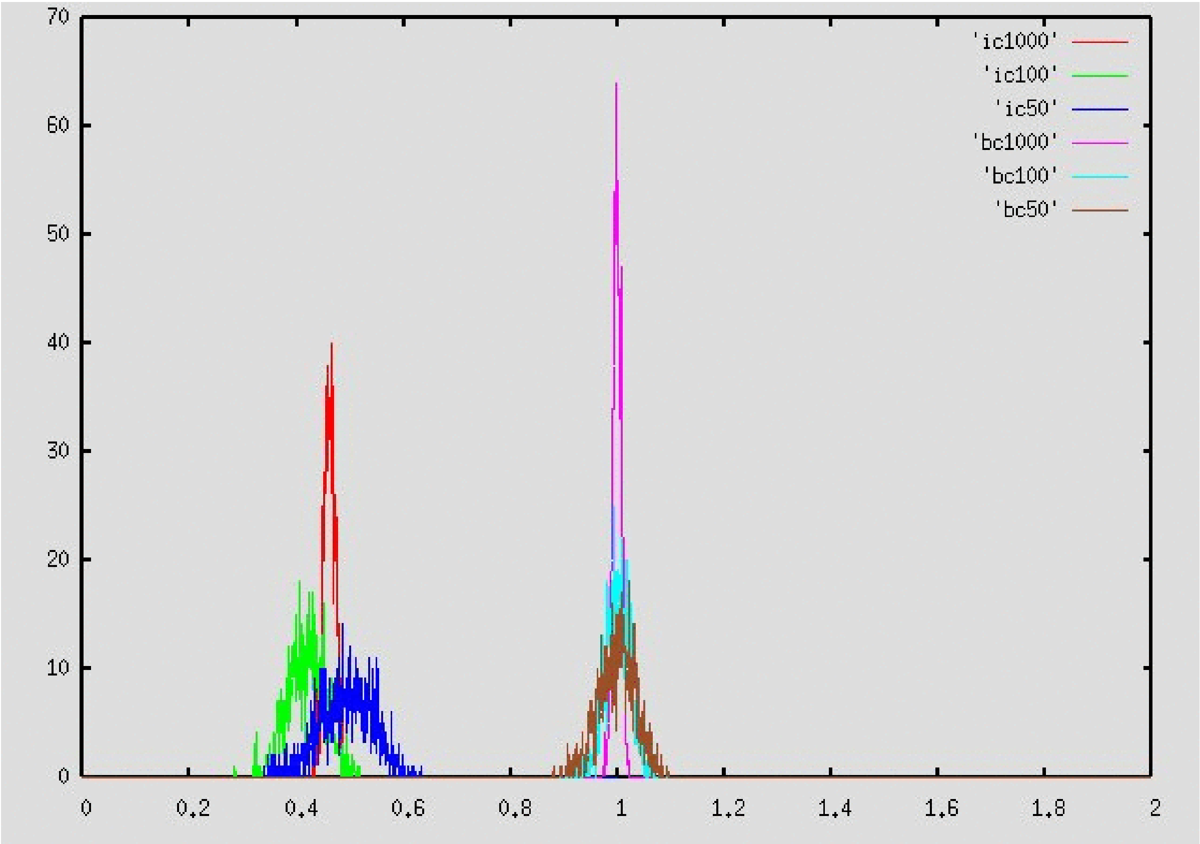
$$prod = (x \times y \times z \times \ldots)\sqrt{3}^{P-1}$$

Quantum Corobs Survive Projection



- Two random phase corobs X,Y
- Encode as arrays of qubit phases
- Measure qubits to form class. corob
- Repeat process or run concurrently
- All Xs will look like noisy versions of each other.
- All Ys will look like noisy versions of each other.

Quantum Randomness Generates Corobs

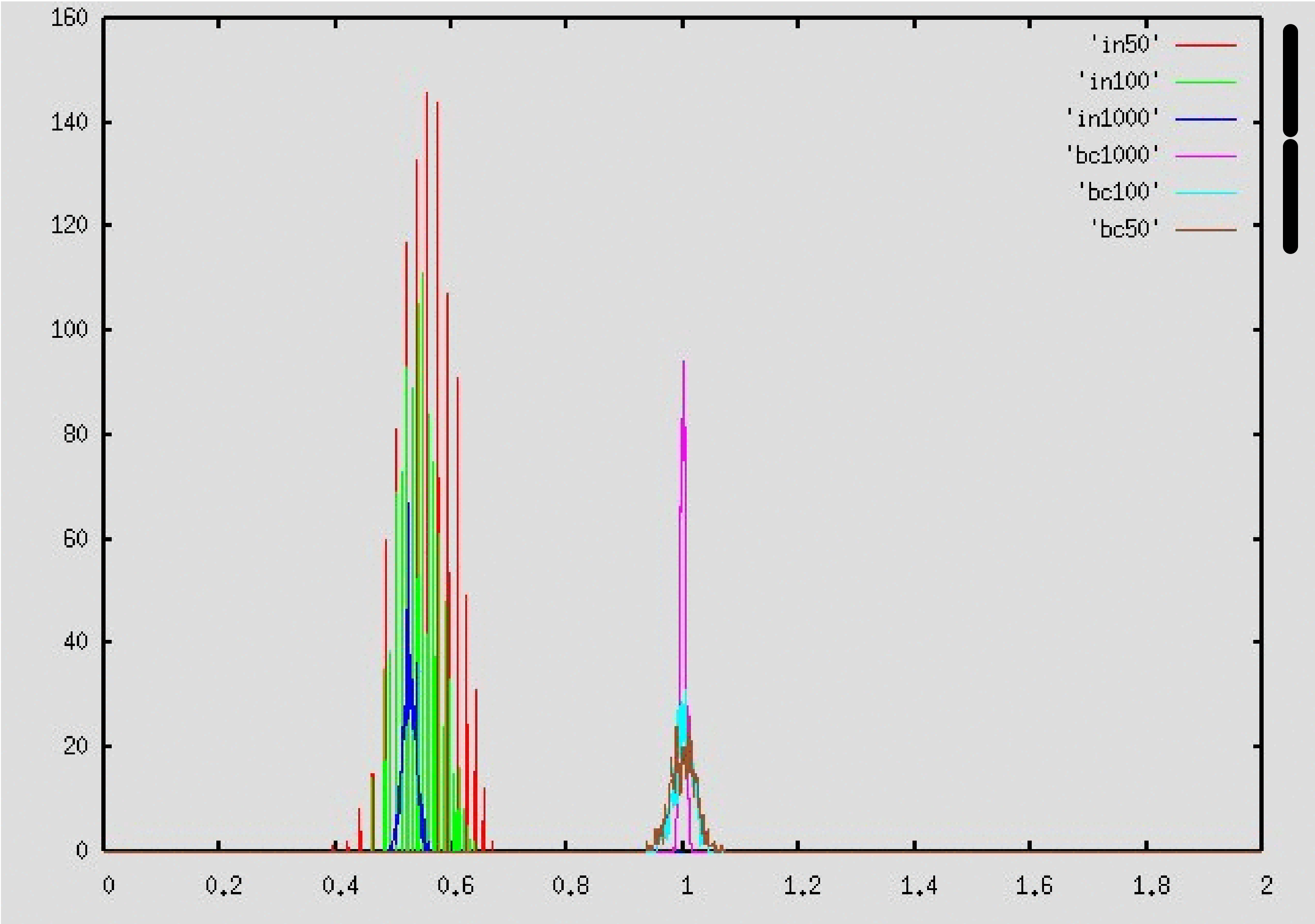


Element	%standist
complex	70.7%
1 qubit	50%
2 qubits	70.7%
4 qubits	95%
>4qubits	100%
ebit (q=2)	50%
ebit (q=3)	50%

Moral: Use arrays of simple qubits or ebits to represent corobs else the quantum randomness will destroy token identity. This suggests simple *ensemble* computing!

Corobs must be random but *repeatable* !!

Ebits Maintain Token Separation

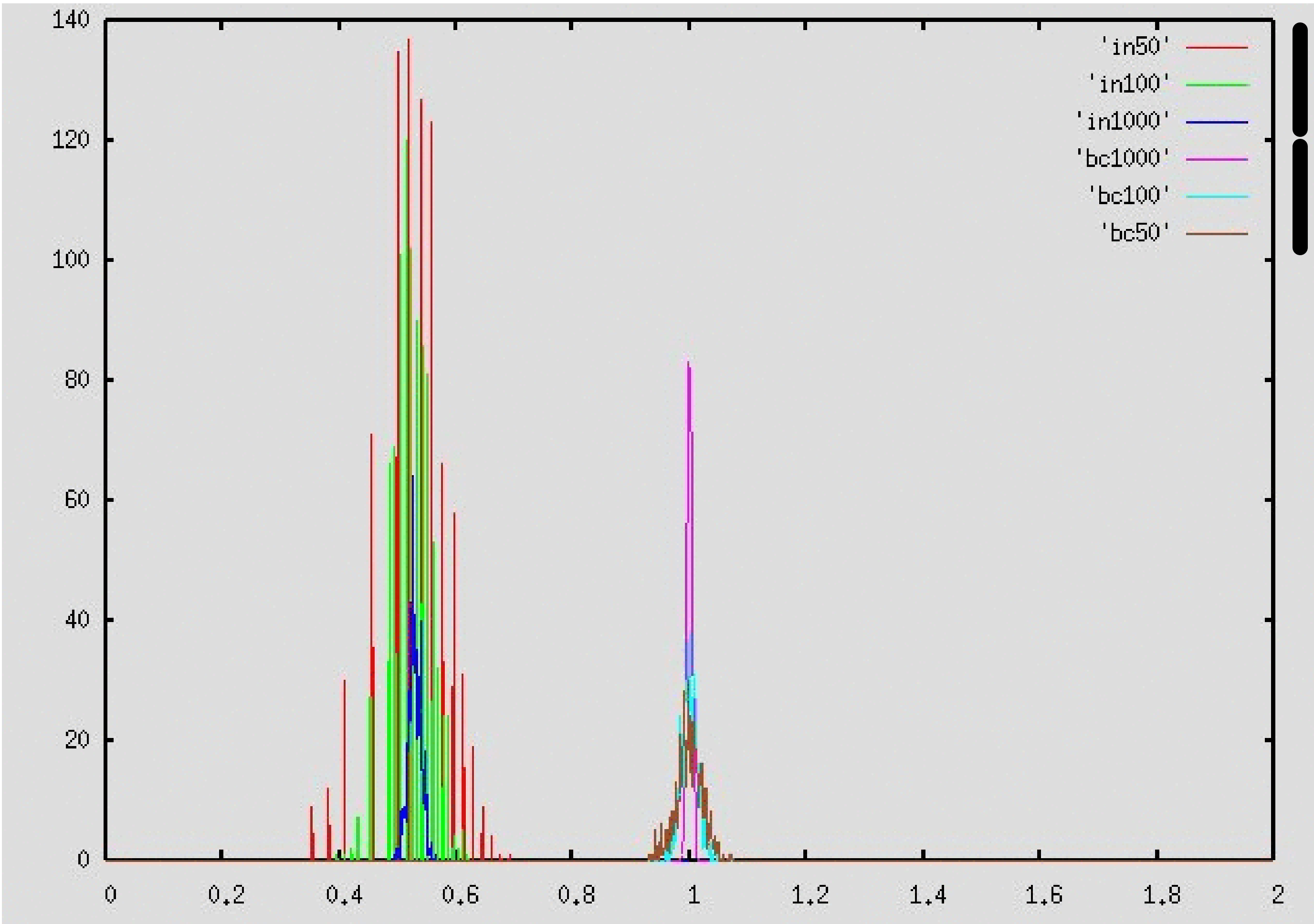


In cluster dist

between clusters dist

Ebit projection cluster separation histogram normalized by Ebit standard distance

Ebit Histogram for 3 entangled qubits



In cluster dist

between clusters dist

Summary and Conclusions

- Information, probability and distance metrics
- Soft tokens or Corobs approach *robustly* expresses classical and quantum computing.
- Strong correlation between corob and quantum computing theories suggests:
  - Corob based languages useful for quantum comp
  - Quantum systems may naturally represent corobs
  - Robustness in corob theory may be useful as natural error correction for quantum systems
  - New high dimensional interpretation of quantum with new insight underlying uncertainty principle