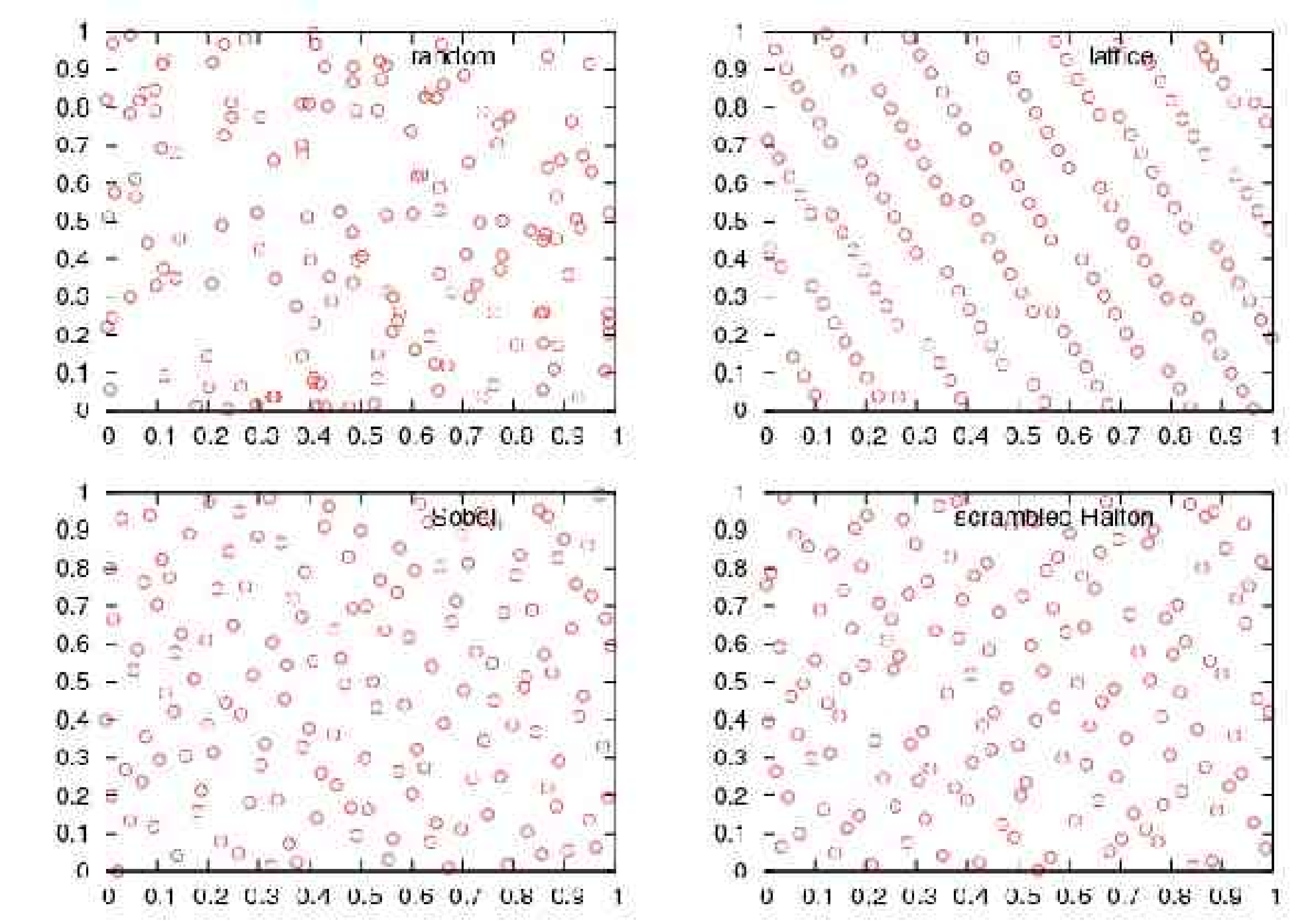
Quasi-random points

(Teytaud, Tao (Inria), Lri (Paris-Sud), UMR-Cnrs 8623;
collabs with S. Gelly, J. Mary, S. Lallich, E. Prudhomme,...)

- Quasi-random points ?
- Dimension 1
- Dimension n
- Better in dimension n
- Strange spaces

Quasi-random points ?

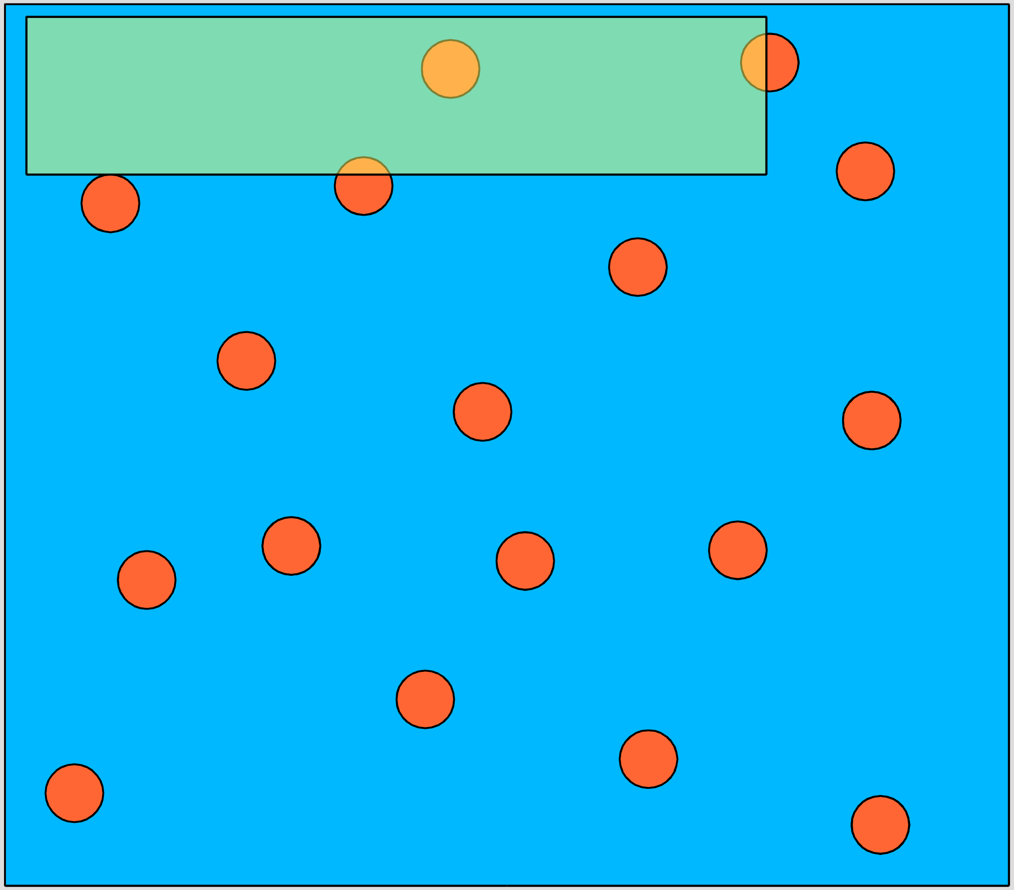


For what ?

- Numerical integration [thousands of papers; Niederreiter 92]
- Learning [Cervellera et al, IEEETNN 2004, Mary phD 2005]
- Optimization [Teytaud et al, EA'2005]
- Modelizat° of random-process [Growe-Kruska et al, IEEEBPTP'03]
- Path planning [Tuffin]

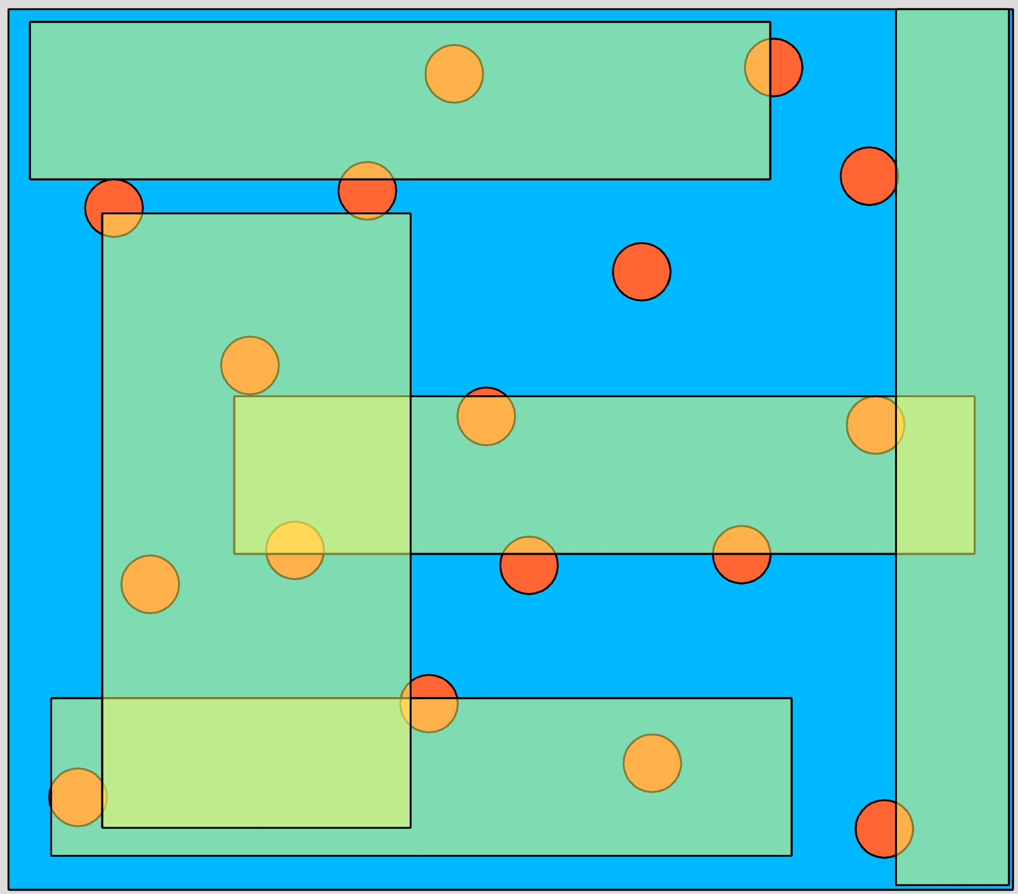
Low Discrepancy ?

Discrepancy = Max |Area – Frequency |



Low Discrepancy ?

Discrepancy2 = mean (|Area – Frequency |2)



Low Discrepancy ?

Random --> Discrepancy ~ sqrt (1/n)

Quasi-random --> Discrepancy ~ log(n)^d/n

Koksma & Hlawka :

error in Monte-Carlo integration

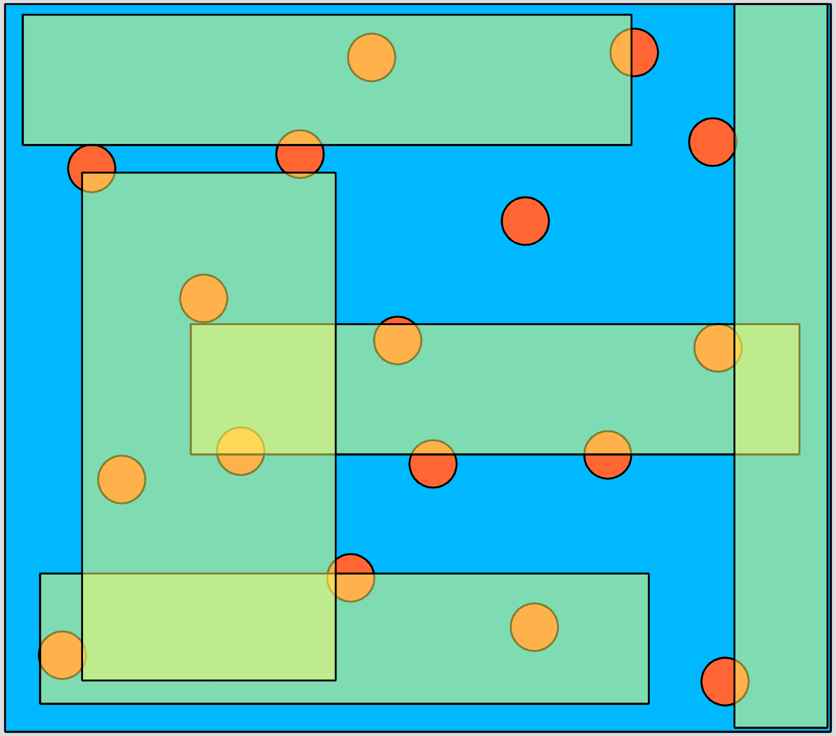
< Discrepancy x V

V= total variation (Hardy & Krause)

(many generalizations in Hickernel,

A Generalized Discrepancy

and Quadrature Error Bound, 1997)

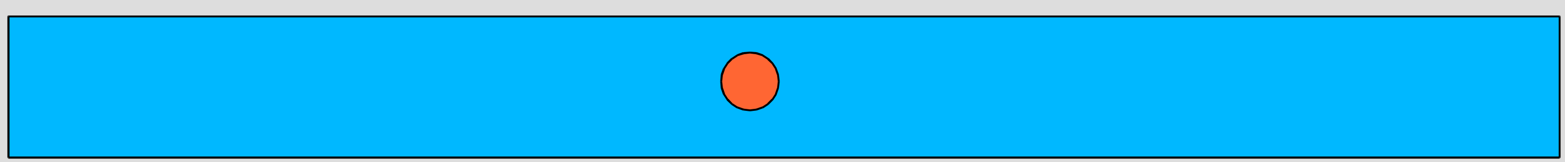


Dimension 1

- What would you do ?

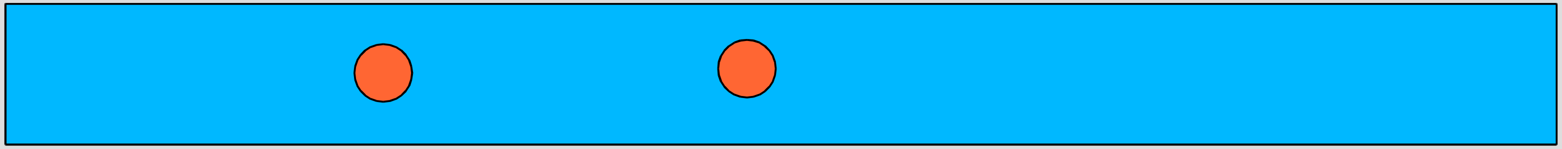
Dimension 1

- What would you do ?



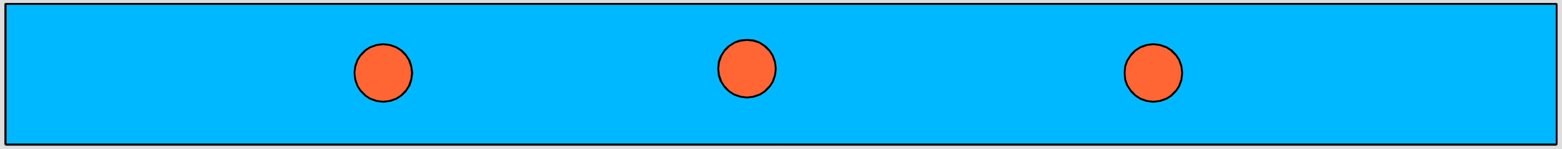
Dimension 1

- What would you do ?



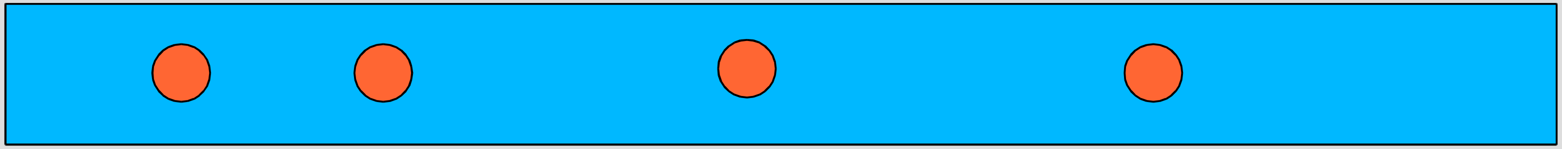
Dimension 1

- What would you do ?



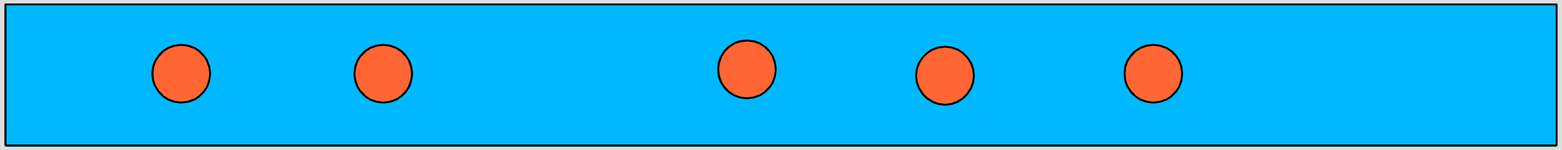
Dimension 1

- What would you do ?



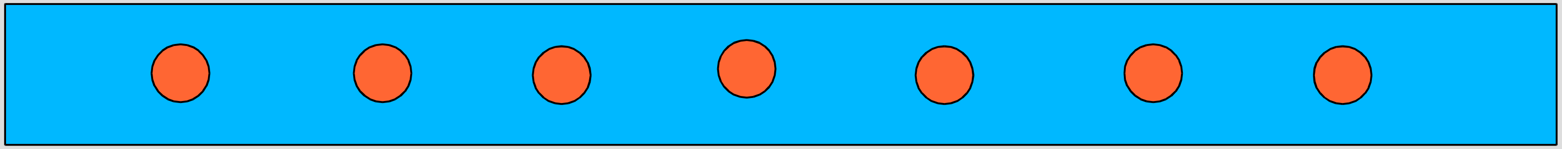
Dimension 1

- What would you do ?



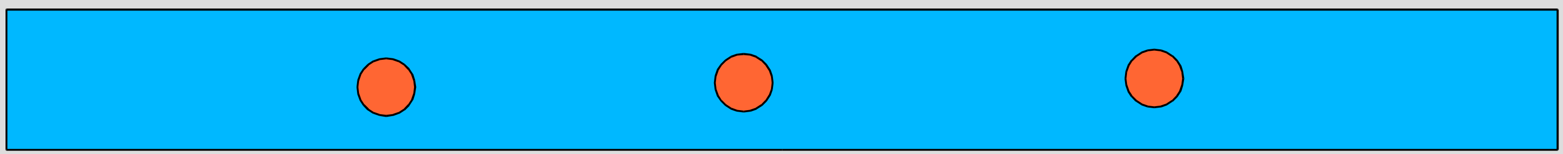
Dimension 1

- What would you do ?
- --> Van Der Corput
- n=1, n=2, n=3...
- n=1, n=10, n=11, n=100, n=101, n=110...
- x=.1, x=.01, x=.11, x=.001, x=.101, ...

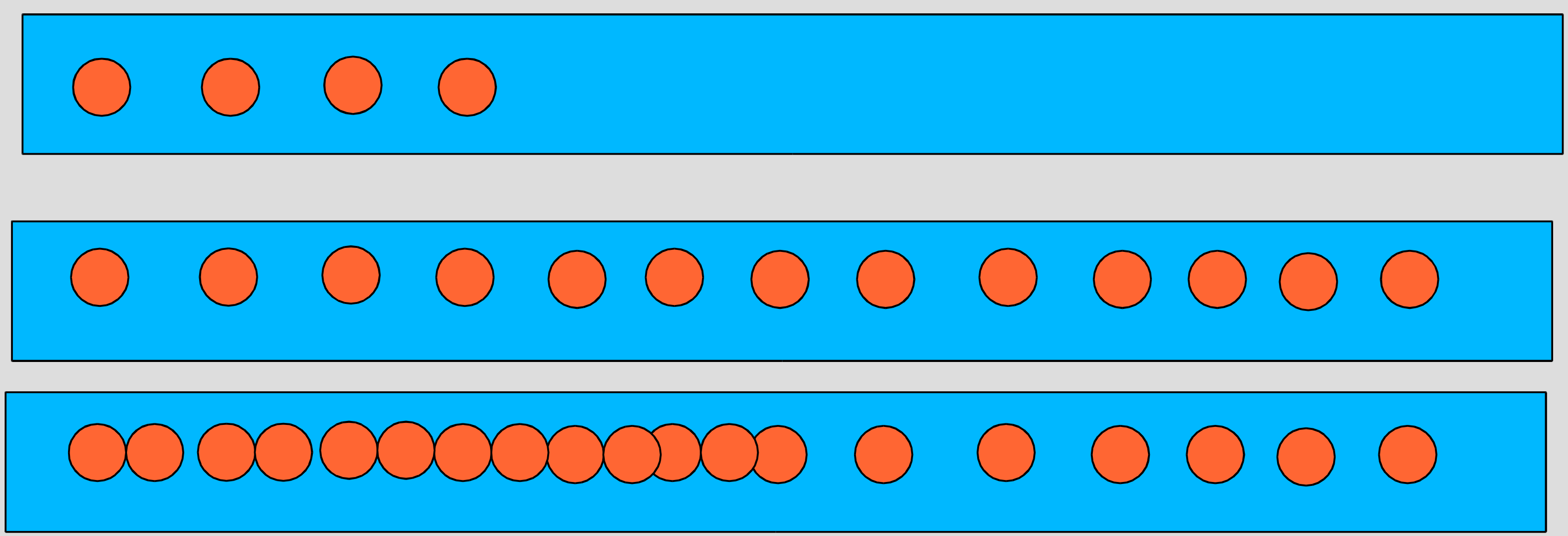


Dimension 1 more general

- p=2, but also p=3, 4, ...

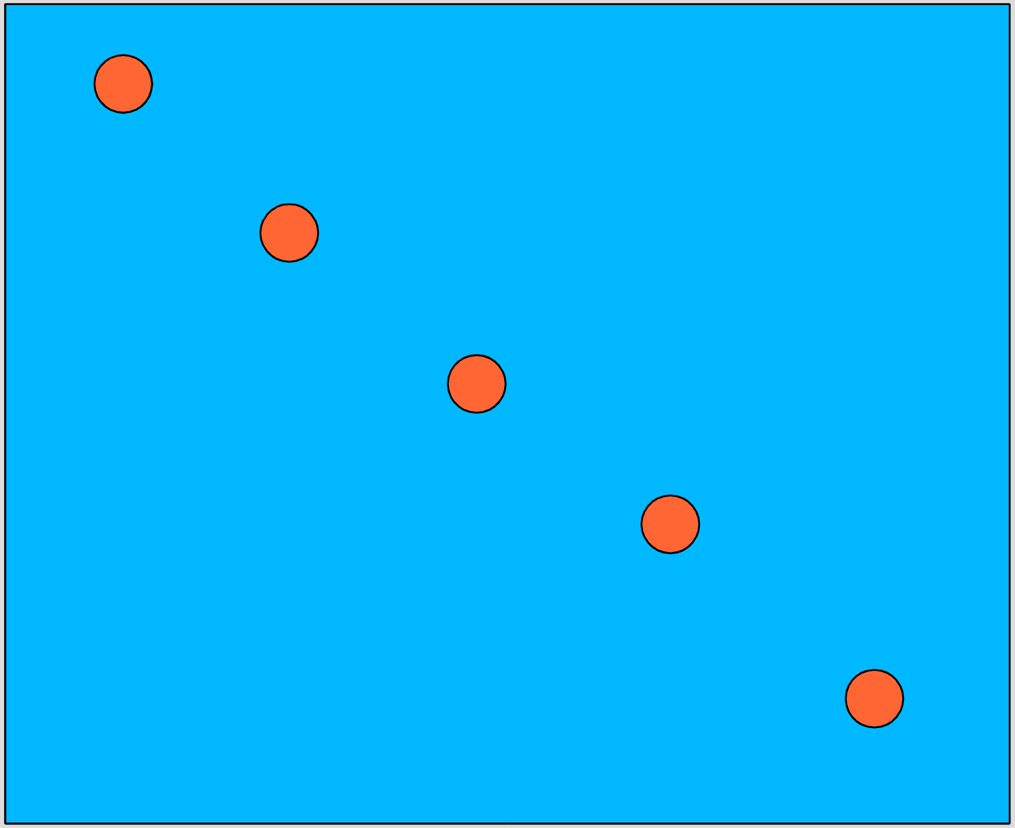


but p=13 is not very nice :



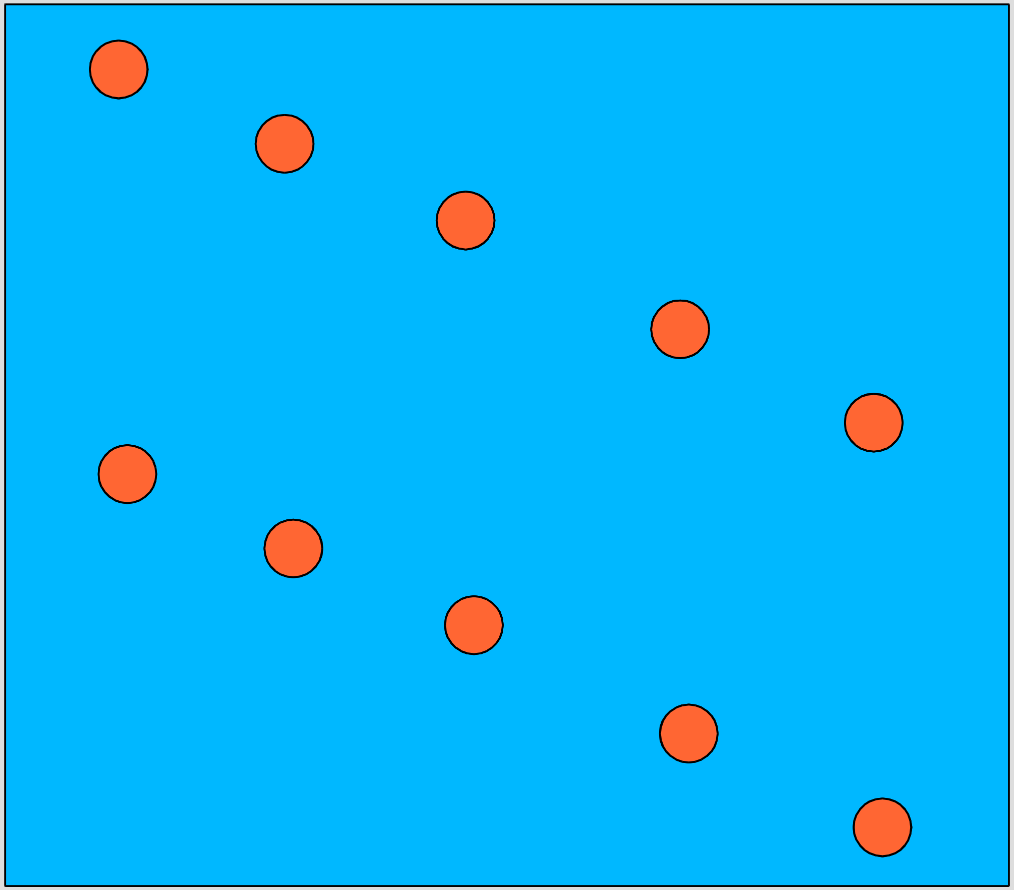
Dimension n

- x --> (x,x) ?



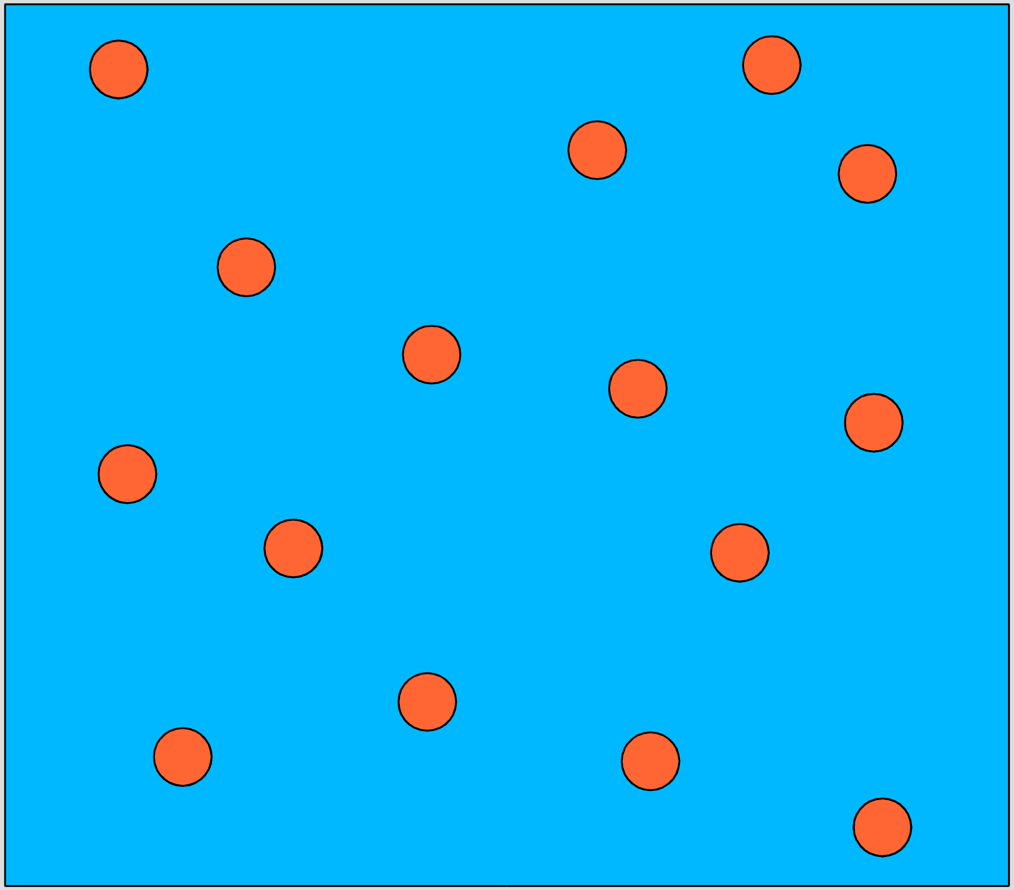
Dimension n

- x --> (x,x') ?



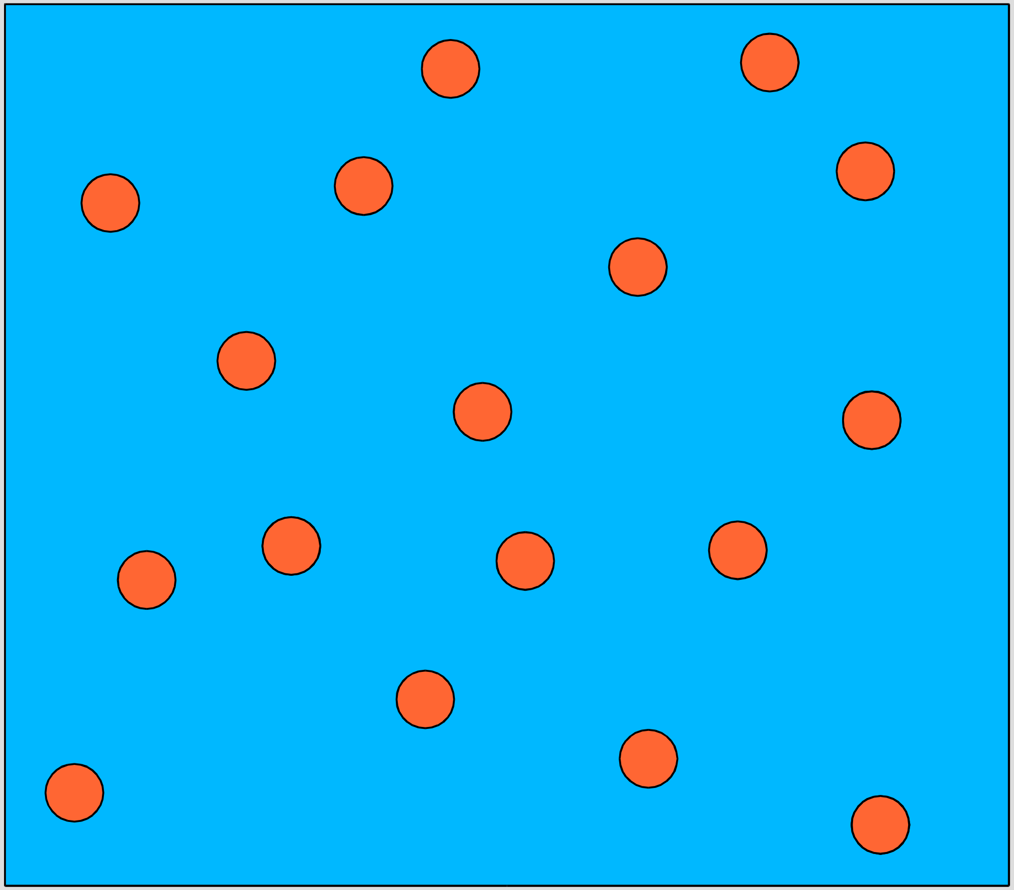
Dimension n : Halton

- x --> (x,x') with prime numbers



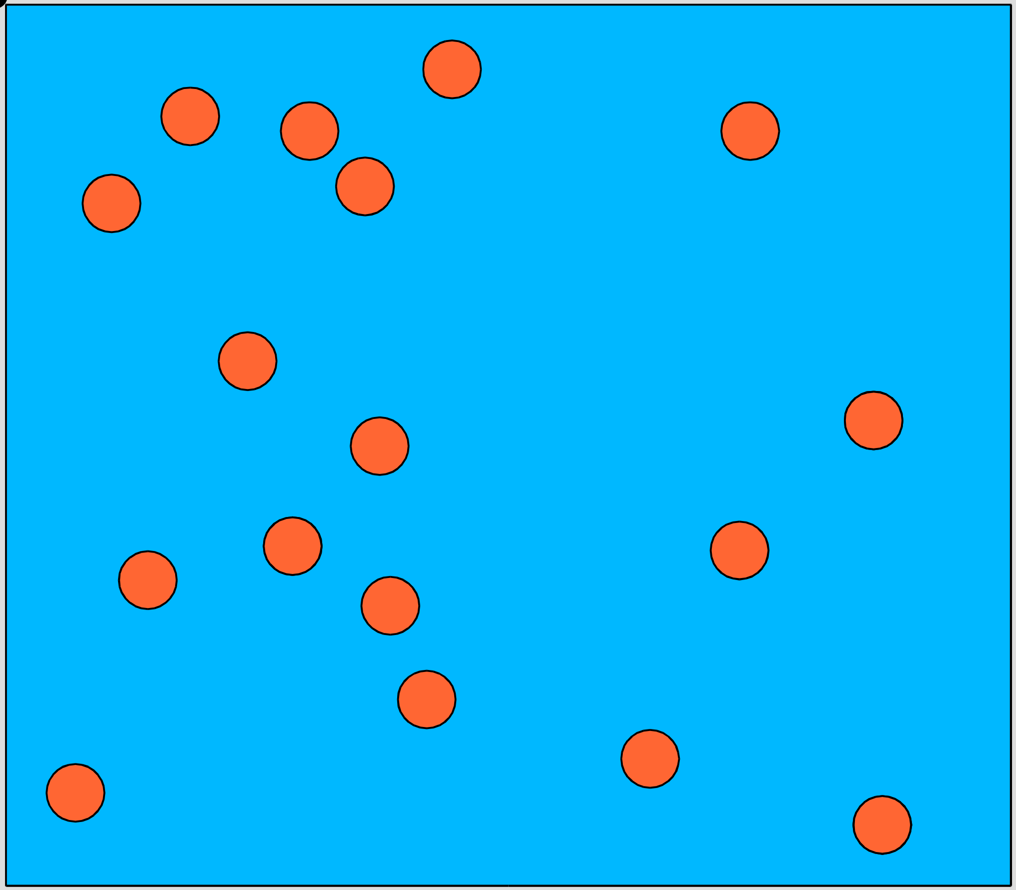
Dimension n+1 : Hammersley

- x --> (n/N,x,x') but --> closed sequence



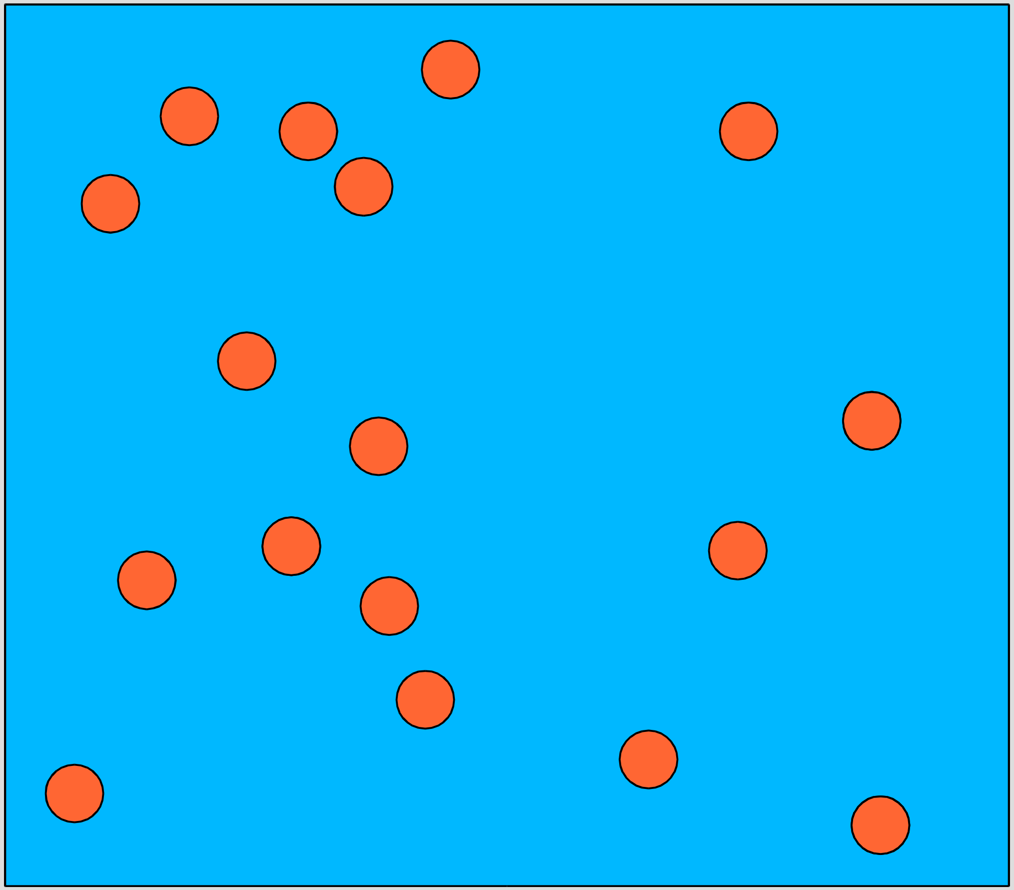
Dimension n : the trouble

- There are not so many small prime numbers



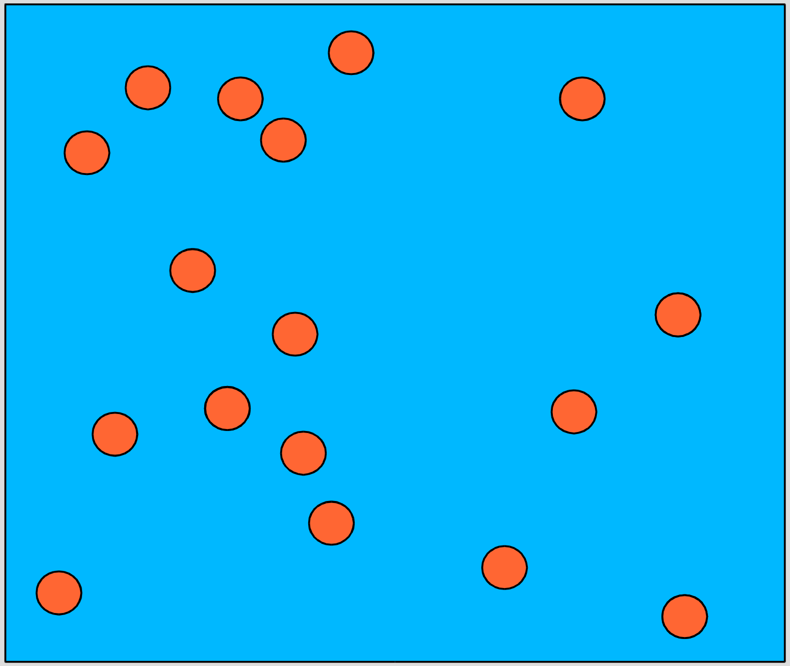
Dimension n : scrambling (when random comes back)

- Pi(p) : [1,p-1] --> [1,p-1]
- Pi(p) applied to ordinate with prime p



Dimension n : scrambling

- Pi(p) : [1,p-1] --> [1,p-1] (randomly chosen)
- Pi(p) applied to ordinate with prime p (there is much more complicated)



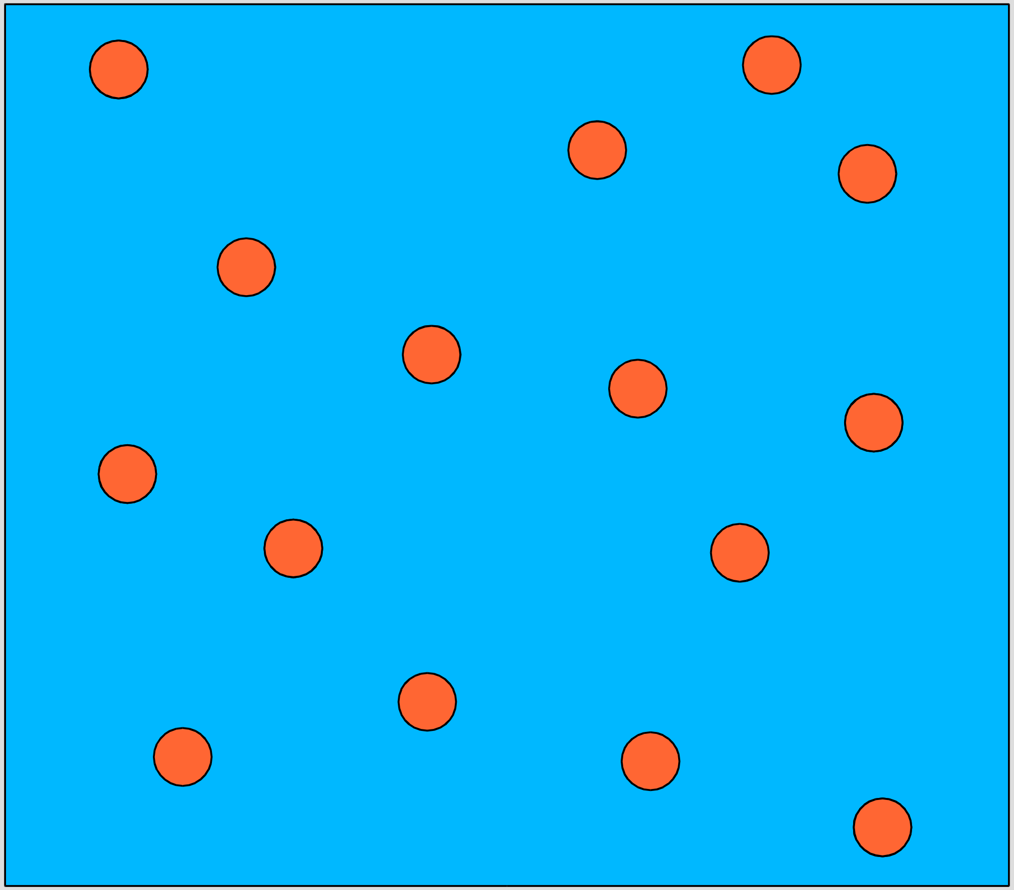
Others ?

- Other discrepancies : why rectangles ?
- Other solutions : lattices

  {x0+nx} modulo 1

Why in the square ?

- Other spaces : gaussians --> spheres



Why in the square ?

- Other spaces : gaussians --> spheres
  - *x in]0,1[^d*
  - *y(i)* such that   *P(N > y(i)) = x (i)*
    with *N* standard gaussian
  - then *y* is quasi-random and gaussian

Why in the square ?

- Other random variables : PDF

$$z = \inf \{ t \,;\, P(Z<t) > x \}$$

==> strange spaces or variables

Why in the square ?

- Other random variables : CDF
  $0<x<1$ --> $z=\inf \{ t ; P(Z<t) > x \}$
  ==> ok for discrete spaces

e.g.

Samplings [IIIA'2006] :

- quasi-random bagging
- quasi-random cross-validation
- quasi-random bootstrap
- quasi-random subagging

We open a parenthesis on quasi-random bagging

(

Quasi-random bagging : hkgnz ?

- Bagging is based on Monte-Carlo sampling :
  - Generate plenty of hypothesis
  - Take the average

- This is exactly like Monte-Carlo integration :
  - Generate plenty of values
  - Take the average

- We can do quasi-Monte-Carlo

Quasi-random bagging (i)

- Bagging:
  - Consider *n* examples
  - For *i=1* to *N*
    - Randomly draw *n* examples with replacement
    - Learn an hypothesis *Hi*
  - Hypothesis = sum *Hi* / *N*
- Subagging : idem but draw
  m << *n* examples
- Quasi-random (su-)bagging ?

Quasi-random bagging (ii)

- Drawing *n* examples among *n* :
  - Build *k* clusters
  - Choose the number of examples in the first cluster ;
  - Choose the number of examples in the second cluster ;
  - ...
  - Choose the number of examples in the last cluster.

Quasi-random bagging (iii)

- Drawing *n* examples among *n* :
  - Build *k* clusters
  - Choose the number of examples in the first cluster (= binomial = fisrt ordinate) ;
  - Choose the number of examples in the second cluster (= binomial = second ordinate) ;
  - ...
  - Choose the number of examples in the last cluster (= binomial = *k*^th ordinate).

Quasi-random bagging (iv)

- Evaluate the *CDF* of the number of examples in cluster *i* (binomial)
- *CDF^{-1}(xi)* = number in cluster *i*

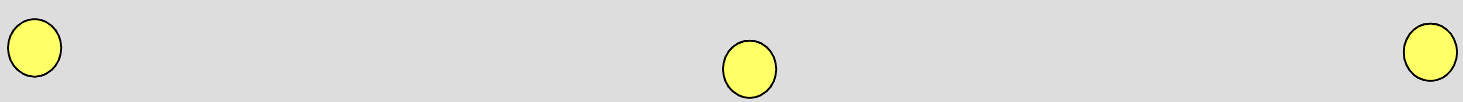
therefore, quasi-random in dimension *k*

We close the parenthesis

)

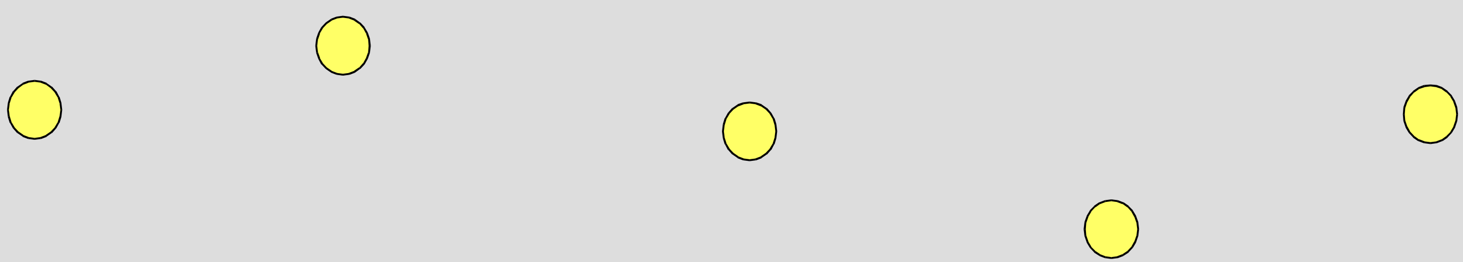
Why not for random walks ?

- 500 steps of random walks ==> huge dimension
- But strong derandomization possible : with dim 3, place 3 points with ad hoc distribution



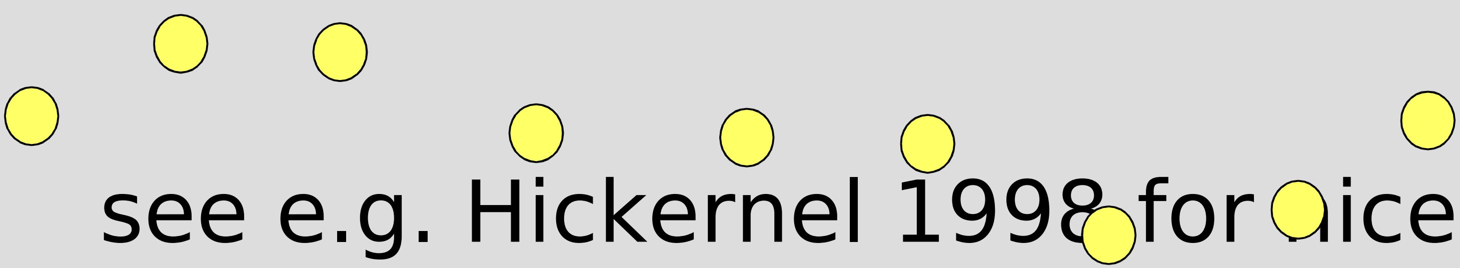
Why not for random walks ?

- 500 steps of random walks ==> huge dimension
- But strong derandomization possible : dim 4 and 5 provide two more points



Why not for random walks ?

- 500 steps of random walks ==> huge dimension
- But strong derandomization possible :



see e.g. Hickernel 1998 for nice generalizations

Very different approaches for derandomization ?

- **Symetries :** instead of
  x in [0,1]
  try

  use x and 1-x
- *2 x (n/2)* points better than *n* points
- ==> antithetic variables
- ==> roughly, it is almost always better, whereas quasi-random might be disappointing

Very different approaches for derandomization ?

- <u>**Control :**</u> instead of estimating

  $E\ f(x)$

- Choose *g* looking like f and estimate

  $E\ (g-f)(x)$

- Then $E\ f = E\ g + E(g-f)$ is much better
- Troubles:
  - You need a good *g*
  - You must be able of evaluating *Eg*

Very different approaches for derandomization ?

- **Pi-estimation :** instead of estimating $E\ f(x)$
- Look for y with density $\simeq \sigma(f) d(x)$
- Then $E\ f(x) = E\ f(y)\ d(x)/d(y)$
  ==> *Variance is much better*
- Troubles:
  - You have to generate y
  - You have to know $\sigma(f)$

Very different approaches for derandomization ?

- **Stratification (jittering) :**
- Instead of generating n points i.i.d
- Generate
  - k points in stratum 1
  - k points in stratum 2
  - ...
  - k points in stratum m
    with m.k=n ==> more stable
    ==> depends on the choice of strata

Biblio (almost all on google)

- Owen, A.B. "Quasi-Monte Carlo Sampling", A Chapter on QMC for a SIGGRAPH 2003 course.
- Fred J. Hickernell, A generalized discrepancy and quadrature error bound, 1998
- B. Tuffin, On the Use of low-Discrepancy sequences in Monte-Carlo methods, 1996
- Matousek, Geometric Discrepancy (book 99)

these slides : http://www.lri.fr/~teytaud/btr2.pdf
or http://www.lri.fr/~teytaud/btr2.ppt