Online Uniformity of Integer Points on a Line ⋆

Tetsuo Asano

School of Information Science, JAIST,
1-1 Asahidai, Nomi, 923-1292 Japan.

Abstract

This letter presents algorithms for computing a uniform sequence of n integer points in a given interval $[0, m]$ where m and n are integers such that $m > n > 0$. The uniformity of a point set is measured by the ratio of the minimum gap over the maximum gap. We prove that we can insert n integral points one by one into the interval $[0, m]$ while keeping the uniformity of the point set at least $1/2$. If we require uniformity strictly greater than $1/2$, such a sequence does not always exist, but we can prove a tight upper bound on the length of the sequence for given values of n and m.

Key words: Algorithm, Uniformity, Discrepancy

1 Introduction

A number of applications need uniformly distributed points over a specific domain. It is commonly known that randomly generated points are not always good enough. In a mesh generation, for example, we have to distribute points uniformly over a region of interest to form good meshes. But, first of all, how can we measure the uniformity of points? In the theory of Discrepancy (3; 4) the uniformity of points is measured by how the number of points in a small region such as an axis-parallel rectangle changes while moving around the domain, more formally by the difference (or discrepancy) between the largest and smallest numbers of points in the moving region. For normalization we usually divide the difference by the area of the moving region. Then, the discrepancy

⋆ Preliminary version of this letter was presented at 24th European Workshop on Computational Geometry (Nancy, France, March 2008)

Email address: t-asano@jaist.ac.jp (Tetsuo Asano).

[1] Supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research on Priority Areas and Scientific Research (B).

is given as the supremum of the ratios for all possible scales of the region. One of difficulties here is hardness of such evaluation since we have to prepare all possible scales and all possible locations.

We consider a special case of such a problem, that is, how to insert n integer points in a given interval $[0, m]$ so that points are uniformly distributed, or in other words, the ratio of the minimum gap over the maximum gap is not so low. We present a simple algorithm for achieving the ratio 1/2 for all integers m and n with $m > n > 0$. It is not trivial at all to achieve the ratio strictly greater than 1/2. In addition, if we require uniformity strictly greater than 1/2, such a sequence does not always exist, but we can prove a tight upper bound on the length of the sequence for given values of n and m.

The problem considered in this letter may open a new direction of discrepancy theory. The first extension from the current discrepancy theory is from uniformity measure for a static set of points to one for a sequence of points. The second extension is from continuous coordinates to discrete ones. This extension is important since now we have a discrete combinatorial optimization problem, which may lead to good approximation algorithms.

2 Problem

Let $m > n > 0$ be arbitrary integers. An (n, m)-*sequence* is a sequence of integers (or points of integral coordinates) $\sigma = (0, m, p_1, \ldots, p_n)$ in the closed interval $[0, m]$. The uniformity of the sequence is measured by the ratio of the minimum gap over the maximum gap where a gap is difference between two consecutive integer points when they are arranged on a line in a sorted order. It may be natural and reasonable to measure the uniformity of a point set $\{0, m, p_1, \ldots, p_n\}$ by the ratio of the minimum gap $\delta_{\min}(0, m, p_1, \ldots, p_n)$ over the maximum gap $\delta_{\max}(0, m, p_1, \ldots, p_n)$, that is, the *(static) uniformity* $\mu_s(0, m, p_1, \ldots, p_n)$ of the set is defined by

$$\mu_s(0, m, p_1, \ldots, p_n) = \frac{\delta_{\min}(0, m, p_1, \ldots, p_n)}{\delta_{\max}(0, m, p_1, \ldots, p_n)}. \tag{1}$$

In this letter we are interested in uniformity achieved by a sequence of points. That is, points are inserted one by one. Every time when a point is inserted, we measure the uniformity of the point set. The worst uniformity we obtain before inserting all the points according to a given point sequence is defined as the *online uniformity* of the point sequence. Formally, we define the *online uniformity* $\mu(0, m, p_1, \ldots, p_n)$ for a point sequence $(0, m, p_1, \ldots, p_n)$ of length n (neglecting the first two points 0 and m) in the interval $[0, m]$ by

$$\mu(0, m, p_1, \ldots, p_n) = \min_{k=1,\ldots,n} \{\mu_s(0, m, p_1, \ldots, p_k)\}. \tag{2}$$

We call an (n, m)-sequence *uniform* if its online uniformity is strictly greater than 1/2.

3 Greedy Algorithm

A natural and naive idea to design a uniform sequence of points in a given interval $[0, m]$ is to repeat inserting a point to break the longest interval (of the maximum gap). It is rather straightforward to generalize this idea to higher dimensions. In higher dimensions we construct a Voronoi diagram for a current set of points and choose one of Voronoi vertices that is farthest from the closest point as the next point to insert. Thus, we call the algorithm *Voronoi Insertion*.

The performance of this greedy algorithm is not so bad. In fact, it achieves the uniformity 1/2 in one dimension (1; 2). However, it is not the case when points are limited to integer points. As a simple example consider a sequence of length 2 for an interval $[0, 6]$. The greedy algorithm chooses 3 as the first point, which is the midpoint of the interval. Then, we have two subintervals of length 3. Since we have to choose only integral points, one of the subintervals is divided into two subintervals of lengths 1 and 2. So, after choosing the two points the minimum gap is 1 while the maximum gap remains 3. So, the uniformity is $1/3 < 1/2$.

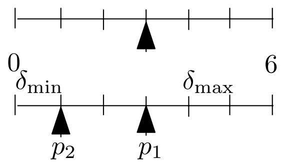


Fig. 1. Behavior of Voronoi Insertion on integer points.

As another example consider a case of $(m, n) = (10, 4)$. For simplicity of arguments we just maintain interval lengths instead of intervals. Initially we have $\{10\}$. By the first point we must have $\{5, 5\}$ or $\{4, 6\}$ since otherwise the uniformity would be worse (smaller) than 1/2. The even partition $\{5, 5\}$ does not lead to uniformity 1/2 because in the next division we have $\{2, 3, 5\}$, whose uniformity is $2/5 < 1/2$. So, $\{4, 6\}$ is the only choice and then we obtain the set $\{4, 2, 4\}$ by dividing the interval of length 6. Now, we can divide 4 into $\{2, 2\}$. Thus, the resulting set of interval lengths is $\{4, 2, 2, 2\}$ with uniformity $2/4 = 1/2$. On the other hand, if we divide 6 into $\{3, 3\}$, we have $\{4, 3, 3\}$. Now there is only one way of dividing the largest gap 4: $4 \to \{2, 2\}$. Then, we have $\{2, 2, 3, 3\}$. We have to divide 3, but there is only one way: $3 \to \{1, 2\}$. Thus, the resulting set is $\{2, 2, 1, 2, 3\}$ with uniformity $1/3 < 1/2$. See Figure 2 for illustration. This example implies that choosing the midpoint of the longest interval may not be so good even if there is a unique midpoint (note that there are two midpoints in an interval of odd length).

Now, a natural question is whether there is an algorithm for finding a sequence of points with uniformity at least 1/2 for any pair of integers m and n with $m > n$. The following lemma answers the question in an affirmative way.

Lemma 1 *There is an algorithm for finding an (n, m)-sequence of n points in an interval $[0, m]$ with uniformity at least $1/2$ for any pair of integers m and n if $m > n > 0$.*

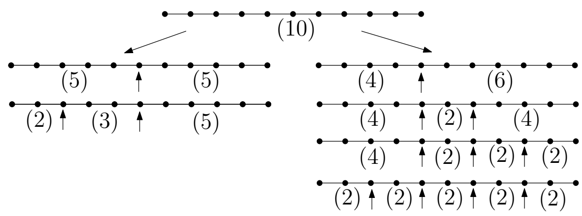


Fig. 2. Partition of an interval of length 10 in two different ways. If we divide 10 into 5, 5 as in the left figure, it is impossible to keep the uniformity $\geq 1/2$. The division $10 \to (4, 6)$ leads to a sequence with uniformity $\geq 1/2$.

Proof: We prove the lemma in a constructive manner. The algorithm iteratively partitions the longest interval (maximum gap). An important thing is to divide an interval of length m into ones of lengths 2^k and the rest $r = m - 2^k$ where k is an integer satisfying $3 \times 2^{k-1} \leq m < 3 \times 2^k$. If m happens to be a power of 2, say 2^ℓ, then 2^ℓ is partitioned into $\{2^{\ell-1}, 2^{\ell-1}$ since in this case we have $k = \ell - 1$. In fact, we have

$$3 \times 2^{\ell-2} \leq 4 \times 2^{\ell-2} = 2^\ell = 2 \times 2^{\ell-1} < 3 \times 2^{\ell-1}.$$

Thus, an interval of length 2^k is exactly halved in a way: $2^k \to 2^{k-1} \to \ldots \to 2 \to 1$.

On the other hand, if we have any other integer, then it is partitioned into a power of 2 and the rest in the manner described above. Because of the definition of the partition, the uniformity is at least 1/2. In fact, if $r = m - 2^k$ is greater than 2^k, then the uniformity is given by

$$2^k/(m - 2^k) \geq 2^k/(3 \times 2^k - 2^k) = 1/2,$$

and if r is at most 2^k then it is given by

$$(m - 2^k)/2^k \geq (3 \times 2^{k-1} - 2^k)/2^k = 1/2.$$

Thus, dividing the interval of length r is *safe* in the sense that it keeps the uniformity $\geq 1/2$. Dividing the interval of length 2^k is also safe since it is divided evenly. □

4 Known Results

Some results are known for the problem defined on real numbers instead of integers. In one dimension, an exact bound on the uniformity is known (1).

Theorem 2 *In one dimension, for any integer $n > 0$ there is a sequence of n points (real numbers) with uniformity $(\frac{1}{2})^{\lfloor n/2 \rfloor/(\lfloor n/2 \rfloor + 1)}$ and also any sequence of n points has* *uniformity at most* $(\frac{1}{2})^{\lfloor n/2 \rfloor/(\lfloor n/2 \rfloor+1)}$. *Such an optimal sequence can be computed in* $O(n)$ *time.*

Theorem 3 *For any integer* $n > 0$, *the greedy algorithm (Voronoi insertion) has uniformity* $1/2$ *in one dimension and* $\sqrt{2}/2$ *in two dimensions.*

5 Uniform Point Sequence

Lemma 1 guarantees that for any pair of integers m and n there is a sequence of n integer points in the interval of length m such that its uniformity is at least $1/2$ if $m > n > 0$. What happens if we want to achieve uniformity strictly greater than $1/2$? Recall that we have defined a uniform sequence to be one with uniformity strictly greater than $1/2$.

First of all, we cannot expect the same property as we have seen in Lemma 1 any more. Suppose we are given an interval $[0, 9]$. Can we find a uniform sequence of 2 points achieving the uniformity strictly greater than $1/2$? The first division is uniquely determined as $\{9\} \to \{4, 5\}$ since none of the other partitions $\{9\} \to \{3, 6\}$, $\{9\} \to \{2, 7\}$, and $\{9\} \to \{1, 8\}$ has uniformity strictly greater than $1/2$. Now, we have to divide the larger gap, 5, into $\{2, 3\}$ since $\{1, 4\}$ is worse. Then, after the second division we have $\{4, 2, 3\}$ whose uniformity is exactly $1/2$. This simple example shows difficulty of this extension. Now we have the following three problems.

Problem 1: Given two integers m and n with $m > n > 0$, determine whether there exists a uniform (n, m)-sequence.
Problem 2: Given an integer $n > 0$, find the smallest integer m such that there is a uniform (n, m)-sequence.
Problem 3: Given an integer $m > 1$, find the largest integer n such that there is a uniform (n, m)-sequence.

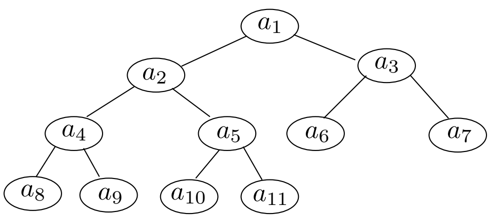
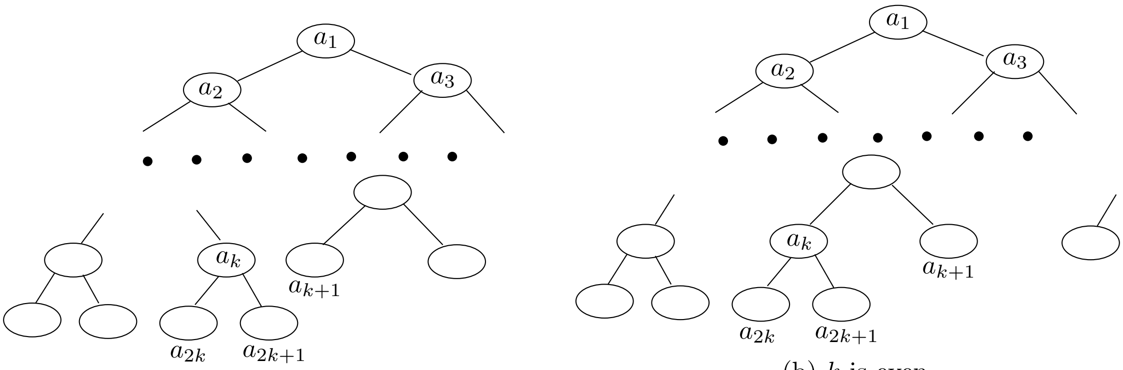


Fig. 3. A tree describing behavior of the algorithm in the proof. A node for an interval of length a_k is divided into two nodes for a_{2k} and a_{2k+1} with $a_k = a_{2k} + a_{2k+1}$. If a_k is the last node having children, then the first leaf node is a_{k+1} and the last one a_{2k+1}.

Let us consider **Problem 2**. In this problem we look for a sequence $(0, m, p_1, \ldots, p_n)$ in an interval $[0, m]$ such that its uniformity is strictly greater than $1/2$. When we insert points $p_1, \ldots, p_n$ in this order into the interval, then we can characterize behavior of the algorithm by how the set of interval lengths changes. We start with the set $\{a_1\}$, where $a_1 = m$. Then, it is partitioned into $\{a_2, a_3\}$ (we assume $a_2 \geq a_3$), and then a_2 is partitioned into $\{a_4, a_5\}$ with $a_4 \geq a_5$. The first important observation here is that we have to partition the longest interval to keep the uniformity $> 1/2$. For dividing an interval that is not longest into two generates an interval of length shorter than half of the longest one, which results in uniformity $< 1/2$. If we always partition the longest interval, then the k-th partition results in the set of interval lengths $\{a_{k+1}, \ldots, a_{2k}, a_{2k+1}\}$. Therefore, it is well described by a tree like a heap (see Figure 3).



(a) k is odd.

(b) k is even.

Fig. 4. Two trees for odd and even integers. The node for a_{k+1} is a left or right child of its parent node depending on whether k is odd or even, respectively.

In the algorithm we divide the intervals of lengths $a_1, a_2, \ldots$ in this order. When we divide a_k, a set of interval lengths is $\{a_k, a_{k+1}, \ldots, a_{2k-2}, a_{2k-1}\}$. Dividing a_k produces two new interval lengths a_{2k} and a_{2k+1}. Since we assume $a_{2k+1} \le a_{2k}$ in our convention, a_{k+1} and a_{2k+1} are the maximum and minimum gaps after the division. Thus, we have

$$a_k = a_{2k} + a_{2k+1}, \text{ and } \frac{a_{2k+1}}{a_{k+1}} > \frac{1}{2}, \quad \text{for} \quad k = 1, 2, \ldots, n.$$

When we are about to divide the longest interval of length a_k, those interval lengths a_k, a_{k+1}, …, a_{2k-2}, a_{2k-1} appear at leaves of the corresponding tree. They are ordered in a way that $a_k \ge a_{k+1} \ge \cdots \ge a_{2k-2} \ge a_{2k-1}$.

Because of the uniformity condition, $a_{2k-1}/a_k > 1/2$ must hold. Since the sum of those values, $a_k + a_{k+1} + \cdots + a_{2k-2} + a_{2k-1}$, is equal to a_1, the length of the original interval, we must minimize the sum to minimize the length of the original interval. What is the smallest value of a_{2k-1}? It depends on whether k is odd or even. See Figure 4.

Case 1: k is odd.

When a_{k+1} is to be divided, the set of interval lengths is $\{a_{k+1}, \ldots, a_{2k}, a_{2k+1}\}$. If we go back to the past divisions, $a_k, a_{k-1}, \ldots$ have been divided. When a_k was divided, we must have had $a_{2k-1}/a_k > 1/2$, that is, $a_{2k-1} > a_k/2$. Since $a_k = a_{2k} + a_{2k+1}$ and we assumed $a_{2k} \ge a_{2k+1}$, it means $a_{2k-1} > a_{2k+1}$. Therefore, a_{2k+1} may be equal to a_{2k}, but it must be strictly smaller than a_{2k-1}. Repeating this argument, we observe that the sum $a_{k+1} + \cdots + a_{2k+1}$ is minimized when $a_{2k+1} = a_{2k}$, $a_{2k} + 1 = a_{2k-1} = a_{2k-2}, \ldots, a_{k+3} + 1 = a_{k+2} = a_{k+1}$. Here note that the node of a_{k+1} is a right child since

k is odd. Taking the constraint $a_{2k+1}/a_{k+1} > 1/2$ into accounts, we can conclude that $a_{2k+1} > (k-1)/2$, that is, a_{2k+1} must be at least $(k+1)/2$.

For the pattern we have

$$\begin{aligned} a_1 &= a_{k+1} + \cdots + a_{2k} + a_{2k+1} \\ &= (3k^2 + 4k + 1)/4. \end{aligned}$$

Case 2: k is even.

The proof proceeds similarly as above, but this time a_{k+1} is not paired. Considering the fact, we have

$$\begin{aligned} a_1 &= a_{k+1} + \cdots + a_{2k} + a_{2k+1} \\ &= (3k^2 + 6k + 4)/4. \end{aligned}$$

The results are summarized in the following theorem.

Theorem 4 *The length of the shortest interval that accepts a uniform point sequence of length n is $(3n^2 + 4n + 1)/4$ if n is odd and $(3n^2 + 6n + 4)/4$ otherwise.*

In a similar manner we can characterize uniform sequences. Using the characterization, it is not so hard to solve the remaining problems. Due to space limit, we omit the detail.

6 Conclusions and Future Works

In this letter we have presented algorithms for generating uniform sequences of points in a given interval. One big difference from the existing study is that points must have integer coordinates. Due to this integrality the problem is now a combinatorial optimization problem. One important extension of our result is to higher dimensions, especially points sets in the plane. Although the problem has a complete solution in one dimension, no optimal solution has been known for point sets in the plane or space. The discrete version of the problem is expected to provide a combinatorial approach to the two-dimensional online discrepancy problem.

References

[1] S. Teramoto, T. Asano, N. Katoh, and B. Doerr, "Inserting Points Uniformly at Every Instance," IEICE Trans. INF. & SYST., Vol. E89-D, 8, pp.2348-2356, 2006.

[2] T. Asano and S. Teramoto, "On-line uniformity of points," Book of Abstracts for 8th Hellenic-European Conference on Computer Mathematics and its Applications, pp. 21-22, Athens, Greece, September, 2007.
[3] B. Chazelle: "The Discrepancy Method: Randomness and Complexity," *Cambridge University Press,* 2000.
[4] J. Matoušek: "Geometric Discrepancy," *Springer,* 1991.