Title	ディスクレパンシーを効率よく求めるアルゴリズムに対する研究
Author(s)	小山, 紘樹
Citation	
Issue Date	2010-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/8925
Rights	
Description	Supervisor:浅野　哲夫，情報科学研究科，修士

An efficient algorithm for computing discrepancy

Hiroki Koyama (0810025)

School of Information Science,
Japan Advanced Institute of Science and Technology

9 February 2010

Keywords: discrepancy, Van der Corput set, sort.

The word discrepancy means " disagreement ", but in field of computational geometry, it means how uniformly points are distributed. For example in one-dimension, points are arranged on the straight line. If the distance to the next point is almost the same for any point, these points are uniformly distributed. Namely, discrepancy corresponding to the point set is low. In two-dimension, we can evaluate how uniformly points are arranged in the plane. However, we cannot simply judge whether points are uniformly distributed from a distance of the point alone like one-dimension. In one-dimension, we can judge it by considering only about the relation to the two neighbors. But, in two-dimension, it is difficult to decide the " neighbors ". Therefore, various methods are considered. For example, when we consider all rectangles including points, in one case it contains a lot of points and in the other case it contains low. The discrepancy can be provided according to these differences. At this time, mostly, the number of points ideally included in the rectangle and the number of actually included points are different. The discrepancy can be calculated according to these differences. And, we can consider not only rectangles but circles and the ovals. Furthermore, there is a method for computing the discrepancy by taking the largest empty circle in the point set, too. Naturally, if the evaluation method changes, the evaluation how points are uniformly distribeted changes, too. Namely discrepancy changes, too.

When considering discrepancy, there are two problems. The first one is a problem on how to evaluate the uniformity of a point set. The second one is a problem on how to generate a point set with high evaluation. In the former, it is interested in how the evaluation changes by changing the figure including points. If all figures are examined, it is limitless. Therefore, we can consider how to choose the figure efficiently. We consider rectangles including points as an example. We have to examine $O(n^4)$ rectangles even when we consider only those rectangles including points on the boundary. We should judge how many points are included. Therefore, it also takes $O(n^5)$ time in a simple method. That is, the computing time of the discrepancy grows very fast when there are a lot of points. Therefore, how efficiently it can be computed becomes a focus in this problem.

Some techniques are known as a method of generating a uniform point set. There are Van der Corput sets and set that inclines points arranged like lattice, etc. as the example. Especially, Van der Corput sets are known the discrepancy is $O(\log n)$. Discrepancy has an application to digital half-toning. It is used when an original image is to be expressed by a limited number of colors. This uniformity becomes important when we want to express brightness levels in a monochrome image. In addition, there is a "Low-discrepancy sequence" which is proved to have low discrepancy. This sequence is used to obtain a numerical value of a definite integral at high speed.

In this thesis, we consider rectangles including a point in a given point set in two-dimension. We compute differences between the number of points that will be ideally included if points are arranged uniformly and the number of points actually included in this rectangle. We present an algorithm for computing the discrepancy for n points in $O(n^3)$ time. Since we do not use any complex technique except sorting, this algorithm can be easily implemented.

In this algorithm, a given point set is sorted in the y-direction. Then, the lower side of a rectangle is fixed by the first point of the sorted point set. Afterwards, we determine the upper side of the rectangle in this sorted order. Now, the upper and lower sides of the rectangle are fixed, and the discrepancy by points between the upper and lower sides is computed. Finally, the maximum discrepancy when the lower and upper sides are fixed is computed by the difference of these discrepancies. When the upper side is moved, the previous points are inserted in the increasing order of their x-coordinate. The discrepancy can be calculated by doing so in $O(n)$ time in the next rectangle. Moreover, there are $O(n^2)$ combinations of the lower and upper sides. Therefore, the discrepancy is finally computed in $O(n^3)$ time.

In addition, the discrepancy of an actual points is computed by using this algorithm in this thesis. There are four kinds of points to compute the discrepancy about random points, points arranged like lattice, set that inclines points arranged like lattice, and Van der Corput sets. Points arranged like lattice is known as a point set has high discrepancy. Set that inclines points arranged like lattice and Van der Corput sets are known as a point set has low discrepancy. The discrepancy of each points is computed while changing the number of points. The relation between the number of points and the discrepancy is investigated by this calculation. And, this calculation confirms that the discrepancy of set that inclines points arranged like lattice and Van der Corput sets is actually low.