Online problems

Online clustering problems

String clustering

Further questions

Online string clustering algorithms

E. Bittner, Cs. Imreh, A. Tomescu

Institute of Informatics University of Szeged
University of Helsinki

Online problems

Online problems
Online clustering problems
String clustering
Further questions

The input is given part by part and the algorithm has to make the decisions without any information on the further parts.

The first published online problem is in the Greek mythology.

The performance of an algorithm is measured by the competitive analysis or by an average case analysis.

An algorithm for a minimization problem is c-competitive if its cost is at most c - times more than the optimal cost.

The first analysis for an online scheduling algorithm was done by Graham in 1966. Since 1980 many results have been achieved and several areas have been developed.

Online unit covering

Online problems
Online clustering problems
String clustering
Further questions

In unit covering, a set of n points needs to be covered by balls of unit radius, and the goal is to minimize the number of balls used.

Charikar et al (2004) gave an upper bound of $O(2^d d \log d)$ and a lower bound of $\Omega(\log d / \log \log \log d)$ on the competitive ratio of deterministic online algorithms in d dimensions. This problem is strictly online in the sense that points arrive one by one, each point needs to be assigned to a ball upon arrival, and if it is assigned to a new ball, the exact location of this ball is fixed at this time.
The tight bounds on the competitive ratio for $d = 1$ and $d = 2$ are respectively 2 and 4.

Online unit clustering on line

In unit clustering the online algorithm is not required to fix the exact position of each ball in advance. The algorithm needs to make sure that a set of points which is assigned to one ball (cluster) can always be covered by that ball, thus the ball can be shifted if necessary.

- Chan and Zarrabi-Zadeh (2009) 2-competitive algorithm for line
- Chan and Zarrabi-Zadeh (2009) 16/11 -competitive randomized algorithm for line
- Epstein, van Stee (2010) 7/4-competitive algorithm for line, 8/5 lower bound on the possible competitive ratio for line
- Ehmsen, Larsen (2010) 5/3-competitive algorithm for line
- in two dimensional problems, usually the L_∞ norm is considered

Online facility location

Online problems
Online clustering problems
String clustering
Further questions

In the facility location problem a metric space is given with a multiset of demand points (elements of the space). The goal is to find a set of facility locations in the metric space which minimizes the sum of the facility cost and assignment cost.

- Meyerson (2001): No constant competitive algorithm exists, An O(log n)- competitve randomized algorithm which is constant - competitive algorithm for randomly ordered inputs
- Fotakis (2003,2007): An $O(log(n)/loglog(n))$-competitive algorithm and a matching lower bound on the possible competitive ratio.
- Anagnostopoulos et al (2004): A simpler O(log n)-competive algorithm, the first average case analysis
- Fotakis (2006) Divéki and Imreh (2010): Facility location with facility movements

Online clustering with variable sized clusters I.

Online problems
Online clustering problems
String clustering
Further questions

In *clustering to minimize the sum of diameters with a fixed cost* an input is given consisting of n requests which are points in a line and the goal is to partition the points into groups called *clusters*. The cost of a cluster C is defined as $1 + \max_{i,j \in C} |i - j|$, that is, the sum of a fixed cost which is scaled to 1, and the diameter of the cluster. The goal function is to find a partition of the input into clusters so that the total cost of the clusters is minimized.

In a flexible model, when a new cluster is opened we need to specify its label, but its coordinates as well as its diameter might be changed by the algorithm in the future. For this model the cost of a cluster may change as new points are assigned to it. In the strict model, when a new cluster is opened we need to specify the coordinates of the interval which will be associated with this cluster

Results on online clustering with variable sized clusters

- Csirik, Epstein, Imreh, Levin (2010): A $\phi = (1+\sqrt{5})/2$-competitive algorithm and matching lower bound in the flexible model. A $1+\sqrt{2}$ -competitive algorithm and matching lower bound in the flexible model (and also in an intermediate model). Matching results in the semi online model of increasing sequences.
- Divéki, Imreh (2011): Analysis of grid based algorithms, in two dimensions with square cost clusters, 9 and 7 competitive algorithms for the strict and the flexible models.
- Fotakis and Koutris (2011): An $\Omega(\log n)$ lower bound in two dimensions with linear cost clusters, and an $O(\log n)$- competitive algorithm.

String clustering with fixed sized clusters

In this model we have to divide a sequence of n-dimensional bitvectors into the minimal number of clusters where each cluster has diameter at most k by the Hamming distance.

In the online model the strings arrive one by one and after the arrival a string we have to assign it to an already existing cluster or to define a new cluster for it.

Greedy algorithm If the string can be assigned to a cluster assign to the first such cluster. Otherwise open a new cluster for it.

Theorem The competitive ratio of Greedy is $3/2$ if $k = 1$, and no online algorithm can have smaller competitive ratio than $3/2$.

Greedy for k=2

Online problems
Online clustering problems
String clustering
Further questions

Theorem Greedy is $\Theta(n)$-competitive if $k = 2$.

Proof idea Since the optimal clusters contain at most $n + 1$ elements it is $O(n)$ -competitive.

But it is not better. Suppose that a sequence $(0, 0, x_i)$,$(1, 1, x_i)$ arrives $i = 1, \ldots, n - 1$ where the set x_1, x_2, x_{n-1} is a set of $n - 2$ dimensional strings having diameter 2.

Then the greedy algorithm forms $n - 1$ clusters each containing two elements.

The optimal solution has two clusters one contains the strings started by $(0, 0)$ the other contains the strings started by $(1, 1)$.

A constant-competitive algorithm?

There are 3 types of clusters for $k = 2$, the faces, the tetrahedrons, and the centered one radius clusters. Only the centered one radius clusters can cause a problem.

Greedy 2 If the new string does not fit any of the opened clusters open a new one radius cluster around it.

It is easy to see that Greedy 2 is not constant competitive considering a one radius cluster without the center.

Conjecture There is a constant competitive algorithm for $k = 2$ which is based on guessing the optimal one radius clusters.

Connection to graph coloring

Online problems
Online clustering problems
String clustering
Further questions

We can define the following graph of restrictions. The set of vertices is the set of strings. Two vertices are connected if the distance of the strings is greater than k. Then our problem is to find the coloring of the graph with minimal colors (the color classes are the clusters).

Corollary The positive results from graph coloring can be applied, but they are very weak. No constant online graph coloring exists. If $P \neq NP$ no constant approximation offline graph coloring algorithm exists.

String clustering with fixed number of clusters

Online problems
Online clustering problems
String clustering
Further questions

In this model we suppose that we can use at most k clusters. The goal is to minimize the maximal diameter. The following algorithm defines only such clusters which has a center.

Algorithm CSC(b) Constrained Sized Clustering:

- If the distance of the new string and one of the centers is less than b then we assign the new string to the cluster of the first such center.
- If no such open cluster exists and we have less than k clusters then we open a new cluster and its center will be this new string.
- Otherwise we consider the closest center and assign the string to its cluster.

The upper bound

Online problems
Online clustering problems
String clustering
Further questions

Theorem $CSC(\sqrt{n})$ is $O(\sqrt{n})$-competitive

Proof: Consider an input sequence. If the maximal diameter of $CSC(\sqrt{n})$ is at most $2\sqrt{n}$, then the result follows since OPT is at least 1. Otherwise consider the first point from each cluster, and the first point which is further than $\sqrt{n}$ from each of the k centers.
This gives us $k+1$ points where their pairwise distance is greater than $\sqrt{n}$. OPT has to put two of them into the same cluster, thus its cost is at least $\sqrt{n}$. But the cost of CSC is at most n and the theorem follows.

The lower bound

Online problems
Online clustering problems
String clustering
Further questions

Theorem: No online algorithm can have smaller competitive ratio than $\Omega(\sqrt{n})$ for string clustering with fixed number of clusters.

Proof: Suppose we have an algorithm which is $o(\sqrt{n})$-competitive. Let the first string is $(0, \ldots, 0)$, and the second string has 1 in the first $\sqrt{n}$ positions and 0 in the others. Then the algorithm must assign them two different clusters.

Continue the sequence with $k-1$ strings where the i-th of them contains n/k 1 in the positions $in/k+1, in/k+2, \ldots in/k+n/k$ and 0 in the other positions.

Then the optimal solution uses one cluster for the first two items and a further cluster for each of the other elements, and has cost $\sqrt{n}$. The online algorithm has a cost of at least n/k.

Online string clustering with variable sized clusters

Online problems
Online clustering problems
String clustering
Further questions

In this model neither the number nor the size of clusters is fixed. The goal is to minimize a weighted sum of the number of clusters and the maximal size.

Algorithm FSC(b) (fixed sized cluster): If we can assign the new string to an opened cluster which does not have larger diameter than b after the assignment then assign it to the first such cluster. Otherwise open a new cluster for it.

Theorem FSC(b) is not constant competitive for any b.

Proof idea: If b is large then we can use two strings with distance $b+1$. Otherwise we can use the $2^{n/(b+1)}$ strings which can be built from the $b+1$ sized blocks.

Further questions

Online problems

Online clustering problems

String clustering

Further questions

- decreasing the gaps
- resource augmentation
- randomized algorithm
- sum of the diameters instead of the maximum
- other special metric space
- other clustering problems

Acknowledgements

TÁMOP-4.2.2/B-10/1-2010-0012

The presentation is supported by the European Union and co-funded by the European Social Fund.
Project title: "Broadening the knowledge base and supporting the long term professional sustainability of the Research University Centre of Excellence at the University of Szeged by ensuring the rising generation of excellent scientists."
Project number: TÁMOP-4.2.2/B-10/1-2010-0012