Theoretical Computer Science 639 (2016) 60–71

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Kinetic clustering of points on the line ☆

Cristina G. Fernandes, Marcio T.I. Oshiro *

Department of Computer Science, University of São Paulo, Brazil

ARTICLE INFO

Article history:
Received 24 October 2015
Received in revised form 9 May 2016
Accepted 14 May 2016
Available online 25 May 2016
Communicated by A. Marchetti-Spaccamela

Keywords:
Kinetic clustering
Line trajectory
Approximation algorithms
Computational geometry

ABSTRACT

The problem of clustering a set of points moving on the line consists of the following: given positive integers n and k, the initial position and the velocity of n points, find an optimal k-clustering of the points. We consider two classical quality measures for the clustering: minimizing the sum of the clusters diameters and minimizing the maximum diameter of a cluster. For the former, we present polynomial-time algorithms under some assumptions and, for the latter, a $(2.71 + \varepsilon)$-approximation.

1. Introduction

Clustering refers to a well-known class of problems whose goal is to partition a set into groups of "similar" elements. The notion of similarity and the format of the partition depend on the application. In this work, we study two clustering problems in a kinetic context, where points move continuously.

There is a plethora of works on kinetic variants of clustering in the literature. Variants might differ both on the type of clustering searched and on the type of movement allowed for the points. Some variants search for a good static clustering for the moving points, while others search for a good clustering that adapts as the points move, sometimes in a controlled way. As in the static case, there are several possibilities for measuring the quality of a clustering, such as the maximum diameter of one of its clusters, or the sum of the diameters of the clusters. In what follows, we focus on results from the literature that relate more closely to ours.

Atallah [1] proposed a model for the points movement where the points are in a d-dimensional space and each coordinate of each point is given by a polynomial on the time variable. Several works adopt this model, sometimes restricting the degree of the polynomial to be bounded by a (usually small) constant. Another model for the points movement was introduced by Basch, Guibas, and Hershberger [2], and allows for changes in the description of a point movement. Specifically, the motion of a point is given by a piecewise function of constant algebraic degree, known as the point *flight plan*. One particular case of this model that is often considered in the literature is the case in which each flight plan consists of a piecewise linear function on the time variable.

Using Atallah's model, Har-Peled [3] showed how to apply a clustering algorithm for the static setting to find a competitive static clustering of the moving points. His objective was to find k centers that cover all the points within a minimum

☆ Research partially supported by CAPES 31005668825, CNPq 308523/2012-1, FAPESP 2013/03447-6, and Project MaCLinC of NAP-NUMEC/MaCLinC-USP 11.1.29453.01.4.

* Corresponding author.
E-mail addresses: cris@ime.usp.br (C.G. Fernandes), oshiro@ime.usp.br (M.T.I. Oshiro).

http://dx.doi.org/10.1016/j.tcs.2016.05.020
0304-3975/

radius. When the polynomials describing the points movement have degree at most μ, his algorithm relaxes the restriction on the number of clusters, allowing at most $k^{\mu+1}$ clusters, in order to achieve a constant approximation ratio with respect to the optimal radius of a k-clustering at any time. Hellweg and Sohler [4] considered the variant of Har-Peled's problem in which the clustering quality measure is the one used in the k-means, that is, the sum of the square of the radius of the clusters, and they presented an algorithm that produces a static $k^{\mu+1}$-clustering for a set of moving points that approximates, at all times, the k-means optimal clustering for these points.

Basch, Guibas, and Hershberger [2] introduced a framework, known as KDS (for kinetic data structure), to efficiently keep an attribute, such as the convex hull or a current pair of closest points, for a set of moving points. This framework was used to provide good clusterings for sets of moving points. For instance, Gao, Guibas, Hershberger, Zhang, and Zhu [5] proposed a randomized algorithm to maintain, as the points move, a constant approximation for a clustering minimizing the number of discrete centers needed to cover all points within a fixed radius. Gao, Guibas, and Nguyen [6] gave a KDS to maintain an 8-approximation for the kinetic k-center. Friedler and Mount improved on this result by presenting a KDS that maintains a $(4+\varepsilon)$-approximation for the kinetic robust k-center (the robust k-center is a generalization of the k-center that accounts for outliers). Bespamyatnikh, Bhattacharya, Kirpatrick, and Segal [7] presented KDS's for 1-center and 1-median of a set of points moving in the plane.

In the first class of mentioned results, one looks for a static clustering for a set of moving points, while in the second class the goal is to keep an (approximately) optimal clustering at all times. With Schabanel [8], we proposed an intermediate version between these two goals, that allows the clustering to change over time, but charges for this, imposing a certain stability in the clustering. The amount one charges for the changes, called *instability cost*, allows for more changes, or less, in the clustering over time. Even in the 1-dimensional case addressed in [8], this variant of the problem turned out quite difficult for all considered clustering quality measures, and even achieving approximations in general seems challenging. Thus it was natural to consider the extreme cases of the instability cost, namely, zero and infinite. The zero instability cost leads to easy problems, as it is enough to solve the corresponding static 1-dimensional clustering problem in each time instant, and that is easy for the considered clustering quality measures. On the other hand, the infinite instability cost leads to the case of looking for a good static clustering of the moving points, which is the case we address in this paper.

So our work is closer to the ones of Har-Peled, and Hellweg and Sohler, as we also look for a static clustering of moving points, instead of a way to keep an (almost) optimal k-clustering all the time. But the goal is slightly different. We do not look for a static clustering that is approximately optimal at each time instant, but for a static k-clustering which is approximately optimal when compared to the best static k-clustering. Specifically, we consider the 1-dimensional case ($d=1$), linear movements ($\mu=1$), and two classical quality measures for the clustering: minimizing the sum of the clusters diameters and minimizing the maximum diameter of a cluster. The diameter of a cluster is measured over the total time period, namely, it is the sum (or integral) of the diameter of the cluster over time. For the first quality measure, we present a polynomial-time algorithm under some assumptions and, for the second one, a $((4+\sqrt{2})/2+\varepsilon)$-approximation for every $\varepsilon>0$.

In Section 2, we formalize our model for the movement of the points and give the definition of the diameter of a cluster in our setting, to precisely state the two variants of clustering we address. In Section 3, we present the polynomial-time algorithm for the first variant and, in Section 4, we present the approximation for the second variant. Some final comments and open problems are stated in Section 5.

2. One dimensional kinetic model and the problems

In our kinetic model, n points move with uniform rectilinear velocity during a continuous time interval. Without loss of generality, the time interval is $[0,1]$. Each point $i \in \{1,2,\ldots,n\}$ has an initial position $x_i(0)$ and its velocity is given by a vector v_i. We only consider points in $\mathbb{R}$, so the position and the velocity are real numbers. A positive/negative velocity indicates a movement to the right/left respectively. This is a particular case of the KDS [2] and Atallah's model [1].

At an instant t in $[0,1]$, the position $x_i(t)$ of a point i with initial position $x_i(0)$ and velocity v_i is given by the function

$$x_i(t) = x_i(0) + v_i t.$$

This function represents a segment on the Cartesian plane, called the *trajectory* of point i, and is given by the pair $(x_i(0), v_i)$. We draw the Cartesian plane with the horizontal axis representing the position x and the vertical axis representing the time t. Since the time interval is always $[0,1]$, the strip of the plane between $t=0$ and $t=1$ will be called *time-strip*.

For our purpose, no two points have the same trajectory, or they can be treated as one. Hence, we assume a one-to-one relation between moving points and their trajectories, and mostly refer to trajectories instead of moving points in what follows.

Given a finite set S of trajectories, a *cluster* is a subset of S and a *k-clustering* is a partition of S into k clusters. Note that, as a cluster might be empty, every k'-clustering for $k'<k$ corresponds to a k-clustering by adding $k-k'$ empty clusters. Conversely, any k-clustering of S with more than $|S|$ clusters may be converted into an $|S|$-clustering by disregarding some empty clusters. So we may assume that $1 \leqslant k \leqslant |S|$. The *left side* of a nonempty cluster C is the piecewise linear function $\min_{i\in C} x_i(t)$ for $t \in [0,1]$. Analogously, the *right side* is $\max_{i\in C} x_i(t)$ for $t\in[0,1]$. The *span* of a cluster C, span(C), is empty if C is empty, otherwise it is the region within the time-strip bounded by the left and right sides of C. The *diameter* of C is the area of its span, denoted by diam(C). See Fig. 1.