Worst-Case and Smoothed Analysis of the Hartigan–Wong Method for k-Means Clustering

Bodo Manthey

Faculty of Electrical Engineering, Mathematics, and Computer Science, University of Twente, Enschede, The Netherlands

Jesse van Rhijn

Faculty of Electrical Engineering, Mathematics, and Computer Science, University of Twente, Enschede, The Netherlands

Abstract

We analyze the running time of the Hartigan–Wong method, an old algorithm for the k-means clustering problem. First, we construct an instance on the line on which the method can take $2^{\Omega(n)}$ steps to converge, demonstrating that the Hartigan–Wong method has exponential worst-case running time even when k-means is easy to solve. As this is in contrast to the empirical performance of the algorithm, we also analyze the running time in the framework of smoothed analysis. In particular, given an instance of n points in d dimensions, we prove that the expected number of iterations needed for the Hartigan–Wong method to terminate is bounded by $k^{12kd} \cdot \text{poly}(n, k, d, 1/\sigma)$ when the points in the instance are perturbed by independent d-dimensional Gaussian random variables of mean 0 and standard deviation σ.

2012 ACM Subject Classification Theory of computation → Randomness, geometry and discrete structures; Theory of computation → Approximation algorithms analysis; Theory of computation → Discrete optimization

Keywords and phrases k-means clustering, smoothed analysis, probabilistic analysis, local search, heuristics

Digital Object Identifier 10.4230/LIPIcs.STACS.2024.52

Related Version *Full Version*: `https://arxiv.org/abs/2309.10368`

Funding *Jesse van Rhijn*: Supported by NWO grant OCENW.KLEIN.176.

1 Introduction

Clustering is an important problem in computer science, from both a practical and a theoretical perspective. On the practical side, identifying clusters of similar points in large data sets has relevance to fields ranging from physics to biology to sociology. Recent advances in machine learning and big data have made the need for efficient clustering algorithms even more apparent. On the theoretical side, clustering problems continue to be a topic of research from the perspective of approximation algorithms, heuristics, and computational geometry.

Perhaps the best-studied clustering problem is that of k-means clustering. In this problem, one is given a finite set of points $\mathcal{X} \subseteq \mathbb{R}^d$ and an integer k. The goal is to partition the points into k subsets, such that the sum of squared distances of each point to the centroid of its assigned cluster, also called its cluster center, is minimized.

Despite great effort to devise approximation algorithms for k-means clustering, the method of choice remains Lloyd's method [11]. This method starts with an arbitrary choice of centers, and assigns each point to its closest center. The centers are then moved to the centroids of each cluster. In the next iteration, each point is again reassigned to its closest center, and the process repeats.

41st International Symposium on Theoretical Aspects of Computer Science (STACS 2024).
Editors: Olaf Beyersdorff, Mamadou Moustapha Kanté, Orna Kupferman, and Daniel Lokshtanov; Article No. 52; pp. 52:1–52:16
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

It is not hard to show that this process strictly decreases the objective function whenever either a cluster center changes position, or a point is reassigned. Hence, no clustering can show up twice during an execution of this algorithm. Since the number of partitions of n points into k sets is at most k^n, the process must eventually terminate.

Although Lloyd's method has poor approximation performance both in theory and in practice [2], its speed has kept it relevant to practitioners. This is in startling contrast to its worst-case running time, which is exponential in the number of points [15].

To close the gap between theory and practice, Arthur et al. have shown that Lloyd's method terminates in expected polynomial time on perturbed point sets, by means of a smoothed analysis [1]. This provides some theoretical justification for the use of Lloyd's method in practice.

Another, less well-known heuristic for clustering is the Hartigan–Wong method [8]. In this method, one proceeds point-by-point. Given an arbitrary clustering, one checks whether there exists a point that can be reassigned to a different cluster, such that the objective function decreases. If such a point exists, it is reassigned to this new cluster. If no such points exist, the algorithm terminates and the clustering is declared locally optimal.

Although at first sight the Hartigan–Wong method might seem like a simpler version of Lloyd's method, it is qualitatively different. If Lloyd's method reassigns a point x from cluster i to cluster j, then x must be closer to the center of cluster j than to that of cluster i. In the Hartigan–Wong method, this is not true; x may be reassigned even when there are no cluster centers closer to x than its current center. This can be beneficial, as Telgarsky & Vattani showed that the Hartigan–Wong method is more powerful than Lloyd's method [14].

To be precise, every local optimum of the Hartigan–Wong method is also a local optimum of LLoyd's method, while the converse does not hold. Telgarsky & Vattani moreover performed computational experiments, which show that the Hartigan–Wong method not only tends to find better clusterings than Lloyd's, but also has a similar running time on practical instances. Despite these promising results, theoretical knowledge of the Hartigan–Wong method is lacking.

In this paper, we aim to advance our understanding of this heuristic. Our contributions are twofold. First, we construct an instance on the line on which the Hartigan–Wong method can take $2^{\Omega(n)}$ iterations to terminate. Considering that k-means clustering can be solved exactly in polynomial time in $d = 1$, this shows that the worst-case running time of the Hartigan–Wong method is very poor even on easy instances. This is in contrast to Lloyd's method, where all known non-trivial lower bounds require $d \geq 2$.

▶ **Theorem 1.** *For each $m \in \mathbb{N}_{\geq 2}$ there exists an instance of k-means clustering on the line with $n = 4m - 3$ points and $k = 2m - 1$ clusters on which the Hartigan–Wong method can take $2^{\Omega(n)}$ iterations to converge to a local optimum.*

Second, we attempt to reconcile Theorem 1 with the observed practical performance of the Hartigan–Wong method. We perform a smoothed analysis of its running time, in which each point in an arbitrary instance is independently perturbed by a Gaussian random variable of variance σ^2.

▶ **Theorem 2.** *Let $n, k, d \in \mathbb{N}$, and assume $4kd \leq n$. Fix a set of n points $\mathcal{Y} \subseteq [0,1]^d$, and assume that each point in $\mathcal{Y}$ is independently perturbed by a d-dimensional Gaussian random variable with mean 0 and standard deviation σ, yielding a new set of points $\mathcal{X}$. Then the expected running time of the Hartigan–Wong method on $\mathcal{X}$ is bounded by*

$$O\left(\frac{k^{12kd+5} d^{12} n^{12.5+\frac{1}{d}} \ln^{4.5}(nkd)}{\sigma^4}\right) = k^{12kd} \cdot \mathrm{poly}(n, k, d, 1/\sigma).$$

Although we do not attain a polynomial smoothed running time in all problem parameters, we note that for Lloyd's method one of the first smoothed analyses yielded a similar $k^{kd}\operatorname{poly}(n, 1/\sigma)$ bound. This was later improved to $\operatorname{poly}(n, k, d, 1/\sigma)$. We therefore regard Theorem 2 as a first step to settling the conjecture by Telgarsky & Vattani that the Hartigan–Wong method, like Lloyd's method, should have polynomial smoothed running time.

We note that Theorem 1 shows that there exists an instance on which there exists some very specific sequence of iterations that has exponential length. In essence, this means that the exponential running time is only shown for a very specific pivot rule for choosing which point to reassign to which cluster in each iteration. By contrast, Theorem 2 holds for *any* pivot rule, not simply for any particular choice.

2 Preliminaries and Notation

Given vectors $x, y \in \mathbb{R}^d$, we write $\langle x, y\rangle$ for the standard Euclidean inner product on $\mathbb{R}^d$, and $\|x\| = \sqrt{\langle x, x\rangle}$ for the standard norm.

Given a set of k clusters $\mathcal{C} = \{\mathcal{C}_1, \ldots, \mathcal{C}_k\}$, a configuration of a cluster $\mathcal{C}_i \in \mathcal{C}$ is an assignment of a set of points to $\mathcal{C}_i$. We will denote the clusters by calligraphic letters, and their configurations by regular letters; i.e., the configuration of $\mathcal{C}_i$ will be denoted C_i. This distinction is sometimes useful. For the majority of this paper, however, we will not make this distinction explicitly, and will refer to both a cluster and its configuration interchangeably by regular letters.

Given a finite set of points $S \subseteq \mathbb{R}^d$, we define the center of mass of S as

$$\operatorname{cm}(S) = \frac{1}{|S|}\sum_{x\in S} x.$$

With this definition, we can formally define the objective function of k-means. Let $C = \{C_i\}_{i=1}^k$ be a partition of a finite set of points $\mathcal{X} \subseteq \mathbb{R}^d$. Then the objective function of k-means is

$$\Phi(C) = \sum_{i=1}^{k}\sum_{x\in C_i}\|x - \operatorname{cm}(C_i)\|^2 = \sum_{i=1}^{k}\Phi(C_i),$$

where we define $\Phi(C_i) = \sum_{x\in C_i}\|x - \operatorname{cm}(C_i)\|^2$. We will also refer to $\Phi(C)$ as the potential function.

For both the worst-case and smoothed complexity bounds, we need to analyze the improvement of a single iteration. Thus, we need a simple expression for this quantity. Lemmas 3 and 4 allow us to obtain such an expression. These results were already obtained by Telgarsky & Vattani [14].

▶ **Lemma 3** (Telgarsky & Vattani [14]). *Let S and T be two disjoint nonempty sets of points in $\mathbb{R}^d$. Then*

$$\Phi(S\cup T) - \Phi(S) - \Phi(T) = \frac{|S|\cdot|T|}{|S|+|T|}\cdot\|\operatorname{cm}(S) - \operatorname{cm}(T)\|^2.$$

▶ **Lemma 4** (Telgarsky & Vattani [14]). *Let S and T be two disjoint nonempty sets of points in $\mathbb{R}^d$ with $|S| > 1$. Suppose we move a point $x \in S$ from S to T. Then*

$$\Phi(S\setminus\{x\}) + \Phi(T\cup\{x\}) - \Phi(T) - \Phi(S) = \frac{|T|}{|T|+1}\|\operatorname{cm}(T) - x\|^2 - \frac{|S|}{|S|-1}\|\operatorname{cm}(S) - x\|^2.$$

Let C be some clustering of $\mathcal{X}$. Suppose in some iteration of the Hartigan–Wong method, we move $x \in C_i$ to C_j. Let the gain of this iteration be denoted $\Delta_x(C_i, C_j)$. Then Lemma 4 tells us that

$$\Delta_x(C_i, C_j) = \frac{|C_i|}{|C_i| - 1}\|x - \mathrm{cm}(C_i)\|^2 - \frac{|C_j|}{|C_j| + 1}\|x - \mathrm{cm}(C_j)\|^2.$$

At first sight, it seems like Lemma 4 leaves open the possibility that a cluster is left empty. The following lemma shows that this can never happen.

▶ **Lemma 5.** *No iteration can leave a cluster empty.*

Proof. Suppose before an iteration, $C_i = \{x\}$ for some $x \in X$, and after the iteration $C_i' = \emptyset$ and $C_j' = C_j \cup \{x\}$, i.e. x is moved from cluster i to cluster j. The gain of this iteration is then (Lemma 3)

$$\Phi(C_i) + \Phi(C_j) - \Phi(\emptyset) - \Phi(C_j \cup \{x\}) = \Phi(C_j) - \Phi(C_j \cup \{x\}) = -\frac{|C_j|}{|C_j| + 1}\|x - \mathrm{cm}(C_j)\|^2 \leq 0,$$

since $\mathrm{cm}(C_i) = x$ and $\Phi(\emptyset) = 0$. Since every iteration must improve the clustering, this concludes the proof. ◀

3 Exponential Lower Bound

In this section, we construct a family of k-means instances on the line on which the Hartigan–Wong method can take an exponential number of iterations before reaching a local optimum. To be precise, we prove the following theorem.

▶ **Theorem 1** (Restated). *For each $m \in \mathbb{N}_{\geq 2}$ there exists an instance of k-means clustering on the line with $n = 4m - 3$ points and $k = 2m - 1$ clusters on which the Hartigan–Wong method can take $2^{\Omega(n)}$ iterations to converge to a local optimum.*

The construction we employ is similar to the construction used by Vattani for Lloyd's method [15]. However, the Hartigan–Wong method only reassigns a single point in each iteration, and we are free to choose which point we reassign. Moreover, we are even free to choose which cluster we move a point to if there are multiple options. This allows us to simplify the construction and embed it in a single dimension, rather than the plane used by Vattani.

We define a set of m gadgets G_i, $i \in \{0, \ldots, m-1\}$. Each gadget except for the "leaf" gadget G_0 consists of four points, and has two clusters $G_i(\mathcal{C}_0)$ and $G_i(\mathcal{C}_1)$ associated with it. Moreover, each gadget except G_0 has three distinguished states, called "morning", "afternoon", and "asleep". The leaf gadget only has two states, "awake" and "asleep ".

During the morning state, a gadget G_i watches G_{i-1}. If G_{i-1} falls asleep, then it is awoken by G_i; this is achieved by moving a point of G_i to one of the clusters of G_{i-1}. This allows G_{i-1} to perform a sequence of iterations, which ends with G_{i-1} back in its morning state.

Meanwhile, G_i performs a sequence of iterations that transition it to its afternoon state. During the afternoon state, it once more watches G_{i-1}. When the latter falls asleep, G_i once again wakes G_{i-1}, and transitions itself to its asleep state.

The leaf gadget G_0, as it does not watch any gadgets, only ever awakens and immediately falls asleep again.

We end the sequence of iterations once gadget $m-1$ falls asleep. Observe that with this construction, G_i falls asleep twice as often as G_{i+1}. With the condition that G_{m-1} falls asleep once, we obtain a sequence of at least 2^{m-1} iterations. With $n = 4m-3$, this yields Theorem 1.

For space reasons, we only describe the instance and the exponential-length sequence here. The proof that this sequence is improving, which completes the proof of Theorem 1, is deferred to the full version.

Formal Construction

We now give a detailed construction of a unit gadget, G. All gadgets except for G_0 are scaled and translated versions of G. The unit gadget is a tuple $G = (S, \mathcal{C}_0, \mathcal{C}_1)$, where $S = \{a, b, p, q\} \subseteq \mathbb{R}$, and $\mathcal{C}_0$ and $\mathcal{C}_1$ are two clusters. The positions of the points in S are given in Table 1. In addition, the gadget is depicted schematically in Figures 1 and 2. Note that the relative positions of the points in these figures do not correspond to Table 1, but are chosen for visual clarity.

Table 1 Positions of the points in $S(G)$, the leaf point f, and the translation vector t_0 between gadgets G_1 and G_2.

Point	a	b	p	q	f	t_0
Position	9	6	5	13	0	8

We remark that the points in Table 1 are not simply chosen by trial-and-error. As will be explained shortly, we can obtain from our construction a series of inequalities that must be satisfied by the points in S. We then obtained these points by solving the model

$$\begin{aligned} \min \quad & a^2 + b^2 + p^2 + q^2 + f^2 + t_0^2 \\ \text{s.t.} \quad & \text{each move decreases the clustering cost,} \\ & a, b, p, q, f, t_0 \in \mathbb{Z} \end{aligned}$$

using Gurobi [7]. The first constraint here amounts to satisfying a series of inequalities of the form $\Delta_x(A, B) > 0$ for $x \in S(G)$ and A, B subsets of the points in a gadget and its neighboring gadgets. For space reasons, we defer their derivation and verification to the full version. The objective function here is purely chosen so that Gurobi prefers to choose small integers in the solution.

To construct G_i from the unit gadget (for $i \geq 1$), we scale the unit gadget by a factor 5^{i-1}, and translate it by $t_i = \sum_{j=0}^{i-1} 5^j t_0$, where $t_0 = 8$. Since each gadget only ever exchanges points with its neighbors in the sequence we are about to construct, it will suffice in proving Theorem 1 to consider only iterations involving G_i, G_{i-1} and G_{i+1} for some fixed $i > 2$. For the leaf gadget, we simply have $G_0 = (S_0, \mathcal{C}_0)$, where $S_0 = \{f\} = \{0\}$.

Before we go on to construct an improving sequence of exponential length, we define the earlier-mentioned states. For ease of notation, we will refer to the points of G_i as a_i, b_i, and so on, and to the clusters of G_i as $\mathcal{C}_0(G_i)$ and $\mathcal{C}_1(G_i)$. Then we say the state of $G_{i>0}$ is:

- asleep, if $C_0(G_i) = \{b_i\}$ and $C_1(G_i) = \{a_i, q_i\}$ (in this state, p_i is in some cluster of G_{i-1});
- morning, if $C_0(G_i) = \{p_i, q_i, b_i\}$ and $C_1(G_i) = \{a_i\}$;
- afternoon, if $C_0(G_i) = \{b_i\}$ and $C_1(G_i) = \{p_i, q_i, a_i\}$.

For the leaf gadget, we say its state is:

- asleep, if $C_0(G_0) = \{f\}$;
- awake, otherwise.

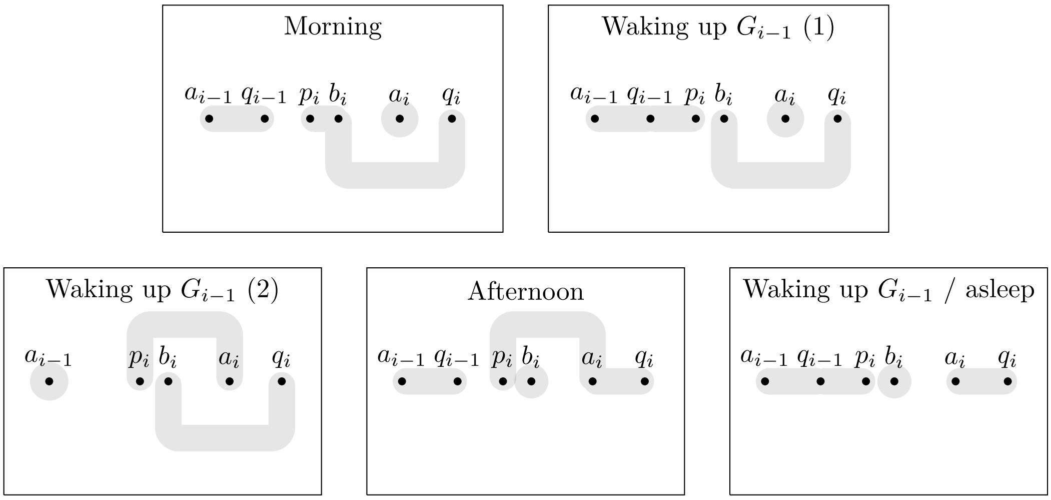
Morning
a_{i-1} q_{i-1} p_i b_i a_i q_i
Waking up G_{i-1} (1)
a_{i-1} q_{i-1} p_i b_i a_i q_i
Waking up G_{i-1} (2)
a_{i-1} p_i b_i a_i q_i
Afternoon
a_{i-1} q_{i-1} p_i b_i a_i q_i
Waking up G_{i-1} / asleep
a_{i-1} q_{i-1} p_i b_i a_i q_i

Figure 1 Schematic depiction of the interactions between G_i and G_{i-1} during the morning and afternoon phases of G_i.

We now explicitly determine a sequence of iterations of exponential length. In the proof of Theorem 1, we show that this sequence is improving. To analyze the sequence, we consider the perspective of G_i as it wakes up G_{i-1} and falls asleep; and then as it is awoken by G_{i+1}. We first consider only the case that $G_{i-1} \neq G_0$. See Figure 1 and Figure 2 for a schematic depiction of the sequence described below.

Morning

We start with G_i in the morning state, and G_{i-1} asleep. To wake up G_{i-1}, the point p_i moves to $\mathcal{C}_1(G_{i-1})$, which currently contains a_{i-1} and q_{i-1}. This triggers the wakeup phase of G_{i-1}; we will analyze this phase later from the perspective of G_i. When the wakeup phase completes, $\mathcal{C}_1(G_{i-1})$ contains a_{i-1} and p_i, and p_i moves to $\mathcal{C}_1(G_i)$. Subsequently q_i moves from $\mathcal{C}_0(G_i)$ to $\mathcal{C}_1(G_i)$. Observe that this puts G_i into the afternoon state.

Afternoon

In this state, G_i is once again watching G_{i-1}. Once the latter falls asleep, p_i moves from $\mathcal{C}_1(G_i)$ to $\mathcal{C}_1(G_{i-1})$, which triggers another wakeup phase of G_{i-1}. Additionally, this move causes G_i to fall asleep. Thus, at the end of the wakeup phase of G_{i-1}, we have G_{i+1} wake up G_i.

Waking up

First, the point p_{i+1} joins $\mathcal{C}_1(G_i)$. Next, p_i moves from $\mathcal{C}_1(G_{i-1})$ to $\mathcal{C}_0(G_i)$. Then, q_i moves from $\mathcal{C}_1(G_i)$ to $\mathcal{C}_0(G_i)$, and finally, p_{i+1} leaves $\mathcal{C}_1(G_i)$, and joins either $\mathcal{C}_1(G_{i+1})$ (if G_{i+1} was in the morning state when waking up G_i) or $\mathcal{C}_0(G_{i+1})$ (if G_{i+1} was in the afternoon state; in this case, the move of p_{i+1} occurs during the wakeup phase of G_{i+1}).

Leaf gadget

The leaf gadget does not watch or wake up any other gadgets. It only wakes up when p_1 moves into $\mathcal{C}_0(G_0)$, and falls asleep again when p_1 moves back to a cluster of G_1.

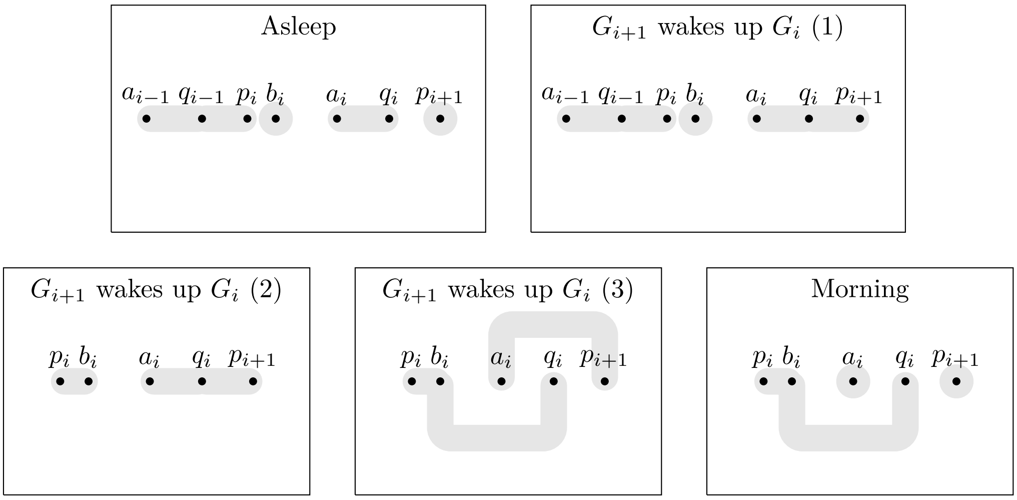
Asleep
a_{i-1} q_{i-1} p_i b_i a_i q_i p_{i+1}
G_{i+1} wakes up G_i (1)
a_{i-1} q_{i-1} p_i b_i a_i q_i p_{i+1}
G_{i+1} wakes up G_i (2)
p_i b_i a_i q_i p_{i+1}
G_{i+1} wakes up G_i (3)
p_i b_i a_i q_i p_{i+1}
Morning
p_i b_i a_i q_i p_{i+1}

Figure 2 Schematic depiction of the interactions between G_i, G_{i-1} and G_{i+1} during the wakeup phase of G_i. Note that the final state of G_i corresponds to the first state depicted in Figure 1.

Initialization

The sequence starts with all gadgets in the asleep, except for G_{m-1}, which is in its morning state.

At every step, we have the gadget with the smallest index that is not asleep wake up the gadget that it is watching. From this sequence of iterations, we can retrieve a series of inequalities, each of which encodes the condition that the gain of every iteration must be positive. To prove Theorem 1, we must show that the points in Table 1 satisfy these inequalities.

An implementation of the sequence described above is provided in the following link: `https://pastebin.com/raw/McdArCWg`.

4 Smoothed Analysis

For a smoothed analysis, the first hope might be to straightforwardly adapt a smoothed analysis of Lloyd's algorithm, e.g. that of Arthur, Manthey and Röglin [1]. On closer inspection, however, such analyses strongly rely on a couple of properties of Lloyd's method that are not valid in the Hartigan–Wong method.

First, in Lloyd's algorithm the hyperplane that bisects two cluster centers also separates their corresponding clusters, since every point is always assigned to the cluster center closest to itself. Second, the two stages of Lloyd's algorithm, moving the cluster centers and reassigning points, both decrease the potential. Neither of these properties are satisfied by iterations of the Hartigan–Wong method. Hence, any analysis that relies on either property cannot be easily repurposed.

Instead, we will use a different technique, more closely related to the analysis of the Flip heuristic for Max-Cut with squared Euclidean distances by Etscheid and Röglin [6]. The main result we will work towards in this section is stated in Theorem 2.

4.1 Technical Preliminaries

Let $\mathcal{Y} \subseteq [0,1]^d$ be a set of n points. Throughout the remainder, we will denote by $\mathcal{X}$ the set of points obtained by perturbing each point in $\mathcal{Y}$ independently by a d-dimensional Gaussian vector of mean 0 and standard deviation $\sigma \leq 1$. Note that this last assumption is not actually a restriction. If $\sigma > 1$, we scale down the set $\mathcal{Y}$ so that $\mathcal{Y} \subseteq [0, 1/\sigma]^d$, and subsequently perturb the points by Gaussian variables with $\sigma = 1$. Since the number of iterations required to terminate is invariant under scaling of the input point set, this is equivalent to the original instance.

Our analysis is based on the standard technique of proving that it is unlikely that a sequence of iterations decreases the potential function by a small amount. For this technique to work, we additionally require the potential function to be bounded from above and from below with sufficiently high probability. Since it is obvious that the potential is non-negative for any clustering, it is enough to guarantee that the perturbed point set $\mathcal{X}$ lies within the hypercube $[-D/2, D/2]^d$ for some finite D. To that end, we have the following lemma.

▶ **Lemma 6.** *Let $D = \sqrt{2n\ln(nkd)}$. Then $\mathbb{P}(\mathcal{X} \not\subseteq [-D/2, D/2]^d) \leq k^{-n}$.*

Similar results to Lemma 6 can be found in previous works on the smoothed analysis of algorithms on Gaussian-perturbed point sets [13, 1]. The only difference in our version is the value of D. Hence, we omit the proof.

Lemma 6 allows us to assume that all points lie within $[-D/2, D/2]^d$ after the perturbation. Formally, we must take into account the failure event that any point lies outside this hypercube. However, since the probability of this event is at most k^{-n}, this adds only a negligible $+1$ to the smoothed complexity bound which we prove in Theorem 2. We therefore ignore the failure event in the sequel.

We need to show that we can approximate the gain of an iteration if we have a good approximation to the cluster centers. Recall that $\Delta_x(C_i, C_j)$ is the gain of moving a point x from C_i to C_j. Since we wish to use approximations to the centers of C_i and C_j, it is convenient to define the variable

$$\Delta_x^{|C_i|,|C_j|}(a,b) = \frac{|C_i|}{|C_i|-1}\|x-a\|^2 - \frac{|C_j|}{|C_j|+1}\|x-b\|^2.$$

This variable is the gain that would be incurred if the centers of C_i and C_j, with fixed sizes $|C_i|$ and $|C_j|$, were a and b. Indeed, note that $\Delta_x^{|C_i|,|C_j|}(\mathrm{cm}(C_i), \mathrm{cm}(C_j)) = \Delta_x(C_i, C_j)$. When their intended values are clear from context, we will often omit the superscripts $|C_i|$ and $|C_j|$ from $\Delta_x^{|C_i|,|C_j|}(a,b)$.

4.2 Approximating Iterations

Before we begin with the analysis, we provide a rough outline. Suppose we tile the hypercube $[-D/2, D/2]^d$ with a rectangular grid of spacing ϵ. Then any point in $[-D/2, D/2]^d$ is at a distance of at most $\sqrt{d}\epsilon$ from some grid point. Since we need the positions of the cluster centers $c_i = \mathrm{cm}(C_i)$ for $i \in [k]$, we guess k grid points c_i' for their positions. If we guess correctly, meaning c_i' is the grid point closest to c_i for each $i \in [k]$, then we can approximate the gain Δ of an iteration by replacing the cluster centers with these grid points in the formula for Δ (Lemma 7).

The price for this approximation is a union bound over all choices of the grid points. However, we can compensate for this by noticing that, when we move a point between clusters, we know exactly how the cluster centers move. Thus, if the guessed grid points are good approximations, we can obtain new good approximations by moving them the same amount. Thus, we only need to guess once, and can use this guess for a sequence of iterations. Then we can bound the probability that all iterations in this sequence yield a small improvement.

▶ **Lemma 7.** *Suppose the point x moves from cluster i to cluster j. Let C_i and C_j denote the configurations of these clusters before this move, and let $c_i = \mathrm{cm}(C_i)$ and $c_j = \mathrm{cm}(C_j)$. Let c'_i and c'_j be two points such that $\|c_i - c'_i\|, \|c_j - c'_j\| \le \epsilon$ for some $0 \le \epsilon \le \sqrt{d}D$. Then*

$$|\Delta_x(C_i, C_j) - \Delta_x(c'_i, c'_j)| \le 9\sqrt{d}D\epsilon,$$

In particular, $\Delta_x(C_i, C_j) \in (0, \epsilon]$ implies $|\Delta_x(c'_i, c'_j)| \le 10\sqrt{d}D\epsilon$.

Proof. Observe that

$$\|x - c_i\|^2 = \|x - c'_i + c'_i - c_i\|^2 = \|x - c'_i\|^2 + \|c_i - c'_i\|^2 + 2\langle c'_i - c_i, x - c'_i\rangle.$$

Thus,

$$\Delta_x(C_i, C_j) = \Delta_x(c'_i, c'_j) + \frac{|C_i|}{|C_i| - 1}\left(\|c_i - c'_i\|^2 + 2\langle c'_i - c_i, x - c'_i\rangle\right) - \frac{|C_j|}{|C_j| + 1}\left(\|c_j - c'_j\|^2 + 2\langle c'_j - c_j, x - c'_i\rangle\right).$$

By the Cauchy-Schwarz inequality, $|\langle c'_i - c_i, x - c'_i\rangle| \le \epsilon \cdot \|x - c'_i\|$. Since all points are contained in $[-D/2, D/2]^d$, it holds that $c_i \in [-D/2, D/2]^d$. From this fact and the assumption that $\|c_i - c'_i\| \le \epsilon \le \sqrt{d}D$, it follows that $\|x - c'_i\| \le \sqrt{d}D$.

Moving $\Delta_x(c'_i, c'_j)$ to the left and taking an absolute value, we then obtain

$$|\Delta_x(C_i, C_j) - \Delta_x(c'_i, c'_j)| \le \left(\frac{|C_i|}{|C_i| - 1} + \frac{|C_j|}{|C_j| + 1}\right) \cdot 3\sqrt{d}D\epsilon.$$

To finish the proof, observe that by Lemma 5 the first term inside the parentheses is at most 2, while the second term is bounded by 1. We then have that $\Delta_x(C_i, C_j) \in (0, \epsilon]$ implies $\Delta_x(c'_i, c'_j) \in (-9\sqrt{d}D\epsilon, (9\sqrt{d}D + 1)\epsilon]$, which yields the lemma. ◀

In the following, we fix a set $A \subseteq \mathcal{X}$ of active points which will move during a sequence of the Hartigan–Wong method. We also fix the configuration of the active points, the sizes of the clusters $|C_1|$ and $|C_2|$ at the start of the sequence, and the order $\pi : A \to [|A|]$ in which the points move. Observe that these data also fix the sizes of the clusters whenever a new point moves.

While performing a sequence of iterations, the cluster centers move. Hence, even if we have a good approximation to a cluster center, it may not remain a good approximation after the iteration. However, if we know which points are gained and lost by each cluster, then we can compute new good approximations to the cluster centers from the old approximations. The following lemma captures this intuition.

▶ **Lemma 8.** *Let t_1, t_2 be two iterations of the Hartigan–Wong method in a sequence in which the points $A \subseteq \mathcal{X}$ move, with $t_1 < t_2$. Suppose in the iterations t_1 through $t_2 - 1$, cluster i loses the points S_- and gains the points S_+. Let $c_i(t)$ denote the cluster center of cluster i before t takes place, and let C_i^t denote its configuration before t. Let $c'_i(t_1) \in \mathbb{R}^d$, and $c'_i(t_2) = \frac{|C_i^{t_1}|}{|C_i^{t_2}|}c'_i(t_1) + \frac{1}{|C_i^{t_2}|}\left(\sum_{x \in S_+} x - \sum_{x \in S_-} x\right)$. Then*

$$\|c'_i(t_2) - c_i(t_2)\| = \frac{|C_i^{t_1}|}{|C_i^{t_2}|} \cdot \|c'_i(t_1) - c_i(t_1)\|.$$

Moreover, if $\|c'_i(0) - c_i(0)\| \le \epsilon$, then $\|c'_i(t_j) - c_i(t_j)\| \le 2|A|\epsilon$ for all $j \in [|A|]$.

Proof. Since the center of a cluster is defined as its center of mass, we can write

$$|C_i^{t_2}|\,\mathrm{cm}(C_i^{t_2}) = \sum_{x \in C_i^{t_1} \cup S_+ \setminus S_-} x = |C_i^{t_1}|\,\mathrm{cm}(C_i^{t_1}) + \sum_{x \in S_+} x - \sum_{x \in S_-} x.$$

Thus,

$$|C_i^{t_2}| c_i(t_2) = |C_i^{t_1}| c_i(t_1) + \sum_{x \in S_+} x - \sum_{x \in S_-} x.$$

Observe then that

$$\|c_i'(t_2) - c_i(t_2)\| = \frac{|C_i^{t_1}|}{|C_i^{t_2}|} \cdot \|c_i'(t_1) - c_i(t_1)\|.$$

This proves the first claim. To prove the second claim, we set $t_1 = 0$ and $t_2 = t_j$ for some $j \in [|A|]$ to obtain

$$\|c_i(t_j) - c_i'(t_j)\| = \frac{|C_i^0|}{|C_i^{t_j}|} \cdot \|c_i(0) - c_i'(0)\| \leq (|A|+1)\epsilon \leq 2|A|\epsilon,$$

since at most $|A|$ points are active during any subsequence. ◀

4.3 Analyzing Sequences

We now know that we can closely approximate the gain of a sequence of iterations, provided that we have good approximations to the cluster centers at the start of the sequence. The next step is then to show that there is only a small probability that such an approximate sequence improves the potential by a small amount. For that, we first require the following technical lemma.

▶ **Lemma 9.** *Let X be a d-dimensional Gaussian random variable with arbitrary mean μ and standard deviation $\sigma \leq 1$, and let $Z = a\|X\|^2 + \langle v, X\rangle$ for fixed $a \in \mathbb{R} \setminus \{0\}$ and $v \in \mathbb{R}^d$. Then the probability that Z falls in an interval of size $\epsilon \leq 1$ is bounded from above by $O\Big(\frac{1}{|a|\sqrt[4]{d}}\sqrt{\frac{\epsilon}{\sigma^2}}\Big)$.*

Proof. Let $Z_i = aX_i^2 + v_iX_i$, so that $Z = \sum_{i=1}^d Z_i$. We define the auxiliary variable $\bar{Z}_i = Z_i + v_i^2/(4a)$ and set $\bar{Z} = \sum_{i=1}^d \bar{Z}_i$. Since a and v are fixed, the densities of Z and $\bar{Z}$ are identical up to translation, and so we can analyze $\bar{Z}$ instead. Observe that $\bar{Z}_i/a = \big(X_i + \frac{v_i}{2a}\big)^2$. Thus, $\bar{Z}/a$ is equal in distribution to $\|Y\|^2$, where Y is a d-dimensional Gaussian variable with mean $\mu + v/(4a)$ and variance σ^2. We see then that $\bar{Z}/a$ has the density of a non-central chi-squared distribution.

For $\lambda \geq 0$, denote by $f(x, \lambda, d)$ the non-central d-dimensional chi-squared density with non-centrality parameter λ and standard deviation σ. Then [10]

$$f(x, \lambda, d) = \sum_{i=0}^{\infty} \frac{e^{-\lambda/2}(\lambda/2)^i}{i!} f(x, 0, d + 2i).$$

Now observe that $f(x, 0, d)$ is bounded from above by $O\Big(1/(\sqrt{d}\sigma^2)\Big)$ for $d \geq 2$. We can thus compute for an interval I of size ϵ

$$\mathbb{P}(\|Y\|^2 \in I) = \int_I f(x, \lambda, d) \leq c \cdot \frac{\epsilon}{\sqrt{d}\sigma^2},$$

for some $c > 0$. Moreover, since probabilities are bounded from above by 1, we can replace the right-hand side by

$$O\left(\frac{\sqrt{\epsilon}}{\sqrt[4]{d}\sigma}\right).$$

Adding in the scaling factor of $1/|a|$ then yields the lemma for $d \geq 2$,

For $d = 1$, we have

$$f(x, 0, 1) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \frac{e^{-\frac{x}{2\sigma^2}}}{\sqrt{x/\sigma^2}}.$$

Let I be an interval of size ϵ. Then

$$\mathbb{P}(\|Y\|^2 \in I) = \int_I f(x, \lambda, 1)\,\mathrm{d}x \leq \sum_{i=1}^{\infty} \frac{e^{-\lambda/2}(\lambda/2)^i}{i!} \int_I f(x, 0, 1 + 2i)\,\mathrm{d}x + \int_I f(x, 0, 1)\,\mathrm{d}x.$$

The first term is bounded by $O(\sqrt{\epsilon}/\sigma)$ by the same argument we used for $d \geq 2$. For the second term, we use the expression for $f(x, 0, 1)$ above to bound the integral as

$$\int_I f(x, 0, 1)\,\mathrm{d}x \leq \frac{1}{\sqrt{2\pi\sigma^2}} \int_0^{\epsilon} \frac{e^{-\frac{x}{2\sigma^2}}}{\sqrt{x/\sigma}}\,\mathrm{d}x = O(\sqrt{\epsilon}/\sigma).$$

This proves the lemma for $d = 1$ when we again add in the scaling factor $1/|a|$. ◀

With Lemma 9, we can show that a single fixed approximate iteration is unlikely to yield a small improvement.

▶ **Lemma 10.** *Let $a, b \in \mathbb{R}^d$ be fixed. Let $\Delta_x(a, b)$ be the improvement of the first move of x in S, if the cluster centers in this iteration are located at a and b. Let I be an interval of size $\epsilon \leq 1$. Then*

$$\mathbb{P}(\Delta_x(a, b) \in I) = O\left(\frac{n}{\sqrt[4]{d}} \cdot \sqrt{\frac{\epsilon}{\sigma^2}}\right).$$

Proof. By Lemma 4, we have

$$\begin{aligned}\Delta_x(a, b) &= \frac{|C_i|}{|C_i| - 1}\|x - a\|^2 - \frac{|C_j|}{|C_j| + 1}\|x - b\|^2 \\ &= \left(\frac{|C_i|}{|C_i| - 1} - \frac{|C_j|}{|C_j| + 1}\right)\|x\|^2 + \left\langle 2\left(\frac{|C_j|}{|C_j| + 1}b - \frac{|C_i|}{|C_i| - 1}a\right), x\right\rangle \\ &\qquad + \frac{|C_i|}{|C_i| - 1}\|a\|^2 - \frac{|C_j|}{|C_j| + 1}\|b\|^2,\end{aligned}$$

where $|C_i|$ and $|C_j|$ denote the sizes of clusters i and j before the iteration, and we assume x moves from cluster i to cluster j.

Since the sizes of the clusters as well as a and b are fixed, the last term in the above is fixed, and hence we may disregard it when analyzing $\mathbb{P}(\Delta_x(a, b) \in I)$. Since x is a Gaussian random variable, we can apply Lemma 9 to find

$$\mathbb{P}(\Delta_x(a, b) \in I) = O\left(\left(\frac{|C_i|}{|C_i| - 1} - \frac{|C_j|}{|C_j| + 1}\right)^{-1} \cdot \frac{1}{\sqrt[4]{d}} \cdot \sqrt{\frac{\epsilon}{\sigma^2}}\right).$$

It remains to bound quantity in the inner brackets from below. Since each cluster is bounded in size by n, we have

$$\frac{|C_i|}{|C_i|-1} - \frac{|C_j|}{|C_j|+1} \geq \frac{n}{n-1} - \frac{n}{n+1} = \frac{2n}{(n-1)(n+1)} \geq \frac{1}{n},$$

and we are done. ◀

As stated at the start of the analysis, analyzing a single iteration is not enough to prove Theorem 2. The following lemma extends Lemma 10 to a sequence of iterations, given a fixed point set $A \subseteq \mathcal{X}$ that moves in the sequence.

► **Lemma 11.** *Fix an active set A and starting cluster sizes $|C_i|$ for $i \in [k]$. Moreover, fix an order $\pi : A \to [|A|]$ in which the points in A move, i.e., $\pi(x) < \pi(y)$ means x moves for the first time before y moves for the first time. Let Δ denote the minimum improvement of a sequence satisfying these hypotheses over all possible configurations of $\mathcal{X} \setminus A$. Then for $\epsilon \leq 1$,*

$$\mathbb{P}(\Delta \leq \epsilon) \leq \left(\frac{2D}{\epsilon}\right)^{kd} \cdot \left(\frac{O(1) \cdot k^{|A|} \cdot d^{3/4} Dn|A|\sqrt{\epsilon}}{\sigma}\right)^{|A|}.$$

Proof. For $x \in A$, let Δ_x denote the improvement of the first move of $x \in A$. We label the points in A as $(x_1, \ldots, x_{|A|})$ according to π. Let $\Delta = (\Delta_i)_{i=1}^{|A|}$.

To compute the vector Δ, we would need to know the configuration and positions of the points $P = \mathcal{X} \setminus A$, since these are required to compute the k cluster centers. However, if we had approximations to the cluster centers in every iteration corresponding to the entries of Δ, then we could compute an approximation to Δ by Lemma 7.

Since the cluster centers are convex combinations of points in $[-D/2, D/2]^d$, we know that the cluster centers at the start of S must also lie in $[-D/2, D/2]^d$. Thus, there exist grid points c'_i ($i \in [k]$) within a distance $\sqrt{d}\epsilon$ of the initial cluster centers.

Knowing these grid points, we would like to apply Lemma 8 in order to update the approximate cluster centers whenever a new point moves. We then need to know the points gained and lost by each cluster between first moves of each $x \in A$. Observe that to obtain this information, it suffices to know the configuration of the active points before the first move of each $x \in A$. Thus, we fix these configurations.

We collect the gain of each first move of a point in A, where we replace the cluster centers by these approximations, into a vector Δ'. By the reasoning above and by Lemmas 7 and 8, if there exist initial cluster centers c_i ($i \in [k]$) such that $\Delta_x \in (0, \epsilon]$ for all $x \in A$, then there exist grid points c'_i, such that $|\Delta'_x| \leq 20|A|dD\epsilon$ for all $x \in A$. (Compared to Lemma 7, we gain an extra factor of $2|A|$ due to Lemma 8.)

By this reasoning, it suffices to obtain a bound on $\mathbb{P}\left(\bigcap_{x\in A} |\Delta'_x| \leq 20|A|dD\epsilon\right)$. We can then take a union bound over these events for all $(D/\epsilon + 1)^{kd} \leq (2D/\epsilon)^{kd}$ choices of c'_i for $i \in [k]$, and a union bound over the configuration of A before the first move of each $x \in A$.

To show that $\mathbb{P}\left(\bigcap_{x\in A} |\Delta'_x| \leq 20|A|dD\epsilon\right)$ is bounded as desired, we consider the following algorithm.

1. Set $t = 1$.
2. Reveal x_t, and compute $\Delta_{x_t}(c'_{i_t}, c'_{j_t})$, where x_t moves from C_{i_t} to C_{j_t}.
3. If $|\Delta_{x_t}(c'_{i_t}, c'_{j_t})| > 20|A|dD\epsilon$, then return GOOD and halt.
4. If $t = |A|$, return BAD.
5. Update the positions of the approximate cluster centers using Lemma 8.
6. Continue executing moves in the sequence until we encounter the first move of x_{t+1}. Observe that the information we fixed before executing this algorithm suffices to compute approximations to the cluster centers whenever a new point moves.
7. Set $t \leftarrow t + 1$ and go to step 2.

The sequence of iterations improves the potential by at most ϵ only if the above algorithm returns BAD. We now argue that

$$\mathbb{P}(\mathsf{BAD}) \leq \left(O(1) \cdot d^{3/4} Dn|A|\sqrt{\epsilon}/\sigma\right)^{|A|}.$$

Let BAD_t be the event that the above algorithm loops for at least t iterations. Then $\mathbb{P}(\mathsf{BAD}) = \mathbb{P}(\mathsf{BAD}_{|A|})$. Since $\mathbb{P}(\mathsf{BAD}_t \mid \neg\mathsf{BAD}_{t-1}) = 0$, we can immediately conclude that for all $t \in \{2, \dots, |A|\}$,

$$\mathbb{P}(\mathsf{BAD}_t) = \mathbb{P}(\mathsf{BAD}_t \mid \mathsf{BAD}_{t-1})\mathbb{P}(\mathsf{BAD}_{t-1}).$$

By Lemma 10, we have $\mathbb{P}(\mathsf{BAD}_t \mid \mathsf{BAD}_{t-1}) \leq O(1) \cdot d^{3/4} Dn|A|\sqrt{\epsilon}/\sigma$. Thus, $\mathbb{P}(\mathsf{BAD}_t)$ is bounded as claimed.

Taking a union bound over all choices of the approximate grid points at the start of the sequence yields the factor $(2D/\epsilon)^{kd}$. Finally, we must take a union bound over the configuration of A before the first move of each $x \in A$, yielding a factor $k^{|A|^2}$, which concludes the proof. ◀

Armed with Lemma 11, we can bound the probability that there exists a sequence in which a fixed number of points moves, which improves the potential by at most ϵ.

▶ **Lemma 12.** *Let $\Delta_{\min}$ denote the minimum improvement of any sequence of moves in which exactly $4kd$ distinct points switch clusters. Then for $\epsilon \leq 1$,*

$$\mathbb{P}(\Delta_{\min} \leq \epsilon) \leq \left(\frac{O(1) \cdot k^{8kd+4} d^{11} D^5 n^{8+\frac{1}{d}} \epsilon}{\sigma^4}\right)^{kd}.$$

Proof. Fix an active set A of $4kd$ distinct points, an order $\pi : A \to [|A|]$ in which the points in A move, and the sizes of the two clusters at the start of the sequence.

We have by Lemma 11

$$\mathbb{P}(\Delta(S) \leq \epsilon) \leq \left(\frac{2D}{\epsilon}\right)^{kd} \left(\frac{O(1) \cdot k^{2kd} \cdot d^{7/4} Dn\sqrt{\epsilon}}{\sigma}\right)^{4kd} = \left(\frac{O(1) \cdot d^7 \cdot k^{8kd} \cdot D^5 n^4 \epsilon}{\sigma^4}\right)^{kd}.$$

We conclude the proof by a union bound over the choices of A, π, and the sizes of the clusters at the start of the sequence, which yields a factor of at most $(4kd)^{4kd} \cdot n^{4kd+1}$. ◀

With Lemma 12, we are in a position to prove the main result of this section. The proof is essentially mechanical, following techniques used in many previous smoothed analyses [1, 3, 4, 5, 6, 13].

▶ **Theorem 2** (Restated). *Let $n, k, d \in \mathbb{N}$, and assume $4kd \leq n$. Fix a set of n points $\mathcal{Y} \subseteq [0,1]^d$, and assume that each point in $\mathcal{Y}$ is independently perturbed by a d-dimensional Gaussian random variable with mean 0 and standard deviation σ, yielding a new set of points $\mathcal{X}$. Then the expected running time of the Hartigan–Wong method on $\mathcal{X}$ is bounded by*

$$O\left(\frac{k^{12kd+5} d^{12} n^{12.5+\frac{1}{d}} \ln^{4.5}(nkd)}{\sigma^4}\right) = k^{12kd} \cdot \mathrm{poly}(n, k, d, 1/\sigma).$$

Proof. First, we recall that the point set $\mathcal{X}$ is contained in $[-D/2, D/2]^d$. This yields an upper bound for the value of the potential function for the initial clustering C,

$$\Phi(C) = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - \mathrm{cm}(C_i)\|^2 \leq kndD^2.$$

We divide the sequence of iterations executed by the Hartigan–Wong method into contiguous disjoint blocks during which exactly $4kd$ distinct points move. By Lemma 12, we know that the probability that any such block yields a bad improvement is small.

Let T be the number of such blocks traversed by the heuristic before we reach a local optimum. Then

$$\mathbb{P}(T \geq t) \leq \mathbb{P}\left(\Delta_{\min} \leq \frac{kndD^2}{t}\right) \leq \min\left\{1, \frac{O(1) \cdot k^{8kd+5} d^{12} D^7 n^{9+\frac{1}{d}}}{\sigma^4} \cdot \frac{1}{t}\right\}.$$

This probability becomes nontrivial when

$$t > \left\lceil \frac{O(1) \cdot k^{8kd+5} d^{12} D^7 n^{9+\frac{1}{d}}}{\sigma^4} \right\rceil =: t'.$$

Observe that $t' = \Omega(kndD^2)$, justifying our use of Lemma 12 above. Thus, we find

$$\mathbb{E}(T) = \sum_{t=1}^{k^n} \mathbb{P}(T \geq t) \leq t' + t' \cdot \sum_{t=t'}^{k^n} \frac{1}{t} \leq t' + t' \cdot \int_{t'}^{k^n} \frac{1}{t}\,\mathrm{d}t \leq t' + t' \cdot \ln(k^n).$$

The upper limit of k^n to the sum is simply the number of possible clusterings of n points into k sets, which is a trivial upper bound to the number of iterations. To conclude, we observe that any block in which exactly $4kd$ distinct points move has a length of at most k^{4kd}, as otherwise some clustering would show up twice. Thus, we multiply $\mathbb{E}(T)$ by k^{4kd} to obtain a bound for the smoothed complexity. Finally, we insert the value of $D = \sqrt{2n\ln(nkd)}$. ◀

5 Discussion

Theorems 1 and 2 provide some of the first rigorous theoretical results concerning the Hartigan–Wong method method that have been found since Telgarsky & Vattani explored the heuristic in 2010 [14]. Of course, many interesting open questions still remain.

Worst-case construction. Theorem 1 establishes the existence of exponential-length sequences on the line, but leaves open the possibility that a local optimum may be reachable more efficiently by a different improving sequence. To be precise: given an instance of k-means clustering on the line and an initial clustering, does there always exist a sequence of iterations of the Hartigan–Wong method of length $\operatorname{poly}(n, k)$ starting from this clustering and ending in a local optimum? Although the $d = 1$ case appears very restricted at first sight, this question seems surprisingly difficult to answer.

In addition, the construction we use in Theorem 1 requires $k = \Theta(n)$ clusters. This opens up the question whether similar worst-case constructions can be made using fewer, perhaps even $O(1)$, clusters. Note that this is not true for Lloyd's method, since the number of iterations of Lloyd's method is bounded by $n^{O(kd)}$ [9], which is polynomial for $k, d \in O(1)$.

Smoothed complexity. Theorem 2 entails, to our knowledge, the first step towards settling the conjecture by Telgarsky & Vattani [14] that the Hartigan–Wong method has polynomial smoothed complexity. Our result is reminiscent of the smoothed complexity bound of Lloyd's method obtained in 2009 by Manthey & Röglin [12], which is $k^{kd} \cdot \operatorname{poly}(n, 1/\sigma)$. In the case of Lloyd's method, the smoothed complexity was later settled to $\operatorname{poly}(n, k, d, 1/\sigma)$ [1].

Observe that our bound is polynomial for constant k and d, and even for $kd \log k \in O(\log n)$. While this is certainly an improvement over the trivial upper bound of k^n, it falls short of a true polynomial bound. We hope that our result can function as a first step to a $\text{poly}(n, k, d, 1/\sigma)$ smoothed complexity bound of the Hartigan–Wong method.

We remark that the exponents in the bound in Theorem 2 can be easily improved by a constant factor for $d \geq 2$. The reason is that in Lemma 9, the factor $\sqrt{\epsilon}$ emerges from the $d = 1$ case, while for $d \geq 2$ we could instead obtain ϵ. We chose to combine these cases for the sake of keeping the analysis simple, as we expect the bound in Theorem 2 would be far from optimal regardless.

Improving the smoothed bound. We do not believe that the factor of $k^{O(kd)}$ is inherent in the smoothed complexity of the Hartigan–Wong method, but is rather an artifact of our analysis. To replace this factor by a polynomial in k and d, it seems that significantly new ideas might be needed.

The factors arise from two sources in our analysis. First, we take a union bound over the configuration of the active points each time we apply Lemma 8, yielding factors of $k^{O(kd)}$. Second, we analyze sequences in which $\Theta(kd)$ points move in order to guarantee a significant potential decrease. This incurs a factor of the length of such a sequence, which is another source of a factor $k^{O(kd)}$. We do not see how to avoid such factors when taking our approach.

One avenue for resolving this problem might be to analyze shorter sequences in which a significant number of points move. Angel et al. used such an approach in their analysis of the Flip heuristic for Max-Cut. They identify in any sequence L of moves a shorter subsequence B, such that the number of unique vertices that flip in B is linear in the length of B. The major challenge is then to find sufficient independence in such a short subsequence, which in our case seems challenging, as we need to compensate for a factor ϵ^{-kd} in Lemma 11.

Since our analysis greatly resembles the earlier analysis of the Flip heuristic for Squared Euclidean Max Cut [6], it might be helpful to first improve the latter. This analysis yields a bound of $2^{O(d)} \cdot \text{poly}(n, 1/\sigma)$. If this can be improved to $\text{poly}(n, d, 1/\sigma)$, then it is likely that a similar method can improve on our analysis for the Hartigan–Wong method as well.

References

1 David Arthur, Bodo Manthey, and Heiko Röglin. Smoothed Analysis of the k-Means Method. *Journal of the ACM*, 58(5):19:1–19:31, October 2011. doi:10.1145/2027216.2027217.

2 David Arthur and Sergei Vassilvitskii. K-means++: The advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, pages 1027–1035, USA, January 2007. Society for Industrial and Applied Mathematics.

3 Matthias Englert, Heiko Röglin, and Berthold Vöcking. Worst Case and Probabilistic Analysis of the 2-Opt Algorithm for the TSP. *Algorithmica*, 68(1):190–264, January 2014. doi:10.1007/s00453-013-9801-4.

4 Matthias Englert, Heiko Röglin, and Berthold Vöcking. Smoothed Analysis of the 2-Opt Algorithm for the General TSP. *ACM Transactions on Algorithms*, 13(1):10:1–10:15, September 2016. doi:10.1145/2972953.

5 Michael Etscheid and Heiko Röglin. Smoothed Analysis of the Squared Euclidean Maximum-Cut Problem. In Nikhil Bansal and Irene Finocchi, editors, *Algorithms - ESA 2015*, Lecture Notes in Computer Science, pages 509–520, Berlin, Heidelberg, 2015. Springer. doi:10.1007/978-3-662-48350-3_43.

6 Michael Etscheid and Heiko Röglin. Smoothed Analysis of Local Search for the Maximum-Cut Problem. *ACM Transactions on Algorithms*, 13(2):25:1–25:12, March 2017. doi:10.1145/3011870.

7 Gurobi Optimization LLC. Gurobi Optimizer Reference Manual. Gurobi Optimization, LLC, 2023.

8 J. A. Hartigan and M. A. Wong. Algorithm AS 136: A K-Means Clustering Algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979. doi:10.2307/2346830.

9 M Inaba, Naoki Katoh, and Hiroshi Imai. Variance-based k-clustering algorithms by Voronoi diagrams and randomization. *IEICE Transactions on Information and Systems*, E83D, June 2000.

10 Norman L. Johnson, Samuel Kotz, and Narayanaswamy Balakrishnan. *Continuous Univariate Distributions, Volume 2*. John Wiley & Sons, May 1995.

11 S. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, March 1982. doi:10.1109/TIT.1982.1056489.

12 Bodo Manthey and Heiko Röglin. Improved Smoothed Analysis of the k-Means Method. In *Proceedings of the 2009 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, Proceedings, pages 461–470. Society for Industrial and Applied Mathematics, January 2009. doi:10.1137/1.9781611973068.51.

13 Bodo Manthey and Rianne Veenstra. Smoothed Analysis of the 2-Opt Heuristic for the TSP: Polynomial Bounds for Gaussian Noise. In Leizhen Cai, Siu-Wing Cheng, and Tak-Wah Lam, editors, *Algorithms and Computation*, Lecture Notes in Computer Science, pages 579–589, Berlin, Heidelberg, 2013. Springer. doi:10.1007/978-3-642-45030-3_54.

14 Matus Telgarsky and Andrea Vattani. Hartigan's Method: K-means Clustering without Voronoi. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 820–827. JMLR Workshop and Conference Proceedings, March 2010.

15 Andrea Vattani. K-means Requires Exponentially Many Iterations Even in the Plane. *Discrete & Computational Geometry*, 45(4):596–616, June 2011. doi:10.1007/s00454-011-9340-1.