Discrete geometry

Lecture 2
“The world is continuous, but the mind is discrete”

David Mumford
Discretization

Continuous world

- Surface $X$
- Metric $d_X$
- Topology

Discrete world

- Sampling
  \[ X' = \{x_1, ..., x_N\} \subset X \]
- Discrete metric (matrix of distances)
  \[ D_X = (d_X(x_i, x_j)) \]
- Discrete topology (connectivity)
How good is a sampling?
Sampling density

- How to quantify **density** of sampling?
- \( X' \) is an \( r \)-covering of \( X \) if

\[
\bigcup_{x_i \in X'} B_r(x_i) = X
\]

Alternatively:

\[
d_X(x, X') \leq r
\]

for all \( x \in X \), where

\[
d_X(x, X') = \inf_{x_i \in X'} d_X(x, x_i)
\]

is the **point-to-set distance**.
Sampling efficiency

- Are all points necessary?
- An $r$-covering may be unnecessarily dense (may even not be a discrete set).
- Quantify how well the samples are separated.
- $X'$ is $r'$-separated if
  \[ d_X(x_i, x_j) \geq r' \]
  for all $x_i, x_j \in X$.
- For $r' > 0$, an $r'$-separated set is finite if $X$ is compact.

Also an $r$-covering!
Farthest point sampling

- Good sampling has to be dense and efficient at the same time.
- Find and $\tau$-separated $\tau$-covering $X'$ of $X$.
- Achieved using farthest point sampling.

- We defer the discussion on
  - How to select $\tau$?
  - How to compute $d_X$?
Farthest point sampling
Farthest point sampling

- Start with some $X' = \{x_1 \in X\}$.
- Determine sampling radius
  $$r = \max_{x \in X} d_X(x, X')$$
- If $r \leq r_{\text{target}}$ stop.
- Find the farthest point from $X'$
  $$x' = \arg \max_{x \in X} d_X(x, X')$$
- Add $x'$ to $X'$
Farthest point sampling

- Outcome: $r$-separated $r$-covering of $X$.
- Produces sampling with **progressively increasing** density.
- A **greedy algorithm**: previously added points remain in $X'$.
- There might be another $r$-separated $r$-covering containing less points.
- In practice used to **sub-sample** a densely sampled shape.
- Straightforward time complexity: $O(MN)$
  
  $M$ number of points in dense sampling, $N$ number of points in $X'$.
- Using **efficient data structures** can be reduced to $O(N \log M)$.
Numerical geometry of non-rigid shapes  Discrete geometry

Sampling as representation

- Sampling **represents** a region on $X$ as a single point $x_i \in X'$.  
- Region of points on $X$ closer to $x_i$ than to any other $x_j$:

$$V_i(X') = \{ x \in X : d_X(x, x_i) < d_X(x, x_j), x_j \neq i \in X' \}$$

- **Voronoi region** (a.k.a. Dirichlet or Voronoi-Dirichlet region, Thiessen polytope or polygon, Wigner-Seitz zone, domain of action).

- To avoid degenerate cases, assume points in $X'$ in **general position**:
  - No three points lie on the **same geodesic**.
    (Euclidean case: no three **collinear** points).
  - No four points lie on the **boundary of the same metric ball**.
    (Euclidean case: no four **cocircular** points).
Numerical geometry of non-rigid shapes  Discrete geometry

Voronoi decomposition

A point $x \in X$ can belong to one of the following

- **Voronoi region** $V_i$ ( $x$ is closer to $x_i$ than to any other $x_j$).
- **Voronoi edge** $V_{ij} = \overline{V_i} \cap \overline{V_j}$ ( $x$ is equidistant from $x_i$ and $x_j$).
- **Voronoi vertex** $V_{ijk} = \overline{V_i} \cap \overline{V_j} \cap \overline{V_k}$ ( $x$ is equidistant from three points $x_i, x_j, x_k$).
Voronoi decomposition
Voronoi decomposition

- Voronoi regions are **disjoint**.
- Their closure

\[ \bigcup_i \overline{V}_i = X \]

covers the entire \( X \).
- Cutting \( X \) along Voronoi edges produces a collection of **tiles** \( \{V_i\} \).
- In the **Euclidean** case, the tiles are **convex polygons**.
- Hence, the tiles are **topological disks** (are homeomorphic to a disk).
Voronoi tessellation

- **Tessellation** of $X$ (a.k.a. cell complex): a finite collection of disjoint open topological disks, whose closure cover the entire $X$.

- In the **Euclidean** case, Voronoi decomposition is **always** a tessellation.

- In the **general** case, Voronoi regions might not be topological disks.

- A valid tessellation is obtained if the sampling $X'$ is **sufficiently dense**.
Non-Euclidean Voronoi tessellations

- **Convexity radius** at a point \( x \in X \) is the largest \( \rho \) for which the closed ball \( \overline{B}_\rho(x) \) is convex in \( X \), i.e., minimal geodesics between every \( x', x'' \in \overline{B}_\rho(x) \) lie in \( \overline{B}_\rho(x) \).

- **Convexity radius** of \( X = \inf \) of convexity radii over all \( x \in X \).

- **Theorem** (Leibon & Letscher, 2000):
  
  An \( r \)-separated \( r \)-covering \( X' \) of \( X \) with \( r < \frac{1}{3} \) convexity radius of \( X \) is guaranteed to produce a valid Voronoi tessellation.

- Gives **sufficient sampling density** conditions.
Sufficient sampling density conditions

Invalid tessellation

Valid tessellation
Voronoi tessellations in Nature
MATLAB® intermezzo

Farthest point sampling and Voronoi decomposition
Representation error

- Voronoi decomposition replaces \( x \in X \) with the closest point \( x^* \in X' \).
- Mapping \( x^* : X \to X' \) copying each \( V_i(X') \) into \( x_i \).
- Quantify the representation error introduced by \( x^* \).
- Let \( x \in X \) be picked randomly with uniform distribution on \( X \).

\[
P(x \in A) = \frac{\mu(A)}{\mu(X)} = \frac{1}{\mu(X)} \int_A da
\]

- Representation error = variance of \( d_X(x, x^*(x)) \)

\[
\varepsilon(X') = \text{Var}(d_X(x, x^*(x))) = \frac{1}{\mu(X)} \int_{x \in X} d_X^2(x, x^*(x)) da
\]

\[
= \frac{1}{\mu(X)} \sum_{i=1}^{N} \int_{x \in V_i(X')} d_X^2(x, x_i) da
\]
Optimal sampling

In the Euclidean case:

$$\varepsilon(X') = \frac{1}{\mu(X)} \sum_{i=1}^{N} \int_{V_i(X')} \|x - x_i\|_2^2 dx$$

(mean squared error).

Optimal sampling: given a fixed sampling size $N$, minimize error

$$X' = \arg \min_{X'} \varepsilon(X') \quad \text{s.t.} \quad |X'| = N$$

Alternatively: Given a fixed representation error $\varepsilon_0$, minimize sampling size

$$X' = \arg \min_{X'} |X'| \quad \text{s.t.} \quad \varepsilon(X') \leq \varepsilon_0$$
Centroidal Voronoi tessellation

In a sampling $X'$ minimizing $\varepsilon(X')$, each $x_i$ has to satisfy

$$x_i = \arg\min_{x \in V_i} \int_{x' \in V_i} d^2_X(x, x') \, dx$$

(intrinsic centroid)

In the Euclidean case – center of mass

$$x_i = \arg\min_{x \in V_i} \int_{V_i} \|x - x'\|_2^2 \, dx' = \frac{\int_{V_i} x \, dx}{\int_{V_i} dx}$$

In general case: intrinsic centroid of $V_i$.

Centroidal Voronoi tessellation (CVT): Voronoi tessellation generated by $X'$ in which each $x_i$ is the intrinsic centroid of $V_i(X')$. 
Lloyd-Max algorithm

- Start with some sampling $X'$ (e.g., produced by FPS)
- Construct Voronoi decomposition $\{V_i(X')\}$
- For each $i$, compute intrinsic centroids

$$ \overline{x}_i = \arg\min_{x \in V_i} \int_{x' \in V_i} d_X^2(x, x') \, da $$

- Update $X' = \{\overline{x}_1, ..., \overline{x}_N\}$

- In the limit $N \to \infty$, $\{V_i(X')\}$ approaches the hexagonal honeycomb shape – the densest possible tessellation.

- Lloyd-Max algorithms is known under many other names: vector quantization, k-means, etc.
Sampling as clustering

Partition the space $X$ into clusters $V_1, \ldots, V_N$ with centers $x_1, \ldots, x_N$ to minimize some cost function.

- **Maximum cluster radius**
  \[
  \varepsilon_\infty(x_1, \ldots, x_N) = \max_i \max_{x \in V_i} d_X(x, x_i)
  \]

- **Average cluster radius**
  \[
  \varepsilon_2(x_1, \ldots, x_N) = \frac{1}{N} \sum_{i=1}^{N} \sum_{x \in V_i} d_X(x, x_i)
  \]

In the discrete setting, both problems are **NP-hard**.

Lloyd-Max algorithm, a.k.a. **k-means** is a heuristic, *sometimes* minimizing average cluster radius $\varepsilon_2$ (if converges globally – not guaranteed).
Farthest point sampling *encore*

- Start with some \( x_1 \in X \), \( R_1 = \infty \)
- For \( i = 2, \ldots, N \)
  - Find the **farthest point**
    \[
    x_i = \arg \max_{x \in X} d_X(x, \{x_1, \ldots, x_{i-1}\})
    \]
  - Compute the **sampling radius**
    \[
    R_i = d_X(x_i, \{x_1, \ldots, x_{i-1}\})
    \]

**Lemma**

- \( R_1 \geq R_2 \geq \ldots \geq R_N \)
- \( \varepsilon_\infty(x_1, \ldots, x_N) = R_{N+1} \)
Proof

- \( R_1 \geq R_2 \geq \ldots \geq R_N \)

For any \( j > i \)

\[
R_j = d_X(x_j, \{x_1, \ldots, x_{j-1}\}) \\
= d_X(x_j, \{x_1, \ldots, x_{i-1}, \ldots, x_{j-1}\}) \\
\leq d_X(x_j, \{x_1, \ldots, x_{i-1}\}) \\
\leq d_X(x_i, \{x_1, \ldots, x_{i-1}\}) = R_i
\]

\( x_i = \arg \max_{x \in X} d_X(x, \{x_1, \ldots, x_{i-1}\}) \)
Proof (cont)

\[ \varepsilon_\infty(x_1, \ldots, x_N) = R_{N+1} \]

\[
\max_{x \in X} d_X(x, \{x_1, \ldots, x_N\}) = \max_{x \in \bigcup_i V_i} d_X(x, \{x_1, \ldots, x_N\})
\]

\[
= \max_{i=1,\ldots,N} \max_{x \in V_i} d_X(x, \{x_1, \ldots, x_N\})
\]

\[
= \max_{i=1,\ldots,N} \max_{x \in V_i} d_X(x, x_i)
\]

\[
= \varepsilon_\infty(\{x_1, \ldots, x_N\})
\]

Since \( x_{N+1} = \arg \max_{x \in X} d_X(x, \{x_1, \ldots, x_N\}) \), we have

\[
\varepsilon_\infty(\{x_1, \ldots, x_N\}) = \max_{x \in X} d_X(x, \{x_1, \ldots, x_N\}) = d_X(x_{N+1}, \{x_1, \ldots, x_N\}) = R_{N+1}
\]
Almost optimal sampling

**Theorem** (Hochbaum & Shmoys, 1985)

Let \( x_1, \ldots, x_N \) be the result of the FPS algorithm. Then

\[
\varepsilon_\infty(\{x_1, \ldots, x_N\}) \leq 2 \min \varepsilon_\infty
\]

*In other words*: FPS is worse than optimal sampling by at most 2.
Idea of the proof

Let $V_1^*, \ldots, V_N^*$ denote the optimal clusters, with centers $x_1^*, \ldots, x_N^*$.

Distinguish between two cases:

- One of the clusters contains two or more of the points $x_1, \ldots, x_N$.
- Each cluster contains exactly one of the points $x_1, \ldots, x_N$. 

![Diagram with points and clusters](image-url)
Proof (first case)

Assume one of the clusters $V_i^*$ contains two or more of the points $x_1, \ldots, x_N$, e.g. $x_j, x_k$

$$2\varepsilon_\infty(\{x_1^*, \ldots, x_N^*\}) =$$

$$2 \max_{i=1,\ldots,N} \max_{x \in V_i^*} d_X(x, x_i^*) \geq$$

$$2 \max_{x \in V_i^*} d_X(x, x_i^*) \geq$$

$$d_X(x_j, x_i^*) + d_X(x_k, x_i^*) \geq d_X(x_j, x_k) \geq R_N$$

triangle inequality

Hence: $$2\varepsilon_\infty(\{x_1^*, \ldots, x_N^*\}) \geq R_N \geq R_{N+1} = \varepsilon_\infty(\{x_1, \ldots, x_N\})$$
Proof (second case)

Assume each of the clusters $V_i^*$ contains exactly one of the points $x_1, \ldots, x_N$, e.g. $x_j$.

Then, for any point $x \in V_i^*$

$$d_X(x, \{x_1, \ldots, x_N\}) \leq d_X(x, x_j) \leq d_X(x, x_i^*) + d_X(x_j, x_i^*)$$

triangle inequality

$$\leq 2 \max_{x \in V_i^*} d_X(x, x_i^*) \leq 2 \max_{i=1,\ldots,N} \max_{x \in V_i^*} d_X(x, x_i^*)$$

$$= 2\epsilon_\infty(\{x_1^*, \ldots, x_N^*\})$$
Proof (second case, cont)

We have: for any $i$, for any point $x \in V_i^*$

$$d_X(x, \{x_1, ..., x_N\}) \leq 2\varepsilon_\infty(\{x_1^*, ..., x_N^*\})$$

In particular, for $x_{N+1}$

$$\varepsilon_\infty(\{x_1, ..., x_N\}) =$$

$$R_{N+1} = d_X(x_{N+1}, \{x_1, ..., x_N\}) \leq 2\varepsilon_\infty(\{x_1^*, ..., x_N^*\})$$