Smoothed Analysis of ICP

David Arthur
Sergei Vassilvitskii

(Stanford University)

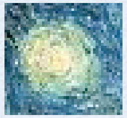
Matching Datasets

Problem: Given two point sets A and B, translate A to best match B.

ICP: Iterative Closest Point

Problem: Given two point sets A and B, translate A to best match B.

Example:

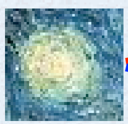
$\mathcal{A} =$ 

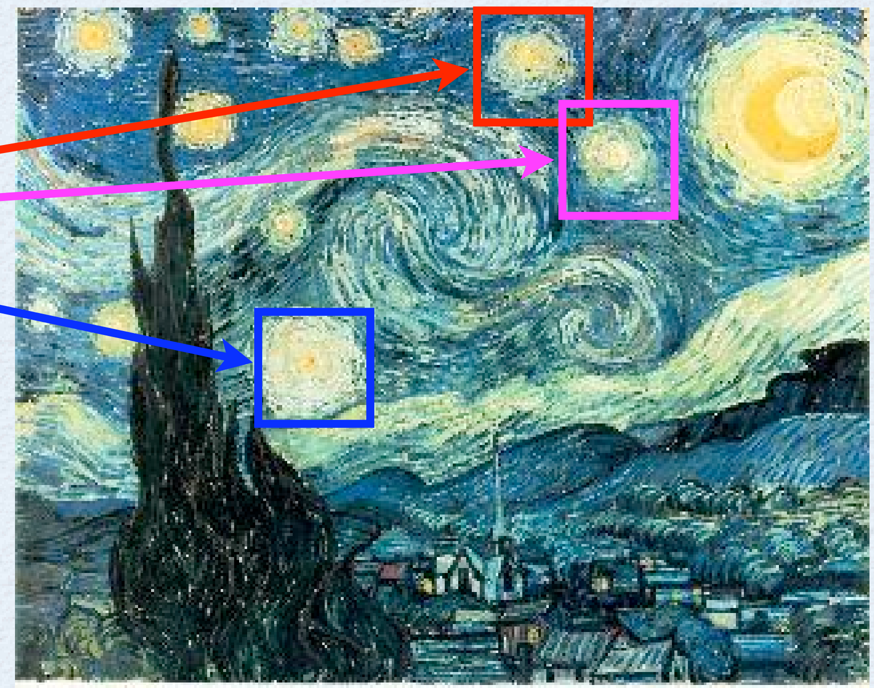
$\mathcal{B} =$ 

ICP: Iterative Closest Point

Problem: Given two point sets A and B, translate A to best match B.

Example:

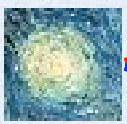
$\mathcal{A} =$ 

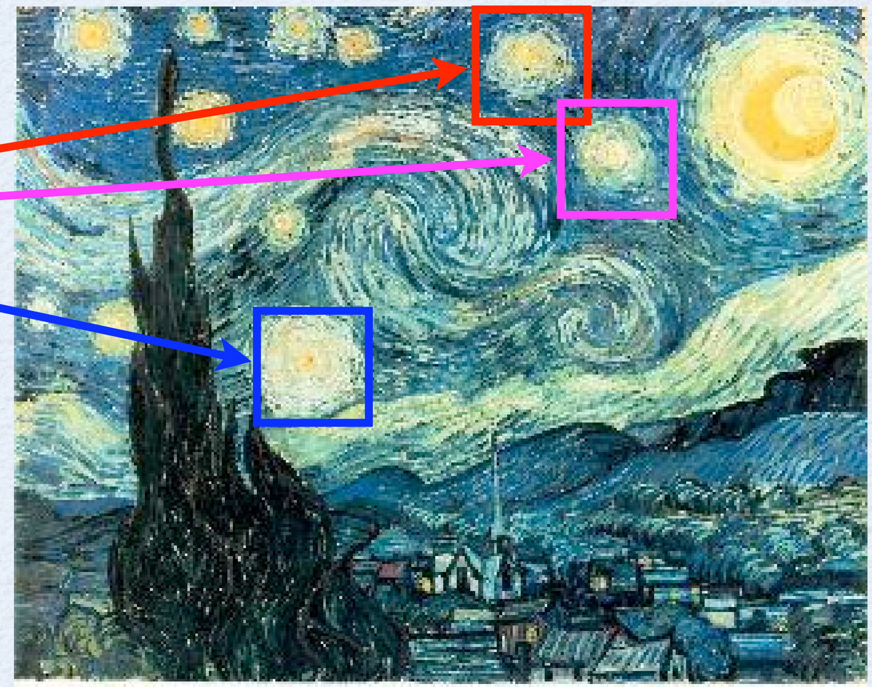
$\mathcal{B} =$ 

ICP: Iterative Closest Point

Problem: Given two point sets A and B, translate A to best match B.

Example:

$\mathcal{A} =$ 

$\mathcal{B} =$ 

Which is best?

$$\min_{x} \phi(x) = \sum_{a \in A} \| a + x - N_{\mathcal{B}}(a + x) \|_2^2$$

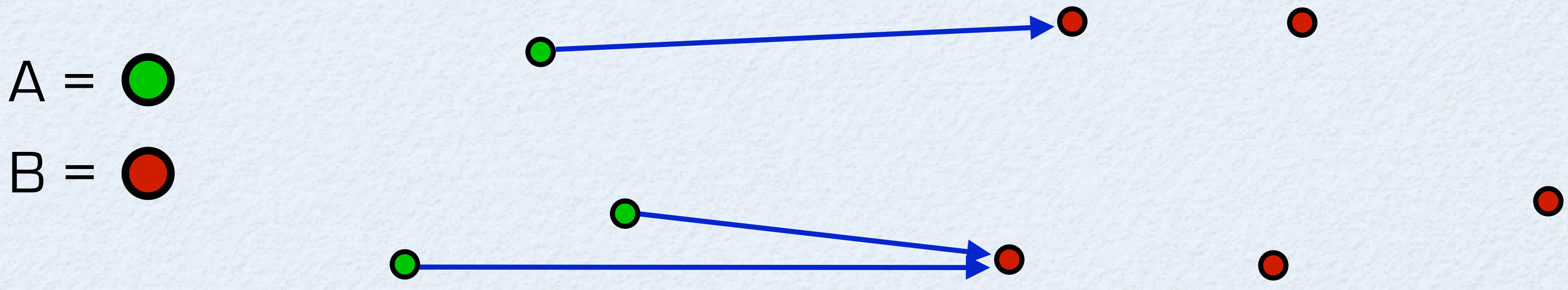
ICP: Iterative Closest Point

Given $\mathcal{A}, \mathcal{B}$, $|\mathcal{A}| = |\mathcal{B}| = n$

1. Begin with some translation x_0

2. Compute $N_{\mathcal{B}}(a + x_i)$ for each $a \in \mathcal{A}$

3. Fix $N_{\mathcal{B}}(\cdot)$, compute optimal $x_{i+1} = \sum_{a \in A} \frac{N_{\mathcal{B}}(a + x_i) - a}{|\mathcal{A}|}$



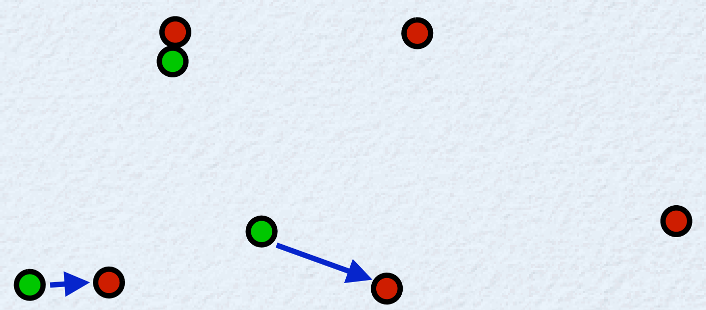
ICP: Iterative Closest Point

Given $\mathcal{A}, \mathcal{B}$, $|\mathcal{A}| = |\mathcal{B}| = n$

1. Begin with some translation x_0
2. Compute $N_{\mathcal{B}}(a + x_i)$ for each $a \in \mathcal{A}$
3. Fix $N_{\mathcal{B}}(\cdot)$, compute optimal $x_{i+1} = \sum_{a \in A} \frac{N_{\mathcal{B}}(a + x_i) - a}{|\mathcal{A}|}$

A =

B =



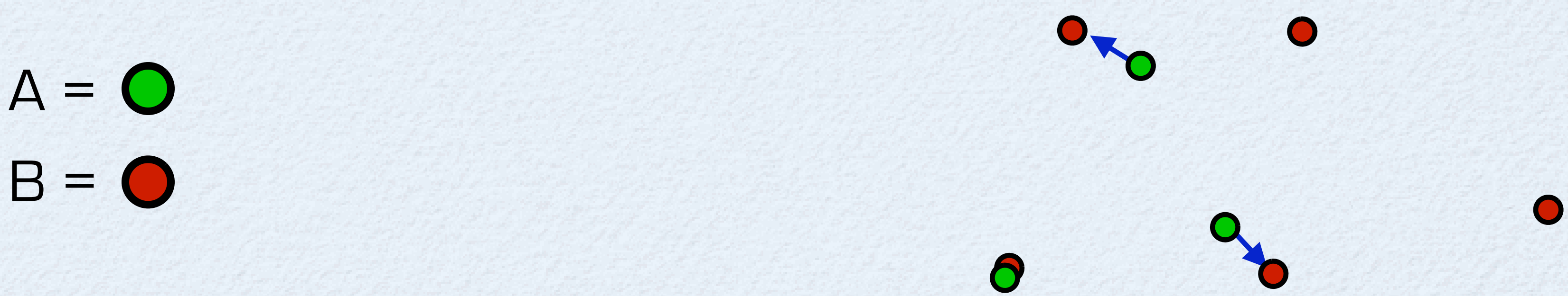
ICP: Iterative Closest Point

Given $\mathcal{A}, \mathcal{B}$, $|\mathcal{A}| = |\mathcal{B}| = n$

1. Begin with some translation x_0

2. Compute $N_{\mathcal{B}}(a + x_i)$ for each $a \in \mathcal{A}$

3. Fix $N_{\mathcal{B}}(\cdot)$, compute optimal $x_{i+1} = \sum_{a \in A} \frac{N_{\mathcal{B}}(a + x_i) - a}{|\mathcal{A}|}$



Notes on ICP

Accuracy: Bad in worst case

Time to converge: Never repeat the $N_{\mathcal{B}}(\cdot)$ function $\Rightarrow O(n^n)$

Notes on ICP

Accuracy: Bad in worst case

Time to converge: Never repeat the $N_{\mathcal{B}}(\cdot)$ function $\Rightarrow O(n^n)$

Better Bounds? [ESE SoCG 2006]: $\Omega(n \log n)$ in 2d

$O(dn^2)^d$

Notes on ICP

Accuracy: Bad in worst case

Time to converge: Never repeat the $N_{\mathcal{B}}(\cdot)$ function $\Rightarrow O(n^n)$

Better Bounds? [ESE SoCG 2006]: $\Omega(n \log n)$ in 2d

$O(dn^2)^d$

We tighten the bounds and show: $\Omega(n^2/d)^d$

Notes on ICP

Accuracy: Bad in worst case

Time to converge: Never repeat the $N_{\mathcal{B}}(\cdot)$ function $\Rightarrow O(n^n)$

Better Bounds? [ESE SoCG 2006]: $\Omega(n \log n)$ in 2d

$O(dn^2)^d$

We tighten the bounds and show: $\Omega(n^2/d)^d$

But ICP runs very fast in practice, and the worst case bounds don't do it justice.

When Worst Case is Too Bad

The theoretician's dilemma:

- An algorithm with horrible worst case guarantees - unbounded competitive ratio, exponential running time …

When Worst Case is Too Bad

The theoretician's dilemma:

An algorithm with horrible worst case guarantees - unbounded competitive ratio, exponential running time …

But: is widely used in practice.

When Worst Case is Too Bad

The theoretician’s dilemma:

- An algorithm with horrible worst case guarantees - unbounded competitive ratio, exponential running time ...
- But: is widely used in practice

From Worst to ...?

- Best-case? Average-case?

When Worst Case is Too Bad

The theoretician's dilemma:

- An algorithm with horrible worst case guarantees - unbounded competitive ratio, exponential running time ...
- But: is widely used in practice

From Worst to ...?

- Best-case? Average-case?
- Smoothed Analysis (Spielman & Teng '01)

Smoothed Analysis

What is smoothed analysis:

- Add some random noise to the input
- Look at the worst case expected complexity

Smoothed Analysis

What is smoothed analysis:

- Add some random noise to the input
- Look at the worst case expected complexity

How do we add random noise?

- Easy in geometric settings... perturb each point by $N(0, \sigma)$
- "Let P be a set of n points in general position..."

Notes on ICP

Accuracy: Bad in worst case

Time to converge: Never repeat the $N_{\mathcal{B}}(\cdot)$ function $\Rightarrow O(n^n)$

Better Bounds? [ESE SoCG 2006]: $\Omega(n \log n)$ in 2d

$O(dn^2)^d$

We tighten the bounds and show: $\Omega(n^2/d)^d$

But ICP runs very fast in practice, and the worst case bounds don't do it justice.

Theorem: Smoothed complexity of ICP is $n^{O(1)} \left(\frac{Diam}{\sigma}\right)^2$

Proof of Theorem

Outline: bound the minimal potential drop that occurs in every step.

Two cases:

1. Small number of points change their NN assignments

$\Rightarrow$ Bound the potential drop from recomputing the translation.

2. Large number of points change their NN assignments

$\Rightarrow$ Bound the potential drop from new nearest neighbor assignments.

In both cases,

Quantify how “general” is the general position obtained after smoothing.

Proof: Part I

Warm up: If every point is perturbed by $N(0, \sigma)$ then the minimum distance between points is at least ϵ with probability $1 - n^2(\epsilon/\sigma)^d$.

Proof: Part I

Warm up: If every point is perturbed by $N(0, \sigma)$ then the minimum distance between points is at least ϵ with probability $1 - n^2(\epsilon/\sigma)^d$.

Proof: Consider two points p and q. Fix the position of p. The random perturbation of q will put it at least ϵ away with probability $1 - (\epsilon/\sigma)^d$.

Proof: Part I

Warm up: If every point is perturbed by $N(0, \sigma)$ then the minimum distance between points is at least ϵ with probability $1 - n^2(\epsilon/\sigma)^d$.

Proof: Consider two points p and q. Fix the position of p. The random perturbation of q will put it at least ϵ away with probability $1 - (\epsilon/\sigma)^d$.

Easy generalization: Consider sets of up to k points.

$$P = \{p_1, p_2, \ldots, p_k\}, Q = \{q_1, q_2, \ldots, q_k\}$$

Then: $\|\sum_{p \in P} p_i - \sum_{q \in Q} q_i\| \geq \epsilon$ with probability $1 - n^{2k}(\epsilon/\sigma)^d$

Proof: Part I

Warm up: If every point is perturbed by $N(0,\sigma)$ then the minimum distance between points is at least ϵ with probability $1-n^2(\epsilon/\sigma)^d$.

Proof: Consider two points p and q. Fix the position of p. The random perturbation of q will put it at least ϵ away with probability $1-(\epsilon/\sigma)^d$.

Easy generalization: Consider sets of up to k points.

$$P=\{p_1,p_2,\ldots,p_k\}, Q=\{q_1,q_2,\ldots,q_k\}$$

Then: $\|\sum_{p\in P} p_i - \sum_{q\in Q} q_i\| \geq \epsilon$ with probability $1-n^{2k}(\epsilon/\sigma)^d$.

We will take $\epsilon=\sigma/poly(n)$ and $k=O(d)$.

Proof: part I (cont)

Recall: $x_{i+1} = \sum_{a \in A} \frac{N_{\mathcal{B}}(a + x_i) - a}{|\mathcal{A}|}$

If only k points changed their NN assignments, then with high probability $\|x_{i+1} - x_i\| \geq \epsilon/n$.

Proof: part I (cont)

Recall: $x_{i+1} = \sum_{a \in A} \frac{N_{\mathcal{B}}(a + x_i) - a}{|\mathcal{A}|}$

If only k points changed their NN assignments, then with high probability $\|x_{i+1} - x_i\| \geq \epsilon/n$.

Fact. For any set S with $c(S)$ as its mean, and any point y .

$$\sum_{s \in S} \|s - y\|^2 = |S| \cdot \|c(S) - y\|^2 + \sum_{s \in S} \|s - c(S)\|^2$$

Thus the total potential dropped by at least: $n \cdot (\epsilon/n)^2 = \epsilon^2/n$

Proof: part II

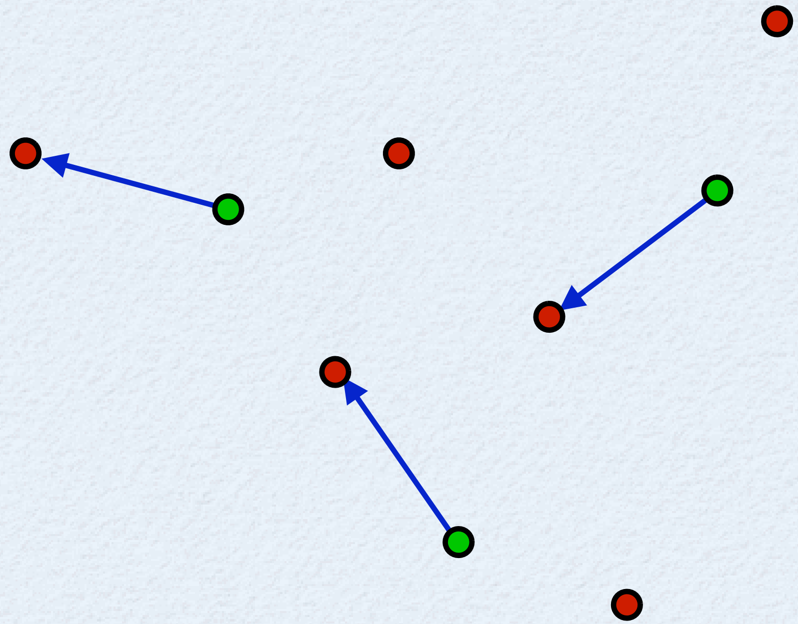
Suppose many points change their NN assignments. What could go wrong?

Proof: part II

Suppose many points change their NN assignments. What could go wrong?

A =

B =

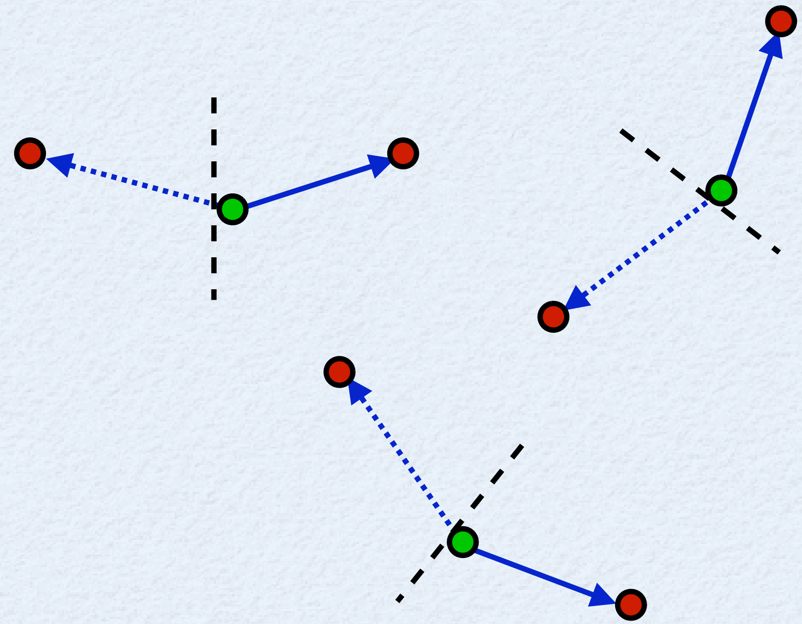


Proof: part II

Suppose many points change their NN assignments. What could go wrong?

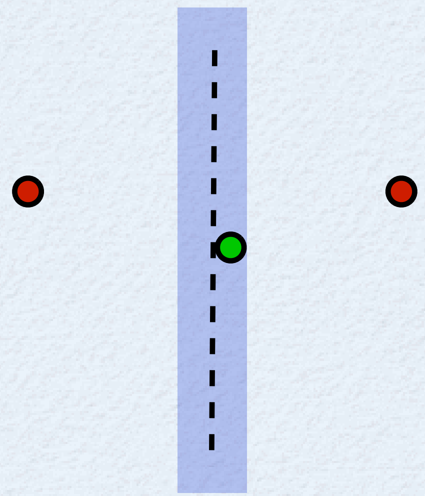
A = 

B = 



Proof: Part II Cont

What can we say about the points? Every active point in $\mathcal{A}$ must be near the bisector of two points in $\mathcal{B}$.

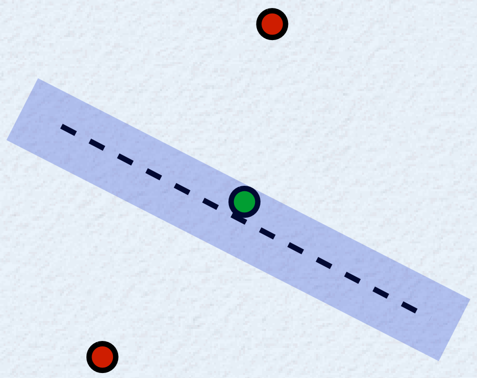


Then the translation vector must lie in this slab.

Proof: Part II Cont

What can we say about the points? Every active point in $\mathcal{A}$ must be near the bisector of two points in $\mathcal{B}$.

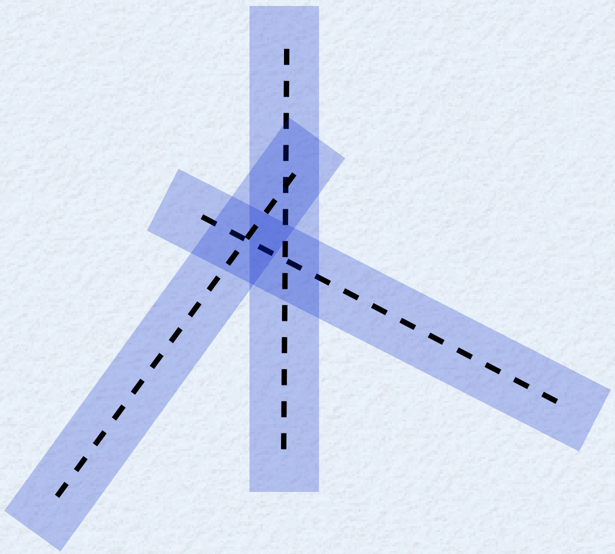
For a different point the slab has a different orientation:



And the translation vector must lie in this slab as well.

Proof: Part II Cont

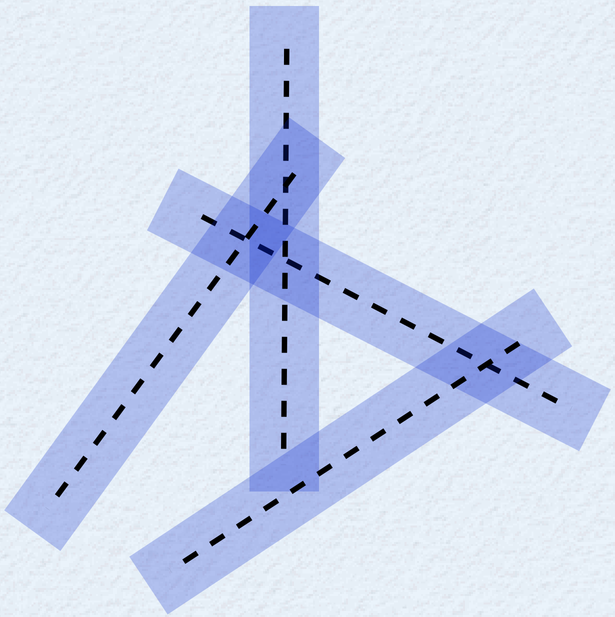
But if the slabs are narrow, because of the perturbation their orientation will appear random.



Proof: Part II Cont

But if the slabs are narrow, because of the perturbation their orientation will appear random.

Intuitively, we do not expect a large ($\omega(d)$) number of slabs to have a common intersection.



Thus we can bound the minimum slab width from below.

Proof: Finish

Theorem. With probability $1 - 2p$ ICP will finish after at most $O(n^{11} d \left(\frac{D}{\sigma}\right)^2 p^{-2/d})$ iterations.

Since ICP always runs in at most $O(dn^2)^d$ iterations, we can take $p = O(dn^2)^{-d}$ to show that the smoothed complexity is polynomial.

Proof: Finish

Theorem. With probability $1-2p$ ICP will finish after at most

$O(n^{11}d\left(\frac{D}{\sigma}\right)^2 p^{-2/d})$ iterations.

Since ICP always runs in at most $O(dn^2)^d$ iterations, we can take

$p = O(dn^2)^{-d}$ to show that the smoothed complexity is polynomial.

Many union bounds $\Rightarrow$ n^{11}

But, linear in d !

Other Geometric Heuristics?

k-means method: Popular iterative clustering algorithm, similar in spirit to ICP.

Worst case upper bound: $O(n^{kd})$ iterations.

Show a smoothed upper bound of $n^{O(k)}$: polynomial in the dimension, consistent with empirical evidence.

Big Open Question: Can we push this to $n^{O(1)}$? (Conjecture: Yes)

Conclusion

Showed worst-case ICP suffers from the curse of dimensionality.

But smoothed ICP is linear in the number of dimensions.

Similar results for the k-means (Lloyd's) method.

Techniques focus on analyzing the separation obtained by the smoothing perturbation.

Thank You