Approximation Algorithms for 3-D Common Substructure Identification in Drug and Protein Molecules

Samarjit Chakraborty

ETH Zurich

Joint work with

Somenath Biswas

Indian Institute of Technology Kanpur

Motivation

Problem:

- Given two drug or protein molecules, find the *maximal* common rigid sub-unit contained in both the molecules

Applications:

- *Pharmacophore* identification required for the design of new drugs
- Understanding mechanisms by which proteins work by identifying the common underlying structure in a group of proteins

The Equivalent Geometric Problem

Find theLCP of two 3-D point sets under ε–congruence

- A point set *P* is ε–congruent to a set *Q* if there exists an isometry I and a bijective mapping *l : P* → *Q* such that for each point $p \in P$, $d(I(P), l(P)) \leq \varepsilon$ (also known as *bottleneck matching measure*)
- The *Largest Common Point Set (LCP)* of two point sets *A* and *B* under ε–congruence is the maximum cardinality subset of *A* which is ε–congruent to some subset of *B*

Approximating the LCP

- Approximate the constraint imposed by ε
- Approximate the size of the LCP

Approximating the ε constraint

Let the required isometry I for the exact solution map:

- point *a* to the ε-ball around *a'*
- point *b* to the ε-ball around *b'*
- point *c* to the ε-ball around *c'*

For the approximation algorithm consider the following isometry I_{approx}:

- map *a* to *a'*
- align the vector *ab* with *a'b'*
- align *c* and *c'* by rotating *c* around the *ab* axis

The Approximation Algorithm

1. For all triplets of points from the set *A*
2. For all triplets of points from the set *B*
3. Compute the induced isometry I_{approx}
4. Apply this isometry to the set *A* and compute the number of distinct matching points in the set *B* which are within 8ε distance from points of *A*
5. Output the isometry which corresponds to the maximum number of matchings

8ε *Approximation Algorithm*

Theorem (follows from Goodrich, Mitchell and Orletsky)

- The algorithm returns a subset of cardinality at least as large as the LCP between *A* and *B* (under ε-congruence).
- Each point of this subset is at most within 8ε distance of a distinct point of the set *B*

Proof

If *a,b* and *c* are the farthest points of the *LCP* ⊆ *A* then

- Each point should be originally ε distance away from a point of *B*
- Mapping *a* to *a'* moves *a* by at most ε from its required position
- Aligning the vectors *ab* with *a'b'* moves *b* by at most *2*ε
- Aligning *c* with *c'* moves *c* by atmost *4*ε
- Hence the total displacement of any point is at most ε + ε + 2ε + 4ε = 8ε

Approximating the Size of the LCP

Definition: For point sets *A* and *B,* isometry I, and a real number ε, let $G(I, \varepsilon, A, B)$ be the bipartite graph where the nodes correspond to the points of *A* and *B,* and the edges join all points *a, b* such that the distance between I $I(a)$ and *b* is at most ε.

Definition: If the cardinality of a common subset is *n* then $\varepsilon_{min}(n)$ denotes the minimum ε for which it exists.

A Partial Decision Algorithm

A Decision Algorithm:

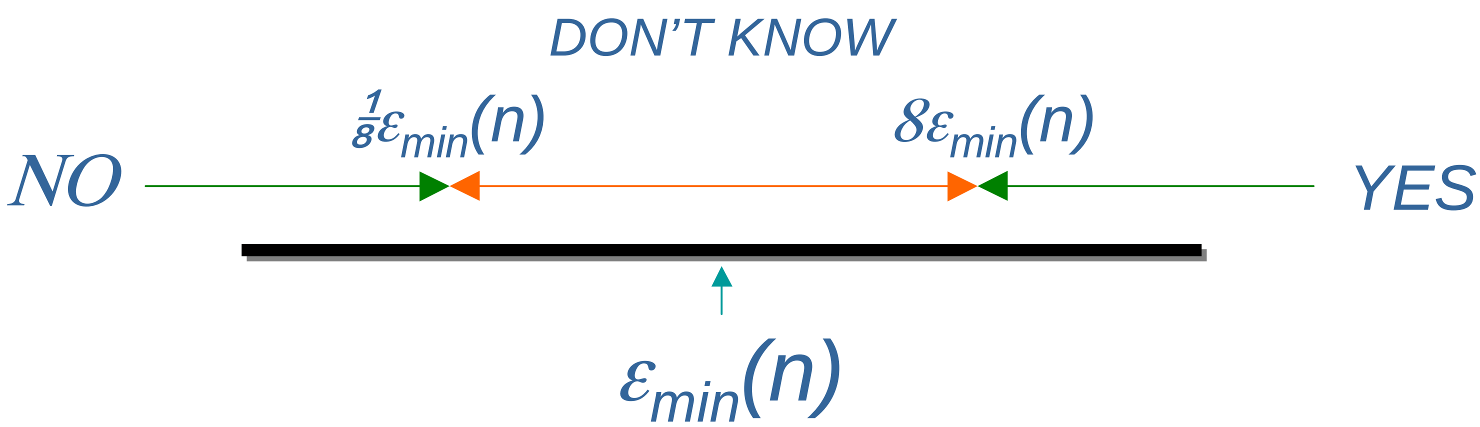
Input: Point sets *A, B,* real ε, and an integer *n*

Output: *YES* if there exists a common subset between *A* and *B* of size at least *n* under ε congruence, otherwise *NO*

Partial Decision Algorithm (based on Schirra):

Output: Might return *DON'T KNOW* if ε lies in the range

$\left[\left(\frac{1}{8}\varepsilon_{min}(n), 8\varepsilon_{min}(n)\right)\right)$


DON'T KNOW
$\frac{1}{8}\varepsilon_{min}(n)$
$8\varepsilon_{min}(n)$
NO
YES
$\varepsilon_{min}(n)$

Algorithm

1. For all triplets of points from *A*
2. For all triplets of points from *B*
3. Compute the induced transformation I_{approx}
4. If $G(I_{approx}, \varepsilon, A, B)$ has a matching of size $\geq n$ then return *YES*

5. *Decision* = *NO*
6. For all triplets of points from *A*
7. For all triplets of points from *B*
8. Compute the induced transformation I_{approx}
9. If $G(I_{approx}, 8\varepsilon, A, B)$ has a matching of size >= n then *Decision* = *YES*

10. If *Decision* = *NO* then return *NO* else return *DON'T KNOW*

Approximating the Size

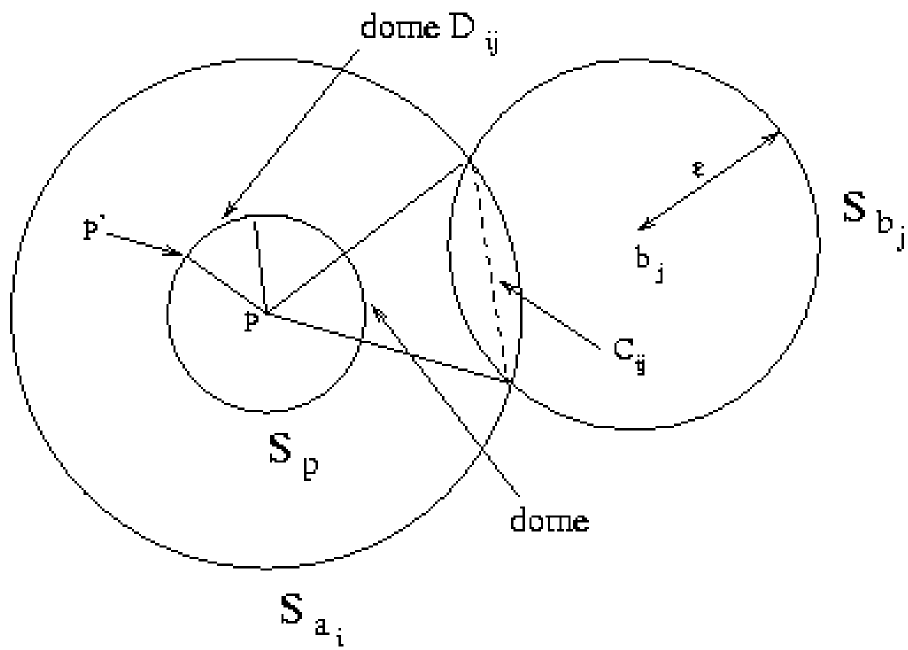
- Given ε find the maximum value of *n* for which the algorithm returns *YES* - This is the lower bound on the size of the LCP
- Find the minimum value of *n* for which the algorithm returns *NO* - This is the upper bound on the size of the LCP

Theorem

$$\max \{n : \varepsilon > 8\varepsilon_{min}(n)\} \leq n_l \leq n_{max}(\varepsilon) < n_u \leq \min \{n : \varepsilon \leq \tfrac{1}{8}\varepsilon_{min}(n)\}$$

An Exact Algorithm for Pure Rotation

Rotating the set A about a fixed point p:



Improving the Approximation Ratios

Lemma (follows from Schirra) Let isometry I, which is a composition of translation and rotation, result in a common subset of size *n* under ε-congruence. Let *a* be a point of this set. Then there is a rotation of the set *A* when translated so that *a* lies on $I(a)$, such that this enables in finding the same subset.

The factor of *8* in the approximation algorithms can be reduced to *2* by making use of the exact algorithm for rotation.

2ε Approximation Algorithm

Input: Point sets *A, B,* real number $\varepsilon > 0$

$n := 0$

for each point $a \in A$

  for each point $b \in B$

    $m :=$ *LCP-ROT*$(t_{ab}(A), B, b, 2\varepsilon)$

    if $(m > n)$ **then** $m := n$

return n

Theorem The algorithm returns a subset of cardinality at least as large as the LCP between *A* and *B* (under ε-congruence) and each point of this subset is at most within 8ε distance of a distinct point of the set *B*

Running Time

$n := 0$	
for each point $a \in A$	$O(n)$
for each point $b \in B$	$O(n)$
$m := LCP\text{-}ROT(\, t_{ab}(A),\, B,\, b,\, 2\varepsilon\,)$	$O(n^{6.5})$
if $(\, m > n\,)$ **then** $m := n$	
return n	

- The overall running time is $O(n^{8.5})$
- Hopcroft and Karp's algorithm for finding the maximum matching in a bipartite graph takes $O(n^{2.5})$ *time*

Improvements in Running Time

- Approximate graph matching when the nodes of the bipartite graph are points in some *d*-dimensional space and the edges are pairs of points within some specified distance (due to Efrat and Itai)
- Random Sampling

Approximation Algorithm for Maximum Matching

- Finds the maximum matching in a graph G where $G(I_{approx}, \varepsilon, A, B) \subseteq G \subseteq G(I_{approx}, (1+\delta)\varepsilon, A, B)$
- δ is a parameter of the algorithm for approximately answering nearest neighbor queries for point sets in $\Re^d$
- $O(n^{1.5} \log n)$ running time in contrast to $O(n^{2.5})$

Resulting Decision Algorithm

- $\varepsilon \geq 8\varepsilon_{min}(n)$ always returns *YES*
- $\varepsilon < \varepsilon_{min}(n)/\ 8(1+\delta)$ always returns *NO*
- $\varepsilon \in [\,\varepsilon_{min}(n)/\ 8(1+\delta),\ \varepsilon_{min}(n)/\ (1+\delta)\,) \cup [\,\varepsilon_{min}(n),\ 8\varepsilon_{min}(n)\,)$ either returns the correct answer or *DON'T KNOW*
- $\varepsilon \in [\,\varepsilon_{min}(n)\ /\ (1+\delta),\ \varepsilon_{min}(n)\,)$ might return *YES, NO,* or *DON'T KNOW*
- The transformation along with the bijective mapping that results in the algorithm to return *YES* results in each point of the set *A* to be within $(1+\delta)\varepsilon$ distance of the corresponding point of the set *B*

Resulting Approximation Algorithm

- max $\{n : \varepsilon > 2\varepsilon_{min}(n)\} \leq n_l \leq n_{max}((1+\delta)\varepsilon)$
  $n_{max}(\varepsilon) < n_u \leq \min \{n : \varepsilon < \varepsilon_{min}(n) / 2 (1+\delta)\}$
- Algorithm for finding the LCP under pure rotation runs in $O(n^{5.5} \log n)$ time in contrast to $O(n^{6.5})$
- Overall running time is $O(n^{7.5} \log n)$

Using Random Sampling

- X is a multiset of cardinality k randomly sampled from the set A
- Running time $O(n^{7.5})$

Theorem

- For any $k \geq \lceil (1/\alpha) \ln(1-q) \rceil$ the decision algorithm returns *YES* with probability atleast q for all $\varepsilon \geq 2\varepsilon_{min}(n)$
- For all $\varepsilon < \frac{1}{2}\varepsilon_{min}(n)$ the algorithm always returns *NO*
- For $\frac{1}{2}\varepsilon_{min}(n) \leq \varepsilon < \varepsilon_{min}(n)$ it either returns *NO* or *DON'T KNOW*

A and B are of cardinality n and $\alpha \leq 1$ is the ratio of the size of the LCP and n

The final algorithm has a running time of $O(n^{6.5} \log n)$ but in contrast to definitely returning *YES*, it returns *YES* only with probability $\geq q$