Optimization in Shape Matching in the Context of Molecular Docking

Vipin Kumar Tripathi

Thesis Supervisors

Dr. Bhaskar Dasgupta and Dr. Kalyanmoy Deb

Department of Mechanical Engineering

Indian Institute of Technology Kanpur

Kanpur, Pin 208 016 INDIA

{vipinkt,dasgupta,deb}@iitk.ac.in

Overview of Presentation

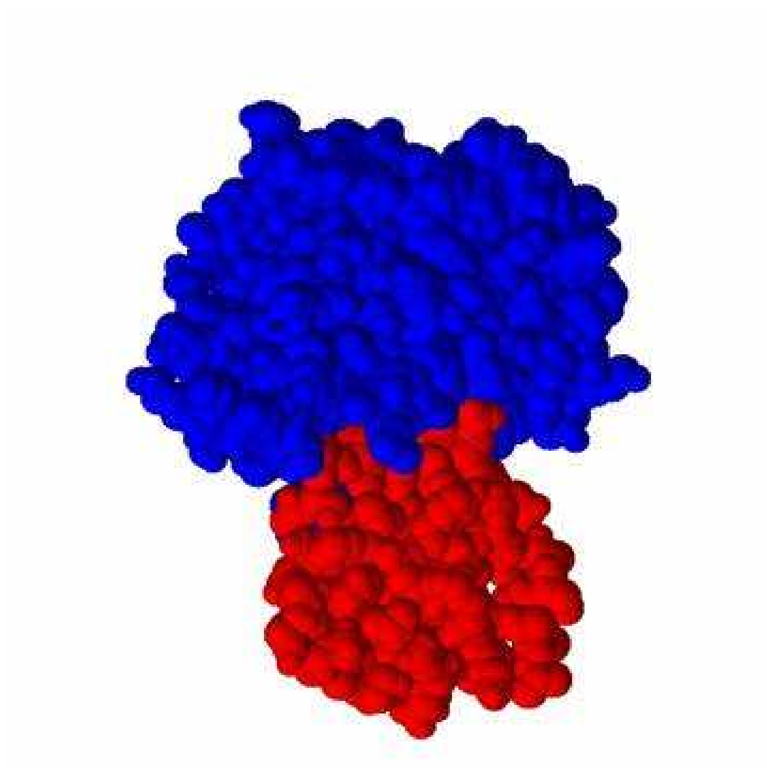
- Introduction
- Literature Review
- Modeling of Shape
- Shape Matching
- Scoring
- Optimization of Shape Matching
- Ranking
- Conclusions and Future Work

Introduction

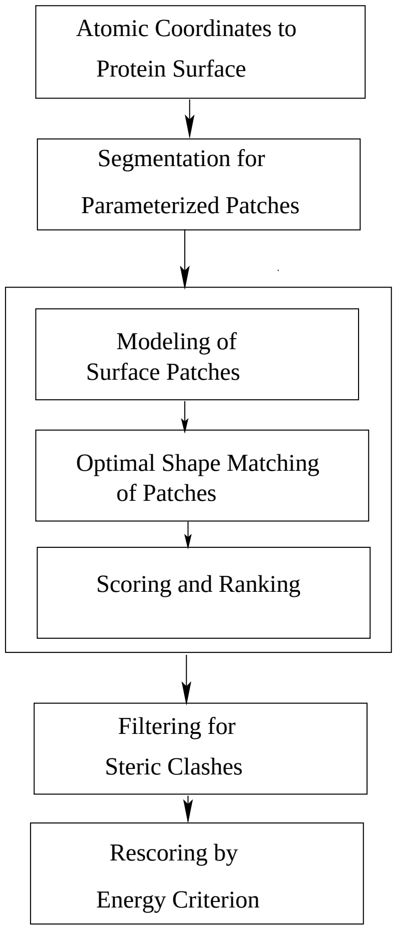
- Optimal shape matching problems occur in different areas
  - Design and Manufacturing
  - Computer Graphics
  - Bio-informatics
  - Archaeology
  - Medical Imaging
- Non-exact shapes increase difficulty

Molecular Docking

- Problem of maximizing the molecular interaction
- Computationally difficult
- Rigid body docking and flexible docking



Docking Procedure



Major Sections of the Problem

- Modeling of molecular shape
- Shape matching
- Scoring
- Optimization of shape matching
- Ranking

Applications of Docking

- Many Biological processes rely upon molecular interactions
- Rational drug design
- Study of Protein folding

Motivation

- Molecular docking is a new research field
- Docking is a challenging optimization problem
- Despite progress, it is far from being solved
- Drawbacks in docking strategies
  - Complex surface representation schemes
  - Time-consuming shape matching strategies
  - Unreliable scoring functions

Objectives of the Present Work

- To develope an algorithm for optimal shape matching
- Fast and efficient computation with simple schemes
- Systematic synthesis of principles from different fields for solution of molecular docking problem
- Simple surface representation scheme
- Development of a fast and robust scoring function

Scope of the Present Work

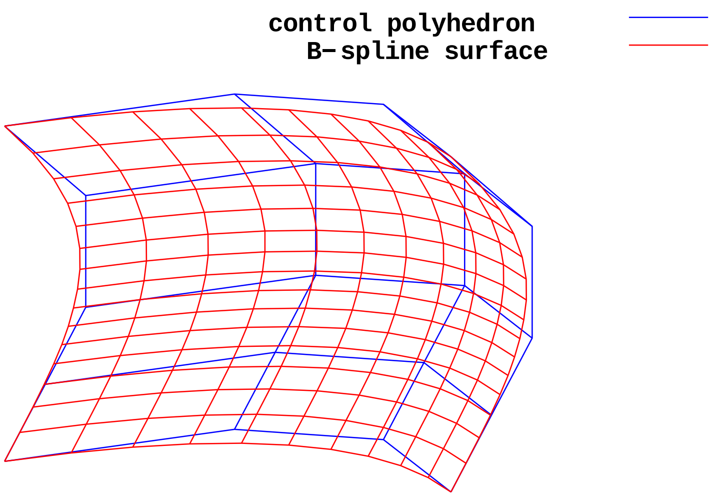
- No consideration of flexibilty of molecules
- Shape complementarity is the only criterion used for docking
- Method expects pre-processed data available on a 2-D grid as input

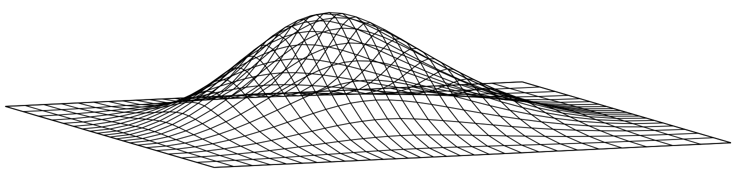
Literature Review

- Principles and methods of docking are described by Kuntz (1992), Abagyan and Totrov (2001) and Halperin et al. (2002).
- Geometry based approach initiated by Kuntz et al. (1982) and elaborated by Connolly (1986), Norel et al. (1999), etc.
- Bajaj et al. (1997) used NURBS for shape representation.
- Connolly (1986), Fischer et al. (1995), Norel et al. (1999), Goldman and Wipke (2000), Gardiner et al. (2001), etc. have used local shape feature matching algorithms.
- Lin et al. (1994), use the area shared between the two matching dots for scoring.

Modeling of Shape

- Molecular surface represented as a network of connected patches
- B-spline surface representation used for patches





B-spline Surface Representation

Tensor product B-spline surface is formulated as

$$\mathbf{Q}(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} \mathbf{P}_{ij} N_{i,k}(u) N_{j,l}(v)$$

where,
$\mathbf{P}_{ij} = (n+1) \times (m+1)$ vertices of the defining polyhedron,
$N_{i,k}(u),\ N_{j,l}(v)$ = Blending functions in u and v parametric directions, respectively.

B-spline Surface Representation Contd..

The B-spline blending functions are computed recursively:

$$N_{i,1}(u) = \begin{cases} 1 & \text{if } t_i \leq u \ < t_{i+1}, \\ 0 & \text{otherwise,} \end{cases}$$

$$N_{i,k}(u) = \frac{(u - t_i)N_{i,k-1}(u)}{t_{i+k-1} - t_i} + \frac{(t_{i+k} - u)N_{i+1,k-1}(u)}{t_{i+k} - t_{i+1}},$$

B-spline Surface Representation Contd..

The t_i are knot values. For an open curve

$$t_i = \begin{cases} 0 & \text{if } i < k, \\ i - k + 1 & \text{if } k \le i \le n, \\ n - k + 2 & \text{if } i > n. \end{cases}$$

Range of parametric variable u is

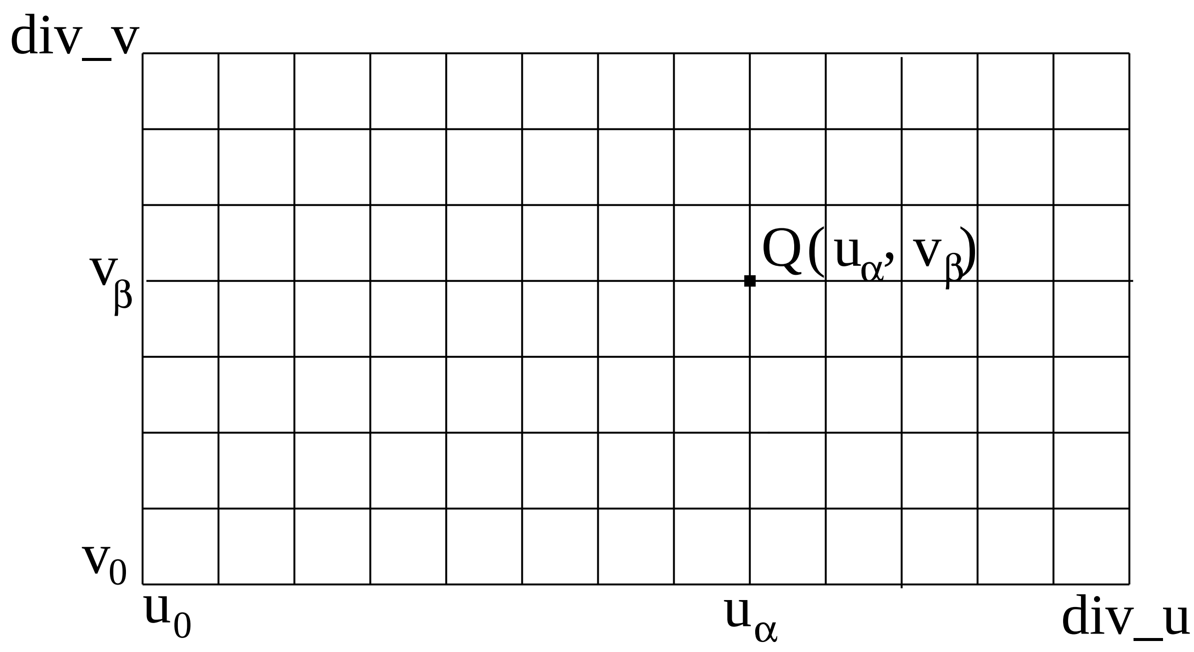
$$0 \le u \le n - k + 2.$$

Statement of Shape Modeling Problem

- Here the inverse problem is of interest, i.e., given a known set of data points on a surface, determine the defining polyhedron for B-spline surface that best approximates the data.
- Solution of the inverse problem fulfils two requirements:
  1. It converts the huge amount of surface data into a small number of control points which are easy to store and process.
  2. With the obtained set of control points, any surface data point corresponding to a given pair of u and v parameters can be computed easily.

Statement of Shape Modeling Problem Contd..

The discussion here is confined to topologically rectangular nets



Statement of Shape Modeling Problem Contd..

For a single surface data point, say, point $\mathbf{Q}(u_\alpha, v_\beta)$

$$\mathbf{Q}(u_\alpha, v_\beta) = \begin{bmatrix} N_{0,k}(u_\alpha) & \cdots & N_{n,k}(u_\alpha) \end{bmatrix} \begin{bmatrix} \mathbf{P}_{ij} \end{bmatrix} \begin{bmatrix} N_{0,l}(v_\beta) \\ N_{1,l}(v_\beta) \\ \vdots \\ N_{m,l}(v_\beta) \end{bmatrix}.$$

For all the data points equation in matrix form

$$[\mathbf{Q}] = [\mathbf{N}_\mathbf{A}^\mathbf{T}][\mathbf{P}_{ij}][\mathbf{N}_\mathbf{B}]$$

Statement of Shape Modeling Problem contd..

Now inverse problem has two parts:

1. How to form the standard $\{\mathbf{q}\} = [\mathbf{A}]\{\mathbf{p}\}$ form ?
2. How to solve system $\{\mathbf{q}\} = [\mathbf{A}]\{\mathbf{p}\}$?

Solution of Inverse Problem

- Part 1:
  The matrix $\mathbf{A}$ can be formed as

$$\mathbf{A}_{I,J} = n_{B(J,I)} \mathbf{N}_{\mathbf{A}}^{\mathbf{T}}$$

  where $n_{B(J,I)}$ is the (J, I)-th element of matrix $\mathbf{N}_{\mathbf{B}}$.

- Part 2:
  System $[\mathbf{A}^{\mathbf{T}}\mathbf{A}]\mathbf{p} = [\mathbf{A}^{\mathbf{T}}]\mathbf{q}$ is solved by Cholesky decomposition.
  If Cholesky decomposition fails, a method based on singular value decomposition is used.

Proposed Algorithm for Modeling of Shape

Step 1 : Input parameters n, k, m, l, div_u, div_v where $(n+1) \times (m+1)$ is the number of control points of the surface, $(div\_u + 1) \times (div\_v + 1)$ is the number of surface data points, k is the order of curve in u parametric direction, l is the order of curve in v parametric direction.

Step 2 : Calculate the knot vectors $\mathbf{t_i}$ in u and v parametric directions.

Step 3 : Calculate blending functions $N_{i,k}(u)$ and $N_{j,l}(v)$ in u and v parametric directions.

Step 4 : Form matrices $\mathbf{N_A^T}$ and $\mathbf{N_B}$.

Step 5 : Form matrix $\mathbf{A}$.

Step 6 : Input vector $\mathbf{q}$ containing one of the coordinates of surface data points.

Step 7 : Form matrix $[\mathbf{A^T A}]$.

Step 8 : Use Cholesky decomposition for solution of system $[\mathbf{A^T A}]\mathbf{p} = [\mathbf{A^T}]\mathbf{q}$.

Step 9 : If Cholesky decomposition fails due to matrix $[\mathbf{A^T A}]$ not being positive-definite, use singular value decomposition of matrix $\mathbf{A}$ to solve $\mid \mathbf{Ap} - \mathbf{q} \mid$ in least square sense, to obtain vector $\mathbf{p}$ of one coordinate of control points.

Step 10 : Repeat steps 6 to 9 for different input vectors containing a set of different coordinate of surface data points every time.

Computational complexity of the algorithm is governed by the steps 7 to 9 and is of the order $O(n^3)$.

Simulation Results

For Cholesky decomposition and singular value decomposition of the matrix the routines from GNU scientific library have been used.

- Surface Data Points: The surface point data for different patches is available at

  `www.iitk.ac.in/kangal/bioinformatics/docking.html.`

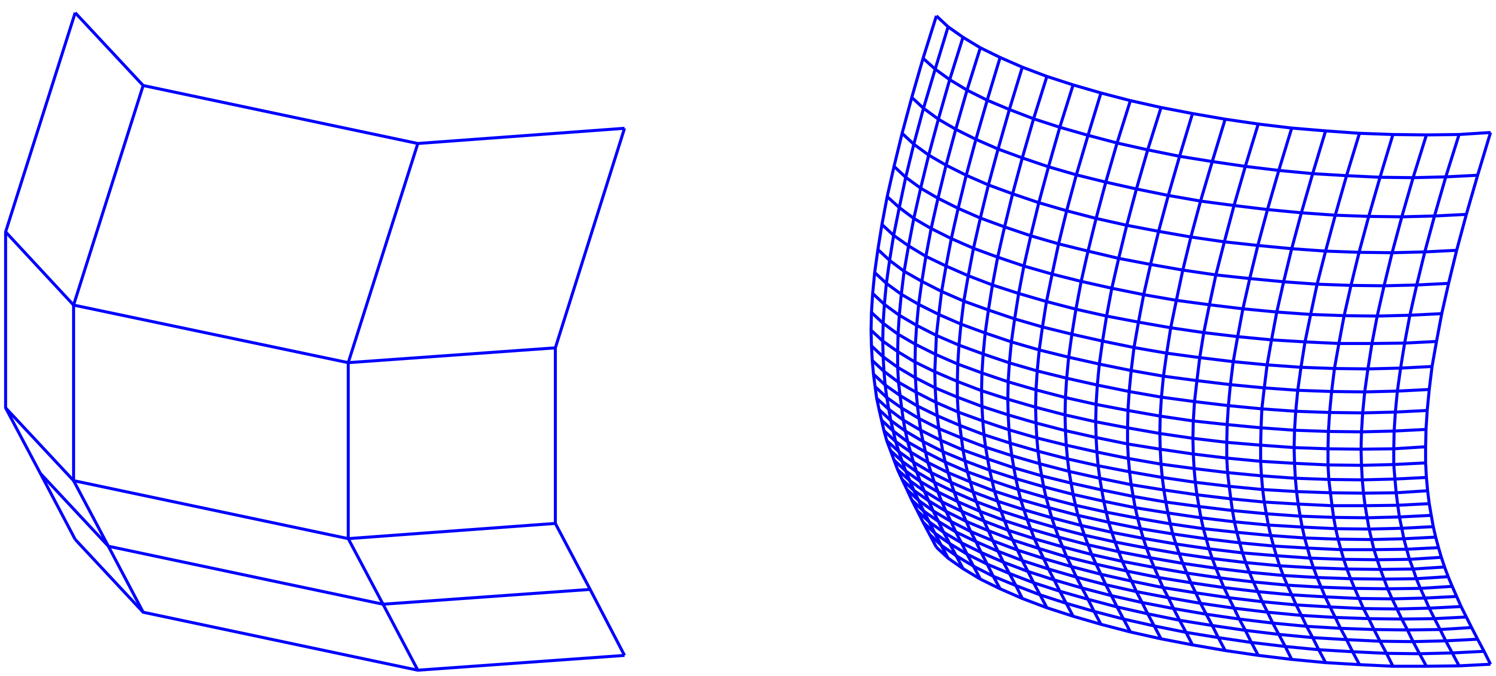
- Control points: The values of parameters taken in the procedure are:

$$\begin{aligned} div\_u &= 20, \quad & div\_v &= 30 \\ n &= 3, & m &= 4 \\ k &= 4, & l &= 4 \end{aligned}$$

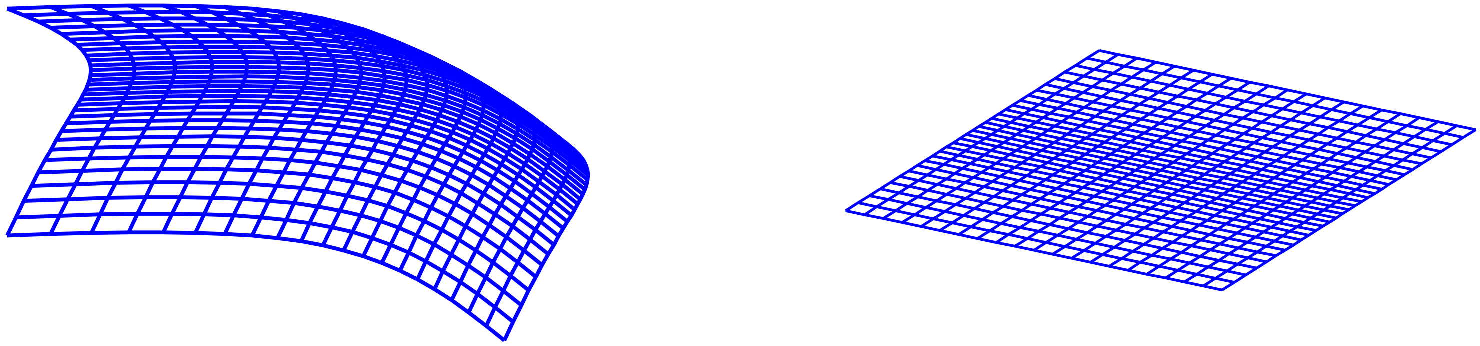
Control Points

-10 0 15
-10 -5 5
-10 -5 -5
-10 -2.5 -10
-10 0 -15
-5 -5 15
-5 -10 5
-5 -10 -5
-5 -7.5 -10
-5 -5 -15
5 -5 15
5 -10 5
5 -10 -5
5 -7.5 -10
5 -5 -15
10 0 15
10 -5 5
10 -5 -5
10 -2.5 -10
10 0 -15

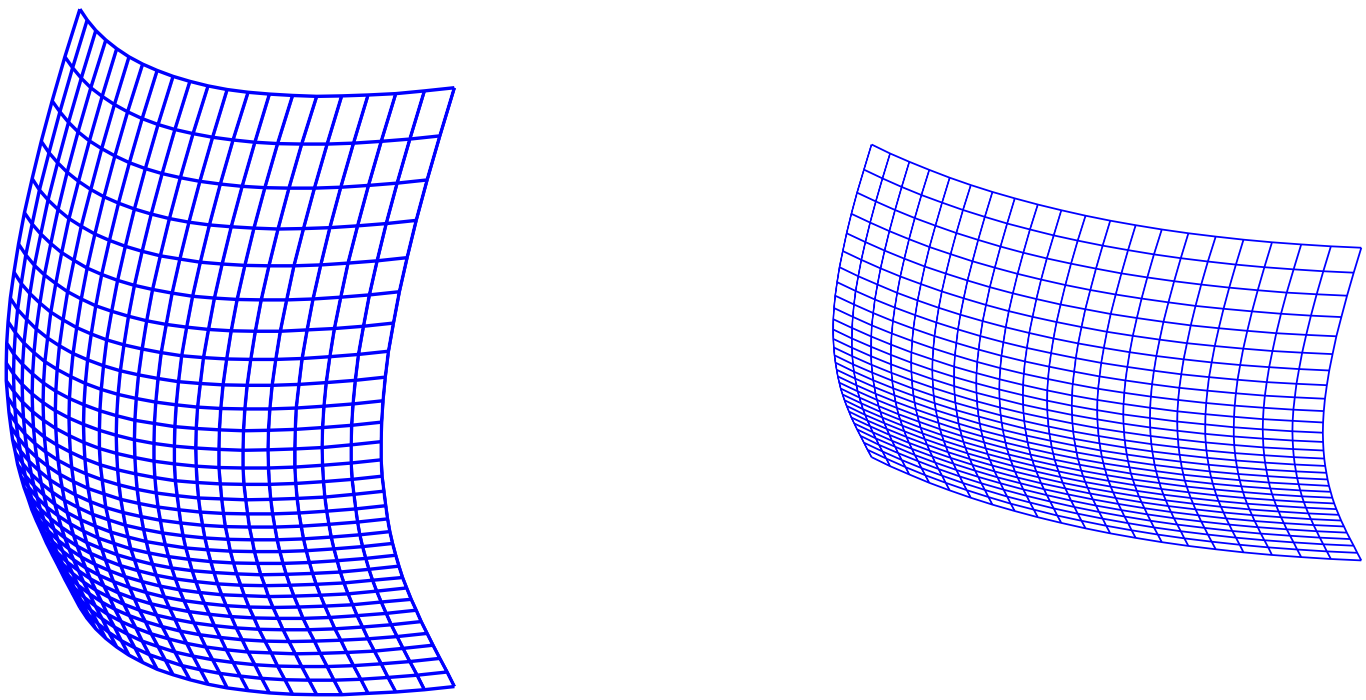
Control Polyhedron and Surface Patch



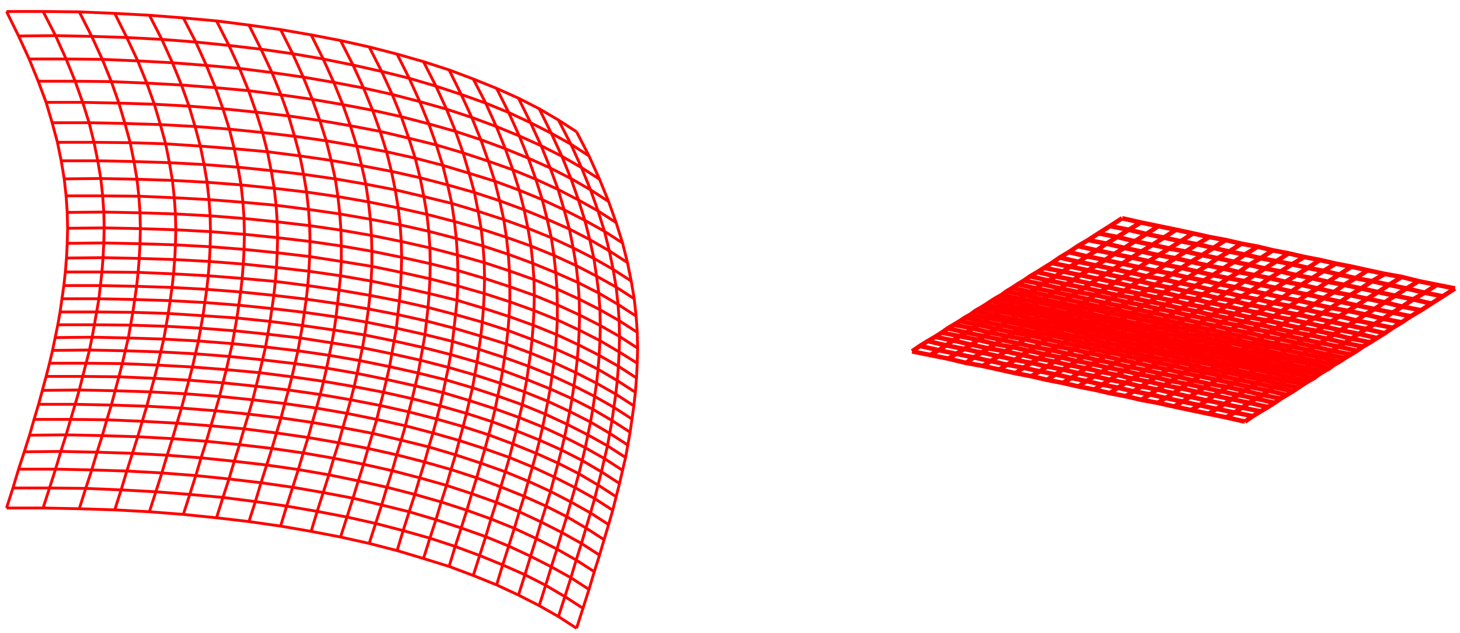
Surface Patches 2 & 3 of First Molecule



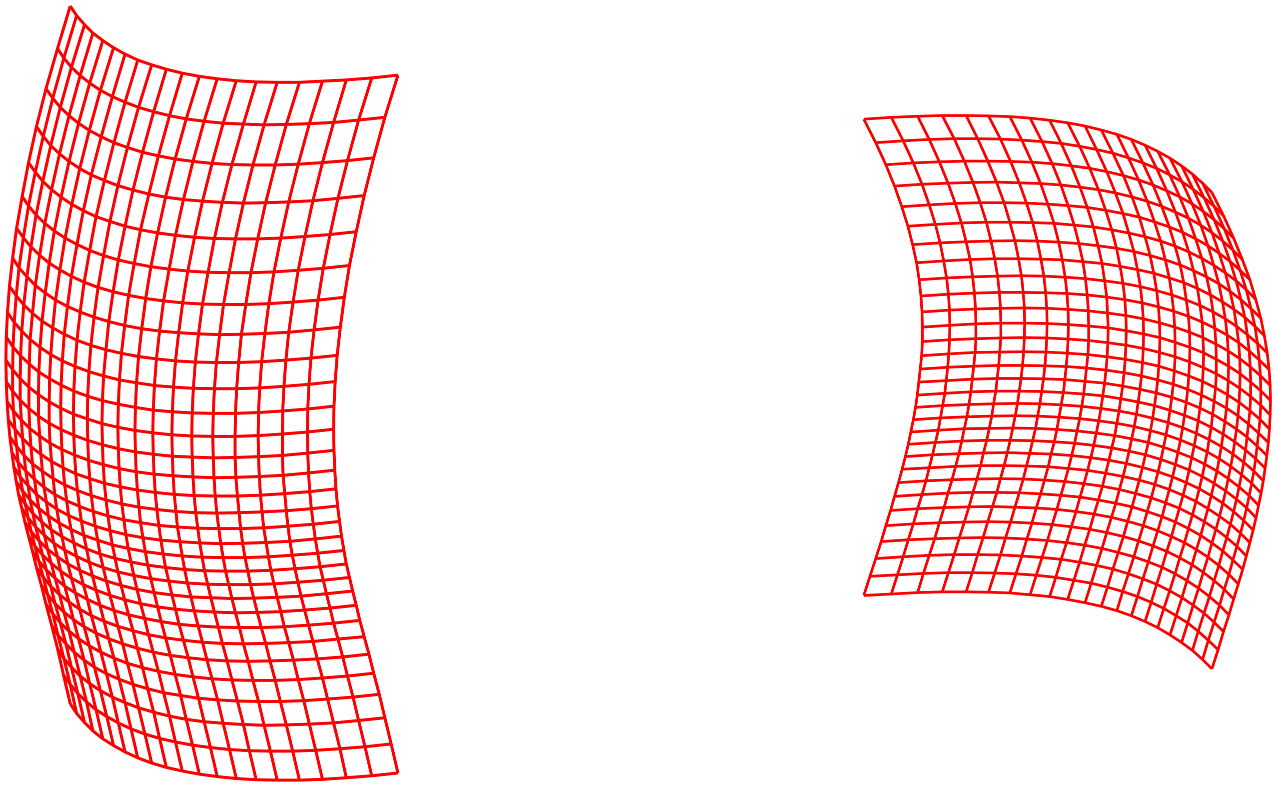
Surface Patches 4 & 5 of First Molecule



Surface Patches 1 & 2 of Second Molecule



Surface Patches 3 & 4 of Second Molecule



Statement of Shape Matching Problem

Given the control points of two surface patches, check whether the patches are complementary. If the patches are complementary, compute:

1. The translation required to match two given surface patches at a desired point.
2. The rotation required to match the surface patches in the desired orientation.

Solution Strategy and Formulation

- The Gaussian curvature K and mean curvature H are defined as

$$K = \frac{LN - M^2}{EG - F^2}$$

$$H = \frac{EN + GL - 2FM}{2(EG - F^2)}.$$

- Principal curvatures are expressed in terms of H and K

$$\kappa_{max} = H + \sqrt{H^2 - K}$$
$$\kappa_{min} = H - \sqrt{H^2 - K}.$$

Check for Complementarity

If

κ_{max1} = maximum curvature of the first surface patch at the desired point,

κ_{min1} = minimum curvature of the first surface patch at the desired point,

κ_{max2} = maximum curvature of the second surface patch at the desired point,

κ_{min2} = minimum curvature of the second surface patch at the desired point,

the surface patches are **not** complementary when

$\kappa_{max1} \times \kappa_{max2} > 0$ or $\kappa_{min1} \times \kappa_{min2} > 0$.

Translation and Rotation

- Translation of the first surface patch to origin

$$\mathbf{P}_{1ijT} = \mathbf{P}_{1ij} - \mathbf{Q}_1$$

- Translation of the second surface patch to origin

$$\mathbf{P}_{2ijT} = \mathbf{P}_{2ij} - \mathbf{Q}_2$$

- Rotation matrix:
  V and **W** are unit vector matrices

$$[\mathbf{x}\ \mathbf{y}\ \mathbf{z}]^T = [\mathbf{q}_{u1}\ \mathbf{m}_1\ \mathbf{n}_1]^T[\mathbf{R}].$$

Translation and Rotation contd..

- Required rotation matrix for system $\mathbf{V}$ with respect to system $\mathbf{W}$

$$[\mathbf{R}] = [\mathbf{V}]^{-1}[\mathbf{W}].$$

- Rotation for the first surface patch

$$\mathbf{P}_{1ijR} = \mathbf{P}_{1ijT} \times \mathbf{R}_1$$

- Rotation for the second surface patch

$$\mathbf{P}_{2ijR} = \mathbf{P}_{2ijT} \times \mathbf{R}_2$$

Proposed Algorithm for Shape Matching

Step 1 : Compute the principal curvatures of both the surface patches and check for complementarity.

Step 2 : If surface patches are complementary, input the values of parameters u and v and the control points $\mathbf{P}_{1ij}$ for the first surface patch. Find translation $\mathbf{Q}_1$.

Step 3 : Obtain changed control points of the first surface patch after translation $\mathbf{P}_{1ijT}$.

Step 4 : Compute triad $(\mathbf{q}_{u1}, \mathbf{m}_1, \mathbf{n}_1)$ and form matrix $\mathbf{V}$ as $[\mathbf{q}_{u1}, \mathbf{m}_1, \mathbf{n}_1]^T$.

Proposed Algorithm for Shape Matching contd.

Step 5 : If principal curvatures of the first surface patch are positive, form matrix $\mathbf{W}$ as $[\mathbf{x}, \mathbf{y}, \mathbf{z}]^T$ and if they are negative, form matrix $\mathbf{W}$ as $[\mathbf{x}, \mathbf{y}, -\mathbf{z}]^T$.

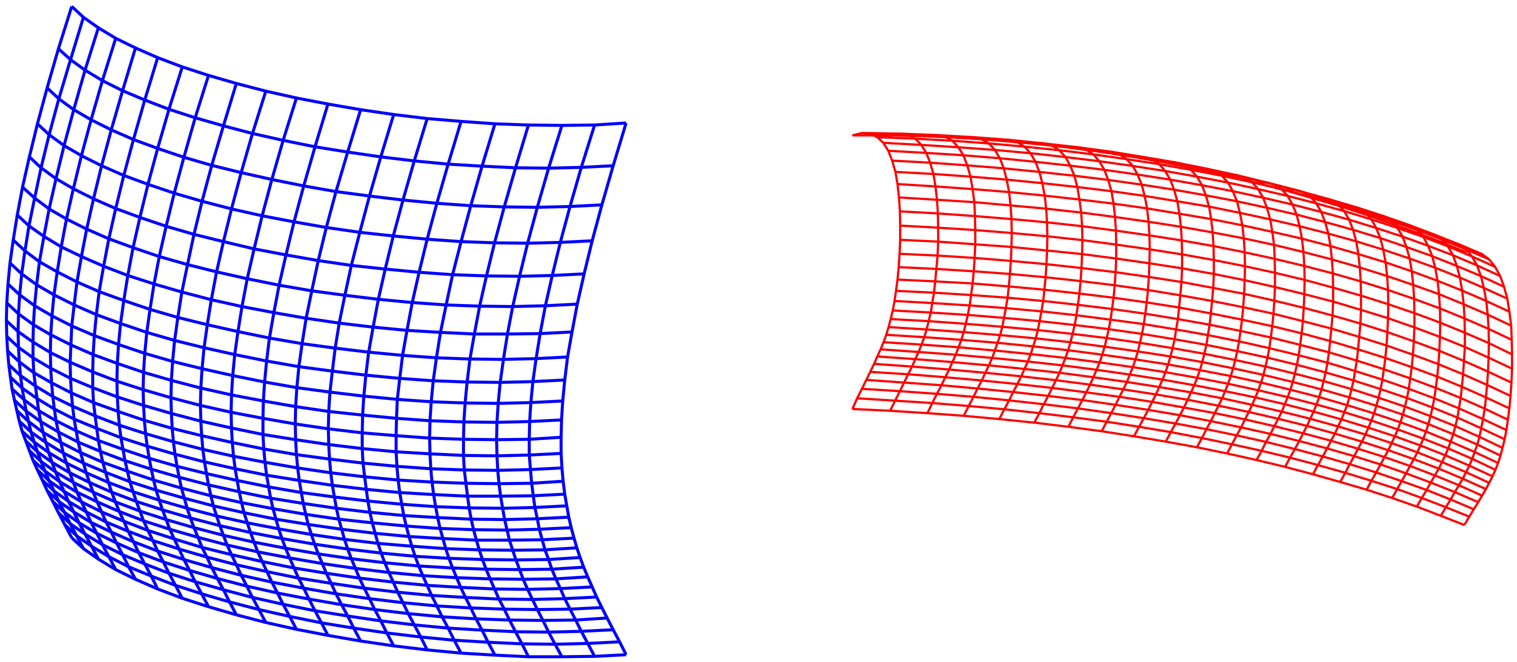
Step 6 : Find rotation matrix $\mathbf{R}_1$.

Step 7 : Obtain changed control points of the first surface patch after rotation $\mathbf{P}_{1ijR}$.

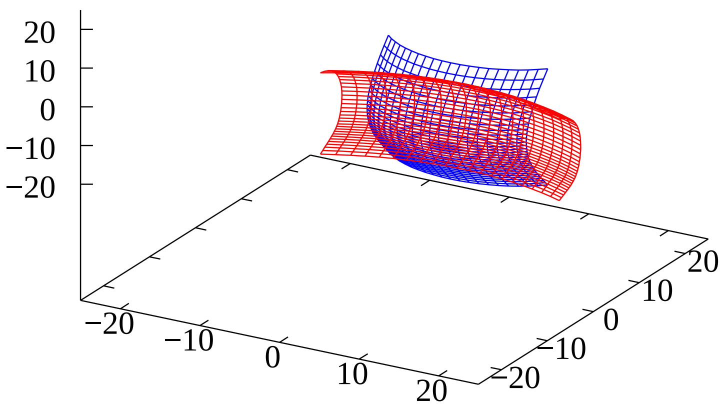
Step 8 : Repeat steps 3 to 7 for the second surface patch.

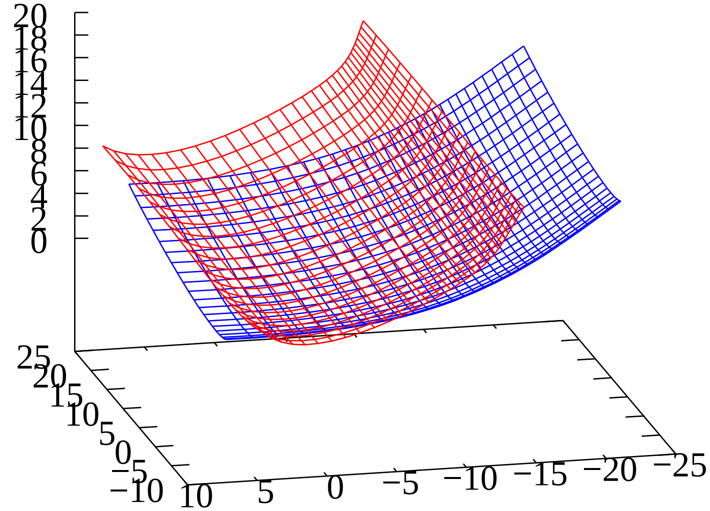
Step 9 : Generate both the surface patches using changed control points after rotation $\mathbf{P}_{1ijR}$ and $\mathbf{P}_{2ijR}$.

Shape matching Results



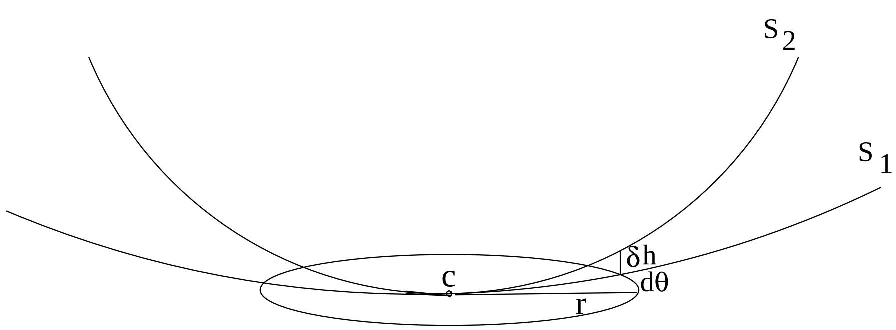
Shape matching Results contd..





Proposed Scoring Function

- Normalized volume mismatch for the best matching part ($\bar{V}$) is the proposed scoring function.
- Volume mismatch (ΔV) between two surfaces can be defined as solid volume entrapped between them up to the radius of the best matching part.



Proposed Scoring Function contd..

- Normalized volume mismatch can be expressed as a dimensionless quantity

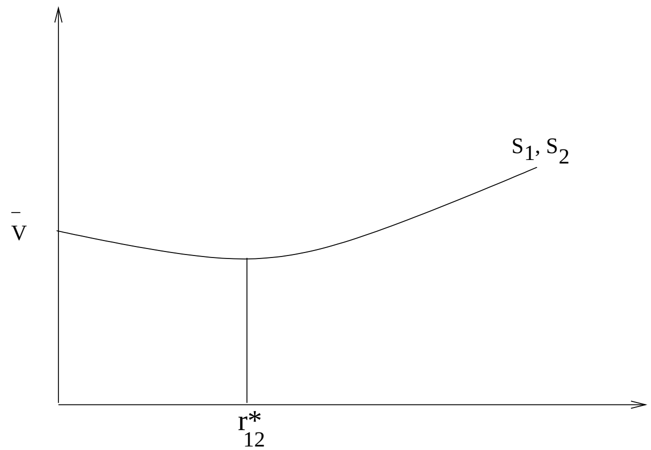
$$\bar{V} = \frac{\Delta V}{{r^*}^3}$$

- Volume mismatch between any two matching surface patches S_1 and S_2 can be expressed as

$$\Delta V = \int \oint (\delta h) r dr d\theta$$

Proposed Scoring Function contd..

- Radius of the best matching portion



- Determination of $\delta\mathbf{h}$ by solving the following simultaneous equations for unknowns u and v,

$$x(u, v) - x_0 = 0$$
$$y(u, v) - y_0 = 0$$

Multi-dimensional Root Finding

- Newton-Raphson method is used for solving simultaneous equations.
  Simultaneous equations can be written

$$\begin{bmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} \end{bmatrix} \begin{bmatrix} \delta u \\ \delta v \end{bmatrix} = \begin{bmatrix} x_0 - x \\ y_0 - y \end{bmatrix} \tag{1}$$

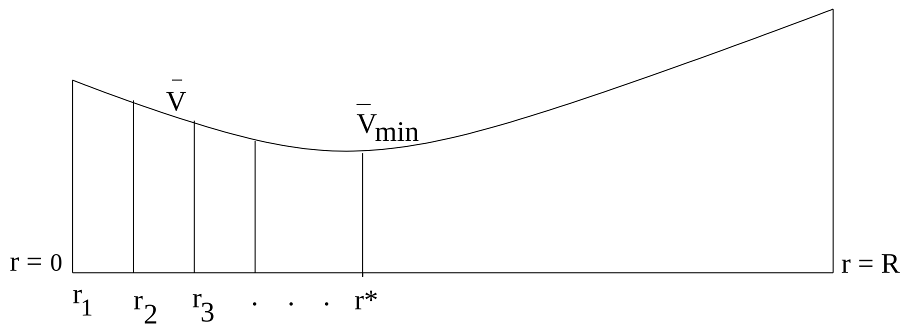
- Corrections to solution vector are obtained as given below

$$\delta u = \frac{(y_0 - y)\frac{\partial x}{\partial v} - (x_0 - x)\frac{\partial y}{\partial v}}{\frac{\partial x}{\partial v} \cdot \frac{\partial y}{\partial u} - \frac{\partial x}{\partial u} \cdot \frac{\partial y}{\partial v}} \tag{2}$$

$$\delta v = \frac{(x_0 - x)\frac{\partial y}{\partial u} - (y_0 - y)\frac{\partial x}{\partial u}}{\frac{\partial x}{\partial v} \cdot \frac{\partial y}{\partial u} - \frac{\partial x}{\partial u} \cdot \frac{\partial y}{\partial v}}. \tag{3}$$

Computing the Score of a Surface Match

- Minimize $\bar{V}(\mathrm{r})$ in the interval $(0, R)$. Here R is the maximum radius common to both the surfaces.



- 1-D minimization using exhaustive search method.
  If $\bar{V}(r_1) \geq \bar{V}(r_2) \leq \bar{V}(r_3)$, the minimum point lies in (r_1, r_3), **Terminate**.

Proposed Algorithm for Computing the Score

Step 1 : Input (u, v) parameters of points at some interval on each boundary of both the surface patches. Compute points on the boundaries by Eq. 1. Compute radii corresponding to these points by formula $r = \sqrt{x^2 + y^2}$. Set minimum $r = R$.

Step 2 : Set $V_1 = 0$, $V_2 = 0$, $V_3 = 0$, $\Delta V = 0$

Step 3 : Set $\Delta r = 0.01$, $\Delta\theta = 0.1$, $r = 0.001$.

Step 4 : Set $\theta = 0$.

Step 5 : Compute $x_0 = r\cos\theta, y_0 = r\sin\theta$.

Step 6 : Input (u, v) parameters of the point of contact on the first surface patch. This is the initial guess to Newton-Raphson method.

Algorithm for Computing the Score contd..

Step 7 : Compute corrections. Add corrections to the previous guess. Compute a point on the first surface Q_1.

Step 8 : compute $error = \sqrt{(x - x_0)^2 + (y - y_0)^2}$. Here x, y are the x, y coordinates of the point Q_1.

Step 9 : If $error < \epsilon$, **Terminate**; set $z_1 = Q_{1z}$ where Q_{1z} is the z coordinate of point Q_1. Else go to step 7.

Step 10 : Input (u, v) parameters of the point of contact on the second surface and repreat steps 7 to 9 for the second surface. Set $z_2 = Q_{2z}$ where Q_{2z} is the z coordinate of point Q_2.

Step 11 : Compute $\delta h = z_2 - z_1$. Compute volume mismatch ΔV by Eq. 1. Update $\Delta V_{new} = \Delta V_{old} + \Delta V$.

Algorithm for Computing the Score contd..

Step 12 : Is $\theta = 2\pi$? If no, set $\theta = \theta + \Delta\theta$, go to step 5 and if yes, compute $\bar{V}$.

Step 13 : Set $V_1 = V_2$, $V_2 = V_3$, $V_3 = \bar{V}$.

Step 14 : If $V_1 \geq V_2 \leq V_3$, set $\bar{V}_{min} = V_2$, **Terminate**;
Else check Is $r \leq R$?, if yes, set $r = r + \Delta r$, go to step 4. If no, **Terminate**.

Simulation Result

Input point for the first surface: $u = 0.2$, $v = 0.2$

Input point for the second surface: $u = 0.2$, $v = 0.3$

Normalized volume mismatch $(\bar{V}) = 3.466E - 03$

Optimization of Shape Matching

- Statement of problem:
  Given two surface patches in any arbitrary orientation, obtain the orientation which minimizes the normalized volume mismatch (the scoring function) between them when these surface patches are matched together.

- Solution Strategy:
  Due to a large number of orientations in which two surface patches can be tried, a two-stage optimization procedure is used for an efficient search of the optimum solution.
  - Curvature difference minimization
  - Scoring function minimization

Curvature Difference Minimization

- Problem: Given two surface patches, find a region on each of them such that the difference of principal curvatures of these regions is minimum.
- Algorithm:

  Step 1: Choose five points randomly in five different regions of both the surface patches.

  Step 2: Compute principal curvatures of the surfaces at these points.

  Step 3: Check for surface complementarity, i.e., $\kappa_{max1} \times \kappa_{max2} > 0$ and $\kappa_{min1} \times \kappa_{min2} > 0$.

  .

Curvature Difference Minimization contd..

Step 4: If surfaces are not complementary, **terminate**. If surfaces are complementary, Compute function $f = (\kappa_{max1} - (-\kappa_{max2}))^2 + (\kappa_{min1} - (-\kappa_{min2}))^2$ for five pairs of points, each pair having a point from both the surface patches.

Step 5: Find the pair of points with minimum f. A point in this pair represents a region on the corresponding surface patch with minimum difference in curvatures.

Scoring Function Minimization

- Problem: Given a vector of (u, v) parameters of a point from each of the two matching surface patches, say, $\mathbf{x} = (u_1, v_1, u_2, v_2)$, as an initial guess, find $\mathbf{x}$ such that the scoring function, $\bar{V}$ is minimum.
- BFGS algorithm is used for scoring function minimization. Starting with an identity matrix, $\mathbf{B}_0 = \mathbf{I}$, the matrix $\mathbf{B}$ is modified at every iteration to finally take the form of the inverse of the Hessian matrix at the minimum point. A search direction is chosen as

$$\mathbf{s}_k = -\mathbf{B}_k \mathbf{g}_k$$

BFGS Update Formula

The following update formula suggested by Broyden, Fletcher, Goldfarb, and Shanno (1970) is used to modify the matrix $\mathbf{B}$

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \left[1 + \frac{\partial \mathbf{g}^T \mathbf{B} \partial \mathbf{g}}{\partial \mathbf{x}^T \partial \mathbf{g}}\right]_k \left[\frac{\partial \mathbf{x} \partial \mathbf{x}^T}{\partial \mathbf{x}^T \partial \mathbf{g}}\right]_k - \left[\frac{\partial \mathbf{x} \partial \mathbf{g}^T \mathbf{B} + \mathbf{B} \partial \mathbf{g} \partial \mathbf{x}^T}{\partial \mathbf{x}^T \partial \mathbf{g}}\right]_k$$

where,

$$\begin{aligned} \partial \mathbf{x}_k &= \text{change in the vector of variables,} \\ \partial \mathbf{g}_k &= \text{change in gradient vector .} \end{aligned}$$

BFGS Algorithm

Step 1 : Input initial guess $\mathbf{x}_0$, $\mathbf{B}_0$ and termination parameters $\epsilon_1, \epsilon_2, \epsilon_3$.

Step 2 : For any k, set $\mathbf{s}_k$.

Step 3 : Compute step size α_k such that $f(\mathbf{x}_k + \alpha_k \mathbf{s}_k)$ is minimum with termination parameter ϵ_1. Set $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{s}_k$.

Step 4 : Compute the update matrix $\hat{\mathbf{B}}_k$.

Step 5 : Set $\mathbf{B}_{k+1} = \mathbf{B}_k + \hat{\mathbf{B}}_k$.

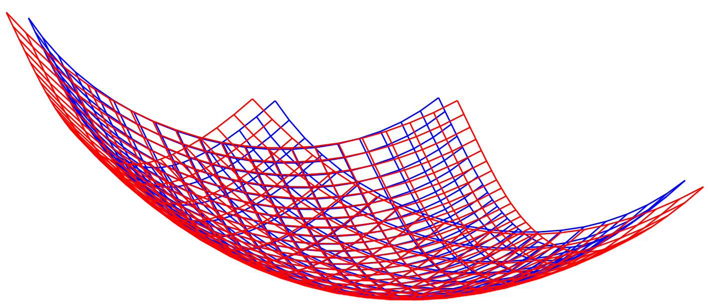
Step 6 : Is $\frac{\|(\mathbf{x}_{k+1} - \mathbf{x}_k)\|}{\|\mathbf{x}_k\|} \leq \epsilon_2$ or $\|\mathbf{g}_{k+1}\| \leq \epsilon_3$? If yes, **Terminate**; Else set $k = k + 1$ and go to step 2.

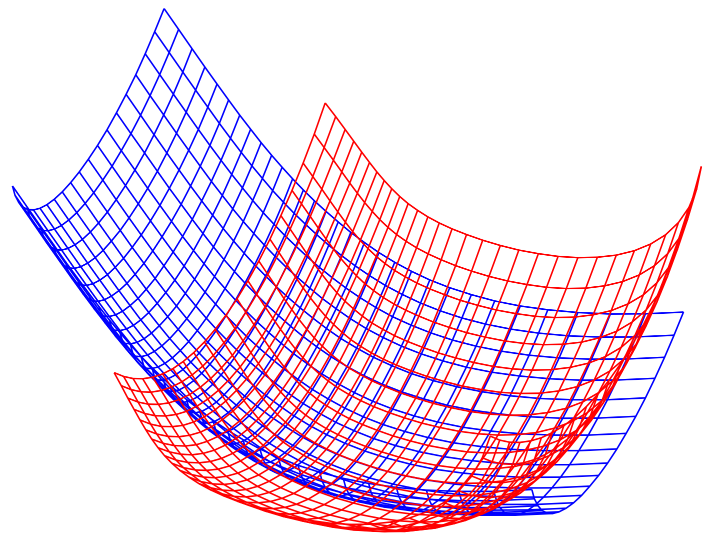
Simulation Results

Table 1: Optimum scores of matching the surface patches with each other

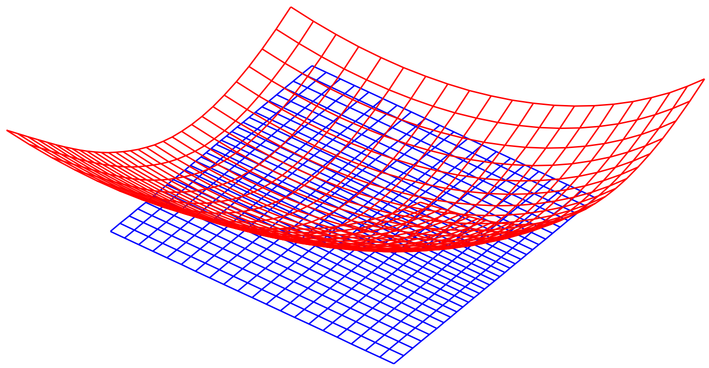
Serial no.	Patch of 1st molecule	Patch of 2nd molecule	Optimum score
1	1	1	9.598E-07
2	1	6	7.880E-04
3	3	1	2.389E-03
4	3	2	2.512E-317
5	4	4	1.786E-04

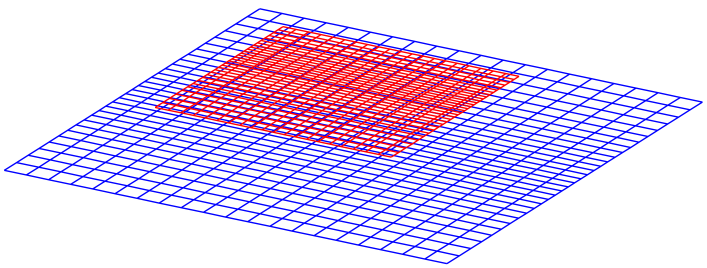
Optimum Match of Patches 1-1 & 1-6



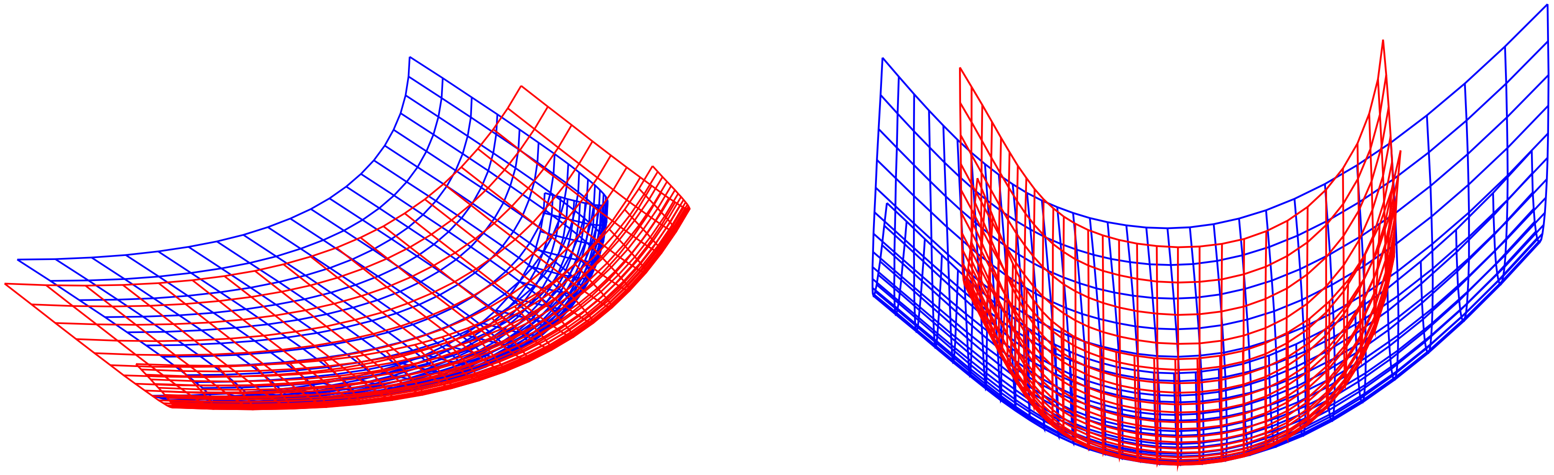


Optimum Match of Patches 3-1 & 3-2

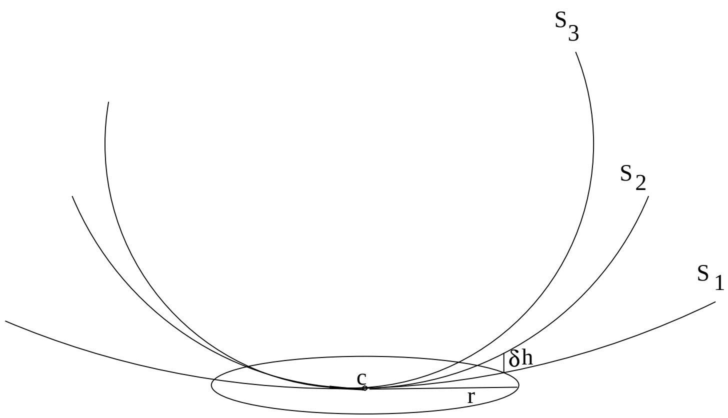


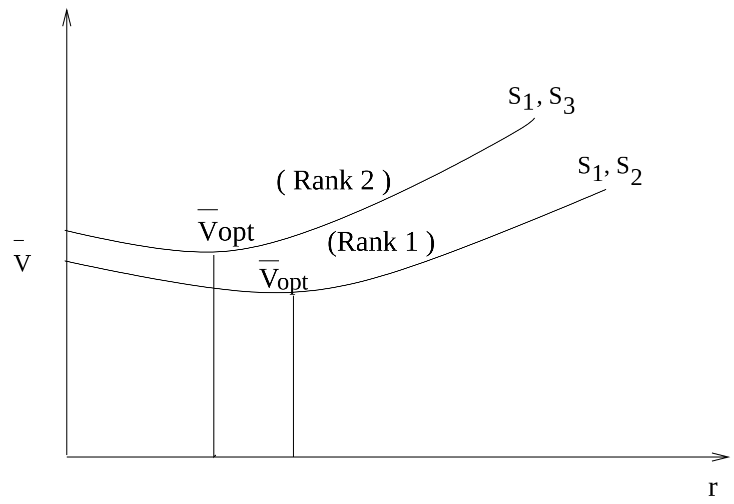


Optimum Match of Patches 4-4 & 7-6



Ranking of Surface Patches



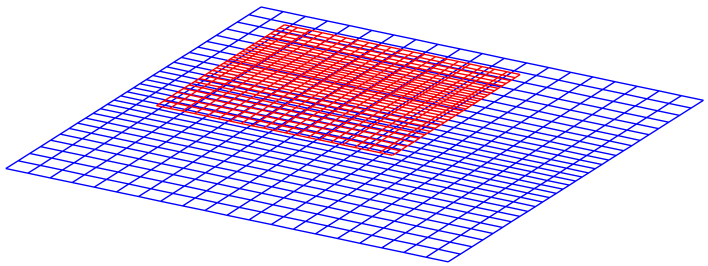


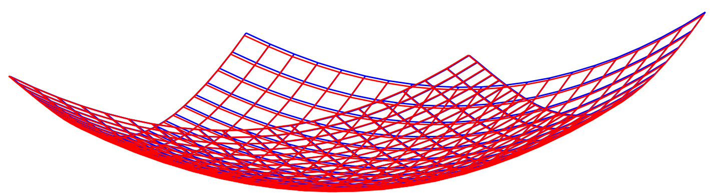
Simulation Results

Table 2: Ranking of matches based on the optimum score

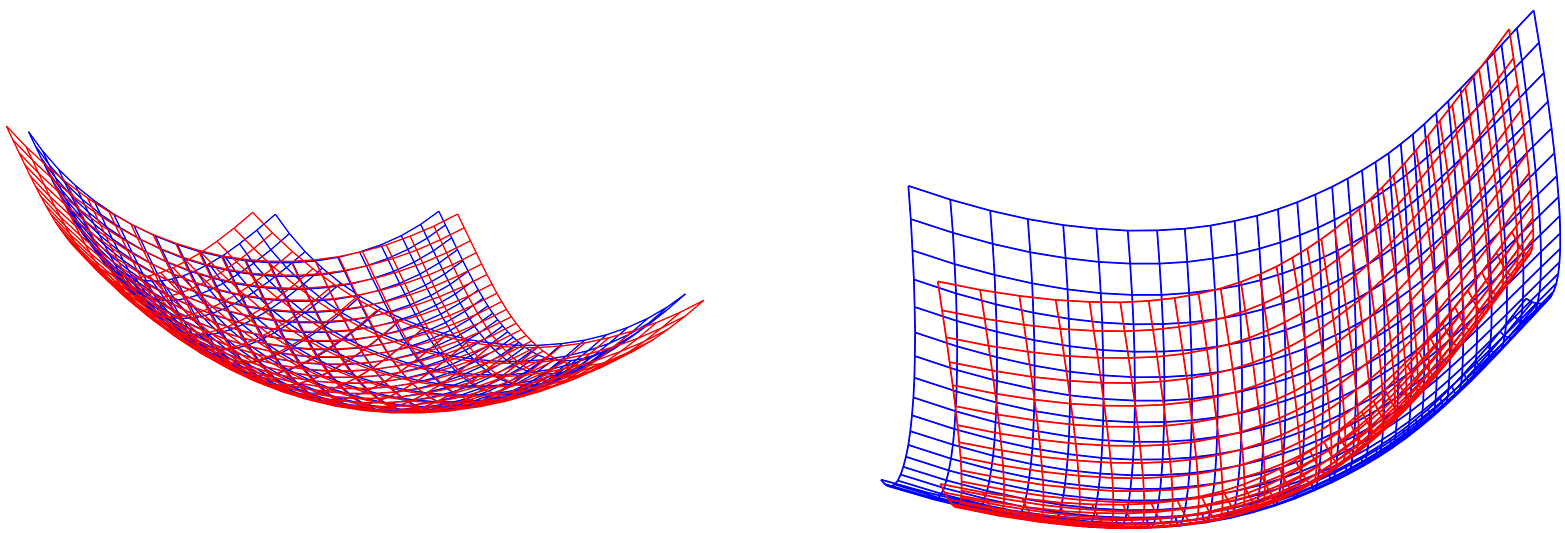
Rank	Patch of 1st molecule	Patch of 2nd molecule	Optimum score
1	3	2	2.512E-317
2	5	7	3.418E-07
3	1	1	9.598E-07
4	7	7	4.337E-06
5	5	6	1.172E-05

Best Five Surface Matches (Rank 1 & 2)

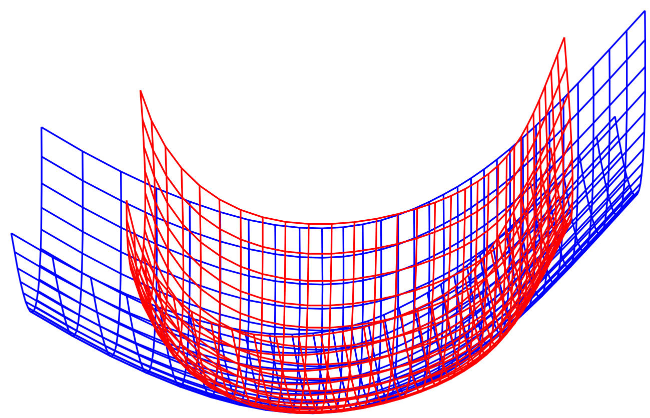




(Rank 3 & 4)



Rank 5



Concluding Remarks

- The algorithm proposed is useful for shape matching of any two objects, in general, and for rigid body molecular docking, in particular.
- The results show that B-spline representation, due to its inherent flexibility, can effectively be used for reconstruction of different types of surface patches.
- The patch-by-patch matching strategy alongwith shape complementarity criterion works well and is suitable for docking.

Concluding Remarks contd..

- Normalized volume mismatch effectively discriminates between better, good and bad matching pairs of surface patches.
- For optimization in shape matching a two-stage optimization procedure is used which speeds up the search through solution space.
- Search of an optimum solution and its rank depends upon a number of factors, like shape complementarity, type of interface, molecular size etc.

Scope for Future Work

- The algorithm is ready now for testing on actual protein surface point data. For this purpose, protein surface data needs to be parameterized on 2-D rectangular grid using a suitable algorithm.
- The scoring function used in the algorithm is purely geometric. The results can further be improved by incorporating physico-chemical parameters, like electrostatics, desolvation, hydrophobicity, atom/residue pairs potentials, etc.
- This output can be used as input for detailed energy optimization algorithms.

Scope for Future Work contd..

- The algorithm is developed considering protein molecules as rigid bodies. Protein side chain movements can be handled at different levels of algorithm.
- Alternatively, side chain movements can be handled by application of a post-processing refinement procedure or by using multiple side-chain conformations at the matching stage.
- Docking algorithm can be improved by adding constraints for steric clashes and symmetry.

Selective References

Kuntz I. (1992). Structure based strategies for drug design and discovery. *Science 257*, 1078-1082.

Abagyan R. and Totrov M. (2001). High throughput docking for lead generation. *Curr Opin Chem Biol 5*, 375-382.

Halperin et al. (2002). Principles of docking: An overview of search algorithms and a guide to scoring functions. *PROTEINS: Structure, Function and Genetics 47*, 409-433.

Kuntz et al. (1982). A geometric approach to macromolecule-ligand interactions. *J. of Mol. Biol. 161*, 269-288.

Connolly M. (1986). Shape complementarity at the hemoglobin $\alpha_1\beta_1$ subunit interface. *Biopolymers 25*, 1229-1247.

Selective References contd..

Norel et al. (1999). Examination of shape complementarity in docking of unbound proteins. *Proteins 36*, 307-317.

Bajaj et al. (1997). NURB based B-rep models from macromolecules and their properties. *Proc fourth symposium on solid modeling and applications, New York*, 217-228.

Fischer et al. (1995). A geometry-based suit of molecular docking processes. *J. of Mol. Biol. 248*, 459-477.

Goldman B. B. and Wipke W. T. (2000). QSD quadratic shape descriptors. *Proteins 38*, 79-94.

Gardiner et al. (2001). Protein docking using GA. *Proteins 44*.

Lin et al. (1994). Molecular surface representation by sparse critical points. *Proteins 18*, 94-101.

Thank You!