Molecular Docking Using Shape Descriptors

BRIAN K. SHOICHET, DALE L. BODIAN* and IRWIN D. KUNTZ
*Department of Pharmaceutical Chemistry, University of California, San Francisco, San Francisco, California, 94143-0446, and *Department of Biochemistry and Biophysics, University of California, San Francisco, San Francisco, California, 94143-0448*

Reprinted from Journal of Computational Chemistry, Vol. 13, No. 3, 1992

Molecular Docking Using Shape Descriptors

Brian K. Shoichet, Dale L. Bodian* and Irwin D. Kuntz†

*Department of Pharmaceutical Chemistry, University of California, San Francisco, San Francisco, California, 94143-0446, and *Department of Biochemistry and Biophysics, University of California, San Francisco, San Francisco, California, 94143-0448*

Received 26 July 1991; accepted 30 September 1991

Molecular docking explores the binding modes of two interacting molecules. The technique is increasingly popular for studying protein–ligand interactions and for drug design. A fundamental problem with molecular docking is that orientation space is very large and grows combinatorially with the number of degrees of freedom of the interacting molecules. Here, we describe and evaluate algorithms that improve the efficiency and accuracy of a shape-based docking method. We use molecular organization and sampling techniques to remove the exponential time dependence on molecular size in docking calculations. The new techniques allow us to study systems that were prohibitively large for the original method. The new algorithms are tested in 10 different protein–ligand systems, including 7 systems where the ligand is itself a protein. In all cases, the new algorithms successfully reproduce the experimentally determined configurations of the ligand in the protein.

INTRODUCTION

Molecular docking fits molecules together in favorable configurations using their topographic features. Practically, docking has been an important technique for the modeling of protein–ligand interactions and has been used in studies of the structural basis of biological function[1,2] and drug design.[3,4] Theoretically, the approach is a relatively tractable instance of the general problem of combinatorial optimization, a focus of much work in recent decades.[5]

One of the first practical suggestions for docking came from Crick,[6] who suggested that complementarity in helical coiled-coils could be modeled as knobs fitting into holes. More recently, workers have used both geometric[7–11] and energy-based methods[2,12–14] to search for fruitful binding modes of ligands in receptors. The geometric methods have focused on matching descriptors of topographical features to generate favorable configurations, while the energy methods have used potential energy functions to guide their search of orientation space.

Docking is computationally difficult because there are many ways of putting the two molecules together and the number of possibilities that must be sampled grows exponentially with the size of the component molecules. The orientation space of two biomolecules, especially when one or more of them is a protein, is so large as to make exhaustive methods prohibitive.[9] This difficulty reflects the many interfaces and multiple minima presented by the surface of a macromolecule; for descriptor-based methods, the docking problem is nondetermined in polynomial time (NP-complete).[15]

We previously developed a rigid body docking method that uses molecular descriptors (DOCK program).[7,8] DOCK could regenerate experimentally determined configurations in several ligand–protein complexes. Like all descriptor-based methods, however, the search time of the algorithm scaled poorly as the number of features describing the molecules increased. This time dependence made docking of macromolecular complexes, for instance, unfeasible. Also, some of the heuristics used in DOCK to help reduce the number of possible matches made predicting the performance of the algorithm difficult. Lastly, the extent of a search was hard to control.

In this article, we discuss new algorithms that make our docking procedure faster and allow it to handle protein–protein systems, which had previously been prohibitively large for our method; we call the new program DOCK2. We describe modifications particular to our implementation of a docking program, as well as changes in algorithmic approach that address general features of the docking problem. At the general level, we address the strong time dependence of the algorithms by a "divide-and-conquer" procedure that separates macromolecules into independent geometric regions that are individually considered as possible interfaces. This modification dramatically improves the way the docking problem scales with the size of the system being docked. We illustrate the improvement by

†Author to whom all correspondence should be addressed.

CCC 0192-8651/92/030380-18$04.00

Table I(a). The four test complexes and structures used in the most extensive testing of the algorithms, including focusing, clustering, scoring, and the different graph-generation methods.

Receptor[a]	Ligand	Number of atoms in receptor[b]	Number of atoms in ligand
Ribonuclease (6rsa)[17]	Uridine vanadate (6rsa)	951[c]	20
Dihydrofolate reductase (3dfr)[18]	Methotrexate (3dfr)	1298	33
Lactate dehydrogenase (5ldh)[19]	NAD-lactate (5ldh)	2560	51
Trypsin (2ptc)[20]	PTI (2ptc)	1595	423

[a]All structures taken from the Protein Data Bank[16] have their reference code in parentheses.
[b]The number of atoms are for the structures actually used in the docking runs. These may differ slightly from those in the pdb files in that atoms that had no density, as recorded in the pdb files, were not used in the docking runs.
[c]Though 6rsa is a neutron structure and contains hydrogen/deuterium coordinates, only heavy atoms were used in the docking runs.

Table I(b). Test complexes and structures used in the protein–protein docking tests DOCK2.

Receptor	Ligand	Number of atoms in receptor	Number of atoms in ligand
Trypsin[25] (2ptn)	PTI[20] (4pti)	1564	449
Subtilisin[22] (2sni)	Chymotrypsin inhibitor[22] (2sni)	1938	513
Subtilisin[26] (1sbc)	Chymotrypsin inhibitor[27] (2ci2)	1920	521
Chymotrypsin[23] (1cho)	Ovomucoid 3rd domain[23] (1cho)	1751	400
Chymotrypsin[28] (5cha)	Ovomucoid 3rd domain[28] (2ovo)	1736	418
Thymidylate synthase monomer[24]	Thymidylate synthase monomer[24]	2143	2143

docking seven different pairs of proteins. Also at the general level, we outline a technique to increase automatically the number of possible configurations generated in regions of likely complementarity. Ideally, this should allow for more efficient sampling of orientation space, moving from low-density sampling in "poor" regions to higher degrees of sampling in regions that have produced favorable configurations. We also take up issues specific to our program. The new program is more systematic in its searches of orientation space, and also more easily controlled in depth of search by the user. We consider three different ways of selecting features for matching and compare the success of each approach at reproducing experimental configurations. Finally, we describe a lattice-based method for evaluating the goodness of fit of the docked complexes, which significantly reduces run times. We test the new algorithms extensively in four crystallographically determined protein–ligand complexes (all structures are taken from the Protein Data Bank[16]): ribonuclease/uridine vanadate,[17] dihydrofolate reductase/methotrexate,[18] lactate dehydrogenase/NAD*-lactate,[19] and trypsin/PTI[20] [Table Ia, Figs. 1–3[21] (see color)]. We show that the methods can be used to regenerate the crystallographic configurations of six other complexes [Table Ib, Table II], where the ligand as well as the receptor is a protein. In their bound (as they occur in the crystal complex) conformations, we dock subtilisin with chymotrypsin inhibitor,[22] chymotrypsin with ovomucoid third domain,[23] and thymidylate synthase monomer with thymidylate synthase monomer[24] to regenerate the dimer. In their unbound conformations (as they occur in isolation of their cognate ligand or receptors), we dock trypsin[25] with PTI,[20] subtilisin[26] with chymotrypsin inhibitor,[27] and chymotrypsin[28] with ovomucoid third domain.[29]

*Abbreviations used: NAD (nicotinamide adenine dinucleotide); rmsd (root mean square deviation); PTI (pancreatic trypsin inhibitor); DHFR (dihydrofolate reductase).

THE DOCKING PROBLEM

The underlying notions in descriptor-based docking have their antecedents in the "lock and key" ideas of Ehrlich.[30] The computational problem is to describe the features that define the shape of the "lock" and "key" and then map the two sets of features together in favorable ways. There are many ways of describing molecules for this purpose.[9,11,31,32] We use spheres that are locally complementary to a molecular surface.[7,33] However the features are described, the next task is choosing which of them to use for matching the two molecules together. This brings up a fundamental difficulty.

Matching features (descriptors) involves selecting a set of some number of points from a larger collection of possibilities. The number of possible sets of features depends combinatorially on the number of features in each set (n), and the total number describing each molecule:

$$\text{Number of sets} = {}^{n}C_{N_r} \times {}^{n}P_{N_l} = {}^{n}P_{N_r} \times {}^{n}C_{N_l}, \quad (1)$$

where N_r is the total number of receptor features and N_l is the total number of ligand features. C and

Table II. Protein–protein docking results.[37]

Receptor[a]	Ligand	Type[b]	Best docked (to crystal structure, rmsd Å)[c]	Total orientations evaluated in docking	Run time (hr:min)
Trypsin (2ptc)[20]	PTI (2ptc)	Bound	0.29	360,366	1:04
Trypsin (2ptn)[25]	PTI (2ptn)[20]	Free	0.52	9,976,471	27:11
Chymotrypsin (1cho)[23]	Ovomucoid 3rd domain (1cho)	Bound	0.72	1,650,604	4:19
Chymotrypsin (5cha)[28]	Ovomucoid 3rd domain (2ovo)[29]	Free	0.82	2,117,929	5:44
Subtilisin (2sni)[22]	Chymotrypsin inhibitor (2sni)	Bound	0.14	1,511,411	5:30
Subtilisin (1sbc)[26]	Chymotrypsin inhibitor (2ci2)[27]	Free	0.64	8,615,720	20:23
Thymidylate synthase monomer[24]	Thymidylate synthase monomer	Bound	0.33	1,886,885	14:08

[a]PDB[16] reference numbers in parentheses.
[b]Two types of calculations were performed, using the bound (from the crystal complex) or the free (from the uncomplexed crystal structures) conformations of the molecules.
[c]rmsd's measured to the crystal complex for bound docking runs and to best fits to the crystal complex in the free conformer runs.[35]

P represent combinations and permutations. When N_r, $N_l \gg n$, (1) can be rewritten:

$$\text{Number of sets} = (N_r)^n \times (N_l)^n. \qquad (2)$$

Therefore, as the number of features in a set rises the number of sets to be considered rises exponentially, as would the computation time of any method that attempts to dock molecules by looking at all possible sets.

Fortunately, for docking it is not necessary to consider sets above a certain size. Four nonplanar points from each set—four features from each molecule—uniquely define a configuration involving two molecules of any shape. The computation of a docking problem thus becomes bounded, minimally but sufficiently, by $(N_r)^4 \times (N_l)^4$.

This still makes for long calculation times when N_r and N_l are large, as is the case in macromolecular docking. One can further improve matters by pruning sets as they are being constructed, a procedure described in detail in the following section. Even with pruning, however, the number of configurations that might plausibly be looked at for two macromolecules is still very large, and this number grows quickly as the size of the ligand and receptor increase. We return to this problem in the next section.

Having described the molecules, chosen sets of features, and mapped the one onto the other, it only remains to evaluate the resulting configurations for the goodness of fit between the ligand and the receptor. There are as many ways of doing this as there are docking programs; most methods use simplified potential functions or some version of shape complementarity. We describe the details of our implementation below.

METHODS

We summarize the method and then take up each point in greater detail in the following paragraphs. Geometric descriptions (spheres or atoms) of local bumps and clefts on ligand and receptor surfaces guide the search of orientation space, the goal being to find orientations that map the bumps of one into the clefts of the other. We look for sets of spheres from the first molecule that have the same internal distances, within a certain tolerance, between their centers as do sets of spheres from the second molecule. We will refer to sets that pass this distance criterion as "matches." Matches are used to define rotation/translation matrixes that map the second molecule onto the first.[34] Configurations of the ligand in the receptor depend, therefore, on the locations of the sphere sets on the surfaces of the respective molecules. An orientation, once found, is subjected to a fast preliminary evaluation of complementarity based on a simple examination of atomic contacts between the receptor and the ligand (scoring). Orientations of the ligand that place it in regions of space occupied by the receptor are discarded. The configurations that pass the excluded volume filter and have enough "good" contacts are saved for further evaluation.

Molecular Description: Spheres

We describe a receptor geometrically using spheres locally complementary to grooves and ridges in its molecular surface[7,33] (Fig. 4, see color). The spheres fill the empty volume of a site, generating its negative image. The centers of these spheres may be thought of as pseudoatoms; they are used as an irregular grid for mapping the ligand into the binding site. Spheres are generated analytically to touch the molecular surface at two points, have their centers along the surface normal to one of the points, and are placed so they do not intersect the surface. The spheres are of different sizes and typically overlap one another within a given pocket in the protein. A collection of overlapping spheres defines a cluster. The molecular surface of a protein will typically have tens of clus-

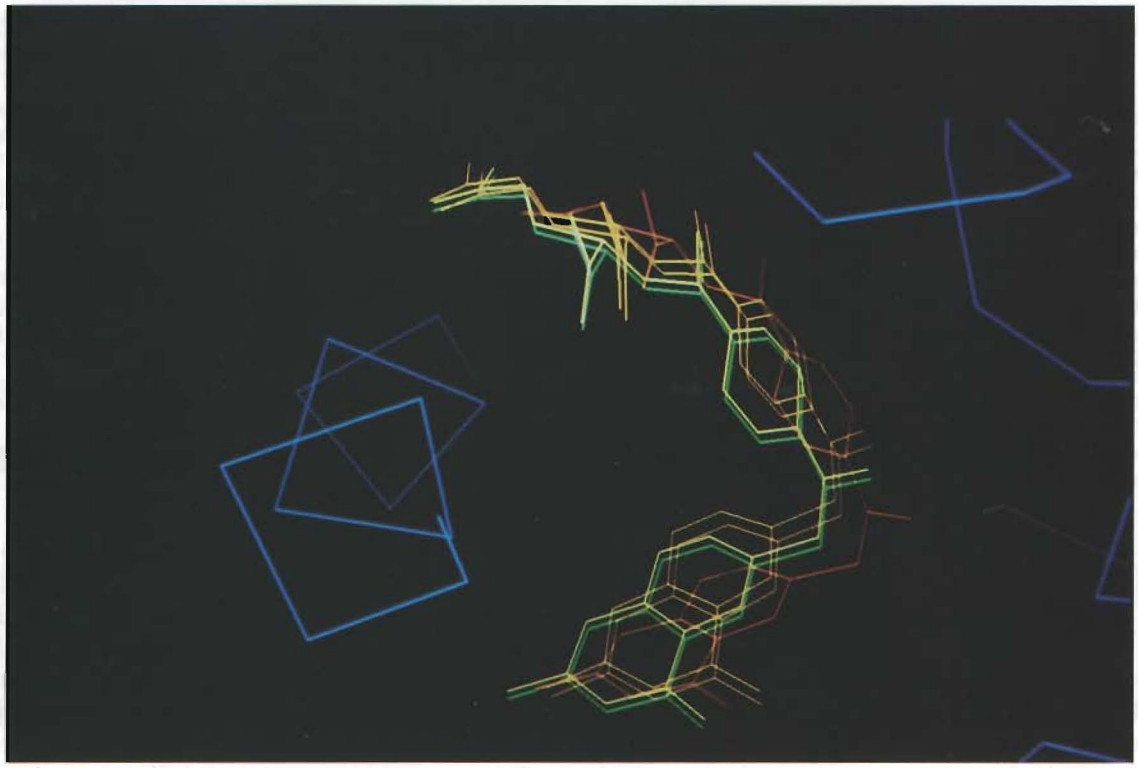


Figure 1. Several low-rmsd dockings of methotrexate in dihydrofolate reductase.[18] Crystallographic configuration in green, docked orientations in yellow, amber, and red, in increasing rmsd, respectively. Protein in blue. Figures 1–4 and 11 were made with the MidasPlus graphics program.[21]

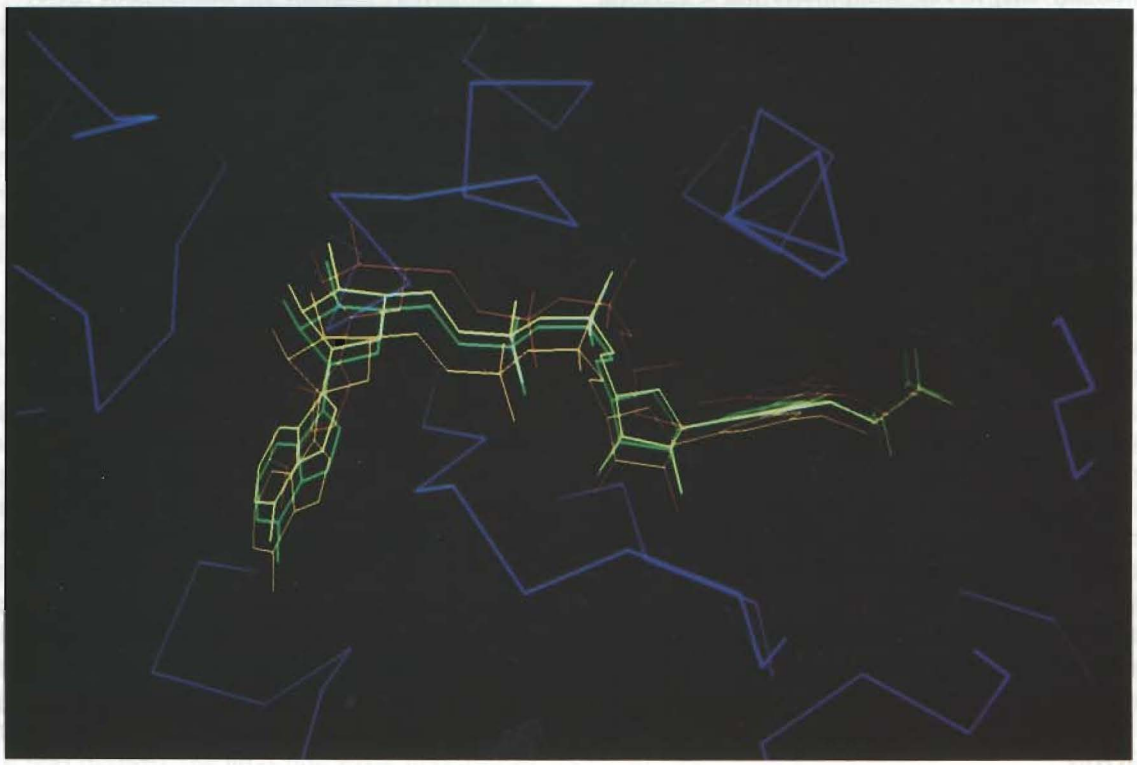


Figure 2. Several low-rmsd dockings of NAD-lactate in lactate dehydrogenase.[19] Crystallographic configuration in green, docked orientations in yellow, amber, and red, in increasing rmsd, respectively. Protein in blue.

ters, each of which describes a potentially interesting site of interaction. The radius of a sphere reflects the concavity of a local region of the molecular surface. The larger the sphere radius, the larger and shallower the pocket that sphere describes. Macromolecular ligands are similarly described, except that the spheres are placed *within* the molecular surface and are complementary to local ridges rather than the grooves. For smaller ligands such as methotrexate, the atom centers are used rather than spheres.[3] Other points can be added to the receptor or the ligand descriptions without loss of generality. These include the center of mass, centers of rings, centers of molecular attraction, bound waters, and so forth.

Molecular Organization: Divide and Conquer

Macromolecules have many descriptors, which leads to a great number of possible dockings. PTI, for example, is described by 292 spheres in one molecule-spanning cluster, about 10 times more descriptors than in a typical drug-type inhibitor such as methotrexate. Given the third- to fourth-power dependence of run time on the number of descriptors,[15] the computation time for docking macromolecular ligands could be as much as 10^4 times longer than for small molecule ligands. In complexes of determined structure, however, the ligand is much larger than the binding site of its receptor, which suggests that most of the ligand's surface will not be involved in any given interface with the receptor. It is thus worthwhile to organize the macromolecules into geometrically distinct subsections, each of which can be matched independently. Ideally, each subsection would describe one potential interface region of the molecule. We call this procedure "subclustering."

The subclustering program (CLUSTER) begins with a relatively large group of spheres from the sphere-generation program (SPHGEN).[7] Each sphere overlaps at least one other sphere in the single-linkage cluster[35] and none outside it. Large spheres span and connect local regions and often have many more connections than do small spheres. We reduce the number of spheres in individual clusters by eliminating the linkages arising from spheres larger than a user-set threshold (Fig. 5). This segregates the spheres into smaller clusters as the threshold is reduced. The procedure is analogous to using articulation vertices to split connected graphs.[36] During this process, the total number of clusters increases. Since we treat each cluster as a potential interface site, the greater number of sites increases the number of possible orientations. This effect is, however, small compared to the combinatorial advantage of restricting the total number of spheres in each site. The ratio of the number of possible matching spheres sets after and before subclustering is:

$$\text{Configs}_{\text{subclustered}}/\text{Configs}_{\text{unclustered}} = \sum_{r\text{clus}} \sum_{l\text{clus}} (N_r'/N_r)^n \times (N_l'/N_l)^n \quad (3)$$

N_r' and N_l' are the number of descriptors in the subclustered groups being matched in the receptor and ligand, respectively, for N_r' and $N_l' \geqslant n$. $r$clus and $l$clus are the numbers of new, subclustered sphere sets for the receptor and the ligand. In this manner, subclustering significantly decreases the number of possible matches necessary to consider. For example, if subclustering reduces the number of spheres in a sphere to ¼ its original size, while the number of total clusters needed to describe the molecule rises from 1 to 4, then the ratio in (3) will be $4^{-(n-1)}$, or 1/64 when n is 4. Of course, further reduction of the search may be possible if attention can be focused on one of the subclusters, such as the active site.

Matching: The Bipartite Graph

We dock ligands into receptor sites by matching subsets of ligand internal distances onto subsets of receptor sphere internal distances. Most of the possible combinations of ligand and receptor descriptors will not lead to successful dockings. It is therefore sensible to prune the matching search tree as soon as possible.

Docking may be posed as a graph theoretical problem.[15] If a ligand has N_l descriptors and a receptor has N_r descriptors, then the number of nodes in the docking graph D is $N_l \times N_r$. An edge exists between two nodes composed of descriptors $(N_l)_i,(N_r)_i$ and $(N_l)_j,(N_r)_j$ (where i and j are ligand or receptor descriptors), from the ligand and the receptor, when the distance $N_l(i,j)$ is the same as $N_r(i,j)$, within some tolerance. A minimal match between a ligand and a receptor occurs when there is a subgraph of D that is completely connected by edges and that has at least four nodes (and therefore six edges) in it. Four nodes must be specified to determine a rotation-translation matrix that preserves chiral information. It is physically impossible to match, simultaneously, most ligand-receptor features. Another way of saying this is that D is sparsely connected, which leads to our method for pruning the search of orientation space.

As in the original program,[7] we use a distance-matching algorithm that calculates whether the receptor and ligand descriptors share the same pairwise distances, within some tolerance level, in a build-up procedure that evaluates the growing graph as each node is added (Fig. 6). A graph that fails the distance check at some number of nodes M, i.e., which is not completely connected due to the addition of the Mth node, will also fail at all numbers greater than M; therefore, we can prune the search

at M. Such pruning dramatically reduces the number of ligand-receptor nodes necessary to consider. We further reduce the search space by biasing the search to long edges representing large internal distances. This is a heuristic that weights long-range information more heavily than local information.

To control the search of orientation space, we organize the receptor spheres based on the internal distances between pairs of sphere centers. Starting with each sphere center, all other centers in the cluster are sorted into "bins" based on their distances to the starting sphere center (Fig. 7). All distances in a certain range will be placed in the same bin. The bins are of adjustable resolution—the larger the distance interval for a bin the more points it will typically contain and the fewer the number of bins overall. DOCK2 can allow overlaps between sequential bins to diminish the effect of discrete distance ranges. Bins are constructed for each receptor sphere. Ligand bins are constructed using the same procedure. The ligand and receptor bins for each pair of starting points are matched based on the distance ranges of the points within them; only those points in bins with similar ranges will be used to generate a graph. The first n-1 receptor-ligand bins (those bins representing the longest distances) that match are chosen for graph generation. Features from a given bin are tried at only one stage in the graph generation; thus, the features from the second bin will always provide the third node in a graph, the original (bin-defining) pair of points defining the first pair of nodes. All centers are ultimately tried as starting points and all centers within a "longest-distance" bin are tried in the generation of the matching graph, unless the graph has been pruned before *any* of the centers in the bin have been tried. Because of the longest-distance heuristic, not all bins are tried; the method is not exhaustive. With the caveat of this heuristic, however, the method is path independent. The number of points in each bin determines how many matches will be attempted. In general, the larger the bins the larger the number of orientations generated. The breadth of search is under user control.

Three Graph Construction Methods

We describe three different methods for choosing which internal distances to compare in the construction of the bipartite graphs, all of which use bin matching. All three algorithms begin by pairing a ligand descriptor with a receptor descriptor. All $N_l \times N_r$ starting pairs are tried.

Fan Algorithm

Descriptors from protein and ligand are chosen based on their distance from an initial starting descriptor in each molecule. The starting point in this method therefore implicitly defines which region of the molecule will be looked at for matching [Fig. 8(a)].

An initial pair of points (a node from D) is picked from among the set of spheres or atoms describing the molecules. The bins for each molecule are independently generated based on the distances of the remaining points from the initial point. The Fan method uses the first $n - 1$ bins, those with the longest distances from the initial point, that have distance ranges that match the second molecule's bins. These bins provide the molecular features, spheres or atoms, used for bipartite graph generation in the matching.

Cat's Cradle Algorithm

Descriptors at a given level of the bipartite graph generation are chosen based on their distance to the descriptor successfully used at the *previous* level of graph generation. As always, we use the longest-distance heuristic. The starting point in this method is therefore less important than in the Fan procedure, and since the longest interpoint distances in the molecule will tend to be found regardless of starting point more of the same nodes will be repeatedly used [Fig. 8(b)].

An initial pair is picked and the bins are generated as in the Fan algorithm. Unlike Fan, only one pair of bins are selected, providing for only the second node in the graph. The next bins, providing the third node possibilities in the graph, are created based on distances from each sphere or atom selected as second nodes. Spheres/atoms in these second-generation bins are biased for longest distances. Multiple third bins are similarly created based on distances from spheres or atoms selected from the second bin, and so on.

Center of Mass Algorithm

This method resembles the Fan method except rather than choosing atoms or spheres based on their distances to the starting pair of points, centers are chosen based on their distances to the center of mass. Except for the starting centers used for the first node, therefore, the centers used in matching will always be the same [Fig. 8(c)].

An initial pair is chosen. Bins are generated based on distances from the center of mass of the molecule. The algorithm then proceeds as in the Fan algorithm: The bins used are the first $n - 1$ bins from the center of mass that have distance ranges that match a set of bins from the second molecule.

Scoring on a Lattice

We score possible orientations of the ligand in the receptor based on atomic contacts between ligand

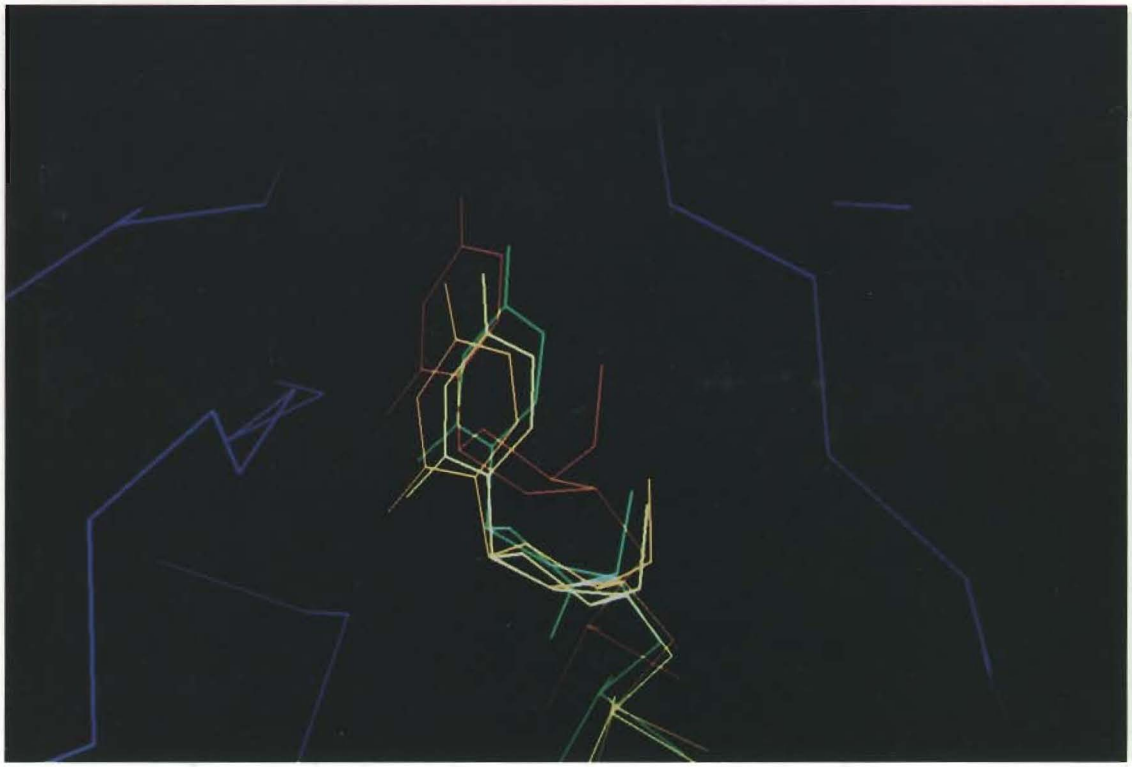


Figure 3. Several low-rmsd dockings of uridine vanadate in ribonuclease.[17] Crystallographic configuration in green, docked orientations in yellow, amber, and red, in increasing rmsd, respectively. Protein in blue.

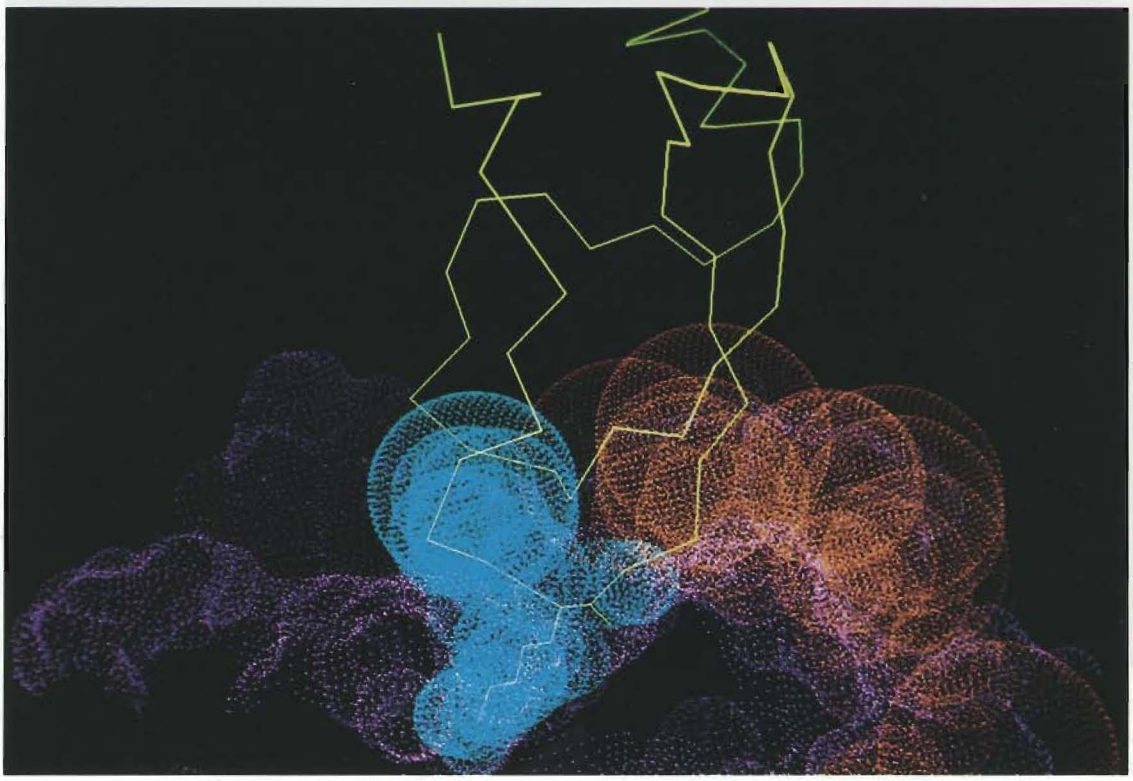
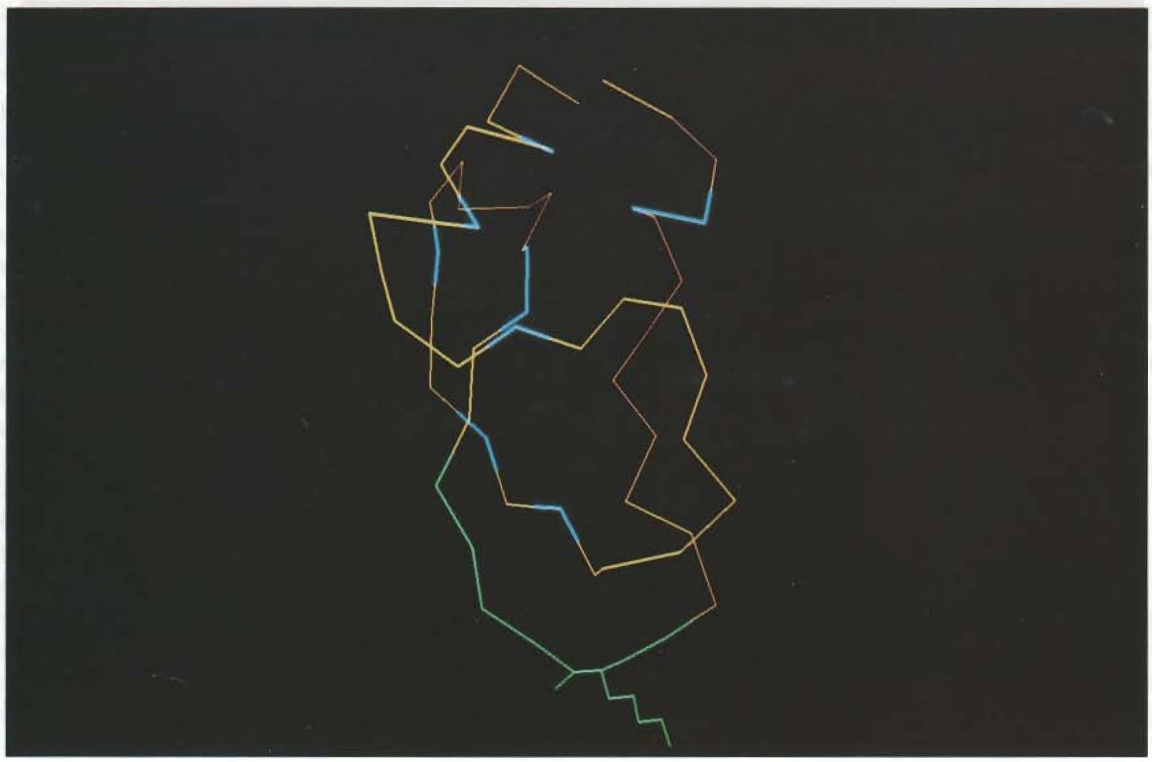
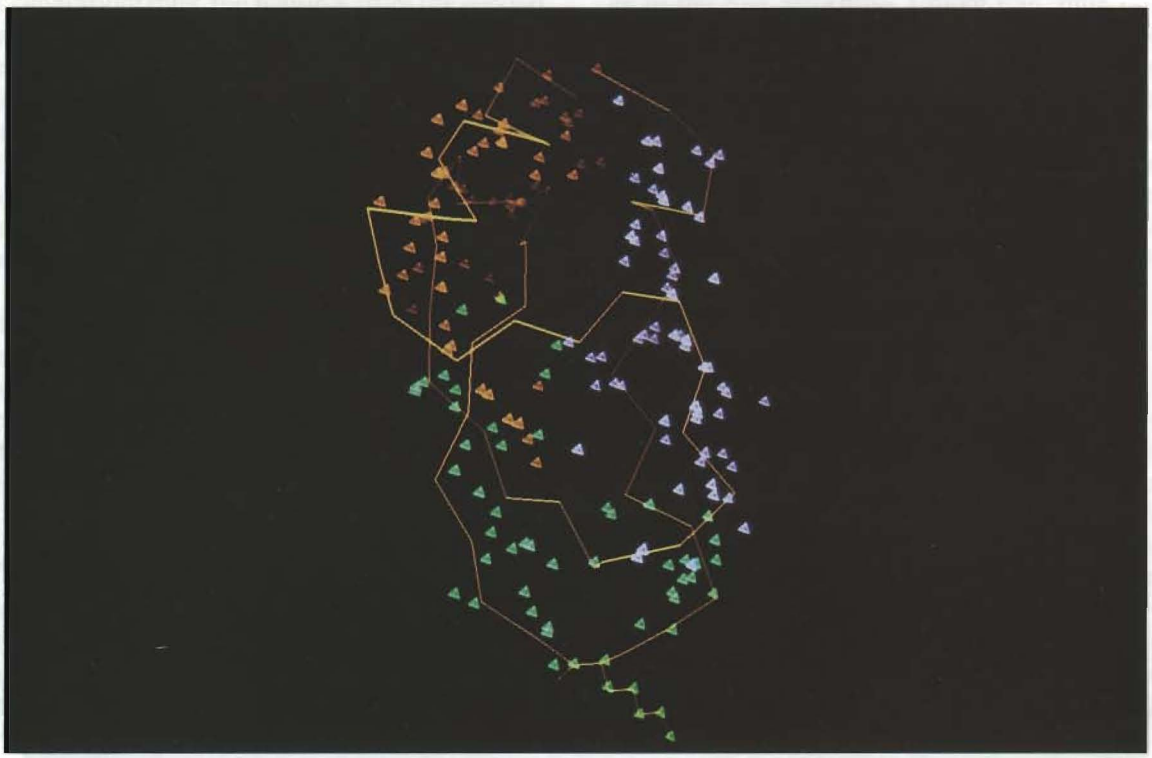


Figure 4. Trypsin spheres. Two subclusters are shown in blue and red, the trypsin molecular surface is colored magenta, and a c-alpha trace of PTI is in yellow. The blue sphere set describes the trypsin specificity pocket.



(a)



(b)

Figure 11. PTI residues organized by structure/function[40] and by subclustering. (a) Structure/function organization of the molecule: The specificity loop of PTI is in green, residues important for the tertiary fold of the molecule are blue, and the rest of the residues are in orange. (b) Subcluster organization of the molecule. Sphere descriptors are represented by triangles. Cluster 3 is in green and clusters 1 and 2 are in orange and magenta, respectively.

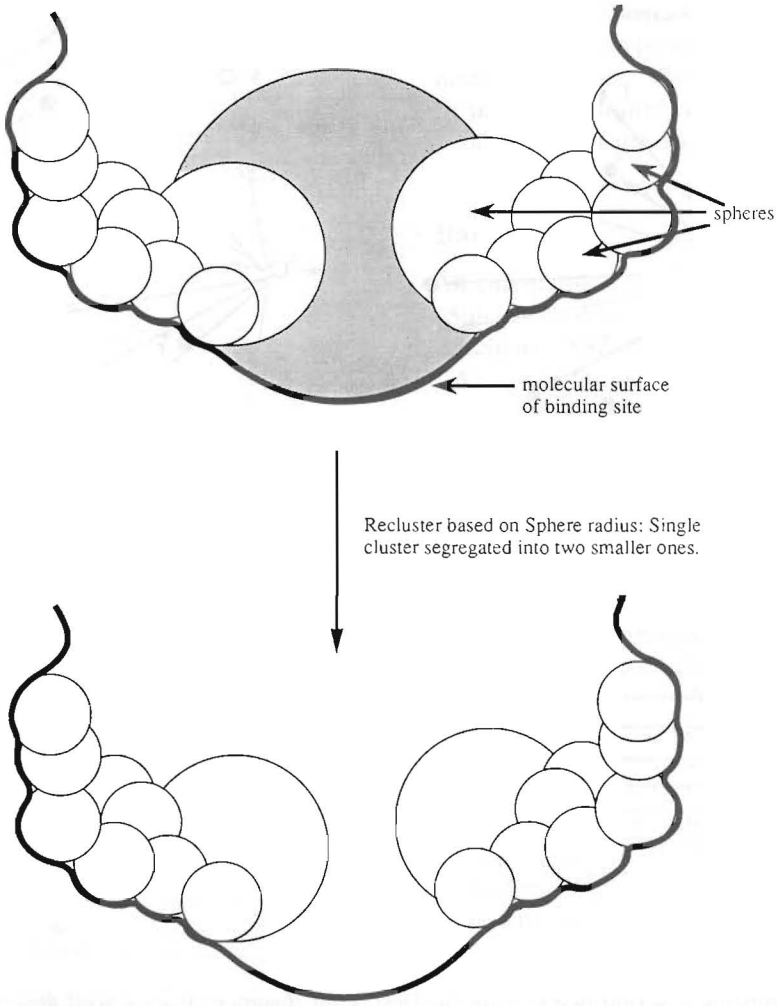
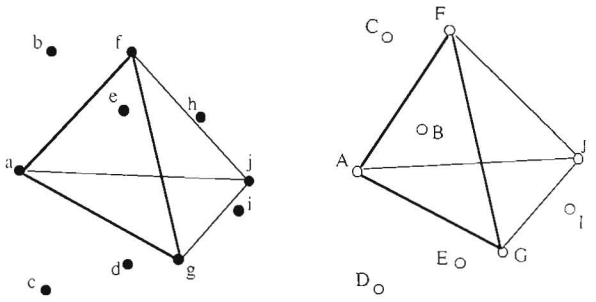


Figure 5. Sphere subclustering. The subclustering algorithm segregates groups of spheres based on their radii. In this example, a large radius sphere (shaded) is removed from the sphere set, removing the link connecting one part of the site to the other and segregating the spheres into two subclusters.



Match if | Distance(i,j)- Distance(I,J) | < tolerance

1st Node: a onto A
2nd Node: j onto J, if aj-AJ < tolerance
3rd Node: f onto F, if af-AF, jf-JF < tolerance
4th Node: g onto G, if ag-AG, jg-JG, fg-FG < tolerance

Figure 6. Internal distance matching. Ligand and receptor internal distances are compared. If the internal distances do not match at a given node, the tree search is "pruned" at this node.

and receptor structures. We calculate an atomic contact "potential" for the receptor by constructing a cubic lattice, which fills the volume of the binding site, and evaluate every point on the lattice on the basis of its contacts with the protein atoms. This lattice is usually calculated once for any given site. A point on the lattice receives a score of one for every receptor atom within a user-defined range of distances and a highly negative score for any contact closer than the low end of this range. We allow the user to distinguish between polar and apolar contacts between the ligand and the receptor atoms by using a second "cutoff" distance parameter (Fig. 9). Thus, ligand atoms are often allowed to come closer to receptor oxygen and nitrogen atoms than to other receptor atoms. The cutoff distances are set by the user—for most systems, we set the polar close contact cutoff to 2.4 Å and used a range of cutoffs between 2.6–2.8 Å for the nonpolar close contacts. In the free conformer protein–protein docking runs, we set the close contact limit to 2.0 Å for both polar

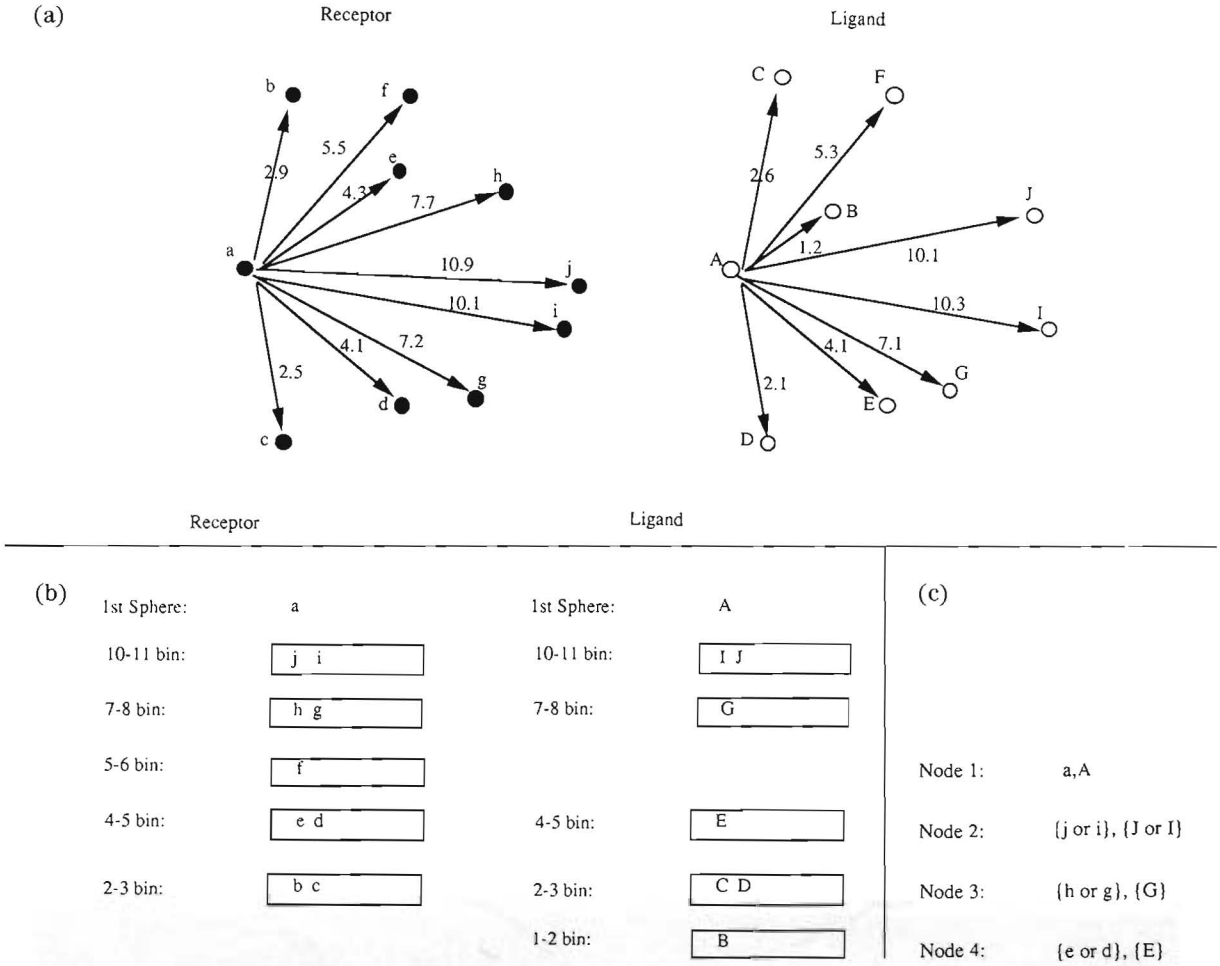


Figure 7. Preorganizing descriptors into bins. (a) Descriptor distances from a seed descriptor, ligand "A" and receptor "a"; all descriptors are used as the seed. (b) Histograms of descriptor internal distances. The resolution of the histograms is user-set; in this figure, they are 1 Å wide. (c) Possible bipartite graphs from (b). For this example, 16 possible graphs can be constructed [1 (possible match at node one) × 4 (possible matches at node two) × 2 (possible matches at node three) × 2 (possible matches at node 4)].

and nonpolar contacts. For the lactate dehydrogenase/NAD-lactate runs we used 2.1 and 2.3 Å as the polar and nonpolar cutoffs, respectively. We set the long-distance cutoff for scoring a contact to 4.5 Å in all runs. Ligand orientations are scored by mapping their atoms onto the nearest lattice points and summing over all of the mapped points. Only one lattice point is used per ligand atom.

The lattice-based scoring differs in three ways from the scoring function used in DOCK versions 1.1 and earlier.[8] First, the lattice method uses a step function for scoring: A ligand/receptor pair of atoms either contributes a score of 1 or 0 or is a "bad contact," whereas the earlier method used a partly continuous exponential function. Second, the lattice method is discontinuous in space since ligand atoms are mapped onto lattice points of some fixed resolution to be scored, while the earlier scoring used the pairwise distance for each atom pair to calculate the score of each ligand–receptor configuration. Last, the lattice method distinguishes between polar and nonpolar contacts, while the earlier method made no distinctions based on atom type.

Sampling and Focusing

To make the sampling procedure as effective as possible for a fixed amount of computer time, we wish to emphasize regions of orientation space where two docking molecules are likely to form productive interfaces and deemphasize regions where this is less likely. We begin by sampling orientation space at a low bin resolution, generating a relatively small number of configurations. As the search proceeds, we monitor whether for a particular first pair of spheres/atoms any of the resulting matches produce configurations with positive scores. If any do, the bins for this set of first points are expanded by the contents of the bins immediately below them in the distance ranking and the graph generation loop is continued

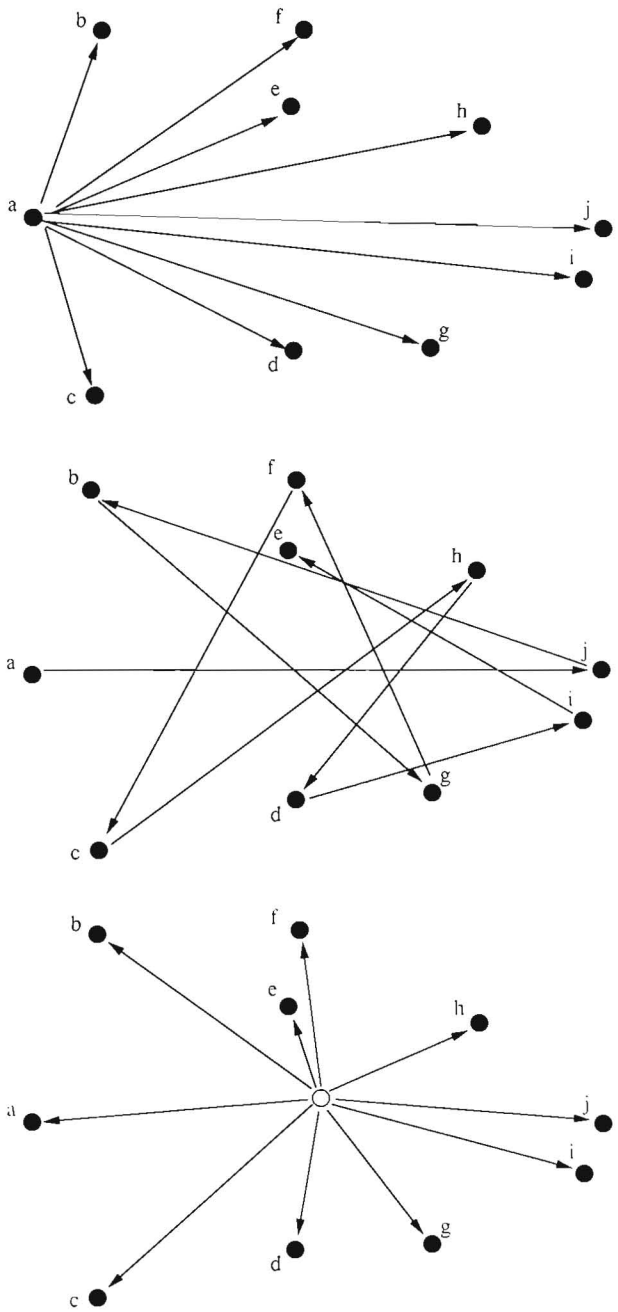


Figure 8. Graph construction methods. (a) Fan procedure. Descriptors are chosen based on their distances from an initial descriptor. (b) Cat's Cradle procedure. Descriptors are chosen based on their distances from the last descriptor chosen. (c) Center of Mass procedure. Descriptors are chosen based on their distances from the center of mass (open circle) of the overall descriptor set.

with these new points. This creates more possibilities for matches in the part of distance space defined by the first pair of successful spheres and atoms. Once a region of orientation space has been examined at this higher level of sampling, the search returns to its former sampling level and proceeds to the next region. More configurations are thus tried in areas that return positive scores than areas that do not. The decision to focus on a region of space from an initial level of general sampling is set dynamically by the program and does not demand human intervention. The user determines only whether to use this feature and how many bin expansions should be performed.

Hardware

All calculations were done on SGI PI 4D/25s, 4D/70 (Silicon Graphics, Inc., Mountain View, CA), and SunSparc (Sun Microsystems, Inc., Mountain View, CA) workstations.

RESULTS

Reproduction of Crystallographic Orientations

We were able to reproduce the experimental configuration of the docked molecules accurately and in a timely fashion in all systems (Table II, Table III). In three of the complexes, we used inhibitors and receptors in their unbound conformations, those adopted by the molecules when they are crystallized independently of their cognate receptor or inhibitor. This was a stringent test of the methodology owing to the conformational differences between the bound and the unbound forms of the molecules.[37]

Having established that we can regenerate the crystallographic configurations of the complexes, we now turn to questions of algorithm performance. We were interested in establishing the relative merits of the three graph construction algorithms we tried: the Fan, Cat's Cradle, and Center of Mass algorithms. We also wished to know how our molecular-organization and sampling techniques contributed to the accuracy and efficiency of the searches.

Comparison of the Graph Construction Algorithms

The different graph construction methods, Fan, Cat's Cradle, and Center of Mass, were tested in four complexes of known structure (Table III), as was an earlier version of DOCK[8] (DOCK1.1). In all four cases, both the Fan and Cat's Cradle algorithms were able to reproduce accurately the crystal complex configuration. The Center of Mass algorithm was not able to reproduce the crystallographic configuration of either the lactate dehydrogenase/NAD-lactate[19] or the trypsin/PTI[20] complex, although it was able to do so for the dihydrofolate reductase/methotrexate[18] and the ribonuclease/uridine vanadate complexes.[17] DOCK1.1 was able to reproduce the crystal complex in the small molecule inhibitor systems, but was not able to do so in trypsin/PTI. The ability to vary the depth of search meant that the Fan and the Cat's Cradle algorithms could always

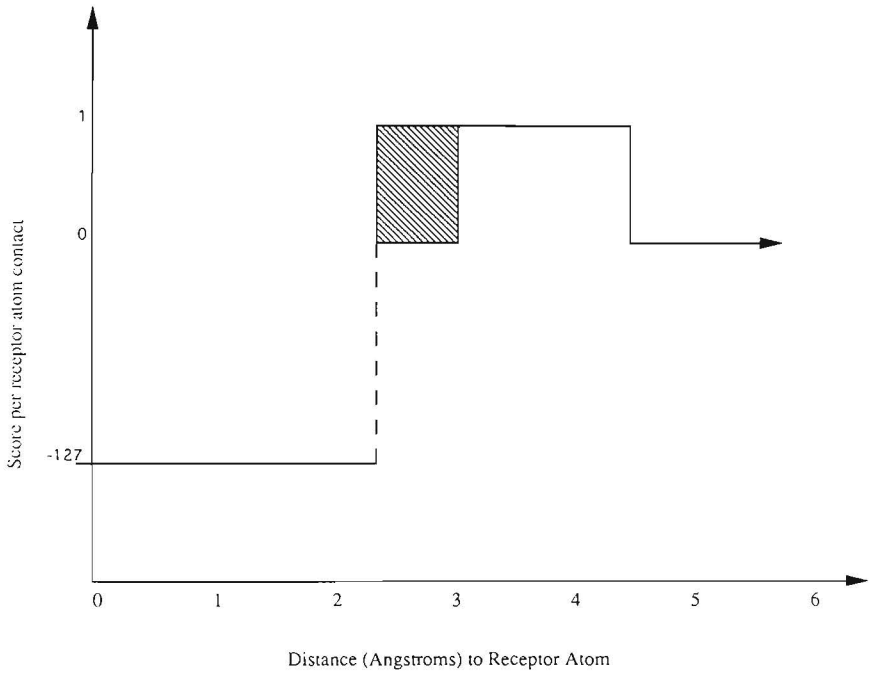


Figure 9. Lattice scoring function. A score of 1 is given to all lattice points within 2.8–4.5 Å of a receptor atom. Lattice points further than 4.5 Å from a receptor atom are given a score of 0. Lattice points closer than 2.4 Å to a receptor atom are considered "bad contacts" and are given very negative score. Lattice points within 2.4–2.8 Å of a receptor nitrogen or oxygen atom (shaded portion of figure) are given a score of 1; points within 2.4–2.8 Å of all other atom types are considered bad contacts and are given very negative scores.

produce lower rmsd configurations than could DOCK1.1.

The Fan algorithm was usually more efficient than the Cat's Cradle algorithm. The Fan method typically produced distributions of ligand configurations biased toward lower rmsd's from the crystal structure result, compared to the other two algorithms, and produced a greater number of low-rmsd dockings in shorter searches (Table III). Fan also produced more acceptable orientations as a percentage of the number tried—this ratio ranged from 1/300 for dihydrofolate reductase/methotrexate (1 ac-

Table III. Comparing the search algorithms.

Protein/inhibitor	Search algorithm	Number of matches[a]	Best rms (Å) to crystal
Trypsin/PTI	Fan	360,336	0.29
	Cat's Cradle	354,346	0.42
	Center of Mass	224,846	4.56
	DOCK1.1[b]	15,453	None found
Dihydrofolate reductase/methotrexate	Fan	5,452	0.35
	Cat's Cradle	11,072	0.99
	Center of Mass	211,155	0.17
	DOCK1.1	16,317	0.86
Lactate dehydrogenase/methotrexate	Fan	69,717	1.55
	Cat's Cradle	2,638	1.27
	Center of Mass	386,043	None w/in 5 Å
	DOCK1.1	78,526	0.89
Ribonuclease/uridine vanadate	Fan	13,737	0.54
	Cat's Cradle	15,916	0.96
	Center of Mass	20,456	0.70
	DOCK1.1	7,955	1.09

[a]Run time is proportional to the number of matches multiplied by the number of atoms in the ligand.

[b]The DOCK1.1 matching algorithm truncates the depth of search of orientation space using heuristics that make it difficult to look at the very high numbers of configurations possible using the bin matching algorithms.

Table IV. Lattice scoring. The correlation of score with rmsd from the crystallographic result for polar and nonpolar lattices is compared.

Receptor	Inhibitor	Polar lattice[a]		Neutral lattice[a]	
		Top 10[b]	R[c]	Top 10[b]	R[c]
Ribonuclease	Uridine vanadate	4/10	−0.39	0/10	−0.28
Dihydrofolate reductase	Methotrexate	6/10	−0.59	0/10	−0.23
Lactate dehydrogenase	NAD-lactate	9/10	−0.60	2/10	−0.22
Trypsin	PTI	7/10	−0.33	4/10	−0.18

[a]Polar lattices distinguish between close contacts to receptor oxygen or nitrogen atoms and all other receptor atom types. Neutral lattices treat all receptor contacts equally.

[b]Number of top 10 scoring orientations calculated by DOCK2 that have rmsd values to the crystallographic result that are less than 2.5 Å.

[c]Correlation between rmsd from the crystallographic configuration as a function of score. The highest correlation is when R is −1 (high score, low rmsd).

ceptable orientation for every 300 tried by DOCK2) to 1/1000 for trypsin/PTI, while for the Cat's Cradle procedure the ratios were worse by a factor of three.

Scoring on the Lattice

The new scoring routine improves run times by a factor of four to five for larger sites (60 or more spheres), compared to the previous scoring method, which explicitly calculated atom–atom contacts between the ligand and the receptor. The ability to distinguish between polar and nonpolar contacts significantly improves the ordering of the docked orientations as a function of score compared to the experimental result (Table IV). With polar scoring, more of the top 10 scoring dockings are within 2.5 Å of the crystallographic result than with nonpolar scoring. We also notice an improved correlation between scores calculated on a polar lattice and rmsd from the crystal structure as compared to scores calculated using a nonpolar lattice. We caution, however, that there is no reason to expect even a monotonic relationship between a measure of complementarity and rmsd. Lattice generation time depends on resolution and the size of the site, but typically takes 1–2 (CPU) min on a SGI PI 4D/25.

Subclustering

Subclustering segregates molecular features into regions that are treated independently for docking. In trypsin, for example, the initial sphere calculation produced an initial set of 102 spheres, which spanned the active site cleft. Subclustering divided this set into several smaller ones, the two largest having 35 and 30 spheres in them. In a similar way, the initial sphere set for the PTI was spread over the entire volume of the molecule and included 292 spheres, approximately as many spheres as there are solvent-exposed atoms in the molecule. Subclustering produced 6 subclusters ranging from 40–90 spheres.

The effect of subclustering in the macromolecular docking calculations was dramatic; we show the results for trypsin/PTI in Table V. For a given number of matches—or run time, which is roughly proportional to the number of matches multiplied by the number of ligand atoms—the accuracy of the configurations produced was much higher in the subcluster runs than in the full cluster set runs. Even in runs involving extremely large numbers of matches, the full cluster dockings could not find configurations that resembled the crystal structure. Subclustering transforms docking from a problem that

Table V. Effects of subclustering on run time and accuracy.

Protein inhibitor	Subclustered spheres				Unclustered spheres			
	Number of clusters[a]	Total spheres[b]	Total matches[c]	Best rmsd to crystal (Å)	Number of clusters[a]	Total spheres[b]	Total matches[c]	Best rmsd to crystal (Å)
Trypsin/PTI	2/3	66/255	137,972	0.30	1/1	102/292	27,099,614[d]	21.7
DHFR/methotrexate	2/1	77/33	5,704	0.35	1/1	89/33	10,967	0.35
Ribonuclease/uridine vanadate	2/1	53/20	4,389	1.79	1/1	76/20	10,022	0.54
Lactate dehydrogenase/NAD-lactate	4/3	145/63	62,736	0.74	1/1	189/51	175,280	0.51

[a]Number of receptor/ligand clusters.

[b]Total number of receptor/ligand spheres in all clusters. Overlaps are permitted between spheres in one subcluster and another. Occasionally, this leads to greater totals of subclustered spheres than are present in the unclustered sets.

[c]Run time is proportional to number of matches multiplied by the size of the ligand.

[d]This run was not allowed to go to completion, but was stopped after those PTI spheres in the interface region with Trypsin as defined in the crystal structure (approximately 1/5 of the molecule) had been searched. The full search would have required many more matches.

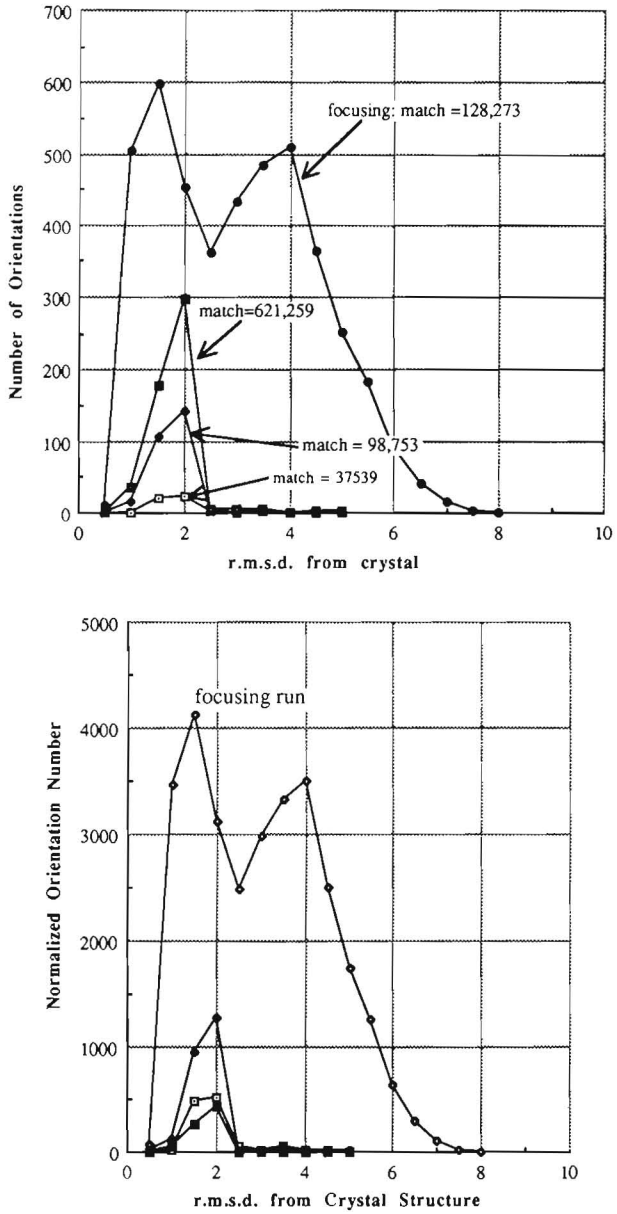


Figure 10. Focusing in trypsin/PTI. Open squares represent a small bin, low match number docking run; the closed triangles represent an intermediate match number run; the closed squares represent the maximum bin size and match number run. The closed circles represent a run with the same bin sizes as the open squares, but with focusing. (a) Number of orientations generated in a docking run plotted against the rmsd of the orientations compared to the crystallographic result. (b) The same data is presented, normalized for match number.

scales, minimally, as the third to fourth power of the size the molecules to one that scales more linearly* with molecular size.

The results of subclustering in the small molecule dockings are less convincing. While the technique reduced run time in most systems, the effect was not as great as in the macromolecular systems; good results could be achieved using the unclustered spheres. Unlike the macromolecular ligand runs, in which subclustering was essential to the regeneration of the experimental result, its value in the small ligand systems was case dependent. In dihydrofolate reductase/methotrexate calculations, subclustering improved run time without sacrificing accuracy. In lactate dehydrogenase/NAD-lactate, run time is improved with only a small decrease in accuracy. Also, fewer high-rmsd configurations of ligand were generated. In the ribonuclease/uridine vanadate system, on the other hand, the shorter run time using the subclustered spheres led to lower accuracy and poorer sampling.

Sampling and Focusing

DOCK2 runs that use focusing return more low-rmsd ligand configurations than runs that sample orientation space at a constant level, even though the latter procedure looks at significantly more matches (Fig. 10). In four test complexes, focusing increased the ratio of high-scoring orientations per match number by a factor of 3–10 (results not shown).

DISCUSSION

Molecular docking searches orientation space for favorable configurations of a ligand in a receptor. Like most search methods with many degrees of freedom, docking can only sample solutions within the space it explores. A docking search will therefore always be faced with a fundamental trade-off between computation time and accuracy or, more correctly, adequate sampling. We are interested in reducing the time of search necessary to produce a given level of accuracy or sampling. The docking algorithm has three basic levels: molecular description, the sampling of orientation space, and the evaluation of configurations. We discuss algorithm modifications at each of these levels and their effect on run time and accuracy. We measure accuracy with reference to the experimental result, the crystal structure of the protein–ligand complex, although we understand that other considerations may also be important.

Accuracy

A basic question for a docking algorithm is how long it takes to get a solution near the experimental structure. In each of our 10 test systems, the new routines reproduced the crystallographic configuration ac-

*It is difficult to determine accurately how the new algorithms scale with molecular size since parametric choices (such as bin size) in the various systems can have a large effect on run time and the quality of the results. We note (Table V) that it took fewer matches (and consequently less time) to arrive at a low-rmsd docking of PTI in trypsin, using the subclustering technique than were required for docking NAD-lactate into lactate dehydrogenase where subclusters were not used, even though PTI/trypsin is by far the larger system.

curately in a reasonable amount of time. This is a compelling result. The test systems we chose varied in their crystallographic resolution (from 2.7 Å for lactate dehydrogenase[19] to 1.9 Å for trypsin[20]); their size (from the 20-atom uridine vanadate to the 2143-atom thymidylate synthase monomer); and their molecular determinants of binding (the ribonuclease complex relies largely on electrostatic recognition, whereas the macromolecular complexes have large hydrophobic components). Generating known complexes starting with macromolecules in their *unbound* conformations is a striking outcome that suggests that the algorithms might be used predictively.[37]

The accuracy of the descriptor-based docking calculations reflects the ability of the spheres to identify local binding grooves and ridges. The efficiency of the calculations reflects the success of the subclustering and focusing techniques in concentrating the search on regions of orientation space likely to have high complementarity. This is the advantage of descriptor-based docking over grid searches of receptor sites.[38,39] Because grid searches are necessarily unbiased regular samplings of orientation space, they are much slower than DOCK2, which preidentifies receptor regions of high local curvature to search in. The ability of DOCK2 to dynamically respond to search results through focusing only accentuates this difference. The advantage of the grid methods is that they will always work, given enough time, whereas descriptor-based docking relies on selecting the appropriate features of the molecules and the avoidance of the combinatorial explosion problem. There will probably be systems that do not lend themselves to description by spheres, such as ones that have a flat protein–protein interface or that cannot be subclustered into independent binding regions. In such systems, DOCK2 will not work, whereas grid based methods will. In the 10 systems we report on in this article, however, DOCK2 can generate accurate reproduction of the crystallographic configuration in minutes or hours on a workstation. Methods using grid searches of orientation space to solve the docking problem can take days on much faster machines.[38,39]

Choosing between the Searching Algorithms

We tried three different methods for choosing which features of one molecule to map onto the those of a second. Both the Fan and Cat's Cradle algorithms reproduced the experimental results in all systems, while the Center of Mass algorithm did so in only two of four complexes it was tested against. The Center of Mass method probably fails because of the relatively few spheres it uses as matching descriptors. Since the Center of Mass matching chooses centers for bipartite graph generation based on a fixed reference point, fewer aspects of the molecule or site will be sampled in graph generation compared with the Fan or Cat's Cradle procedures, where the reference point is different for each first pair of spheres. Choosing between the Fan and Cat's Cradle algorithms is more difficult on theoretical grounds. The Cat's Cradle algorithm will more often sample the longest internal distance of a molecule or site while building the bipartite graph and will therefore more consistently use the principal topographic features of the molecules in matching. The Fan algorithm, on the other hand, will generally sample more of the features of a molecule or site. Practically, the Fan method seems to perform more efficiently than the Cat's Cradle method, although this result might reflect our implementation and should be tested for other systems.

Scoring on the Lattice

The improvement in run time with lattice scoring more than justifies its decreased resolution compared to scoring in a continuous space. Although the scoring scheme used in the lattice implementation is simpler than in the previous versions of the program,[8] scores from the two methods correlate well with each other (results not shown). The exact numerical score for any given orientation will, of course, differ between the two metrics, as described in the Methods. Since there is no good physical reason to choose one scoring function over the other, we used the simpler function in this work. The introduction of polar differentiation in the scoring scheme improves the correlation between a configuration's score and its similarity to the crystallographic result compared to nonpolar scoring. Such a correlation must, however, be interpreted cautiously. While it is gratifying that the highest scoring configurations in our test cases closely resemble the crystallographic result, we note that there are often configurations whose scores are almost as high that do not resemble it. This is most apparent in the dockings of the unbound conformations of the protease/protease-inhibitor pairs. The shape-based scoring is potentially weakest when comparing the complementarity of different putative ligands for the same receptor, which is what is done in inhibitor design applications of DOCK.[4] While the method continues to prove itself useful in the design of novel inhibitors[4] (Shoichet, unpublished results; Bodian, unpublished results), rankings of molecules based on their DOCK score should not be overinterpreted.

Subclustering

The fundamental change in our approach that allows us to treat macromolecular docking is our modification of the clustering algorithm. The introduction of a radial cutoff organizes and segregates molecular

features into topographically distinguishable regions. By reducing the size of each cluster, we overcome the strong time dependence of docking with system size. We assume that two regions that are distinguishable are also independent. The method should be evaluated by two criteria: Does it genuinely separate a molecule into physically distinct regions and does it increase the efficiency of the search without compromising its accuracy?

In the protein–protein complexes, for which the subclustering technique is most important and useful, the issue of how subclusters correspond to physical regions of the molecule can be addressed by organizing the residues of a protein along structural and functional lines. Residues of PTI, for instance, can be assigned a role either in stabilizing the tertiary fold of the molecule or in binding to trypsin[40] [Fig. 11(a), see color]. Comparing PTI[20] organized by subclusters [Fig. 11(b), see color] to the structure/function organization of the molecule, one notices that the subclusters correspond to either structural or functional regions. The binding loop residues in Fig. 11(a) are completely described by cluster 3 in Figure 11(b). The hydrophobic pocket residues in Figure 11(a) are found in clusters 1 and 2, which divide the nonbinding part of PTI between them. The binding loop cluster has few overlaps with the hydrophobic regions of the molecule. Subclustering thus seems to do a good job of separating PTI into physically and functionally distinct regions.

Clustering improved the efficiency of the docking runs dramatically without sacrificing accuracy in all the protein–protein complexes we tested. The results for PTI/trypsin with and without subclustering are compared in Table V. Without subclustering, macromolecular docking is not practical for our method.

The results for the small molecule test cases are less persuasive. While subclustering still shortened run times, the unclustered spheres always reproduced the crystallographic results in reasonable amounts of time. In lactate dehydrogenase/NAD, for example, subclustering meant performing 12 different docking runs for all combinations of ligand and receptor clusters. While this did improve run times and lead to better distributions of the docked orientations around the crystallographic result (Table V), the unclustered sphere sets clearly provided an adequate description of the molecules. The different impacts of subclustering on the macromolecular ligand and small ligand systems might reduce to a question of size. The small molecule ligands, and their cognate receptor binding grooves, simply lack the heterogeneous topologies that make subclustering necessary for the macromolecular systems. Having said this, subclustering *does* reduce search times in the small ligand systems and will be a useful technique whenever one wants to target specific regions of a site for a docking search or when docking to a large receptor site.

Sampling and Focusing

In molecular docking, it is important to be able to survey the general features of configuration space and then concentrate on those areas that offer the greatest possibilities for complementarity. The focusing algorithm guides a search by expanding the number of molecular descriptors in regions of distance space that return favorable orientations, leading to longer searches in these regions than in regions that do not return favorable matches at the initial low-density sampling. The technique is essentially a variation on the tree-search-with-pruning approach: Rather than cutting off branches due to a failure of an early node, branching is increased due to an early success.

Focusing improves the efficiency of a search of orientation space. Both the number of low-rmsd configurations and the number of high-scoring configurations per number of matches increase dramatically with focusing, which allows for faster searches to achieve the same degree of accuracy (Fig. 10). Focusing is most effective in protein–protein complexes. In a search of trypsin/PTI using small bins (low initial sampling levels) and high sensitivity to focusing signals, the number of matches necessary to achieve either a high score or a low rmsd value to the crystallographic result was reduced by a factor of 100 compared to a run where focusing was not done (Fig. 12). The nonfocusing run shows a monotonic increase in the best score or rmsd result achieved with the level of sampling up to a maximum value. This is the expected result for a discrete, unbiased sampling of configuration

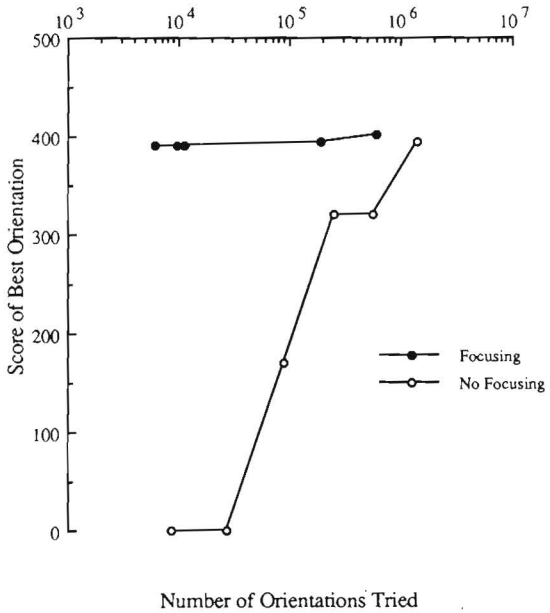


Figure 12. Sampling issues in focusing. Maximum score achieved vs. number of orientations tried. Solid circles, using focusing procedure; empty circles, no focusing.

space. The focusing run, on the other hand, achieves a result very close to the maximum almost immediately and improves only slightly as more and more orientations are looked at.

One caveat for focusing is that it often increases the number of high-scoring orientations distant from the crystal configuration, as well as increasing the number of orientations close to the crystallographic result (Fig. 10). This reflects the different spaces in which orientations are first generated and then evaluated. In generating a ligand–receptor configuration, we match internal distances between molecular descriptors. Focusing increases the number of descriptors to match in a particular region of *distance* space. Two points that are the same distance to a third point will not necessarily be close to one another in Cartesian space, and focusing based on distance information can therefore lead to the inclusion of descriptors from a different region of the molecule than the one that was involved in the initial, low-level sampling match. This explains the broadening of the rmsd distributions, which are measured in Cartesian space. Notwithstanding this feature, focusing always increases the number of high scoring configurations as a percentage of matches tried. Since the signal to focus on a particular region is score based, this is perhaps a more consistent metric.

Sampling configuration space at variable densities is a physically sound approach to a problem that can have an infinite number of solutions. We have outlined a procedure for focusing that meshes easily with our docking algorithm—other methods are certainly conceivable. The general approach is not limited to molecular docking, but should be useful in any method where low-density sampling can guide high-resolution searches. Such methods might include docking on a regular lattice[12,38] or in grid searches of conformation space, where smaller step sizes (step size here being a torsion angle, an Euler angle, or a translation) would be used in low-energy regions of the energy surface and larger step sizes in high-energy regions. Guided searches are implicitly implemented in Monte Carlo methods for simulating molecular dynamics,[41] although here it is not step size but rather time spent in a particular region of space that is modified, so the analogy is only approximate.

UNSOLVED PROBLEMS

Using low-resolution representations of molecules, either in the form of potential functions or topography, to guide searches of molecular interactions is a general problem in the field.[32,42] In this article, we have shown how methods for organizing high-resolution information can be used to address this issue. Both subclustering and focusing do not, however, use actual low-resolution information as a guide. It would be conceptually appealing, and practically rewarding, to use low-resolution information to prune the search tree. This would allow one to limit searches that use high-resolution features of the molecules, which are the most expensive computationally, to regions of likely complementarity. To our knowledge, the general problem of using resolution to guide searches remains largely unaddressed.

The scoring function that DOCK2 uses to evaluate configurations, although improved from the one used in DOCK, is still too simplistic. Our concern previously had been that a more complex scoring function, of the sort used in molecular mechanics for instance, could only be used at the cost of reducing the amount of orientations we could look at. This should not be the case for a lattice-based scoring method, however.[2]

Finally, we have not discussed the issue of conformational flexibility. The degrees of freedom in a docking problem that allows for conformational as well as configurational sampling are potentially very large, which implies either that searching such a space would be very slow or very incomplete or both. There are ways to reduce the degrees of freedom of this problem. If one defines local regions of space as interaction zones, and only looks at conformations in this region while keeping the rest of the system rigid, then the issue becomes more tractable.[43,44] Alternatively, if one keeps the protein rigid and only allows the generally much smaller ligand to sample conformation space, the size of the space is similarly reduced.[2] This method suffers in situations where conformational accommodation takes place largely at the receptor, which some have argued is the general case.[45]

Applications

The modifications encoded in DOCK2 improve our ability to model biological systems[37] and design novel enzyme inhibitors[4] (Shoichet, unpublished results; Bodian, unpublished results). The changes in matching algorithm give the user more control over the depth of search in docking calculations, while the lattice scoring makes the program faster and leads to more sensible evaluations of orientations. The subclustering technique will be useful in systems where one wishes to explore particular regions of a molecule in detail while downplaying others. Subclustering will also significantly improve run time efficiency in large sites, and the technique is essential for the docking of two macromolecules, an area of current pharmaceutical interest. The focusing technique will improve the efficiency of a search, judged by the number of complementary orientations per number of matches tried. Focusing will be especially useful when highly detailed searches of a particular regions are required, as will be the case

when trying to reproduce or predict a biological complex or when trying to capture particular details of a binding interaction.

CONCLUSIONS

DOCK2 can reproduce the experimental configurations of protein–ligand complexes in a wide variety of systems. We have shown how docking can be changed from a problem that scales as the fourth power of system size to one that scales linearly with it. This is achieved by breaking down the description of the molecules into independent pieces and concentrating high-resolution searches of orientation space on those regions that return favorable complexes at low resolution.

The success of any feature-based docking scheme depends on how descriptors are chosen for matching between molecules. We have found that the Fan and Cat's Cradle internal distance algorithms work well, while the Center of Mass method does not. We have described how new routines allow for variable depth searches and more efficient scoring of orientations. Compared to our previous implementations,[7,8] the current methods allow faster and more complete searches of orientation space. The algorithm is well suited to macromolecular docking, which the earlier methods were unable to treat.

The authors thank Elaine Meng, Renee DesJarlais, Richard Lewis, and Andrew Leach for their helpful comments throughout this work. Financial support was provided by the National Institutes of Health (GM-31497, GM-39552, and 5T32 GM08120 for DLB). We used the graphics facilities of the UCSF computer graphics laboratory, R. Langridge, director (RR-1081), for some of this work.

References

1. R.L. DesJarlais, R.P. Sheridan, J.S. Dixon, I.D. Kuntz, and R. Venkataraghavan, *J. Med Chem.*, **29,** 2149 (1986).
2. D.S. Goodsell and A.J. Olson, *Proteins*, **8,** 195 (1990).
3. P.J. Goodford, *J. Med. Chem.*, **27,** 557 (1984).
4. R.L. DesJarlais, G.L. Seibel, I.D. Kuntz, P.O. de Montellano, P.S. Furth, J.C. Alvarez, D.L. DeCamp, L.M. Babé, and C.S. Craik, *Proc. Natl. Acad. Sci. USA*, **87,** 6644 (1990).
5. D.L. Miller and J.F. Pekny, *Science*, **251,** 754 (1991).
6. F.H.C. Crick, *Acta Crystallogr.*, **6,** 689 (1953).
7. I.D. Kuntz, J.M. Blaney, S.J. Oatley, R. Langridge, and T.E. Ferrin, *J. Molec. Biol.*, **161,** 269 (1982).
8. R. DesJarlais, R.P. Sheridan, G.L. Seibel, J.S. Dixon, I.D. Kuntz, and R. Venkataraghavan, *J. Med. Chem.*, **31,** 722 (1988).
9. M.L. Connolly, *Biopolymers*, **25,** 1229 (1985).
10. R.H. Lee and G.D. Rose, *Biopolymers*, **24,** 1613 (1985).
11. P. Zielenkiewicz and R. Andrzej, *J. Theor. Biol.*, **111,** 17 (1984).
12. S.J. Wodak and J. Janin, *J. Molec. Biol.*, **124,** 323 (1978).
13. P.J. Goodford, *J. Med. Chem.*, **28,** 849 (1985).
14. S.J. Wodak, M. De Crombrugghe, and J. Janin, *Prog. Biophys. Molec. Biol.*, **49,** 29 (1987).
15. F.S. Kuhl, G.M. Crippen, and D.K. Friesen, *J. Comp. Chem.*, **5,** 24 (1984).
16. F.C. Bernstein, T.F. Koetzle, G.J.B. Williams, E.F. Meyer Jr., M.D. Brice, J.R. Rodgers, O. Kennard, T. Shimanouchi, and M. Tasumi, *J. Molec. Biol.*, **112,** 535 (1977).
17. B. Borah, C.W. Chen, W. Egan, M. Miller, and A. Wlodawer, *Biochemistry*, **24,** 2058 (1985).
18. J.T. Bolin, D.J. Filman, D.A. Matthews, R.C. Hamlin, and J. Kraut, *J. Biol. Chem.*, **257,** 13650 (1982).
19. U.M. Grau, W.E. Trommer, and M.G. Rossmann, *J. Molec. Biol.*, **151,** 289 (1981).
20. M. Marquart, J. Walter, J. Deisenhofer, W. Bode, and R. Huber, *Acta Crystallogr., Sect. B*, **39,** 480 (1983).
21. T.E. Ferrin, C.C. Huang, L.E. Jarvis, and R. Langridge, *J. Molec. Graph.*, **6,** 13 (1988).
22. C.A. McPhalen and M.N.G. James, *Biochemistry*, **27,** 6582 (1988).
23. M. Fujinaga, A.R. Sielecki, R.J. Read, W. Ardent, M.J. Laskowski, and M.N.G. James, *J. Molec. Biol.*, **195,** 397 (1987).
24. W.R. Montfort, K.P. Perry, E.B. Fauman, J.S. Finer-Moore, G.F. Maley, L. Hardy, F. Maley, and R.M. Stroud, *Biochemistry*, **29,** 6964 (1990).
25. J. Walter, W. Steigemann, T.P. Singh, H. Bartunik, W. Bode, and R. Huber, *Acta Crystallogr., Sect. B*, **38,** 1462 (1982).
26. J.J. Neidhart and G.A. Petsko, *Protein Engng.*, **2,** 271 (1988).
27. C.A. McPhalen, and M.N.G. James, *Biochemistry*, **26,** 261 (1987).
28. R.A. Blevins and A. Tulinsky, *J. Biol. Chem.*, **260,** 4264 (1985).
29. W. Bode, O. Epp, R. Huber, M. Laskowski, and W.J. Ardelt, *Eur. J. Biochem.*, **147,** 387 (1985).
30. P. Ehrlich, *Chem. Berichte*, **42**, 17 (1907).
31. G.D. Rose, *Nature*, **272,** 586 (1978).
32. S.E. Leicester, J.L. Finney, and R. Bywater, *J. Molec. Graph.*, **6,** 104 (1988).
33. M.L. Connolly, *Science*, **221,** 709 (1983).
34. D.R. Ferro and J. Hermans, *Acta Crystallogr.*, **A33,** 345 (1977).
35. J.A. Hartigan, *Clustering Algorithms*, Wiley, New York, 1975.
36. R.J. Wilson, *Introduction to Graph Theory*, Longman, Harlow, UK, 1985.
37. B.K. Shoichet and I.D. Kuntz, *J. Molec. Biol.*, **221**, 327 (1991).
38. F. Jiang and S.H. Kim, *J. Molec. Biol.*, **201,** 79 (1991).
39. H. Wang, *J. Comp. Chem.*, **12,** 746 (1991).
40. J.R. Read and M.N.G. James, In *Proteinase Inhibitors*, A.J. Barrett and G. Salvesen, Eds., Elsevier, Amsterdam, 1986, pp. 301–335.
41. N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E.J. Teller, *Chem. Phys.*, **21,** 1087 (1953).
42. W.F. van Gunsteren and H.J.C. Berendsen, *Angew. Chem. Int. Ed. Engl.*, **29,** 992 (1990).
43. J.W. Ponder and F.M. Richards, *J. Molec. Biol.*, **193,** 775 (1987).
44. C. Wilson, J.E. Mace, and D.A. Agard, *J. Molec. Biol.*, **220,** 495 (1991).
45. A. Warshel and M. Levitt, *J. Molec. Biol.*, **103,** 227 (1976).