Efficient Clustering of Molecular Conformations

(Abstract)

B. COOK * L. E. KAVRAKI* R. MOTWANI†

During the process of pharmaceutical drug design, computational chemists often compute different low-energy conformations of small drug molecules (ligands). Such conformations are used in solving docking problems [5] and in pharmacophore identification [3]. Most methods that identify different low-energy conformations of ligands employ clustering to partition the generated conformations (typically tens of thousands) into sets that capture geometric similarity. Since the generation and clustering of conformations is a time-consuming operation [2], recent work focuses on devising faster algorithms for these problems. In this paper, we consider the problem of clustering conformations of ligands with 3-20 rotatable bonds. We report a number of experiments which indicate that while clustering based on distance measures defined over the workspace (W-space) has much better quality than clustering based on distance measures defined over the conformation space (C-space), the running times required are much worse in the former. We suggest a technique for obtaining performance close to that of W-space clustering, using time close to that of C-space clustering. Although our testbed is molecular conformations, our results may be relevant to other applications such as clustering of images and documents.

Our starting point is the furthest-neighbor clustering algorithm of Gonzalez [4]. This algorithm minimizes the maximum distance of a conformation from the head of its cluster and guarantees a solution within two times the optimum. The algorithm proceeds as follows. The first conformation is assigned as the head of the first cluster and all conformations belong to that cluster. At each subsequent step, the conformation which is the furthest away from the the head of its cluster is selected to head a new cluster. All conformations are then revisited. If they are closer to the selected conformation than to the head of their current cluster, they are moved to the newly generated cluster. This procedure is repeated until a specified number of clusters is reached, or until the maximum distance of a conformation from the head of its cluster falls below a predefined threshold. The algorithm runs in $O(kn)$ time, where n is the number of conformations being clustered and k is the number of clusters produced.

A commonly-used distance for measuring similarity is a W-space distance called least root mean squared distance (l_{lrmsd}) [1]. To compute it a singular value decomposition algorithm (SVD) is used to find the transformation that minimizes the quantity $\sqrt{(\sum_1^N d_i^2)/N}$, where N is the number of atoms in the molecule and d_i is the Euclidean distance of the corresponding atoms in the molecule. Table 1 reports experiments with 10,000 conformations of 4 different molecules. The time required to produce 60 clusters is shown in column 4. All experiments are done on a SGI R10000 and the code is written in C++. The quality of the clustering is shown by reporting the average of the l_{lrmsd} distances of each conformation from the head of its corresponding cluster [2]. We call this quantity the average cluster radius; refer to the first graph in Figure 1. l_{lrmsd} is a good measure of the geometric similarity of two conformations [1], but it is relatively expensive to compute since it considers the distances of all atoms in the two conformations and uses the SVD algorithm whose running time is $O(N^3)$.

We can achieve a substantial reduction in the running time of the furthest-neighbor clustering algorithm if we use a less expensive distance measure. Distances that involve only C-space parameters are good

*Department of Computer Science, Rice University, Houston, TX 77005

†Department of Computer Science, Stanford University, Stanford, CA 94305

candidates. These distances consider only the R rotatable bonds of the ligands which are typically much smaller than N, the number of atoms in the molecule (see columns 2 and 3 of Table 1). An obvious measure of distance in C-space is $l_1 = \sum_1^R |d\theta_i|$, where $d\theta_i$ is the difference in the values of the corresponding rotatable bonds in the two conformations under consideration. Variations of this measure are possible. For example, one can weight the terms so as to reflect the "kinematic distance" of the corresponding bond from the "leaves" of the molecular tree. The motivation for the weighting scheme is that changes in the values of the rotatable bonds in the middle of the molecule have a greater effect on the overall shape of the molecule than changes in the values of bonds that are close to the "leaves" of the molecule.

Figure 1 compares the results given by W-space clustering and the results obtained using different C-space metrics. For uniformity and accuracy we report in Figure 1 the average cluster radius as measured by l_{lrmsd} independent of which distance metric was used for creating the clusters. The first graph shows the results obtained with W-space clustering. Then we report results with different C-space distances: (a) $l_1^{adapt} = \sum_1^R w_i^{adapt}$, with weights computed as follows. First a conformation c is selected. Then each bond is spun around and l_{lrmsd} is stored as the weight of the bond for the considered rotation. (b) l_1 with weights that are the cubes of the kinematic distances of the rotatable bonds; (c) l_1 with weights that are the squares of the kinematic distances of the rotatable bonds; (d) l_1 with linear weights — actual kinematic distances; (e) l_1 with sublinear weights — square root of the kinematic distances; and, (f) l_1 without weights as defined above. The running time of all C-space schemes is roughly the same, and all are substantially less than in the case of W-space clustering (see Table 1, column 5). However, the quality of the results is clearly inferior to that of W-space clustering. Weighting helps but does not significantly improve the overall quality of the clustering. Experiments with l_2 or l_{inf} measures in C-space, with or without weighting, seem to give similar results.

A good compromise is to use distance measures in the W-space and the C-space in an interleaved manner: clustering is first based on a C-space distance and k_1 clusters are produced; then, the more expensive W-space distance is used to cluster the conformations in each of k_1 clusters into k_2 clusters, for a total of $k = k_1 k_2$ clusters. We refer to this scheme as *mixed clustering*. An analysis of the running time of mixed clustering (omitted in this abstract) reveals that the running time is dominated by $O(k_2 n N^3)$ and that the ratio of the running times of W-space clustering over mixed clustering is approximately k/k_2.

Figure 2 plots the results of W-space clustering in comparison with those of the best C-space clustering (adaptive weights) and those of mixed clustering (C-space clustering with adaptive weights into k_1 sets followed by W-space clustering into k_2 sets). The corresponding running times are shown in Table 2. Note that the quality of clustering is improved for a substantial reduction in the running time: if C-space clustering in 5 sets is first performed and is followed by W-space clustering, we achieve results within 17% of the W-space clustering results for a 5-fold reduction in the running time. If C-space clustering partitions into a larger number of sets, for example 10, then we achieve results within 25% of the W-space clustering for a 8-fold reduction in the running time.

Figure 3 shows how W-space and C-space distances are related and provides an intuitive explanation for the good performance of mixed clustering. We consider a sample of 2,000 pairs of a typical drug molecule (ace16). For each pair i, x_i denotes the l_{lrmsd} distance of the two configurations and y_i denotes their l_1^{adapt} distance. In the figure we plot averages of $d_i = 1 - \frac{|x_i - y_i|}{(x_i + y_i)}$, where $\frac{|x_i - y_i|}{(x_i + y_i)}$ can be regarded as the percent difference between the calculated l_{lrmsd} and l_1^{adapt} distances for pair i. Averages over moving windows of 10, 50, and 100 pairs are shown. Despite the noise present for small distances, there is a clear trend for large distances to be strongly correlated. Thus, the creation of k_1 clusters using l_1^{adapt} is approximately correct since large C-space distances are used to determine the partitioning and these are likely to be large W-space distances as well. Interesting questions arise for the analysis of the new scheme (relation between W-space and C-space distances) and its performance when the sample to be clustered contains noisy data.

Acknowledgment

This paper was inspired from work done in the context of a collaboration between Pfizer Central Research at Stanford University. The authors would like to thank D. Halperin, who initially proposed the idea of clustering using different metrics, P. Finn, J.-C. Latombe, S. Venkatasubramanian for discussions and help in various stages of this project. B. Cook was partially supported from the Keck Center for Molecular Biology at Rice University under BIR 94-13229. L. Kavraki's research is supported by startup funds from Rice University and NSF CAREER Award IRI 9702288. Rajeev Motwani's research is partially supported by an Alfred P. Sloan Research Fellowship, an IBM Faculty Partnership Award, an ARO MURI Grant DAAH04-96-1-0007, and NSF Young Investigator Award CCR-9357849, with matching funds from IBM, Mitsubishi, Schlumberger Foundation, Shell Foundation, and Xerox Corporation.

References

[1] K. S. Arun, T. Huang, and S. D. Blostein. Least-squares fitting of two 3-d point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):698–700, 1987.

[2] R. Duda and E. Hart. *Pattern Classification and Scene Analysis*. John Wiley and Sons, New York, 1973.

[3] P. Finn, L. Kavraki, J.-C. Latombe, R. Motwani, C. Shelton, S. Venkatasubramanian, and A. Yao. Rapid: Randomized pharmacophore identification. In *Proceedings of the International Symposium on Computational Geometry*, pages 324–333, Nice, France, 1997.

[4] T. F. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38:293–306, 1985.

[5] S. Kearsley, D. Underwood, R. Sheridan, and M. Miller. Flexibases: A way to enhance the use of molecular docking methods. *Journal of Computer-Aided Molecular Design*, 8:565–582, 1994.

Molecule	Atoms	Rotatable Bonds	W-space clustering (sec)	C-space clustering (sec)
1tlp	69	10	239.7	11.5
4tmn	70	15	241.6	15.8
5tmn	64	13	229.4	13.9
ace16	32	6	144.0	7.8

Table 1: Running times for the Gonzales algorithm with W-space and C-space distance measures (60 clusters).

Molecule	mixed 5-12 (sec)	mixed 10-6 (sec)	mixed 12-5 (sec)
1tlp	55.8	30.6	21.4
4tmn	55.3	30.8	21.7
5tmn	52.4	29.1	20.8
ace16	31.3	16.9	11.4

Table 2: Running times for mixed clustering. C-space clustering is followed by W-space clustering (60 clusters total).

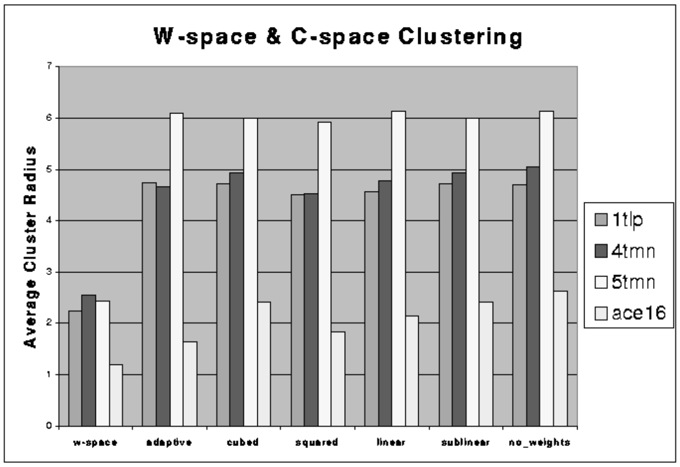

Figure 1: Comparison of W-space and C-space clustering using different weighting schemes.

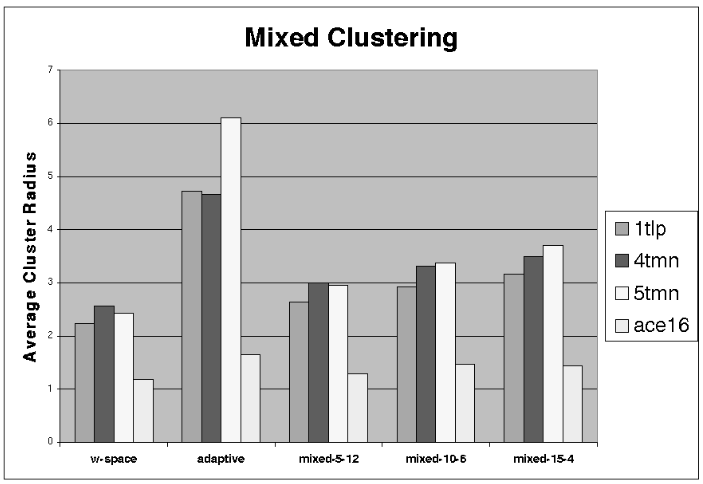

Figure 2: Comparison of W-space, C-space, and mixed clustering.

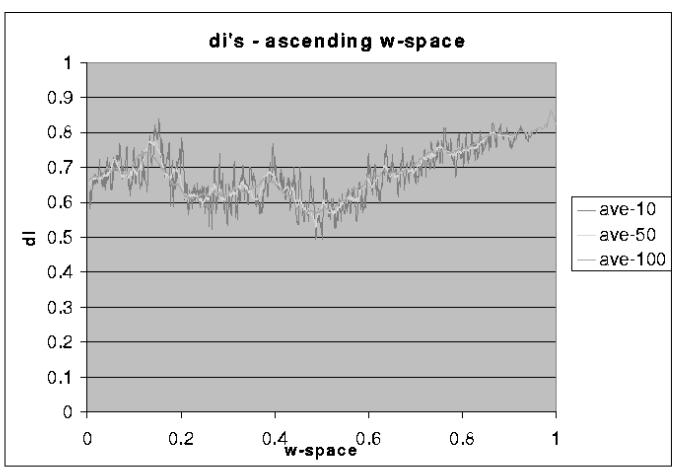

Figure 3: Relation of W-space l_{lrmsd} distance and C-space distance l_1^{adapt} for ace16.